PeopleSoft®

EnterpriseOne B73.3.1 System Administration PeopleBook

June 1999

J.D. Edwards World Source Company
One Technology Way
Denver, CO 80237

SKU B7331CEASY

Table of Contents

Overview to System Administration

The guides that comprise the *Configuration Planning and Setup* suite are designed for use by Configurable Networking Computing (CNC) specialists, OneWorld system administrators, and network/server administrators. The assumption throughout these guides is that the initial OneWorld installation is complete and the standard data sources, path codes, and environments are defined. These guides tells you how to make changes or additions to the configuration setup after the initial installation.

The *Configuration Planning and Setup* suite consists of the following guides:

- *Configurable Network Computing Implementation*. This guide is geared primarily towards CNC specialists and contains the following topics:
    - Understanding middleware and verifying that you have specified the correct middleware for your servers
    - Understanding data sources and verifying that the necessary ones have been created
    - Understanding and creating path codes and environments
    - Working with the Object Configuration Manager
    - Understanding the different modes of processing
    - Understanding a typical OneWorld customer configuration
- *System Administration*. This guide is written mainly for OneWorld system administrators and contains the following topics:
    - Understanding and setting up data replication
    - Setting up printers
    - Using the Work with Servers program
    - Setting up user profiles
    - Setting up OneWorld security
    - Understanding and working with data dictionary administration
    - Understanding vocabulary overrides
    - Understanding transaction processing
    - Working with media objects and imaging
    - Using the universal table browser
    - Understanding OneWorld naming conventions
    - Understanding the jde.ini file

- *Package Management*. This guide is for OneWorld system administrators and others who manage custom modifications to the OneWorld environments. *Package Management* contains the following topics:
    - Package management planning and setup
    - OneWorld modification rules
    - Object management
    - Building packages
    - Deploying packages
    - Server packages
    - Multitier deployment
- *Server and Workstation Administration*. This guide is written primarily for network administrators and contains the following topics:
    - Understanding Snapshot (multiclient installer)
    - Server administration
    - Troubleshooting the workstation
    - Troubleshooting the server

Although every attempt has been made to organize the information in the *Configuration Planning and Setup* guides according to related tasks, a CNC specialist, OneWorld administrator, or network administrator might find that the information needed to perform the duties of that position is described in more than one guide. For example, the person who is responsible for setting up path codes, environments, and data sources (described in the *Configurable Network Computing Implementation Guide*) might also be responsible for building and deploying packages (described in the *Package Management Guide*).

The *Configuration Planning and Setup* suite is the central location for all CNC-related tasks except:

- Initial installation of OneWorld. See the *OneWorld Installation Guide.*
- OneWorld upgrade and cumulative updates. See the *OneWorld Upgrade Guide.*
- Network infrastructure and third-party software setup and maintenance. This information is provided by the applicable software or hardware vendor. J.D. Edwards does not provide documentation.

You do not need a complete understanding of the installation process to perform configuration planning and setup tasks. However, in order to use the *Configuration Planning and Setup* guides it is important that you understand what the installation accomplishes. This guide, *Configurable Network Computing Implementation*, contains a section that provides complete overview information of the installation process.

Understanding OneWorld Roles

The OneWorld implementation methodology defines specific roles:

- CNC consultant and CNC administrator
- Custom solution consultant and application developer
- Application consultants and application project leaders
- Hardware, network, and third-party software consultants and administrators

Each of these roles are performed by both a consultant and a customer. After implementation, the role of the consultant is diminished. Therefore it is critical that the customer ensures that adequate training occurs for each of the roles to be assumed by their personnel.

CNC Consultant and CNC Administrator

The CNC consultant and CNC administrator are involved with the installation of OneWorld and the setup of environments, users, security, distributed processing, and data replication. They are also responsible for the setup of version control and testing of various CNC configurations. The CNC consultant and CNC administrator control the deployment of OneWorld software throughout the company.

Custom Solution Consultant and Application Developers

OneWorld custom solution consultants resolve business issues by developing applications. Their primary responsibilities include designing the modifications with upgrades in mind, and developing, testing, and introducing the customized software. While the CNC administrator performs the version control functions that build and deploy software, the customer solution consultant must help develop the internal procedures for application development cycle for your business.

Application Consultants and Application Project Leaders

After OneWorld is installed, configured, and rolled out, the application consultants will continue in their role as product experts. Although application consultants do not implement the CNC configurations, they must understand how OneWorld handles distributed processing, data replication, environments, and so on, because these application issues influence the CNC decisions. In addition, application consultants must become very good at troubleshooting potential problems.

Hardware, Network, and Third-Party Software Consultants and Administrators

Implementing OneWorld includes many tasks that are outside the scope of J.D. Edwards services. Third-party consultants provide these services as well as supplementing our staff as CNC consultants, network architects, custom modification consultants, and so on.

Understanding The System Administration Guide

The *System Administration Guide* focuses primarily on how to:

- Use the flexibility of the CNC architecture to optimize the OneWorld installation to your enterprise.
- Extend an initial installation (CRP) to meet the practical requirements.
- Recognize, address, and solve the day-to-day issues that arise in a dynamic enterprise.

Although this guide is aimed primarily at OneWorld system administrators, those with other job functions may find the information useful or essential to their positions as well.

This guide includes the following topics:

- ❑ Middleware
- ❑ Data Replication
- ❑ Data Dictionary Administration
- ❑ Printing OneWorld Reports
- ❑ The Work With Servers Program
- ❑ User Profiles
- ❑ Security
- ❑ Vocabulary Overrides
- ❑ Transaction Processing
- ❑ Scheduler
- ❑ Media Objects and Imaging
- ❑ Universal Table Browser

- OneWorld Naming Conventions
- The jde.ini File

Middleware

In a client/server environment, applications must communicate across different platforms. These platforms can have different communication protocols, database management systems, and hardware operating systems. For clients to talk to servers and servers to talk to other servers, a mechanism must exist that can bridge multiprotocol and multivendor issues. This mechanism is a layer of software called middleware, which resides between the operating system and the business applications. It is important to have an application architecture that is based on a single, consistent middleware strategy.

J.D. Edwards provides the following types of middleware:

JDENet Communication Middleware	Performs the connections from workstation to server and server to server, and sends messages for distributed requests. It is a peer-to-peer, message-based, socket-based, multiprocess communication middleware solution.
JDEBase Database Middleware	Provides platform-independent application programming interfaces (APIs) for multidatabase access. These APIs are used in two ways: • By OneWorld applications that dynamically generate platform-specific Structured Query Language (SQL), depending on the data source request. • As open APIs for writing advanced business functions in the C language. OneWorld uses these APIs to dynamically generate platform-specific SQL statements. JDEBase also provides workstation-to-server and server-to-server database access. To accomplish this, OneWorld is integrated with a variety of third-party database drivers, such as IBM Client Access/400 database software, Open Database Connectivity (ODBC), and Oracle Connectivity Interface (OCI). The exact driver requirements are listed in the *Hardware and Software Requirements* sections of the platform-specific OneWorld installation guides.

By the end of this section you should:

- ❑ Understand JDENet communication middleware
- ❑ Understand JDEBase database middleware

Understanding JDENet Communication Middleware

For two computers to communicate with each other, they must share a communication protocol, or set of protocols. A communication protocol is a formal set of rules that specifies the format and relationship for exchanging data among different devices. The following are the protocol layers with which the communication middleware is concerned:

Network layer	The network layer handles addressing and routing information, error checking, and retransmission requests.
Transport layer	The transport layer provides connection-oriented data-delivery services across networks. These protocols provide end-to-end data exchanges in which systems maintain a session or connection with each other for the reliable sequenced exchange of data. OneWorld supports the Transmission Control Protocol/Internet Protocol (TCP/IP) protocol suite.
Application layer	The application layer provides application-to-application interaction and data exchange. JDENet is the application layer communication middleware.

JDENet Communication Middleware

JDENet is the J.D. Edwards proprietary communications middleware package that provides for server-to-server and client-to-server communication. Features include:

- Socket-based communication
- Message-based communication
- OneWorld process-based design
- OneWorld dedicated process design
- jde.ini settings
- Load balancing design
- Server Administration Workbench (SAW)

JDENet provides client-to-server and server-to-server communications. With JDENet, any machine on a peer-to-peer network can function as a client or a server at any given time. A machine on a peer-to-peer network functions as a client when it initiates a request and as a server when it responds and provides a service to a client“s request. Typically, in OneWorld, the majority of requests originate from an end user’s workstation. However, servers can also initiate requests from other servers. An example of this is server-to-server push replication.

With JDENet, communications among the machines on the network occur through an exchange of messages. Examples of JDENet messages include requests and responses for business functions, batch jobs, data replication, and OneWorld signon security. For example, a machine can send a message requesting a remote business function call. The target machine responds by executing the business function locally and returning the results to the requesting machine.

Requests can be synchronous or asynchronous. A synchronous message, such as a business function call, requires the client to wait for the server to complete the request. An asynchronous request, such as a batch process, allows the client to continue with another task while OneWorld processes the request. In some circumstances, business functions can also be called asynchronously.

Socket-Based Communication

Sockets provide a duplex communication channel between one or more systems. JDENet uses stream sockets to provide end-to-end communications. Sockets guarantee that the data arrives intact.

Message-Based Communication

Message-based communication means applications send service requests for logic or data in the form of messages that are received and stored in a queue for processing. The middleware handles message transmission, which allows the workstation application to process other tasks. Without messaging services to handle these jobs, the application must wait until the request is handled and the results returned.

Messaging is most appropriate for applications that are event driven. It is the opposite of remote procedure calls (RPC), which are synchronous. The message packaging and handshaking of JDENet ensures that the message transmission is complete.

OneWorld Process-Based Design

In a Windows environment, although client workstations can have more than one copy of OneWorld installed, only one instance of the OneWorld client can be running on that machine at any one time. When communicating with a OneWorld server, the OneWorld client uses dynamic-link libraries (DLLs) to run an internal network process. This process is the client-side portion of the JDENet communications middleware and is also referred to as a net process.

On the OneWorld server-side, the JDENet process communicates with the OneWorld client and routes request messages to appropriate dedicated processes. In turn, the dedicated processes route work to the appropriate server platform-specific logic processes, such as DLLs, shared libraries, and job queues. A server can have multiple OneWorld main processes, multiple OneWorld dedicated processes, multiple DLLs, shared libraries, and job queues.

The advantage of this client/server architecture is that multiple workstations can make requests to the same server at one time. You can control the number of workstations that can make and maintain a session connection to a main server process. Further, you can define the total number of dedicated processes (and the number of each type) that OneWorld uses to process specific types of workstation requests.

There is a relationship between the net processes, the dedicated processes, and the logic processes. This relationship is specifically defined by the jde.ini file on the enterprise server. Every enterprise server must have at least one OneWorld network process, referred to as a JDENET_n job. This job handles network connections and traffic for OneWorld.

Network Processes

The OneWorld network process is called a JDENET_n job. The purpose of this constantly-running job is to handle OneWorld communication messages by monitoring the network for incoming messages to OneWorld and also monitoring OneWorld for outgoing messages to the network. As defined in the OneWorld initialization file (jde.ini) for each server, there can be multiple JDENET_n processes. Regardless of the number of JDENET_n processes, the initial JDENET_n process is the Master Listener. The Master Listener is the only process that directly monitors the network for OneWorld messages and passes those messages to additional JDENET_n processes, if defined, or to other jobs for processing as required.

If multiple JDENET_n jobs are specified, OneWorld starts the jobs as required, allocating a job to each incoming request. When the maximum number of JDENET_n processes are started, OneWorld automatically alternates assigning the incoming requests between the currently running JDENET_n jobs until the maximum number of connections is reached. This provides a degree of load balancing between OneWorld network processes. If, on a given server, the maximum number of connections for the JDENET_n job is met, a workstation or server cannot initiate an additional OneWorld session on that server until an existing session connection is ended. This is because, by design, all connections to JDENET_n persist for the duration of a OneWorld session.

For example, assume the server jde.ini files specifies that four JDENET_n processes are allowed. The first incoming JDENET_n request is handled by the Master Listener, which is the initial JDENET_n process that is run at server start-up. When a second incoming message to JDENET_n is received, the Master Listener receives the request and assigns it to a second JDENET_n process, which it then starts. These assignments persist for the duration of the session between the requesting device and this server. The same process occurs for the third and fourth JDENET_n requests. When the fifth request is received, that request is assigned to the first JDENET_n process, and the cycle continues.

The responsibility of the JDENET_n process is to handle the network layer of communication. That is, it does not perform any actual logic processing but instead performs message handling. If the JDENET_n job determines that the incoming message is a request for logic processing, it routes the request to an appropriate JDENET_k job, called a OneWorld kernel process.

Kernel Processes

The responsibility of the JDENET_k job is to handle the two-way routing to and from the various logic processes while the JDENET_n job handles the return delivery to the appropriate machine. OneWorld uses a number of predefined and dedicated kernel types, which are essentially virtual servers. Each kernel type is responsible for a specific OneWorld process. See *OneWorld Dedicated Processes*.

OneWorld determines an appropriate JDENET_k job based on message identifiers generated by OneWorld. The JDENET_k job is the OneWorld process that provides the link between the JDENET_n job and the appropriate platform-specific processing job. The JDENET_k process is only applicable to servers.

Examples of logic processes include DLLs for Windows NT platforms, shared libraries (.sl) for UNIX platforms, and JDENet processes for AS/400 platforms.

OneWorld Dedicated Process Design

Because JDENet is a message-based architecture, you can label each message with a message type identifier. JDENet uses this identifier to route messages to a range of processes dedicated to handling requests that fall within a specified range. The following process types have been defined:

- Type 1. OneWorld internal and testing processes
- Type 2. OneWorld batch process (UBE) pass-through
- Type 3. Data replication requests
- Type 4. Security processes
- Type 5. Transaction manager and lock manager
- Type 6. Remote Master Business Function (MBF)
- Type 7. JDBNET Server-to-Server
- Type 8. Package Install
- Type 9. Server Administration Workbench (SAW)
- Type 10. Scheduler
- Type 11. Package Build

Additionally, dedicated processes also allow third parties to hook into and interface with the JDENet architecture. That is, third parties can write their own platform-specific logic processes that conform to OneWorld published APIs. Third-party developers should design these programs (for example, .dll, .sl, or job queue) to process only a specific and currently unused range of OneWorld messages. Complete requirements for third-party applications are found in the *API Online Help,* which is in OneWorld online help.

jde.ini Settings

There are a variety of settings that control the definition and function of the JDENET_n and JDENET_k processes. These settings are contained with the following main sections of the server-specific jde.ini:

- [JDENET]
- [JDENET_KERNEL_DEF1]
- [JDENET_KERNEL_DEF2]
- [JDENET_KERNEL_DEF3]
- [JDENET_KERNEL_DEF4]
- [JDENET_KERNEL_DEF5]
- [JDENET_KERNEL_DEF6]
- [JDENET_KERNEL_DEF7]
- [JDENET_KERNEL_DEF8]
- [JDENET_KERNEL_DEF9]
- [JDENET_KERNEL_DEF10]
- [JDENET_KERNEL_DEF11]

JDENET_n Settings

The characteristics of the JDENET_n jobs are controlled by the following jde.ini settings. These settings are fully described in the following section. For the network processes, there are four user-definable values (shown in *italic type*) in the [JDENET] section of the jde.ini.

```
[JDENET]
netPgmName=network program name
maxNetProcesses=number of jdenet_n processes
maxNetConnections=number of connections per jdenet_n process
newProcessThresholdConnects=number of connections
```

JDENET_k Settings

For servers only, the characteristics of the JDENET_k jobs are controlled by the following jde.ini settings. These settings are fully described in the following section. For the kernel process, there are three user definable settings in the [JDENET] section of the jde.ini. There are also a number of individual sections in the jde.ini that are used to define the kernel types. In these kernel type definition sections, there are only three user definable values (shown in *italic type*) for each kernel type.

```
[JDENET]
KrnlPgmName=kernel program name
maxKernelProcesses=number of jdenet_k processes
maxKernelRanges=number of kernel ranges
```

```
[JDENET_KERNEL_DEF1]
dispatchDLLName=platform-specific program name
dispatchDLLFunction=JDENET_DispatchMessage
maxNumberOfProcesses=value
beginningMsgTypeRange=0
endingMsgTypeRange=255
newProcessThresholdRequests=value
```

```
[JDENET_KERNEL_DEF2]
dispatchDLLName=platform-specific program name
dispatchDLLFunction=JDEK_DispatchUBEMessage
maxNumberOfProcesses=value
beginningMsgTypeRange=256
endingMsgTypeRange=511
newProcessThresholdRequests=value
```

```
[JDENET_KERNEL_DEF3]
dispatchDLLName=platform-specific program name
dispatchDLLFunction=DispatchRepMessage
maxNumberOfProcesses=value
beginningMsgTypeRange=512
endingMsgTypeRange=550
newProcessThresholdRequests=value
```

```
[JDENET_KERNEL_DEF4]
dispatchDLLName=platform-specific program name
dispatchDLLFunction=JDEK_DispatchSecurity
maxNumberOfProcesses=value
beginningMsgTypeRange=551
endingMsgTypeRange=580
newProcessThresholdRequests=value
```

```
[JDENET_KERNEL_DEF5]
dispatchDLLName=platform-specific program name
dispatchDLLFunction=TM_DispatchTransactionManager
maxNumberOfProcesses=value
beginningMsgTypeRange=601
endingMsgTypeRange=650
newProcessThresholdRequests=value
```

```
[JDENET_KERNEL_DEF6]
dispatchDLLName=platform-specific program name
dispatchDLLFunction=JDEK_DispatchCallObjectMessage
maxNumberOfProcesses=value
beginningMsgTypeRange=901
endingMsgTypeRange=1156
newProcessThresholdRequests=value
```

```
[JDENET_KERNEL_DEF7]
dispatchDLLName=platform-specific program name
dispatchDLLFunction=JDEK_DispatchJDBNETMessage
maxNumberOfProcesses=value
beginningMsgTypeRange=1201
endingMsgTypeRange=1456
newProcessThresholdRequests=value
```

```
[JDENET_KERNEL_DEF8]
dispatchDLLName=platform-specific program name
dispatchDLLFunction=JDEK_DispatchPgkInstallMessage
maxNumberOfProcesses=value
beginningMsgTypeRange=1501
endingMsgTypeRange=1756
newProcessThresholdRequests=value
```

```
[JDENET_KERNEL_DEF9]
dispatchDLLName=platform-specific program name
dispatchDLLFunction=JDEK_DispatchSAWMessage
maxNumberOfProcesses=value
beginningMsgTypeRange=2001
endingMsgTypeRange=2256
newProcessThresholdRequests=value
```

```
[JDENET_KERNEL_DEF10]
dispatchDLLName=platform-specific program name
dispatchDLLFunction=JDEK_DispatchScheduler
maxNumberOfProcesses=value
beginningMsgTypeRange=2501
endingMsgTypeRange=2756
newProcessThresholdRequests=value
```

```
[JDENET_KERNEL_DEF11]
dispatchDLLName=platform-specific program name
dispatchDLLFunction=JDEK_DispatchPkgBuildMessage
maxNumberOfProcesses=value
beginningMsgTypeRange=3001
endingMsgTypeRange=3256
newProcessThresholdRequests=value
```

Purpose of [JDENET] jde.ini Settings

This section presents a table that defines the only user-definable settings in the [JDENET] section of the jde.ini. These sections contain important settings that define the JDENET_n and JDENET_k processes. These settings are listed in the order that they typically appear in the jde.ini file. Where applicable, associated settings are cross-referenced. A complete description of every setting within the jde.ini file is contained in *The jde.ini File* section of this guide.

[JDENET] - User-definable Settings

Setting	Value
netPgmName	Defines the name and, if applicable, the full path of the JDENET_n job. Typical Value (non-AS/400 platforms): jdenet_n Typical Value (AS/400 platforms): /QSYS.LIB/B733NET.LIB/JDENET_N.PGM
krnlPgmName	Defines the name and, if applicable, the full path of the JDENET_k job. Typical Value (non-AS/400 platforms): jdenet_k Typical Value: (AS/400 platforms): /QSYS.LIB/B733NET.LIB/JDENET_KPGM
maxNetProcesses	Defines the maximum number of JDENET_n jobs that can run on this OneWorld server. This value can be from 1 to *n*. You can increase this value for a server that is expecting heavy JDENET message flow. If multiple JDENET_n jobs are specified, OneWorld starts the jobs as required allocating a job to each request. When the maximum number of JDENET_n processes are started, OneWorld automatically alternates between the currently running JDENET_n jobs until the maximum number of connections is reached. This provides a degree of load balancing between OneWorld network processes. If, on a given server, the maximum number of connections for the JDENET_n job is met, a client or server cannot initiate an additional OneWorld session on that server until an existing session connection is ended. This is because, by design, all connections to JDENET_n are persistent for the duration of a OneWorld session. Typical Value (all platforms): 1 Related Value: maxNetConnections

Setting	Value
maxNet Connections	Defines the maximum number of connections that can connect to a single JDENET_n job. You can increase this value for a server that is expecting higher connection volumes. This is the product of the value specified in maxNetProcesses and maxNetConnections. For example, if you specify 3 JDENET_n processes and 800 maxNetconnections, a total of 2400 connections are enabled for this server. Typical Value (all platforms except Windows NT): 800. Typical Value (Windows NT platforms): 400 Related Value: maxNetProcesses
maxKernel Processes	Defines the maximum number of processes that the JDENET_k job can call. This is the product of all definition types (defined by maxKernelRanges) and the number of each kernel type (defined by maxNumberOfProcesses for each kernel type). For example, if you have 11 ranges of kernel types, and you specify each kernel can run two instances, you must set the value of this setting to 22. Typical Value (all server platforms): 11 Related Value: maxNumberOfProcesses in each [JDENET_KERNEL_DEF*x*] section
maxKernelRanges	Defines the maximum number of kernel types that the JDENET_k job can call. A kernel type is defined by a specific definition labeled in the jde.ini as [JDENET_KERNEL_DEF*x*] where *x* is a valid number associated with a dedicated kernel process type. OneWorld preassigns each kernel type to handle a specific range of messages. Typical Value (all server platforms): 11 Related Values: See following tables for each [JDENET_KERNEL_DEF*x*] range
newProcess Threshold Connects	Defines the threshold at which OneWorld begins a new JDENET_n process. A value of 0 specifies that the Master Listener JDENET_n process will always start a new JDENET_n process until the maxNetProcesses is met. A non zero value specifies that the Master Listener JDENET_n process will start a new JDENET_n process only when all existing JDENET_n processes have outstanding connections greater than the value specified in newProcessThresholdConnects. For example, if the default value of 0 is used, and maxNetProcesses is set to a value greater than one, the Master Listener JDENET_n process starts a new JDENET_n process for each request up to the maximum specified by the value of maxNetProcesses. Also for example, if a value of 5 is used, and maxNetProcesses is set to a value greater than one, the Master Listener JDENET_n process starts a new JDENET_n process only when each JDENET_n process is already assigned to handle 5 requests. Typical Value (all platforms): 0 Related Values: maxNetProcesses

Purpose of [JDENET_KERNEL_DEF*x*] jde.ini Settings

This section presents a table that defines the [JDENET_KERNEL_DEF*x*] section of the jde.ini. Because each dedicated kernel requires a separate definition, there are 11 uniquely-numbered kernel definition sections. Depending on the platform, the kernel code might be different according to the specialized uses of the kernels. Also, each kernel is defined to handle a specific range of messages.

A complete description of each setting in the jde.ini file is contained in *The jde.ini File* section of this guide. The following table describes the user-definable settingss for the JDENET_k processes.

[JDENET_KERNEL_DEF*x*] - User-definable Settings

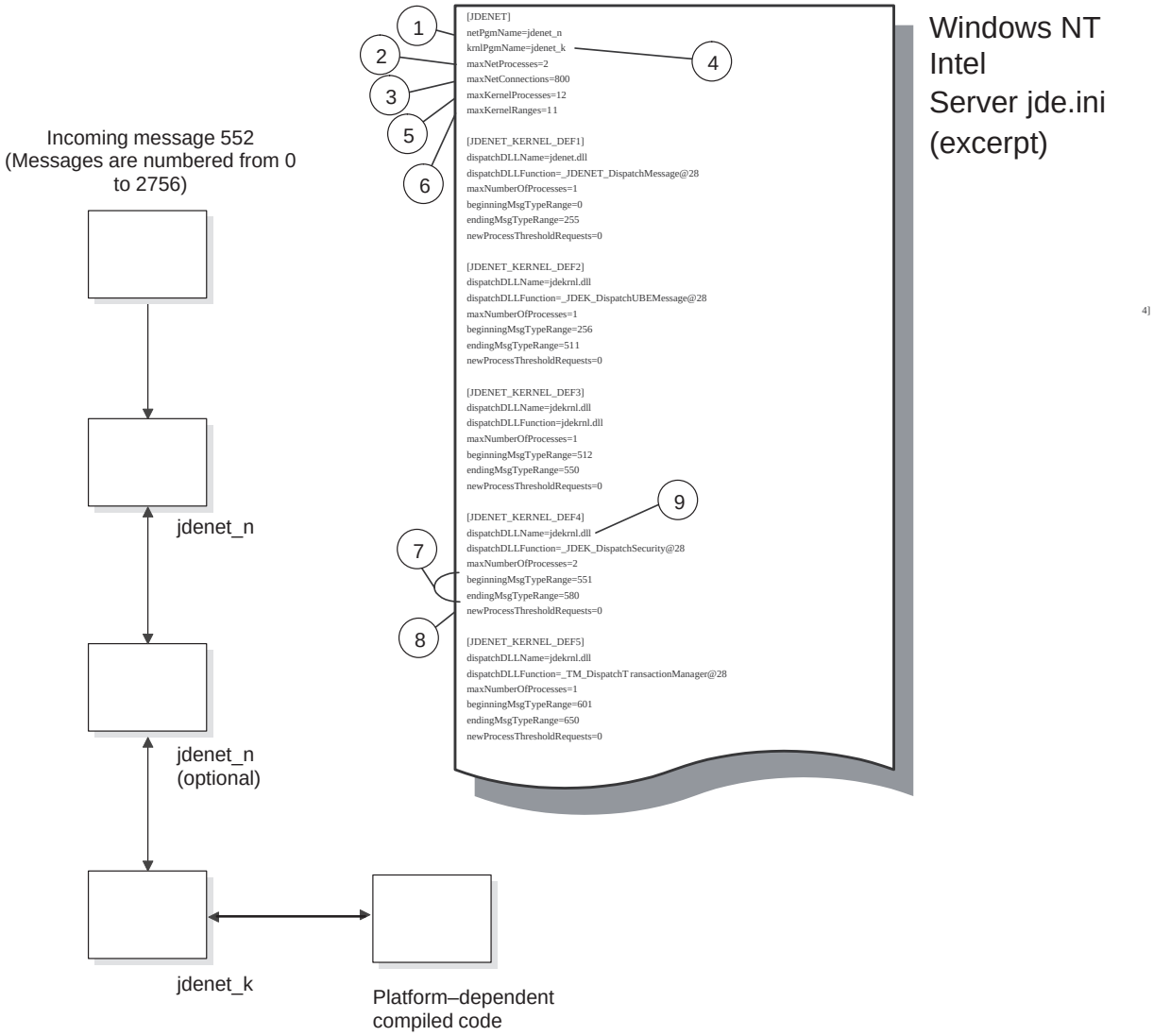
Setting	Value
dispatchDLLName	Defines the name of the JDENET_k job running on a OneWorld server. The program name varies according to server platform and kernel function. OS/400 on AS/400 - Typical Values: DEF1: JDENET DEF2: JDEKRNL DEF3: JDEKRNL DEF4: JDEKRNL DEF5: JDEKRNL DEF6: JDEKRNL DEF7: JDEKRNL DEF8: JDEKRNL DEF9: JDESAW DEF10: JDEKRNL DEF11: JDEKRNL Windows NT on Intel/Compaq AlphaServer - Typical Values: DEF1: jdenet.dll DEF2: jdekrnl.dll DEF3: jdekrnl.dll DEF4: jdekrnl.dll DEF5: jdekrnl.dll DEF6: jdewapi.dll DEF7: jdekrnl.dll DEF8: jdekrnl.dll DEF9: jdesaw.dll DEF10: jdekrnl.dll DEF11: jdekrnl.dll

Setting	Value
	UNIX (HPUX) on HP9000 - Typical Values: DEF1: libjdenet.sl DEF2: libjdeknet.sl DEF3: libjderepl.sl DEF4: libjdeknet.sl DEF5: libtransmon.sl DEF6: libjdewapi.sl DEF7: libjdeknet.sl DEF8: libjdeknet.sl DEF9: libjdeknet.sl DEF8: libjdeknet.sl DEF9: libjdesaw.sl DEF10: libjdeschr.sl DEF11: libjdeknet.sl UNIX (AIX) on RS/6000 - Typical Values: DEF1: libjdenet.so DEF2: libjdekrnl.so DEF3: libjdekrnl.so DEF4: libjdekrnl.so DEF5: libjdekrnl.so DEF6: libjdewapi.so DEF7: libjdekrnl.so DEF8: libjdekrnl.so DEF9: libjdesaw.so DEF10: libjdekrnl.so DEF11: libjdekrnl.so
maxNumberOf Processes	Defines the maximum number of processes that can run for this kernel definition type. You can increase this value for a server that is expecting higher connection volumes for a particular kernel function. For example, if your server is dedicated to a particular function like package builds, you could increase this value in the [JDENET_KERNEL_DEF9] section. If you increase this value, you will also need to increase the value of the maxKernelProcesses in the [JDENET] section. Typical Value (all platforms): 1 Related Value: [JDENET] maxKernelProcesses
newProcess ThresholdRequests	Defines the threshold at which OneWorld begins a new JDENET_k process for the specified kernel process. A value of 0 specifies that JDENET_k will always start a new instance of a specific JDENET_k process until the value by that processes' maxNumberOfProcesses setting is met. A nonzero value specifies that OneWorld will start a new JDENET_k process only when all existing instances of this JDENET_k processes have outstanding connections greater than the value specified in that processes' maxNumberOfProcesses setting. Typical Value (all platforms): 0 Related Values: maxNumberOfProcesses

JDENet Middleware Process Flow

The following diagram shows the process flow from JDENET_n to JDENET_k as it might be set in a typical environment.

Example: JDENet Middleware Process Flow



As numbered on the preceding diagram, the following steps occur:

1. The Master Listener JDENET_n process accepts the message request. In this example, the name of the network job is jdenet_n as defined by the following setting:

```
[JDENET]
krnlPgmName=jdenet_n
```

2. The Master Listener JDENET_n process reads the jde.ini and determines if multiple network jobs are specified. In this example, multiple network jobs are allowed as defined by the following setting:

```
[JDENET]
maxNetProcesses=2
```

3. The Master Listener JDENET_n process reads the jde.ini and determines if the maximum network connections have been established for this server. The number of network connections is the product of the values specified for the maximum network processes and the maximum network connections. In this example, the JDENET_n process checks to see if less than 1,600 connections are established as defined by reading the following settings:

```
[JDENET]
maxNetProcesses=2
maxNetConnections=800
```

   If required and allowed, the JDENET_n process starts another JDENET_n process.

4. The JDENET_n process accepts the message and determines that it is a logic request. It reads the jde.ini and determines the name of the kernel job. In this example, the name of the kernel job is jdenet_k as defined by the following setting:

```
[JDENET]
krnlPgmName=jdenet_k
```

5. The JDENET_n process reads the jde.ini and determines if the maximum number of kernel processes has been met. In this example, that value is defined by the following setting:

```
[JDENET]
maxKernelProcesses=11
```

6. The JDENET_n process reads the jde.ini and determines if the maximum number of kernel ranges has been met. If neither the maximum number of processes or maximum number of ranges has been exceeded, the request is passed to the JDENET_k job. In this example, the value of the maximum number of ranges is specified by the following setting:

```
[JDENET]
maxKernelRanges=11
```

7. Based on the message type number, the JDENET_k job determines the kernel definition type. In this case, the message number is 552. By definition this message number falls into the range defined as kernel type definition 4, which is the Security Server. In this example, the type 4 kernel is defined by the following settings:

```
[JDENET_KERNEL_DEF4]
dispatchDLLName=jdekrnl.dll
dispatchDLLFunction=_JDEK_DispatchSecurity@28
maxNumberOfProcesses=2
beginningMsgTypeRange=551
endingMsgTypeRange=580
newProcessThresholdRequests=0
```

8. The kernel job reads the jde.ini to determine the maximum number of logic processes that are allowed for Type 4 kernel processes. In this example, the value for the maximum number of Type 4 kernel processes is specified by the following setting:

```
[JDENET_KERNEL_DEF4]
maxNumberOfProcesses=2
```

   If required and allowed, the JDENET_k process starts another platform-specific logic process.

9. The JDENET_k job reads the jde.ini to determine which platform-specific process to call for the actual logic processing. In this example, the server platform is an Windows NT with an Intel processor. Therefore, the platform-specific process is a dynamic link library (DLL) named jdekrnl.dll. If that kernel job is not already started, the JDENET_k process call causes it to start. If the job is already started, the request is queued to the next available logic process identified to run this message type request. In this example, the platform-specific logic process is identified by the following setting:

```
[JDENET_KERNEL_DEF4]
dispatchDLLName=jdekrnl.dll
```

Load Balancing Design

Because a server can have multiple net processes (JDENet) and multiple dedicated internal processes, you must establish a configuration that provides an optimal number of processes on a server. This system administration concept is called load balancing. Load balancing allows OneWorld to maximize its overall performance by using the available processing power of the platform on which it is running.

The load balancing configuration is controlled by various interrelated runtime settings in the jde.ini file on the server. These settings are in the [JDENET] and applicable [JDENET_KERNEL_DEF*x*] sections. All relevant settings are listed and described in the preceding section. You can control two types of processes:

- Network jobs (JDENET_n)
- Dedicated kernel process jobs (JDENET_k)

You can set the number of network jobs to control how many users are simultaneously connected to each net job. For example, if you have 200 users and you want a net process for each 100 users, you can specify two net processes. The controlling setting is:

```
[JDENET]
maxNetProcesses=variable
```

You can set the number of dedicated process jobs to control how many instances of each dedicated process type are running. For example, if you think that increasing the number of Type 2 dedicated processes might improve the processing performance for batch process (UBE) requests, you can increase the value. The controlling setting is contained in the definition for each dedicated process type (the *x* variable value in JDE_KERNEL_DEF*x*, where allowable values are from 1 to 11):

```
[JDENET_KERNEL_DEFx]
maxNumberOfProcesses=variable
```

If you increase the number of individual dedicated processes, you must also increase the value of maxKernelProcesses in the [JDENET] section. For example, if maxKernelProcesses=11:

```
[JDENET]
maxKernelProcesses=11
```

and you increase the number of Type 2 processes from 1 to 2:

```
[JDENET_KERNEL_DEF2]
maxNumberOfProcesses=2
```

you must increase the maxKernelProcesses value from 11 to 12:

```
[JDENET]
maxKernelProcesses=12
```

Server Administration Workbench (SAW)

You can use the OneWorld Server Administration Workbench (SAW) monitor programs to observe the number of OneWorld network connections and other load-based factors that might be used to evaluate the performance of server processes. This information is useful to make load balancing decisions, such as increasing the number of processes or network connections to increase performance. You can also use the monitoring programs to enable various logs and traces useful for troubleshooting purposes.

This SAW application provides the server administrator with vital statistics concerning the internal functions of OneWorld. The functionality of SAW includes embedded notification capabilities. Using configurable e-mail or Pager push delivery mechanisms, specified server administrators are notified:

- Every time a server is started or stopped
- Of initial and changed server status, such as up or down
- Of overloaded conditions when a process' configurable outstanding request threshold is reached

The SAW program also allows you to do the following:

- View the number of active JDENet and OneWorld processes running
- View incoming processing requests that are issued by a OneWorld process
- Shut down OneWorld running on the server
- View workstation and server connection information
- View the OneWorld workstation and server configuration settings
- Notify workstations if there is a server shut down
- Allow the collection and display of OneWorld message-flow statistics.

See Also

- *Server Administration Workbench (SAW)* in the *Server and Workstation Administration Guide*

Understanding JDEBase Database Middleware

Different database management systems (DBMS) have their own version of Structured Query Language (SQL). A database middleware layer allows a common interface to interpret the various versions of SQL. J.D. Edwards has a database middleware product called JDEBase, which is a common set of application programming interfaces (APIs) that programmers can call to request data and perform data manipulation logic. JDEBase interprets the generic APIs and converts the SQL into the appropriate statements for OneWorld to access the database.

Multiple databases in a distributed environment require a monitoring program to ensure database integrity. This monitoring program is referred to as a transaction monitor or a lock manager. The JDEBase database middleware has an embedded lock manager.

JDEBase Database Middleware

JDEBase provides a set of APIs to the developer and a set of translation programs to OneWorld. The translation programs are embedded in the data source definitions. For example, suppose a data request for address master is made. The Object Configuration Manager (OCM) determines which data source contains the requested table. The data source master provides the database information. This includes the required driver, which is loaded to translate the request into the appropriate SQL statements.

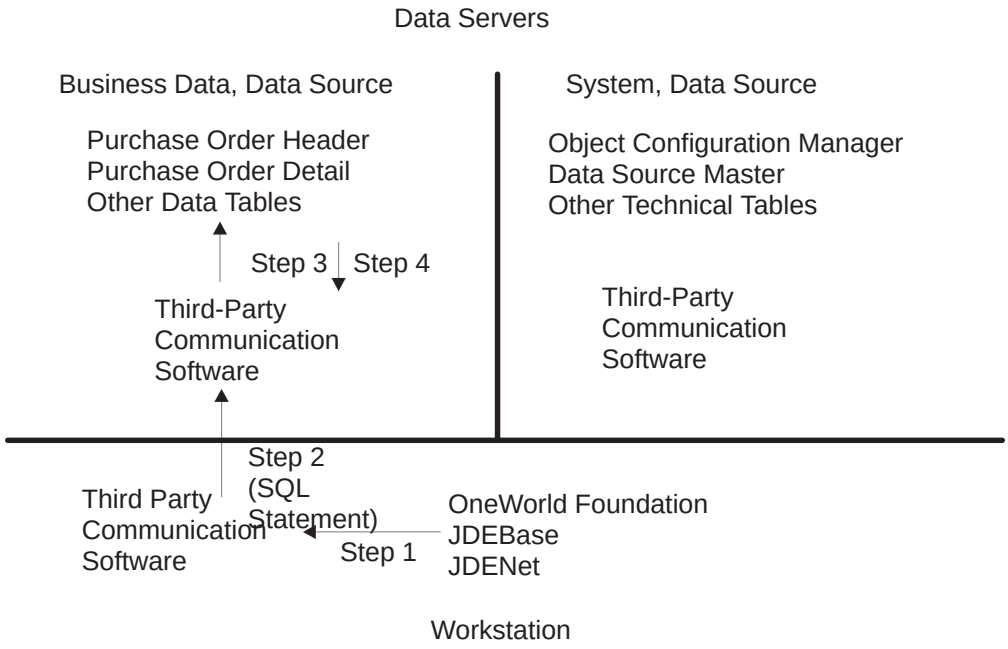
The following functions are provided by JDEBase:

- Ability to insulate developers from platform-specific SQL coding
- Rapid development of native drivers
- Server-to-server communication
- Transaction control using a lock manager

Database Middleware Flow

The following figure illustrates the data-request process. For this example, the user enters search criteria for a Purchase Order and clicks Find.

Database Middleware Flow



OneWorld accommodates any number of data servers. This illustration shows only two.

1. OneWorld sends a data request to JDEBase on the workstation. The request includes the form's data structure and any values needed to locate the record.
2. JDEBase does the following:
   - Builds data structures from the application structures, creates the actual SQL statement or equivalent commands, and passes it to the third party communication software, which resides on the workstation. Examples of third-party communication software are IBM Client Access, Microsoft ODBC, or Oracle SQLNET.
   - Manages the physical connections to the database.
   - Manages optimal fetch algorithms.
   - Performs all binding.
   - Passes result codes back to the application. Errors are logged in the jde.log.
3. The third-party communication software on the workstation passes data to the third-party communication software on the server. The server then accesses the table and returns the data back to the third-party communication software.

4. Finally, the third-party communication software passes data back to the JDEBase and the OneWorld Foundation processes.

Data Replication

Data replication is the process of replicating (copying) data server-to-server or server-to-workstation. Data replication is sometimes an affective way to increase the performance of your network because you can replicate data to a location close to the processing logic. For example, replicating data to a departmental server greatly reduces overall network traffic for table validation. A reduction in network traffic is particularly important in a wide area network (WAN) environment.

This section defines the concepts of replicating data across your enterprise and how to use the Data Replication (P98DREP) application to set up your replicated data.

This section contains the following topics:

- Understanding data replication
- Setting up data replication
- Replicating Data Dictionary Changes

Understanding Data Replication

You can replicate data from a central location, such as an enterprise server, to workstations or servers to increase the performance of your network. For example, you can greatly reduce network traffic in a wide area network (WAN) environment by replicating data to a Workgroup Server. Data replication requires you to thoroughly plan how you want the configurable network computing environment to work. The following are decisions you must make:

- Which replication engine should you use?
- What level of support will you require to administer the chosen replication facilities?
- What tables should you replicate and where should the replicated tables reside?

The first decision usually involves whether to use the OneWorld replication facilities, the tools provided by the database vendor, or the tools provided by a third-party vendor. Each solution offers unique advantages. If you choose the replication tools in OneWorld or the replication tools of the database vendor, such as Oracle or SQL, you receive the benefit of having the replication tools integrated into the software or the database administration functions respectively. If you choose a third-party replication tool, you can meet cross-vendor replication needs.

Regardless of the data replication tools you choose, the issue of support is important to your solution. The allocation of adequate administrative resources to monitor data replication in any production environment is critical. You must preserve the integrity of replicated data and master error recovery methodologies.

Data replication provides many advantages but also adds many responsibilities. When you use replication facilities, you need to designate an administrator to regularly monitor the replication process to maintain data integrity. When you compare the advantages of data replication to the large amount of maintenance required, consider the following benefits:

- Data replication allows store and forward transactions. If you replicate all the data necessary for a user to enter transactions on a workstation, such as a laptop, then that user can enter transactions while the workstation is disconnected from the enterprise servers. See *Working with Store and Forward Processing* for information about using store and forward processing.

- Data replication improves network performance. You can improve performance by replicating static tables (such as the data dictionary, user defined codes, and menus) to the workstation. This reduces the strain on the servers by having less data retrieved, allowing the workstations to handle the table input/output.

Concepts of Data Replication

You should become familiar with the following concepts and terminology of data replication.

What is a Published Table and a Publisher Machine?

A published table is the central copy of a table that you want to replicate to other machines. A publisher is the server responsible for the published table. The Data Replication Publisher table (F98DRPUB), which resides in the system data source, identifies all of the published tables and the publishers associated with those tables in the enterprise. When you replicate header-detail type files, you should configure replication to work with the files as a set. Information that resides in the Data Replication Publisher table includes:

- Name of the published table
- Data source where the published table resides
- The name of the machine that serves as the publisher
- Enable/Disable flag

What is a Replicated Table and a Subscriber Machine?

A replicated table is the copy of the published table that resides on the subscriber. A subscriber is the machine, whether server or workstation, that you designate as responsible for the replicated copy of a published table. The Data Replication Subscribers table (F98DRSUB), which resides with the Data Replication Publisher table (F98DRPUB) in the system data source, identifies all of the subscriber machines for each published table. A published table can have multiple subscribers. Information that resides in the Data Replication Subscribers table includes:

- Name of the subscriber machine
- Name of the published table that you replicate on the subscriber
- Data source where the replicated table resides
- Replication type for the replicated table: push, pull, just-in-time replication, or none (for third party replication products)
- In Synch flag
- Enable/Disable flag

In special cases, the subscriber can be the same machine as the publisher. An example of such a case is when you configure a database-only workgroup server across a WAN. The enterprise server acts as both the publisher and the subscriber for processing; however, while the published table resides on the enterprise server, the subscribed table resides on the workgroup server.

See Also

- *Push Replication for Database-Only Workgroup Servers*
- *Understanding NON Mode Replication*

What is the Data Replication Change Log?

When you change data in a published table, the publisher writes a record in the Data Replication Change Log (F98DRLOG). The record contains the nature of the change, such as add, change, or delete, and a copy of the actual changed data. The record also contains the counter of subscribers that require notification for the change. Each time a subscriber successfully receives notification, OneWorld removes the value for the subscriber in the counter. Each server that you set up as a publisher must contain the Data Replication Change Log in the server map data source. This setup ensures that the subscribers of this publisher can receive any pending changes to the published table.

What is the Data Replication Pending Change Notifications Table?

A header-detail relationship exists between the Data Replication Change Log (F98DRLOG) and the Data Replication Pending Change Notifications table (F98DRPCN). The Data Replication Pending Change Notifications table contains a record for each subscriber that needs to update the replicated information. The table contains a status flag that indicates whether the publisher sent a notification message to a given subscriber. When the Subscriber acknowledges receipt of the notification, OneWorld deletes the records in the Data Replication Pending Change Notifications table and removes the value for the subscriber in the counter in the Data Replication Change Log. When all deliveries for the change have been acknowledged, OneWorld deletes the associated header record in the Data Replication Change Log. Each server that you set up as a publisher must contain the Data Replication Pending Change Notifications table in the server map data source.

What is the In-Synch Flag?

The In-Synch flag indicates whether the data of the replicated table on a subscriber matches the published table on the publisher. When you add a new subscriber, the flag defaults to "N", which means out of synch. This setting causes OneWorld to copy the published table to the subscriber, which provides a match at the start of the replication process for that table. Unless for a specific reason, do not change the In-Synch flag.

Any time you sign onto OneWorld from a subscriber workstation with tables that are not synchronized with the published tables, a message appears to notify you that the subscriber is not synchronized with the publisher. This notification gives you the opportunity to copy the table to your machine. If you copy the table, the replication process automatically changes the subscriber record so that the In-Synch flag setting is "Y".

The synchronization process works differently when server to server, or push replication, exists. Push replication does not use the In-Synch flag. When you set up data replication, you must ensure that the publisher tables and the subscriber tables match. If the tables do not match, you must manually copy the publisher table to the subscriber.

When a workstation subscriber is out of synchronization, the machine is effectively disabled; however, the workstation still receives the notification message each time a OneWorld session starts on the machine.

What is Forced Synchronization?

You can set up the workstation to perform a forced synchronization of all replicated tables. A forced synchronization means that the publisher updates all replicated data regardless of the messages that exist in the Data Replication Change Log table (F98DRLOG) or the Data Replication Pending Change Notification table (F98DRPCN). Generally, the workstation performs a forced synchronization the first time you sign on to OneWorld on the machine. This synchronization occurs because the following setting does not appear in the jde.ini file:

```
[REPLICATION]
ForcedSync=forced synchronization 0/1 (off/on)
```

The setting does not appear if you do not include the setting in the jde.ini file on your deployment server. After OneWorld performs the forced synchronization on the workstation, OneWorld writes a "ForcedSync" entry to the workstation jde.ini file with forced synchronization turned off. You can manually force synchronization of all replicated tables by changing the setting from 0 (off) to 1 (on). You can also perform a forced synchronization the first time you run OneWorld on a workstation by writing the "ForcedSync" entry into the workstation jde.ini file with a value of 1. For example, system administrators might force synchronization if they suspect data corruption in any of the OneWorld data replication tables (F98DRLOG or F98DRPCN) such as missing entries for subscriber updates.

See Also

- *Modifying the Workstation and Server jde.ini Files*

What is the Enable/Disable Flag?

This flag determines whether to enable or disable data replication. When you disable a publisher or a subscriber, you deactivate data replication for the machine. For example, when you initially setup data replication, you create multiple publisher and subscriber records; however, you should not enable the records until your replication configuration is completely setup.

The publisher does not log modifications to a disabled published table in the Data Replication Change Log (F98DRLOG) or the Data Replication Pending Change Notifications table (F98DRPCN). When you change an enabled published table with some enabled subscribers and some disabled subscribers, the publisher logs the modifications in the Data Replication Change Log, but the publisher only enters records for enabled subscribers in the Data Replication Change Notifications table.

Caching Replication Information

When either a server or a workstation starts OneWorld, OneWorld reads the Data Replication Publisher table (F98DRPUB) and caches the table locally. OneWorld also reads the Data Replication Subscribers table (F98DRSUB) and caches subscriber records on the machine. By locally caching this information, the database management component of OneWorld middleware (JDEBASE) can detect and manage modifications to published tables. You should point users, even WAN users, to a single copy of F98DRPUB and F98DRSUB in the main enterprise server database.

Workstations and servers cache information about published tables and subscribers when a OneWorld session begins. This cached data is critical to the detection of modifications to published tables. The system administrator must closely monitor the status of replicated data to maintain the integrity across the enterprise. You can refresh the cache on publisher hosts from the Work with Publishers form using the Refresh Hosts option on the Form menu. Also, internal user organization procedures should require that you restart OneWorld on all machines in the enterprise when you change the Data Replication Publisher table (F98DRPUB) and the Data Replication Subscribers table (F98DRSUB).

When JDEBASE detects a change to a published table, the publisher of the changed table receives a message. The publisher then makes the appropriate entries in the Data Replication Change Log (F98DRLOG) and the Data Replication Pending Change Notifications table (F98DRPCN) to notify the subscribers of this table.

The following examples outline this process for different situations.

Planning a Replication Strategy

When you plan the deployment of OneWorld tables in a distributed data environment, you should consider that OneWorld applications might access multiple database tables during application processing. For example, an application might access data from one table that requires data from another table. These application table dependencies require you to replicate certain tables together. If you split associated tables across multiple locations, you will decrease the performance of your distributed configuration. Also, you should replicate some groups of tables together for functional and performance reasons.

Some limitations exist in OneWorld for distributed data and joined business views that span multiple databases. This limitation impacts any application that uses these business views.

OneWorld bases joins on the mappings of individual tables regardless of the location of the tables. JDB database middleware (JDEBASE) performs cross data source join operations.

Important: The Conference Room Pilot (CRP) will determine whether the replication of specific joined tables is practical for your enterprise. You should formulate the specific strategy for replication of OneWorld tables as part of the standard CNC Conference Room Pilot (CRP).

Important: A critical part of your implementation strategy is determining whether the replication of specific joined tables benefits your enterprise. Be sure to make this determination early in the implementation process, for example, during the Conference Room Pilot (CRP).

Tables Suitable for Replication

This topic lists groups of tables that you can replicate to improve performance in distributed and WAN environments.

System Tables

System tables reside in the System Data Source. Typically, client memory caches system tables, so OneWorld accesses these tables only at signon; however, you might replicate certain system tables to improve the signon performance of clients in distributed and WAN environments.

Use caution if you replicate system tables, because the requirements for system table replication are specific to each individual enterprise. The CRP should highlight the tables that your enterprise requires you to replicate.

Constants Tables

Constants tables contain system and application constants specific to an application, such as information specific to General Ledger, or generally applicable to all applications, such as Automatic Account Instructions (AAIs).

You must map all replicated constants tables to the same location. The following table lists examples of constants tables that you can replicate in distributed environments and WAN environments:

Table	Description
F0007	Work Day Calendar
F0008	Date Fiscal Patterns
F0009	General Constants Tables
F0010	Company Constants Tables
F0012	Automatic Accounting Instructions (AAIs) Master
F0013	Currency Codes
F0014	Payment Terms
F00141	Advanced Payment Terms
F0015	Currency Exchange Rates
F00151	Currency Exchange Rates (F0015 Header)
F0022	Tax Rules
F0025	Ledger Type Master File
F3009	Job Shop Manufacturing Constants
F40070	Preference Master
F40073	Preference Hierarchy
F4008	Tax Areas
F4009	Distribution/Manufacturing Constants
F40095	Default Locations/Printers
F40203	Order Activity Rules
F40205	Line Type Control Constants

Table	Description
F4095	Distribution/Manufacturing – AAI Values
F41001	Inventory Constants
F41003	Unit of Measure Standard
F98101	Imaging Constants

Master Tables

OneWorld application developers determine which tables are master tables. You must map all master tables to the same location. Because of the volume of data that these tables contain and the number of changes that these tables undergo, master tables are "non-static" or not suitable for replication using OneWorld replication tools. However, if you replicate these tables using native database or third party replication tools, you can improve the performance of distributed environments and WAN environments.

The requirements for system table replication are specific to each individual enterprise. The CRP should highlight the tables that your enterprise requires you to replicate.

Language Support Tables

The OneWorld architecture accommodates multiple languages. OneWorld assigns all users a language preference key within their user profiles. The language key is a two-character field that determines the language for OneWorld forms and reports. Only specific tables contain a language key.

You must map all language tables to the same location. For reduced administration and increased security, J.D. Edwards suggest that you replicate language tables to Workgroup Servers in distributed environments and WAN environments.

The following table lists all the language-enabled tables in OneWorld:

Table	Description
F0004D	User Defined Code Types – Languages **(1)**
F0005D	User Defined Codes – Languages **(1)**
F0006D	Business Unit Alternate Description Master **(2)**
F00090D	Supplemental Database Language Preference **(2)**
F0012D	AAI Alternate Description Master
F0083	Menu Text Override File

Table	Description
F00921	Menu Path File
F0901D	Account Master Alternate Description **(2)**
F4101D	Item Master – Alternative Description **(2)**
F5192D	Supplier Analysis Alternate Language Description **(2)**
F9202	Data Field Display Text
F9203	Data Item Alpha Descriptions
F98306	Processing Option Text
F98750	Forms Design Aid Text Information
F98760	Report Design Aid Text Information

Notes:

(1) This table is also considered a UDC table. This table must be replicated as a group with F0004 and F0005.

(2) This is a OneWorld application language table and it should be replicated as a group with the F98* tools language tables.

User Defined Code (UDC) Tables

OneWorld attaches user defined codes (UDCs) to data items using the data dictionary and sometimes the Report Design Tool (RDA). These codes list the valid values for a data item. English versions of user defined codes reside in the F0004 and F0005 tables. See *Language Support Tables* for information about language-enabled UDC tables, such as F0004D and F0005D.

In distributed and WAN environments, you can replicate all UDC tables to a Workgroup Server.

Important: If you replicate UDC tables, you must replicate the tables together. Do not separate these tables.

The following table lists the UDC tables available for replication:

Table	Description
F0004	User Defined Code Types

Table	Description
F0005	User Defined Codes
F0004D	User defined Code Types – Languages
F0005D	User Defined Codes – Languages

Menus

You can replicate all menus in OneWorld. You must map all menu tables to the same location. In distributed environments and WAN environments, you can replicate these tables to a Workgroup Server.

Important: If you replicate menu tables, you must replicate the tables together. do not separate these tables.

The following table lists the menu tables available for replication:

Table	Description
F0082	Menu Master File
F00821	Menu Selections File
F0083	Menu Text Override File
F0084	Menu Path File

Note: You should build the following tables at the WAN site using the Build Menu Word Search Master Table (R00825) batch application. Do not replicate these tables; however, you should point workstation OCM mappings for these tables to the workgroup server.

- Menu Word Search Master Table (F00825)
- Menu Word Search Occurrences File (F00826)

Tables Unsuitable for Replication

The following table lists OneWorld tables that you cannot replicate using OneWorld replication tools:

Group	Comment
Generic Text (GT)	As a general rule, you cannot replicate tables containing Generic Text (GT) using the OneWorld replication tool; although, you can consider using third party tools. The following list provides examples of these tables: • **GT92002** Data Dictionary – Glossary Information
System Application Tables	Only one copy of the next number tables, F0002 and F00021, can exist in any OneWorld environment. These tables must reside on a Database Server or an Enterprise Server. You cannot replicate these tables to a Workgroup Server or to an Application Server. The following list provides examples of these tables: • **F0002** Next Numbers – Automatic • **F00021** Next Numbers by Company/Fiscal Year – Auto
System	Typically, OneWorld system programs, as opposed to application programs, use system tables. The system tables reside in the System Data Source. The following list provides examples of these tables: • **F98701** Next ID Master • **F98950** User Overrides • **F98DRLOG** Data Replication Change Log • **F98DRPCN** Data Replication Pending Change Notification • **F98OWSEC** OneWorld Security
Transaction or Balance Tables	J.D. Edwards does not recommend using the OneWorld data replication tools to replicate transaction or balances tables or any other highly changing tables. The replication of these tables requires a significant administration effort and exposes your enterprise to data integrity issues. the following list provides examples of these tables: • **F0902** Account Balances File • **F0911** Account Ledger • **F4211** Sales Order Detail • **F4311** Purchase Order Detail

Group	Comment
Application Worktables	Application worktables are temporary tables unique to an application session. Occasionally, OneWorld uses these tables instead of temporary cached memory. The following list provides examples of these tables: • **F40UI801** Generic Error Table • **F42UI01** Sales Order Header Cache File (MBF) • **F42UI11** Sales Order Detail Cache File (MBF)

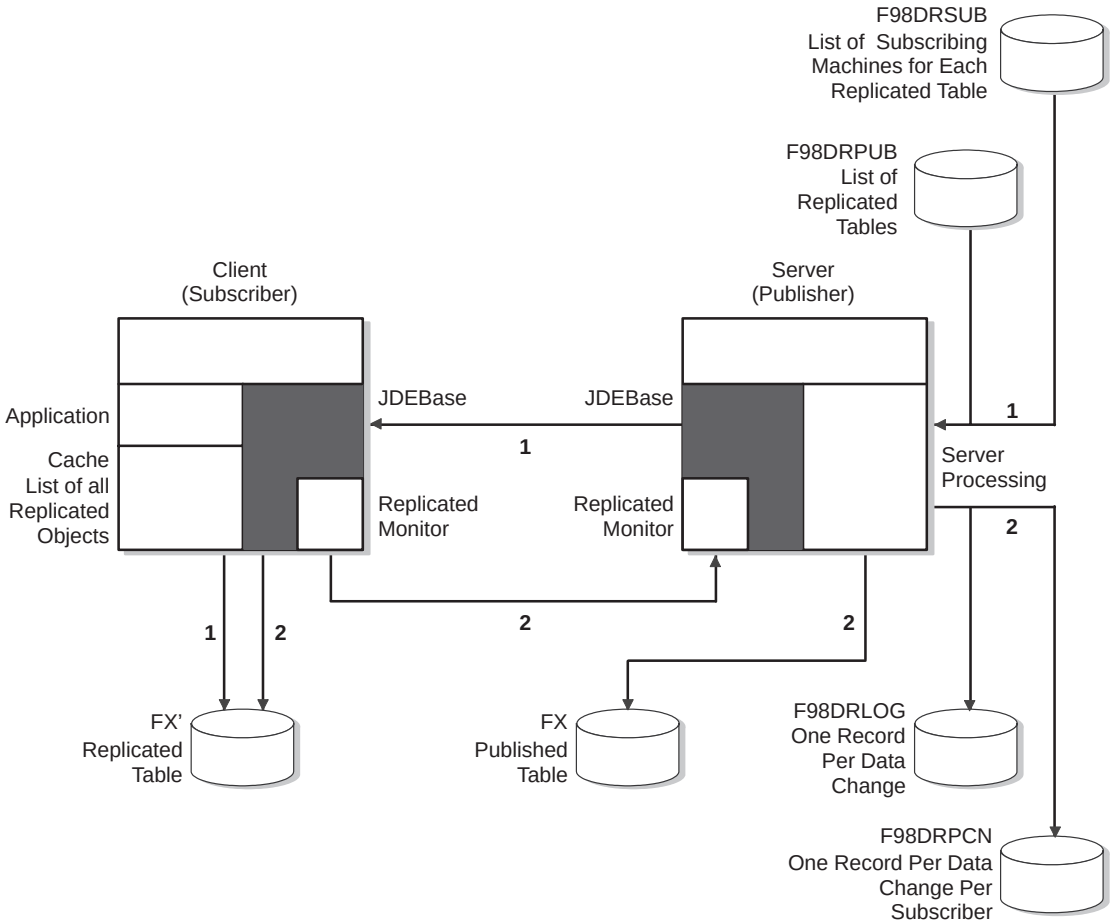
Understanding Pull Replication

Pull replication is the recommended type of replication for workstations. A subscriber using pull replication must be a machine that runs OneWorld.

When you start OneWorld on a subscriber that uses pull replication, the subscriber machine retrieves, or pulls, any pending changes queued in the Data Replication Pending Change Notifications table (F98DRPCN) for that subscriber. Based on records retrieved from F98DRPCN and the corresponding entries in the Data Replication Change Log (F98DRLOG), the subscriber updates the replicated tables.

The following illustration provides an example of pull replication processing:

Pull Replication



(1) When you start a subscriber:

At startup, the subscriber copies all replicated table information that resides in the Data Replication Publisher table (F98DRPUB) and subscription information for the subscriber contained in the Data Replication Subscribers table (F98DRSUB) and stores the information in cache. Also, the subscriber processes any undelivered replication messages that reside in the Data Replication Pending Change Notifications table (F98DRPCN). The subscriber updates the replicated tables when these messages process.

(2) When you change a published table:

If the subscriber modifies a replicated table, the Replication Monitor routine of JDEBASE detects the change and triggers a process that modifies the published table. JDEBASE sends a synchronous message to the publisher, which responds by updating the Data Replication Change Log (F98DRLOG) with the changed data and the Data Replication Pending Change Notifications table (F98DRPCN) with the list of subscribers that also need the change.

Maintaining Data Integrity Between Subscribers and Publishers for Pull Replication

If the subscriber is a workstation, replicated tables reside in the Microsoft Access database on the workstation. When you change the data in the local subscriber tables, OneWorld automatically changes the data in the published tables on the server. This process is identical to the process used for JITR replication. The following list of steps details this process:

▶ **To maintain data integrity between subscribers and publishers**

1. The subscriber sends a message to the publisher of the changed table.
2. The publisher updates the published table.
3. The publisher writes the appropriate entries in the Data Replication Change Log (F98DRLOG) and the Data Replication Pending Change Notifications table (F98DRPCN).
4. The publisher replicates the changes to all other subscribers based on the entries in F98DRLOG and F98DRPCN.

Understanding Just-In-Time Replication

Just-in-time replication (JITR) is another method of replication that can be used when the Subscriber is a workstation.

Any changes and deletions a user does not need during an active session process in accordance with pull replication when the user signs on to a workstation.

Just-in-time replication works in conjunction with the following OneWorld processes:

Validating a User Defined Code (UDC) Field

If a search for a record in the local replicated table fails, OneWorld automatically performs a second search in the published table. If the record resides in the published table, the publisher immediately updates the record in the replicated table.

For example, assume a user enters a sales order and types in a new line type. When the user exits the field, OneWorld validates the line type against the local copy for that table. If the data source does not contain the new line type, OneWorld checks the data source of the published table. The validation succeeds and the replication process copies the new line type to the local replicated table.

Using Visual Assist

Visual assist always looks to the published table's data source.

When using visual assist, OneWorld reads the data from the published table. If you select a value that does not exist in the replicated table, OneWorld copies the data to the data source of the subscriber.

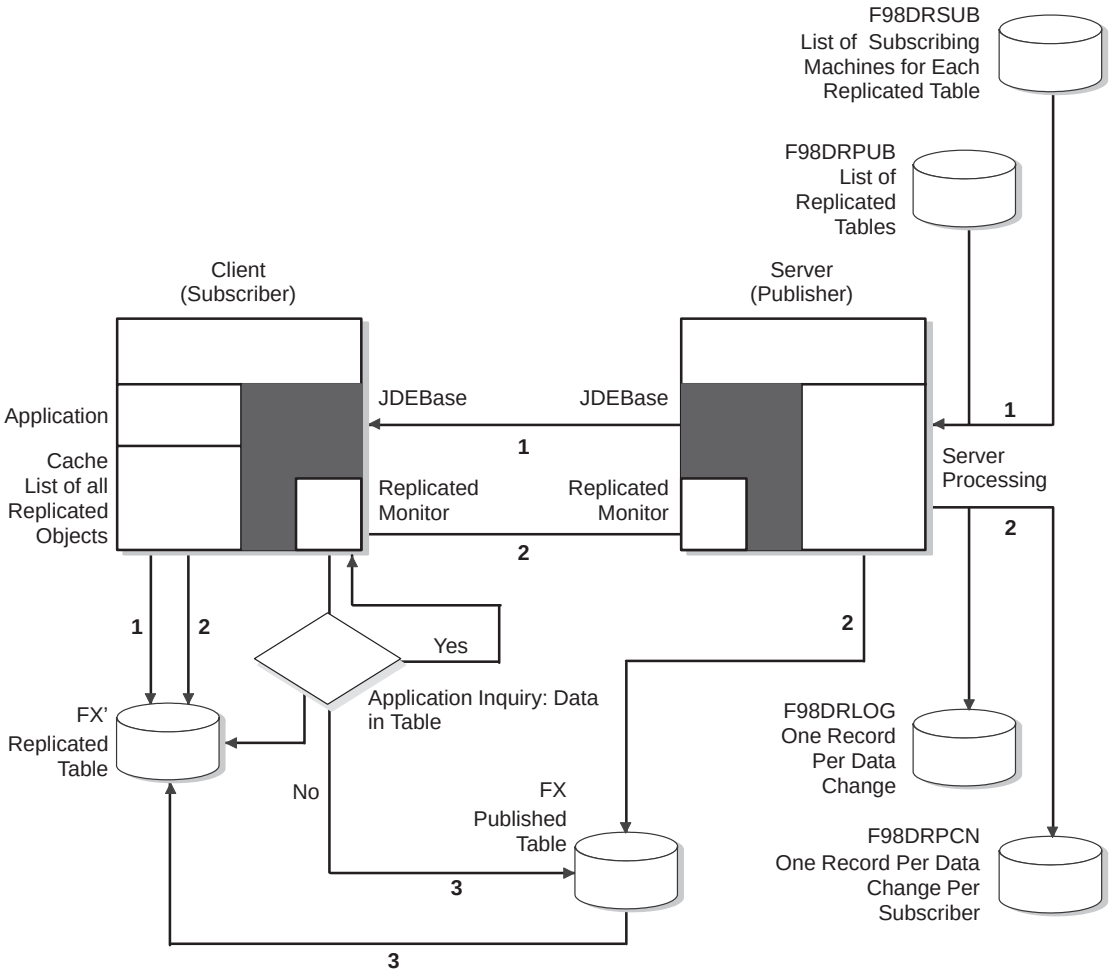
For example, assume a user enters a sales order and uses visual assist for the unit of measure field. When you set up just-in-time replication for a table, OneWorld always looks to the published table to display the valid values for that table. Selecting the new unit of measure causes just-in-time replication to copy the value from the published table to the subscriber table.

Data Dictionary

By default, data dictionary and global table specification files on a workstation yser JITR for data dictionary replication. See *Replicating Data Dictionary Changes* in this guide.

The following illustration provides an example of just-in-time replication processing:

Just–In–Time Replication



(1) When you start a subscriber:

At startup, the subscriber copies all replicated table information that resides in the Data Replication Publisher table (F98DRPUB) and subscription information for the subscriber contained in the Data Replication Subscribers table (F98DRSUB) and stores the information in cache. Also, the subscriber processes any undelivered replication messages that reside in the Data Replication Pending Change Notifications table (F98DRPCN). The subscriber updates the replicated tables when these messages finish processing and OneWorld removes the respective entries from F98DRPCN. OneWorld removes the associated entries in the Data Replication Change Log (F98DRLOG) after all subscribers receive the change.

(2) When you change a published table:

If the subscriber modifies a replicated table, the Replication Monitor routine of JDEBASE detects the change and triggers a process that modifies the published table. JDEBASE sends a synchronous message to the publisher, which responds by updating the Replication Change Log (F98DRLOG) and the Data Replication Pending Change Notifications table (F98DRPCN).

(3) When a workstation performs a query:

If the workstation queries a replicated table for row validation and cannot find the row, then the workstation queries the published table for the row. If the row resides in the published table, JDEBASE immediately replicates the row to the replicated table.

Maintaining Data Integrity Between Subscribers and Publishers for JITR Replication

If the subscriber is a workstation, replicated tables reside in the Microsoft Access database on the workstation. When you change the data in the local subscriber tables, OneWorld automatically changes the data in the published tables on the server. This process is identical to the process used for pull replication. The following list of steps details this process:

▶ To maintain data integrity between subscribers and publishers

1. The subscriber sends a message to the publisher of the changed table.
2. The publisher updates the published table.
3. The publisher writes the appropriate entries in the Data Replication Change Log (F98DRLOG) and the Data Replication Pending Change Notifications table (F98DRPCN).
4. The publisher replicates the changes to all other subscribers based on the entries in F98DRLOG and F98DRPCN.

Understanding Push Replication

Use push replication for server to server replication.

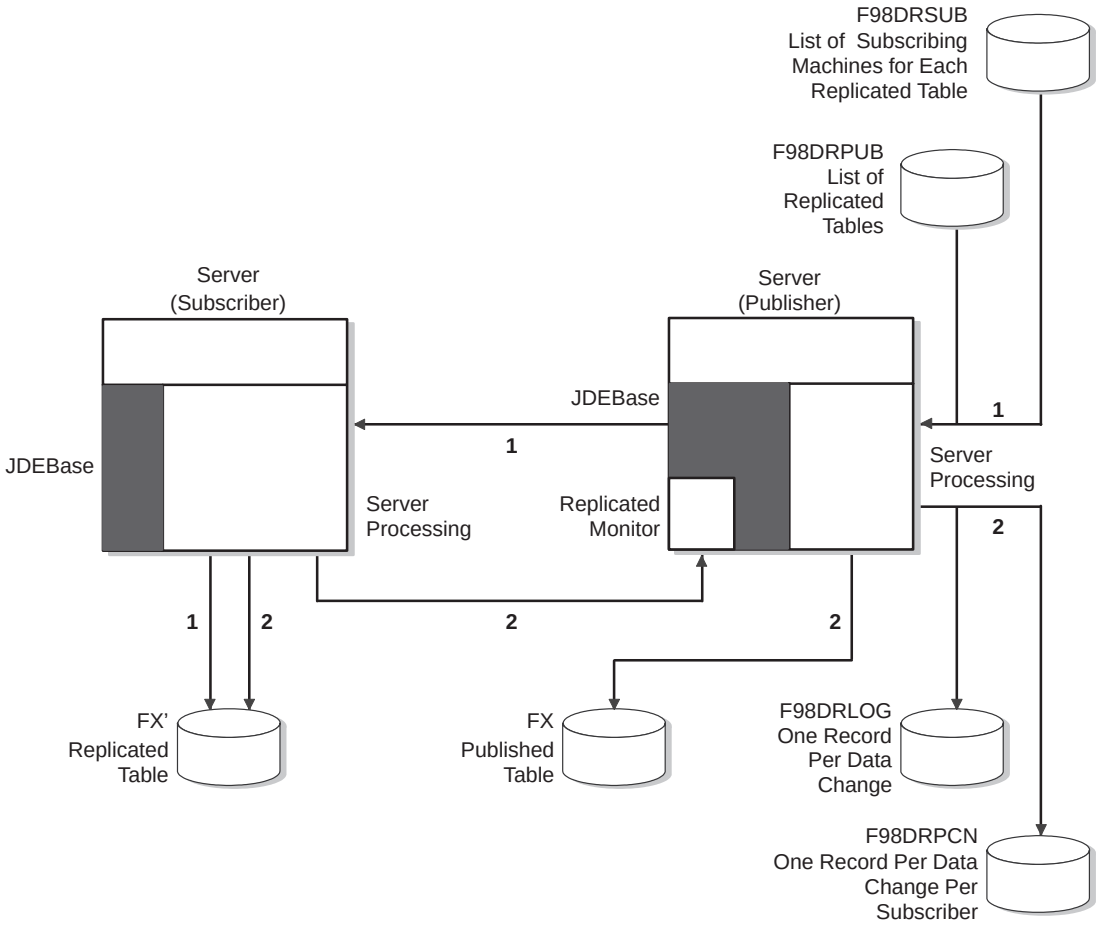
Push Replication is the most efficient method of replication and the easiest to administer. When you change a published table using push replication, the publisher immediately replicates the information on the subscriber. If the subscriber is not on line, the subscriber receives the modifications using the pull method when you restart OneWorld on the subscriber.

Push replication also occurs when subscriber tables change. If a valid Data Replication Publisher table (F98DRPUB) and a valid Data Replication Subscribers table (F98DRSUB) exist in cache on the workstation making the change, OneWorld records replication changes in the Data Replication Change Log (F98DRLOG) and the Data Replication Pending Change Notifications table (F98DRPCN) whenever a publisher table or a subscriber table changes.

When a network outage prevents push replication, you must restart the subscriber to receive the modifications made to the published table. If you do not restart the subscriber, the subscriber continues to use out of date information.

The following illustration provides an example of push replication processing:

Push Replication



(1) When you start a subscriber:

At startup, the subscriber copies all replicated table information that resides in the Data Replication Publisher table (F98DRPUB) and subscription information for the subscriber contained in the Data Replication Subscribers table (F98DRSUB) and stores the information in cache. Also, the subscriber processes any undelivered replication messages that reside in the Data Replication Pending Change Notifications table (F98DRPCN). The subscriber updates the replicated tables when these messages process.

(2) When you change a published table:

When you change a published table, the publisher records the modification in the Data Replication Change Log (F98DRLOG) and the Data Replication Pending Change Notifications table (F98DRPCN). The subscriber immediately updates the replicated tables if the subscriber is online. If the subscriber is offline, the publisher saves the message then delivers the message when the subscriber comes back online.

Push Replication for Database-Only Workgroup Servers

Workgroup Servers are database servers configured to run with replicated data in distributed environments or remote WAN locations. To provide a configuration that requires low maintenance, you can deploy a database-only Workgroup Server rather than duplicate the complex configuration of a full-scale Enterprise Server. A database-only Workgroup Server does not run any OneWorld server code. The server only supports a database.

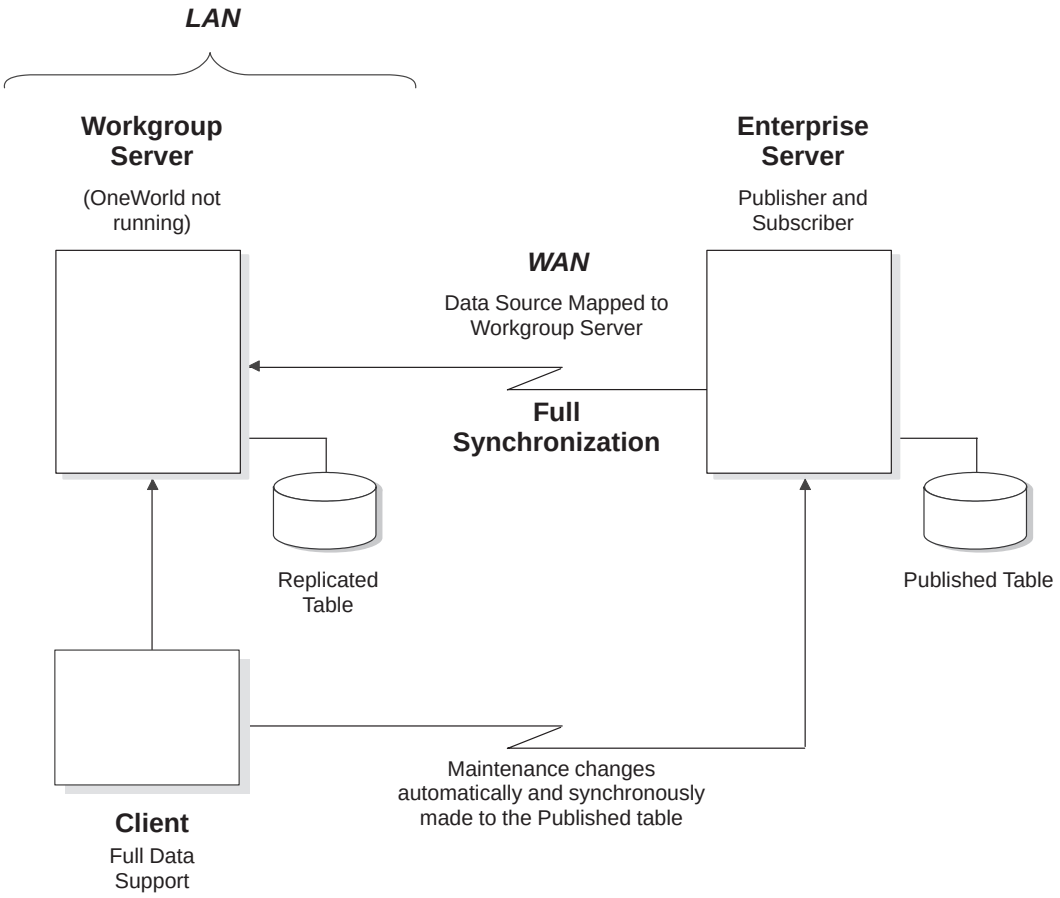
Important: You cannot use a workgroup server when you use an AS/400 as the publisher server. When you replicate data across platforms from an AS/400, OneWorld *must* reside on the workgroup server because JDBNet needs to run on the workgroup server for replication to occur successfully.

The method of Push replication for database-only Workgroup Servers allows you to set up the Enterprise Server as both a subscriber and a publisher for the replicated tables using the standard Push methodology. However, you set up the database-only Workgroup Server as the subscriber data source for the replicated tables. This setup is possible without OneWorld code running on the database-only Workgroup Server. This method ensures the synchronization of the replicated tables on the database-only Workgroup Server with the published tables on the Enterprise Server. When you change replicated data on the database-only Workgroup Server, you simultaneously change the data on the Enterprise Server.

On a local area network (LAN), you can use a database-only workgroup server, which does not run OneWorld, as a subscriber for the replication of published tables. In this configuration, an attached workstation is a nonsubscriber that points to the replicated data on the workgroup server. If the workstation changes the replicated data, the published table does not change automatically; therefore, workstations attached to subscriber workgroup servers should only use the replicated tables on the workgroup server for validation purposes, which are read only. If you need to make changes to the replicated tables, you should use a separate environment on the workstation, by logging on to another environment with OCM mappings to the central published table, to update the central published table. After you update the central published table, use push replication to update the subscribing workgroup server.

The following illustration provides an example of the Push replication method for a database-only Workgroup Server:

Push Replication Method for Database-only Workgroup Servers



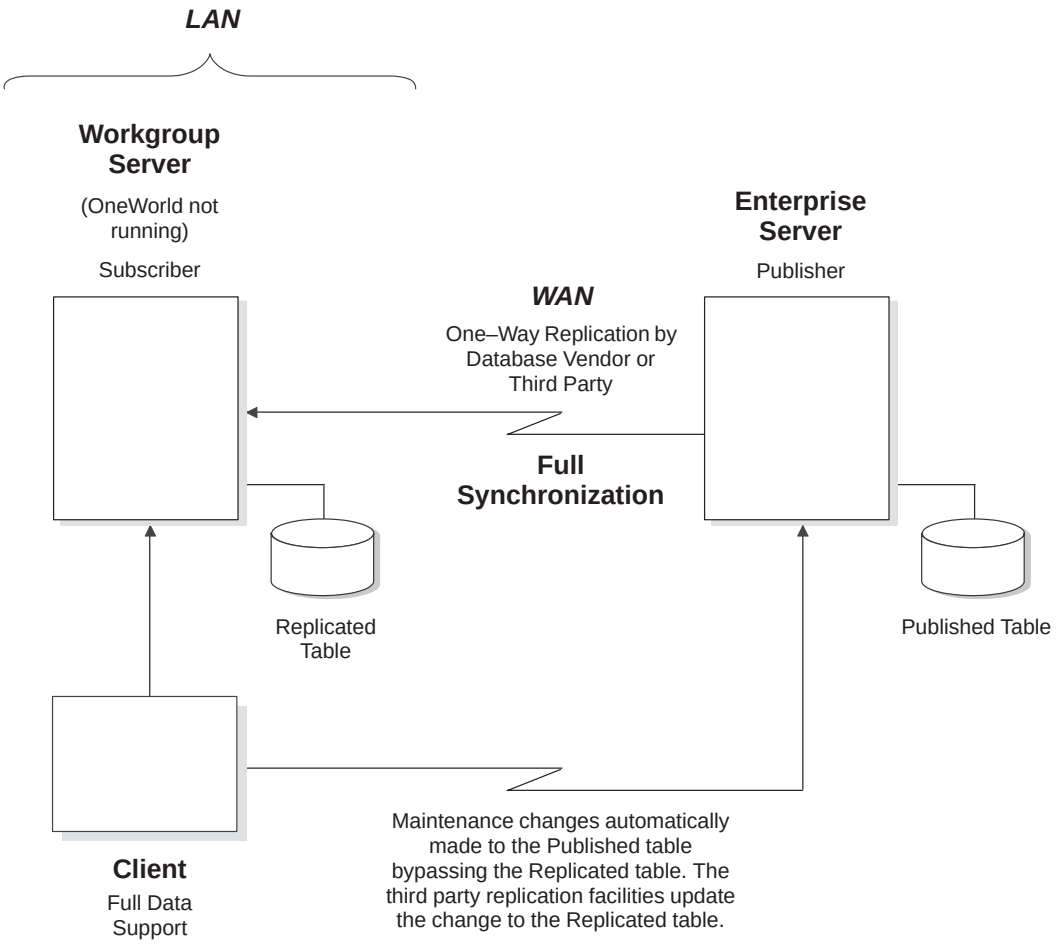
Third Party Replication

With a NON subscription type, you update data in the published table directly from the client even though the OCM mappings for the client point to the database-only Workgroup Server that subscribes to the Enterprise Server. This setup allows the client to access the replicated data on the Workgroup Server for the validation of data and perform “inserts,” “updates,” and “deletes” to the published table on the Enterprise Server. *Third party* replication tools then implement replication services for the configuration.

Primarily, the advantage to this strategy is that a separate client session is not necessary to update master copies of replicated data on the publisher. Also, you do not need to maintain a OneWorld install on the Workgroup Server. This server can remain database-only. The disadvantage is that a change made to a published table might not immediately replicate to the subscribing database-only Workgroup Server.

The following illustration provides an example of NON mode replication:

NON Mode for Database–Only Workgroup Servers

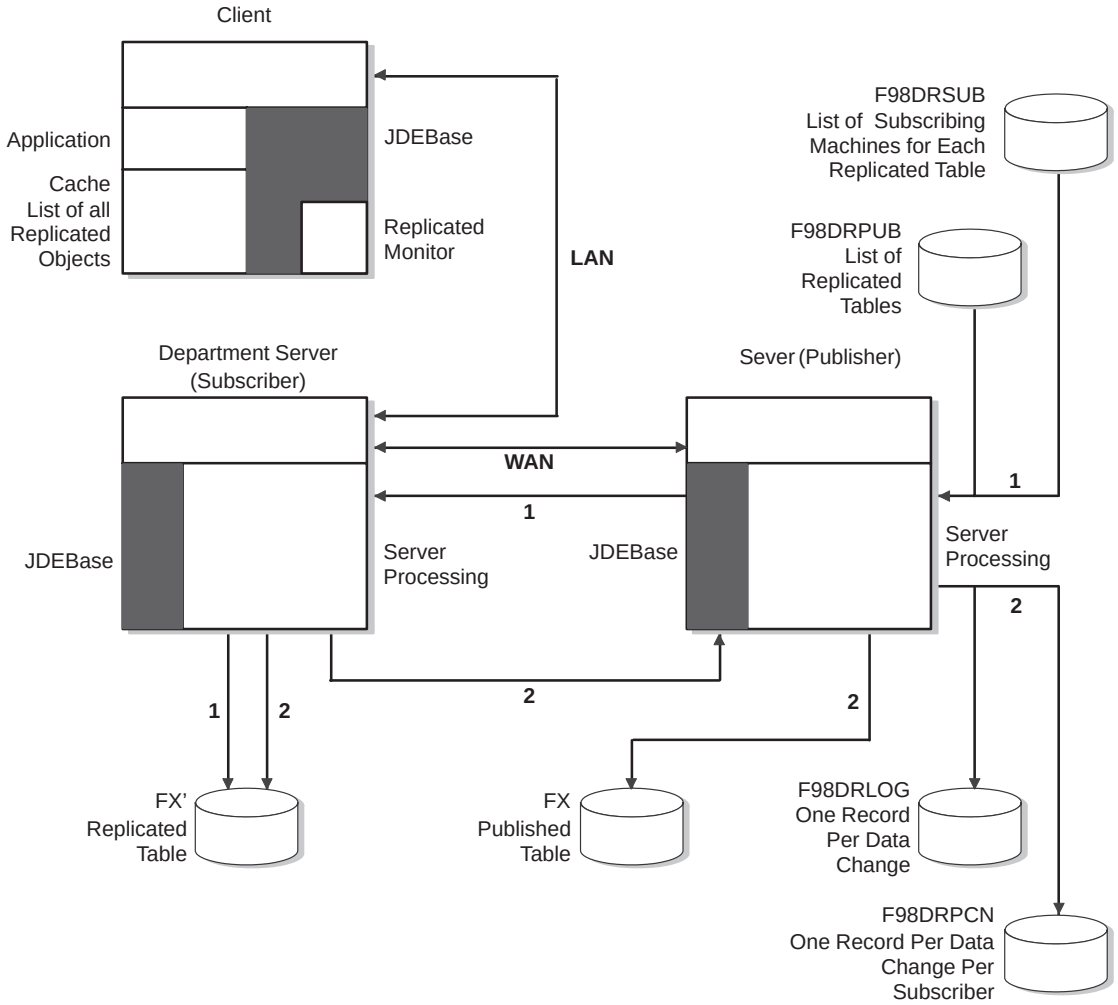


Workgroup Servers on a Wide Area Network

In a Wide Area Network (WAN) environment, you should consider using Workgroup Servers to store replicated information for users who reside in a remote location.

The following illustration provides an example of a typical Workgroup Server setup in a WAN environment:

Workgroup Server as Subscriber



(1) When you start a subscriber:

At startup, the subscriber copies all replicated table information that resides in the Data Replication Publisher table (F98DRPUB) and subscription information for the subscriber contained in the Data Replication Subscribers table (F98DRSUB) and stores the information in cache. Also, the subscriber processes any undelivered replication messages that reside in the Data Replication Pending Change Notifications table (F98DRPCN). The subscriber updates the replicated tables when these messages finish processing.

(2) When you change a published table:

When you change a published table, the publisher records the modification in the Data Replication Log (F98DRLOG) and the Data Replication Pending Change Notifications table (F98DRPCN). The subscriber immediately updates the replicated tables if the subscriber is online. If the subscriber is offline, the publisher saves the message then delivers the message when the subscriber comes back online.

Note 1: The Object Configuration Manager for the workstation points to replicated data on the Workgroup Server.

Note 2: When you change replicated data on a workstation in a WAN environment, the publisher does not record the modifications. If you want to update replicated data, you should start a separate session of OneWorld to directly change the published table.

Servers as Both Publishers and Subscribers and Chain Replication

The J.D. Edwards replication facilities support the capability to configure a server as both a publisher and a subscriber. You can also configure your enterprise for chain replication. Chain replication means that you setup a server to subscribe to a published table, then that same server is a publisher to another subscriber. Chain replication greatly complicates data replication, so you must carefully design and manage a configuration that uses chain replication. OneWorld supports one-way push replication technology; therefore, to maintain data integrity you should avoid modifications to replicated data in the middle of a chain. In a configuration that uses chain replication, you should make all changes to replicated data at the beginning of the chain to ensure the maintenance of data integrity.

Selective Replication Using Table Conversion

Data replication cannot selectively replicate. For example, if you want to replicate only the Address Book records with a search type of C and a specific category code, you need to use Table Conversion to perform the replication. You control the replication by manually submitting a table conversion, so you should use this replication method sparingly. You might consider using Table Conversion to replicate data in the following situations:

- Infrequent replication of data for a group of tables
- Replication of records based on specific data selection
- Replication of transaction data required by store and forward processing

You should only use Table Conversion to replicate data in isolated situations that do not require the synchronized replication of data, as in real-time data visibility.

Setting Up Data Replication

Use the Data Replication application to set up specific tables as publishers and subscribers. This method of replication, which is integrated into OneWorld middleware, automatically detects changes to publishers and notifies the subscribers of the changes, or updates subscribers as the changes are used.

With a coexistence configuration, if change detection is important to you, do not modify publisher tables using WorldSoftware programs, unless you set up subscribers with just-in-time replication. J.D. Edwards integrated the change detection into the OneWorld middleware, which will not detect changes made by WorldSoftware programs.

The following list provides the steps for data replication setup:

- Create publishers
- Create subscribers

Before you set up data replication, it is important to understand the size of your organization so you can devise an appropriate plan, both for now and for the future. The plan should answer the following questions:

- Can I follow any of the case studies to help me organize the data replication flow? If so, which one?
- Which tables do I want replicated and why?
- What type of replication should each table use?
- Who is going to be the data replication administrator?

When you have answered these questions you can set up data replication for a conference room pilot (CRP) environment where any issues are resolved among a few selected users. After a successful trial has been accomplished, you can implement replication throughout the enterprise.

This topic contains the following:

- ❑ Working with publishers
- ❑ Working with subscribers
- ❑ Enabling and Disabling Publishers and Subscribers
- ❑ Creating initial publishers and subscribers via a batch process

- ❑ Copying a publisher and its associated subscribers
- ❑ Setting Up the Environment for Data Replication
- ❑ Modifying the server and workstation jde.ini files
- ❑ Viewing the replication logs

Once you have seen the details of the application, you will be presented with three models that data replication can be based off, and then an example plan for using it in your production environment.

Working with Publishers

This topic explains how to add and delete publishers. After you have added a publisher, you can proceed to adding subscribers, as explained in this section. This topic contains the following:

- ❑ Adding a publisher
- ❑ Deleting a publisher

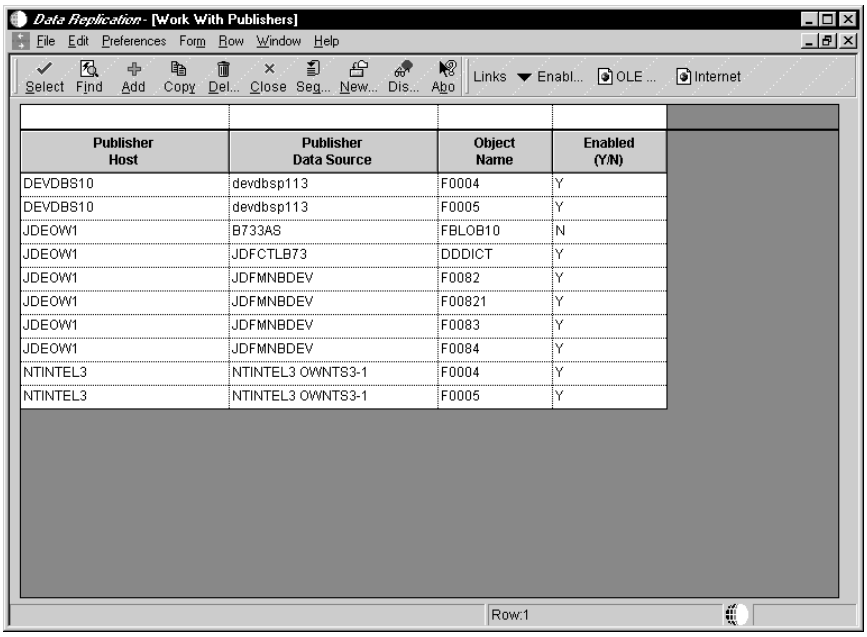
Adding a Publisher

Use this application to add a publisher table (F98DRPUB). The publisher table identifies all of the tables that you want replicated.

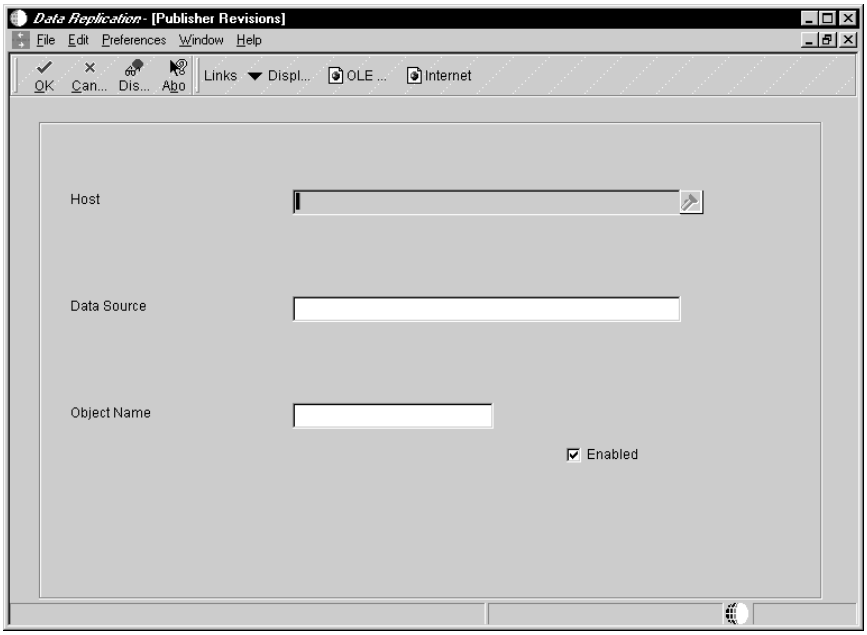
▶ **To add a publisher**

On the System Administration Tools (GH9011) menu

1. Choose Data Replication (P98DREP).



2. On the Work With Publishers form, click Add.



3. On the Publisher Revisions form, complete the following fields:

   - Data Source

     The Host field automatically populates with the server machine name for the server associated with the data source in the data source definition.

   - Object Name

Field	Explanation
Data Source	The 30-character name of the data source where the published object resides.
Object Name	The OneWorld architecture is object based. This means that discrete software objects are the building blocks for all applications, and that developers can reuse the objects in multiple applications. Each object is stored in the Object Librarian. Examples of OneWorld objects include: • Batch Applications • Interactive Applications • Business Views • Business Functions • Business Functions Data Structures • Event Rules • Media Object Data Structures
Enabled	The purpose of disabling a publisher or subscriber is to turn off replication for that publisher or subscriber without deleting the record. For example, suppose you are preparing to activate replication. You will create many publisher and subscriber records, but do not want the records enabled until the day you activate replication. Disabling a publisher effectively disables all its subscribers, as replication is completely turned off for the publisher table. Changes made to a publisher table will not be logged in the Data Change Log nor the Pending Change Notification Log. A publisher may have some subscribers enabled and other subscribers disabled. Changes to the publisher table will be stored in the Data Change Log. Only subscribers that are enabled who have not received the changes will have records in the Pending Change Notification. . . . . . *Form-specific information* . . . . . . This flag determines whether data replication is enabled or disabled.

Deleting a Publisher

You cannot modify a publisher. Instead, you must delete it, and create a new one.

▶ To delete a publisher

On the System Administration Tools (GH9011) menu

1. Choose Data Replication (P98DREP).
2. On the Work With Publishers form, highlight a publisher, and click Delete.

   This deletes the publisher along with any subscribers.

Working with Subscribers

This topic explains how to add, synchronize, and delete subscribers. This topic contains the following:

- ❑ Adding a subscriber
- ❑ Synchronizing and unsynchronizing subscribers
- ❑ Deleting subscribers

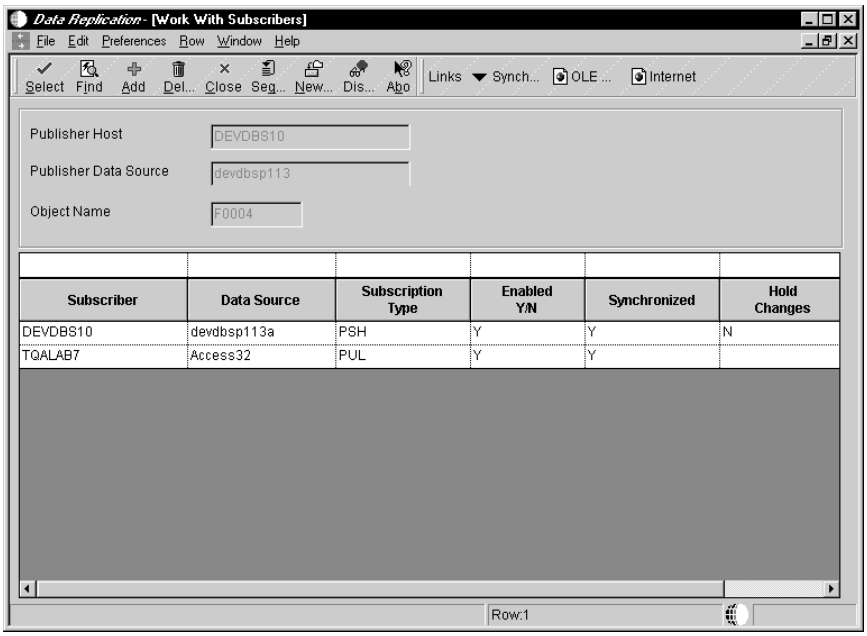
Adding a Subscriber

Use this application to add subscriber tables.

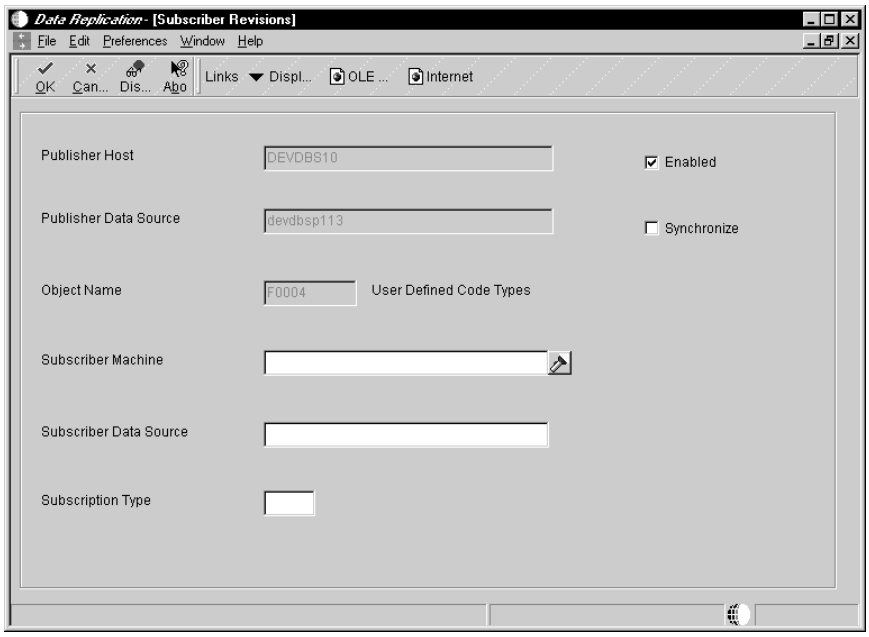
▶ To add a subscriber

On the System Administration Tools (GH9011) menu

1. Choose Data Replication (P98DREP).
2. On the Work With Publishers form, highlight a publisher, and from the Row menu, choose Subscribers.



3. On the Work With Subscribers form, click Add.



4. On the Subscriber Revisions form, complete the following fields:
   - Subscriber Machine (if the subscriber is a departmental server, type the server's machine name into this field because departmental servers do not appear on the visual assist Machine Search & Select form)
   - Subscriber Data Source
   - Subscription Type
5. Choose whether you want the following flags turned ON or OFF:
   - Enabled
   - Synchronize
   - Hold Changes (only appears after you enter a departmental server into the Subscriber Machine field)

   When you add a subscriber record it is automatically set up as not synchronized.

Field	Explanation
Subscriber Machine	The name of the network server where data resides or where objects can be executed. . . . . . *Form-specific information* . . . . . . On this form, Subscriber Machine is the machine name that is set up to receive replicated changes for a published table. A subscriber is a combination of the machine and a data source on that machine.

Field	Explanation
Subscriber Data Source	A 30-character name of the data source where the subscribed object resides.
Subscription Type	Subscription Type identifies the type of replication performed for a subscriber. Server subscribers must be set up as Push replication. Workstation subscribers can be set up as Pull replication or Just-in-time replication (JITR). Push subscribers receive changes immediately. If the change cannot be replicated to a server because OneWorld is not running, the message is stored until OneWorld on the server subscriber is restarted, at which time the server pulls the change. Pull subscribers receive the change the next time they sign on to OneWorld. JITR subscribers receive new records 'just in time.' However, deletions and updates are replicated the next time the subscriber machine signs onto OneWorld.
Enabled	This flag determines whether data replication is enabled or disabled. The purpose of disabling a subscriber is to turn off replication for that subscriber without deleting the record. For example, suppose you are preparing to activate replication. You will create many publisher and subscriber records, but do not want the records enabled until the day you activate replication. A publisher may have some subscribers enabled and other subscribers disabled. Changes to the publisher table will be stored in the Data Change Log. Only subscribers that are enabled who have not received the changes will have records in the Pending Change Notification.
Synchronize	Checking this field will sychronize the object. If this field is not checked then the object is not in synch.
Hold Changes	Causes all changes to a published table to be held for this server subscriber. When the server subscriber is ready to receive the changes, the Hold Changes flag should be turned off. Then when OneWorld on the server is restarted, all changes that have been held for this subscriber are pulled. Hold Changes is not necessary for workstation subscribers.

Synchronizing and Unsynchronizing Subscribers

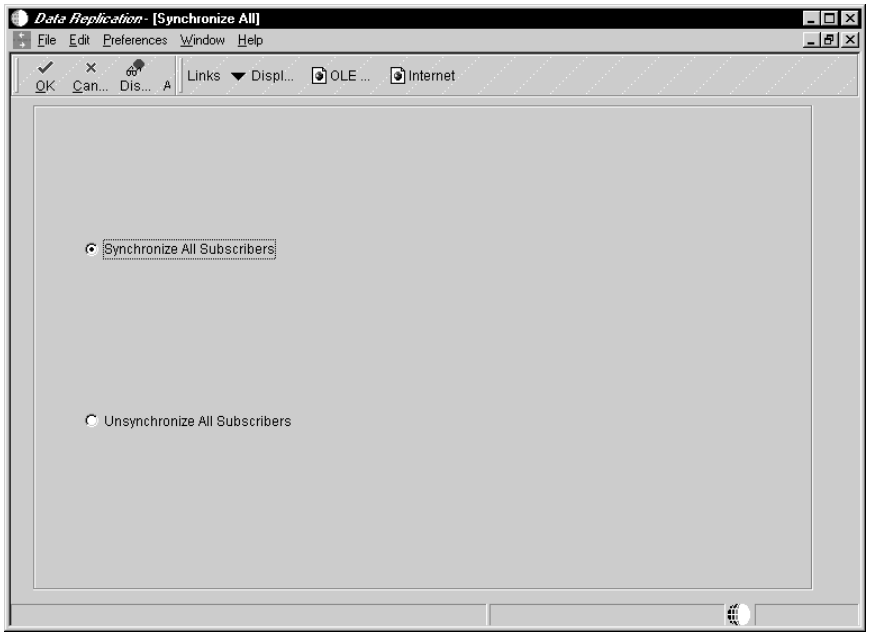
There are several different methods available to synchronize or unsynchronize subscribers depending on the outcome wanted. All methods involve setting the In Synch flag for the subscribers. The In Synch flag indicates whether a subscriber and a publisher table match. This topic contains the following:

- To synchronize and unsynchronize all subscribers
- To synchronize and unsynchronize subscribers of one publisher
- To synchronize and unsynchronize selected subscribers

▶ To synchronize and unsynchronize all subscribers

On the System Administration Tools (GH9011) menu

1. Choose Data Replication (P98DREP).
2. On the Work With Publishers form choose Synchronize All from the Form menu.



3. On the Synchronize All form, choose one of the following options, and click OK:
    - Synchronize All Subscribers (the In Synch flag for all of the subscribers for all publishers changes to Y, meaning OneWorld considers those subscribers to be in synch with their associated publisher)

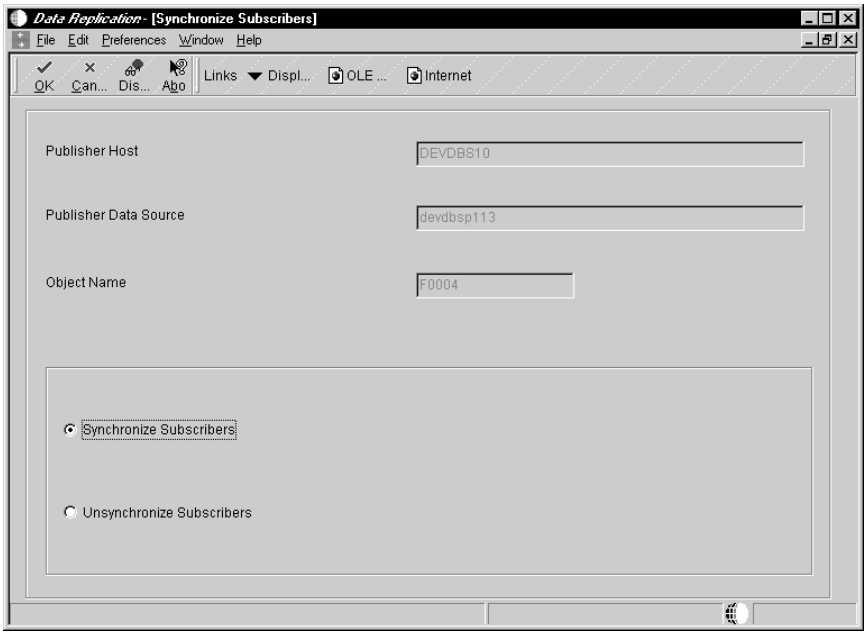
- Unsynchronize All Subscribers (the In Synch flag for all of the subscribers for all publishers changes to N, meaning OneWorld considers those subscribers to be *out* of synch with their associated publisher. The next time a user signs on, OneWorld will tell the user that the table is out of synch. If the user chooses to synchronize the table, OneWorld will copy it and switch the In Synch flag to Y)

When initially adding a subscriber, do not change the In Synch flag unless you are certain that the publisher and subscriber tables are identical.

▶ To synchronize and unsynchronize subscribers of one publisher

On the System Administration Tools (GH9011) menu

1. Choose Data Replication (P98DREP).
2. On the Work With Publishers form, highlight a publisher, and from the Row menu, choose Synchronize.



3. On the Synchronize Subscribers form, choose one of the following options, and click OK.

   - Synchronize Subscribers (the In Synch flag for all of the subscribers of the publisher that you chose changes to Y, meaning OneWorld considers those subscribers to be in synch with their associated publisher)

- Unsynchronize Subscribers (the In Synch flag for all of the subscribers of the publisher that you chose changes to N, meaning OneWorld considers those subscribers to be *out* of synch with their publisher. The next time a user signs on, OneWorld will tell the user that the table is out of synch. If the user chooses to synchronize the table, OneWorld will copy it and switch the In Synch flag to Y)

When initially adding a subscriber, do not change the In Synch flag unless you are certain that the publisher and subscriber tables are identical.

▶ To synchronize and unsynchronize selected subscribers

On the System Administration Tools (GH9011) menu

1. Choose Data Replication (P98DREP).
2. On the Work With Publishers form, highlight a publisher, and from the Row menu, choose Subscribers.
3. On the Work With Subscribers form, highlight one or more subscribers, and from the Row menu, choose Synchronize.

   The Synchronized field toggles between Y and N for synchronized and unsynchronized.

Deleting Subscribers

The following explains how to delete subscribers:

- Delete all subscribers
- Delete individual subscribers

▶ To delete all subscribers

On the System Administration Tools (GH9011) menu

1. Choose Data Replication (P98DREP).
2. On the Work With Publishers form, highlight a publisher, and from the Row menu, choose Delete Subscribers.

   OneWorld deletes all of the subscribers for that publisher, but the publisher record remains.

▶ To delete individual subscribers

On the System Administration Tools (GH9011) menu

1. Choose Data Replication (P98DREP).
2. On the Work With Publishers form, highlight a publisher, and from the Row menu, choose Subscribers.
3. On the Work With Subscribers form, highlight one or more subscribers, and click Delete.

Enabling and Disabling Publishers and Subscribers

You can enable and disable data replication for publishers and subscribers. This topic contains the following:

- ❑ Enabling and disabling all publishers and subscribers
- ❑ Enabling and disabling subscribers of one publisher
- ❑ Enabling and disabling selected subscribers

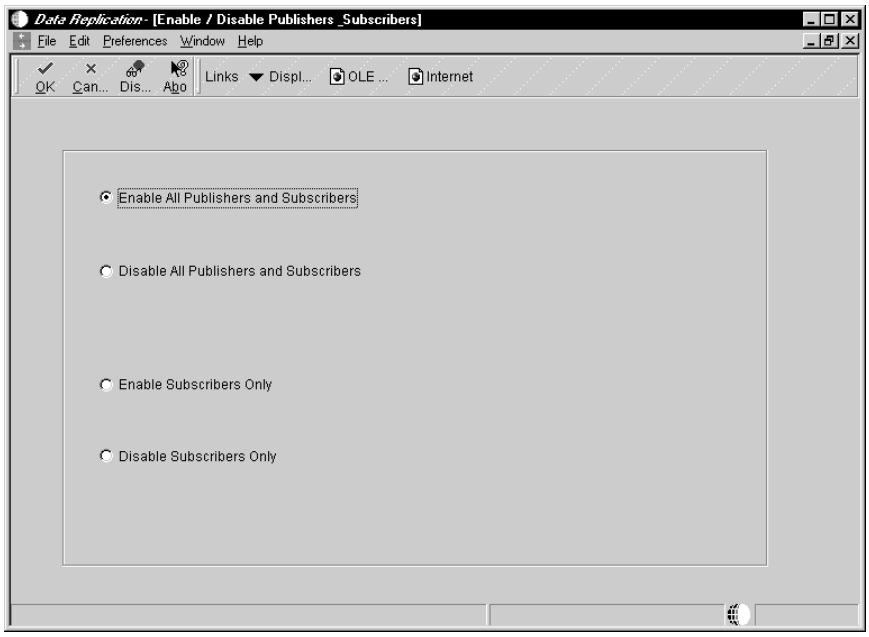
Enabling and Disabling all Publishers and Subscribers

You can enable and disable data replication for all *existing* publishers and subscribers, or enable and disable the data replication for all subscribers.

▶ To enable and disable all publishers and subscribers

On the System Administration Tools (GH9011) menu

1. Choose Data Replication (P98DREP).
2. On the Work With Publishers form, from the Form menu, choose Enable/Disable.



3. On the Enable / Disable Publishers & Subscribers form, choose one of the following, and click OK:
   - Enable All Publishers and Subscribers
   - Disable All Publishers and Subscribers
   - Enable Subscribers Only
   - Disable Subscribers Only

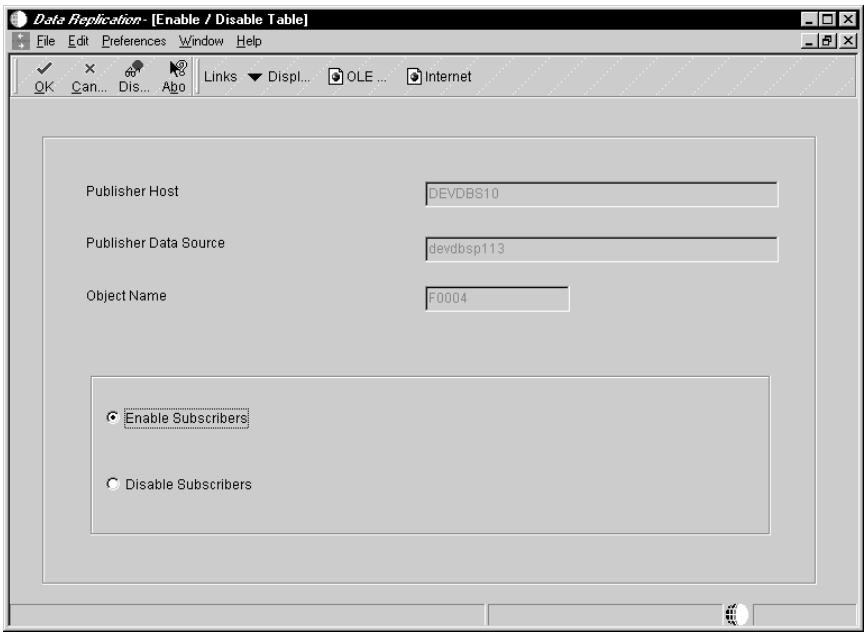
Enabling and Disabling Subscribers of one Publisher

You can enable and disable data replication for subscribers of a specific publisher.

▶ To enable and disable subscribers of one publisher

On the System Administration Tools (GH9011) menu

1. Choose Data Replication (P98DREP).
2. On the Work With Publishers form, highlight a publisher, and from the Row menu, choose Enable/Disable Subs.



3. On the Enable / Disable Table form, choose one of the following, and click OK:
   - Enable Subscribers
   - Disable Subscribers

Enabling and Disabling Selected Subscribers

You can enable and disable data replication for selected subscribers of a specific publisher.

▶ To enable and disable selected subscribers

On the System Administration Tools (GH9011) menu

1. Choose Data Replication (P98DREP).
2. On the Work With Publishers form, highlight a publisher, and from the Row menu, choose Subscribers.
3. On the Work With Subscribers form, highlight one or more subscribers, and from the Row menu, choose Enable/Disable.
4. On the Enable / Disable Table form, choose one of the following, and click OK:
   - Enable Subscribers
   - Disable Subscribers

Field	Explanation
Enable Subscribers	This flag determines whether data replication is enabled or disabled. The purpose of disabling a subscriber is to turn off replication for that subscriber without deleting the record. For example, suppose you are preparing to activate replication. You will create many publisher and subscriber records, but do not want the records enabled until the day you activate replication. A publisher may have some subscribers enabled and other subscribers disabled. Changes to the publisher table will be stored in the Data Change Log. Only subscribers that are enabled who have not received the changes will have records in the Pending Change Notification.

Creating Initial Publishers and Subscribers Via a Batch Process

This topic explains how to set up a publisher with subscribers using the Create Publisher and Subscriber Records (R00960) batch process. This simplifies replication setup by allowing you to create hundreds of subscriber records with this one process.

This batch process reads the Machine Identification (F00960) table, which stores information about each OneWorld workstation. From information that you type into the processing option on how to create a publisher, you can create a publisher record and subscriber records for each workstation in F00960. If you do not want all workstations set up as subscribers, then use data selection to create records for a particular set of workstations. Remember that you should normally create the publisher and its subscribers as disabled and not synchronized. When you are ready to use replication in your production environment, you can enable and synchronize (if the subscriber is a server) the records.

After you create a publisher with its many subscribers, you can use the copy feature in the Data Replication application to create additional publisher/subscriber records. This process should save you from manually entering hundreds of subscriber records.

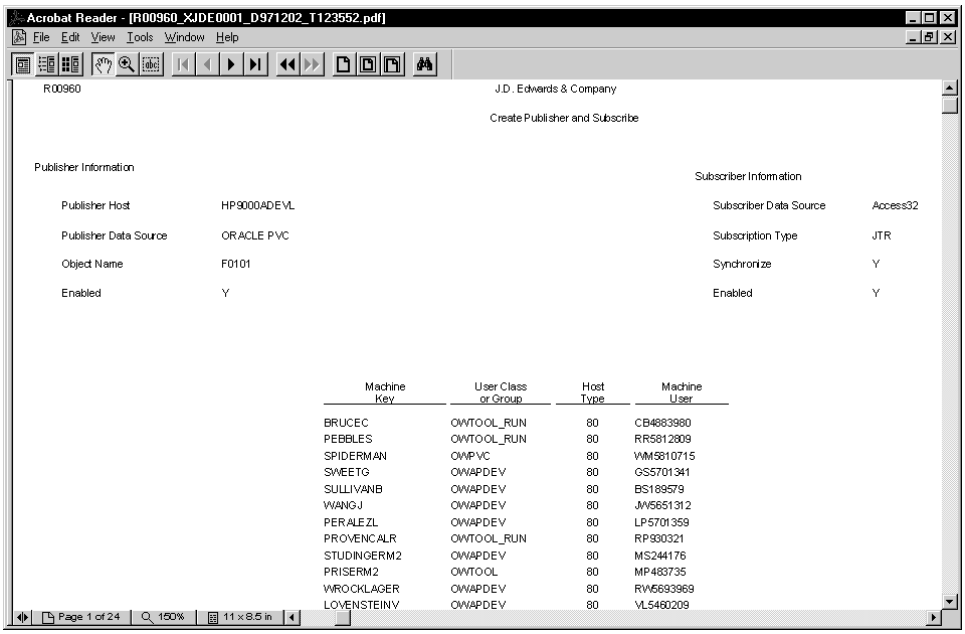
▶ To set up data replication with a batch process

On the Advanced Operations (GH9012) menu

1. Choose Create Publisher and Subscriber Records (R00960).
2. On the Work With Batch Versions form, highlight a version, and click Select. The versions are set up to run in either Proof or Final mode.
3. On the Versions Prompting form, click Data Selection, and click Submit.
4. On the Criterion Design form, choose from the appropriate columns to specify the publisher and subscribers.
5. On the Processing Options form, enter information into the following fields:

- Option 1, enter a '1' if you want to run the report in Proof mode or leave this field blank to run the report in Final mode. Versions have already been set up for this.
- Option 2, complete the following fields for your publisher record:
  - Publisher Data Source
  - Object Name
  - Enabled
- Option 3, complete the following fields for your subscriber records:
  - Subscriber Data Source
  - Subscription Type
  - Enabled
  - Synchronize

The following is an example of this report:



The following are possible errors that can occur when running this report:

- Error Inserting the Publisher Record (the publisher record could not be written)
- Error Inserting the Above Subscriber Record (the previous subscriber record could not be written)

Copying a Publisher and its Associated Subscribers

After you have added one publisher and its subscribers, you can copy that publisher to create new publishers. This means you do not have to re-add subscribers (possibly in the hundreds) to new publishers.

Remember that the easiest way to create the initial publisher/subscriber records from which to copy from is to run the Create Publisher and Subscriber Records (R00960) batch process, which reads the Machine Identification (F00960) table to create subscriber records for all workstations.

▶ To copy a publisher and its associated subscribers

On the System Administration Tools (GH9011) menu

1. Choose Data Replication (P98DREP).
2. On the Work With Publishers form, highlight a publisher, and click Copy.
3. On the Publisher Revisions form, change data source, object name information, or both. If you change the data source, and the changed data source is on a different machine, the Host field changes.
4. Click OK.

   OneWorld creates the new publisher and all of its associated subscribers.

Setting Up the Environment for Data Replication

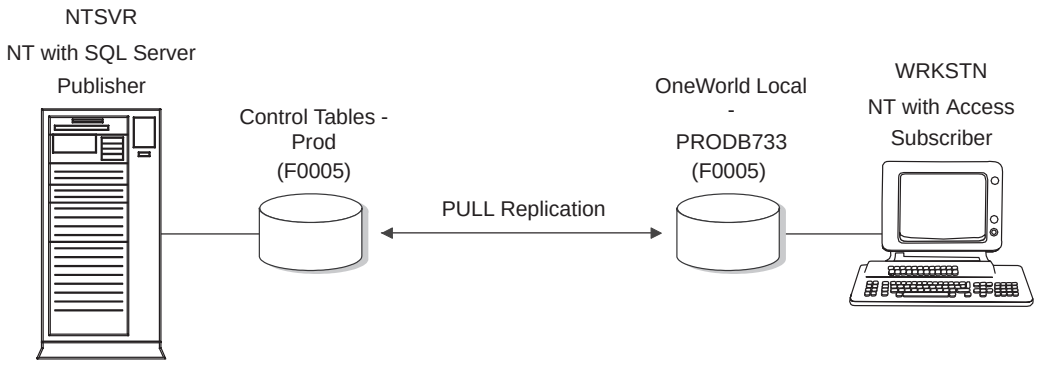
Before using OneWorld Data Replication, you must set up your environments to use the replication application. This process involves setting up your Object Configuration Manager mappings, your data sources, and the JDBNET database driver (if data tables will reside on different computing platforms).

This section provides information about the setup of the following example configurations:

- ❑ Setting Up a One-Tier Replication Environment
- ❑ Setting Up a Two-Tier Replication Environment
- ❑ Setting Up Replication to a Non-OneWorld Workgroup Server
- ❑ Setting Up Two-Tier Replication with Different Platforms

Setting up a One-Tier Replication Environment

This example uses simple one-tier PULL replication between a OneWorld server and a local workstation.



Setting up OCM Mappings for One-Tier Replication

In order for Data Replication to function correctly, it is important that you have only one centralized copy of the Data Replication tables (F98DRPUB, F98DRSUB, and F98DRENV). These central tables are typically stored in the System - B733 data source. You should have active OCM mappings for *PUBLIC on all platforms for all environments that point to these tables.

If these records are not present on a server, when that server receives a replication message it is likely that the error Publisher Cache entry not found will be written to the JDE.LOG file, because the server's cache will not contain the correct publisher and subscriber information. If you receive this message, you can verify the location of the F98DRPUB and F98DRSUB tables by looking in the JDEDEBUG.LOG file.

Publishers should not have any OCM mappings for their F98DRLOG and F98DRPCN tables, but subscribers should have mappings that point to their publisher's Server Map data source for these tables.

The Server Map for the publisher (NTSRV - B733 Server Map) should have the following ACTIVE mappings in the Object Configuration Master (F986101) table:

Environ	Object	Data Source	User
PRD733	F98DRPUB	System - B733	*PUBLIC
PRD733	F98DRSUB	System - B733	*PUBLIC
PRD733	F98DRENV	System - B733	*PUBLIC

These mappings exist so that the publisher is able to locate the Publisher and Subscriber definitions and cache them when OneWorld is started on this machine.

The System data source for the subscriber workstations (System B733) should have the following ACTIVE mappings in the Object Configuration Master (F986101) table:

Environ	Object	Data Source	User
PRD733	F98DRPUB	System - B733	*PUBLIC
PRD733	F98DRSUB	System - B733	*PUBLIC
PRD733	F98DRENV	System - B733	*PUBLIC
PRD733	F98DRLOG	NTSRV - B733 Server Map	*PUBLIC
PRD733	F98DRPCN	NTSRV - B733 Server Map	*PUBLIC

The mappings for F98DRPUB, F98DRSUB, and F98DRENV exist so that the subscribers are able to locate the Publisher and Subscriber definitions and cache them when OneWorld is started on those machines. The mappings for F98DRLOG and F98DRPCN exist so that the subscribers are able to read their pending changes from the publisher machine when a change has been made to a published table. In some cases, you may not have OCM mappings for F98DRENV, but your data replication can still function properly.

Setting up Data Sources for One-Tier Replication

At a minimum, you should have data sources defined for each data source being pointed to in your OCM mappings shown in the previous section. In addition, each publisher should have a data source for the published data and each subscriber should have a data source for the subscribed data.

The Server Map data source for the publisher (NTSRV - B733 Server Map) should have the following data source definitions in the Data Source Master (F98611) table:

Type	Data Source Name	Server	Database Name	Owner	DLL Name
DB	System - B733	NTSRV	System - B732	SYSB733	JDBODBC.DLL
DB	Control Tables - Prod	NTSRV	Control Tables - Prod	PRODDTA	JDBODBC.DLL

The data source for System - B733 is how the publisher machine accesses the F98DRPUB, F98DRSUB, and F98DRENV tables. The data source for Control Tables - Prod is how the publisher machine accesses the published table (F0005).

The System data source for the subscriber workstations (System - B733) should include the following data source definitions in the Data Source Master (F98611) table:

Type	Data Source Name	Server	Database Name	Owner	DLL Name
DB	System - B733	NTSRV	System - B732	SYSB733	JDBODBC.DLL
SVR	NTSRV	NTSRV	NTSRV - B733 Server Map	SVMB733	JDBODBC.DLL
DB	NTSRV - B733 Server Map	NTSRV	NTSRV - B733 Server Map	SVMB733	JDBODBC.DLL
DB	OneWorld Local - PRODB733	LOCAL	OneWorld Local - PRODB733		JDBODBC.DLL

The data source for System - B733 is how the subscriber machines access the F98DRPUB, F98DRSUB, and F98DRENV tables. The data sources for NTSRV and NTSRV - B733 Server Map are how the subscriber machines access the F98DRLOG and F98DRPCN tables on the publisher machine. The data source for OneWorld Local - PRODB733 is how the subscriber accesses the subscribed table (F0005) that is being kept locally.

Setting up Publisher and Subscriber Records

The publisher machine is defined through the Work with Publishers application, and a record for each publisher is entered in the F98DRPUB table. A publisher record for simple one-tier replication would appear as follows:

Publishing Machine	**Published Data Source**	**Object**
NTSRV	Control Tables - Prod	F0005

This record indicates that the NTSRV machine is the publisher for the F0005 table in its Control Tables - Prod data source.

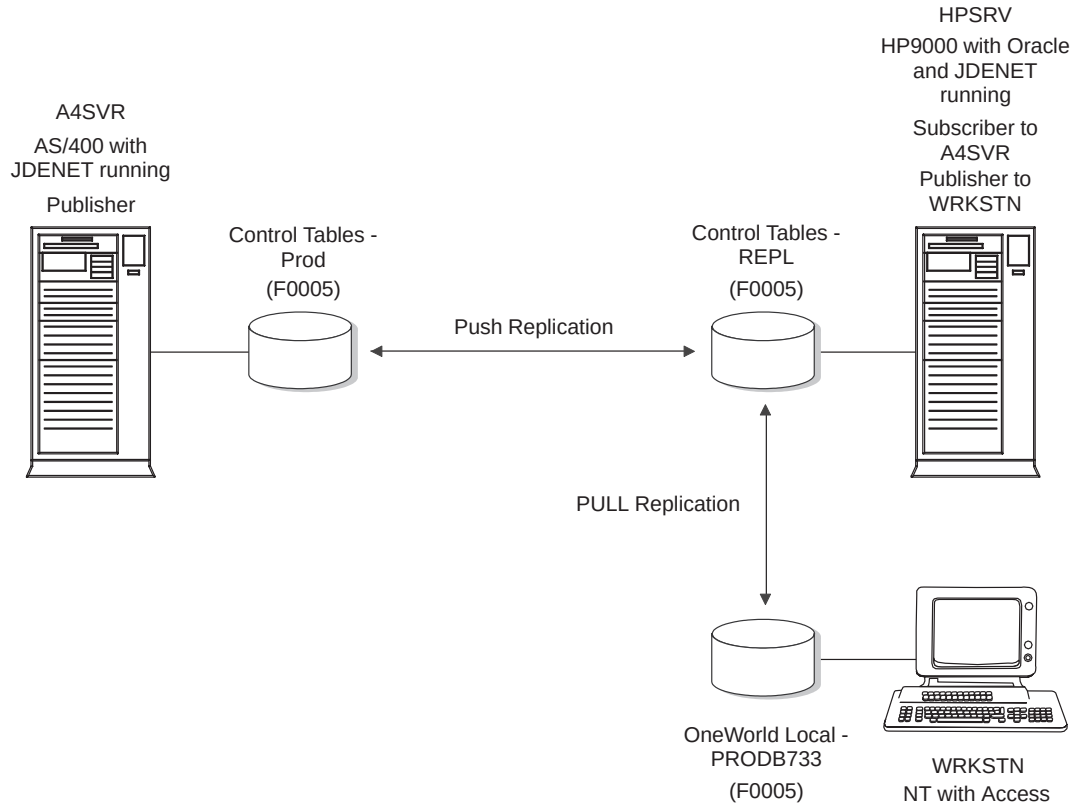
Subscriber machines are defined through the Work with Subscribers application, and a record for each subscriber is entered in the F98DRSUB table. A subscriber record for this example would appear as follows:

Subscriber Machine	**Subscriber Data Source**	**Sub Type**	**Sync**	**Enabled**	**Hold**
WRKSTN	OneWorld Local - PRODB733	PUL	N	Y	N

This record indicates that the WRKSTN machine is a PULL subscriber of the previously defined publisher. WRKSTN will keep its copy of the subscribed data in its OneWorld Local - PRODB733 data source. Currently the data replication application knows that replication is enabled for this subscriber and that the subscribed table is out of sync with the published table.

Setting up a Two-Tier Replication Environment

This example is a two-tier configuration, with PUSH replication from one OneWorld server (enterprise server) to another OneWorld server (workgroup server) and PULL replication between the workgroup server and a local workstation.



Setting up OCM Mappings for Two-Tier Replication

In order for Data Replication to function correctly, it is important that you have only one centralized copy of the Data Replication tables (F98DRPUB, F98DRSUB, and F98DRENV). These central tables are typically stored in the System - B733 data source. You should have active OCM mappings for *PUBLIC on all platforms for all environments that point to these tables.

Publishers should not have any OCM mappings for their F98DRLOG and F98DRPCN tables, but subscribers should have mappings that point to their publisher's Server Map data source for these tables.

The Server Map data source for the enterprise server (NTSRV - B733 Server Map) should have the following ACTIVE mappings in the Object Configuration Master (F986101) table:

Environ	Object	Data Source	User
PRD733	F98DRPUB	System - B733	*PUBLIC
PRD733	F98DRSUB	System - B733	*PUBLIC
PRD733	F98DRENV	System - B733	*PUBLIC

These mappings exist to enable the publisher machine to locate the Publisher and Subscriber definitions and cache the definitions when OneWorld begins to run on this machine.

The Server Map data source for the workgroup server (WGSRV - B733 Server Map) should have the following ACTIVE mappings in the Object Configuration Master (F986101) table:

Environ	Object	Data Source	User
PRD733	F98DRPUB	System - B733	*PUBLIC
PRD733	F98DRSUB	System - B733	*PUBLIC
PRD733	F98DRENV	System - B733	*PUBLIC

The mappings for F98DRPUB, F98DRSUB, and F98DRENV exist to enable the WGSRV publisher to locate the Publisher and Subscriber definitions and cache the definitions when OneWorld begins to run on this machine.

The System data source for the subscriber workstations (System - B733) should have the following ACTIVE mappings in the Object Configuration Master (F986101) table:

Environ	Object	Data Source	User
PRD733	F98DRPUB	System - B733	*PUBLIC
PRD733	F98DRSUB	System - B733	*PUBLIC
PRD733	F98DRENV	System - B733	*PUBLIC

The mappings for F98DRPUB, F98DRSUB, and F98DRENV exist to enable the subscribers to locate the Publisher and Subscriber definitions and cache the definitions when OneWorld begins to run on those machines.

Setting up Data Sources for Replication

At a minimum, you should have data sources defined for each data source being pointed to in your OCM mappings shown in the previous section. In addition, each publisher should have a data source for the published data and each subscriber should have a data source for the subscribed data.

The Server Map data source for the publisher machines should have the following data source definitions.

The Server Map data source for the enterprise server (NTSRV - B733 Server Map) should have the following data source definitions in the Data Source Master (F98611) table:

Type	Data Source Name	Server	Database Name	Owner	DLL Name
DB	System - B733	NTSRV	System - B732	SYSB733	JDBODBC.DLL
DB	Control Tables - Prod	NTSRV	Control Tables - Prod	PRODDTA	JDBODBC.DLL

The data source for System - B733 is how the publisher machine (NTSRV) accesses the F98DRPUB, F98DRSUB, and F98DRENV tables. The data source for Control Tables - Prod is how the NTSRV publisher machine accesses the published table (F0005).

The Server Map data source for the workgroup server (WGSRV - B733 Server Map) should have the following data source definitions in the Data Source Master (F98611) table:

Type	Data Source Name	Server	Database Name	Owner	DLL Name
DB	System - B733	NTSRV	System - B732	SYSB733	JDBODBC.DLL
SVR	NTSRV	NTSRV	NTSRV - Server Map	SVMB733	JDBODBC.DLL
DB	NTSRV - Server Map	NTSRV	NTSRV - Server Map	SVMB733	JDBODBC.DLL
DB	Control Tables - REPL	WGSRV	Control Tables - REPL	PRODDTA	JDBODBC.DLL

The data source for System - B733 is how the publisher machine (WGSRV) accesses the F98DRPUB, F98DRSUB, and F98DRENV tables. The data sources for NTSRV and NTSRV - Server Map are needed in order to access the F98DRLOG and F98DRPCN tables on the publisher machine. The data source for Control Tables - REPL is how the WGSRV publisher machine accesses its published table (F0005).

The System data source for the subscriber workstations (System - B733) should include the following data source definitions in the Data Source Master (F98611) table:

Type	Data Source Name	Server	Database Name	Owner	DLL Name
DB	System - B733	NTSRV	System - B732	SYSB733	JDBODBC.DLL
SVR	WGSRV	WGSRV	WGSRV - B733 Server Map	SVMB733	JDBODBC.DLL
DB	WGSRV - B733 Server Map	WGSRV	WGSRV - B733 Server Map	SVMB733	JDBODBC.DLL
DB	OneWorld Local - PRODB733	LOCAL	OneWorld Local - PRODB733		JDBODBC.DLL

The data source for System - B733 is how the subscriber machines access the F98DRPUB, F98DRSUB, and F98DRENV tables. The data sources for WGSRV and WGSRV - B733 Server Map are required in order to access the F98DRLOG and F98DRPCN tables on the publisher machine (WGSRV). The data source for OneWorld Local - PRODB733 is how the subscriber accesses the subscribed table (F0005) that is being kept locally.

Setting up Publisher and Subscriber Records for Replication

The publisher machines are defined through the Work with Publishers application, and a record for each publisher is entered in the F98DRPUB table. Publisher records for two-tier replication would appear as follows:

Publishing Machine	Published Data Source	Object
NTSRV	Control Tables - Prod	F0005
WGSRV	Control Tables - REPL	F0005

The first record indicates that the NTSRV machine is the publisher for the F0005 table in the Control Tables - Prod data source and the WGSRV machine will be the subscriber. The second record indicates that, in addition to being a subscriber to the NTSRV machine, the WGSRV machine is the publisher for the F0005 table in the Control Tables - REPL data source and the local workstations will be the subscribers.

Subscriber machines are defined through the Work with Subscribers application, and a record for each subscriber is entered in the F98DRSUB table. Subscriber records for this example would appear as follows:

Subscriber Machine	Subscriber Data Source	Sub Type	Sync	Enabled	Hold
WGSRV	Control Tables - REPL	PUS	Y	Y	N

This record indicates that the WGSRV machine is a push subscriber of the NTSRV publisher machine. WGSRV will keep its copy of the subscribed data in the Control Tables - REPL data source. Currently the data replication application knows that replication is enabled for this subscriber and that the subscribed table is in synchronization with the published table.

Subscriber Machine	Subscriber Data Source	Sub Type	Sync	Enabled	Hold
WRKSTN	OneWorld Local - PRODB733	PUL	N	Y	N

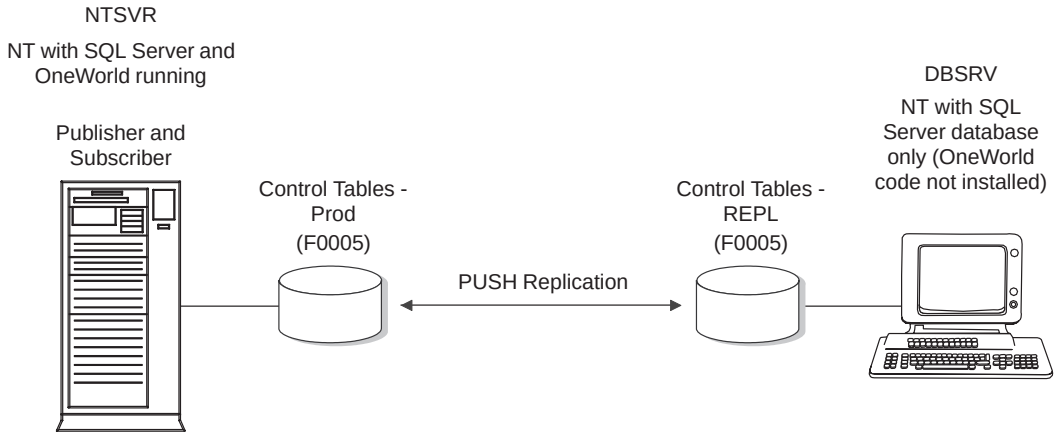
This record indicates that the WRKSTN machine is a pull subscriber of the WGSRV publisher machine. WRKSTN will keep its copy of the subscribed data in the OneWorld Local - PRODB733 data source. Currently the data replication application knows that replication is enabled for this subscriber and that the subscribed table is out of synchronization with the published table.

Setting up Replication to a Non-OneWorld Workgroup Server

The third example configures PUSH replication from a OneWorld enterprise server to a non-OneWorld database-only workgroup server.

Do not use this mode of replication on an AS/400 workgroup server. To act as a publisher, the AS/400 workgroup needs JDBNET to access the database server. OneWorld needs to reside on the workgroup server for JDBNET to run on the workgroup server.

Also, do not use an Alpha server as a publisher for an AS/400 subscriber. Alpha servers do not use Client Access, therefore, the publisher is unable to update data on the AS/400.



Setting up OCM Mappings for Data Server Replication

In order for Data Replication to function correctly, it is important that you have only one centralized copy of the Data Replication tables (F98DRPUB, F98DRSUB, and F98DRENV). These central tables are typically stored in the System - B733 data source. You should have active OCM mappings for *PUBLIC on all platforms for all environments that point to these tables.

Publishers should not have any OCM mappings for their F98DRLOG and F98DRPCN tables, but subscribers should have mappings that point to their publisher's Server Map data source for these tables.

The Server Map data source (NTSRV - B733 Server Map) for the publisher machine should have the following ACTIVE mappings in the Object Configuration Master (F986101) table:

Environ	Object	Data Source	User
PRD733	F98DRPUB	System - B733	*PUBLIC
PRD733	F98DRSUB	System - B733	*PUBLIC
PRD733	F98DRENV	System - B733	*PUBLIC

These mappings exist to enable the publisher machine to locate the Publisher and Subscriber definitions and cache the definitions when OneWorld begins to run on this machine.

The OCM mappings required by workstations are stored in the System - B733 data source, and should the following ACTIVE mappings.

The System data source (System - B733) for the subscriber workstations should have the following ACTIVE mappings in the Object Configuration Master (F986101) table:

Environ	Object	Data Source	User
PRD733	F98DRPUB	System - B733	*PUBLIC
PRD733	F98DRSUB	System - B733	*PUBLIC
PRD733	F98DRENV	System - B733	*PUBLIC

The mappings for F98DRPUB, F98DRSUB, and F98DRENV exist to enable the subscribers to locate the Publisher and Subscriber definitions and cache the definitions when OneWorld begins to run on those machines.

Setting up Data Sources for Replication

At a minimum, you should have data sources defined for each data source being pointed to in your OCM mappings shown in the previous section. In addition, each publisher should have a data source for the published data and each subscriber should have a data source for the subscribed data.

The Server Map data source (NTSRV - B733 Server Map) for the publisher machine should have the following data source definitions in the Data Source Master (F98611) table:

Type	Data Source Name	Server	Database Name	Owner	DLL Name
DB	System - B733	NTSRV	System - B732	SYSB733	JDBODBC.DLL
DB	Control Tables - Prod	NTSRV	Control Tables - Prod	PRODDTA	JDBODBC.DLL
DB	Control Tables - REPL	NTSRV	Control Tables - REPL	REPLDTA	JDBODBC.DLL

The data source for System - B733 is how the publisher machine (NTSRV) accesses the F98DRPUB, F98DRSUB, and F98DRENV tables. The data source for Control Tables - Prod is how the *publisher* logic on NTSRV accesses the published table (F0005).

The data source for Control Tables - REPL is how the subscriber logic on NTSRV accesses the subscribed table on the workgroup server (DBSVR). Since the subscriber logic is running on NTSVR, then NTSVR must have an ODBC data source definition that points to the SQL Server database on DBSVR. The ODBC data source definition is the where DBSVR is identified as the machine on which the data tables actually reside. OneWorld only needs to know about the NTSVR machine, since it is both the publisher and subscriber.

Setting up Publisher and Subscriber Records for Replication

The publisher machines are defined through the Work with Publishers application, and a record for each publisher is entered in the F98DRPUB table. The Publisher record for replication to a database-only workgroup server would appear as follows:

Publishing Machine	Published Data Source	Object
NTSRV	Control Tables - Prod	F0005

This record indicates that the NTSRV machine is the publisher for the F0005 table in its Control Tables - Prod data source.

Subscriber machines are defined through the Work with Subscribers application, and a record for each subscriber is entered in the F98DRSUB table. The Subscriber record for this example would appear as follows:

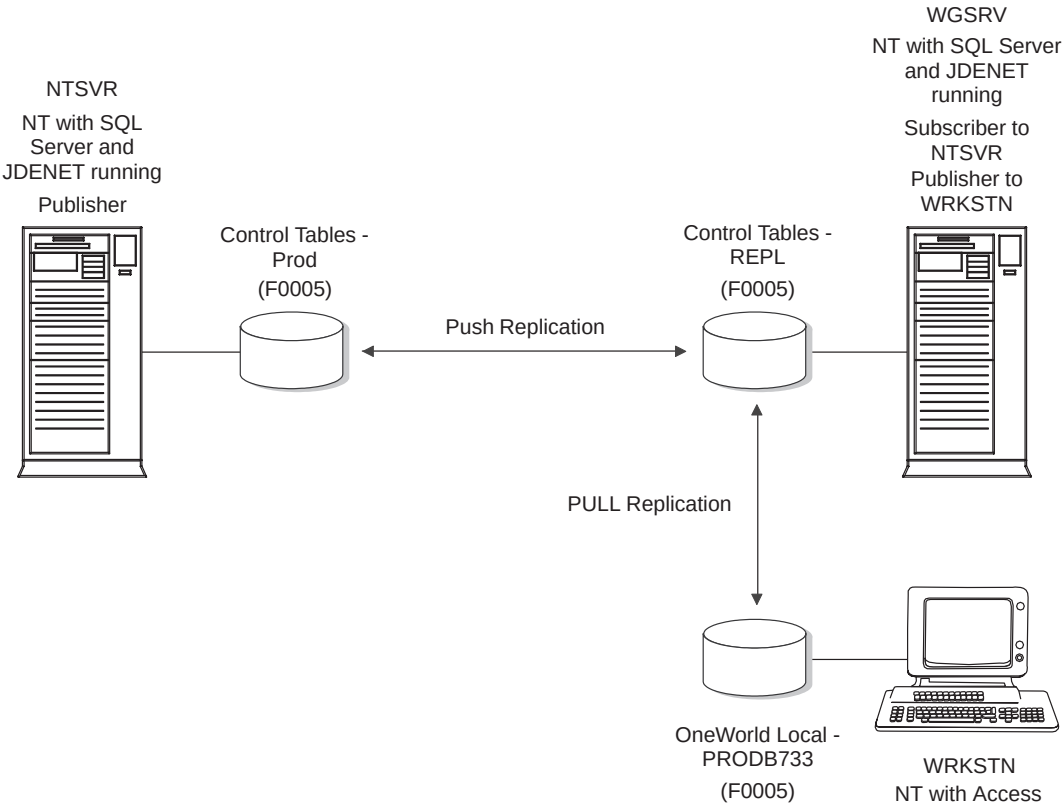
Subscriber Machine	Subscriber Data Source	Sub Type	Sync	Enabled	Hold
NTSVR	Control Tables - REPL	PUS	Y	Y	N

This record indicates that the NTSRV machine is a PUSH subscriber of the NTSRV *publisher* machine. NTSRV will access its copy of the subscribed data through its Control Tables - REPL data source. Currently the data replication application knows that replication is enabled for this subscriber and that the subscribed table is in sync with the published table.

You can point workstations to the database-only workgroup server to read the replicated F0005 table, but you should not allow a workstation to insert, update, and delete data in this table. The workgroup server does not run OneWorld; therefore, the server cannot communicate changes in a replicated table to the publisher. If the workstation changes the replicated table on the workgroup server, the table on the enterprise server and the table on the workgroup server will no longer be in synchronization. Workstations that use a database-only workgroup server should use the replicated tables for validation purposes only. The replicated tables on the workgroup server should be thought of as read-only copies of the published tables.

Setting up Two-Tier Replication with Different Platforms

The fourth example is a two-tier configuration with PUSH replication between two OneWorld servers that run on different platforms, and PULL replication between the workgroup server and a local workstation. This configuration will require the use of the JDBNET driver to communicate between differing platforms.



Setting up OCM Mappings for JDBNET Replication

In order for Data Replication to function correctly, it is important that you have only one centralized copy of the Data Replication tables (F98DRPUB, F98DRSUB, and F98DRENV). These central tables are typically stored in the System - B733 data source. You should have active OCM mappings for *PUBLIC on all platforms for all environments that point to these tables.

Publishers should not have any OCM mappings for their F98DRLOG and F98DRPCN tables, but subscribers should have mappings that point to their publisher's Server Map data source for these tables.

The Server Map data source (A4SRV - B733 Server Map) for the AS/400 should have the following ACTIVE mappings in the Object Configuration Master (F986101) table:

Environ	Object	Data Source	User
PRD733	F98DRPUB	System - B733	*PUBLIC
PRD733	F98DRSUB	System - B733	*PUBLIC
PRD733	F98DRENV	System - B733	*PUBLIC

These mappings exist to enable the publisher machine to locate the Publisher and Subscriber definitions and cache the definitions when OneWorld begins to run on this machine.

The Server Map data source (HPSRV - B733 Server Map) for the HP9000 should have the following ACTIVE mappings in the Object Configuration Master (F986101) table:

Environ	Object	Data Source	User
PRD733	F98DRPUB	System - B733	*PUBLIC
PRD733	F98DRSUB	System - B733	*PUBLIC
PRD733	F98DRENV	System - B733	*PUBLIC

The mappings for F98DRPUB, F98DRSUB, and F98DRENV exist to enable the publisher machine to locate the Publisher and Subscriber definitions and cache the definitions when OneWorld begins to run on this machine.

The System data source (System - B733) for the subscriber workstations should have the following ACTIVE mappings in the Object Configuration Master (F986101) table:

Environ	Object	Data Source	User
PRD733	F98DRPUB	System - B733	*PUBLIC
PRD733	F98DRSUB	System - B733	*PUBLIC
PRD733	F98DRENV	System - B733	*PUBLIC

The mappings for F98DRPUB, F98DRSUB, and F98DRENV exist to enable the subscribers to locate the Publisher and Subscriber definitions and cache the definitions when OneWorld begins to run on those machines.

Setting up Data Sources for Replication

At a minimum, you should have data sources defined for each data source being pointed to in your OCM mappings shown in the previous section. In addition, each publisher should have a data source for the published data and each subscriber should have a data source for the subscribed data.

The Server Map data source (A4SRV - B733 Server Map) for the AS/400 should have the following data source definitions in the Data Source Master (F98611) table:

Type	Data Source Name	Server	Database Name	Owner	DLL Name
DB	System - B733	A4SRV	System - B732	SYSB733	DBDR
DB	Control Tables - Prod	A4SRV	Control Tables - Prod	PRODDTA	DBDR

The data source for System - B733 is how the publisher machine (A4SRV) accesses the F98DRPUB, F98DRSUB, and F98DRENV tables. The data source for Control Tables - Prod is how the A4SRV publisher machine accesses the published table (F0005). The DBDR database driver (DLL Name) is being used for these data sources on the AS/400 server.

The Server Map data source (HPSRV - B733 Server Map) for the HP9000 should have the following data source definitions in the Data Source Master (F98611) table:

Type	Data Source Name	Server	Database Name	Owner	DLL Name
DB	System - B733	A4SRV	System - B732	SYSB733	LIBJDBNET.SL
SVR	A4SRV	A4SRV	A4SRV - Server Map	SVMB733	LIBJDBNET.SL
DB	A4SRV - Server Map	A4SRV	A4SRV - Server Map	SVMB733	LIBJDBNET.SL
DB	Control Tables - REPL	HPSRV	Control Tables - REPL	PRODDTA	LIBORA73.SL

The data source for System - B733 is how the publisher machine (HPSRV) accesses the F98DRPUB, F98DRSUB, and F98DRENV tables. The data sources for A4SRV and A4SRV - Server Map are needed in order to access the F98DRLOG and F98DRPCN tables on the publisher machine. The data source for Control Tables - REPL is how the HPSRV publisher machine accesses its published table (F0005). The LIBJDBNET.SL database driver (DLL Name) is being used for the data sources residing on the AS/400, while the LIBORA73.SL database driver (DLL Name) is being used for the data sources residing in Oracle.

The System data source (System - B733) for the subscriber workstations should have the following data source definitions in the Data Source Master (F98611) table:

Type	Data Source Name	Server	Database Name	Owner	DLL Name
DB	System - B733	A4SRV	System - B732		JDBODBC.DLL
SVR	HPSRV	HPSRV	HPSRV - B733 Server Map	SVMB733	JDBOCI73.DLL
DB	HPSRV - B733 Server Map	HPSRV	HPSRV - B733 Server Map	SVMB733	JDBOCI73.DLL
DB	OneWorld Local - PRODB733	LOCAL	OneWorld Local - PRODB733		JDBODBC.DLL

The data source for System - B733 is how the subscriber machines access the F98DRPUB, F98DRSUB, and F98DRENV tables. The data sources for HPSRV and HPSRV - B733 Server Map are required in order to access the F98DRLOG and F98DRPCN tables on the publisher machine (HPSRV). The data source for OneWorld Local - PRODB733 is how the subscriber accesses the subscribed table (F0005) that is being kept locally. The JDBODBC.DLL database driver DLL Name is being used to access LOCAL and AS/400 data sources, while the JDBOCI73.DLL database driver DLL is being used to access Oracle.

Setting up Publisher and Subscriber Records for Replication

The publisher machines are defined through the Work with Publishers application, and a record for each publisher is entered in the F98DRPUB table. Publisher records for two-tier replication would appear as follows:

Publishing Machine	Published Data Source	Object
A4SRV	Control Tables - Prod	F0005
HPSRV	Control Tables - REPL	F0005

The first record indicates that the A4SRV machine is the publisher for the F0005 table in its Control Tables - Prod data source, and the HPSRV machine will be the subscriber. The second record indicates that, in addition to being a subscriber to the A4SRV machine, the HPSRV machine is the publisher for the F0005 table in the Control Tables - REPL data source, and the local workstations will be the subscribers.

Subscriber machines are defined through the Work with Subscribers application, and a record for each subscriber is entered in the F98DRSUB table. Subscriber records for this example would appear as follows:

Subscriber Machine	Subscriber Data Source	Sub Type	Sync	Enabled	Hold
HPSRV	Control Tables - REPL	PUS	Y	Y	N

This record indicates that the HPSRV machine is a PUSH subscriber of the A4SRV publisher machine. HPSRV will keep a copy of the subscribed data in the Control Tables - REPL data source. Currently, the data replication application knows that replication is enabled for this subscriber and that the subscribed table is in synchronization with the published table.

Subscriber Machine	Subscriber Data Source	Sub Type	Sync	Enabled	Hold
WRKSTN	OneWorld Local - PRODB733	PUL	N	Y	N

This record indicates that the WRKSTN machine is a PULL subscriber of the HPSRV publisher machine. WRKSTN will keep a copy of the subscribed data in the OneWorld Local - PRODB733 data source. Currently the data replication application knows that replication is enabled for this subscriber and that the subscribed table is out of synchronization with the published table.

Modifying the Workstation and Server jde.ini Files

You can also set up forced synchronization. Generally, your workstation will run a forced synchronization the first time anyone signs on to ensure that the replicated tables on the publisher and the subscriber match. If you require the synchronization of all replicated tables, you can manually set the ForcedSync setting in your jde.ini file to perform a synchronization. Complete the following tasks:

- Modify the workstation jde.ini files
- Modify the server jde.ini file
- Setting up a workstation for forced synchronization

▶ To modify the workstation jde.ini files

For all workstations that have OneWorld installed, you must change the workstation's jde.ini file in one of two ways:

- Manually change the jde.ini file on each workstation (not recommended).
- Change the jde.ini file on the deployment server and re-deploy a package. You can deploy an update package with the Replace JDE.ini flag checked, or you can deploy a full or partial package, because they also replace the jde.ini file.

Whichever method you choose, complete the following:

1. Locate the jde.ini file, either on each workstation, or on the deployment server in the release share path:

   \\B732\client\misc\jde.ini

2. Using an ASCII editor, such as Notepad, view the jde.ini file to ensure the accuracy of the following settings:

```
[SECURITY]
DefaultEnvironment=default environment name


[DEBUG]
RepTrace=replication trace 1/0 (ON/OFF)
```

▶ To modify the server jde.ini file

1. Locate your server jde.ini file.
2. Using an ASCII editor, such as Notepad, view the jde.ini file to ensure accuracy of the following settings:

```
[SECURITY]
User=user ID
Password=database password
DefaultEnvironment=default environment name


[DEBUG]
RepTrace=replication trace 1 or 0 (for ON/OFF)
```

Variable	Description
user ID	This is the database ID that has access to the replication files (F98DRPUB, F98DRSUB, F98DRENV).

Variable	Description
database password	This is the database password that has access to the replication files (F98DRPUB, F98DRSUB, F98DRENV).
default environment name	Any valid environment for the path code in which the publisher resides.
replication trace 0/1 (off/on)	You can enable replication trace if you want to perform troubleshooting on your replication process. When you enable this trace, the replication process sends additional information to the jde.log file. Do not leave replication trace on permanently, because the jde.log file will become too large. The valid values are: 0 = OFF 1 = ON

▶ To set up a workstation for forced synchronization

1. Locate the workstation jde.ini file.
2. Using an ASCII editor, such as Notepad, set the value for the following setting to 1:

```
[REPLICATION]
ForcedSync=forced synchronization 0/1 (off/on)
```

The next time you sign on to OneWorld on the machine, OneWorld will perform a forced synchronization of all replicated tables. After the forced synchronization, OneWorld automatically turns off the setting.

Note: If the [REPLICATION] section does not contain the ForcedSync setting, OneWorld performs a forced synchronization, then OneWorld automatically writes ForcedSync=0 into the [REPLICATION] section.

Variable	Description
forced synchronization 0/1 (off/on)	You can set a machine to perform a forced synchronization of replicated tables. When you force synchronization, OneWorld copies all replicated tables from the publisher to the subscriber when you sign on to OneWorld on the machine. The valid values are: • 0=off • 1=on

Viewing the Replication Logs

You can view the various replication logs for:

- ❑ Outstanding changes for a given host
- ❑ Publisher changes the Pending Change Log did not receive
- ❑ Pull subscribers that have not retrieved their changes

Viewing Outstanding Changes for a Given Host

View the Replication Change Log (F98DRLOG) to see all changes to published tables (for the selected host) that have not been delivered to all enabled subscribers. The Pending Deliveries field displays the subscribers which have not received a particular change.

▶ To view outstanding changes for a given publisher

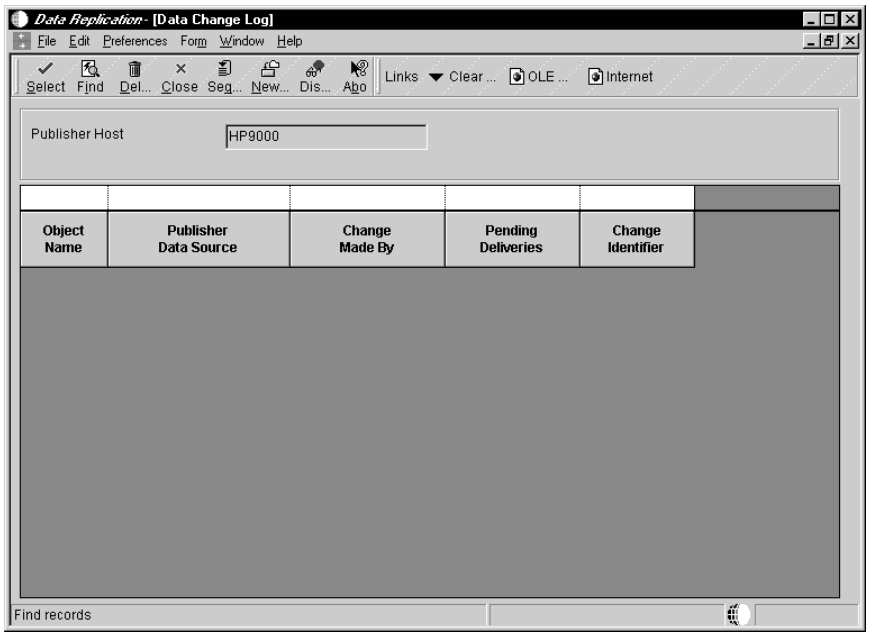
On the System Administration Tools (GH9011) menu

1. Choose Data Replication (P98DREP).

   The Work With Publishers form appears.

2. Choose Data Change Log from the Form menu.
3. On the Machine Search & Select form, highlight the publisher host, and click Select.

   The Data Change Log form appears. This form shows the number of pending messages the replication process needs to deliver for a given change.



4. To view the subscribers waiting to receive a change, highlight a row, and click Select. See *Viewing Pull Subscribers that have not Retrieved Their Changes* in this guide for information.
5. You can delete any log entries that you know the replication process will never send. However, be sure that the entries in the Pending Change Notification Log are deleted *first*. To verify, on the Pending Deliveries form, select the log entry and then delete any pending deliveries shown. Then on the Data Change Log form, highlight the log, and click Delete.

Field	Explanation
Change Made By	The name of the network server where data resides or where objects can be executed. . . . . . *Form-specific information* . . . . . . On this form, the Change Made By field indicates the machine name where the change was initiated. You can use this information to track down the person who used this machine to change a publisher table.
Pending Deliveries	Number of pending deliveries for a particular change to a published object.

Field	Explanation
Change Identifier	A generic ID for FDA and RDA specifications. . . . . . *Form-specific information* . . . . . . A unique number used to retrieve the modified data from the Data Change Log table. On this form, the Change Identifier is used to track a change made to a published table. You can use this number to see which changes have not been delivered to specific subscribers.

Viewing Publisher Changes the Pending Change Log did not Receive

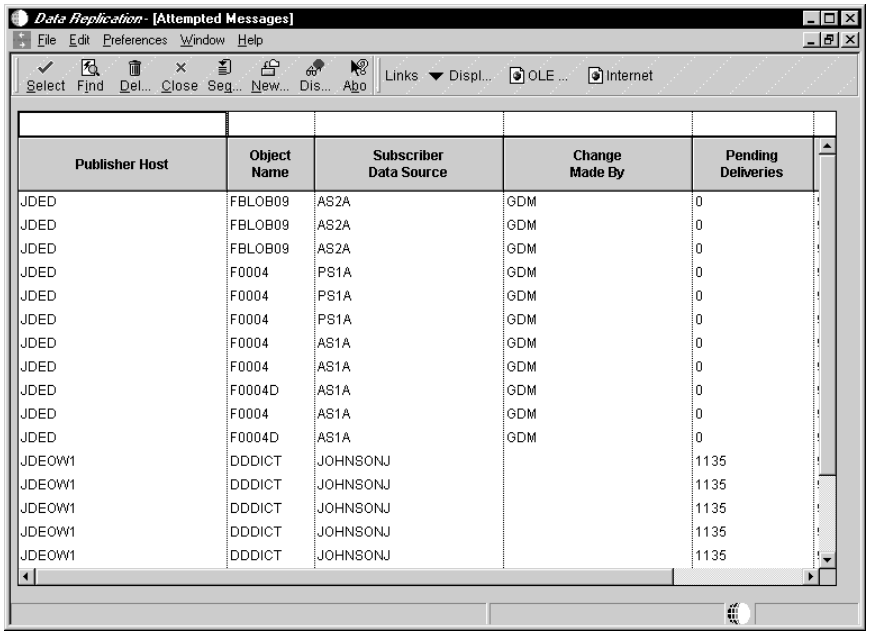
View this log to find changes that the Pending Change Notification Log (F98DRPCN) never received. For example, if a user makes a change to a table, the workstation writes this change to the Replication Change Log (F98DRLOG). But if OneWorld on that server is down, then the Pending Change Notification Log does not receive that message. The change is then stored in the Replication Change Log and the Pending Change Notification Log in the system data source. OneWorld replicates these changes to subscribers the next time the machine that made the changes signs onto OneWorld.

▶ To view publisher changes the Pending Change Log did not receive

On the System Administration Tools (GH9011) menu

1. Choose Data Replication (P98DREP).
2. On the Work With Publishers form choose Attempted Messages from the Form menu.

The Attempted Messages form appears.



OneWorld stores Attempted Messages in the Data Replication Change Log (F98DRLOG) table in the system data source. You should monitor these messages regularly, addressing those that are still relevant, and removing those that are not.

Viewing Pull Subscribers that have not Retrieved Their Changes

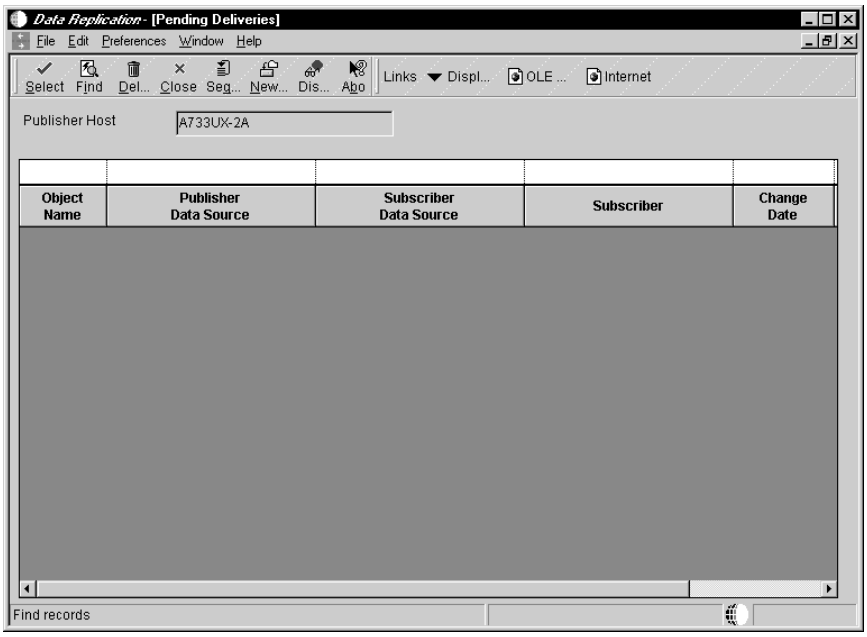
This log allows you to view pull subscribers that have not signed onto OneWorld since a publisher table changed.

▶ To view pull subscribers that have not retrieved their changes

On the System Administration Tools (GH9011) menu

1. Choose Data Replication (P98DREP).
2. On the Work With Publishers form choose Pending Deliveries from the Form menu.
3. On the Machine Search & Select form, highlight the publisher host, and click Select.

   The Pending Deliveries form appears. This form reads the Pending Change Notification Log (F98DRPCN).



4. To view outstanding changes for a given publisher, highlight a row, and click Select. See *Viewing Outstanding changes for a Given Publisher* in this guide for information.

5. You can delete any messages that you know the replication process will never send. On the Pending Deliveries form, highlight the message, and click Delete.

Replicating Data Dictionary Changes

The central master copy of the Data Dictionary resides on an enterprise database server in OneWorld relational database (RDB) tables (F92*). Regardless of whether you activate data replication through the Data Replication application (P98DREP), the OneWorld replication engine automatically builds a copy of data dictionary items in the OneWorld Table Access Management (TAM) format on every OneWorld workstation in the enterprise. This setup enhances performance when a table or application accesses the data dictionary.

Initially, OneWorld creates empty data dictionary TAM files. As a user runs applications on the workstation, the replication engine detects the requests for data dictionary items and loads the data items into the workstation TAM files using Just-in-time-replication (JITR). The workstation downloads each data dictionary item once and maintains the attributes for the item in the local TAM specifications regardless of whether the RDB tables containing the item change on the server. This form of data dictionary replication is the simplest and is always active.

When you transfer data dictionary files to a server, you must use the Recreate Replicated Data Dictionary (R92TAM) batch application and the TAMFTP.exe program.

To synchronize the RDB tables and the TAM files on the workstation, you must set up and enable data dictionary replication through P98DREP.

This section contains the following topics:

- Understanding data dictionary replication
- Setting up data dictionary replication
- Copying data dictionary files to a server using TAMFTP.exe

Understanding Data Dictionary Replication

By default, OneWorld only loads data dictionary items into the local TAM files the first time a OneWorld application calls the data item. When the master copy of the data dictionary changes on the server, the workstation data dictionary in the TAM files does not automatically receive those changes unless the change is an addition of a new data dictionary item. You must manually set data replication for the data dictionary TAM files to automatically receives changes and deletions.

The following list provides the various ways in which you can set up data dictionary replication:

- Default just-in-time replication (JITR) for add only
- Just-in-time replication (JITR) for change and delete
- Data Dictionary Replication Flow for a Workgroup Server Configuration
- Replicating data dictionary tables on a workgroup server
- Replicating data dictionary tables on logic servers

Default Just-In-Time-Replication for Add Only

This style of replication provides only updates to the workstation TAM files for new data dictionary items that do not reside already on the workstation.

For example, when a user adds a new data dictionary item, this change occurs on the server with the data dictionary tables (F92*). The workstation receives the changes during a OneWorld session when a user accesses a field that uses the new data dictionary item.

This style of data dictionary replication does not support changes or deletions to current data dictionary items. When the workstation accesses the data dictionary item that another user changed or deleted, the workstation TAM files have no way of knowing that the data dictionary items are different than the items that currently exist in the local TAM files on the workstation. To automatically update changes and deletes, you need to set up and enable data replication for the data dictionary manually through the Data Replication application (P98DREP) as described later in this section.

If you use the default JITR data dictionary replication, to propagate changes or deletions of data dictionary items to workstations, you need to clear the TAM files by deleting the DDDICT.*, DDTEXT.*, and GLBTBL.* files from the following directory:

- \b7*pathcode name*\spec

The next time you log on to OneWorld from that workstation, OneWorld will automatically create new TAM files using the current data dictionary.

Just-In-Time-Replication (JITR) for Change and Delete

This replication style allows you to set replication so that when a user changes or deletes an existing data dictionary item, those modifications automatically replicate to the workstation. You must manually set up and enable data replication through the Data Replication application (P98DREP) as described later in the chapter.

When you replicate the data dictionary to the workstation TAM files, you must name the object DDDICT.

Data Dictionary Replication Flow for a Workgroup Server Configuration

To increase performance, you can use a workgroup server to add an extra tier in your configuration. You can set up this workgroup server as both a subscriber and a publisher. Workstations will point to the workgroup server as the publisher from which to pull the tables for pull replication. The following process flow assumes that you will use a workgroup server as a subscriber server. If you do not use a workgroup server, the workstation pulls tables directly from the publisher server for replication purposes. See *Replicating the Data Dictionary on a Workgroup Server.*

▶ Default data dictionary

1. The publisher server for the data dictionary tables (F92*) replicates the tables to the subscriber workgroup server through push replication.
2. Through pull replication, when a workstation logs on to OneWorld, OneWorld converts the data dictionary RDB tables into TAM from either the workgroup server or the enterprise server depending on server is local to a given workstation.
3. The workstation stores the data dictionary TAM files in the local TAM database on the workstation.
4. When the workstation requires a data dictionary item, OneWorld accesses the local TAM database for the data dictionary item.
5. A workstation adds a new data dictionary item to the data dictionary on the publisher server or the subscriber workgroup server based on OCM mappings.

   Changes then propagate from the publisher data dictionary level, F92* tables, to all other subscriber data dictionaries, (F92* tables, and then to all workstations.

Replicating Data Dictionary Tables on a Workgroup Server

Important: When you use an AS/400 as the publisher server, OneWorld *must* reside on the workgroup server because JDBNet needs to run on the workgroup server for replication to occur successfully.

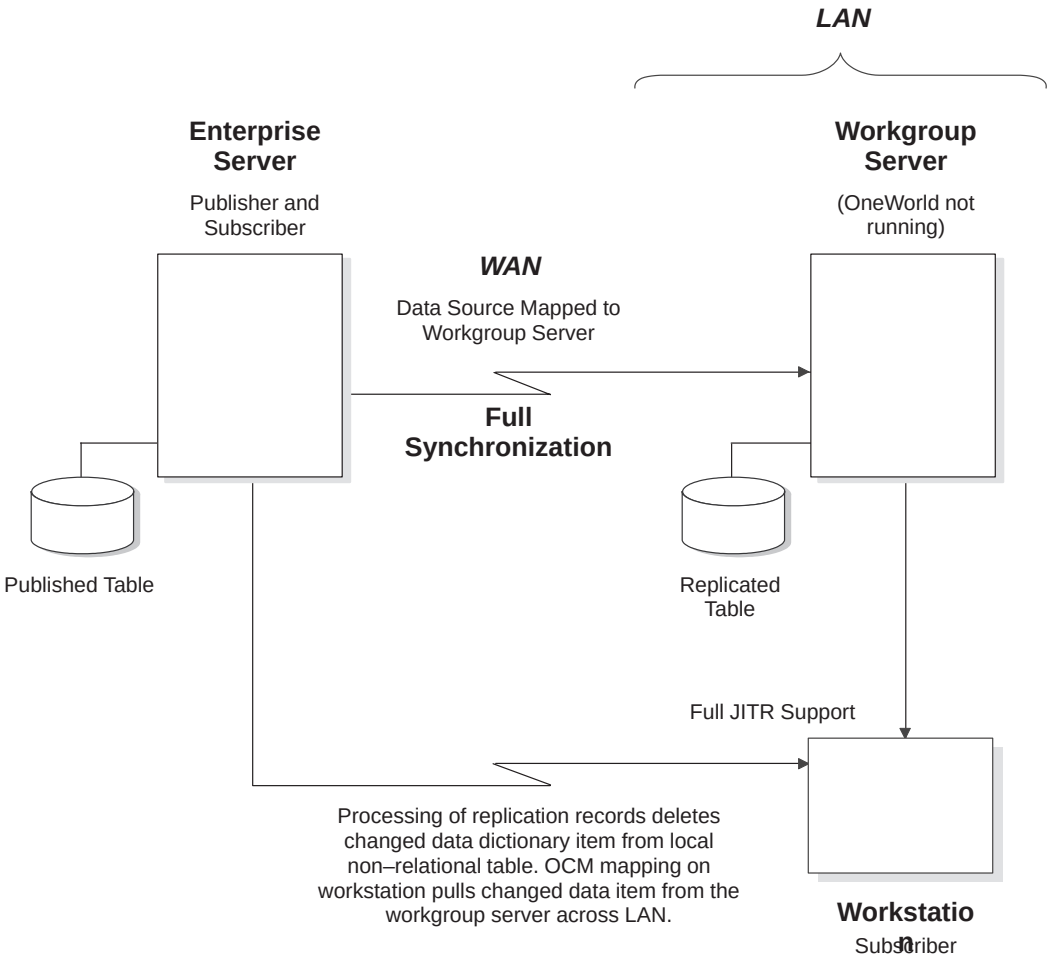
If you support a large number of workstations in a remote office, you can replicate data dictionary tables to workgroup servers in a Wide Area Network (WAN) environment to reduce network traffic. This type of replication reduces network traffic by allowing remote workstations to retrieve data dictionary items from the workgroup server rather than from the home office publisher. When the remote workstations eventually build the required items into resident TAM files, OneWorld requires still less network traffic to retrieve data.

You can also set up a workgroup server as a database-only workgroup server to support your data replication strategy. A database-only workgroup server does not run any OneWorld server code. Use the following setup:

- Set up the enterprise server as both publisher and subscriber. The data source for the subscriber on the enterprise server resides on the database-only workgroup server. You must set up the enterprise server as a subscriber to perform the logic for the workgroup server because the workgroup server is database-only. This setup synchronizes the workgroup server with the RDB data dictionary tables.
- Set up the client as a subscriber to the published tables on the enterprise server.
- Map the client Object Configuration Manager (OCM) to the workgroup server. This setup ensures that JITR replicates to the client from the workgroup server across the LAN rather than across the WAN.

The following illustration provides an example of this configuration:

Replicating the Data Dictionary to Workgroup Servers



Replicating Data Dictionary TAM Specifications on Logic Servers

Logic servers do not use the just-in-time data dictionary concept. Instead, OneWorld supports a pre-built data dictionary for these machines in TAM format. To provide TAM data dictionary specifications for logic server processing, you must run the R92TAM UBE, then you need to transfer TAM specifications to the server using TAMFTP.exe. You should use this process when changes or additions to the data dictionary occur and you need to update the data dictionary TAM specifications on the server.

See Also

Managing Data Dictionary Changes for more information about the data dictionary.

Setting Up Data Dictionary Replication

When you set up data dictionary replication, you need to set up pointers that the publisher can follow to determine the data dictionary specification tables on a workstation. These pointers provide the path code that the publisher uses to locate these specification tables.

Although you enter information for the pointer in a field called Data Source, the pointer is not an actual data source. The pointer only directs the publisher to the appropriate path code to ensure that data dictionary replication completes successfully.

▶ To set up data dictionary replication

On the System Administration Tools (GH9011) menu

1. Choose Data Replication (P98DREP).
2. On the Work with Publishers form, choose Environment Map from the Form menu.
3. On the Work with Data Dictionary Environments form, click Add.
4. On the Data Dictionary Environment Revisions form, complete the following fields:

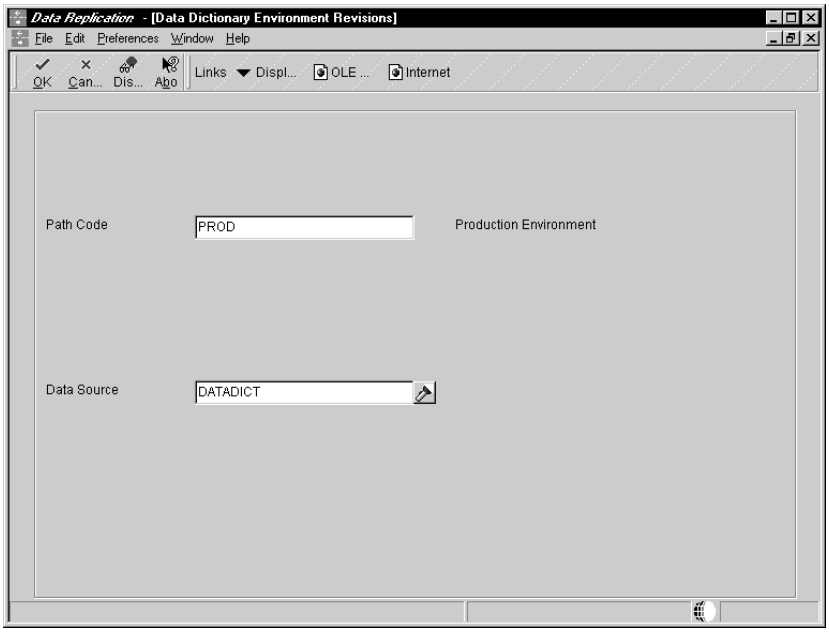
   - Path Code

     Type the path code for the workstation, such as, PROD or DEV.

   - Data Source

     Type DATADICT in this field.

     Note: This not a real data source. The replication process uses these “data source” entries to determine the path code for the data dictionary TAM specification files on the workstation.



5. Create one publisher record using the following Object Name:
   - DDDICT

   The publisher data source is the data source where your data dictionary tables reside. See *Adding a Publisher* for more information.

6. Add one subscriber for each machine to which you want to replicate the data dictionary.

   The data source for the subscriber must be DATADICT to match the entry you made on the Data Dictionary Environment Revisions form. See *Adding a Subscriber* for more information.

Copying Data Dictionary Files to a Server Using TAMFTP.exe

When you need to replicate changes to the data dictionary on a server, you must run the Recreate Replicated Data Dictionary (R92TAM) batch application on a workstation and then transfer the TAM specifications using TAMFTP.exe.

Complete the following tasks in order:

- (1) Recreate replicated data dictionary files
- (2) Transfer data dictionary files to your enterprise server

To recreate replicated data dictionary files

1. From the Data Dictionary Design menu (GH951), choose Recreate Replicated Data Dictionary (R92TAM).
2. On Work with Batch Versions, choose version XJDE0001.
3. If you have an alternative language, such as Japanese, perform the following step. Otherwise, skip to the next step.

   On Version Prompting, from the Row menu, choose Processing Options and enter the following information, then click OK:

   - Language Preference.

     Use the flashlight button to select the correct value for your language.

   - All Languages

     Type a 1 to build all languages.

4. Click Select.
5. On Version Prompting, from the Form menu, choose Advanced.
6. On Advanced Version Prompting, turn on the Override Location option.
7. Click OK.
8. On Version Prompting, click Submit.
9. On Work with Batch Versions, choose the location where you want your batch job to process, then click Select.
10. After the job finishes, log off OneWorld.

To transfer data dictionary files to a server

1. From a client workstation, locate the tamftp.exe program. It is typically in the /b7/system/bin32 directory.
2. Run the program by double-clicking it.
3. Enter JDE as the user and password. Choose the environment you are currently installing.

   A DOS window appears to display the program.

   Note: If you need to stop the program, click the X button in the upper right corner of the DOS window.

4. The program prompts you with the following questions. The appropriate response or choice of responses is noted beneath each.

   - Do you want to copy all of the supported spec types (y/n)?

     Enter N.

   - The program prompts you with each type of specification file.

     Enter Y for dddict and ddtext. Enter N for all other specification types.

   - Do you want to create a new database for each spec type to copy (y/n)?

     Enter Y.

   - Are the spec files to copy located in D:\b7\prod\spec (y/n)?

     Enter N.

     As prompted, enter the actual directory where the specification files exist, for example, d:\b7. This is the \b7 directory on the client workstation.

   - Do you want to pack the files to go in c:\temp (y/n)?

     If this is where you want to store packed files (.pak), enter Y. If you want to specify a different directory, enter N.

     If you entered N for the previous question, the program prompts you to enter an alternative directory. Verify that the directory exists on the client workstation.

- Do you want the pack files to be deleted at the end (y/n)?

  If you want to delete the .pak files from the client workstation in the \temp directory specified in the previous step, enter Y. If you do not want to delete the .pak files from your client workstation, enter N.

- Do you want the TAM FTP log file to go in c:\ (y/n)?

  If you want the program to write the log file to the root directory on the c: drive of your workstation, enter Y.

  If you want to specify a drive or directory for the log file, enter N.

  The name of the log file is tamftp.log.

  If you specify no, the program prompts you to enter a valid drive and path.

- What machine do you want to TAM FTP to?

  Specify the name of the enterprise server that is the target for the specification file transfer. You can specify either a valid TCP/IP name or an IP address.

- In what directory do you want to put the specs to copy?

  Specify a valid directory on the enterprise server that is the target for the specification file transfer. You should create this directory before running tamftp and verify that the directory is different from the specification directory where OneWorld resides. The program moves the unpacked specification files would after the file transfer finishes.

- Do you want to unpack log entries made (y/n)?

  Enter N. The program logs all necessary information into the tamftp.log on the client workstation. You do not need to duplicate this information on the enterpriser server.

When you press Enter after you answer the last question above, the tamftp program starts and displays the following message:

OneWorld TAM FTP in progress...

When the tamftp program finishes, the following message appears:

The spec copy is done.

Press the Enter key to quit.

5. Press Enter.

6. Review the tamftp log in the location you specified for results, for example c:\.
7. Move the dddict and ddtext specification files from the target directory on the enterprise server to each path code spec directory, for example, DEV733\spec.

Data Dictionary Administration

Just as a dictionary contains word definitions, the J.D. Edwards data dictionary is a central repository that contains data item definitions and attributes. These attributes determine how a data item:

- Appears on reports and forms
- Validates data entry within an application
- Assigns column and row descriptions
- Provides text for field-sensitive help

The Oneworld data dictionary is *active*, because changes that you make are automatically reflected in applications without recompiling the software.

You should assign one or two people to be your data dictionary administrator for each application area in your OneWorld enterprise. The data dictionary administrator should be experienced with OneWorld and have a fairly comprehensive knowledge of their product area, such as financial or manufacturing. The data dictionary administrator makes all additions, changes, and deletions to data items for the product group. Such changes are reflected in the pristine data dictionary on your enterprise server.

Data dictionary administrators should consider the following:

- If your setup is similar to the suggested setup in the *Typical Customer Configuration* section of the *CNC Implementation Guide*, then all environments share the same data dictionary. Therefore, the administrator can sign onto any environment to make changes. It is highly recommended that you use the Security Workbench to assign application security on the Data Dictionary application (P98DREP) to prevent unauthorized users from making data dictionary changes. See *Working with Security Workbench* in this guide.
- If you are running a coexistence enterprise, you must create all of your data items in both WorldSoftware and OneWorld, because the two products cannot share the same data dictionary.

Replicating the data dictionary is also a system administrator task. You will need to know where OneWorld stores the publisher data dictionary and how to manage data dictionary changes using data replication. If you are a co-existence customer, you need to know how to synchronize WorldSoftware and OneWorld data dictionaries. This section also describes how an administrator uses reports to update display decimals in the data dictionary.

This section contains the following:

- ❑ Understanding data dictionary replication
- ❑ Synchronizing WorldSoftware and OneWorld data dictionary changes
- ❑ Updating display decimals

Before You Begin

Before you begin working with the data dictionary, you should be familiar with the concepts in:

- *Data Dictionary* in the *Tools Guide*. This section describes using the data dictionary and defining a data item.

Understanding Data Dictionary Replication

The data dictionary resides in relational files on a server. This is the publisher data dictionary, where you would make all data dictionary changes that you want replicated to servers and workstations. OneWorld stores the publisher data dictionary in the following tables:

- Data Item Master (F9200)
- Data Field Display Text (F9202)
- Data Item Alpha Descriptions (F9203)
- Data Dictionary - Error Message Information (F9207)
- Data Field Specifications (F9210)
- Data Dictionary - Smart Fields (F9211)
- Media Objects Storage (F00165)

OneWorld supports a just-in-time data dictionary. When a user accesses an application for the first time, OneWorld installs (just in time) all data dictionary information required to run that application. This means that when installing a package (full or partial) to a machine, data dictionary information is not included.

OneWorld replicates data dictionary information from the relational data dictionary to the workstation's specification tables. Anytime OneWorld cannot find data dictionary information in the specification tables, it retrieves the information (just in time) from the relational data dictionary. This process occurs regardless of any data dictionary replication you have set up.

You should set up data dictionary replication to replicate changed data dictionary items to workstations. The following explains the process for data dictionary replication to workstations:

- A person with security to the data dictionary application changes the publisher data dictionary (the relational tables on the enterprise server). If your environment configuration follows the "Typical Customer Configuration" then all environments map the data dictionary to the publisher tables; therefore, the administrator can make the change from any environment. You must set up data dictionary replication for workstations as pull replication.

- The next time a user signs onto OneWorld, it deletes from their workstation's specification tables those data items that were changed. The next time the user accesses an application that requires that data dictionary information, OneWorld retrieves it just in time from the publisher data dictionary.

  Changes to row and column descriptions are not automatically replicated to workstations. To replicate a row or column data dictionary change, complete the following:

  - Build and deploy a full or partial package, or
  - If you know the application(s) that you want the row or column change to be reflected in, you can build and deploy an update package with those applications as either just-in-time installation item types, or object item types.

However, glossary changes are dynamic and do not need to be replicated because workstations read the Media Object Detail (F00165) table each time a glossary is accessed. You access glossaries via field-level help (pressing the F1 key or using the Help "What's This" menu item). Enterprise servers do not use the just-in-time data dictionary concept. Therefore, changes must be replicated to servers through a batch process. See "Replicating Data Dictionary Changes to Servers" in this guide for information.

This topic contains the following:

❑ Replicating data dictionary changes to workstations

❑ Replicating data dictionary changes to servers

Replicating Data Dictionary Changes to Workstations

This topic explains how to replicate data dictionary changes to workstations, and it contains the following:

❑ Concepts of replicating data dictionary changes to workstations

❑ Setting up data dictionary workstation replication

Before You Begin

- Data dictionary administrators should be familiar with the principles of data replication as explained in *Data Replication*.

Concepts of Replicating Data Dictionary Changes to Workstations

OneWorld stores data dictionary information on each workstation in a permanent cache under the local \path code\spec directory as glbtbl.xdb and glbtbl.ddb. If you change data items are changed and you want to immediately replicate the changes to workstations, you must use the Data Replication (P98DREP) application. You can use this application to notify subscriber workstations of changes.

The data dictionary replication software detects when an item is changed and maintains tables containing replication messages. When an item is changed, the next time a subscriber machine signs onto OneWorld, it pulls a replication message that tells it which item has changed. As a result, OneWorld on the workstation modifies the permanent cache for the data dictionary by deleting that data item. Then the next time that workstation users runs an application that requires the deleted data item, OneWorld detects that the information is not in the permanent cache and pulls the information from the publisher data dictionary (the relational database tables).

Important: Avoid having two separate data dictionaries. Typically, you should share one data dictionary between your development and production environments. That is, there should be a single data dictionary in a single path code shared by each environment. However, if you make changes to data items that applications use in the live production environment, and these changes will cause those applications to fail, then you need to have a separate data dictionary for your development environment.

Under most circumstances, data dictionary changes made after going “live” with OneWorld are to add new items and modify items that are for applications still under development. If you need two data dictionaries, then you are using replication in a way that J.D. Edwards has not tested. If you do have a separate data dictionary for development and production environments, only set up data dictionary replication for the production data dictionary.

Setting Up Data Dictionary Workstation Replication

This task explains how to replicate data dictionary changes to workstations. Set up your replication as follows:

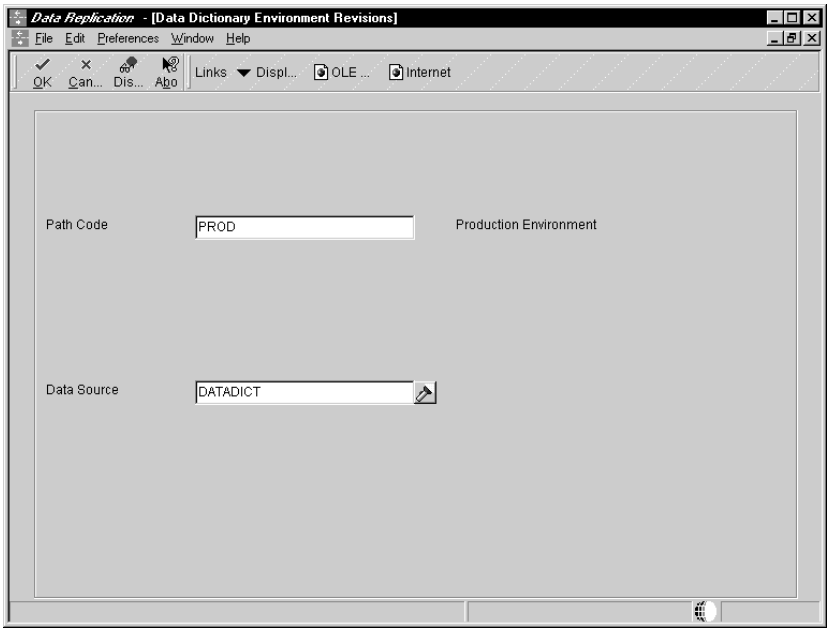
- Add only one publisher record for workstation data dictionary replication, using the object name DDDICT. This object name represents all of the data dictionary files.
- Set up one subscriber record for each machine to which you want to replicate.

The subscriber data source must be the same data source you enter on the Work With Data Dictionary Environments form, as explained in this task.

▶ To set up data dictionary workstation replication

On the System Administration Tools menu (GH9011) menu

1. Choose Data Replication (P98DREP).
2. On the Work With Publishers form, choose Environment Map from the Form menu.
3. On the Work With Data Dictionary Environments, click Add.



4. On the Data Dictionary Environment Revisions form, enter one record for each path code. The data source on this form is not a real data source. It is used as a link to the subscribers to determine which path codes should receive the data dictionary changes. Enter the data sources as follows:

   production data source = DATADICT

   development data source = DATADICT

   Use DATADICT as a literal for both values, unless you have a different data dictionary for each path code (which is *not* recommended). If you do, use a different value for each, such as DATADICTPROD and DATADICTDEV.

5. Create one publisher record using the Object Name: DDDICT. The publisher data source is the data source where your data dictionary tables reside. See *Adding a Publisher* in the *Data Replication* section for more information.
6. Add one subscriber for each machine to which you want to replicate. The data source must be the same as you entered on the Data Dictionary Environment Revisions form (DATADICT). See *Adding a Subscriber* in the *Data Replication* section for more information.

   When adding these subscribers, make sure that the Enabled field on the Subscriber Revisions form is set to Y; otherwise, replication will not occur.

Important: If the Pending Change Log (F98DRPCN) and the Replication Change Log (F98DRLOG) reside on an AS/400, the Client Access ODBC data source must be set to "Do Not Translate." If it is not set properly, the replication messages will be corrupted.

Replicating Data Dictionary Changes to Servers

This topic describes how to replicate the publisher data dictionary stored in relational files to the replicated subscriber data dictionaries. This is server-to-server or server-to–workstation push replication. OneWorld logic servers access a replicated data dictionary stored in the following specification files:

- JDEKRNL.ddb
- JDEKRNL.xdb
- DDDICT.ddb
- DDDICT.xdb
- DDTEXT.ddb
- DDTEXT.xdb

You have the choice of replicating only the changed items or replicating all changes, including deleted items. This topic contains the following:

❑ Replicating only the changed items

❑ Refreshing the data dictionary

Before You Begin

❑ Before continuing with this topic, become familiar with the principles of data replication as explained in the *Data Replication* section.

Replicating Only the Changed Items

If there are data dictionary changes, you can use a batch process to replicate them before deploying a new package to the enterprise server. This process replicates changes from the relational data dictionary (publisher) to the replicated data dictionaries (subscribers) on other servers, but does not delete items from the replicated data dictionaries. To delete items in the replicated data dictionary, you must use the global update application. See *Replicating All Changes (Global Update)* in this section for information.

The administrator's workstation must have the most recent installation of your production path code. If you cannot install a workstation package that includes the most recent replicated data dictionary specifications, then manually copy them to your local machine. These files include:

- JDEKRNL.ddb
- JDEKRNL.xdb
- DDDICT.ddb
- DDDICT.xdb
- DDTEXT.ddb
- DDTEXT.xdb

The administrator's workstation (the machine from where you are running this batch process) must have a complete data dictionary residing on it. Because workstations use a distributed (just-in-time) data dictionary, their copy of the data dictionary will not contain a complete set of all data dictionary items. Therefore, J.D. Edwards provides a complete data dictionary with the installation process so that you can copy it to the machine that runs this batch process for servers.

Before You Begin

❑ Copy the complete data dictionary from your deployment server's base installation directory (*Deployment Server Name**OneWorld Release*\Data Dictionary) to the workstation's b7*path code*\spec directory. You should only need to do this once, assuming you continue to use the same workstation for the server replication process. Each time you refresh the data dictionary tables you might want to save them to a file server in case you need to reload the workstation.

▶ To replicate only the changed items

On the Data Dictionary Design menu (GH951)

1. Choose Replicate Data Dictionary Changes (R92001T).
2. On the Work With Batch Versions form, select the version.
3. On the Version Prompting form, choose Data Selection, and click Submit.
4. On the Data Selection form, choose Date - Updated and make it greater than the date of the last package build, and click OK.
5. After the batch process has completed, copy the refreshed data dictionary specifications from your local b7/*path code*/spec directory to the enterprise server that stores the master data dictionary in a relational database. The file transfer process is site-dependent and varies according to enterprise server platform. The transfer process should take into account the differences in byte alignment between the workstation and server platforms.

Refreshing the Data Dictionary

The global update batch process completely refreshes the data dictionary from the relational format (publisher data dictionary) to the replicated specification tables. This program starts with empty replicated tables; therefore, any deletions made to the relational tables are reflected in the replicated specification tables.

This application creates the new specification tables under the administrator's local b7 directory; therefore, make sure you have an extra 50 MB on the drive where OneWorld resides, and 100 MB on your C drive for swapping. This process takes four hours. After it completes the refresh, you see the following:

```
b7
   dddict.ddb [new tables]
   dddict.xdb
   ddtext.ddb
   ddtext.xdb
   path code name
     spec
       dddict.ddb [old tables]
       dddict.xdb
       ddtext.ddb
       ddtext.xdb
```

▶ To replicate all changes (Global Update)

1. From a workstation, on the Data Dictionary Design menu (GH951), choose Recreate Replicated Data Dictionary (R92TAM), and run version XJDE0001.
2. Copy the new data dictionary specification files to the server. The file transfer process is site-dependent and varies according to enterprise server platform. The transfer process should take into account the differences in byte alignment between the workstation and server platforms.

Synchronizing WorldSoftware and OneWorld Data Dictionaries

This topic only applies to customers with coexisting WorldSoftware and OneWorld installations.

You must separately maintain the OneWorld data dictionary and the WorldSoftware data dictionary. This means that any changes you make to the OneWorld data dictionary you also need to make in the WorldSoftware data dictionary, and vice versa. To help you maintain parity between the two data dictionaries, J.D. Edwards provides a batch process that compares the WorldSoftware data dictionary on the AS/400 with the OneWorld data dictionary.

- If an item is in the WorldSoftware data dictionary but not in OneWorld, this process adds it to OneWorld.
- If an item is in OneWorld but not in WorldSoftware, the process prints this information on an exception report. You must either add the item to the WorldSoftware data dictionary or delete the item from the OneWorld data dictionary.
- The process replaces the glossary information, row headings, and column headings in OneWorld with that information from WorldSoftware.
- Item specifications that both WorldSoftware and OneWorld use, such as data item size, system code, or decimal places, are overwritten in OneWorld with information from WorldSoftware.

The two data dictionaries should not get out of synch if you are adding and changing items to both data dictionaries. However, for integrity reasons you should run this report, first in proof mode and, if necessary, in update mode to keep the data dictionaries alike.

▶ To synchronize coexistence data dictionaries

1. Sign onto the AS/400 WorldSoftware, and select the environment where the WorldSoftware data dictionary resides.
2. Select DREAM Writer, Versions List.
3. Enter Form P99800, and select Version XJDE0001.
4. In Processing Options, enter the library that contains the WorldSoftware data dictionary and the library that contains the OneWorld data dictionary. To start the update, enter 1 for update mode. To review the update before the process actually begins, enter 0 for proof mode.

The update process takes about 30 minutes. It produces reports with the following information:

- Data items added to OneWorld
- Items in OneWorld that are not in WorldSoftware
- Differences found

Updating Display Decimals

You can change the position of the display decimal for specific field classes, such as currency or quantity fields. For example, data items that belong to the currency class come with the display decimal set at two, but you can change the display decimal to any number, such as four. So instead of seeing 100.00, you would see 100.0000

You should only change the display decimal value in a CRP environments before any live production data is entered. This is because OneWorld does not have a data conversion feature, so if users change display decimals *after* users have entered data, the data entered before changing the display decimals will be wrong.

As with any data dictionary change, when you update display decimals, the replication method you chose (batch or workstation) will replicate the changes throughout your enterprise.

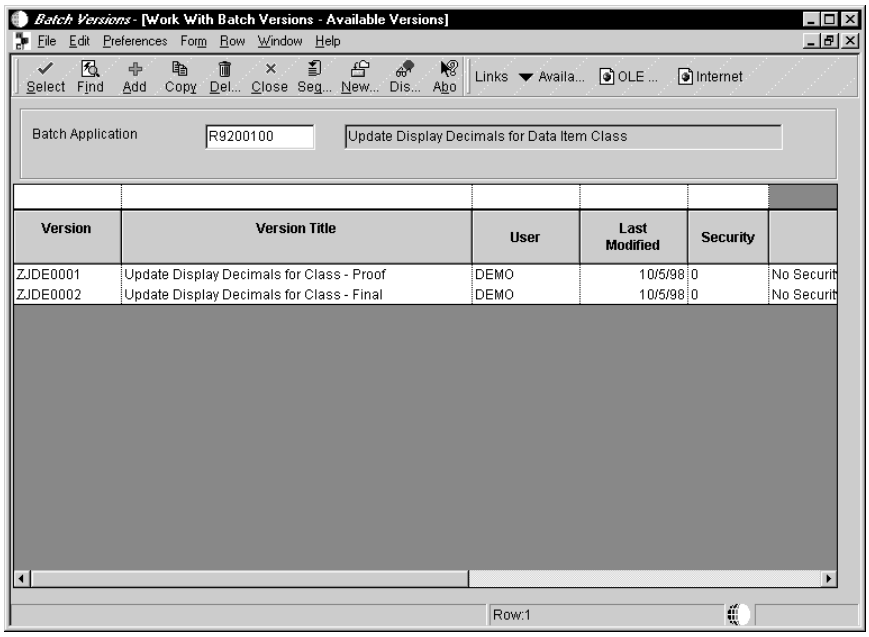
Before You Begin

❑ Turn OFF data replication by disabling the publishers or by turning off replication for the DDDICT and F92* tables; otherwise, OneWorld could create hundreds of replication messages. See *Enabling and Disabling Publishers and SubscRibers* in the *Data Replication* section.

▶ To update display decimals

On the System Administration menu (GH9011)

1. Choose Batch Versions (P98305).



2. On the Work With Batch Versions form, type R9200100 in the Batch Application field, then click Find.
3. Choose one of the following versions and click Select:
   - Update Display Decimals for Class – Proof

     The proof version produces only a report of what the process would do.

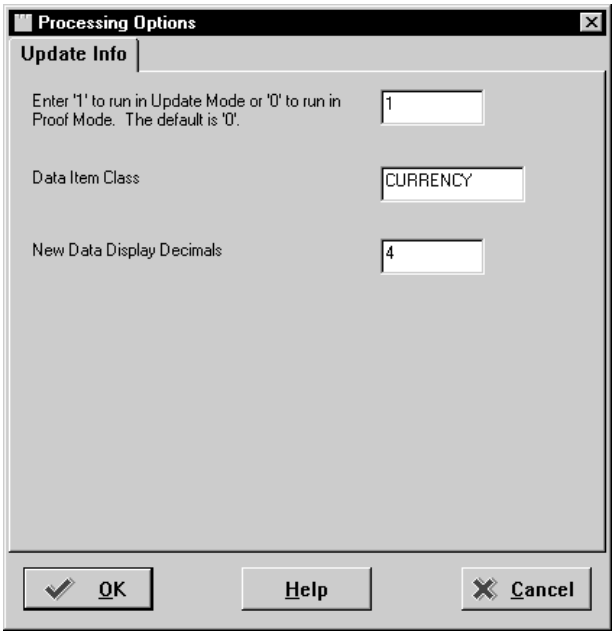
   - Update Display Decimals for Class – Final

     The final version actually makes the changes.
4. On the Version Prompting form, click Data Selection, and then click Submit.
5. On the Data Selection form, create the following statement:

   If BC Data Item Class (F9210) = “CURRENCY”

   CURRENCY is a literal value that you can replace with any other data item class that uses display decimals, such as QTYINV.
6. Click OK. If you changed the Data Selection statement, first click Update, then click OK.

The Processing Options form appears (the fields are blank when the form appears). The following is an example of what you can enter into the fields.



7. Complete the following fields:
   - Enter ‘1’ to run in Update Mode or ‘0’ to run in Proof Mode
   - Data Item Class
   - New Data Display Decimals
8. Click OK.
9. Each user should delete the glbtbl.xdb and glbtbl.ddb specification files, found in the b7\path code\spec directory on each workstation.

Field	Explanation
Enter ‘1’ to run in Update Mode or ‘0’ to run in Proof Mode	The values are: • 1 runs the batch process in Update Mode, which updates OneWorld with the new display decimal • 0 runs the batch process in Proof Mode, which allows you to preview the change before updating OneWorld
Data Item Class	Enter the data item class you want to change, such as CURRENCY or QTYINV.
New Data Display Decimals	Enter a number that represents how many decimals you want OneWorld to display.

Printing OneWorld Reports

The Printers menu (GH9013) provides a single point of entry for configuring your printers within OneWorld, regardless of platform considerations. The applications available on this menu allow you to define printers for workstations and enterprise servers. These definitions reside in OneWorld tables that are maintained by the Printer Definitions application (P98616).

In addition to creating your own reports, OneWorld includes a number of predefined reports and report versions, which you can use and modify for your business needs. OneWorld creates reports using the batch engine and generates these reports in Portable Document format (PDF). You can view the PDF files using the Adobe Acrobat Reader software.

Reports process as batch applications without user interaction. When a user submits a report for processing, the user makes choices, such as the selection and sequencing of data to include in the report, the location where the report will process, logging capabilities to monitor how the report processes, and the printer on which the report prints.

See the *OneWorld Foundation Guide* for more information about submitting and printing a report.

This section describes the following:

- Understanding OneWorld printing
- Working with the Printers application
- Generating and retrieving logs for your report

Understanding OneWorld Printing

When you submit a OneWorld Report, the batch engine generates a Portable Document Format (PDF) file. OneWorld then transforms this file into one of three Page Description Language (PDL) formats, PCL, PostScript, or line printer text, depending on the type of printer that prints the report.

The batch engine uses a device context to create the PDF file. This device context consists of information such as page size and the printable area of a page. On a Windows platform, the device context is obtained through Windows APIs that return this information about either a named printer or the default printer. On servers, OneWorld generates this information from the printer tables.

OneWorld uses a specific logical path to determine to which printer to send a report. The batch engine locates the logical printer name using the following order. If the first method does not return a printer name, the batch engine uses the next subsequent method.

- (1) Processing event rules on the Report Design Aid (RDA) Do Initialize Printer event
- (2) Using the name passed from RDA specifications to the batch job at the time when the job initially launches
- (3) Determining the default printer for the current user, environment, and host from the Printer Definition table (F98616)

Based on the information that the batch engine retrieves, the engine determines how much information will fit on a page and begins to process the report. This means that the engine creates a PDF file with a specific paper size, font, and so on. However, if you run a report using one device context, such as PostScript or PCL, and then send the output to print on a device with a different context, such as a line printer, unexpected results might occur.

After the PDF file is created, you can view it on your workstation or print it. You view a report using Adobe Acrobat Reader. You can print the report from the Reader. If you run a report on an Intel-based workstation that runs Windows NT, the workstation uses Acrobat Reader to send the PDF to the printer. For other platforms, or if Adobe Reader is not available on the workstation, then the output is transformed through a OneWorld filter. For servers, the output is either PostScript, PCL, or text.

The following list provides some additional printing considerations:

- Running Reports on the Server
- Running Reports on the Workstation

- Print-time characteristics
- Print Settings for the Workstation jde.ini

Running Reports on the Server

When you submit a report to the server, the engine prompts you for a printer name previously defined using the Printer Definitions application. Then the server automatically creates a PDF file using the settings associated with the selected printer, unless event rules (ER) override those printer settings. You can, however, affect how your report prints on the server before you generate a PDF file by changing settings in the printer applications. When you view the report on the server, OneWorld copies the PDF file from the server to the local \b7\PrintQueue directory on your workstation.

When you run a report, you also have the option of turning on logging capabilities. If you access the Advanced form when you submit your report, you can turn on logging capabilities. When you view a log, your workstation stores the file in the \b7\PrintQueue directory.

See Also

- *Generating and Retrieving Logs for Your Report* for more information about the location of the PrintQueue directory on a server.

Running Reports on the Workstation

When you choose to run a report and view the output on the screen, the engine tries to connect to the printer defined in Report Design. If the engine cannot connect or if there is no printer defined, the engine uses the default printer for your workstation. Using the settings it retrieves, the engine creates a PDF file and displays the report through Acrobat Reader. The PDF file is stored in your local \b7\PrintQueue directory.

When you run a report locally and send the output to a printer, the engine displays a dialog box that gives you the option to change the printer, paper size, and so on. The initial printer shown in this dialog box is the one defined in RDA or the default Windows printer if none was defined. The engine connects to the printer defined in the printer dialog and retrieves the associated settings. Using these settings, the engine creates a PDF file and sends the PDF file to printers in a background process through Acrobat Reader.

Print-Time Characteristics

The system assigns print-time characteristics based on settings you specify when you print the PDF file after the system generates the file at runtime. You print jobs using the Work with User Jobs application (P986116). On OneWorld enterprise servers, this application prints a previously written PDF file. The inclusion of certain printer parameters that control the final appearance of the report is determined by the hardware/software platform of the enterprise from which you print the report.

The following list provides the characteristics by platform that you can specify at print-time. You can also specify these characteristics when you launch the batch job.

UNIX (HP9000 and RS/6000)

- Page orientation
- Logical printer

AS/400

- Page orientation
- Logical printer

NT (Intel)

- Page orientation

NT (Alpha)

- Page orientation

Print Settings for the Workstation jde.ini

The workstation jde.ini settings control whether a report prints immediately and whether OneWorld saves the output after processing the report.

```
[NETWORK QUEUE SETTINGS]
PrintImmediate=TRUE/FALSE
SaveOutput=TRUE/FALSE
```

Setting	Description
PrintImmediate	Specifies whether the system automatically prints the report after processing is complete. Valid values are: TRUE. The system processes the report on the server, generates a PDF file, converts the PDF to the appropriate PDL for the defined printer, then prints the report. FALSE. The system processes the report on the server, but does not automatically print the report. Users must use the Work with Servers application to manually print the report.
SaveOutput	Specifies whether the system saves or deletes the output after you view or print the job. Valid values are: TRUE. The system saves the output after you have viewed or printed the job. FALSE. The system deletes the output after you have viewed or printed the job.

Working with the Printers Application

The Printers menu (GH9013) provides a single point of entry to use when you configure your printers in OneWorld, regardless of platform considerations. The applications available on this menu allow you to define printers for workstations and enterprise servers. These definitions are stored in OneWorld tables that the Printer Definitions application (P98616) maintains.

The Printer Definitions application comprises the following applications:

- Work with Paper Types

  Use this application to define the length and width of the paper on which you print reports. The Paper Definition table (F986162) stores this information.

- Work with Physical Printers

  Use this application to provide names for the physical printer (machine) in OneWorld. Based on the physical printer name, you can define the specifications for the printer in OneWorld, such as what paper types the printer can use and the host (server) for the printer. The Printer Capability table (F986164) stores this information.

- Work with Conversion Programs

  Use this application to define how OneWorld converts the PDF file generated during processing to the appropriate format for your printer. The Output Conversions table (F986164) stores this information.

- Work with Logical Printers

  Use this application to associate the attributes for paper types, physical printers, and conversion programs with a logical printer name . A logical printer is the name you give a printer in OneWorld. This OneWorld name connects to a physical printer name that represents the actual device on the network. For line printers, you can also set page specifications, such as Lines Per Inch (LPI), Characters Per Inch (CPI), Columns Per Page (CPP), and Lines Per Page (LPP). The Printer Security table (F98616) stores this information.

- Work with Default Logical Printers

  Use this application to define a default printer for a valid user name, environment, and host combination. The Default Printer Table (F986161) stores this information.

After you define your printers, the system locates the logical printer name using the following order each time you start a batch job. If the first method does not return a printer name, the batch engine uses the next subsequent method.

- (1) Processing event rules on the RDA Do Initialize Printer event
- (2) Using the name passed from either job records or RDA specifications to the batch job at the time when the job initially launches
- (3) Determining the default printer for the current user, environment, and host from the Printer Definition table (F98616)

After the batch engine determines the printer, the system creates the PDF file using the characteristics of the logical printer as the basis for page layout considerations. After the system completes processing the batch report, the printer application determines which filter application to run, if any, and to what physical device to send the output.

J.D. Edwards recommends you perform the printer definition tasks in the sequence presented in this chapter. In some cases, you cannot perform a subsequent task until you complete a prior task. You must set up printers on each server platform you use in your enterprise.

This section uses the following case study to provide examples for each task:

- Case Study: Setting Up Printers

Complete the following tasks in sequence:

- ❑ (1) Defining paper types
- ❑ (2) Defining physical printer
- ❑ (3) Defining printer output conversion programs
- ❑ (4) Setting up a logical printer
- ❑ (5) Setting up a default logical printer

Case Study: Setting Up Printers

This section uses a case study of Company JDE to provide examples of how a printer set up functions in the "real world." Company JDE needs to process OneWorld reports on enterprise servers and then send the output to printers. The configuration for Company JDE comprises the following characteristics relevant to running a report:

- A laser printer (JDEprn14), which supports postscript and PCL

  The laser printer supports these three paper types: letter, legal, and A4.

- An EBCDIC line printer (JDEprn15)

- Three servers for running OneWorld reports: AS400Server, NTServer, and HPServer.

  The OneWorld Printers application recognizes theses servers as *hosts*.

- Company JDE also uses these three environments in OneWorld: AS400ENV, NTENV, and HPENV.

This case study will use USER01 as the user ID.

Defining Paper Types

You define paper types to add or modify different specifications for the types of paper you use in your printer. You cannot delete a paper type definition if the definition exists in the Printer Capability table (F986163). See *Defining Printer Capabilities.*

When you define paper types, you must associate those paper types with a physical printer. The physical printer must support the paper types that you define for the physical printer definition in OneWorld. For example, if you define the LEGAL paper type for a physical printer, that printer must have the capacity to use legal sized paper. See *Defining a Printer.*

Complete the following tasks:

- Display paper types
- Add a paper type
- Modify a paper type
- Delete a paper type

▶ To display paper types

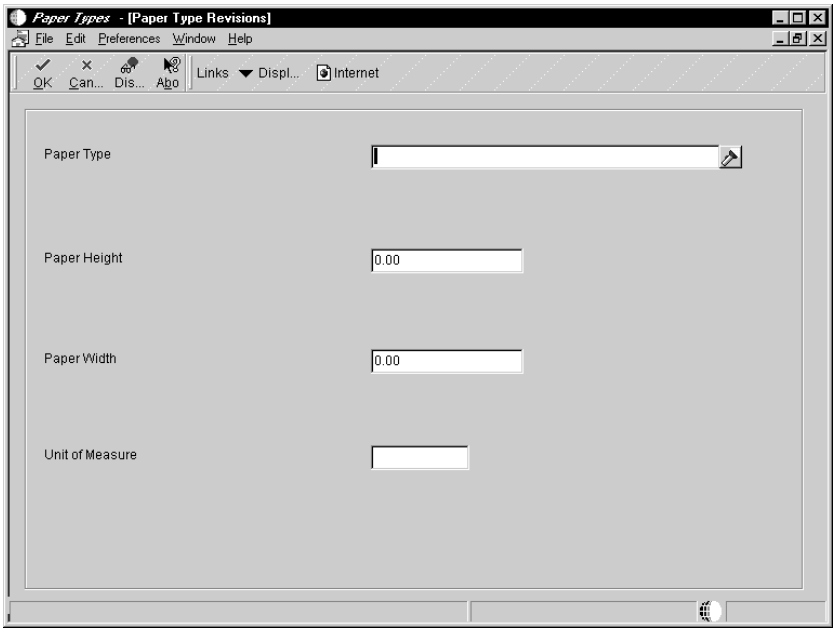
On the Printers menu (GH9013), choose Work with Paper Types (P98616).

On Work With Paper Types, click Find.

▶ To add a paper type

On the Printers menu (GH9013), choose Work with Paper Types (P98616).

1. On Work With Paper Types, click Add.



2. Complete the following required fields:

   - Paper Type

     You need to set up the UDC for the custom paper type in the User Defined Codes (P0004A) application. The system code and UDC for paper type are H98 and PT respectively. See *User Defined Codes* in the *OneWorld Foundation Guide*.

   - Paper Height
   - Paper Width

     The maximum width you can enter is 22 inches or 558.8 millimeters.

   - Unit of Measure

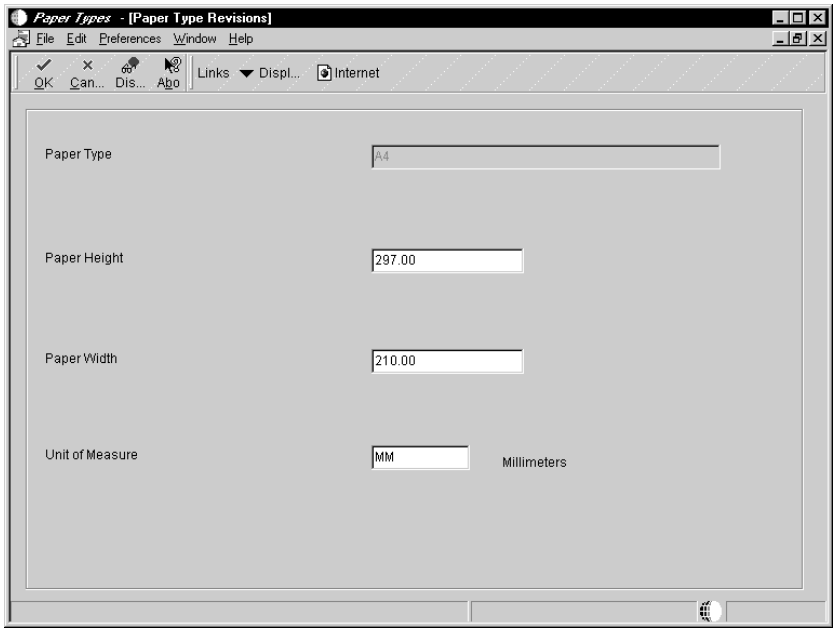
Field	Explanation
Paper Type	A user defined code (H98/PT) that indicates the type of printer paper, such as letter or legal. For example, LETTER, LEGAL, and A4.
Paper Width	A value that specifies the width of the paper for this paper type. This value is in the unit of measure specified by Unit of Measure.
Paper Height	A value that specifies the height of the paper for this paper type. This value is in the unit of measure specified by Unit of Measure.

Field	Explanation
Unit of Measure	A user defined code (00/UM) that indicates the quantity in which to express an inventory item, for example, CS (case) or BX (box). . . . . . *Form-specific information* . . . . . Indicates the unit of measure in which the paper height and width are entered. Example: IN = Inches MM = Millimeters

▶ To modify a paper type

On the Printers menu (GH9013), choose Work with Paper Types (P98616).

1. On Work With Paper Types, click Find and choose an existing paper type definition.



2. On Paper Type Revisions, revise the following required fields as necessary:
   - Paper Height
   - Paper Width

     The maximum width you can enter is 22 inches or 558.8 millimeters.

   - Unit of Measure

▶ To delete a paper type

On the Printers menu (GH9013), choose Work with Paper Types (P98616).

1. On Work with Paper Types, click Find, select the paper type, then click Delete.
2. On Confirm Delete, click OK.

   The item you deleted should no longer appear in the grid.

Defining Physical Printer

You must define OneWorld printer capabilities compatible with your physical printer. Use the Work with Physical Printers application (P98616) to display and define capabilities for your printers. When you display physical printer capabilities, OneWorld displays the names of the physical printers in the Printer Definition table (F98616). The host uses these names to connect to a printer.

OneWorld displays physical printer capabilities in a format according to the convention of Windows Explorer. That is, the system displays the printer capabilities in a directory tree structure. Each component on the tree structure is called a node. OneWorld names the nodes with a user defined code (98/TP). You can expand and collapse the hierarchical information display by clicking on the + and - nodes.

You cannot modify physical printer capabilities; you can only add or delete these capabilities. You can, however, copy a preexisting physical printer and rename the printer to save time when you need to add multiple, similar printers.

For the AS/400, the physical printer name must be the same as the outqueue name.

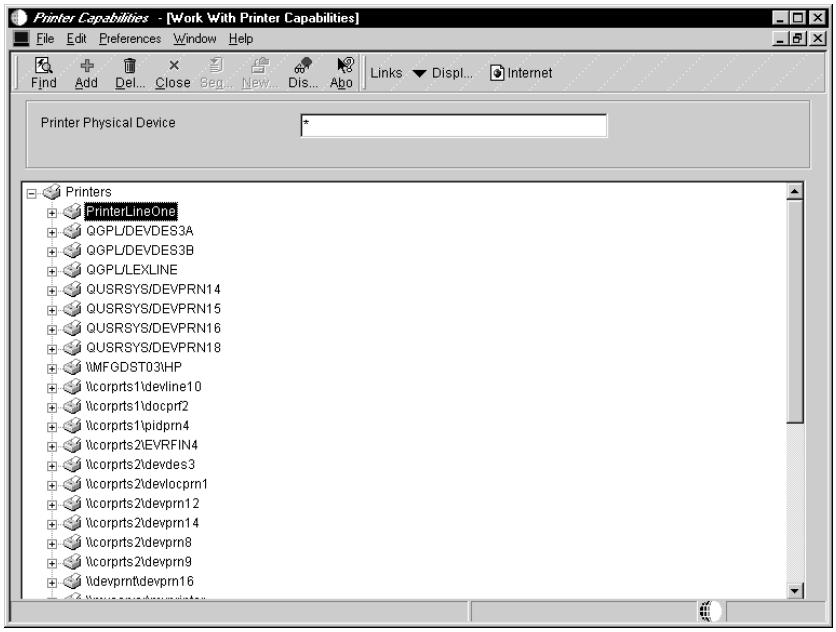
Complete the following tasks:

- Display physical printer capabilities
- Add physical printer capabilities
- Copy physical printer capabilities
- Delete physical printer capabilities

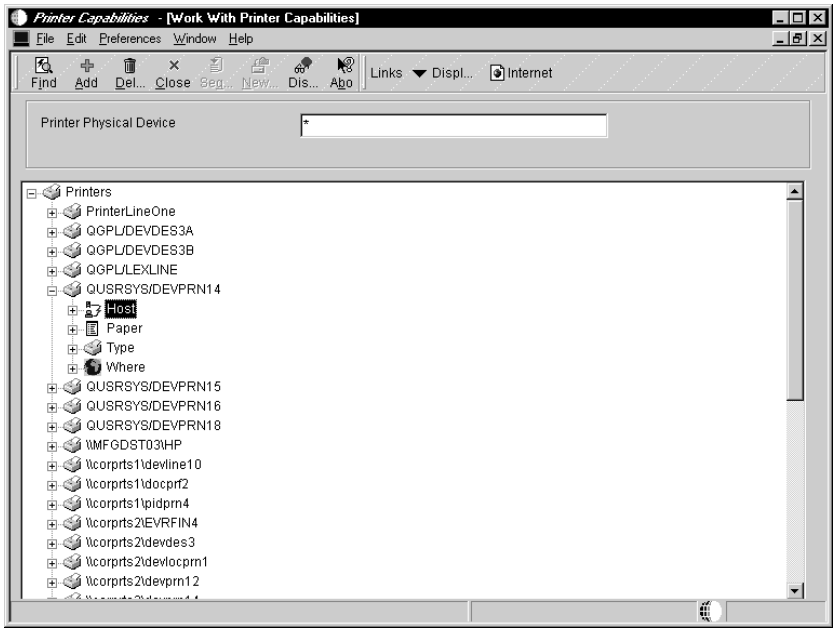
▶ To display physical printer capabilities

On the Printers menu (GH9013), choose Work with Physical Printers (P98616).

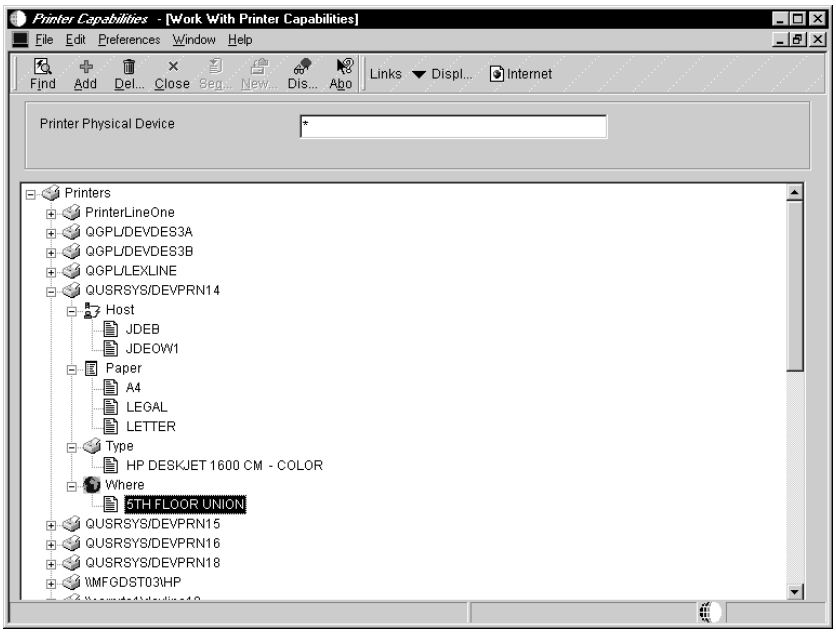
1. On Work with Printer Capabilities, click Find.
2. On the displayed directory tree structure, locate and choose a printer.



3. Double click on the printer to display a list of capability types.



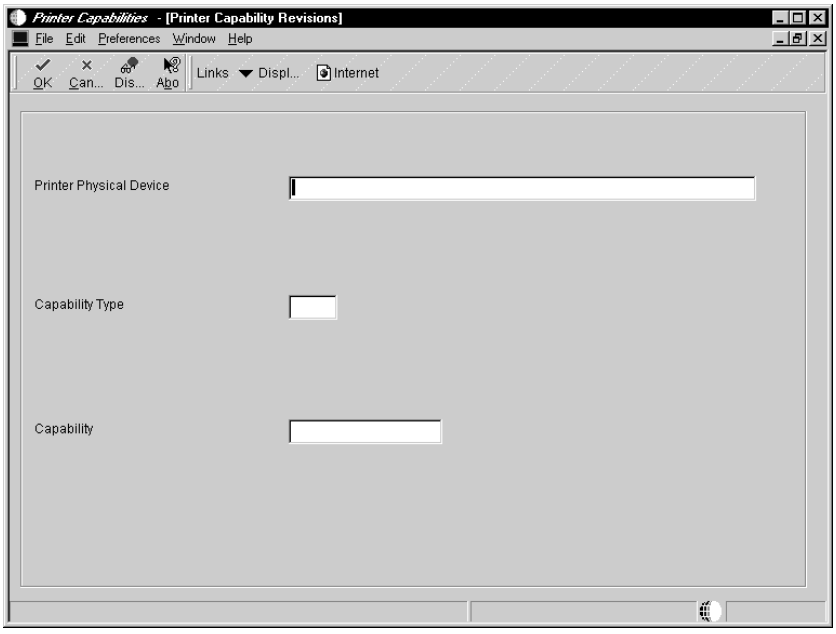
4. Double click on any capability to display the defined attributes for that capability.



▶ To add physical printer capabilities

On the Printers menu (GH9013), choose Work with Physical Printers (P98616).

1. On Work with Physical Printers, click Add.



2. Complete the following required fields:
   - Physical Printer

In this field, you must manually type the name of a physical printer that exists on the network. If you use multiple platforms, you must define a physical device for each platform using the following naming conventions:

- AS/400: *library name/outqueue name*

  For the AS/400, the physical printer name must be the same as the outqueue name. If you use the default QGPL library to store your outqueues, you need only enter the outqueue name in this field.

  Example: DEVDES3A

  If your outqueues reside in a library other than the default QGPL library, you need to enter the library name and the outqueue name in this field.

  Example: QUSERSYS/DEVDES3A

  Note: When you qualify your outqueue name with the library name, you avoid possible name conflicts that might result in the submission of your report at an unexpected outqueue.

- Windows NT: *server name**printer name*

  Example: \\corprts1\docprf2

- UNIX: *printer name* (no slashes)

  Example: devprn16

- Capability Type

  The capability types H (Host) and P (Paper) are required. You must enter information for these capabilities for OneWorld to properly print to the physical printer. The capability types T (Type) and W (Where) are for informational purposes only.

- Capability

3. Click OK.

   OneWorld refreshes the form but leaves the name you typed in the Physical Printer field. Repeat the previous steps in this task to enter additional capabilities.

4. After you finish adding capabilities for a physical printer, click Cancel to return to Work with Physical Printers.

Field	Explanation
Printer Physical Device	The name assigned to a specific printer on a network, such as Accounting_Printer_1. For OneWorld, this name cannot contain a space character, even if allowed by your operating system.
Capability Type	The user defined code (98/TP) that indicates the type of printer capability.
Capability	Special capabilities of this printer as determined by the Capability Type. For Example, the capabilities for Paper are LETTER, A4, and LEGAL; whereas, the capabilities for Host are the names of the machines where the job processes.

Example: Defining Physical Printer Capabilities

Your printer capabilities should look similar to the following examples for company JDE:

- QGPL/JDEPRN14
  - Host
    - AS400Server
  - Paper
    - A4
    - LEGAL
    - LETTER
  - Type
    - Laser Printer
  - Where
    - 5th Floor Union
- QGPL/JDEPRN15
  - Host
    - AS400Server
  - Paper
    - LETTER

❑ Type

- EBCDIC LinePrinter

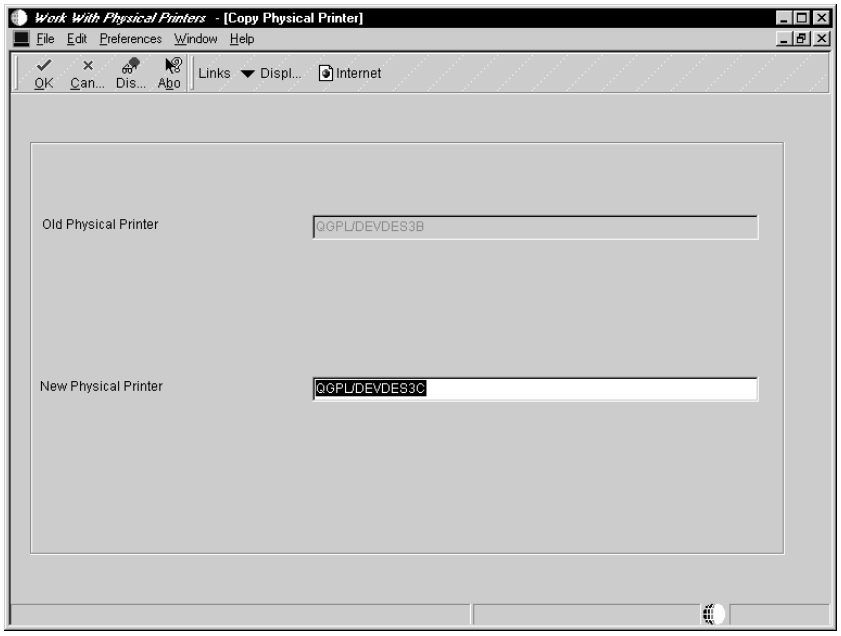
❑ Where

- 5th Floor Union

▶ To copy physical printer capabilities

On the Printers menu (GH9013), choose Work with Physical Printers (P98616).

1. On Work with Physical Printers, under Printers, select a physical printer, then click Copy.



2. On Copy Physical Printer, complete the following field, then click OK:
   - New Physical Printer

     Type a name for the new physical printer in this field.
3. On Work with Physical Printers, click Find to refresh the list of physical printers.

▶ To delete physical printer capabilities

On the Printers menu (GH9013), choose Work with Physical Printers (P98616).

1. On Work with Physical Printers, click Find, select the physical printer or the associated capability, then click Delete.
2. On Confirm Delete, click OK.

   The item you deleted should no longer appear in the tree structure.

Defining Printer Output Conversion Programs

The batch engine generates reports in the form of Adobe PDF files. OneWorld uses conversion programs to convert a PDF file to the appropriate format for a printer. The conversion programs support formats that include Postscript (*JDE PS), Line Printer (*JDE LINE), and PCL (*JDE PCL).

Use Work with Conversion Programs (P98616) to define conversion programs. You must identify a specific conversion program and supply the associated parameter string that OneWorld passes to the host environment. After you define conversion programs, you can use the definition to complete the Conversion Program field on Logical Printer Revisions. See *Setting Up a Printer Definition.*

The values you enter for the parameter string depend on the conversion program and the platform you use. In turn, the conversion program you use in your logical printer definition depends on the output printer device and your enterprise server and software platforms.

The following table lists the printer and platform types along with their associated conversion program and parameter string values. All values are case-sensitive, and you must enter them exactly as shown in the table.

You must define a conversion program for each host that processes batch jobs.

Printer Output Conversion Values

Type of Printer	Platform	Conversion Program	Parameter String
PostScript	Windows NT (Intel and Alpha), AS/400, RS6000	*JDE PS	-s POSTSCRIPT_PRINTER -l jdekrnl -f convertPDFToPS
	HP	*JDE PS	-s POSTSCRIPT_PRINTER -l ube -f convertPDFToPS
PCL	Windows NT (Intel and Alpha), AS/400, RS6000	*JDE PCL	-s PCL_PRINTER -l jdekrnl -f convertPDFToPCL
	HP	*JDE PCL	-s PCL_PRINTER -l ube -f convertPDFToPCL
Line Printer	Windows NT (Intel and Alpha), AS/400, RS6000	*JDE LINE	-s LINE_PRINTER -l jdekrnl -f convertPDFToLP
	HP	*JDE LINE	-s LINE_PRINTER -l ube -f convertPDFToLP

The parameter string uses the following values:

- -s represents the string name
- -l represents the library name

  Note: This value is the letter "l," not the numeral "1."

- -f represents the function name

In the parameter string, the value for the library where the conversion program resides varies depending on the platform. For example, the value for NT, AS/400, and RS6000 is -l jdekrnl; whereas, the value for HP is -l ube. These values are the actual library names.

When you define the conversion program for your printer, be sure that you choose the correct program for you type of printer. For example, for a postscript printer use *JDE PS, for a PCL printer use *JDE PCL, and for a line printer use *JDE LINE.

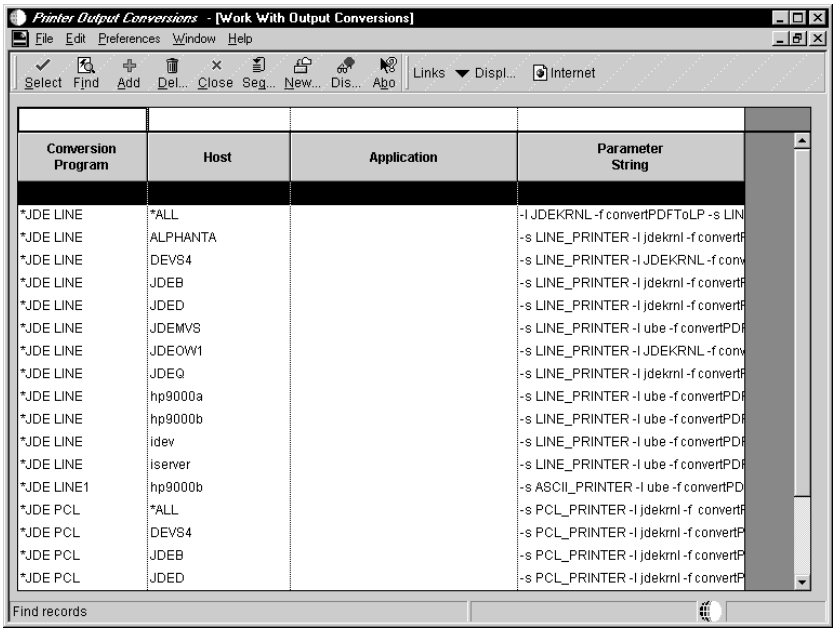
Complete the following tasks:

- Display conversion programs
- Define a conversion program
- Modify a conversion program
- Copy a conversion program
- Delete a conversion program

▶ To display conversion programs

On the Printers menu (GH9013), choose Work with Conversion Programs (P98616).

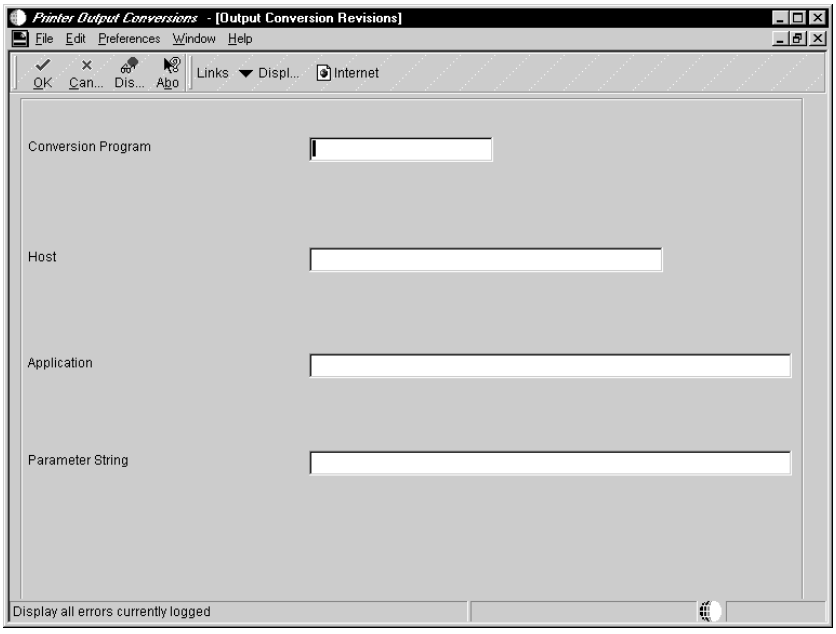
On Work with Conversion Programs, click Find.



▶ To define a conversion program

On the Printers menu (GH9013), choose Printer Output Conversions (P98616).

1. On Work with Conversion Programs, click Add.



2. On Conversion Program Revisions, complete the following fields:
   - Conversion Program

Currently the only valid conversion programs are those that are internally defined to OneWorld. Third party vendors might provide alternate conversion programs.

- Host
- External Application
- Parameter String

Field	Explanation
Conversion Program	The name of an executable program. For OneWorld: This name is the system name of a form that is associated with an application. To determine the system name of a form, open the form and choose About OneWorld from the Help menu.
Host	The host on which the job is submitted to execute.
Application	A printer conversion path and program that is executable.
Parameter String	A generalized 10 character parameter value passed to a called program. . . . . . *Form-specific information* . . . . . . A specific parameter string that calls a OneWorld-internal conversion program. Valid values are dependent on the type of printer and the hardware and software platform that controls the printer. Refer to the current documentation for actual values.

Example: Defining a Conversion Program

Your printer output conversion values should look similar to the following examples for company JDE:

Conversion Program	Host	Parameter String
*JDE LINE	*ALL	-s LINE_PRINTER -l jdekrnl -f convertPDFToLP

This example associates the *JDE LINE conversion program with the *ALL host, which represents any and every host in OneWorld, and the parameter string for a non-HP host.

In the parameter string, the value for the library where the conversion program resides varies depending on the platform. In this example, the host is non-HP and, therefore, requires -l jdekrnl for the library name. This value is the actual library name.

Conversion Program	Host	Parameter String
*JDE PS	HPServer	-s POSTSCRIPT_PRINTER -l ube -f convertPDFToPS

This example associates the *JDE PS conversion program with the HPServer host and the parameter string for an HP host.

In the parameter string, the value for the library where the conversion program resides varies depending on the platform. In this example, the host is HP and, therefore, requires -l ube for the library name. This value is the actual library name.

Conversion Program	Host	Parameter String
*JDE PCL	*ALL	-s PCL_PRINTER -l jdekrnl -f convertPDFToPCL

This example associates the *JDE PCL conversion program with the *ALL host, which represents any and every host in OneWorld, and the parameter string for a non-HP host.

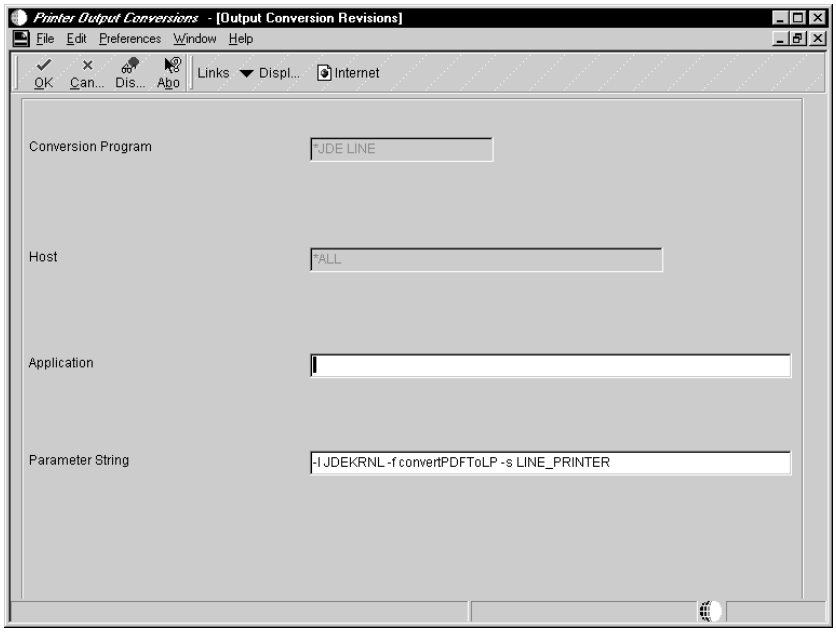
In the parameter string, the value for the library where the conversion program resides varies depending on the platform. In this example, the host is non-HP and, therefore, requires -l jdekrnl for the library name. This value is the actual library name.

For more information about printer output conversions, see the table under *Defining Printer Output Conversion Programs*.

▶ To modify a conversion program

On the Printers menu (GH9013), choose Printer Output Conversions (P98616).

1. On Work with Conversion Programs, click Find and choose an existing conversion program.



2. On Conversion Program Revisions, revise the following field as necessary:
    - Parameter String

▶ To copy a conversion program

On the Printers menu (GH9013), choose Printer Output Conversions (P98616).

1. On Work with Conversion Programs, click Find, select an existing conversion program, then click Copy.
2. On Copy Information, click OK.
3. On Conversion Program Revisions, revise the following fields as necessary, and click OK:
    - Conversion Program

      If you change the conversion program, you will need to modify the parameter string.

    - Host

      Change this field to create a conversion program that processes on a different server.
4. On Work with Conversion Programs, click Find to refresh the grid.

▶ To delete a conversion program

On the Printers menu (GH9013), choose Printer Output Conversions (P98616).

1. On Work with Conversion Programs, click Find, select the conversion program, then click Delete.
2. On Confirm Delete, click OK.

   The item you deleted should no longer appear in the grid.

Setting Up a Logical Printer

You set up a OneWorld printer by assigning a physical and logical name. You associate this printer with some basic printer characteristics such as paper type, conversion program, and line printer characteristics. These values are stored in the Printer Definition table (F98616). You cannot delete a physical printer definition that exists in F98616.

Line printer parameters are valid at print-time and override those settings that you might have chosen when you created the report.

Line Printer Considerations

The following characteristics determine most of the fundamentals for the appearance of a report before the batch engine creates the PDF file:

- Printer characteristics

  These are dependent on the device where the batch report prints.

- RDA characteristics

  These are determined when you are creating the report using the Report Design Aid (RDA) tool.

  When you use RDA to create reports, you should consider how the report will appear on the intended printer. This includes consideration of fonts, character spacing, and line spacing. For example, when you print to a line printer, use a fixed pitch font, such as New Courier 10 pt.

- Event rules

  ER can assign a printer name using a system function.

To set up printer definitions, complete the following tasks:

- Add a logical printer
- Modify a logical printer
- Copy a logical printer
- Delete a logical printer

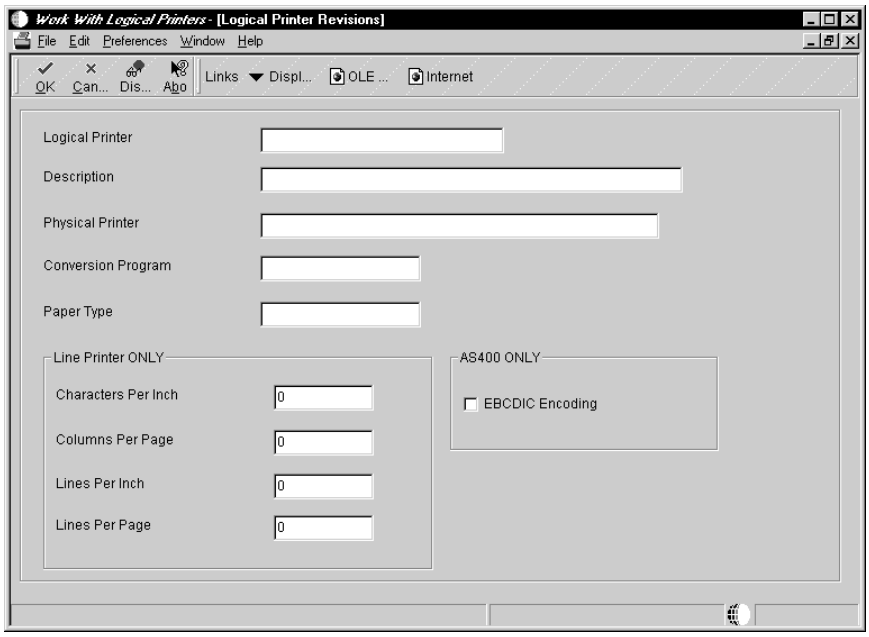
Before You Begin

- You must define paper types and associate those paper types with a physical device. You cannot complete a printer definition if the paper types are not defined for that specific physical device. The visual assist shows you the currently defined (and thus available) paper types for each physical printer. See *Defining Paper Types.*
- You must define physical printers and associate printer capabilities with a physical device. You cannot complete a printer definition if the printer capabilities are not associated with that specific physical device. The visual assist shows you the currently defined (and thus available) printer capabilities for each physical printer. See *Defining Physical Printer Capabilities.*
- You must define conversion programs. See *Defining Printer Output Conversion Programs.*

▶ To add a logical printer

On the Printers menu (GH9013), choose Work with Logical Printers (P98616).

1. On Work with Logical Printers, click Add.



2. Complete the following required fields:

   - Logical Printer
   - Description
   - Physical Printer
   - Conversion Program

     The conversion program that you choose depends on the printer type. For example, use *JDE PS for a postscript printer, *JDE PCL for a PCL printer, and *JDE LINE for a line printer.

   - Paper Type

3. If you are using a line printer as the output device, you can define additional line printer characteristics that define the page size and orientation. For example, the combination of Lines Per Page and Lines Per Inch determines the height of the page, and the combination of Characters Per Page and Characters Per Inch determines the width of the page.

   Complete the following optional fields:

   - Characters Per Inch

     The value you enter in this field determines the number of characters the given physical printer allows in one horizontal inch.

   - Columns Per Page

     The value you enter in this field determines the number of characters that appear in one line of text in the given report.

- Lines Per Inch

  The value you enter in this field determines the number of lines of text that the given physical printer allows in one vertical inch.

- Lines Per Page

  The value you enter in this field determines the number of lines of text that the given physical printer allows on one printed page.

4. **AS/400 only:** Click the following option to determine whether your output is EBCDIC or ASCII:

   - EBCDIC Encoding

     Turn on this option when you send your output from an AS/400 to an EBCDIC line printer.

5. Click OK to save your logical printer definition.
6. On Work with Logical Printers, click Find to refresh the grid.

Field	Explanation
Printer Name	The default printer device.
Description	A long description of a printer.
Physical Device	The name assigned to a specific printer on a network, such as Accounting_Printer_1. For OneWorld, this name cannot contain a space character, even if allowed by your operating system.
Conversion Program	The name of an executable program. For OneWorld: This name is the system name of a form that is associated with an application. To determine the system name of a form, open the form and choose About OneWorld from the Help menu.
Paper Type	A user defined code (H98/PT) that indicates the type of printer paper, such as letter or legal. For example, LETTER, LEGAL, and A4.
Characters Per Inch	A line printer parameter that specifies the number of characters per horizontal inch. Enter the number of characters per inch that is supported by your line printer. For example, 10 or 12.
Characters Per Page	A line printer parameter that specifies the number of columns per page. For example, 80 or 132.
Lines Per Inch	A line printer parameter that specifies the number of lines per vertical inch that are printed on a page. For example, 6 or 8.
Lines Per Page	A line printer parameter that specifies the number of lines per page. For example, 60 or 66.

Example: Setting Up a Printer Definition

Your printer definition values should look similar to the following examples:

Logical Printer	Physical Printer	Paper Type	Conversion Program
JDELaser Postscript AS400 - Letter	QGPL/JDEPRN14	LETTER	*JDE PS

In this example, the company JDE set up the logical printer name JDELaser Postscript AS400 - Letter for the physical printer QGPL/JDEPRN14, where QGPL represents the library name that stores the outqueues and JDEPRN14 represents the outqueue name and the printer for the outqueue. The logical printer name is the name OneWorld uses to recognize the physical printer. The paper type for the logical printer is LETTER. You must use the *JDE PS conversion program because the printer type is postscript.

A single physical printer can support both postscript and PCL.

Logical Printer	Physical Printer	Paper Type	Conversion Program
JDELaser PCL NT - Legal	\\devprnt\jdeprn14	LEGAL	*JDE PCL

In this example, the company JDE set up the logical printer name JDELaser PCL NT - Legal for the physical printer \\devprnt\jdeprn14, where devprnt represents the server name and jdeprn14 represents printer name. The logical printer name is the name OneWorld uses to recognize the physical printer. The paper type for the logical printer is LEGAL. You must use the *JDE PCL conversion program because the printer type is PCL.

A single physical printer can support both postscript and PCL.

You can set the paper types for A4 and LEGAL for the same physical printer. If you copy the record for a logical printer, then change the paper type. You might also include the paper type in the name for the logical printer. See the following task in this section:

- *Copy a logical printer*

Logical Printer	Physical Printer	Paper Type	Conversion Program
JDELine UNIX	jdeprn15	LETTER	*JDE LINE

In this example, the company JDE set up the logical printer name JDELine UNIX for the physical printer jdeprn15. The logical printer name is the name OneWorld uses to recognize the physical printer. The paper type for the logical printer is LETTER. You must use the *JDE LINE conversion program because the printer type is line.

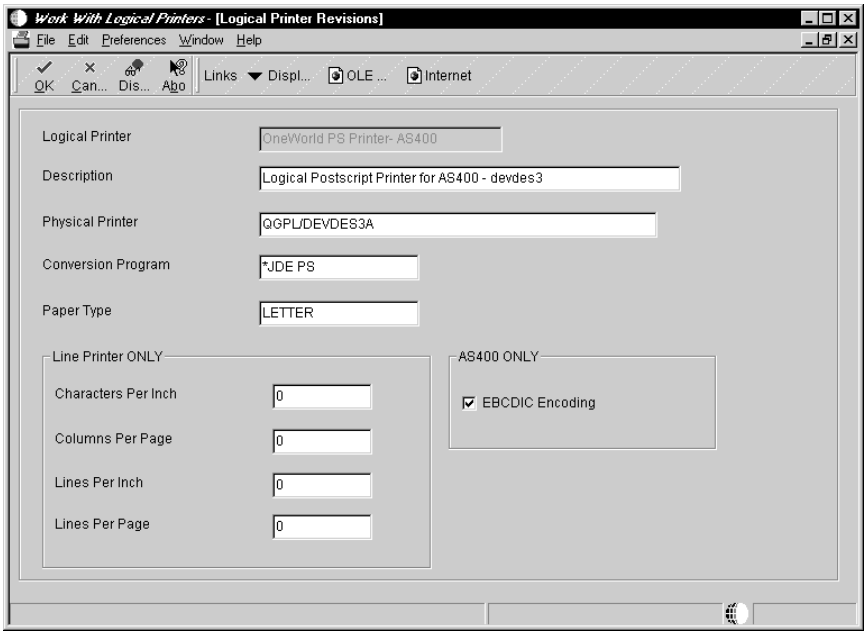
The following values represent examples of what you might enter for your line printer characteristics. These values are only examples and do not represent a standard.

- 10 characters per inch (CPI)
- 132 columns per page (CPP)
- 6 lines per inch (LPI)
- 51 lines per page (LPP)

▶ To modify a logical printer

On the Printers menu (GH9013), choose Work with Logical Printers (P98616).

1. On Work with Logical Printers, click Find and choose an existing logical printer definition.



2. On Logical Printer Revisions, revise the following fields as necessary:
   - Description
   - Physical Printer
   - Conversion Program
   - Paper Type
3. If you are modifying a line printer definition, revise the following fields as necessary:
   - Characters Per Inch

     The value you enter in this field determines the number of characters the given physical printer allows in one horizontal inch.

   - Columns Per Page

The value you enter in this field determines the number of characters that appear in one line text in the given report.

- Lines Per Inch

  The value you enter in this field determines the number of lines of text that the given physical printer allows in one vertical inch.

- Lines Per Page

  The value you enter in this field determines the number of lines of text that the given physical printer allows on one printed page.

4. **AS/400 only:** Click the following option to determine whether your output is EBCDIC or ASCII:

   - EBCDIC Encoding

     Turn on this option when you send your output from an AS/400 to an EBCDIC line printer.

5. Click OK to save your logical printer definition.
6. On Work with Logical Printers, click Find to refresh the grid.

▶ To copy a logical printer

On the Printers menu (GH9013), choose Work with Logical Printers (P98616).

1. On Work with Logical Printers, click Find, select a record, then click Copy.

   On Copy Information, click OK.

2. On Logical Printer Revisions, revise the following fields as necessary:

   - Logical Printer

     This is a required field. You must change the logical printer name for your new logical printer definition.

   - Description
   - Physical Printer

     If you change the physical printer to a machine that uses a different printer type, you also need to change the conversion program. For example, if the printer type for the physical printer changes from postscript to PCL, change the conversion program from *JDE PS to *JDE PCL.

   - Conversion Program

Verify that your conversion program is compatible with the printer type. For example, postscript printers must use *JDE PS, PCL printers must use *JDE PCL, and line printers must use *JDE LINE.

- Paper Type

  If you want to add definitions for other paper types to the same physical printer, you need to change this field for the new logical printer definition.

3. For modifications to a line printer definition, revise the following fields as necessary:

   - Characters Per Inch

     The value you enter in this field determines the number of characters the given physical printer allows in one horizontal inch.

   - Columns Per Page

     The value you enter in this field determines the number of characters that appear in one line text in the given report.

   - Lines Per Inch

     The value you enter in this field determines the number of lines of text that the given physical printer allows in one vertical inch.

   - Lines Per Page

     The value you enter in this field determines the number of lines of text that the given physical printer allows on one printed page.

4. Click the following checkbox to determine whether your output is EBCDIC or ASCII:

   - EBCDIC Encoding

     Turn on this option when you send your output from an AS/400 to an EBCDIC line printer.

5. Click OK to save your logical printer definition.
6. On Work with Logical Printers, click Find to refresh the grid.

▶ To delete a logical printer

On the Printers menu (GH9013), choose Work with Logical Printers (P98616).

1. On Work with Logical Printers, click Find, select the logical printer, then click Delete.
2. On Confirm Delete, click OK.

   The item you deleted should no longer appear in the grid.

Setting Up a Default Logical Printer

You set up default printers by connecting the logical name of a OneWorld printer to a *user name*/*environment*/*host name* combination. The following list provides an explanation for each of these values:

- User ID

  This value identifies the user who submitted the batch report. The value of the User ID field can be either a specific user name or *PUBLIC, where *PUBLIC indicates that the record applies to all users.

- Environment

  This value identifies the OneWorld environment from which the user submitted the batch report. The value of the environment field can be either a specific environment or *ALL, where *ALL indicates that the record applies to all environments.

- Host Name

  This value identifies the host on which the user submits the batch report to process. The value for the Host field can be either a specific host or *ALL, where *ALL indicates that the record applies to all hosts.

  Note: *ALL is valid only if all hosts are the same platform.

The batch engine determines a default printer by searching the Default Printer table (F986161). When determining a default printer, the system searches the table for a valid combination of user ID, environment, and host name values.

This task uses the following example to help explain the process of setting up a default printer:

- Example: Determining the Printer for a Report

Complete the following tasks:

- Add a default logical printer
- Change a default logical printer
- Copy a default logical printer

- Change the status of a default logical printer
- Delete a default logical printer

OneWorld begins by searching for specific information. OneWorld next looks for general information, such as *ALL entries for environment and host name and *PUBLIC for user name. If OneWorld finds no specific information, the system uses the default entry of *PUBLIC, *ALL, *ALL, for *user, environment,* and *host* respectively, to determine a default printer.

The following table provides an example of the hierarchical order used to determine the default printer. If the first valid combination fails, OneWorld tries the second, and so on.

Default Logical Printers - Valid Combinations					
	F986161 Table Field Names				
Valid Combination	User Name	Environment	Host Name	Logical Printer Name	Status
1	*user*	*env*	*host*	*printer1*	AV / NA
2	*user*	*env*	*ALL	*printer2*	AV / NA
3	*user*	*ALL	*host*	*printer3*	AV / NA
4	*user*	*ALL	*ALL	*printer4*	AV / NA
5	*PUBLIC	*env*	*host*	*printer5*	AV / NA
6	*PUBLIC	*env*	*ALL	*printer6*	AV / NA
7	*PUBLIC	*ALL	*host*	*printer7*	AV / NA
8	*PUBLIC	*ALL	*ALL	*printer8*	AV / NA

Example: Determining the Logical Printer for a Report

User USER01 needs to run the G/L by Business Unit report (R09420) to review a set of transactions. When the system formats the batch report, the system uses the following process flow to search the Default Printer table (F986161) for an entry that matches the user name, environment, and host name associated with the printer where the report prints:

▶ Process flow for determining the logical printer for a report

F986161 Table Field Names					
User Name	Environment	Host Name	Logical Printer Name	Status	Description
USER01	HPENV	HPServer	JDELaser Postscript UNIX	AV	Active
*PUBLIC	*ALL	*ALL	JDELaser Postscript NT	AV	Active

1. OneWorld first searches in F986161 for a USER01 user entry.
2. Next, OneWorld checks the environment associated with USER01.

   An entry for NTENV and USER01 exists in the table.

3. Next, OneWorld checks the host assigned to the user ID/environment combination.

   OneWorld finds the host NTServer in connection with this user ID/environment combination.

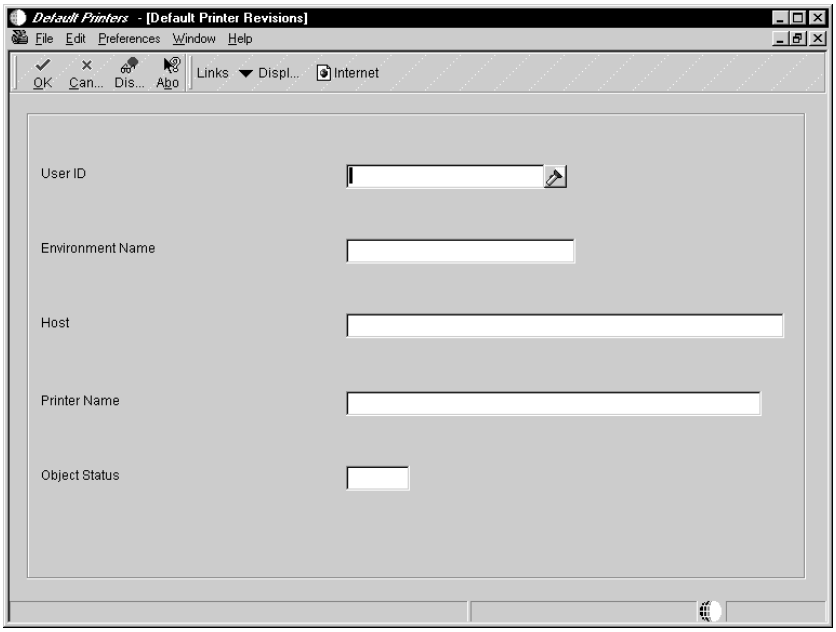
4. With the combination of USER01, NTENV, and NTServer, OneWorld determines that the default printer is JDELaser Postscript NT. The status for this printer is active (AV).

   When the system finds such an entry in the table, and the record is active, the system assigns the applicable printer name as the default printer for the entry.

▶ To add a default logical printer

On the Printers menu (GH9013), choose Work with Default Logical Printers (P98616).

1. On Work with Default Logical Printers, click Add.



2. On Default Logical Printer Revisions, complete the following required fields:
   - User ID
   - Environment Name

- Host
- Printer Name
- Object Status

Field	Explanation
User ID	For World, the IBM-defined user profile. For OneWorld, the identification code for a user profile.
Environment Name	For World, the Environment name is also called the Plan Name and is used to uniquely identify an upgrade environment for Install/Reinstall. For OneWorld (Install Applications), the environment name is also called the Plan Name and is used to uniquely identify an upgrade environment for Install/Reinstall. For OneWorld (Environment or Version Applications), this is the path code that identifies the location of the application or version specification data.
Host	The host on which the job is submitted to execute.
Printer Name	The default printer device.

Example: Setting Up a Default Logical Printer

Your default printer settings should look similar to the following examples:

F986161 Table Field Names					
User Name	**Environment**	**Host Name**	**Logical Printer Name**	**Status**	**Description**
USER01	HPENV	HPServer	JDELaser Postscript UNIX	AV	Active
*PUBLIC	*ALL	*ALL	JDELaser Postscript NT	AV	Active

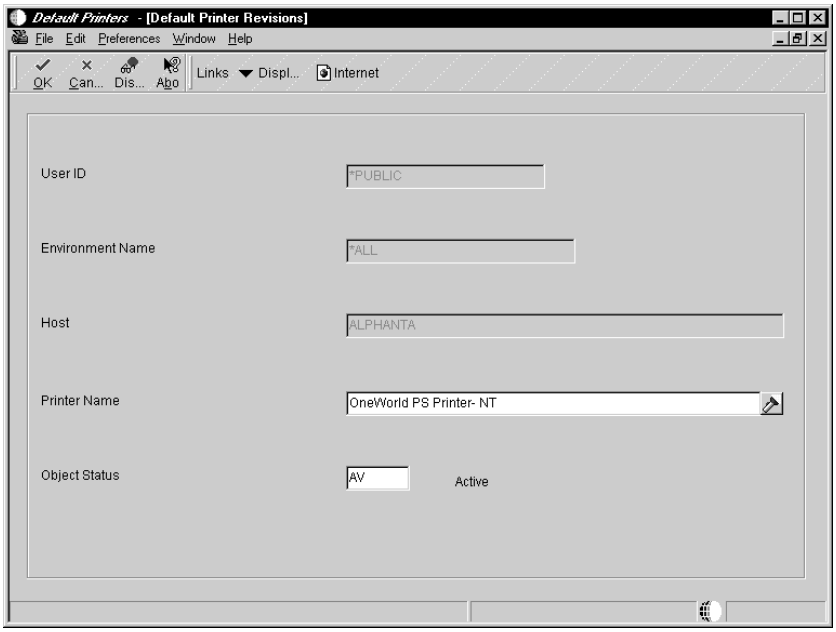
In this example, company JDE set the default printer for USER01 as JDELaser Postscript UNIX. The Default Printer Table (F986161) determines USER01's printer first by determining the user name, USER01, then the environment, HPENV, then the host name, HPServer. This combination uses JDELaser Postscript UNIX as the default printer.

The second entry in this example is the system default. If F986161 cannot reconcile an entry for the user name, the table uses *PUBLIC. If F986161 cannot reconcile the environment or the host name, the table uses *ALL. This combination of *PUBLIC, *ALL, *ALL ensures that a report that fails to meet any specific criteria in F986161 does not cause an error that stops batch processing in a batch queue.

To change a default logical printer

On the Printers menu (GH9013), choose Work with Default Logical Printers (P98616).

1. On Work with Default Logical Printers, click Find and choose an existing default printer definition.



2. On Default Logical Printer Revisions, revise the following field:
   - Printer Name

To copy a default logical printer

On the Printers menu (GH9013), choose Work with Default Logical Printers

1. On Work with Default Logical Printers, click Find, then select a default printer and click Copy.
2. On Default Printer Revisions, revise the following fields as necessary:
   - User ID

     Change this field to give the current default logical printer settings to another user.

   - Environment Name

     Change this field to specify another environment from which a report will be submitted.

- Host

  Change this field to specify another server where a report will process.

- Printer Name
- Object Status

3. Click OK to save your changes.
4. On Work with Default Logical Printers, click Find to refresh the grid.

▶ To change the status of a default logical printer

On the Printers menu (GH9013), choose Work with Default Logical Printers (P98616).

1. On Work With Default Printers, click Find and choose an existing default printer definition.
2. On Default Revisions, revise the following field:
   - Object Status

▶ To delete a default logical printer

On the Printers menu (GH9013), choose Work with Default Logical Printers

1. On Work with Default Logical Printers, click Find, select the record in the grid, then click Delete.
2. On Confirm Delete, click OK.

   The item you deleted should no longer appear in the grid.

Generating and Retrieving Logs for Your Report

When you run a OneWorld report, you can specify whether you want to create logs for the report. The logs you can create are the jde.log and the jdedebug.log. These logs allow you to review how your reports process on the server. These logs reside in a specific directory on the server. Your jde.ini settings determine the location of this directory. Also, depending on the platform you use, the jde.ini setting differs slightly. The following list provides sample jde.ini settings for the directory where your report logs reside:

- AS/400

```
[INSTALL]
DefaultSystem=B733SYS
```

  Example path: B733SYS\PRINTQUEUE

- UNIX

```
[INSTALL]
B733=/usr/oneworld/output
```

  Example path: /usr/oneworld/output/PrintQueue

- Windows NT Server

```
[INSTALL]
B733=d:\oneworld\output
```

  Example path: d:\oneworld\output\PrintQueue

The default directory for your log files is PrintQueue, which becomes a subdirectory to the directory you designate in the jde.ini file. You can change the location of this directory as necessary.

These jde.ini settings also determine the location where your report output resides after processing. If you set your jde.ini to save the output for your reports, OneWorld saves a PDF file for the report in the report output directory.

To create logs for your report

On System Administration Tools (GH9011)

1. Choose Batch Versions (P98305).

   The Work with Batch Versions form appears. On this form you can locate and run reports. Also, you can modify version detail information, data selection, and data sequencing.

2. Type an application ID in the Batch Application field and click Find. For example, to locate a version for the One Line Per Address report, type R014021.
3. In the grid, highlight a version to submit, then click Select.

   The Version Prompting form appears. On this form, you can choose to change the data selection, change the data sequencing, and access the Advanced Operations form.

4. From the Form menu, choose Advanced.

   The Advanced Operations form appears. On this form, you can override the location where your report processes, activate the jde.log, activate the jdedebug.log, and modify the level of information that your logs include.

5. Modify the following information, then click OK:
   - Logging (JDE.log)

     Turn on this option to activate a basic log that helps you determine when a fault occurs during a batch process.

   - Tracing (JDEDEBUG.log)

     Turn on this option to turn on advanced UBE logging that includes details about the batch process.

   - UBE Logging Level

     The value you enter here, from 0-6, determines the level to which your batch process log shows errors ranging from error messages to object level messages and UBE function messages.

     Note: When you choose a high value to receive more technical information, you also receive all the information for the lower values. For example, when you enter a value of 6 (UBE function messages), you also receive information for values 0-5.

6. On Version Prompting, click Submit to run your report and create your logs.

Field	Explanation
JDE Logging	When the batch job is run on a server, this field allows you to indicate if JDE logging should be enabled for the execution. If the server is already set to perform JDE logging, it occurs regardless of how this field is set.
Tracing	When the batch job runs on a server, this field indicates whether tracing is enabled for execution of the job. If the server is already set to perform tracing, it occurs regardless of how this field is set.
UBE Logging Level	Indicates the type of error logging that occurs when the batch job runs. The following list describes the different levels: 0 Error Messages 1 Informative Messages and Log Entry 2 Section Level Messages 3 Object Level Messages 4 Event Rule Messages 5 Database Mapping Messages 6 UBE Internal Function Calls, Textout Values

Work with Servers

The Work With Servers program (P986116) provides a central location from which system administrators can monitor and control:

- Server jobs
- OneWorld subsystems

As a system administrator, you can use the Work with Servers program to print, view, remove, terminate, release, or hold any jobs that currently reside in a queue on any OneWorld server. Similarly, workstation users can control only those jobs submitted by them. This is generally restricted to only those jobs associated with a specific user ID.

You can also use the Work with Servers program to end and to stop OneWorld subsystems. In addition, you can view the status of OneWorld subsystems that are running or are waiting to process jobs.

This section describes the following:

- ❑ Working with server jobs
- ❑ Working with OneWorld subsystems

Working with Server Jobs

Using the Work with Servers application, system administrators can print, view, delete job records from the outqueue. They can also terminate, release, or hold any jobs that currently reside in a queue on any OneWorld server. Similarly, using the Submitted Reports applications, workstation users can generally control only those jobs submitted by them.

You should use OneWorld security to restrict access to the Work with Servers application. Access to this program should generally granted only to administrator-level users. This is because XJDE0001 version of Work with Servers version this program (P986116) allows users to view and control server jobs for all users. End users should be restricted to the XJDE0002 version which is known as the Submitted Reports application (P986116). This version of the application restricts users to viewing and modifying only those jobs which were initially submitted under their User ID. Both programs are located on the System Administration Tools menu (GH9011).

From within Work with Servers, on the Submitted Job Search form, you can access the following options from the Row menu:

- Print allows you to print jobs with a status of E or D. Using a standard print form, you can print to your default printer or print to another available printer.
- View Job allows you to launch the Adobe Acrobat Reader program and display your report online. You can review your report online, and then print to your default printer or another available printer.
- View Log allows you to view the jde.log and the jdedebug.log.
- Delete (found on the toolbar) allows you to delete the record and the job from the outqueue. Use this on jobs with status E or D.
- Terminate ends the job if it is processing. This option does not remove the job, but moves it to an E (end) status, thereby allowing a user to view the logs.
- Hold stops a job until you release it again.
- Release removes the H (held) status from a job and sends it into the queue.

This topic contains the following:

- ❑ Printing jobs
- ❑ Viewing reports online
- ❑ Viewing the logs for a job
- ❑ Terminating jobs
- ❑ Holding and releasing jobs

Checking the Status of Reports

After you submit your report, you can check the status of your job in the queue. Depending on the status of your job, you can perform such tasks as printing your report, viewing your report output online, deleting your report, and holding your report in the queue.

▶ To check the status of reports

On System Administration Tools (GH9011)

1. Choose Work with Servers (P986116).

   The Work with Servers form appears. On this form you can access a list of servers used to process batch applications.

2. Click Find to locate all servers, or use the query by example line to locate a specific server.
3. Choose a server with which to work and click Select.

   The Submitted Job Search form appears. From this form you can, print, view, delete, and hold your job. You can also view logs with detailed information about how your report processed.

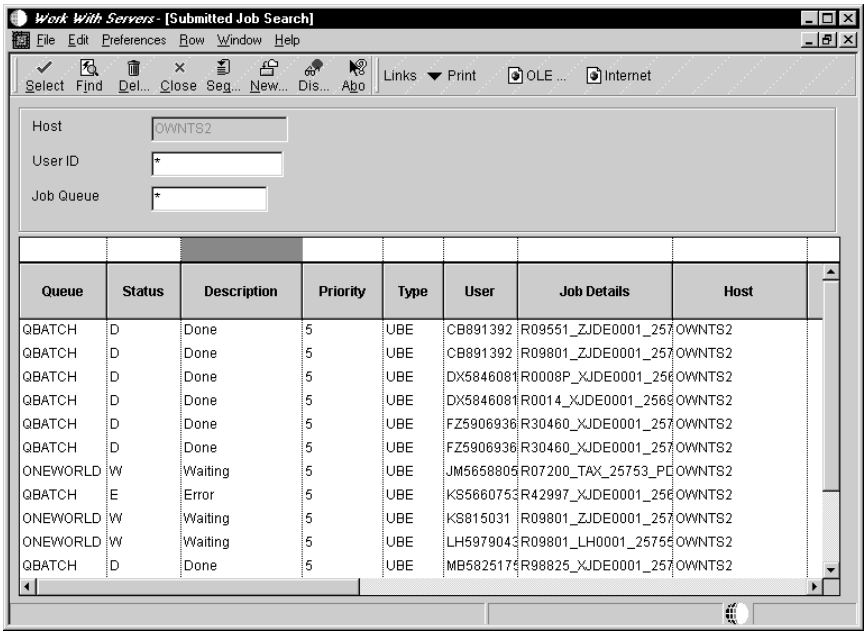
4. Complete the following fields then click Find to search for submitted jobs:
   - User ID

     The default user ID is the one signed onto the current OneWorld session. This user ID can be changed if you wish to work with a report submitted by a different user. You can use a wildcard (*) to find the user you want.

   - Job Queue

     Enter the name of the logical queue on the server for which you want to view jobs.

The detail portion of the form displays the jobs and their status. You can use the visual assist in the Status field to reads the UDCs for status codes in your installation.



Changing the Priority and the Printer for Jobs

To move your job up or down in the queue, you can change the priority of the job while the job is at the status of W (Waiting). You might choose to move more important jobs up in the queue and move those with less priority down in the queue. You can also choose to override the location where your job prints.

▶ To change the priority and the printer for jobs

On System Administration Tools (GH9011)

1. Choose Work with Servers (P986116).

   The Work with Servers form appears. On this form you can access a list of servers used to process batch applications.

2. Click Find to locate all servers or use the query by example line to locate a specific server.
3. Choose a server with which to work.

4. From the Row menu, choose Server Jobs.

   The Submitted Job Search form appears. By default, jobs are listed for the User ID for the requesting workstation. Depending on your application security level, you can change the User ID field and the Job Queue field to search for other jobs.

   Note: A job must be at a status of W (Waiting) to change the priority.

5. Choose a job with which to work and click Select.

   The Job Maintenance form appears. On this form, you can review information about your batch job and modify the priority of the job and the printer on which the job will print.

6. Modify the following information, then click OK:
   - Job Priority
   - Printer

Field	Explanation
User ID	For World, The IBM-defined user profile. For OneWorld, the creator or submitter of the version or job.
Job Queue	The job queue to which the job was submitted. On the AS/400 this is an actual system job queue. On other systems it is a OneWorld logical queue.
Job Priority	The priority level of a submitted job. Jobs will execute based on this priority. Values 0-9 are valid where 0 is the highest priority.
Printer	Specifies the name of the printer device on which you want your job to print. If you do not know the name, you can choose Change Printer from the Form menu to access a list of available printers.

Printing Jobs

For jobs with a status of D (Done) and E (Error), you can send your job directly to your default printer without viewing the PDF file online. A status of D means that the processing for your job completed successfully. A status of E means that an error occurred during processing. If you print a job with a status of E, you print an error log to aid you when you troubleshoot your report. Refer to *Troubleshooting the Workstation* and *Troubleshooting the Enterprise Server* in the *Server and Workstation Administration* guide for more information about troubleshooting and logs.

To print jobs

On System Administration Tools (GH9011)

1. Choose Work with Servers (P986116).

   The Work with Servers form appears. On this form you can access a list of servers used to process batch applications.

2. Click Find to locate all servers or use the query by example line to locate a specific server.
3. Highlight a server with which to work.
4. From the Row menu, choose Server Jobs.

   The Submitted Job Search form appears. Jobs specific to your user ID appear in the grid on this form by default. Depending on your security level, you can change the User ID field and the Job Queue field to search for other jobs.

5. Highlight the job you want to print, then from the Row menu, choose Print.

   The Printer Selection form appears. This form provides printer-specific information as well as information about the format of your report.

You can change the printer for the job by choosing Change Printer from the Form menu. Also, you can determine whether your report prints as a landscape or a portrait document.

6. Click OK to print your job.

Viewing Reports Online

After your job finishes processing on the server, you can view the report output online. For most jobs, the output will be in Portable Document Format (PDF) which is viewable with Adobe Acrobat Reader. When you view your report output online, OneWorld also creates a PDF file for the report in the following directory on your workstation:

- \b7\PrintQueue

You can attach PDF files to e-mail messages, move or copy the files, and because most current Web browsers can read PDF files, you can post your reports to a Web site. Also, you can copy text from Acrobat Reader to the clipboard and paste the text into other applications.

Before You Begin

Before you view your report online, verify that you have Adobe Acrobat Reader installed on your workstation.

▶ To view reports online

On System Administration Tools (GH9011)

1. Choose Work with Servers (P986116)

   The Work with Servers form appears. On this form you can access a list of servers used to process batch applications.

2. Click Find to locate all servers, or use the query by example line to locate a specific server.
3. Choose a server with which to work.
4. From the Row menu, click Select (or choose Server Jobs).

   The Submitted Job Search form appears. Jobs specific to your user ID appear on this form by default. Depending on your security level, you can change the User ID field and the Job Queue field to search for other jobs.

5. Choose the job you want to view, then from the Row menu, choose View Job.

   Adobe Acrobat Reader opens to display an online version of your report output. Refer to Adobe Acrobat Reader online help for more information about using Acrobat Reader.

Viewing the Logs for a Job

You can view logs that detail the steps taken while your job processed. From the Submitted Job Search form, you can access the jde.log and the jdedebug.log for your report. These logs are helpful if you need to troubleshoot why a report resulted in error. These logs exist on the machine where the job ran, which may not necessarily be the same machine as your workstation.

The jde.log is a general purpose log used to track error messages generated by OneWorld processing. The jde.log tracks any fault that might occur within OneWorld including whether the sign on is successful. When you are looking for startup errors you should read the jde.log from the top down. For other errors, you should read from the bottom up. See *Working With the Workstation Log Files* and *Working with the Enterprise Server Log Files* in the *Server and Workstation Administration Guide.*

The jdedebug.log contains API calls and SQL statements as well as other messages. You can use this log to determine the point int time when normal execution stopped. The system does not use jdedebug.log to track errors. Instead, it uses this log to track the timing of OneWorld processes. See *Working With the Workstation Log Files* and *Working with the Enterprise Server Log Files* in the *Server and Workstation Administration Guide.*

▶ To view the logs for a job

On System Administration Tools (GH9011)

1. Choose Work with Servers (P986116)

   The Work with Servers form appears. On this form you can access a list of servers used to process batch applications.

2. Click Find to locate all servers, or use the query by example line to locate a specific server.
3. Choose the server that processed the job you want to troubleshoot.
4. From the Row menu, click Select choose Server Jobs.

   The Submitted Job Search form appears. Jobs specific to your user ID appear on this form by default. Depending on your security level, you can change the User ID field and the Job Queue field to search for other jobs.

5. Choose the job for which you want to view a log, then from the Row menu, choose View Logs.

   The View Logs form appears. On this form, you can choose to view the jde.log and the jdedebug.log.

6. Click OK to view the logs.

   Note: If you choose both the jde.log and the jdedebug.log, the logs open in the same window. To view the logs separately, you must choose the logs separately.

Terminating Jobs

If your job is processing, you can manually end the job. When you terminate a job, you do not delete the job, but you move the job to the status of E (Error). With the job at the status of E, you can print an error log or delete the job.

▶ To terminate jobs

On System Administration Tools (GH9011)

1. Choose Work with Servers (P986116)

   The Work with Servers form appears. On this form you can access a list of servers used to process batch applications.

2. Click Find to locate all servers, or use the query by example line to locate a specific server.
3. Choose a server with which to work.
4. From the Row menu, click Select (or choose Server Jobs).

   The Submitted Job Search form appears. Jobs specific to your user ID appear on this form by default. Depending on your security level, you can change the User ID field and the Job Queue field to search for other jobs.

5. Choose the job to terminate, then from the Row menu, choose Terminate.

   Note: A job must be at a status of P (Processing) to terminate the job.

6. Click Find to update the detail area.

   The status of the job changes to E (Error).

Holding and Releasing Jobs

If a job is at the status of W (Waiting), you can hold the job. You might choose to hold a job if the job is large enough to impact the performance of the server on which it processes. You can release the job at a time when an impact to performance is not an issue, for example, after regular business hours.

If you want to stop a job that is at a status of P (Processing), you must terminate the job. When you terminate a job, you do not remove the job, but you move the job to the status of E (Error). You cannot restart a job after you terminate the job. You must resubmit the job to the server.

Complete the following:

- Hold a job
- Release a job

▶ To hold a job

On System Administration Tools (GH9011)

1. Choose Work with Servers (P986116)

The Work with Servers form appears. On this form you can access a list of servers used to process batch applications.

2. Click Find to locate all servers, or use the query by example line to locate a specific server.
3. Choose a server with which to work.
4. From the Row menu, click Select (or choose Server Jobs).

   The Submitted Job Search form appears. Jobs specific to your user ID appear on this form by default. Depending on your security level, you can change the User ID field and the Job Queue field to search for other jobs.

5. Choose the job to hold, then from the Row menu, choose Hold.
6. Click Find to update the detail area.

   The status of the job changes to H (Hold).

▶ To release a job

On System Administration Tools (GH9011)

1. Choose Work with Servers (P986116)

   The Work with Servers form appears. On this form you can access a list of servers used to process batch applications.

2. Click Find to locate all servers, or use the query by example line to locate a specific server.
3. Choose a server with which to work.
4. From the Row menu, click Select (or choose Server Jobs).

   The Submitted Job Search form appears. Jobs specific to your user ID appear on this form by default. Depending on your security level, you can change the User ID field and the Job Queue field to search for other jobs.

5. Choose the job to release, then from the Row menu, choose Release.

   The job must be at the status of H (Hold).

6. Click Find to update the detail area.

   The status of the job changes to reflect the position of the job in the queue, for example, W (Waiting), S (In Queue), or P (Processing).

Working with OneWorld Subsystems

OneWorld uses various subsystems to offload processing from the enterprise server. This chapter provides you with details on:

- ❑ Understanding OneWorld Subsystems
- ❑ Locating OneWorld subsystems running on a server
- ❑ Viewing job records for OneWorld subsystems
- ❑ Terminating OneWorld Subsystems

Understanding OneWorld Subsystems

This discussion provides you with a basic understanding of how a subsystem is defined and used within OneWorld. You will also find out how you can enable OneWorld subsystems.

What Are OneWorld Subsystems?

The term *subsystem* is an industry generic term, usually indicating a system that is a subprocess to an operating system. On AS/400 server platforms, a subsystem is a logical process that is used to run system jobs, whether they are OneWorld or other application jobs. For UNIX, a OneWorld subsystem is functionally equivalent to a daemon. On UNIX and Windows NT server platforms, system jobs are processed in queues. These queues are functionally equivalent to subsystems on the AS/400.

Within OneWorld, subsystems are defined as continuously running batch jobs that run independently of, and asynchronously with, OneWorld applications. These OneWorld subsystem jobs function within the operating system's logical process or queue defined for the server platform. You can configure OneWorld to use one or more subsystems. See *Defining Subsystem Jobs* in the *Tools Guide.*

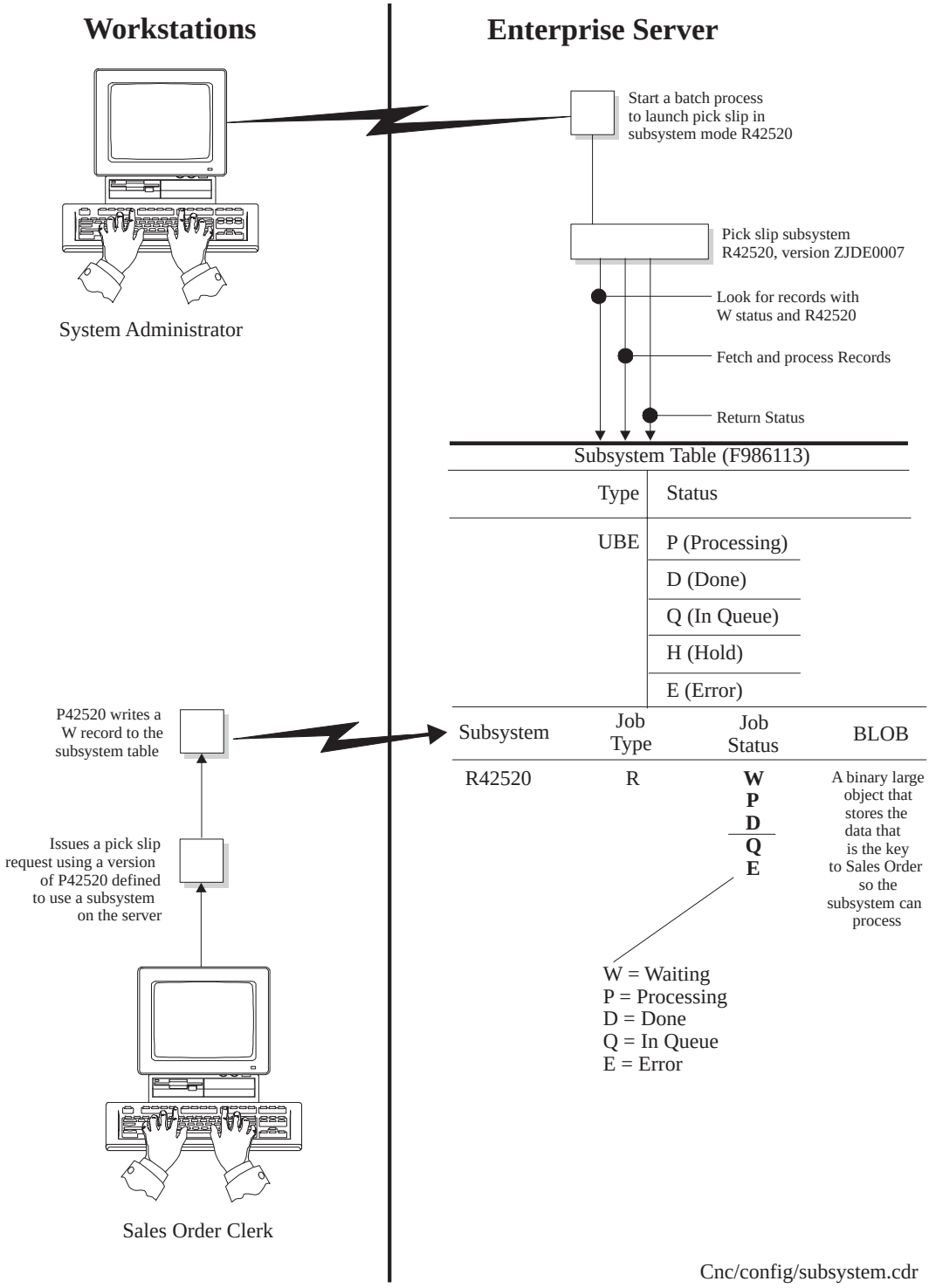
How Does OneWorld Use Subsystems?

Some OneWorld applications are design to use subsystems. For example, you can instruct Sales Order Processing to print pick slips through a OneWorld subsystem. You activate a subsystem through the processing options of an interactive application. Then you create a specific version of the interactive application using that processing option to run the application in subsystem mode. When started, OneWorld subsystems run continuously looking for and processing requests from OneWorld applications. Subsystems run until you terminate them.

Typically, you use subsystem jobs running on the enterprise server to offload processor resources from the workstation. Instead of queuing requests and running them in batches at specified points in the day, you can direct the requests to a subsystem where they are processed in real time. For example, you might be executing the Sales Order Entry application on a workstation and want to print pick slips. If you are using a version of pick slips that has the Subsystem Job function enabled, the request is executed by a OneWorld subsystem job. The pick slip request is routed to and processed by the subsystem job on the defined enterprise server. As a result, no additional processing resources are required of the workstation machine to actually print the pick slip.

When an application issues a request for a job to run in a subsystem, it places a record in the Subsystem Job Master table (F986113). These records are identified by subsystem job name and contain status and operational indicators. Embedded in the record is key information that allows the OneWorld subsystem to process the record without additional interaction with the requesting application. The continuously running OneWorld subsystem monitors the records in this table. If the subsystem finds a record with its process ID and appropriate status indicators, it processes the record and updates the status accordingly.

The following illustration shows the logical sequence of events associated with subsystems.



How Do You Enable OneWorld Subsystems?

To prevent excessive processing overhead during server startup, and to prevent unnecessary uses of processor resources for OneWorld subsystems jobs that might be in use, you must manually start OneWorld subsystems. This is generally the responsibility of the system administrator or manager-level user. The manual start is done by running a version of a OneWorld batch process that has a processing option set to enable the use of subsystems.

As described in the following table, the manner in which you initially control the creation and start up of these subsystems and queues depends upon your server platform.

Platform	Subsystem or Queue	Description
AS/400	JDENET	There is one AS/400 subsystem that is used for OneWorld. This subsystem is automatically started when you issue the OneWorld startup command STRNET. The subsystem name is version-specific. For example, for release B73.2 the subsystem name is JDEB732. To process requests destined for OneWorld subsystems, you must define a specific job queue running under the JDENET subsystem. For example, a job queue might be named QBATCH. User requests for OneWorld subsystem-defined batch jobs are executed by the job queue based upon definition in the AS/400 user profile. See *Understanding Batch Process Administration for AS/400* in the *Server and Workstation Administration Guide*.
UNIX	jdequeue	There can be one or more queues for OneWorld in UNIX environments. These queues can be named the same or differently. You define queues by parameters in the startup shell script RunOneWorld.sh. To process requests destined for OneWorld subsystems, you must define one or more queues. For example, a jdequeue might be named QBATCH. User requests for OneWorld subsystem-defined batch jobs are executed by the job queue based upon their process ID. See *Understanding Batch Process Administration for UNIX* in the *Server and Workstation Administration Guide*.
Windows NT	jde.ini settings	There can be one or more queues for OneWorld in Windows NT environments. These queues must have the same name. You define queues by settings in the jde.ini file. To process requests destined for OneWorld subsystems, you must define the name and number of queues in the [NETWORK QUEUE SETTINGS] section of the jde.ini file. For example, a jdequeue might be named QBATCH. User requests for OneWorld subsystem-defined batch jobs are executed by the job queue based upon their process ID. See *Understanding Batch Process Administration for NT* in the *Server and Workstation Administration Guide*.

System administrators can display all OneWorld subsystems running on a server using the Work with Server Jobs application (P986113). Use this application to:

- Locate a list of OneWorld subsystems running on a server
- Locate a list of OneWorld subsystem records that are unprocessed (not available for AS/400 servers)
- Locate the current record that a OneWorld subsystem is processing (not available for AS/400 servers)
- Stop or delete any OneWorld subsystem

Locating OneWorld Subsystems Running on a Server

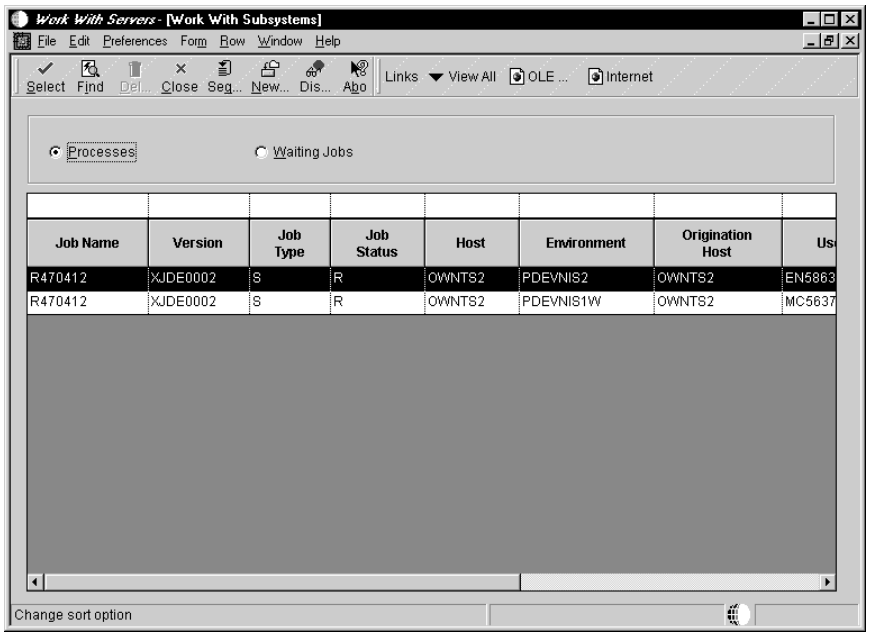
You can use Work With Servers to determine which OneWorld subsystems are currently running or waiting on a particular server. The running subsystems are identified by report number and version.

▶ To locate OneWorld subsystems running or waiting on a server

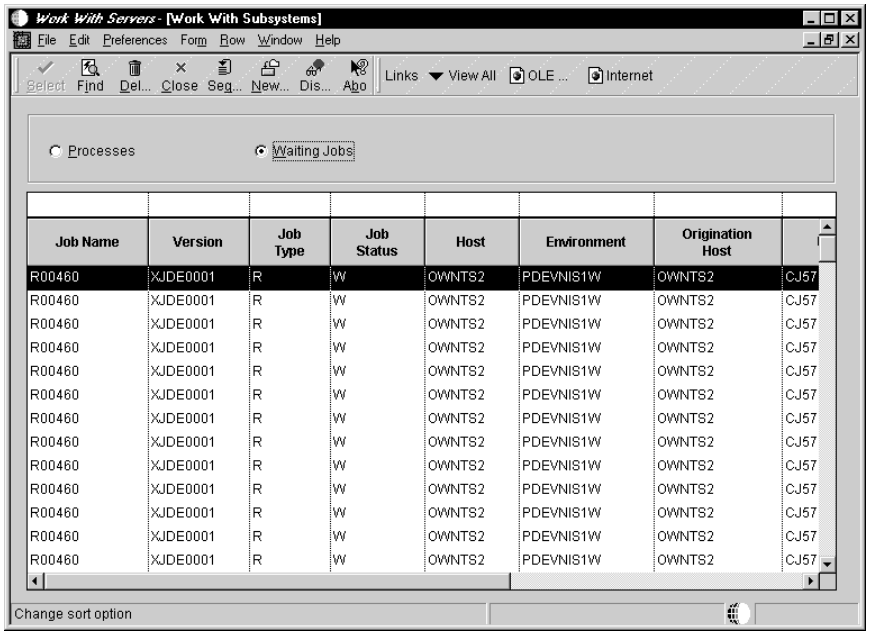
On System Administration Tools (GH9011)

1. Choose Work with Servers (P986116).
2. On Work With Servers, click Find to locate all servers, or use the query by example line to locate a specific server.
3. Choose a server with which you wish to work.
4. From the Row menu, choose Subsystem Jobs.
5. On Work With Server Jobs, click one of the two options:
   - Processes

     A process is a subsystem that is waiting for work. This is identified by an "S" (subsystem job) value in the Job Type field.



- Waiting Jobs



Waiting jobs are report jobs that are queued for a subsystem. This is identified by a “R” (subsystem record) value in the Job Type field.

All currently running OneWorld subsystems are displayed. The status of each subsystem is shown by codes in the following fields:

- Job Type

  This field indicates whether the job is a subsystem record record or a subsystem job. Valid values are:

  R Subsystem Record

  S Subsystem Job

- Job Status

  This field indicates whether the status of the subsystem job or record. Valid values are:

  W Subsystem record waiting

  P Subsystem record processing

  E Subsystem record to end the job

  R Subsystem job running

Viewing Job Records for OneWorld Subsystems

Different OneWorld processes write records to the Subsystem Job Master table (F986113). Each record is identified with a status code that identifies subsystem request types and operational status. You can use Work With Server Jobs to view the records in this table.

Before You Begin

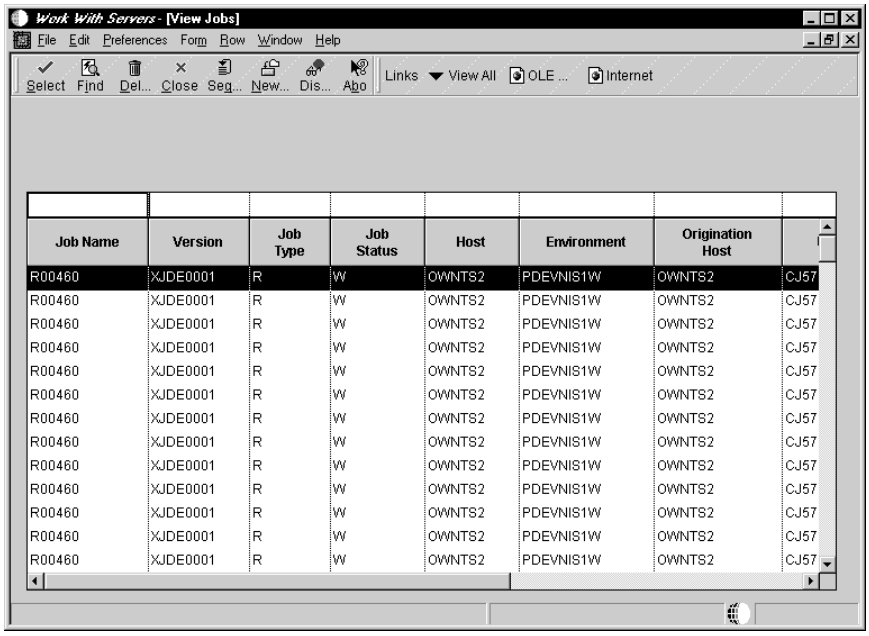
- Locate a OneWorld subsystem job. See *Locating OneWorld Subsystems Running on a Server*

▶ To view job records for OneWorld subsystems

On Work With Subsystems

1. Click Find, choose a record in the grid, then from the Row menu, choose View Jobs.
2. On View Jobs, click Find.

A list is displayed for all server jobs in the Subsystem Job Master (F986113) with an R (subsystem job running) job type.



Terminating OneWorld Subsystems

You can use Work With Server Jobs to terminate OneWorld subsystems. Two methods of termination are available:

- Stopping a subsystem job causes it to terminate after it completes processing the current record. Additional unprocessed records in the Subsystem Job Master table (F986113) will not be processed, and no new records can be written. The unprocessed records will essentially be lost. That is, the process that initiated the record is not notified that the record was not processed.
- Ending a subsystem job causes it to terminate after processing all of the existing subsystem records. No new records can be written to the Subsystem Job Master table (F986113).

Complete the following tasks:

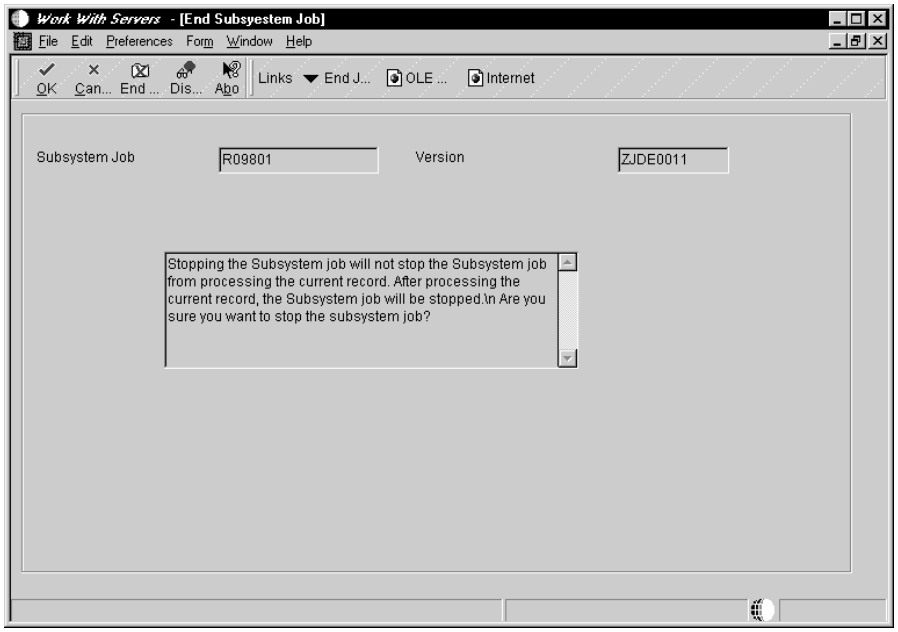
- Stop a OneWorld subsystem
- End a OneWorld subsystem

To stop OneWorld subsystems

On Work With Subsystems

1. Locate a running subsystem.
2. Choose the running subsystem that you want to stop.
3. From the Row menu, choose Stop Subsystem.

   Note: If you are viewing Waiting Jobs from Work with Server Jobs, or if you are viewing subsystem jobs by choosing the View Jobs from Work With Server Jobs, the Stop Subsystem selection is disabled from the Row menu selection.



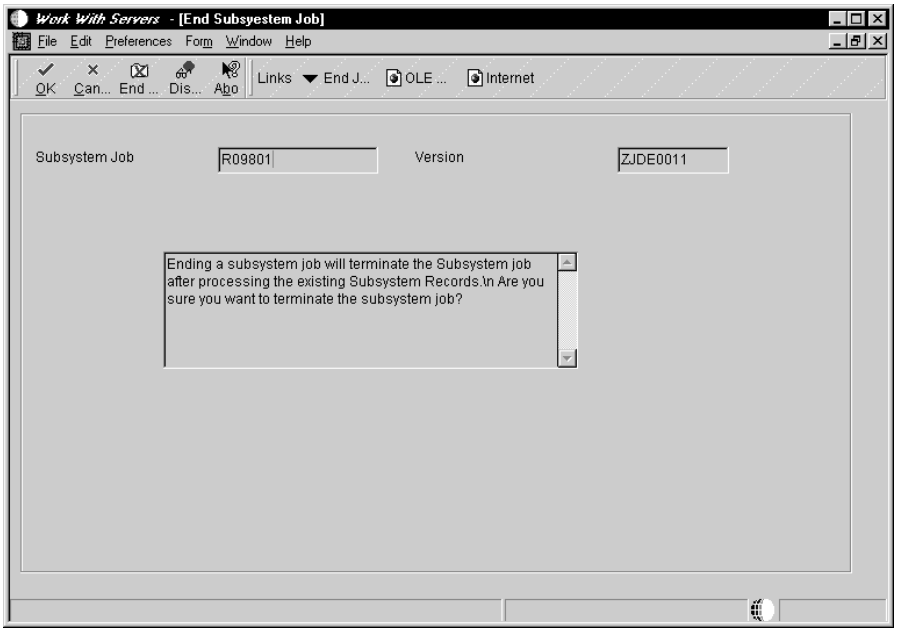
4. On End Subsystem Job, click OK.

To end OneWorld subsystems

On Work With Subsystems

1. Locate a running subsystem.
2. Choose the running subsystem that you want to end.
3. From the Row menu, choose End Subsystem Job.

   Note: If you are viewing Waiting Jobs from Work With Subsystems, the End Subsystem selection is disabled from the Row menu selection.



4. On End Subsystem Job, click OK.

Menu Design

You use Menu Design (P0082) to create, change, delete, copy, and filter menus and menu selections. Menu Design assists you in the following:

- Managing menus and menu selections
- Managing text overrides
- Defining runtime messages for menu selections
- Indicating the consequences of using particular menu selections
- Copying menu selections

In this section you will learn how to:

- ❑ Working with menus
- ❑ Working with menu selections
- ❑ Working with menu selection revisions

Understanding Menus

A menu is the entry point for running reports and applications. The Menu Master (F0082) table stores the following information, which identifies and characterizes the menu:

- Identifying information (ID and related system code)
- Level of detail
- Menu classification
- Menu Text Override (F0083)

Menu Filtering

OneWorld automatically filters menus based on your user ID so that only menu selections that apply to your job appear on your workstation. This feature allows you to maintain one set of menus (one database) with hundreds of menu selections, but you see only those menus that apply to your job.

Menus are filtered so the following do not appear:

- Menu selections that you or other users do not have authority to access. For example, Employee Information in Human Resources Management.
- Menu selections that are country-code-specific. If your user profile country code matches the country code for that menu, then the selection displays. For example, menu selections that pertain only to Canadian users, such as Canadian tax-related selections, appear only to Canadian users.
- WorldVision menu selections that are not installed on your workstation. For example, if WorldVision (the J.D. Edwards AS/400 product) is not installed on your workstation, that selection does not appear on the menu. You can distinguish a WorldVision menu selection from a OneWorld menu selection by looking at the Job to Execute number. A WorldVision Job to Execute number begins with a J, for example, J3413.

Menu Design Tables

Menu Design stores information in the following tables:

Menu Master (F0082)	Defines all menus but not the selections.
Menu Selection (F00821)	Contains the type of selection to be executed, selection consequences, and version information.
Menu Text Override (F0083)	Contains menu selection descriptions.
Menu Path (F0084)	Contains the menu selection icons.

Working with Menus

Menus are the entry point to J.D. Edwards applications and reports. To access an application or report from a menu, the application or report must be attached to a menu selection on the menu.

This chapter describes the following:

- Defining a new menu
- Reviewing selections for a menu
- Printing a menu report

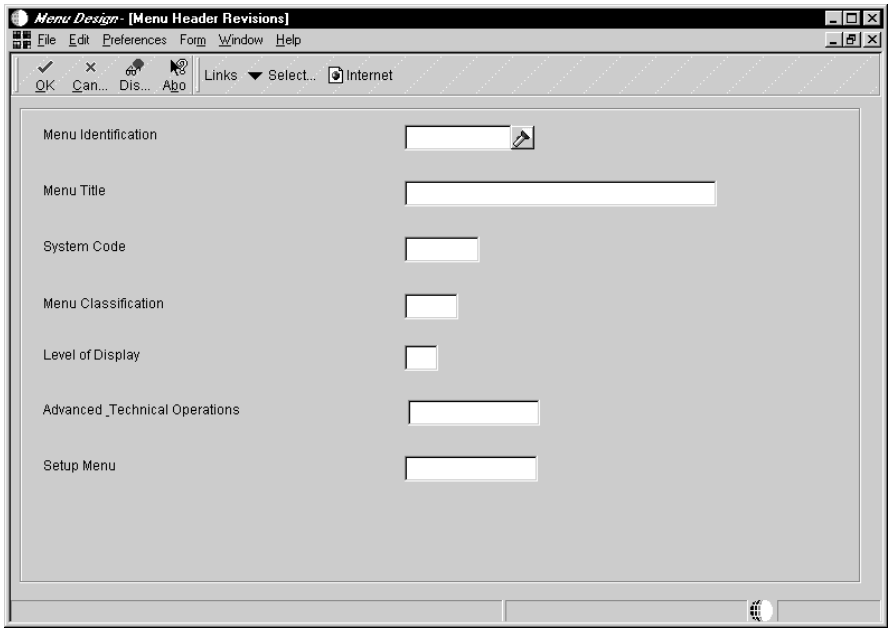
Defining a New Menu

You can define a menu to include selections that enable you to access the applications and reports you need from one location.

▶ To define a new menu

On System Administration Tools (GH9011)

1. Choose Menu Design (P0082).
2. On Work With Menus, click Add**.**



3. On Menu Header Revisions, complete the following fields:
   - Menu Identification
   - Menu Title
   - System Code
   - Menu Classification
   - Level of Display
   - Advanced & Technical Operations
   - Setup Menu

Field	Explanation
Menu Identification	The menu name, which can include up to nine characters. J.D. Edwards standards are: • Menu numbers are preceded with a G prefix. • The two characters following the prefix are the system code. • The next characters further identify the menu. • The 4th character specifies a specific skill level. • The 5th character distinguishes two menus of the same system with the same skill level. For example, the menu identification G0911 specifies the following: G Prefix 09 System code 1 Display level/skill level 1 First menu

Field	Explanation
Menu Title	Contains menu titles and menu selection descriptions.
Menu Classification	The menu classification indicates the type of a menu. For example: a JDE Master menu or Company Master menu.
Level of Display	The Level of Display field contains a number or letter identifying the level at which menus and processing options are displayed. The levels of display are as follows: A Product Groups (for example, Job Cost, Manufacturing) B Major Products (for example, GL, AP) 1 Basic Operations 2 Intermediate Operations 3 Advanced Operations 4 Computer Operations 5 Programmers 6 Senior Programmers.
Advanced & Technical Operations	For World: The advanced operations key is used to direct the menu selection '27' (Advanced Operations) to the appropriate menu. This menu designation must be preceded with an asterisk (*). For example, the General Accounting Advanced Operations menu would be *A093. . . . . . *Form-specific information* . . . . . . For OneWorld: Each menu may optionally have an Advanced & Technical Operations menu to which it is associated and it is displayed as the last menu selection. This is normally used to categorize more advanced tasks.
Setup Menu	For World: The technical operations control key is used to direct the menu selection '29' (Technical Operations) to the appropriate menu. This menu designation must be preceded with an asterisk (*). For example, the General Accounting Technical Operations Menu would be *A094. . . . . . *Form-specific information* . . . . . . For OneWorld: The menu you enter in this field is associated with the menu item description: Setup Menu. This menu is automatically displayed at the bottom of the menu you specify in the Menu Identification field.

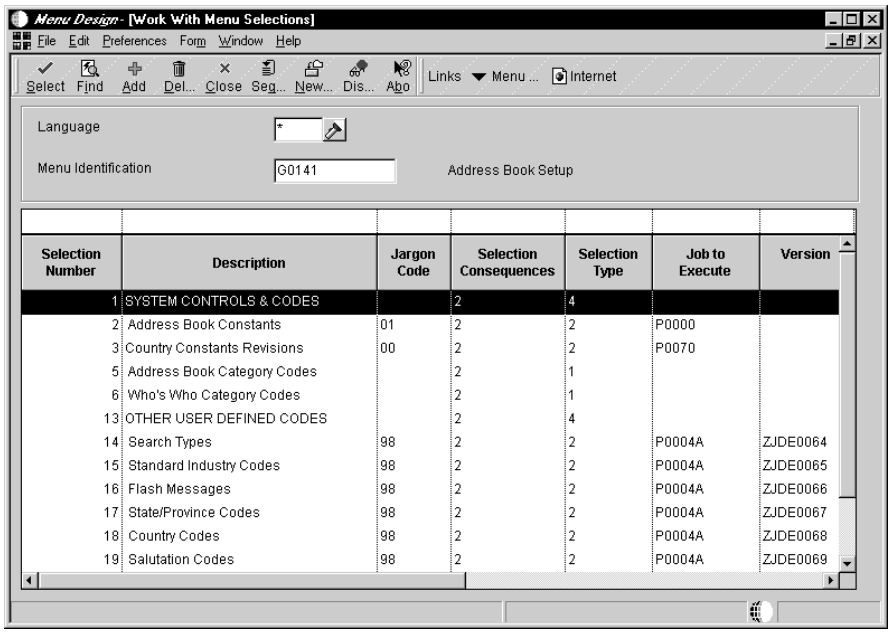
Reviewing Selections for a Menu

You can review the selections included on a specific menu. You can accomplish this from the Work With Menus form or from the Menu Header Revisions form.

▶ To review selections for a menu

On System Administration Tools (GH9011)

1. Choose Menu Design (P0082).
2. On Work With Menus, locate and choose a menu you wish to review.



If you are working with the Menu Header Revisions form, choose Selections from the Form menu.

In either instance, the Work With Menu Selections form appears, from which you can view and edit selections for a specific menu.

Printing a Menu Report

You can print a report that lists the menus. You can print menus only or menus and menu selections. To print a menu report, choose Menu Print from the Form menu.

Example: Print Menus Report

R0082P J.D. Edwards & Company 4/25/97 13:33:23
Print Menus Page – 1

Menu	Description	SY	LOD	Class
DEMO	APPLICATIONS	00		

Sel #	Description	Job To Execute	Form Name	Version	Ctry	Appl Ovrd	Run Time Message	SC
0	APPLICATIONS							1
1	Journal Entries	P0911		ZJDE0001				3
2	General Journal Posting	R09801						3
3	Company Numbers and Names	P0010						3
4	Budget vs.Actual Comparison	P09210		ZJDE0001				2
5	Account Ledger Inquiry	P0911L						1
6	Original Budget Update	P14102		ZJDE0001				3
7	Online Consolidations	P09218		ZJDE0001				1
8	Manufacturing Variance Inquiry	P3102		ZJDE0001				1
11	Bill of Material	P3002		ZJDE0001				1
12	Routings	P3003		ZJDE0001				1
13	Forecasting	P3460		ZJDE0001				1
14	Customer Service	P4210		ZJDE0001				1
15	Planners Workbench	P3401		ZJDE0003				1
16	Schedulers Workbench	P31225		ZJDE0001				1

Working with Menu Selections

Work With Menu Selections displays available selections for the selected menu. Use Work With Menu Selections when you are:

- ❑ Adding or changing a menu selection
- ❑ Adding an application to a menu
- ❑ Adding or changing Web addresses on OneWorld Explorer Help
- ❑ Creating a Web view subheading on a menu
- ❑ Linking menus
- ❑ Creating fast path selections

Adding or Changing a Menu Selection

To add a menu selection for an application or report, you must first name the menu selection by assigning a description and unique selection number before you can add a menu selection for an application or report to the menu.

After naming a menu selection, indicate the selection type and define it.

- Selection Type specifies the type of program that is executed for the menu selection. You use OneWorld Application, OneWorld Report, World Vision, or Windows Application to execute a specific application, report, or program.
- The Subheading selection type does not perform an action. You use it to logically group menu selections on the menu. Subheading selections appear on the menu only in Web view.

Note: When you delete a menu, you delete the menu and any menu selections available on that menu. The applications called by the menu selections are not deleted, and you can access these applications from other menus.

- You use the Menu selection type to call another menu.

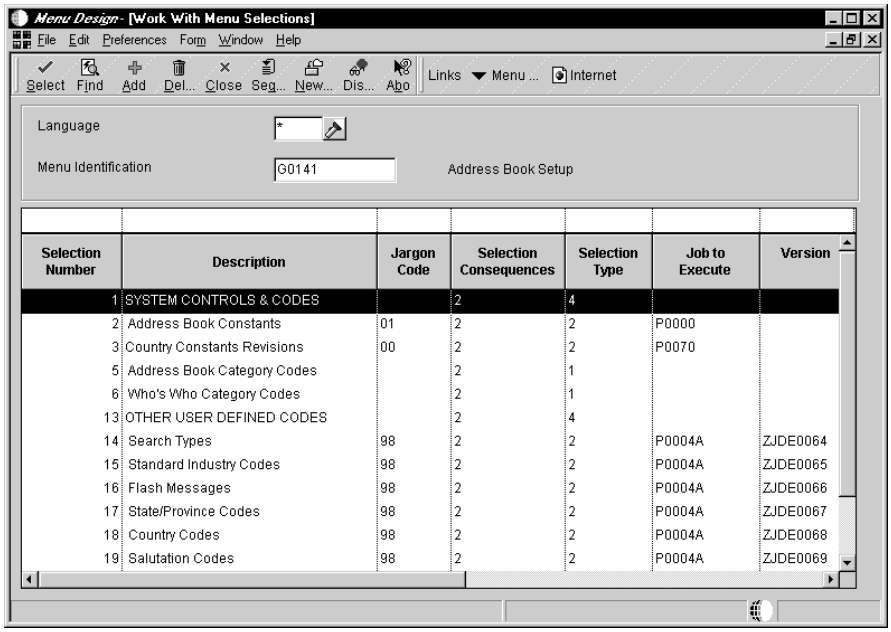
Adding or changing a menu selection consists of the following:

- Naming the menu selection
- Defining the menu selection

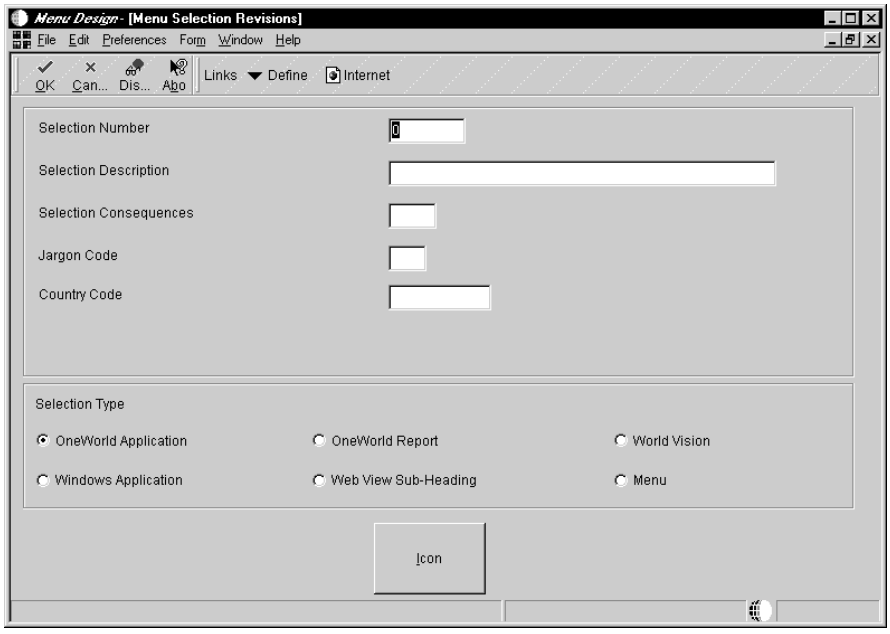
▶ To name a menu selection

On System Administration Tools (GH9011)

1. Choose Menu Design (P0082).
2. On Work With Menus, find and choose the menu that has a selection you wish to name.



3. On Work With Menu Selections, click Add or choose an existing selection to change.



4. On Menu Selection Revisions, complete the following required fields:
   - Selection Number
   - Selection Description
   - Selection Consequences

   If you chose an existing selection to change, the Selection Number field is disabled.

5. The following fields are optional, complete if necessary:
   - Jargon
   - Country Code

   If the base language is a double-byte language, a Search Description field is shown below the Country Code field. Enter the single-byte search description to be used by Menu Word Search. Menu Word Search uses only single-byte search descriptions.

Field	Explanation
Selection Number	Used to determine the order of menu items and allow them to be selected by this number.
Selection Description	Contains menu titles and menu selection descriptions.
Selection Consequences	The selection consequences tell the user what the consequences are in taking that specific menu selection.

Field	Explanation
Jargon	A code used to designate the reporting system number for entering specific text or “jargon”. See User Defined Codes, system code ’98’, record type ’SY’ for a list of valid values.
Country Code	The Menu Country/Region Codes field contains the region code (3 bytes) for all 24 menu selections for each menu record. This region code is used to mask those international selections that are country specific; i.e. 1099 processing in the US and VAT tax processing in Europe.

▶ To define the menu selection

On System Administration Tools (GH9011)

1. Choose Menu Design (P0082).
2. Find and choose the menu on which you wish to define a menu selection.
3. On Work With Menu Selections, choose an existing selection to define.
4. On Menu Selection Revisions, indicate one of the following Selection Types:
   - OneWorld Application
   - OneWorld Report
   - World Vision
   - Windows Application
   - Sub-Heading
   - Menu
5. From the Form menu, choose Define.

   A form that is specific to the selection type appears.
6. Define options for the selection type indicated.
7. Click OK.

Adding an Application to a Menu

You can add OneWorld applications and reports, WorldVision applications, and Windows applications to a menu. You can also link a menu to another menu.

Complete the following tasks:

- Add a OneWorld application to a menu
- Add a OneWorld report selection to a menu

- Add a WorldVision application to a menu
- Add a Windows application to a menu

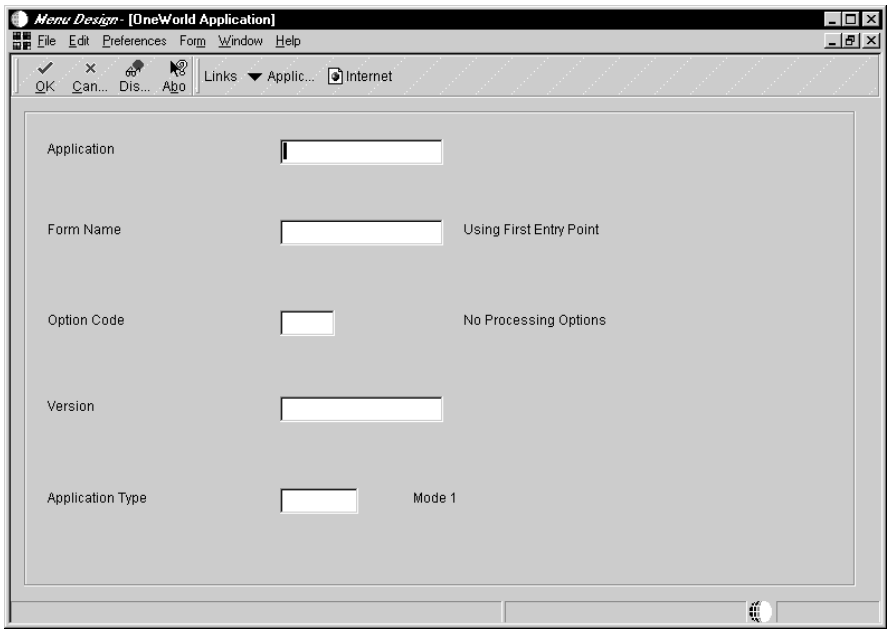
▶ To add a OneWorld application to a menu

You can use this procedure to add a OneWorld application created in Forms Design as a menu selection.

On System Administration Tools (GH9011)

1. Choose Menu Design (P0082).
2. Find and choose the menu on which you wish to add a OneWorld application.
3. On Work With Menu Selections, click Add.
4. On Menu Selection Revisions, complete the following required fields:
   - Selection Number
   - Selection Description
   - Selection Consequences

   If you chose an existing selection to change, the Selection Number field is disabled.
5. Click the OneWorld Application option, and from the Form menu, choose Define.



6. On OneWorld Application, complete the following fields:

   - Application
   - Form Name

     You can use this field to define a specific entry point for the OneWorld application. If you leave this blank, the program's first entry point is used.

   - Option Code
   - Version
   - Application Type

7. From the form menu, choose Application to view and choose from a list of available applications. Likewise, from the Form menu you can choose Versions to search for available versions.

Field	Explanation
Option Code	For World, this code specifies the function of a menu selection using the DREAM Writer when F18 is pressed. F18 may be locked out by simply replacing code 1 with 3 or code 2 with 4. This code, in conjunction with the version number and the option key, provide the following functions: Code 1 version — mandatory; option key — form i.d. F18 displays processing options. Selection = blind DREAM Writer execution. 2 version — blank; option key — form i.d. F18 displays DREAM Writer versions list. Selection = DREAM Writer versions list. 2 version — not blank; option key — form i.d. F18 displays DREAM Writer versions list. Selection = blind execution, batch. Review the HELP instructions for Menu Information (Menu Locks) (P0090) for a detailed explanation of codes related to job submission and control. For OneWorld, this code specifies whether the user will be prompted for additional information prior to running the application. Available values are: 0 No processing options 1 Blind execution (no prompt) 2 Prompt for version 3 Prompt for values
Version	Version identifies a specific set of data selection and sequencing settings for the application. Versions may be named using any combination of alpha and numeric characters. Versions that begin with 'XJDE' or 'ZJDE' are set up by J.D. Edwards.

Field	Explanation
Application Type	Complete with a user–defined, alphanumeric value. This field exists in the JDE user profile and within each menu and menu selection record. When security is active, the value of this field in the user profile is compared with the value in the corresponding menu lock. The values must be equal in the user profile and menu lock to access the menu. A blank in this field in the user profile gives the user all authority. A blank in this field in the menu record indicates no security exists on this menu.

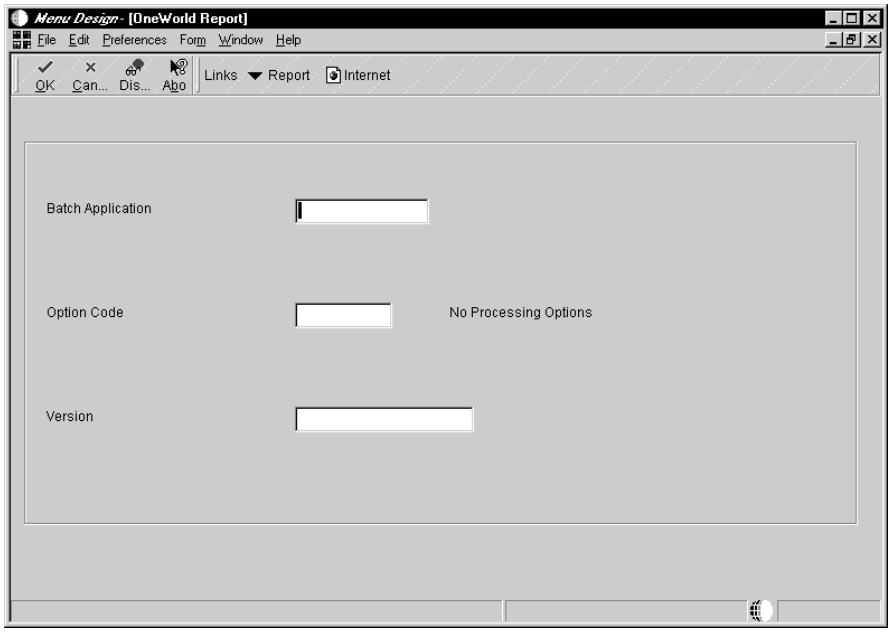
▶ To add a OneWorld report selection to a menu

You can add a report created in the OneWorld Report Design tool as a menu selection.

On System Administration Tools (GH9011)

1. Choose Menu Design (P0082).
2. Find and choose the menu on which you wish to add a OneWorld report selection.
3. On Work With Menu Selections, click Add.
4. On Menu Selection Revisions, complete the following required fields:
   - Selection Number
   - Selection Description
   - Selection Consequences

   If you chose an existing selection to change, the Selection Number field is disabled.
5. Click the OneWorld Report option, and from the Form menu, choose Define.



6. On OneWorld Report, complete the following fields:
   - Batch Application
   - Option Code
   - Version

   Use Form menu options to view and choose from a list of reports and versions.

▶ To add a WorldVision application to a menu

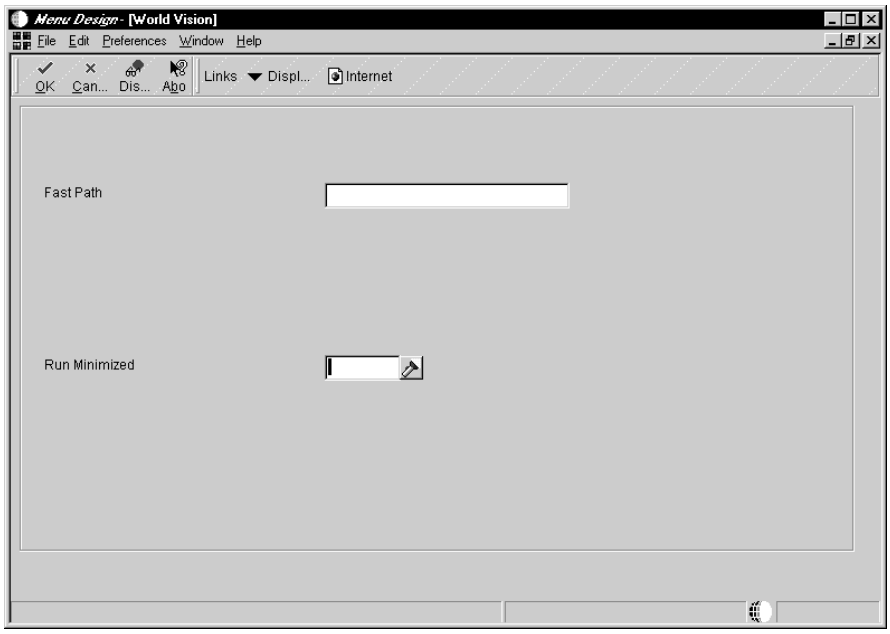
You can add a WorldVision application as a menu selection.

On System Administration Tools (GH9011)

1. Choose Menu Design (P0082).
2. Find and choose the menu on which you wish to add a WorldVision selection.
3. On Work With Menu Selections, click Add.
4. On Menu Selection Revisions, complete the following required fields:
   - Selection Number
   - Selection Description
   - Selection Consequences

   If you chose an existing selection to change, the Selection Number field is disabled.

5. Click the WorldVision option, and from the Form menu, choose Define.



6. On WorldVision, complete the following fields:
   - Fast Path
   - Run Minimized

Field	Explanation
Fast Path	The path field contains the path used for client based menus. The path describes where the application is located on your computer or network. A path includes the drive, folders, and subfolders that contain the application to be executed. To specify the path for a World Vision menu selection, the path includes the selection number, slash, menu.
Run Minimized	The Run Minimized flag determines if a Windows application is to be minimized to an icon when you open it.

▶ To add a Windows application to a menu

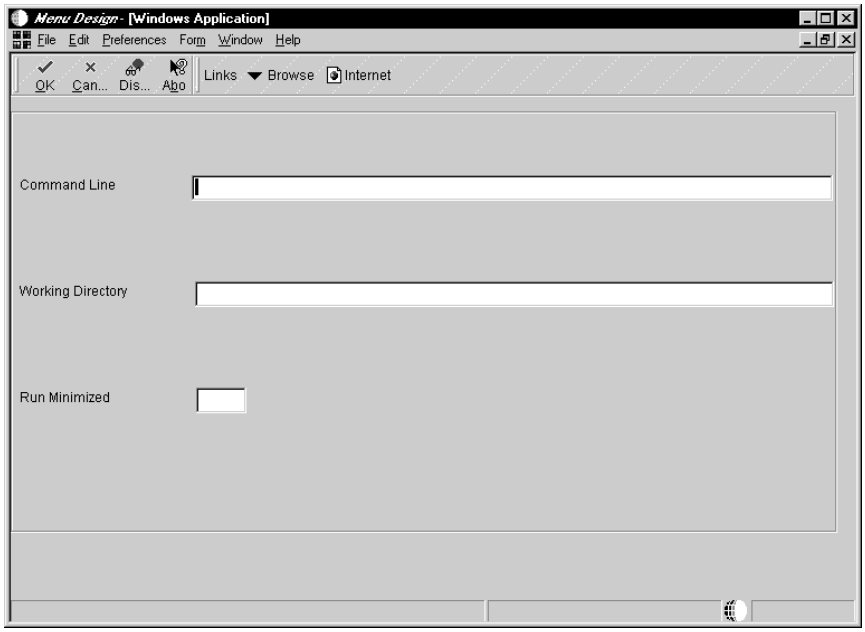
You can add any Windows applications as menu selections. For example, you can add the Windows programs such as Calendar, Clock, Note Pad, or Write.

On System Administration Tools (GH9011)

1. Choose Menu Design (P0082).
2. Find and choose the menu on which you wish to add a Windows application.

3. On Work With Menu Selections, click Add.
4. On Menu Selection Revisions, complete the following required fields:
   - Selection Number
   - Selection Description
   - Selection Consequences

   If you chose an existing selection to change, the Selection Number field is disabled.
5. Click the Windows Application option, and from the Form menu, choose Define.



6. On Windows Application, complete the following fields, or click the Browse button to search for the Windows application:
   - Command Line

     Enter the executable of the Windows application you want to add to the OneWorld menu, such as winword.exe.
   - Working Directory

     Enter the path on your local machine where the Windows application resides.
   - Run Minimized

Field	Explanation
Command Line	The path field contains the path used for client based menus. The path describes where the application is located on your computer or network. A path includes the drive, folders, and subfolders that contain the application to be executed. To specify the path for a World Vision menu selection, the path includes the selection number, slash, menu.
Working Directory	The path field contains the path used for client based menus. The path describes where the application is located on your computer or network. A path includes the drive, folders, and subfolders that contain the application to be executed. To specify the path for a World Vision menu selection, the path includes the selection number, slash, menu.

Adding or Changing Web Addresses on OneWorld Explorer Help

You can add Web addresses or change some of the addresses that appear on the Help menu of OneWorld Explorer. From the Help menu on OneWorld Explorer, there is an option called “J.D. Edwards on the Web.” From this option, a list of Web addresses appear, and there is a line that separates the addresses. The addresses above this line, which include J.D. Edwards Home Page and Contact Us, are hard-coded into OneWorld, which means you cannot change them. You can, however, change the Web addresses below the line, or add your own Web addresses to the list.

▶ To add or change Web addresses on OneWorld Explorer Help

On System Administration Tools (GH9011)

1. Choose Menu Design (P0082).
2. On Work With Menus, in the Menu ID query-by-example field, type HELP, then click Find.

   The Web Access menu appears.
3. Choose the Web Access menu and click Select.
4. On Work With Menu Selections, click Add to add a new Web address, or choose a row and click Select to change an existing Web address.
5. On Menu Selection Revisions, complete the following required fields:
   - Selection Number
   - Selection Description
   - Selection Consequences

If you chose an existing selection to change, the Selection Number field is disabled.

6. Click the Windows Application option, and from the Form menu, choose Define.
7. On Windows Application, complete the following fields:
   - Command Line

     Enter the Web address that you want to add to the Help menu, such as http://www.jdedwards.com.

   - Working Directory

     Because you are entering a Web address, you do not need to complete this field.

   - Run Minimized

Creating a Web View Subheading on a Menu

Use Web subheadings to logically group menu selections on the menu. Subheadings appear on the menu in Web view only and do not perform an action.

▶ To create a Web view subheading selection

On System Administration Tools (GH9011)

1. Choose Menu Design (P0082).
2. Find and choose the menu on which you wish to create a subheading.
3. On Work With Menu Selections, click Add.
4. On Menu Selection Revisions, complete the following field:
   - Selection Number
5. Click the Web View Sub-Heading option.
6. Click OK to complete the Web view subheading assignment.

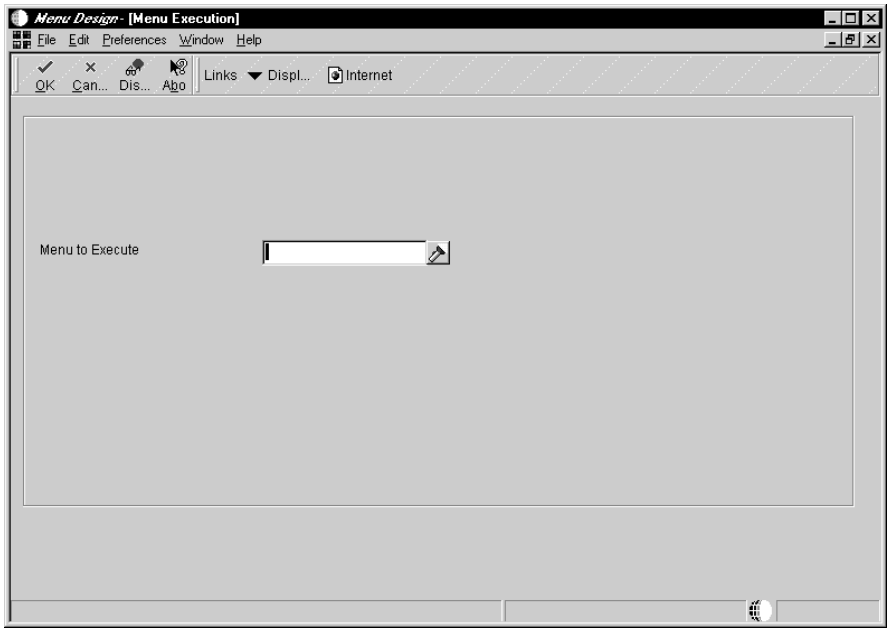
Linking Menus

You can add a menu selection that displays another menu.

▶ To link another menu to a menu

On System Administration Tools (GH9011)

1. Choose Menu Design (P0082).
2. Find and choose the menu on which you wish to add a selection.
3. On Work With Menu Selections, click Add.
4. On Menu Selection Revisions, click the Menu option.
5. From the Form menu, choose Define.



6. On Menu Execution, complete the following field or click the visual assist to search for menus:
   - Menu to Execute

Field	Explanation
Menu to Execute	The specific menu to be executed as a selection on a menu.

Creating Fast Path Selections

You can quickly move among menus and applications by using fast path commands. A fast path command is:

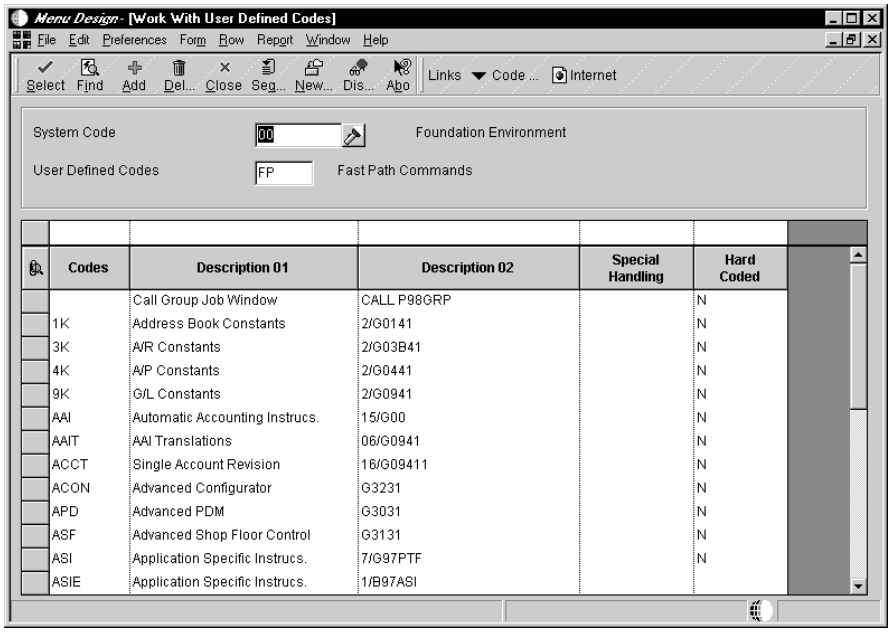
- An abbreviation that is either shipped with J.D. Edwards demo data or which you define to suit your business environment. For example, the fast path OL takes you to the application Object Librarian so you can work with OneWorld objects.
- A combination of menu selection and menu number. For example 2/G01 (menu selection number 2 on menu number G01) takes you to Work With Addresses in Address Book. As you become more familiar with OneWorld menu abbreviations, you might find fast path a quicker way to navigate to an application.

You can set up your own a fast path abbreviations to access frequently-used applications.

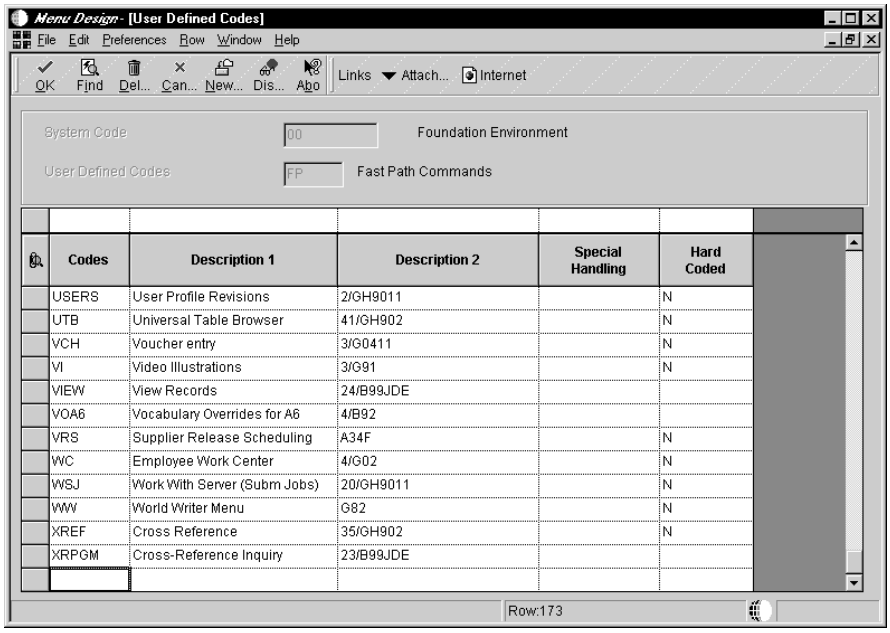
▶ To create a fast path selection

On System Administration Tools (GH9011)

1. Choose Menu Design (P0082).
2. On Work With Menus, find and choose the menu that has a selection for which you wish to create a fast path.
3. On Work With Menu Selections, from the Form menu, choose Fast Path Revs.



4. On Work With User Defined Codes, click Add.



5. On User Defined Codes, click inside the grid, then press the Ctrl and End keys to display the bottom of the grid.
6. To add a user defined code for a new fast path, complete the following required fields in the last row of the grid:

   - Codes
   - Description 1
   - Description 2

   You enter the abbreviation for the fast path in the Code field. Enter the description of the abbreviation, such as the name of the menu selection, in the Description 01 field. Enter the selection number and menu number in the Description 02 field.

To determine the selection number for the fast path you created (for example, selection number 2 on menu G01), use Work With Menu Selections. Do not count the menu selections in OneWorld Explorer because the menu might be filtered.

Working with Menu Selection Revisions

You can revise your existing menu selections to change menu text for languages, change menu selection text, or renumber a menu selection. This chapter describes the following tasks:

- ❑ Copying a menu selection
- ❑ Changing menu text for languages
- ❑ Changing menu selection text
- ❑ Renumbering a menu selection

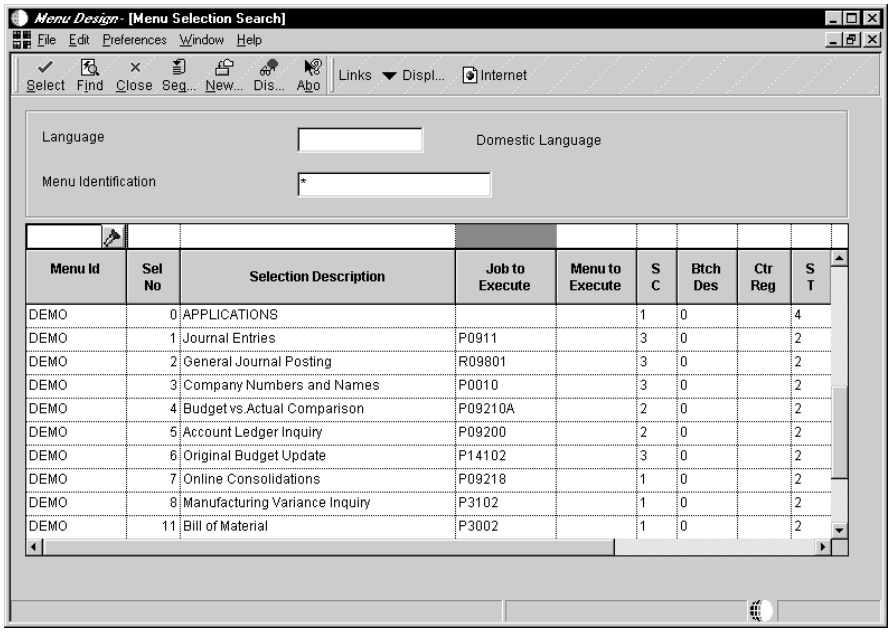
Copying a Menu Selection

You can copy an existing menu selection and attach it to another menu.

▶ To copy a menu selection

On System Administration Tools (GH9011)

1. Choose Menu Design (P0082).
2. On Work With Menus, find and choose the menu that has a selection you wish to copy.
3. On Work With Menu Selections, click Add.
4. On Menu Selection Revisions, enter the new selection number and choose Copy from the Form menu.



5. On Menu Selection Search, choose an existing menu selection.

   The selection description, consequences, and type are inserted into the newly added menu selection.

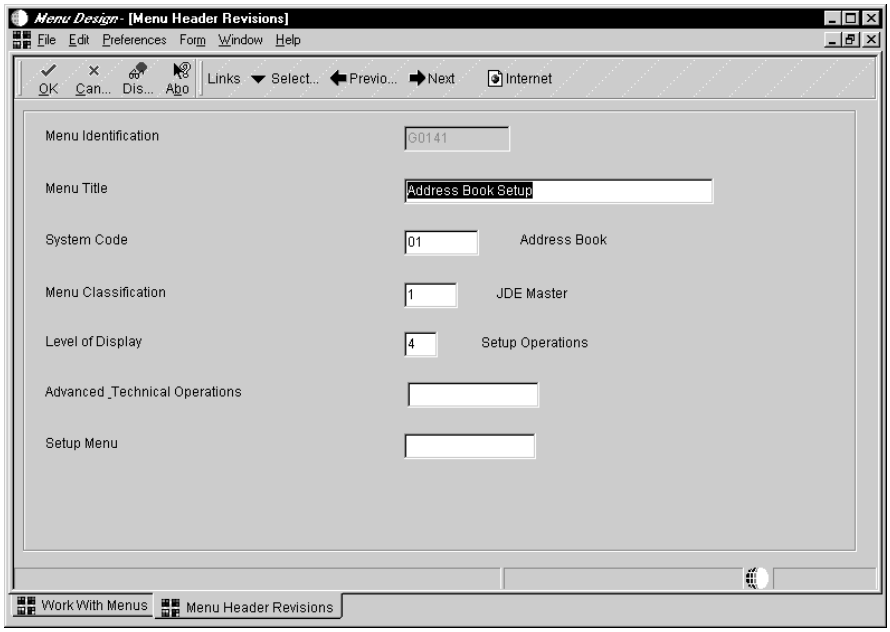
Changing Menu Text for Languages

You use Title Overrides to change the language and description of a menu selection.

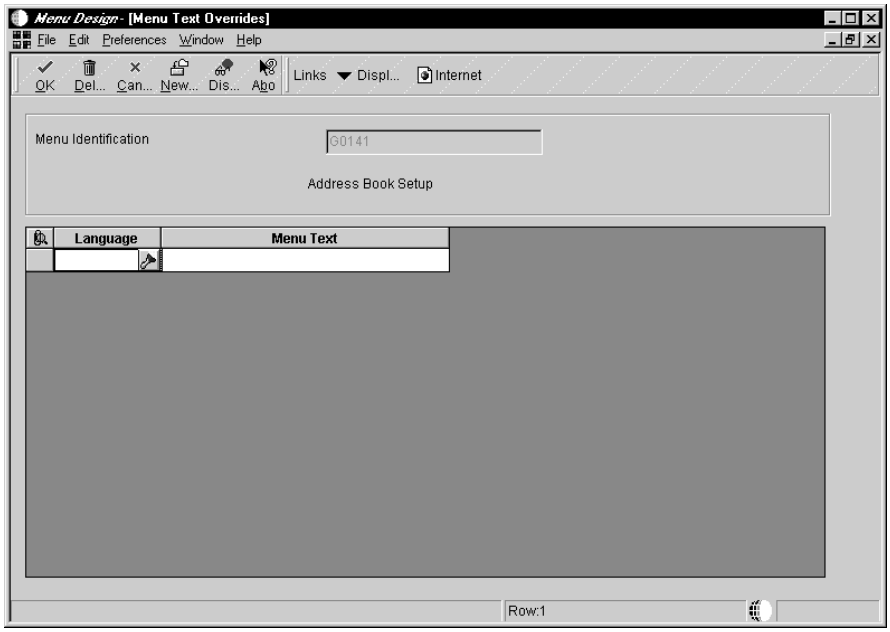
▶ To change menu text for languages

On System Administration Tools (GH9011)

1. Choose Menu Design (P0082).
2. On Work With Menus, find and choose the menu for which you wish to change menu text.
3. On Work With Menus, choose Header from the Row menu selection.



4. On Menu Header Revisions, from the Form menu, choose Title Overrides.



5. On Menu Text Overrides, enter the desired language code (such as S for Spanish) and text description (such as the Spanish description of the menu selection) in the following fields and click OK.

   - Language
   - Menu Text

Field	Explanation
Language	A user defined code (system 01/type LP) that specifies a language to use in forms and printed reports. Before any translations can become effective, a language code must exist at either the system level or in your user preferences.
Menu Text	Contains menu titles and menu selection descriptions.

Changing Menu Selection Text

You use Text Overrides to change a menu selection's text.

▶ To change menu selection text

On System Administration Tools (GH9011)

1. Choose Menu Design (P0082).
2. On Work With Menus, find and choose the menu for which you wish to change menu's selection text.
3. On Work With Menu Selections, choose the menu selection you want to change and click Select.
4. On Menu Selection Revisions, choose Text Overrides from the Form menu.
5. On Menu Text Overrides, complete the following fields and click OK:
   - Language
   - Menu Text

   Changes in menu text are not displayed until the changed menu is closed and reopened. You can also choose Refresh from the View menu to update the menu text.

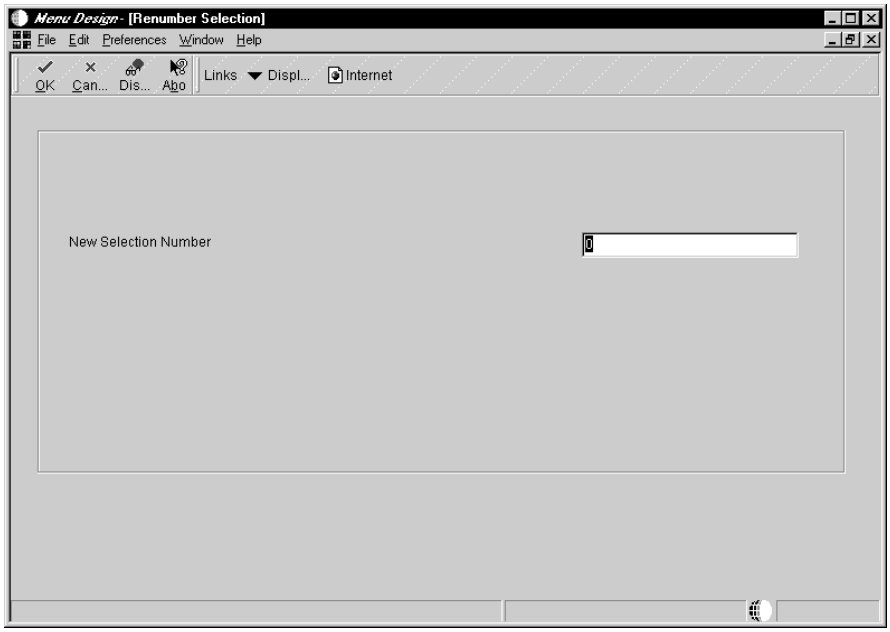
Renumbering a Menu Selection

You can renumber menu selections from both Work With Menu Selections and Menu Selection Revisions. You can edit each selection to change the sequence of selections on a menu. You cannot rearrange menu selections by clicking and dragging them.

To renumber a menu selection

On System Administration Tools (GH9011)

1. Choose Menu Design (P0082).
2. On Work With Menus, find and choose the menu for which you wish to change menu's selection number.
3. On Work With Menu Selections, choose the menu selection you want to change and click Select.
4. On Menu Selection Revisions, from the Form menu, choose Renumber.



5. Complete the following field, and click OK.
   - New Selection Number

Field	Explanation
New Selection Number	Used to determine the order of menu items and allow them to be selected by this number.

User Profiles

A user profile defines a specific user or group of users to OneWorld. Profiles define such information as the group to which a user belongs, a list of environments that a user or group can select when signing onto OneWorld, and the language preference of the user or group.

This section defines the concepts of user profiles and how to use the User Profiles (P0092) application.

This section contains the following:

- Understanding user profiles
- Adding new users
- Setting up user profiles

Understanding User Profiles

You can use the User Profiles application (P0092) to define specific users or groups to OneWorld. This description includes:

- The group to which a user belongs, such as an accounts payable clerk as part of the AP group. Groups are an important aspect of OneWorld. By assigning users to groups, system administrators can set user preferences and securities based on the groups rather than the individual user.
- The environments that the user can select when starting OneWorld.
- The language preference and country code for the text that displays on OneWorld's menus, forms, and country-specific applications.

How Group Profiles Make Profiling Easier

Group preferences eliminate the need to set up preferences to each individual user profile. By assigning individual users to a group, you can perform assignments once for the group and have those settings available to all of the individual users belonging to that group. Of course, you can also specify different preferences for each user. The individual user settings override the group settings, but if no user profile information exists, OneWorld takes the information from the group profile. OneWorld uses groups for the following purposes:

- Environments
- User overrides
- Application security
- Creation of signon security records

Tables Used by the User Profiles Application

The User Profiles application (P0092) uses the following tables:

- Library Lists - User (F0092)
- User Display Preferences (F00921)
- User Display Preferences Tag File (F00922)
- User Access Definition (F00925)

- Library List Control (F0093)
- Library List Master File (F0094)

Adding New Users

You can create user profiles one at a time using the User Profiles interactive application (P0092), or you can create multiple profiles at a time using batch processes. When you are ready to create user profiles for the first time, you might need to create hundreds of profiles. In this case, use the batch processes to create the profiles. But if you only need to add a few users, use the User Profiles application.

This topic is a checklist for all the steps needed to add a new user. These steps do not include installing OneWorld on a workstation or third-party set-up issues, such as network user IDs.

This topic contains the following:

- ❑ Adding an individual user
- ❑ Adding multiple users

Adding an Individual User

The following procedure describes the steps you need to perform when adding user profiles one at a time.

▶ To add an individual user (checklist)

1. If you plan to create a new group for the user you are adding, add an Address Book record with a valid search type code (for example, E for employee). See *Entering Address Book Records* in the *Address Book Guide.*
2. If the existing group profiles are not acceptable for this new user, add a group profile using the User Profiles application (P0092). See *Creating User and Group Profiles* in this section for information.
3. Add an Address Book record for the new user. See *Entering Address Book Records* in the *Address Book Guide.*
4. Add a user profile and assign the user to a group. Determine if the user needs any overrides to the group setup, for example, whether the user needs fast path enabled. See *Creating User and Group Profiles* in this section for information.
5. Add signon security records for this user. See *Working with Signon Security* in this guide for information.

6. Add any security workbench overrides for the user, if the user needs different security than the group. See *Working with Security Workbench* in this guide for information.
7. Populate the machine table for the user's machine using the Machine Identification application (P9654A). See *Defining Machines* in the *Package Management Guide* for information.
8. Add the user's machine as a subscriber for all publisher tables that should be replicated to the user's workstation. See *Setting Up Data Replication* in this guide for information.
9. Add any new user overrides for the user, if the user needs different user overrides than the group. See *Working with User Overrides* in this guide for information.

Adding Multiple Users

The following is a checklist of the steps you need to perform when you use the batch process to add multiple user profiles at one time. This batch process automates the process of user profile creation.

▶ To add multiple users (checklist)

If you have the processing option for user profiles set to validate address book numbers, you should begin at Step 1. See *Understanding Processing Options for User Profiles* in this section for additional information. If you do not have the processing option enabled, begin at Step 2.

1. Using the Address Book Revisions application (P010102), create address book records for groups that you will use in user profiles.
2. Using the User Profiles application (P0092), add the group profiles. See *Creating User and Group Profiles* in this section for information.
3. Populate the various Address Book tables. If you are migrating data from a non-OneWorld system, you can populate the data tables with a table conversion. Otherwise, you can manually add data to the Address Book tables.
4. Run the User Profile Creation (R0092) batch process to create user profile records from existing Address Book records. Normally this report will be based upon Address Book records with a search type for employees (E). You have the option of picking one default group to put everyone in, or you can run the report more than once to put people in different groups. See *Creating Profiles Using a Batch Process* in this section for information.
5. Adjust each user's group assignments. Determine what group you want an individual placed in and manually assign each user to a group. Change the user's environments if they are not standard to that group. See *Setting Up User Profiles* in this section for information.

The following settings are dictated by group:

- Environments
- User Overrides
- Application Security

When deciding which group a user should reside in, consider application security as the most important group. This is because application security has the most extensive setup, and managing overrides to the group security will be more difficult than, for example, managing overrides to deployment preferences.

Note: Signon security is not based on groups, because individuals must have their own OneWorld passwords. There is a facility with signon security to quickly create individual security records by group, but after the records are created, security is by individual. See *Working with Signon Security* in this guide for information.

6. Run the User Profiles Summarization (R00921) batch process to view your new user profiles. See *Summarizing Group Profiles* in this section for information.
7. Create Security Workbench records for all groups and any individual overrides to those groups. See *Working with Security Workbench* in this guide for information.
8. Create signon security records. You can create signon security records for all individuals within a group by entering one record for the group. See *Working with Signon Security* in this guide for information.
9. Manually populate the Machine/Group Identification table (F00960). This table is automatically populated each time a machine signs onto OneWorld. However, if you intend to use schedule packages or set the machine up for data replication before users have signed onto OneWorld, you must manually populate this table.
10. Create one replication publisher record for each table you plan to replicate. Run the Create Publisher and Subscriber Records (R00960) batch process to populate the subscribers for that publisher with the information from the machine table. You only need to run this report to create one set of subscriber records, because replication has a copy button that you should use to create additional publisher/subscriber records. See *Setting Up Data Replication* in this guide for information.
11. Create user overrides for groups. Normally you will not create any overrides for individuals, because they can easily create their own as they use the software. See *Working with User Overrides* in this guide for information.

Setting Up User Profiles

This topic describes the different ways to use the User Profile application (P0092) to set up user profiles. When setting up profiles, the system administrator should perform the following task.

▶ To set up user profiles (overview)

1. Create all of the group profiles for the enterprise.
2. Create a user profile for every user.

   Optionally, assign that user to a group profile.

3. Assign to each group or user the following preferences:
   - Environments, to determine the environments you want available to each group or user.
   - Display preferences, to determine OneWorld display characteristics such as language, date format, and country code. The Display Preferences are controlled on the main User Profile Revisions form.

Caution: If you are setting up user profiles during the installation process, you *must* sign onto your deployment server using the deployment environment. After you have completed the installation process, you can add or modify user profiles from any machine *except* the deployment server.

This topic contains the following:

- ❑ Creating and modifying user and group profiles
- ❑ Copying user and group profiles
- ❑ Assigning environments to user or group profiles
- ❑ Assigning business preferences to user or group profiles
- ❑ Understanding processing options for user profiles
- ❑ Creating profiles using a batch process

Creating and Modifying User and Group Profiles

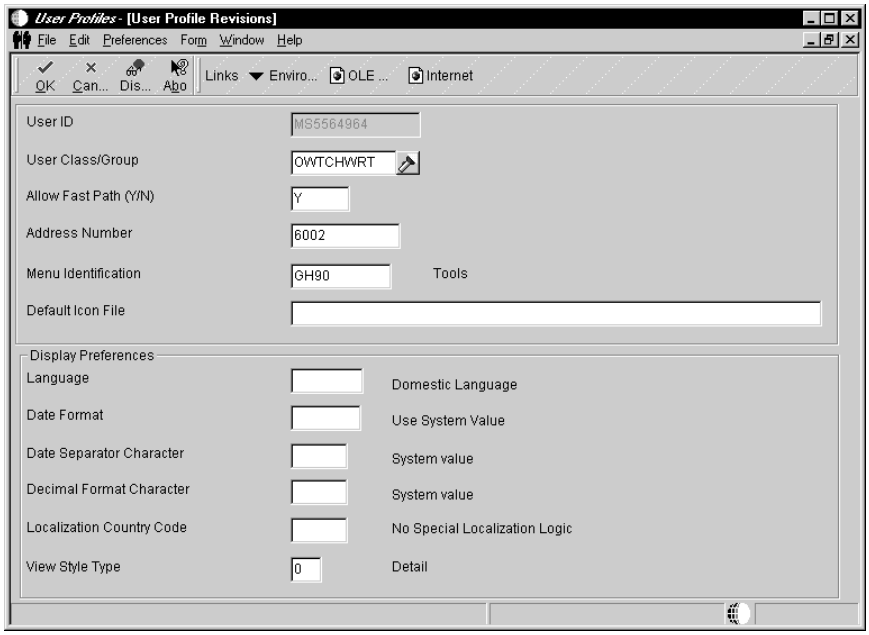
The system administrator needs to create a user profile for every user. The user profile defines certain setup and display features, such as access to fast path, language, date format, or country code. The administrator should first create all of the group profiles needed for the enterprise. This makes creating profiles easier, because instead of defining specific environments, packages, and machine configurations to each user, administrators can define them for the group. If an individual in a group needs a different setup, assign them different ones at the user level, which will override the group settings.

If you select a country code for a user or group, OneWorld's menu filtering process will display for that user or group any special menu selections unique to that country code. For example, if you entered CAN (Canada), that user or group would see on the appropriate menu the Canadian Tax Information application, which users without that country code would not see.

▶ To create and modify user and group profiles

On the System Administration Tools (GH9011) menu

1. Choose User Profiles (P0092).
2. On the Work With User Profiles form, do one of the following operations:
   - If you want to create a new profile, click Add.
   - If you want to modify an existing profile, click Find, choose a profile in the grid, then click Select.



3. On the User Profile Revisions form, in the header section, do one of the following to create an individual or group profile:
   - To create an individual profile, do the following:
     - In the User ID field, type the user ID for the individual.

       When you modify user or group profiles, this field displays the user ID or the name of the group. You cannot type new information in this field when you modify a profile.

     - In the User Class/Group field, type the group to which the individual belongs, such as ACCOUNTING. You can leave this field blank if the user does not belong to a group.
   - To create a group profile, do the following:
     - In the User ID field, put the group name, such as ACCOUNTING.
     - In the User Class/Group field, you must put the literal value: *GROUP

     Important: You can modify the group to which a user belongs by typing a new group name in this field; however, when you modify profile information for a group, you must use the *GROUP literal value. Do not modify this field for group profiles.

4. In the header area of the form, complete the remaining fields.
   - Allow Fast Path (Y/N)
   - Address Number
   - Menu Identification
   - Default Icon File
5. In the Display Preferences box, complete the following fields, then click OK.
   - Language
   - Date Format
   - Date Separator Character
   - Decimal Format Character
   - Country
   - View Style Type

Field	Explanation
User ID	For World, the IBM-defined user profile. For OneWorld, the identification code for a user profile.

Field	Explanation
User Class/Group	A profile used to classify users into groups for security purposes. Some rules for creating a User Class/Group are as follows: • The 'Class/Group' profile must begin with * so that it does not conflict with any System profiles. • The 'User Class/Group' field must be blank when entering a new group profile.
Allow Fast Path (Y/N)	The Fast Path flag is used to specify whether individual users may use the "Fast Path" method of processing within the J.D. Edwards menu program. This data field allows the values of blank, Y or N. blank user is allowed to use fast paths Y user is allowed to use fast paths N user is NOT allowed to use fast paths. . . . . . *Form-specific information* . . . . . If you choose No, OneWorld will not display the fast path bar. If you set this in conjunction with the Menu Identification field on this form, you will restrict the user to the initial menu and those menus called by the initial menu.
Address Number	A number that identifies an entry in the Address Book system. Use this number to identify employees, applicants, participants, customers, suppliers, tenants, and any other address book members.
Menu Identification	The menu name, which can include up to nine characters. J.D. Edwards standards are: • Menu numbers are preceded with a G prefix. • The two characters following the prefix are the system code. • The next characters further identify the menu. • The 4th character specifies a specific skill level. • The 5th character distinguishes two menus of the same system with the same skill level. For example, the menu identification G0911 specifies the following: G Prefix 09 System code 1 Display level/skill level 1 First menu
Default Icon File	The path field contains the path used for client based menus. The path describes where the application is located on your computer or network. A path includes the drive, folders, and subfolders that contain the application to be executed.

Field	Explanation
Language	A user defined code (01/LP) that specifies a language to use in forms and printed reports. Before any translations can become effective, a language code must exist at either the system level or in your user preferences.
Date Format	This is the format of a date as stored in the database. If you leave this value blank, the value will display according to the settings of the operating system on the workstation. With NT, the Regional Settings in the Control Panel control the settings for the operating system of the workstation.
Date Separator Character	The character entered in this field is used to separate the month, day, and year of a given date. NOTE: • If an asterisk is entered (*), a blank is used for the date separator. • If left blank, the system value is used for the date separator.
Decimal Format Character	The character entered in this field is used to signify the fractions from whole numbers (the positions to the left of the decimal). If left blank, the system value is used as the default.
Country	A user defined code (00/CN) that identifies a country. The country code has no effect on currency conversion. The Address Book system uses the country code for data selection and address formatting.
View Style Type	The view style determines how new menu information is displayed to the user in the OneWorld Explorer.

Copying User and Group Profiles

You can copy all or part of a user profile. In copying an entire user or group profile (display and environment preferences), you are creating a new user profile with the information from another profile. In copying part of a user profile, you are copying the environment preferences to an already existing user profile from another profile.

▶ To copy user or group profiles

On the System Administration Tools menu (GH9011)

1. Choose User Profiles (P0092).
2. On the Work With User Profiles form, find, then choose a user profile, and do one of the following:

- To copy an entire profile (the display, environment, and deployment preferences), click Copy. The User Profile Revisions form appears. This creates a new profile, so the user profile you create cannot already exist in OneWorld.

  See *Creating User and Group Profiles* in this section for more information.

- To copy environment preferences, from the Row menu, choose Copy Environment. The User Environment Revisions form appears. This copies environment preferences from one user profile to another. The user profile you copy to must already exist.

  See *Assigning Environments for User and Group Profiles* in this section for more information.

3. In the User ID field, enter a user ID or group name to copy the profile into, and change any other information. Click OK when you are finished.

Assigning Environments to User and Group Profiles

You can assign a list of environments that each group or user can choose from when starting OneWorld. Each time users start OneWorld they can choose from the environments assigned for their group if they do not have a user profile-specific environment assignment. You can assign one or more environments to choose from.

▶ To assign or delete environments

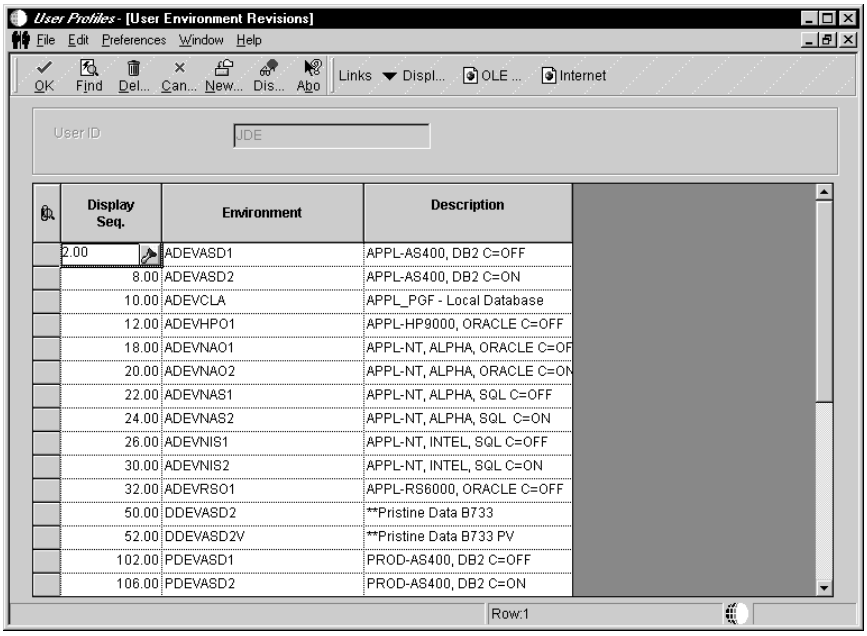
On the System Administration Tools menu (GH9011)

1. Choose User Profiles (P0092).

   The Work with User Profiles form appears.

2. Click Find, then choose a user profile.
3. From the Row menu, choose Environments.

The User Environment Revisions form appears. This form displays the list of environments available for a particular group.



4. To add a new environment, on the last row, complete the following fields:
   - Display Seq
   - Environment
5. To delete an environment from the list, choose the environment and click Delete.
6. Click OK.

Field	Explanation
Display Seq.	A number that the system uses to sequence information.
Environment	The name associated with a specific list of libraries. The J98INITA initial program uses these library list names to control environments that a user can sign on to. These configurations of library lists are maintained in the Library List Master table (F0094). For OneWorld, this field represents a valid environment that can be used to run OneWorld. The environment encompasses both a path code (objects) and a data source (data). When put together, users have a valid workplace within OneWorld.

Assigning Business Preferences to User and Group Profiles

You can assign business preferences to user profiles to create customized processes in conjunction with the OneWorld Workflow application. You define the codes for the preferences based on industry, business partner, or customer. You need to create a Workflow process that begins based on whether a specific code resides in the user profile.

For example, you assign the code CUS for a customer business preference, then create a Workflow process that begins whenever a user profile with the CUS business preference enters a sales order. The Workflow process might send a message, update a database, start another application, and so on.

See Also

- *Creating User and Group Profiles (System Administration)*
- *Creating Workflow Processes (Enterprise Workflow Management)*

▶ To assign a business preference

On the System Administration Tools menu (GH9011)

1. Choose User Profiles (P0092).

   The Work with User Profiles form appears.

2. Click Find, choose a user profile, then click Select.

   The User Profile Revisions form appears.

3. From the Row menu, choose Business Preferences.

   The Business Preferences form appears.

4. Complete any or all of the following fields, then click OK:

   - Industry Code

     This field associates the user profile with a specific industry, such as manufacturing.

   - Business Partner Code

     This field associates the user profile with a specific business partner.

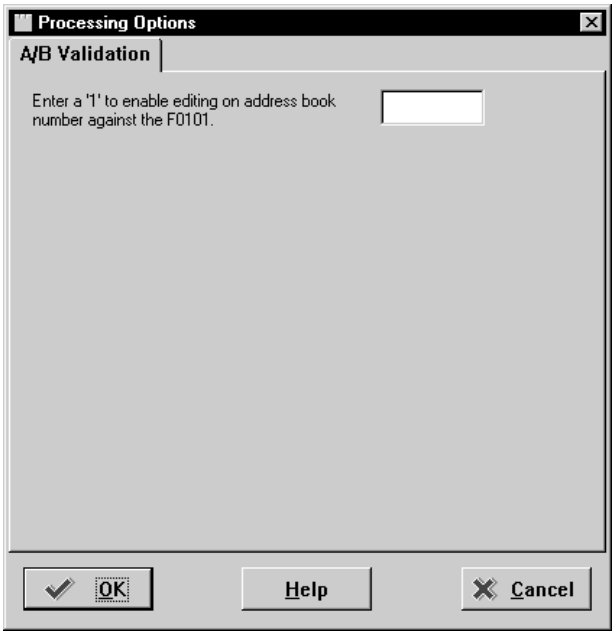
   - Customer Code

     This field associates the user profile with a specific customer.

Note: You can click Cancel on the Business Preferences form to cancel the addition of the current business preference.

Understanding Processing Options for User Profiles

The User Profiles application (P0092) has the following processing option:



On the Processing Options form, enter “1” to enable, or “0” (or leave blank) to disable Address Book validation.

- When enabled, this processing option causes User Profiles to validate, upon creation of a user profile, each new user ID against the Address Book Master (F0101) table. As a result, you cannot create a user profile for a user that is not already defined in the Address Book Master table. J.D. Edwards recommends that you enable this setting to ensure that Work Center will operate correctly because that application requires valid address book numbers.
- When disabled, this processing option allows you to create user profiles by for Address Book entries that do not yet exist in the Address Book Master (F0101) table.

Creating Profiles Using a Batch Process

If Address Book records already exist for employees, you can run a batch process to automatically create user profiles from those Address Book records. This process can save time, ensure accuracy between your Address Book and user profile records, and ease the transition of taking OneWorld to production.

You can create user profiles through the Create User Profile from A/B records batch process (R0092). With this process you can assign display and environment preferences to users. This process allows you to create literally hundreds of new user profiles at a time.

If you need to add just a few users, you should use the User Profile (P0092) application.

The User Profiles Summarization (R00921) report is useful if you need to review a list of user and group user profile definitions. This report summarizes the environment or environments assigned to a group, lists the users in the group, and notes any additional environments assigned specifically to an individual user. J.D. Edwards provides two default versions that allow you to summarize either all groups or only specific groups.

Complete the following tasks:

- Run the User Profiles Creation (R0092) batch application
- Run the User Profiles Summarization (R90021) report

Before You Begin

❑ Create all of the *group* profile information using the User Profile application (as explained in this section). You should have the following already defined:

- Group profiles
- Environments the groups can access

▶ To run the Create User Profiles from A/B Records (R0092) batch application

On the Advanced Operations (GH9012) menu

1. Choose Create User Profiles from A/B Records (R0092).
2. On the Work With Batch Versions form, choose the J.D. Edwards default version (XJDE0001) or equivalent for your installation, then click Select.
3. On the Versions Prompting form, click Data Selection, then click Submit.
4. On the Data Selection form, create a logic statement that describes the set of users for which you want to create profiles. This form already has a search type of “E” (employees) populated, assuming the users are all employees. You may want to narrow this selection by submitting it only for a range of employees.

   After you finish with the Criterion Design form, the Processing Options form appears.

5. On the Processing Options form, enter the following information:
    - Option 1: Enter one of the following values:
        - "1" to run this report in proof mode, which means that the report only provides an example of what would happen if you were to run the report in final mode.
        - Leave blank to run this report in final mode, which creates the user profiles you specified and issues a report showing the profiles created.
    - Option 2: Use to define the user profile record being created for each user. following user profile information:
        - Enter "1" to populate the User ID field with the users' address book number plus their initials.

          Note: Typically, user profiles are created with the users' initials preceding their Address Book number.

        - Leave this field blank to use just the address book number.

      The following are the user profile fields you need to complete for Option 2:

        - Group
        - Fast Path
        - Language
        - Date Format
        - Data Separator Character

- Data Format Character
- Country

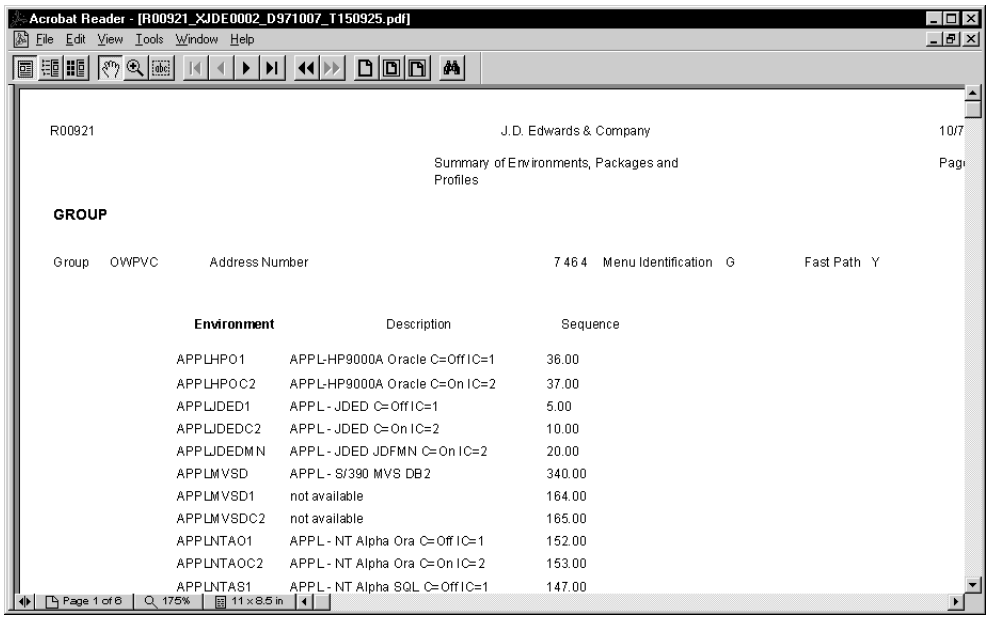
- Option 3: Enter any additional environments that you want the user to have access to instead of the environments already established for the user's group.

▶ To run the User Profiles Summarization (R00921) report

On the Advanced Operations menu (GH9012)

1. Choose Summarize Group Profile Information (R00921).
2. On the Work With Batch Versions form, choose a version, and click Select. The J.D. Edwards default version XJDE0001 creates a report for all group profiles in the enterprise. The J.D. Edwards default version XJDE0002 creates a report about a specific group profile that you specify.
3. On the Versions Prompting form, click Data Selection, and click Submit.
4. On the Data Selection form, create a logic statement that describes the group profiles you want to summarize. Click OK when finished.

   The following illustration provides an example of the User Profiles Summarization Report (R0092):



Security

OneWorld security enables a security administrator to control security for individual users and groups of users. The security administrator can control (secure or unsecure) users and groups from the following features:

- Application security. Controls access to or installation of specific applications.
- Action security. Controls ability to perform specific actions, such as adding, changing, deleting, selecting, or copying.
- Table row security. Controls access to a specific list or range of records within a table.
- Table column security. Controls access to a specific column within a table. Columns are represented in OneWorld as a field on a form or report.
- Processing option security. Controls the viewing or changing of the values for processing options, which would affect how that application works. It also controls whether users are allowed to prompt for versions of that application.
- User signon and database security. Prevents user access to the database from outside of OneWorld.

The Security Workbench application (P00950) uses the Security Workbench table (F00950).

The User Security application (P98OWSEC) uses the OneWorld Security table (F98OWSEC).

This section describes the following:

- ❑ Understanding Security Workbench
- ❑ Working with Security Workbench
- ❑ Understanding signon security
- ❑ Working with signon security

Understanding Security Workbench

This chapter describes the following information you need to understand Security Workbench:

- ❑ Understanding users, groups, and *PUBLIC
- ❑ Understanding how OneWorld checks security
- ❑ Understanding cached security information
- ❑ Understanding security types
- ❑ Understanding OneWorld object-level security
- ❑ Security and coexistence with WorldSoftware
- ❑ Identifying users and objects for security

Understanding Users, Groups, and *PUBLIC

The OneWorld security administrator can set up security for:

- A particular user. Controls security by a specific OneWorld user ID.
- A group of users. Controls security by group ID. This allows you to group users based on similar job requirements, such as putting all of the accounts payable clerks in one group, whose group ID could be AP.
- All users. Controls security for all users, designated by the ID type *PUBLIC in the Group ID field. The designation *PUBLIC is a special group ID within OneWorld that automatically includes all users within it. You can use this ID to apply security even if you do not specifically have a record set up for it in user profiles.

Understanding How OneWorld Checks Security

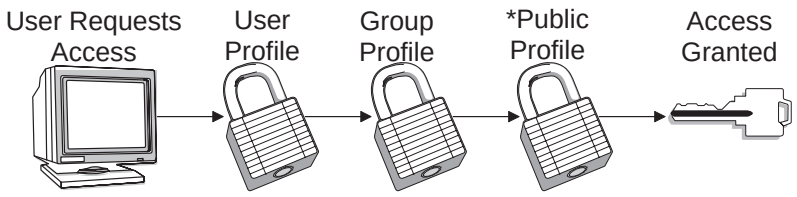
When a user attempts to access an application or perform an action, OneWorld:

1. Checks for any security for that particular user ID.

   If security exists for that individual user ID, OneWorld displays a message informing the user that they cannot proceed.

2 If there is no security for that user ID, OneWorld checks the group profile (if that user is part of a specific group), and *PUBLIC, in turn, for security.

   If no security is established at any of these levels, OneWorld lets the user continue. OneWorld also provides software license security through protection codes, and requires user validation at signon and when accessing new data sources.



Understanding Cached Security Information

OneWorld caches certain security information from the Security Workbench table (F00950). This information is cached in the workstation's memory cache for OneWorld. If system administrators make changes to the F00950 table, those changes will not be immediately realized on workstations that are logged on while the change is being made. Such workstations must log off and back on before the security changes are enabled.

Understanding Security Types

At specific object levels, you can set the following levels of security, alone or in any combination, for users and groups:

Application Security	Secure users from running and/or installing a particular application or a particular form within an application.

Action Security

Secure users from executing a particular action, such as adding, deleting, revising, inquiring, or copying a record.

Row Security

Secure users from accessing a particular range or list of records in any table.

For example, if you secure a user from accessing data about business units 1 through 10, the user cannot view the records that pertain to those business units.

Use row security to duplicate WorldSoftware Cost Center security.

Column Security

Secure users from viewing a particular field or changing a value for a particular field. This can be a database or non-database field that is defined in the data dictionary, such as the work/calculated fields.

For example, if you secure a user from viewing the Salary field on the Employee Master application, the Salary field does not display on the form when that particular user accesses that application.

Processing Option Security

Secure users from changing the values of processing options, or from prompting for versions and prompting for values for specific applications.

For example, if you secure a user from changing the processing options for Address Book Revisions, they could still view the processing options (if you did not secure them from prompting for values), but they would not be able to change any of the values.

If you secure them from prompting for versions, they would not be able to see the versions for a specific application, which means they would not be able to choose a different version of an application from the version that the administrator set up for them.

Tab Security

Secure users from seeing a tab or tabs on a given form.

Exit Security

Secure users from menu bar exits on OneWorld forms. These exits call applications and allow users to manipulate data. Exit security also provides restrictions to the hyper-button.

Exclusive Application Security

Sets security specific to an application regardless of any other security that might be set. When you set exclusive application security for a user, that user gains access to just the specific application that you define. All other security still applies.

External Calls Security	Secure users from accessing stand-alone executables that exist external to OneWorld. These external executables, which might include design tools, system monitors, and debugging tools, are specific to OneWorld.

Understanding OneWorld Object-level Security

OneWorld security is at the object level. This means you can secure specific objects within OneWorld, which provides flexibility and integrity for your security. For example, you can secure a user from a specific form, and no matter how the user tries to access the form (via menu or any application that calls that form), OneWorld prevents them from accessing that form. Though setting up good security is always a challenge, OneWorld simplifies the process by allowing you to set security for literally hundreds of objects at a time by securing all objects on a specific menu, or securing all objects under a specific system code. But remember, it is the objects that are secured; OneWorld does not support menu nor system code security. Object security provides a higher integrity. For example, if you were to secure a specific menu, to prevent users from accessing the applications on that menu, the users might still be able to access those applications through some other menu or some other application that accesses those applications you were trying to secure.

Security and Coexistence with WorldSoftware

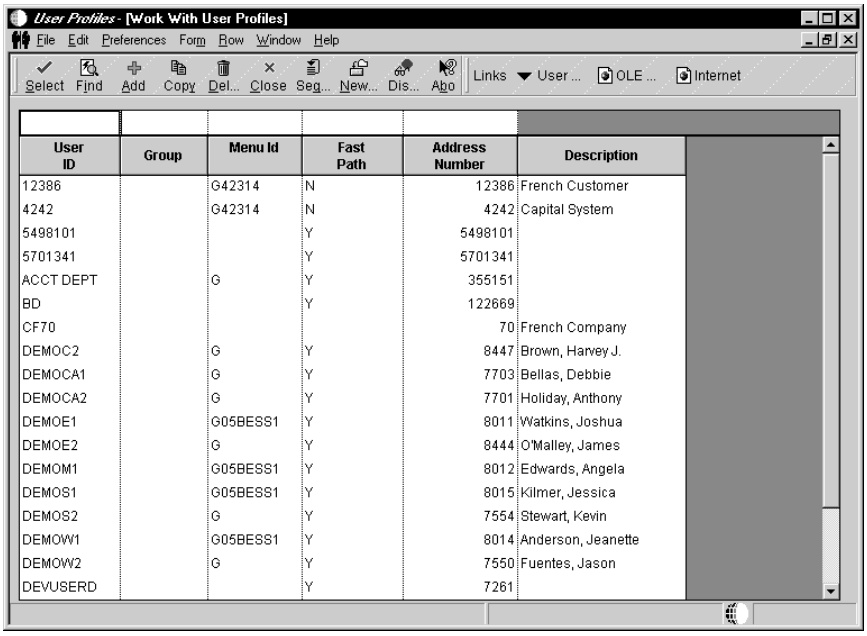
If you have a coexistence environment, where you share data and applications between OneWorld and WorldSoftware, you need to maintain two independent sets of security profiles: one for WorldSoftware and one for OneWorld.

Identifying Users and Objects for Security

To set up security you first need to identify the users and the objects you want to secure, as explained in this task.

To identify users and objects for security

1. Identify for whom you want to set up security. Use the Work With User Profiles form to find a user or group ID. See “Setting Up User Profiles” in this guide for information on how to use this form.



User ID	Group	Menu Id	Fast Path	Address Number	Description
12386		G42314	N	12386	French Customer
4242		G42314	N	4242	Capital System
5498101			Y	5498101	
5701341			Y	5701341	
ACCT DEPT		G	Y	355151	
BD			Y	122669	
CF70				70	French Company
DEMOC2		G	Y	8447	Brown, Harvey J.
DEMOCA1		G	Y	7703	Bellas, Debbie
DEMOCA2		G	Y	7701	Holiday, Anthony
DEMOE1		G05BESS1	Y	8011	Watkins, Joshua
DEMOE2		G	Y	8444	O'Malley, James
DEMOM1		G05BESS1	Y	8012	Edwards, Angela
DEMOS1		G05BESS1	Y	8015	Kilmer, Jessica
DEMOS2		G	Y	7554	Stewart, Kevin
DEMOW1		G05BESS1	Y	8014	Anderson, Jeanette
DEMOW2		G	Y	7550	Fuentes, Jason
DEVUSERD			Y	7261	

2. Identify what you want to secure:

   - For all security, identify which application, form, report, or table you want secured. This is the object name, such as F0101 for the Address Book Master file, P0101 for the Address Book application, or *ALL for all objects.
   - For only row and column security, also identify which columns (data items) you want secured. This is the data dictionary item name, such as MCU for the Business Unit/Branch Plant field, or CO for the Company Name. Column security can be for dictionary items that are not in database tables.

Working with Security Workbench

This chapter describes the following information you need work with Security Workbench:

- ❑ Setting up application security
- ❑ Setting up action security
- ❑ Setting up row security
- ❑ Setting up column security
- ❑ Setting up processing option security
- ❑ Setting up tab security
- ❑ Setting up exit security
- ❑ Setting up exclusive application security
- ❑ Setting up external calls security
- ❑ Copying security for a user or a group
- ❑ Deleting security on the Work with User/Group Security form

Note: You can also secure the query-by-example (QBE) line and the fast path tool. For more information about securing these features, see *Working with the Object Configuration Manager* in the *CNC Implementation* guide and *Setting Up User Profiles* in this guide respectively.

Setting Up Application Security

This task explains how to add, revise, and remove application security. You can secure users from running and/or installing a particular application or a particular form within an application. This task also explains how to add a *ALL object, how to change all of the applications for a particular user or group from unsecured to secured, and how to set security for all but one form in an application.

▶ To set up application security

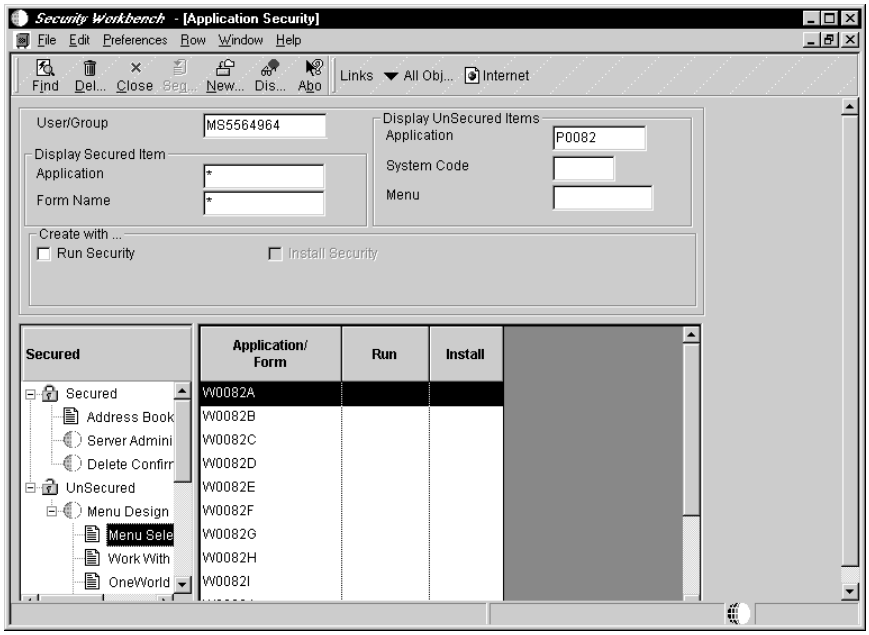
On the Security Maintenance menu (GH9052)

1. Choose Security Workbench (P00950).

   The Work with User/Group Security form appears.

2. From the Form menu, choose Set Up Security, then Application.

   The Application Security form appears.



3. Complete the following fields, and click Find:

   - User/Group

     Enter a complete user or group ID, which includes *PUBLIC, but not wildcards.

The following fields are mutually exclusive:

- Application

  Enter an application name, such as P0101. You can also use an asterisk * by itself or in alphanumeric combinations to search for all or subsets of all applications.

- Form Name

  Enter a form name, such as W0101G. You can also use an asterisk * by itself or in alphanumeric combinations to search for all or subsets of all forms.

Current security settings for the user or group appear under the Secured node in the tree. Expand the node to view the individual applications and forms that are secured. After expanding the node, the secured applications and forms also appear in the grid.

4. Complete *one* of the following fields that appear in the Display UnSecured Items heading, and click Find:

   - Application

     You can use *ALL in this field to select *all* OneWorld objects.

     In the grid, this special object appears as *ALL and displays the security that you defined for the object, such as Run Security or Install Security. The *ALL object acts as any other object and you can use the Revise Security and Remove All options from the Row menu.

   - System Code
   - Menu

   You must perform this step before you can add new security. This step provides a list of applications and forms for you to choose from.

   Your search (application, system code, or menu) appears under the UnSecured node. Expand the node to view individual applications and forms that do not already have security set for them. After expanding the node, the individual applications and forms also appear in the grid.

   For example, to set security on the Menu Design application, you would first need to display it under the UnSecured node. To do this, you would enter P0082 in the Application field, and click Find. The Menu Design node appears when you expand the UnSecured node. You could then expand Menu Design to secure a form within the application.

5. Perform the following actions as necessary to add, change, or remove application security:

Add security to an application

Under the Create With heading, click one or both of the following options:

- Run Security
- Install Security

  Use this option for just-in-time installation only.

Then do one of the following:

- Drag applications and forms from the UnSecured node to the Secured node.
- From the Row menu, choose All Objects to move all applications to the Secured node.
- From the Row menu, choose Secure to All to move all objects beneath the UnSecured node to the Secured node.

The applications or forms now appear under the Secured node with the appropriate security.

For example, to set run security on the Menu Design application (P0082), turn on the Run Security option, then drag the Menu Design node from the UnSecured node to the Secured node. The grid would reflect the run security that you set for this application. This would mean that the user you entered could *not* run the Menu Design application.

Change security for an application

Under the Secured node, choose an application or form, click one or both of the following options, and then from the Row menu, choose Revise Security:

- Run Security
- Install Security

  Use this option for just-in-time installation only.

The values under the Run and Install fields in the grid change accordingly.

Remove security from an application

Do one of the following:

- Under the Secured node, choose an application or form, and click Delete.

- Under the Secured node, drag an application or form from the Secured node to the UnSecured node.
- On the Row menu, choose Remove All. This moves *all* applications and forms from the Secured node to the UnSecured node.

Field	Explanation
User/ Group	For World, The IBM-defined user profile. For OneWorld, the identification code for a user profile.
Object Name	The OneWorld architecture is object based. This means that discrete software objects are the building blocks for all applications, and that developers can reuse the objects in multiple applications. Each object is stored in the Object Librarian. Examples of OneWorld objects include: • Batch Applications • Interactive Applications • Business Views • Business Functions • Business Functions Data Structures • Event Rules • Media Object Data Structures
Object	A user defined name or remark.
Security Type	OneWorld Security Type options are: • Action Security. Places security to enable or disable users from using specified functions such as Add, Change, Delete, and Copy. • Application Security. Places security to enable or disable users from viewing or using specified applications or reports. • Column Security. Places security on specified data fields to be non-display or read only. • Row Security. Places security to enable or disable users from performing Add, Change, Delete or View actions to records from the database. • Process Option Security. Places security on viewing or changing the values of processing options, or places security on prompting for versions for specific applications.
Description	A user defined name or remark.

Field	Explanation
Data Item	An identifier that refers to and defines a unit of information. It is a 32-character, alphabetical field that does not allow blanks or special characters such as $ % & , . + or @. The data item cannot be changed. It forms the C-code data name (for example AddressNumber) that is used in business functions, data structures, and event rules. Also identify a data item by the alias or alpha description.
From Value	The From Data Value is used by the row security routines to determine a lower range for the data item in the specified table. It is used in conjunction with the Thru Data Value to define the range of data that the security applies to.
Thru Value	The Thru Data Value is used by the row security routines to determine a upper range for the data item in the specified table. It is used in conjunction with the From Data Value to define the range of data that the security applies to.
Alias	For World, the RPG data name. This data field has been set up as a 10-byte field for future use. Currently, it is restricted to 4 bytes so that, when preceded by a 2-byte table prefix, the RPG data name will not exceed 6 bytes. Within the Data Dictionary, all data items are referenced by this 4-byte data name. As they are used in database tables, a 2-character prefix is added to create unique data names in each table specification (DDS). If you are adding an error message, this field must be left blank. The system assigns the error message number using next numbers. The name appears on a successful add. You should assign error message numbers greater than 5000. Special characters are not allowed as part of the data item name, with the exception of #, @, $. You can create protected data names by using $xxx and @xxx, where you define xxx. For OneWorld, a code that identifies and defines unit of information. It is an 8-character, alphabetical code that does not allow blanks or special characters such as: % & , . +. Create new data items using system codes 55-59. You should name your new data items with a dollar sign ($). For example, $DTAI. The alias cannot be changed.
View	This code designates whether a user has the authority to view data either on a specific application or form (processing option and column security) or for a specific table and data item (row security). This code is set up through Security Workbench by user or group for every table, application, or form requiring security. *ALL can be used to designate all tables or applications.

Field	Explanation
Add	This code designates whether a user has the authority to perform an add, either on a specific application or form (action and column security) or for a specific table and data item (row security). This code is set up through Security Workbench by user or group for every table, application, or form requiring security. *ALL can be used to designate all tables or applications.
Change	This code designates whether a user has the authority to perform changes, either on a specific application or form (action, processing option, and column security) or for a specific table and data item (row security). This code is set up through Security Workbench by user or group for every table, application, or form requiring security. *ALL can be used to designate all tables or applications.
Delete	This code designates whether a user has the authority to perform deletes, either on a specific application or form (action security) or for a specific table and data item (row security). This code is set up through Security Workbench by user or group for every table, application, or form requiring security. *ALL can be used to designate all tables or applications.
OK/ Select	This code designates whether a user has the authority to perform operations associated with the OK or Select button on a specific application or form (action security).
Copy	This code designates whether a user has the authority to perform operations associated with the Copy button on a specific application or form (action security).
Scroll to End	This code designates whether a user is given the option to scroll to the end of the data selected or if they will only be given the choice to view it a page at a time. This option is application or form specific (action security).
Prompt for Versions	This code designates whether a user will be allowed to prompt for the version of an application to run from a menu selection. This option is application specific and is set up by the user.
Prompt for Values	This code designates whether a user has the authority to 'VIEW' records in programs that are using Action Code Security. The code is set up in Action Code Security Revisions (P0003) by user for every program requiring security by action code.
Run	This code designates whether a user has the authority to run a specific application.
Install	This code designates whether a user has authority to run a Just In Time Installation (JITI) of a specific application.

Field	Explanation
Syst Code	A user defined code (98/SY) that identifies a J.D. Edwards system.

Setting Up Action Security

This task explains how to add, revise, and remove action security. You can secure users from executing a particular action, such as adding, deleting, revising, inquiring, or copying a record. At the end of this task are additional topics explaining how to add a *ALL object and how to move all of the applications for a particular user or group from unsecured to secured.

▶ To set up action security

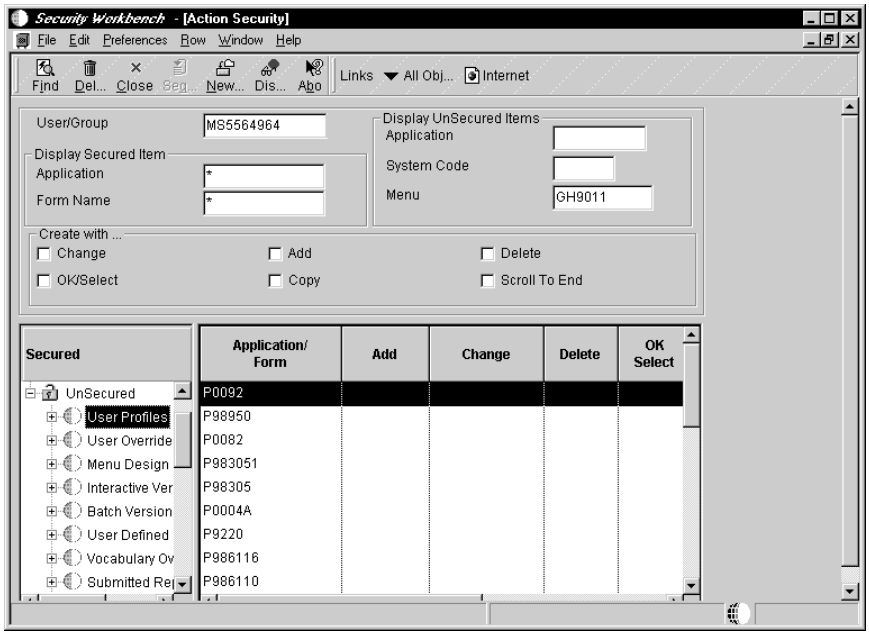
On the Security Maintenance menu (GH9052)

1. Choose Security Workbench (P00950).

   The Work with User/Group Security form appears.

2. From the Form menu, choose Set Up Security, then Action.

   The Action Security form appears.



3. Complete the following fields, and click Find:

   - User/Group

     Enter a complete user or group ID, which includes *PUBLIC, but not wildcards.

   The following fields are mutually exclusive:

   - Application

     Enter an application name, such as P0101. You can also use an asterisk * by itself or in alphanumeric combinations to search for all or subsets of all applications.

   - Form Name

     Enter a form name, such as W0101G. You can also use an asterisk * by itself or in alphanumeric combinations to search for all or subsets of all forms.

   Current security settings for the user or group appear under the Secured node in the tree. Expand the node to view the individual secured applications and forms. After expanding the node, the individual applications and forms that are secured also appear in the grid.

4. You must perform this step before you can add new security, because this step provides a list of applications and forms for you to choose from. To search on applications and forms you want to secure, complete *one* of the following fields that appear under the Display UnSecured Items heading, and click Find:

   - Application
   - System Code
   - Menu

   Your search (application, system code, or menu) appears under the UnSecured node. Expand the node to view individual applications and forms. After expanding the node, the individual applications and forms also appear in the grid.

   For example, to set security on applications under the System Administration Tools menu, you would enter GH9011 in the Menu field, and click Find. All of the applications and menus attached to GH9011 appear after you expand the UnSecured node. You could then expand the applications and menus.

5. Perform the following actions as necessary to add, change, or remove action security:

Add action security

Under the Create With heading, click one or more of the following options:

- Change
- Add
- Delete
- OK/Select
- Copy
- Scroll To End

Then do one of the following:

- Drag applications and forms from the UnSecured node to the Secured node.
- From the Row menu, choose All Objects to move all applications to the Secured node.
- From the Row menu, choose Secure to All to move all objects beneath the UnSecured node to the Secured node.

The applications or forms now appear under the Secured node with the appropriate action security.

For example, to set delete security on an application, turn on the Delete option. Next, you could drag the application from the UnSecured node to the Secured node. The grid would reflect the delete security that you set for these applications. This would mean that the user you entered could *not* perform the delete action on any applications you placed under the Secured node.

Change action security

Under the Secured node, choose an application or form, click on one or more of the following options, and then from the Row menu, choose Revise Security:

- Change
- Add
- Delete
- OK/Select
- Copy
- Scroll To End

The values under the Add, Change, Delete, OK Select, Copy, Scroll to End fields in the grid change accordingly.

Remove action security

Do one of the following:

- Under the Secured node, choose an application or form, and click Delete.
- Under the Secured node, drag an application or form from the Secured node to the UnSecured node.
- On the Row menu, choose Remove All. This moves *all* applications and forms from the Secured node to the UnSecured node.

Setting Up Row Security

This task explains how to add and revise row security. You can secure users from accessing a particular range or list of data in any table. Use row security sparingly because it can have an adverse affect on performance. Additional processing occurs for each data item that you set with row security.

You can set up row security at three levels: user, group and *PUBLIC. OneWorld first looks for row security at the user level, then at the group level, and then *PUBLIC. If you set any security at a higher level, such as at the user level, OneWorld ignores any security set at lower levels, such as at the group or *PUBLIC levels. If you set security at a higher level, you must ensure that you are very complete with the setup.

Before you can set up row security, you must turn on row security in Data Dictionary Design. The first steps in this task describe this process.

Complete the following tasks:

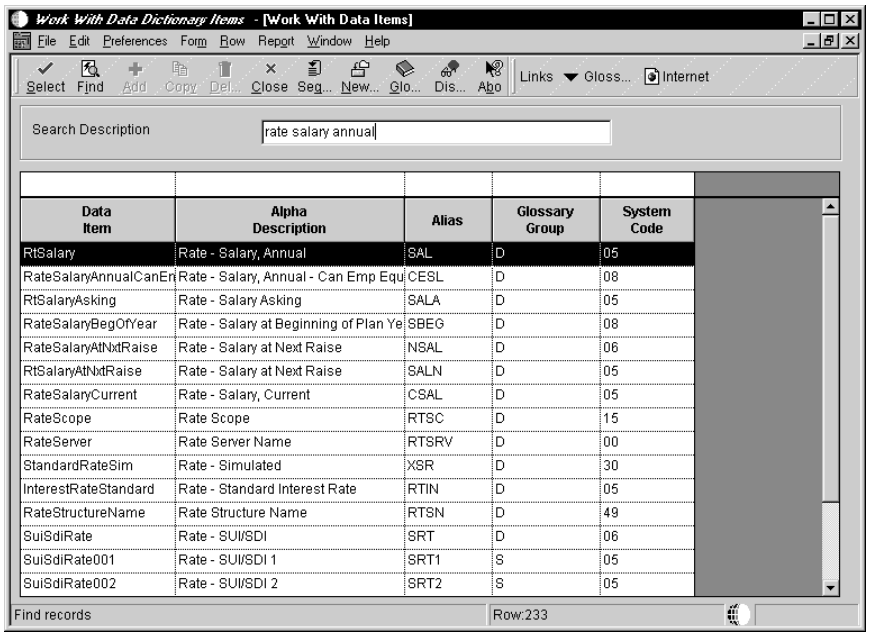
- Set up row security
- Delete row security on the Row Security Revisions form

▶ To set up row security

On the Data Dictionary Design menu (GH951)

1. Choose Work with Data Dictionary Items (P92001).

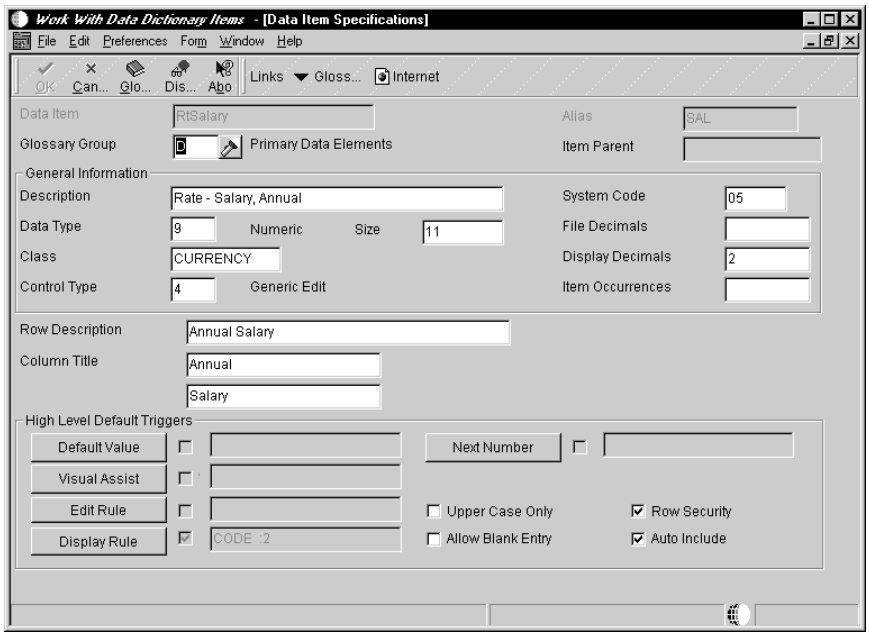
   The Work with Data Items form appears.



2. Click Find, choose a data item that you want to secure, then click Select.

   Note: You can enter search criteria in the Search Description field and the QBE line to narrow your search.

   The Data Item Specifications form appears.



3. Turn on the Row Security option and click OK.

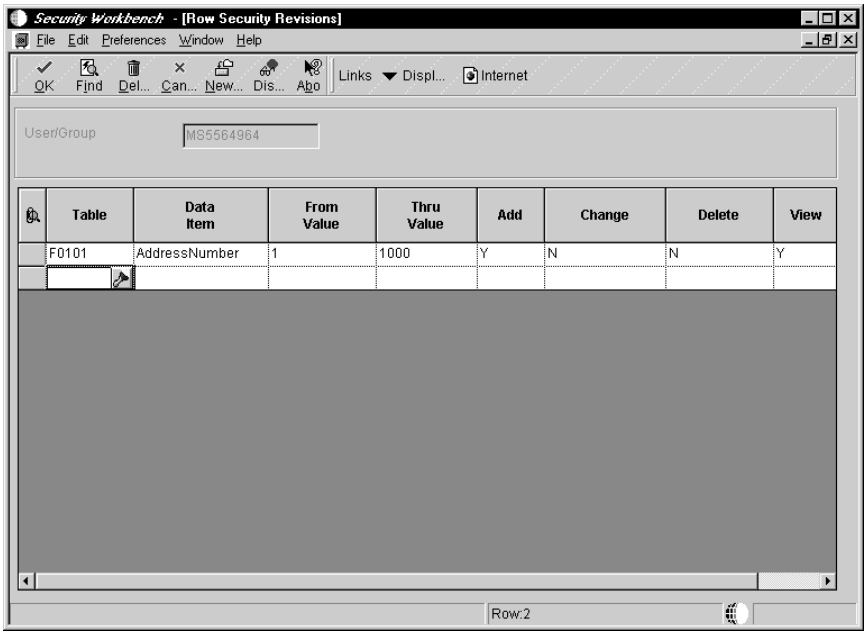
   This option must be turned on for row security to work.

4. On the Security Maintenance menu (GH9052), choose Security Workbench (P00950).

   The Work with User/Group Security form appears.

5. From the Form menu, choose Set Up Security, then Row.

   The Row Security Revisions form appears.



6. Complete the following field, and click Find to display current row security:

   - User/Group

7. Enter information in the following fields either in the first open grid row (to add security) or in a pre-existing grid row (to change security):

   - Table

     You can use *ALL in this field.

   - Data Item (required)
   - From Value (required)
   - Thru Value
   - Add
   - Change
   - Delete
   - View

8. Click OK to save your security information.

▶ To delete security on the Row Security Revisions form

On the Row Security Revisions form

1. Complete the following field, then click Find:
   - User/Group

   Note: If you accessed the Row Security Revisions form from the Work with User/Group Security for a specific record, the user or group associated with the security record appears in the User/Group field by default.
2. Highlight the security record or records in the grid, then click Delete.
3. On Confirm Delete, click OK.
4. Click OK when you finish deleting row security.

   If you do not click OK after you delete the row security record or records, OneWorld does not save the deletion.

Setting Up Column Security

This task explains how to add and revise column security. You can secure users from viewing a particular field or changing the value for a particular field. This can be a database field or a field that is defined in the data dictionary but is not in the database.

Complete the following tasks:

- Set up column security
- Delete security on the Column Security revisions form

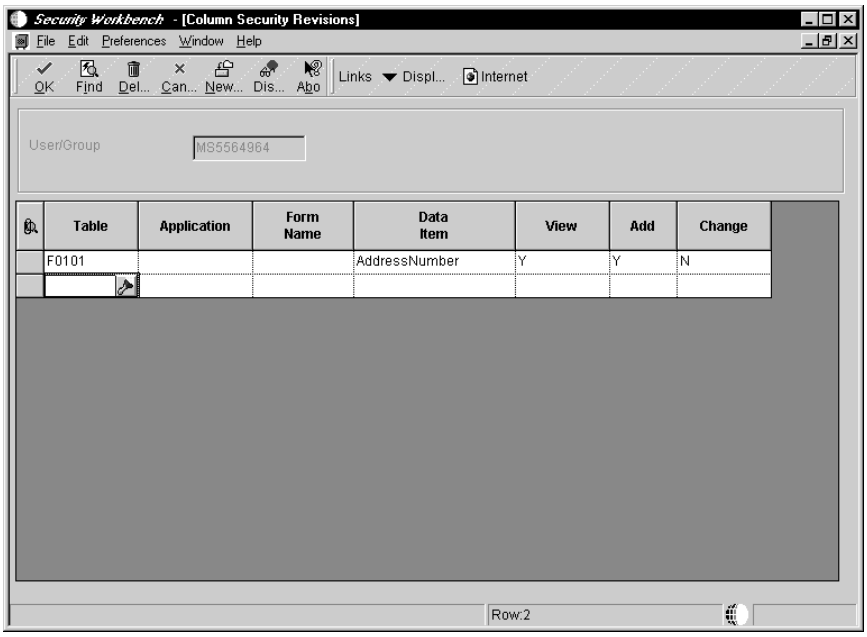
▶ To set up column security

On the Security Maintenance menu (GH9052)

1. Choose Security Workbench (P00950).

   the Work with User/Group Security form appears.
2. From the Form menu, choose Set Up Security, then Column.

   The Column Security Revisions form appears.



3. Complete the following field, and click Find to display current column security:
   - User/Group
4. To add new security, in the last row on the grid enter information into only *one* of the following fields:
   - Table
   - Application
   - Form Name

   You can use *ALL in any of these fields; however, after *ALL is used for a table, application, or form for a specific data item, you cannot use *ALL again for that data item.
5. Complete the following fields.
   - Data Item
   - View
   - Add
   - Change
6. To change security, change the grid row values.
7. Click OK when you are finished entering security.

▶ To delete security on the Column Security Revisions form

On the Column Security Revisions form

1. Complete the following field, then click Find:
   - User/Group

   Note: If you accessed the Column Security Revisions form from the Work with User/Group Security for a specific record, the user or group associated with the security record appears in the User/Group field by default.

2. Highlight the security record or records in the grid, then click Delete.
3. On Confirm Delete, click OK.
4. Click OK when you finish deleting column security.

   If you do not click OK after you delete the security record or records, OneWorld does not save the deletion.

Setting Up Processing Option Security

This task explains how to add, revise, and remove processing-option security. You can secure users from changing, prompting for values, and prompting for versions of specific processing options. By itself, setting security that prohibits users from prompting for versions will not prevent them from changing values in the processing option. If you do not want users changing processing option values at all, you may want to set security so that users are secured from prompt for value and prompt for versions.

This task also explains how to add a *ALL object and how to move all of the applications for a particular user or group from unsecured to secured.

▶ To set up processing option security

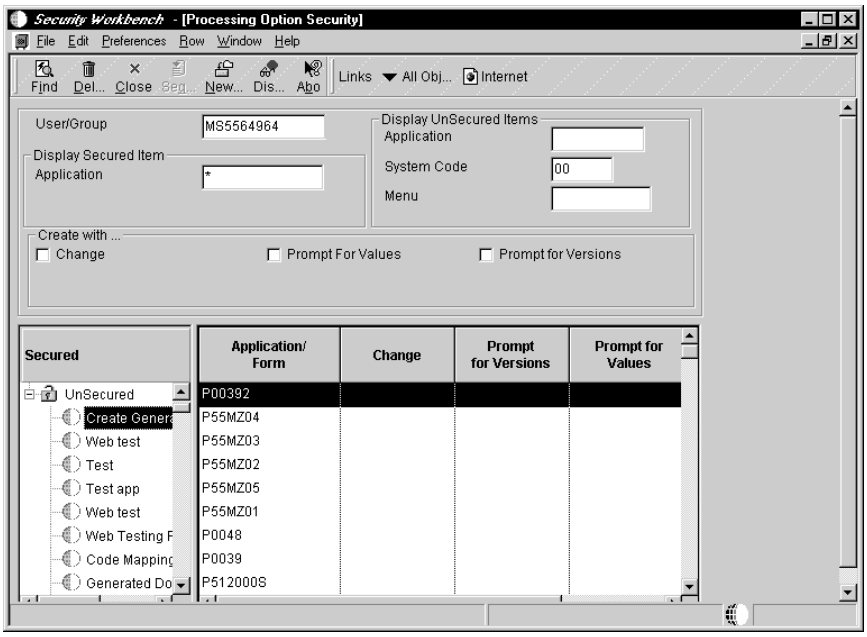
On the Security Maintenance (GH9052) menu

1. Choose Security Workbench (P00950).

   The Work with User/Group Security form appears.

2. From the Form menu, choose Set Up Security, then Processing Option.

   The Processing Option Security form appears.



3. Complete the following fields, and click Find:

   - User/Group

     Enter a complete user or group ID, which includes *PUBLIC, but not wildcards.

   - Application

     Enter an application name, such as P0101. You can also use an asterisk * by itself or in alphanumeric combinations to search for all or subsets of all applications.

   Current security settings for that user or group appear under the Secured node in the tree. Expand the node to view the individual secured applications. After expanding the node, the applications that are secured also appear in the grid.

4. Complete *one* of the following fields that appear under the UnSecured Items For heading, and click Find:

   - Application
   - System Code
   - Menu

   You must perform this step before you can add new security This step provides a list of applications from which to choose.

   Your search (application, system code, or menu) appears under the UnSecured node. Expand the node to view applications (interactive and batch) and/or menus with interactive or batch applications. After expanding the node, the applications also appear in the grid.

For example, to set security on applications within the 00 system code, you would enter 00 in the System code field, and click Find. All of the applications (interactive and batch) attached to system code 00 appear after you expand the UnSecured node.

5. Perform the following actions as necessary to add, change, or remove processing option security:

Add security to processing options

Under the Create With heading, click one or more of the following options, then drag applications from the UnSecured node to the Secured node:

- Change
- Prompt for Values

  When you turn on this option, you automatically turn on the Change option.

- Prompt for Versions

Then do one of the following:

- Drag applications from the UnSecured node to the Secured node.
- From the Row menu, choose All Objects to move all applications to the Secured node.
- From the Row menu, choose Secure to All to move all objects beneath the UnSecured node to the Secured node.

The applications now appear under the Secured node with the appropriate security.

For example, to set prompt-for-values security, which also automatically sets change security, turn on the Prompt for Values option. Next, drag one application at a time from the UnSecured node to the Secured node. The grid reflects the prompt-for-values and change security that you set for these applications. This means that the user you entered could *not* prompt for values or change processing options on any applications you dragged to the Secured node.

Change security for processing options

Under the Secured node, choose an application, click one or more of the following options, and then from the Row menu, choose Revise Security:

- Change
- Prompt for Values

  When you turn on this option, you automatically turn on the Change option.

- Prompt for Versions

The values under the Change, Prompt for Values, and Prompt for Versions fields in the grid will change accordingly.

Remove security from processing options

Do one of the following:

- Under the Secured node, choose an application, and click Delete.
- Under the Secured node, drag an application from the Secured node to the UnSecured node.
- On the Row menu, choose Remove All. This moves *all* applications from the Secured node to the UnSecured node.

Setting Up Tab Security

This task describes how to add, change, and remove security for forms as tabs. You can secure users from changing the name of the tab and viewing the form that you call using the tab.

If you secure a user from an application, you cannot also secure the user from certain tabs on a form in that application. This restriction prevents redundant "double" security. Similarly, if you secure a user from a tab, you cannot secure the user from the application that contains the tab.

▶ To set up tab security

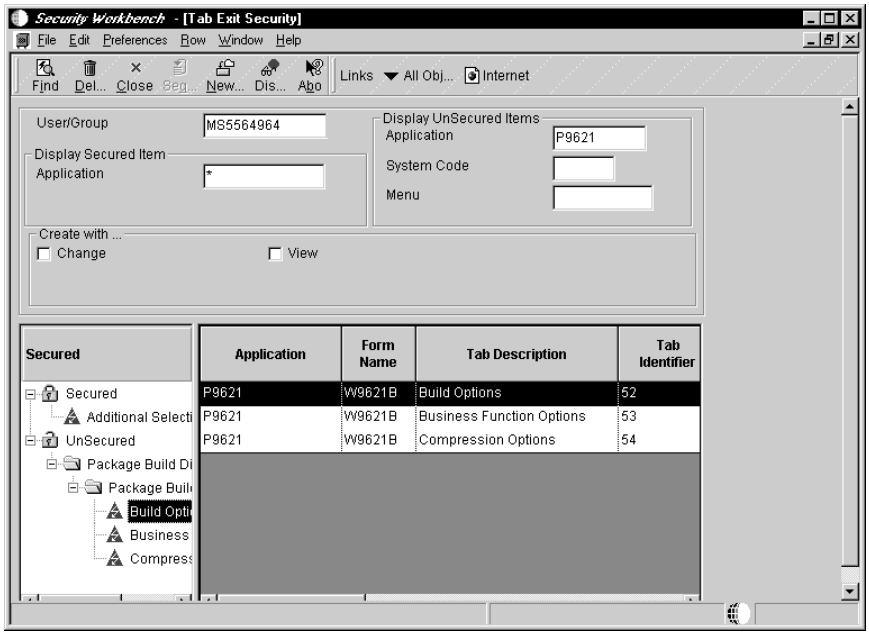
On the Security Maintenance menu (GH9052)

1. Choose Security Workbench (P00950).

   The Work with User/Group Security form appears.

2. From the Form menu, choose Set Up Security, then Tab Security.

   The Tab Exit Security form appears.



3. Complete the following fields, and click Find:

   - User/Group

     Enter a complete user or group ID, which includes *PUBLIC, but not wildcards.

   - Application

     View security for a specific application, or use an asterisk * by itself to perform a wildcard search on all applications.

   Current security settings for that user or group appear under the Secured node in the tree. Expand the nodes to view the secured tabs. After expanding the node, the secured tabs also appear in the grid.

4. Complete *one* of the following fields that appear in the Display UnSecured Items heading, and click Find:

   - Application

     You can use *ALL in this field to select *all* OneWorld objects.

     In the grid, this special object appears as *ALL and displays the security that you defined for the object, such as Run Security or Install Security. The *ALL object acts as any other object and you can use the Revise Security and Remove All options from the Row menu.

- System Code
- Menu

You must perform this step before you can add new security. This step provides a list of applications from which to choose.

Your search (application, system code, or menu) appears under the UnSecured node. Expand the nodes to view applications (interactive and batch) and the associated tabs. After expanding the node, the applications or tabs also appear in the grid.

For example, to set security for tabs in applications within the 00 system code, you would enter 00 in the System code field, and click Find. All of the applications (interactive and batch) attached to system code 00 appear after you expand the UnSecured node.

5. Perform the following actions as necessary to add, change, or remove tab security:

Add security to a tab

Under the Create With heading, click on one or more of the following options, then drag tabs from the UnSecured node to the Secured node:

- Change

  Click this option to prohibit a user or group from changing information on that tab page.

- View

  Click this option to hide the tab from the user or the group.

The hyper-button exit that you dragged appears under the Secured node the appropriate security.

For example, to set up change security, click the Change option. Next, drag tabs one at a time from the Unsecured node to the Secured node. The grid reflects the change security that you set for the tabs. This security means that the user you entered cannot change the tabs that you dragged to the Secured node.

Change security for a tab

Under the Secured node, choose a tab, click on one or more of the following options, then from the Row menu, choose Revise Security:

- Change

  Click this option to prohibit a user or group from changing the name of the tab.

- View

  Click this option to hide the tab from the user or the group.

The values under the Change and View fields in the grid change accordingly.

Remove security from a tab

Do one of the following:

- Under the Secured node, choose a tab, and click Delete.
- Under the Secured node, drag a tab from the Secured node to the UnSecured node.
- On the Row menu, choose Remove All. This moves *all* tabs from the Secured node to the UnSecured node.

Setting Up Exit Security

This task describes how to add, change, and remove security for the menu bar exits on OneWorld forms. These exits call applications and allow users to manipulate data. Exit security also provides restrictions for the hyper-button.

▶ To set up exit security

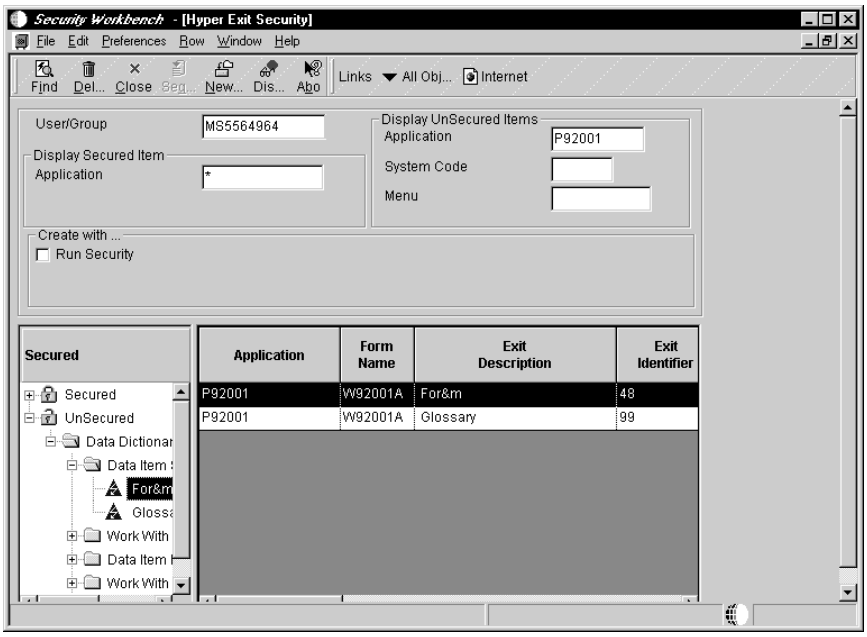
On the Security Maintenance (GH9052) menu

1. Choose Security Workbench (P00950).

   The Work with User/Group Security form appears.

2. From the Form menu, choose Set Up Security, then Exit Security.

   The Hyper Exit Security form appears.



3. Complete the following fields, and click Find:

   - User/Group

     Enter a complete user or group ID, which includes *PUBLIC, but not wildcards.

   - Application

     View security for a specific application. You can also use an asterisk * by itself or in alphanumeric combinations to search for all or subsets of all applications.

   Current security settings for that user or group appear under the Secured node in the tree. Expand the node to view the individual secured applications, such as interactive and batch. After expanding the nodes, the secured hyper-button exits also appear in the grid.

4. Complete *one* of the following fields that appear in the Display UnSecured Items heading, and click Find:

   - Application

     You can use *ALL in this field.

   - System Code
   - Menu

   You must perform this step before you can add new security. This step provides a list of applications from which to choose.

Your search (application, system code, or menu) appears under the UnSecured node. Expand the nodes to view applications (interactive and batch) and hyper-button exits. After expanding the nodes, the hyper-button exits also appear in the grid.

For example, to set security on hyper-buttons in applications within the 00 system code, you would enter 00 in the System code field, and click Find. All of the applications (interactive and batch) attached to system code 00 appear after you expand the UnSecured node.

5. Perform the following actions as necessary to add, change, or remove exit security:

Add security to an exit

Under the Create With heading, click the following option, then drag exits from the UnSecured node to the Secured node:

- Run Security

The exit that you dragged appears under the Secured node the appropriate security.

For example, to set Run Security, click the Run Security option. Next, drag exits one at a time from the UnSecured node to the Secured node. The grid reflects the security that you set for these exits. This security means that the user you entered cannot use the exit.

Change security for an exit

Under the Secured node, choose an exit, click the following option, and then from the Row menu, choose Revise Security:

- Run Security

The values under the Run field in the grid change accordingly.

Remove security from an exit

Do one of the following:

- Under the Secured node, choose an exit, and click Delete.
- Under the Secured node, drag an exit from the Secured node to the UnSecured node.
- On the Row menu, choose Remove All. This moves *all* exits from the Secured node to the UnSecured node.

Setting Up Exclusive Application Security

This task describes how to grant access to otherwise secured information through one exclusive application. For example, you might use row security to secure a user from seeing salary information, but the user needs to run a report for payroll that includes salary information. You can grant access to the report, including the salary information, using exclusive application security. OneWorld still secures the user from all other instances where salary information would appear.

Complete the following tasks:

- Add access with exclusive application security
- Delete access on the Exclusive Application Security form

▶ To add access with exclusive application security

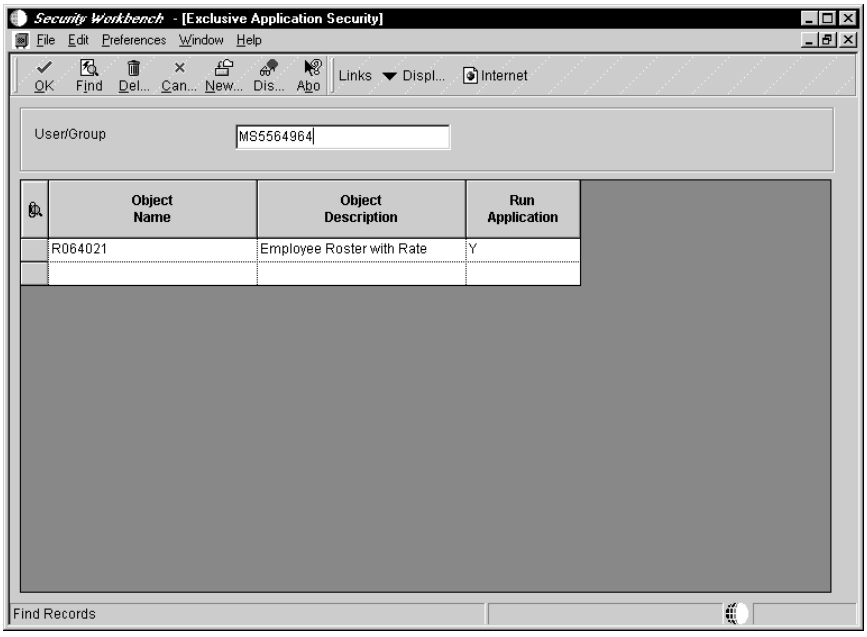
On the Security Maintenance menu (GH9052)

1. Choose Security Workbench (P00950).

   The Work with User/Group security form appears.

2. From the Form menu, choose Set Up Security, then Exclusive Application.

   The Exclusive Application Security form appears.



3. Complete the following field:

   - User Group

     Enter a complete user or group ID, which includes *PUBLIC, but not wildcards.

4. Complete the following fields in the detail area:

   - Object Name

     Enter the name of the exclusive application through which you want to allow access (the security). For example, to change the security for user of the Vocabulary Overrides application, enter P9220 in this field.

   - Run Application

5. Click OK to save the information.

▶ To delete access on the Exclusive Application Security form

On the Exclusive Application Security form

1. Complete the following field, then click Find:

   - User/Group

   Note: If you accessed the Exclusive Application Security form from the Work with User/Group Security for a specific record, the user or group associated with the security record appears in the User/Group field by default.

2. Highlight the security record or records in the grid, then click Delete.
3. On Confirm Delete, click OK.
4. Click OK when you finish deleting exclusive application security.

   If you do not click OK after you delete the security record or records, OneWorld does not save the deletion.

Setting Up External Calls Security

This task describes how to secure users and groups from access to external call applications. In OneWorld, certain applications exist that are not internal to OneWorld; they are stand-alone executables. For example, the Report Design Tool resides on menu GH902 is a stand-alone application. You can also call this application externally using the RDA.exe. By default, this file resides in the \B7\SYSTEM\Bin32 directory.

To set up security for external calls

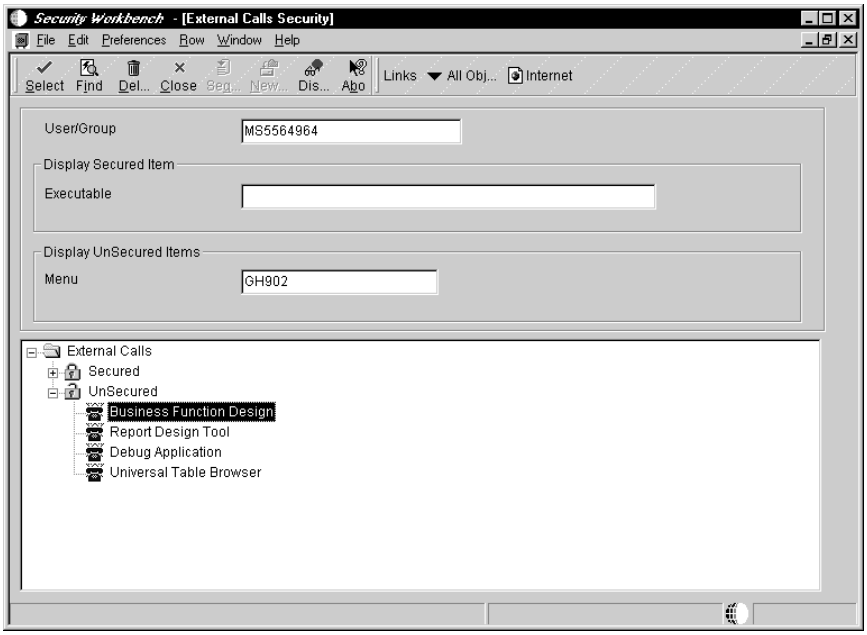
On the Security Maintenance menu (GH9052)

1. Choose Security Workbench (P00950).

   The Work with User/Group Security form appears.

2. From the Form menu, choose Set Up Security, then External Calls.

   The External Calls Security form appears.



3. Complete the following fields, and click Find:

   - User/Group

     Enter a complete user or group ID, which includes *PUBLIC, but not wildcards.

   - Path

     Enter the name of the external application, such as, debugger.exe. When you enter information into this field, OneWorld performs a search only on the indicated application.

   - Menu

     Enter the name of a menu, such as GH902, to list the external applications that reside on that menu.

Current security settings for that user or group appear under the Secured node in the tree. Expand the node to view the individual secured applications, such as debugger.exe.

4. Do one of the following to set the security:
   - Drag the external application between the Secured node and the UnSecured node to add or remove security.
   - From the Row menu, choose Secure All.

     The Secure All option secures all external applications that appear beneath the UnSecured node.

   - From the Row menu, choose Remove All.

     The Remove All option removes all external applications from beneath the Secured node and moves them beneath the UnSecured node.

Copying Security for a User or a Group

You can copy the security information for one user or group and use this information for another user or group. When you copy security, you can either overwrite the current security for the user or group, or you can add the new security information to the existing security information.

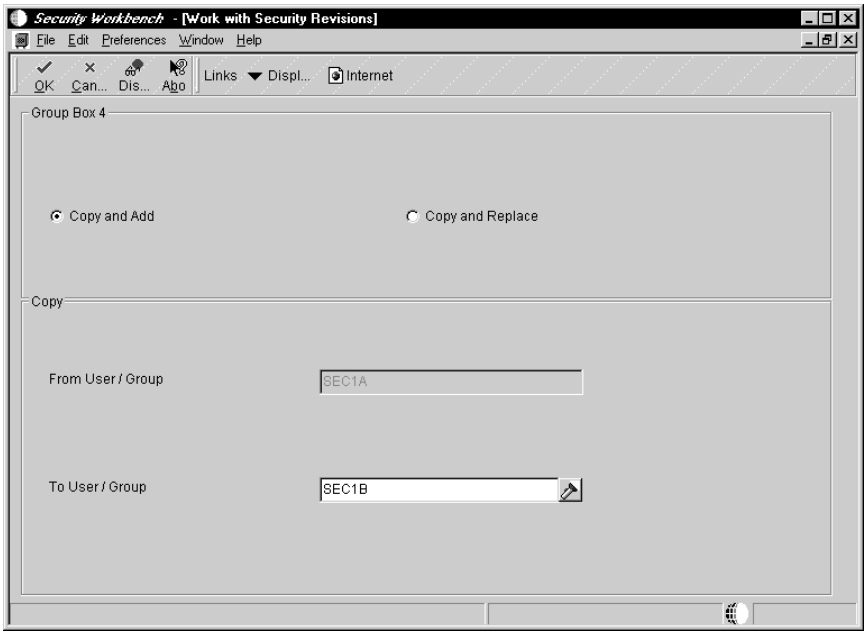
▶ To copy security for a user or a group

On the Work with User/Group Security form

1. Click Find, choose a record in the grid, then click Copy.

   Note: You can enter search criteria in the QBE line to narrow your search.

The Work with Security Revisions form appears.



2. Click one of the following options:

   - Copy and Add

     When you copy and add security settings, you do not overwrite preexisting security for user or group.

   - Copy and Replace

     When you copy and replace security settings, OneWorld deletes the security information for a user or group then copies the new security information from the selected user or group.

3. Complete the following field, then click OK:

   - To User / Group

   OneWorld saves the security information and returns you to the Work with User/Group Security form.

Deleting Security on the Work with User/Group Security Form

In addition to deleting security records on the forms specific to the security type, such as application, row, or external calls, you can delete security records on the Work with User/Group Security form.

▶ To delete security on the Work with User/Group Security form

On the Work with User/Group Security form

1. Click Find, choose a record in the grid, then click Delete.

   Note: You can enter search criteria in the QBE line to narrow your search.

2. On Confirm Delete, click OK.

   Security Workbench deletes the security record then refreshes the grid.

Understanding Signon Security

OneWorld security runs on a logic server in a dedicated internal process. You create a security table on your data server that stores information, such as:

OneWorld User	The user ID used to sign on to OneWorld.
OneWorld Password	The user's password that OneWorld validates when signing on to OneWorld.
System User and System Password	The system user and password is the actual user and password that is used to connect to all database management systems (DBMS). If the OneWorld environment includes more than one DBMS, you can create different system users and passwords for each data source.
Change Frequency	The frequency of password changes required by OneWorld.
Last Change	The date the OneWorld password was last changed.

You must define a security record for each OneWorld user either by group or individual. J.D. Edwards recommends that you map multiple OneWorld users to the same system user. For example, each user can use the same system user that OneWorld uses to connect to OneWorld database management systems. By setting up your security in this manner you can simplify database administration of users and passwords.

You can also set up unified logon with OneWorld to simplify signon security. When you setup unified logon for OneWorld, OneWorld uses Windows NT Authentication to verify OneWorld security. This verification allows OneWorld signon security to use the network signon information a user supplies when he or she signs on to Windows; OneWorld does not require the user to enter another user ID and password when he or she signs on to OneWorld.

Who Has Access to the Security Table?

Provided that you keep the system user and password secure, no users have direct access to the OneWorld Security table (F98OWSEC). The exception to this is for OneWorld system administrators, who maintain the security information. The OneWorld security server has access to the OneWorld Security table (F98OWSEC) through JDENet.

You must perform all validation and changes of OneWorld passwords through a JDENet message to the enterprise server with the security table. Upon validating a OneWorld password, the JdeNet message returns the system user and password that you enter. These are encrypted across the network. Internally, all connections to databases are done using this system password.

You should place database security, through your database management system, on the OneWorld Security table (F98OWSEC). You should also assign OneWorld object security to P98OWSEC so users cannot access the object except to enter User Password Revisions (W98OWSECD). See *Setting Up User Security.*

J.D. Edwards recommends that you do not replicate the Security table (F98OWSEC) to the workstations.

Understanding Password Encryption

You can enter the initial OneWorld signon password for each user in a number of ways:

- Manually typing it in
- Using a default password established through the signon security processing options
- Having OneWorld enter it automatically because that user already has an existing security record

When manually entering a password, or when using the processing option default password, for a new user you will be able to see the password because you are typing it in. But when you revise this record at another time OneWorld will have encrypted the password and all you will see are asterisks (****). The number of asterisks does not represent the number of characters in the password. The user security application does not “know” what the password is. The application is given a flag that indicates that a password has been entered. OneWorld stores the actual password on the security server in the OneWorld Security Table (F98OWSEC) within a binary object. OneWorld accesses the binary object when the user security application requests a change or inquiry.

What Are the Steps for Setting Up Security?

The following is a checklist that presents an overview of the steps required to set up security.

▶ To set up security (overview)

1. Ensure that the Security table (F98OWSEC) is located on your enterprise server in the system data source and that the table is mapped to the correct data source through the Object Configuration Manager.

   If your system data source resides on your enterprise server, the security table should reside in the system data source. However, if your system data source is located on the deployment server (or other servers), the security table should be moved to the server map data source for your enterprise server.

   If you have more than one logic server, J.D. Edwards recommends that you use only one as your security server.

2. From within your DBMS, place database security on this table to prevent a user from accessing the object except to enter passwords through User Password Revisions.
3. Place security on the logic server's jde.ini file. This is required because the DBMS user ID and password to the OneWorld Sign On Security table are stored in this file.
4. Create security records for individual users. This includes assigning:
   - Data source
   - System user
   - System password
   - OneWorld password
   - User Status
   - Allowed number of invalid signon attempts (optional)
   - Change frequency (optional)

   Note: If you intend to use unified logon, every user in the OneWorld security database requires a unique user ID.

5. Verify and modify the jde.ini file on your OneWorld logic server for your platform environment.

   If you use unified logon, you need to change the settings for unified logon in the [SECURITY] section in addition to the normal OneWorld [SECURITY] settings.

6. If you use unified logon with your OneWorld security, set up a unified logon server for each instance of OneWorld on each server. For example, if you have an NT server with multiple releases of OneWorld, you need a unified logon server for each release on the server.

   The unified logon server differentiates between instances of OneWorld based on the port numbers for these instances. For example, if the port number for OneWorld is 6104, the port number for the associated unified logon server is also 6104. Other instances and unified logon servers use different port numbers.

7. Verify and modify the jde.ini file that will be deployed to your server's workstation installations.
8. Require signon security for all machines.

What Is the Process Flow for OneWorld Sign On Security?

OneWorld provides sign on security with an architecture designed to provide user security for OneWorld and the logically attached database management systems. The security architecture prevents you from viewing your database or system password and therefore having the ability to bypass OneWorld applications to view and change data.

▶ Process flow for standard sign on security (overview)

1. Oneworld workstations sign on to OneWorld using their OneWorld user ID and password. These workstations can be networked or stand-alone workstations, laptop computers, or other OneWorld hosts.

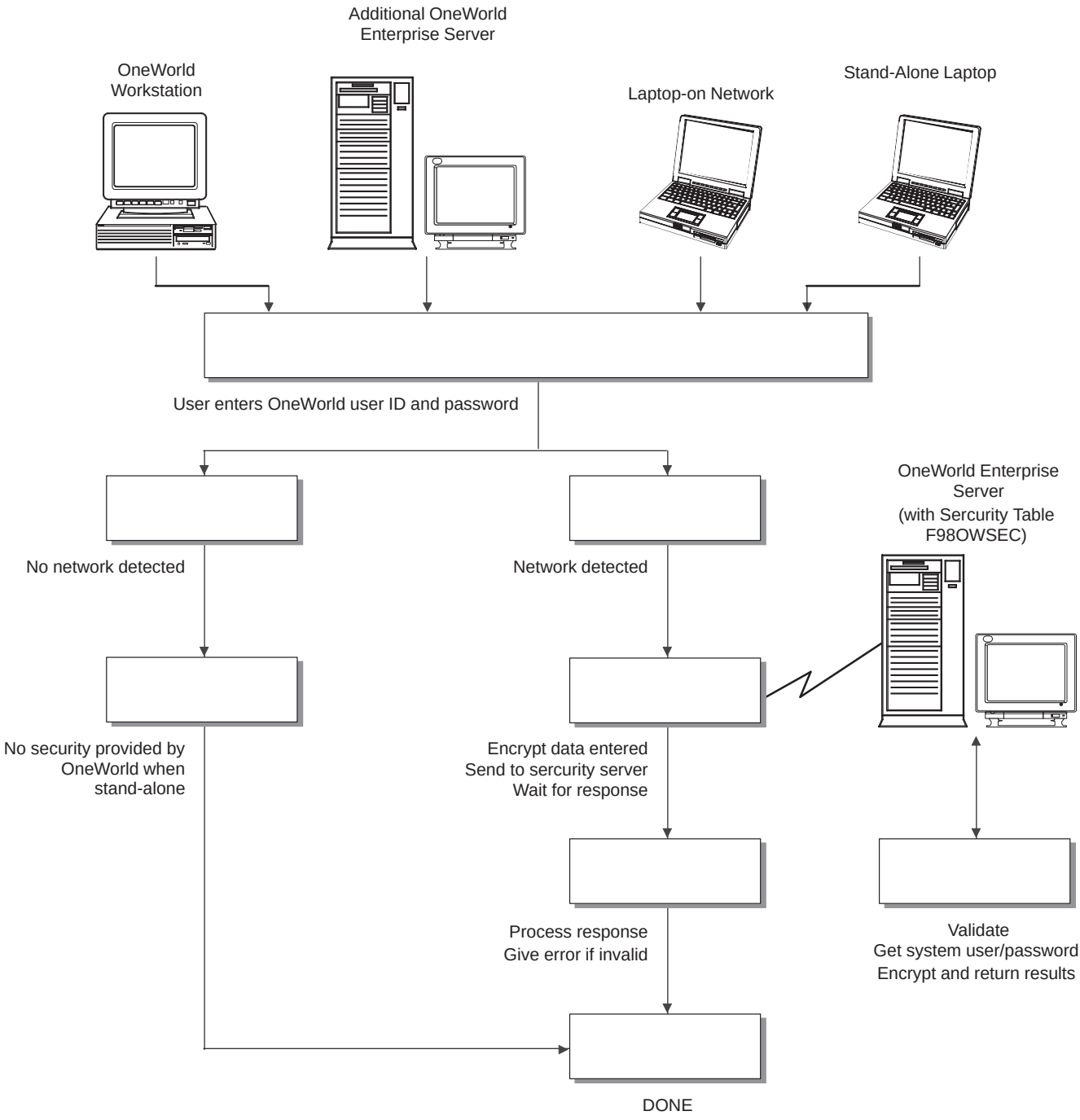
   If you enter a valid user ID and password, as validated against the local OneWorld installation, the start-up process continues.

2. As OneWorld starts up, it tries to detect an operational network environment.

   If a network is not detected, OneWorld allows local operation in a store-and-forward mode. Because the workstation or laptop computer is not connected to a network nor a OneWorld enterprise server, no validation can be performed against the security table. Therefore, security is limited to that provided by the local workstation or laptop installation.

   If a network is detected, OneWorld encrypts the password information and sends it over the network to the OneWorld enterprise server. The enterprise server checks the incoming validation request against a table of valid users and passwords. If the user ID and password information are valid, OneWorld accepts the signon values and returns the system ID and password to the logically attached database servers. This information is also encrypted on the enterprise server prior to broadcast on the network.

The following illustration provides an example of a process flow model for standard OneWorld signon security:



▶ Process flow for sign on security with unified logon (overview)

1. A user starts up OneWorld on a workstation.
2. OneWorld verifies that unified logon is active, then sends an authentication request to the unified logon server based on the domain user ID.

Note: The unified logon server is not a physical server, but rather a device that verifies signon security against the domain signon security maintained by Windows NT.

During jdesnet initialization, jdesnet activates the unified logon server thread. The unified logon server ends automatically when jdesnet ends.

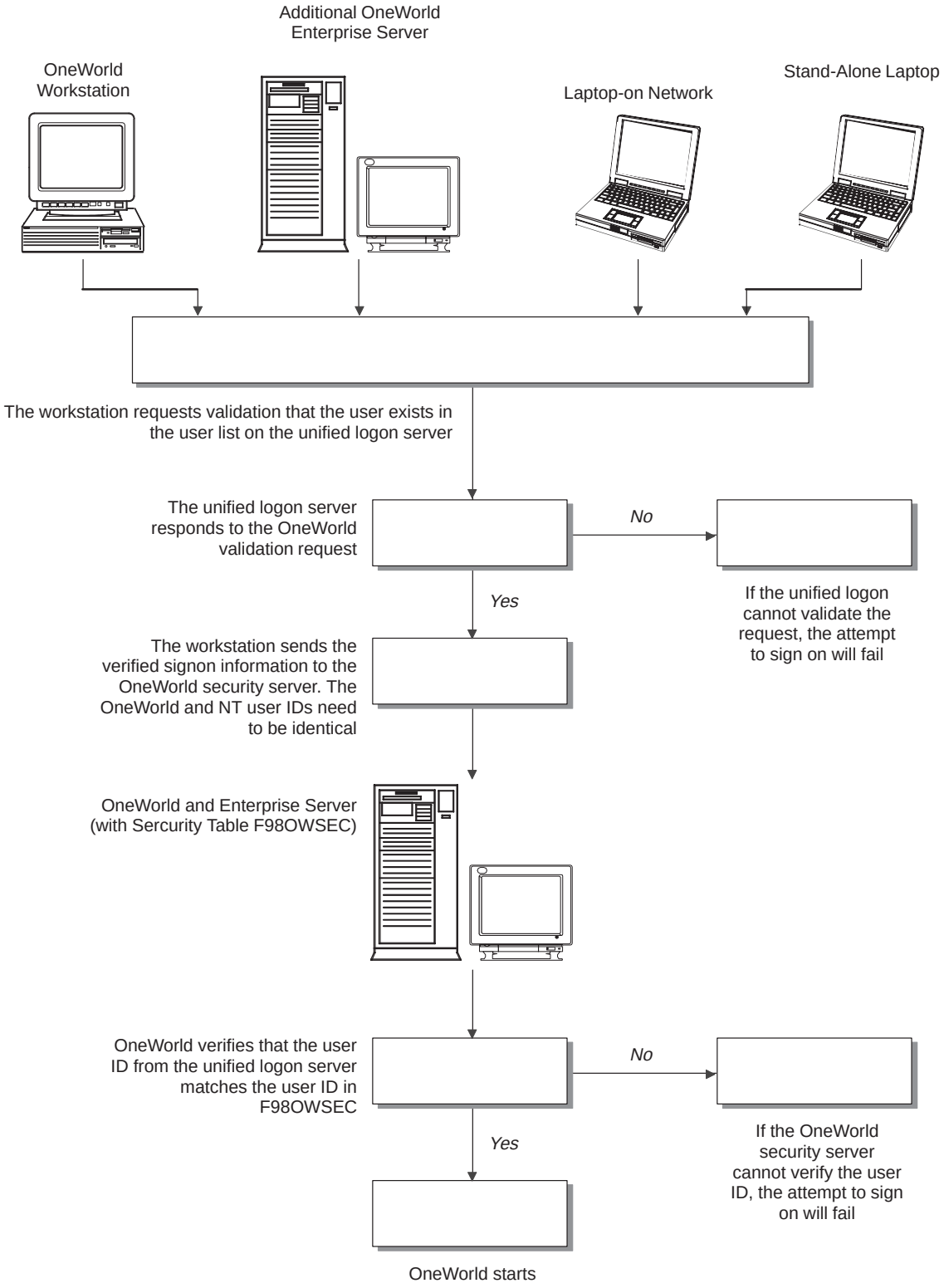
3. The unified logon server searches its user list for an entry that matches the domain user ID. When the server finds a match, the server sends a validation request to the OneWorld enterprise server.
4. The OneWorld enterprise server verifies that the response from the unified logon server matches the security information in the Security Table (F98OWSEC).
5. If the security information from the user list on the unified logon server matches the security information in F98OWSEC on the enterprise server, the start-up process continues.
6. The first time a user signs on to OneWorld with unified logon, the Environment Selection appears. The user must enter an environment in the Environment field. Click the checkbox to set the environment as the default and avoid the Environment Selection form on subsequent signon attempts.

Note: The ShowUnifiedLogon setting in the [SECURITY] section of the jde.ini file allows users to reset whether the Environment Selection form appears at signon. This feature allows users to change the environment at a later time. The following example describes the jde.ini file setting:

```
[SECURITY]
ShowUnifiedLogon=0 or 1
```

Value	Description
0	A value of 0 for ShowUnifiedLogon disables the Environment Selection form. When you click the checkbox on the Environment Selection form to set a default environment, you set this value to 0.
1	A value of 1 for ShowUnifiedLogon enables the Environment Selection form. When a user signs on to OneWorld, the Environment Selection form appears to allow the user to choose an environment. This is the default setting for ShowUnifiedLogon

The following illustration provides an example of a process flow model for OneWorld signon security with unified logon:



Working with User Security

Use the User Security (P98OWSEC) application to create, test, and change user security for OneWorld and the logically attached database management systems. The security architecture prevents you from viewing your database or system password and therefore having the ability to bypass OneWorld applications to view and change data. J.D. Edwards uses an encryption algorithm to ensure that applications other than OneWorld security cannot access passwords transmitted across the OneWorld network.

You can also setup up a unified logon server for a OneWorld server. The unified logon server enables OneWorld to use the domain logon information to determine user security for OneWorld; in a OneWorld unified logon scenario, a user only needs to enter a user ID and a password at network logon.

This topic contains the following:

- Creating and revising user security
- Adding and revising data sources for user security
- Changing jde.ini file for user security
- Running a security analyzer report
- Setting up unified logon

Creating and Revising User Security

You can create security records one at a time for each of your users, you can set security for a group of users, or you can set security for all users. You should use this feature to initially set up user security. The User Security application provides a copy function to simplify the creation of security records for individual users.

Tip: J.D. Edwards recommends that you create a “model” user with security information that you can copy to create other OneWorld users. Typically, users within a specific group use similar security information.

You should keep user security simple. Managing OneWorld user IDs and system (database) user IDs can become very complicated quickly. The simplest way to set up user security is to have all OneWorld data sources share the same system user ID and password by leaving the data source field blank when you initially create user security records for users or groups on the Security Revisions form. When you leave the data source field blank, OneWorld automatically enters DEFAULT in the field. The DEFAULT data source allows you to create one security record for all users. Each time a user accesses a table through a OneWorld application, OneWorld searches for a security record for that user and that specific data source where the table resides. If OneWorld does not find a specific record, then OneWorld uses the default data source, which would be the security record you created with the DEFAULT data source field.

Although you should try to maintain as few system user IDs as you can, occasions will arise that require you to set up database security in addition to the OneWorld object and user security for specific users and specific tables. For example, you might need to create system users with additional authority than the normal system user.

Before You Begin

❑ Create user profiles. See *User Profiles.*

Complete the following tasks:

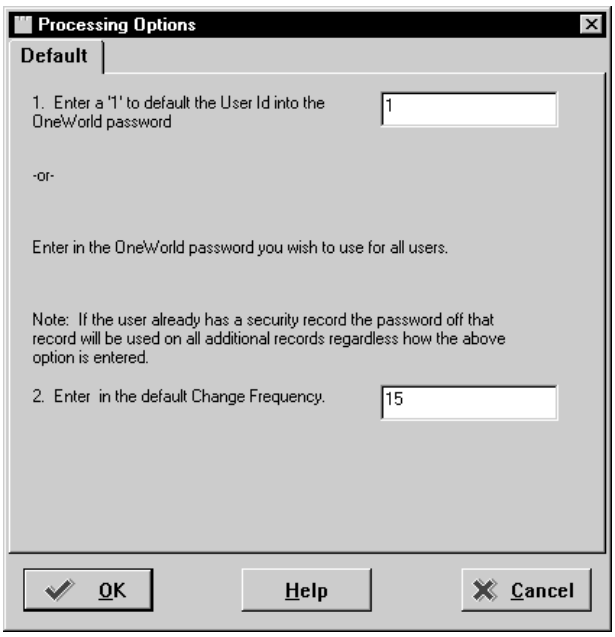
- Set processing options for user security
- Create user security
- Copy user security
- Revise user and group security
- Delete user security
- Revise all user security
- Change a signon password (administrators only)

  Perform this task to reset forgotten passwords for users.

- Review security history
- Require user security

▶ To set processing options for user security

The User Security (P98OWSEC) application has the following processing options that you can use to set a default password when creating user security for users or groups and to set a default change frequency for the password:



- Option 1: Enter one of the following:
  - "1" to use the User ID as the password when creating user security
  - Enter a password that you want to use as the default for all users when you create user security

  For all existing user security records, the passwords remain as they are, and any new passwords needed will reflect their current password rather than the password entered into the processing option.

- Option 2: Enter a default (in days) for how often a user needs to change his or her user password.

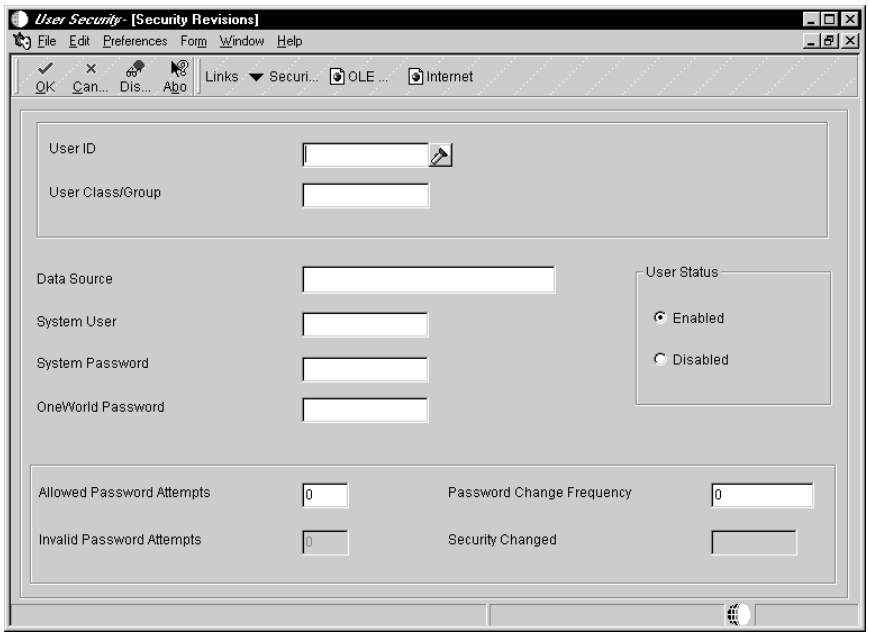
▶ To create user security

Before you can create user security, you must first set up all user and group records in the Address Book Revisions (P01012) application, create a user profile in the User Profiles (P0092) application, and attach the proper Address Book record to the user or group profile. You should also review and set the appropriate processing options before using the User Security (P98OWSEC) application for the first time. See *To Set Processing Options for Signon Security.*

Tip: J.D. Edwards recommends that you create a "model" user with security information that you can copy to create other OneWorld users. Typically, users within a specific group use similar security information.

On the Security Maintenance menu (GH9052), choose User Security (P98OWSEC).

1. On the Work with User Security form, click Add.



2. On Security Revisions, complete one of the following fields:

   - User ID

     If you enter a user ID in this field that already exists, you can modify data source information for the user. OneWorld disables all other fields and options for the user ID.

   - User Class/Group

     If you enter a user class or group in this field that already exists, you will overwrite the security record for the user class or group when you enter information on the form.

   Note: When you type information in one of these fields, OneWorld disables the other field. For example, if you type GROUP1 in the User Class/Group field, the User ID field becomes grey and unavailable for data entry.

3. Complete the following fields:

   - Data Source

     If you leave this field blank, you will set security for all data sources. DEFAULT appears in the Data Source field when you tab out of the field. See *Data Sources* in the *CNC Implementation Guide* for data source usage.

   - System User
   - System Password
   - OneWorld Password

At a minimum, J.D. Edwards recommends you complete the System user and System Password fields.

If you create records by group or for all users at one time, the OneWorld Password field populates according to the processing option you choose. Refer to *Understanding Processing Options for Signon Security.*

4. Under User Status, choose one of the following options:

   - Enabled

     With User Status enabled, security allows the user to signon to OneWorld. This option is the default setting when you create user security.

   - Disabled

     With User Status disabled, security prohibits the user from signing on to OneWorld.

   Note: If a user commits a security violation, such as exceeding the maximum number of allowed password attempts, OneWorld automatically sets the value for User Status to Disabled. The system administrator must access the user security record for the user and set User Status to Enabled before the user can sign on to OneWorld. Also, the system administrator can access Administrative Password Revisions to reset the password of the user, which also restores a user profile to the status of enabled. Refer to *Changing OneWorld User Signon Passwords*.

5. If you want to set limits on the OneWorld password for users, complete the following fields:

   - Allowed Password Attempts

     Enter the number of invalid password attempts allowed before OneWorld disables access for the user.

   - Password Change Frequency

     Enter the number of days until OneWorld next requires the user to change his or her password.

6. Click OK to save the current user security information.

   When you create group security, OneWorld creates the user security IDs to the various data sources for all the user attached to the group with passwords based on processing option 1. Users can now sign on and change their passwords by choosing the Change Password from the View User Option menu selection. See *Changing User OneWorld Signon Passwords.*

7. When you finish, click Cancel.

Field	Explanation
User ID	For World, the IBM-defined user profile. For OneWorld, the identification code for a user profile.
User Class/Group	A profile used to classify users into groups for security purposes. Some rules for creating a User Class/Group are as follows: • The 'Class/Group' profile must begin with * so that it does not conflict with any System profiles. • The 'User Class/Group' field must be blank when entering a new group profile.
All Users	When this box is checked OneWorld security will be added for all User setup in the Library List - User table (F0092) that do not have security currently setup for them in the OneWorld Security table (F98OWSEC).
Data Source	The data source to which the user is secured unless a valid system use rand system password exists in the record. Blank indicates all data sources.
System User	Identifies the actual user that OneWorld will use to connect to the database management systems (DBMS) you have specified as the data source. The system user you define here must exactly match the user value defined in the DBMS.
System Password	Identifies the password that OneWorld will use to connect to the database management systems (DBMS) you have specified as the data source. This password is associated with the System User. The system password you define here must exactly match the password value defined in the DBMS. There is no password expiration on the system password; only the OneWorld password.
OneWorld Password	Identifies the user password that OneWorld will use to validate when signing on to OneWorld. This is the only field that end users are allowed to change through User Password Revisions. J.D. Edwards recommends that when you set up users for the first time, you set their OneWorld password to a value equal to their OneWorld User ID.
Allowed Password Attempts	The number of signon attempts a user can make before that user profile is disabled.
Change Frequency	Identifies the number of days before OneWorld requires that a user change their OneWorld password.

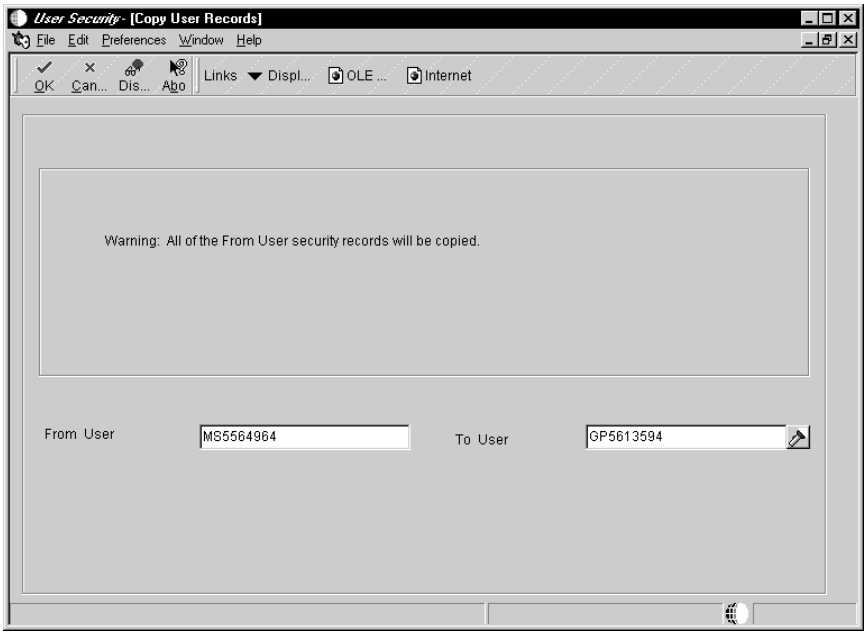
▶ To copy user security

On the Work with User Security form

1. Find the user, choose the appropriate record in the tree structure, then click Copy.

   Tip: J.D. Edwards recommends that you create a “model” user with security information that you can copy to create other OneWorld users. Typically, users within a specific group use similar security information.

   The Copy User Records form appears.



2. Complete the following field, then click OK:

   - To User

     Type a valid user in this field.

     Note: A user profile must already exist for a user before you can create user security records for the user. Also, when you copy security records to a user, security records must not already exist for the user. If you try to copy user security to a user with existent user security records, you will receive an error message.

▶ To revise user and group security

On the Security Maintenance menu (GH9052)

1. Choose User Security (P98OWSEC).

2. On Work with User Security, complete either of the following:
   - User ID
   - Group
3. Choose the appropriate record in the tree structure, then from the Row menu, choose Revise Security.
4. On Security Detail Revisions, complete the following fields as necessary:

   Note: For a group, choose the appropriate option from the Change box to enable each field.

   - User Status

     Under User Status, you can enable or disable a user profile.

   - Password Change Frequency
   - Allowed Password Attempts

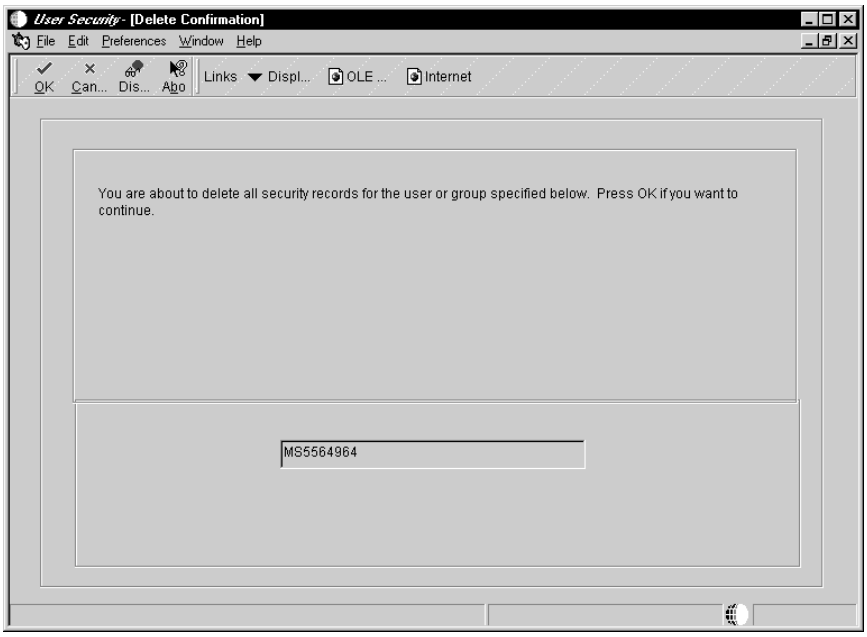
▶ To delete user security

On the Work with User Security form

1. Find the user or group, choose the appropriate record in the tree structure, then click Delete.

   Note: If you choose a record from the grid and click Delete, you will remove the data source for the user, not user security.

   The Delete Confirmation form appears.



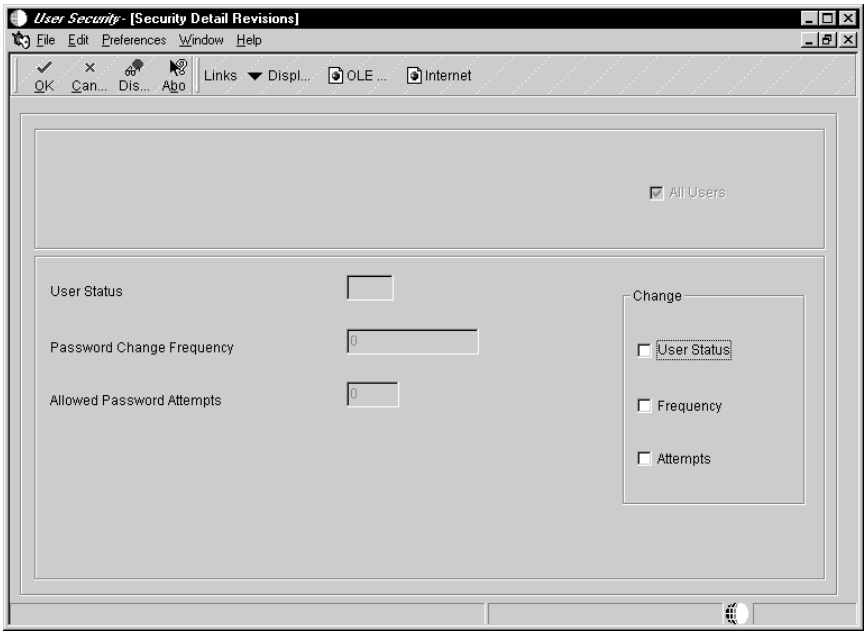
2. Click OK to delete all user security records for the user or group.

To revise all user security

On the Work with User Security form

1. From the Form menu, choose Revise All.

   The Security Detail Revisions form appears.



2. In the Change box, click any of the following options to enable the related field:
   - User Status
   - Frequency
   - Attempts
3. Complete any of the following fields, then click OK:
   - User Status

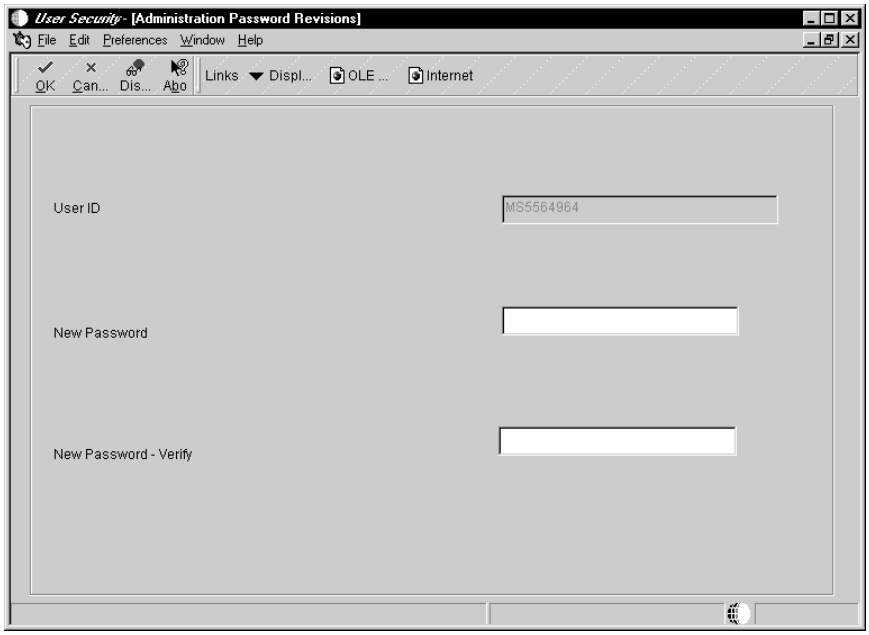
     This field allows you to enable or disable user profiles.

   - Password Change Frequency
   - Allowed Password Attempts

▶ To change a signon password (administrators only)

On the Security Maintenance menu (GH9052), choose Administrative Password Revisions (P98OWSEC).

Note: You can also access Administrative Password Revisions from the User Security application. On Work with User Security, find the user, choose the user in the tree structure, then choose Password Revisions from the Row menu.



On the Administration Password Revisions form, complete the following fields and click OK:

- User ID
- New Password

  On this form, OneWorld does not restrict your password choices. Any password is valid.

- New Password - Verify

Field	Explanation
User ID	Identifies the OneWorld user ID. By default OneWorld populates this field with the value of the currently logged on user.
New Password	Identifies new value for the user password that OneWorld will use to validate when signing on to OneWorld. This value should be different that the Old Password value. The value you specify in this field will become effective the next time you sign on to OneWorld.
New Password Verify	Identifies a duplicate of the value you specified in the New Password field. The value you enter here must exactly match the value you enter in the New Password field.

See Also

- *Changing Your User Options* in the *OneWorld Foundation* guide for details on how to change a password at the end user level

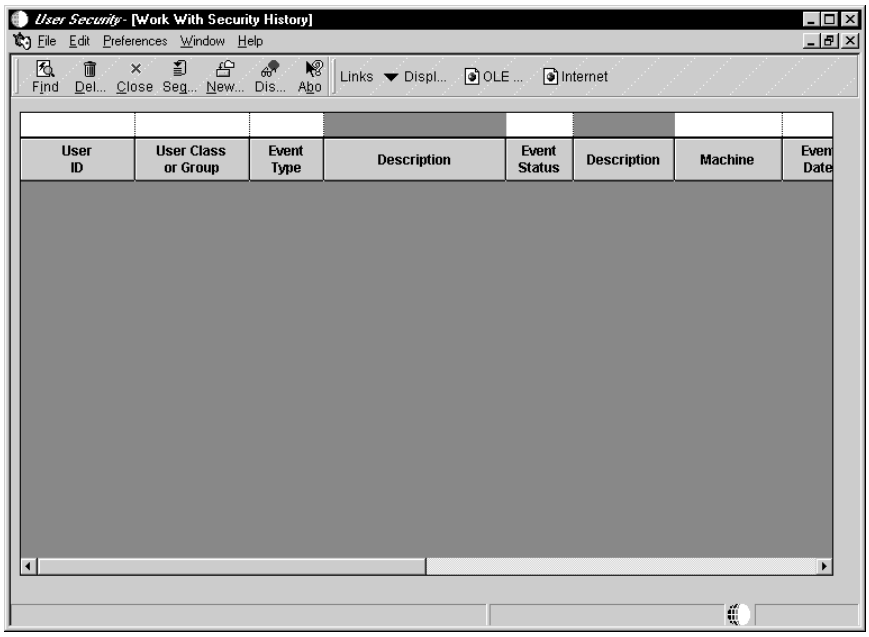
▶ To review security history

The [SECURITY] section in your server jde.ini must include the following setting for OneWorld to record security history:

```
[SECURITY]
History=1
```

On the Security Maintenance menu (GH9052)

1. Choose User Security (P98OWSEC).
2. On Work with User Security, from the Form menu, choose Security History.



3. On Work with Security History, click Find.

   If you know the specific user or group, type the name of the user or group in the appropriate field on the query by example line and press Enter. You can also search for specific information for all users. For example, to see the users who were deleted on a given day, you can search on Event Type 06 (Delete User) and specific Event Date.

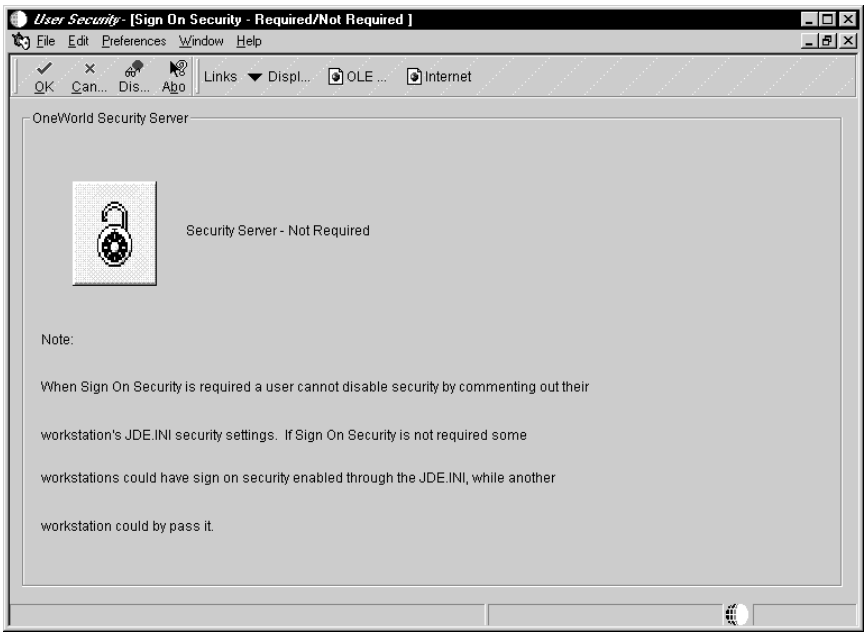
4. Review the security history records that appear in the detail area.

▶ To require signon security

Use this feature to require all machines to use OneWorld signon security. This procedure only enables mandatory security for the environment that you are signed onto when making this change.

On the Security Maintenance menu (GH9052)

1. Choose User Security (P98OWSEC).
2. On the Work with User Security form, choose Req/Not Req from the Form menu.



3. On the Sign On Security - Required/Not Required form, click the lock icon to change the Security Server to “required” or “not required.”

   Remember, if you set up your security as Not Required, and have security turned on through the jde.ini file on the enterprise server, users with jde.ini files that comment out signon security will still not have access to any data sources without knowledge of the system user ID and password. When attempting to access a table in a secured data source, users will receive a database password entry form. If system user IDs and passwords are confidential, no one will be able to access to your secured tables.

```
maxNumberOfProcesses=1
```

Adding and Revising Data Sources for User Security

Add data sources to user and group records in user security to authorize users and groups to access OneWorld databases. You can revise the system user and system password for existent data sources.

Complete the following tasks:

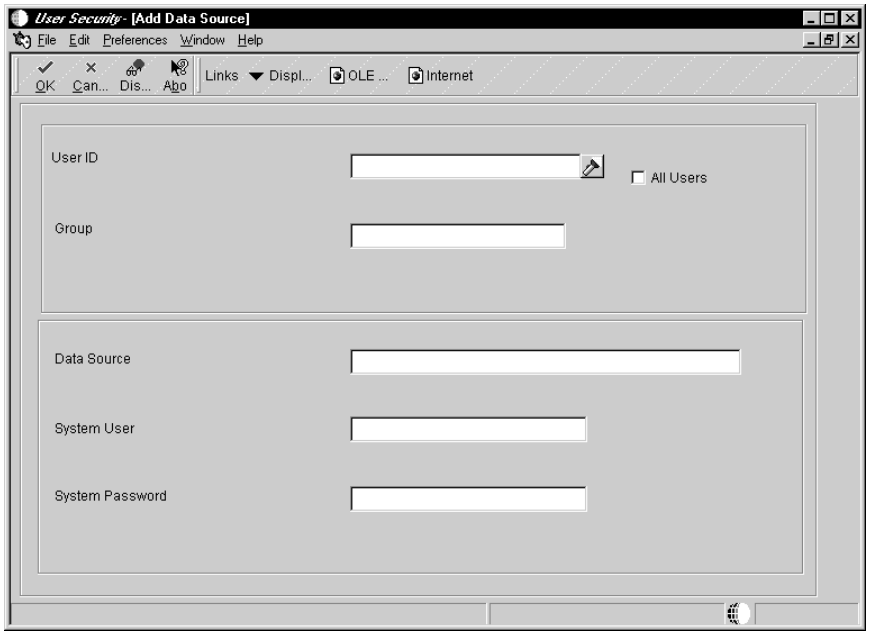
- Add a data source to a user, a group, or all users
- Revise a data source for a user, a group, or all users
- Remove a data source for a user, a group, or all users

To add a data source to a user, a group or all users

On the Security Maintenance menu (GH9052)

1. Choose User Security (P98OWSEC).
2. On the Work with User Security form, from the Form menu, choose Add Data Source.

   The Add Data Source form appears.



3. Complete one of the following fields or options:
   - User ID

     Complete this field to add a data source to a specific user.
   - Group

     Complete this field to add a data source to a specific group.
   - All Users

     Turn on this option to add a data source to all users.

4. Complete the following fields and click OK:

   - Data Source

     Leave this field blank to set the data source information for all data sources. When you leave this field blank, OneWorld automatically enters DEFAULT in the field.

   - System User
   - System Password

▶ To revise a data source for a user, a group, or all users

On the Work with User Security form

1. Complete the following field and click Find:

   - Data Source

2. Choose the appropriate record in the tree structure, then from the Row menu, choose Revise Data Source.

   You can choose a user, a group, or a data source from the tree. If you choose a data source, you will affect all users with any changes you make.

   The Data Source Revisions form appears. If you chose a specific user or group, this form displays the user ID or the group name with the data source information. If you chose only the data source, this form displays the All Users option with the data source information.

3. Complete the following fields, and click OK:

   - System User
   - System Password

   The information you enter in these fields is necessary to access databases within OneWorld. Depending on what you chose from the tree on Work with User Security, this information will apply to a specific user, a specific group, or all users.

▶ To remove a data source for a user, a group, or all users

On the Work with User Security form

1. Complete the following field and click Find:

   - Data Source

2. Choose the appropriate record in the tree structure, and click Delete.

Note: For a user, you can also highlight a grid row for the user and click Delete.

The Remove Data Source form appears. If you chose a data source for a specific user or group, this form displays the user ID or the group name with the data source name. If you chose only the data source, this form displays only the data source name.

Caution: If you performed your search by data source without including a specific user or group, when you click OK on Remove Data Source, you remove the data source for ALL users.

3. Click OK to remove the data source.

Changing the jde.ini File for User Security

You must modify the enterprise server and the workstation jde.ini files to enable and synchronize security settings between the enterprise server and workstation.

Note: For OneWorld workstations, enable security by changing settings in the workstation jde.ini file. You should make these changes on the deployment server-resident jde.ini file that will be delivered to the workstation through a package installation.

Complete the following tasks:

- Change the workstation jde.ini file for security
- Change the enterprise server jde.ini file for security

▶ To change the workstation jde.ini file for security

1. Locate the jde.ini file that will be sent to the workstation as part of a package installation. This file is located on the OneWorld deployment server in the release share path:

   \\B*xxx*\CLIENT\MISC\jde.ini

   where *xxx* is the installed release level of OneWorld. For example, B731 or B732.

2. Using a text editor such as Notepad, view the jde.ini file to ensure accuracy of the following setting:

```
[SECURITY]
SecurityServer=Enterprise Server Name
DefaultEnvironment=Default Environment
```

See the following table for an explanation of the variable values.

Setting	Value
Security Server	A valid value is the name of your OneWorld enterprise server. This value must match between the workstation and the enterprise server in order for workstations to sign on and run batch reports on the enterprise server.
DefaultEnvironment	Identifies any valid environment. If no value is specified here, then security is not enabled for that workstation.

Setting Auxiliary Security Servers in the Workstation JDE.INI

You can set within the [SECURITY] section of the workstation jde.ini file one to ten (1-10) auxiliary security servers, as shown:

```
[SECURITY]
NumServers=Numeric Value
SecurityServer=Enterprise Server Name (primary)
SecurityServer2=Enterprise Server Name (auxiliary)
SecurityServer3=Enterprise Server Name (auxiliary)
```

Setting	Value
NumServers	This numeric value is the total number of security servers (primary and auxiliary) you set under the [SECURITY] section of the jde.ini file. For example, if you set one primary and four auxiliary servers, the NumServers value would be 5. You can set NumServers to any value 1-10. If you do not include the NumServers setting, OneWorld assumes that you have only one server.
SecurityServer **SecurityServer*n***	A valid value is the name of a OneWorld enterprise server. Your primary and auxiliary security server names must all be valid enterprise servers. The values must match between the workstation and enterprise servers in order for workstations to logn onto and run batch reports from the enterprise server. The variable value *x* can be an number between 1 and 10. This number defines the auxiliary security server.

Changing the Time-Out Value Due to Security Server Communication Error

You might need to change a setting to your workstation jde.ini file if you are receiving an error such as: Failure to Communicate with Security Server.

Change the following section:

```
[JDENET]
connectTimeout=30
```

▶ To change the enterprise server jde.ini file for security

Verify your server jde.ini file settings as shown in the task. You use these settings to specify OneWorld internal security parameters, valid users and passwords, environments, and data sources. At the end of this task are additional topics concerning the server jde.ini file.

1. Locate your enterprise server's jde.ini file.
2. Using an ASCII editor such as Notepad, view the jde.ini file to ensure accuracy of the following settings:

```
[JDENET_KERNEL_DEF4]
dispatchDLLName=name of host dll
dispatchDLLFunction=JDEK_DispatchSecurity
maxNumberOfProcesses=1
beginningMsgTypeRange=551
endingMsgTypeRange=580
newProcessThresholdRequests=0

[SECURITY]
SecurityServer=Enterprise Server Name
User=user ID
Password=user password
ServerPswdFile=TRUE/FALSE
DefaultEnvironment=default environment
```

See the following table for an explanation of the variable values.

Setting	Value
dispatchDLLName	Valid values for enterprise server host platforms are: • HP9000 libjdeknet.sl • RS/6000 libjdeknet.sl • Windows NT (Intel) jdekrnl.dll • Windows NT (Compaq AlphaServer) jdekrnl.dll • AS/400 JDEKRNL For UNIX platforms, values are case-sensitive.
SecurityServer	A valid value is the name of your OneWorld enterprise server. This value must match between the workstation and the enterprise server in order for workstations to run batch reports on the enterprise server.
User	The ID of a user with access to the OneWorld Sign On Security table (F98OWSEC). This is the ID used to connect to the DBMS. Therefore, this value is must exactly match that of the target DBMS.
Password	The password for the user ID with access to the OneWorld Sign On Security table (F98OWSEC). This is the password used to connect to the DBMS. Therefore, this value is must exactly match that of the target DBMS.

Setting	Value
ServerPswdFile	This parameter is valid on for OneWorld servers operating under UNIX operating systems. The setting of this parameter determines whether OneWorld uses special password handling for batch reports running on the server. Set the value to TRUE to instruct OneWorld to enable special handling of passwords. Set the value to FALSE to disable special handling. When OneWorld runs a batch report on the server, it runs the report using a string of line commands and parameters that includes the “user password”. Under UNIX operating systems, it is possible to use the ps command to query the status of a job and view the parameters that were used to start the process. As a security measure, you can enable special handling by OneWorld. When enabled, OneWorld does not include the “user password” in the parameter list for a batch process. Instead, it includes the name of a file that contains the “user password”. This file is destroyed as soon as the batch report reads the password.
DefaultEnvironment	Valid values are the name of a valid environment for accessing the security table. For example, PRODB714.

Setting Auxiliary Security Servers in the Server JDE.INI

You can set within the [SECURITY] section of the server jde.ini file one to ten auxiliary security servers. You set multiple auxiliary security servers to establish levels of default servers. For example, if a machine cannot access a given security server, the machine tries the next security server defined in the [SECURITY] section. The settings for auxiliary security servers are as follows:

```
[SECURITY]
NumServers=Numeric Value
SecurityServer=Enterprise Server Name (primary)
SecurityServer2=Enterprise Server Name (auxiliary)
SecurityServer3=Enterprise Server Name (auxiliary)
```

Setting	Value
NumServers	This numeric value is the total number of security servers (primary and auxiliary) you set under the [SECURITY] section of the jde.ini file. For example, if you set one primary and four auxiliary servers, the NumServers value would be 5. You can set NumServers to any value 1-10. If you do not include the NumServers setting, OneWorld assumes that you have only one server.
SecurityServer **SecurityServer*x***	A valid value is the name of a OneWorld enterprise server. Your primary and auxiliary security server names must all be valid enterprise servers. The values must match between the workstation and enterprise servers in order for workstations to log onto and run batch reports from the enterprise server. The variable value *x* can be an number between 1 and 10. This number defines the auxiliary security server.

Verifying Security Processes in the Server JDE.INI

It is recommended that you define only one process for the security network. You can set multiple processes, but they will most likely not be necessary. Under the [JDENET_KERNEL_DEF4] section of your server jde.ini file, verify the following:

Running a Security Analyzer Report

This process generates two separate reports that provide you with an analysis of OneWorld security. The first report (R98OWSECA) is organized and sorted by data source. A blank data source means that security for the System User ID is applicable to all data sources. The Security Analyzer by Data Source report (R98OWSECA) is based on data that it reads from the OneWorld Security table (F98OWSEC).

The second report (R98OWSECB) is organized by user or group. The Security Analyzer by User or Group report (R98OWSECB) is also based on data that it reads from the OneWorld Security table (F98OWSEC).

Complete the following tasks:

- Run the Security Analyzer by Data Source Report (R98OWSECA)
- Run the Security Analyzer by User or Group Report (R98OWSECB)

Running the Security Analyzer by Data Source Report (R98OWSECA)

This report presents security analysis information for each data source, each user ID, and each group. The report is sorted by data source and then by user ID. The following columnar data is displayed:

- Data Source

  Identifies the data source to which the user is secured. Blank indicates all data sources.

- User ID

  Identifies the user ID.

- User Group

  Identifies the identification code for a user profile.

- System User ID

  Identifies the actual user that OneWorld uses to connect to the database management system (DBMS) you have specified as the data source. The system user defined here must exactly match the user value defined in the DBMS.

- Change Frequency

  Indicates the number of days before OneWorld requires that a user change their OneWorld password. This can be set by individual user ID or by group.

- Source Password Changed

  Indicates the date that a user's password was last changed.

- Invalid Sign-Ons

  Indicates the number of invalid signon attempts by a user. If the retry count value exceeds the number of allowed attempts, the user profile is disabled.

- Allowed Attempts

  Indicates the number of sign-on attempts a user can make before that user profile is disabled.

- User Status

  Indicates whether the user can sign on to OneWorld. Valid values are 01 (enabled) and 02 (disabled).

- Status

  A description code that displays the status of the User Status field. Valid values are enabled (01) and disabled (02).

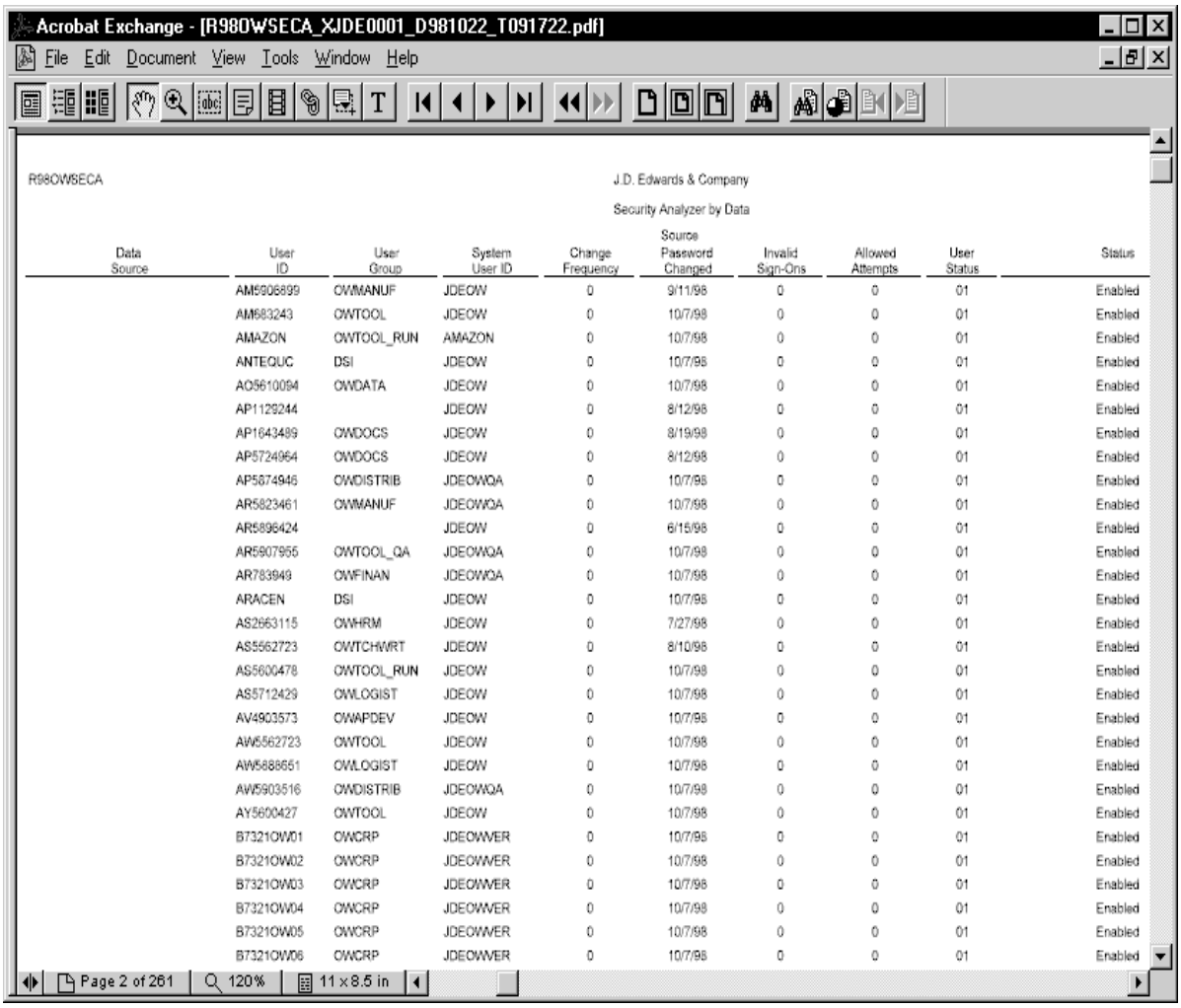
▶ To run the Security Analyzer by Data Source Report (R98OWSECA)

On the Security Maintenance menu (GH9052)

1. Choose Security Analyzer by Data Source (R98OWSECA).
2. On the Work With Batch Versions form, choose a version, and click Select. The J.D. Edwards default version one (XJDE0001) creates a report for all user IDs for all data sources.
3. On the Versions Prompting form, click Submit.

Example: Security Analyzer by Data Source Report (R98OWSECA)

This example shows an excerpt from the Security Analyzer by Data Source Report report (R98OWSECA).



Data Source	User ID	User Group	System User ID	Change Frequency	Source Password Changed	Invalid Sign-Ons	Allowed Attempts	User Status	Status
	AM5906899	OWMANUF	JDEOW	0	9/11/98	0	0	01	Enabled
	AM683243	OWTOOL	JDEOW	0	10/7/98	0	0	01	Enabled
	AMAZON	OWTOOL_RUN	AMAZON	0	10/7/98	0	0	01	Enabled
	ANTEQUC	DSI	JDEOW	0	10/7/98	0	0	01	Enabled
	AO5610094	OWDATA	JDEOW	0	10/7/98	0	0	01	Enabled
	AP1129244		JDEOW	0	8/12/98	0	0	01	Enabled
	AP1643489	OWDOCS	JDEOW	0	8/19/98	0	0	01	Enabled
	AP5724964	OWDOCS	JDEOW	0	8/12/98	0	0	01	Enabled
	AP5874946	OWDISTRIB	JDEOWQA	0	10/7/98	0	0	01	Enabled
	AR5823461	OWMANUF	JDEOWQA	0	10/7/98	0	0	01	Enabled
	AR5896424		JDEOW	0	6/15/98	0	0	01	Enabled
	AR5907955	OWTOOL_QA	JDEOWQA	0	10/7/98	0	0	01	Enabled
	AR783949	OWFINAN	JDEOWQA	0	10/7/98	0	0	01	Enabled
	ARACEN	DSI	JDEOW	0	10/7/98	0	0	01	Enabled
	AS2663115	OWHRM	JDEOW	0	7/27/98	0	0	01	Enabled
	AS5562723	OWTCHWRT	JDEOW	0	8/10/98	0	0	01	Enabled
	AS5600478	OWTOOL_RUN	JDEOW	0	10/7/98	0	0	01	Enabled
	AS5712429	OWLOGIST	JDEOW	0	10/7/98	0	0	01	Enabled
	AV4903573	OWAPDEV	JDEOW	0	10/7/98	0	0	01	Enabled
	AW5562723	OWTOOL	JDEOW	0	10/7/98	0	0	01	Enabled
	AW5888651	OWLOGIST	JDEOW	0	10/7/98	0	0	01	Enabled
	AW5903516	OWDISTRIB	JDEOWQA	0	10/7/98	0	0	01	Enabled
	AY5600427	OWTOOL	JDEOW	0	10/7/98	0	0	01	Enabled
	B7321OW01	OWCRP	JDEOWVER	0	10/7/98	0	0	01	Enabled
	B7321OW02	OWCRP	JDEOWVER	0	10/7/98	0	0	01	Enabled
	B7321OW03	OWCRP	JDEOWVER	0	10/7/98	0	0	01	Enabled
	B7321OW04	OWCRP	JDEOWVER	0	10/7/98	0	0	01	Enabled
	B7321OW05	OWCRP	JDEOWVER	0	10/7/98	0	0	01	Enabled
	B7321OW06	OWCRP	JDEOWVER	0	10/7/98	0	0	01	Enabled

Running the Security Analyzer by User or Group Report (R98OWSECB)

This presents security analysis information for each user ID, each group, and each data source. The report is sorted either by user ID or user group, as controlled by a processing option. The following columnar data is displayed:

- User ID

  Identifies the user ID.

- User Group

  Identifies the identification code for a user profile.

- Change Frequency

  Indicates the number of days before OneWorld requires that a user change their OneWorld password. This can be set by individual user ID or by group.

- Data Source

  Identifies the data source to which the user is secured. Blank indicates all data sources.

- System User ID

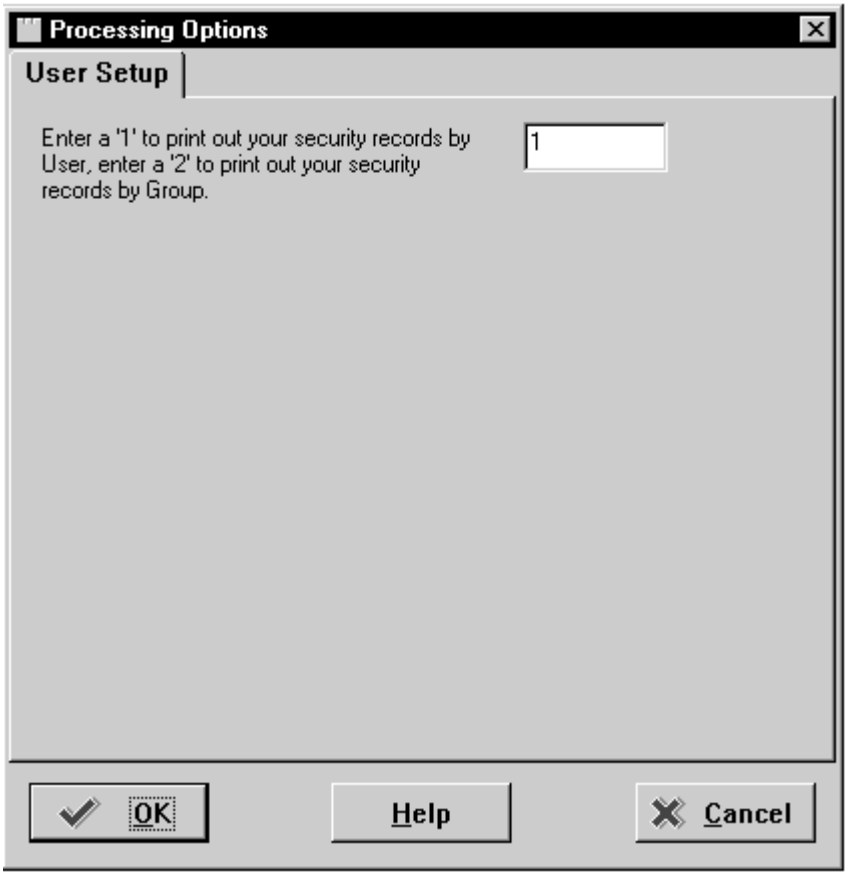
  Identifies the actual user that OneWorld uses to connect to the database management system (DBMS) you have specified as the data source. The system user defined here must exactly match the user value defined in the DBMS.

▶ To run the Security Analyzer by User or Group Report (R98OWSECB)

On the Security Maintenance menu (GH9052)

1. Choose Security Analyzer by User or Group (R98OWSECB).
2. On the Work With Batch Versions form, choose a version, and click Select. The J.D. Edwards default version one (XJDE0001) creates a report for all user IDs for all data sources.

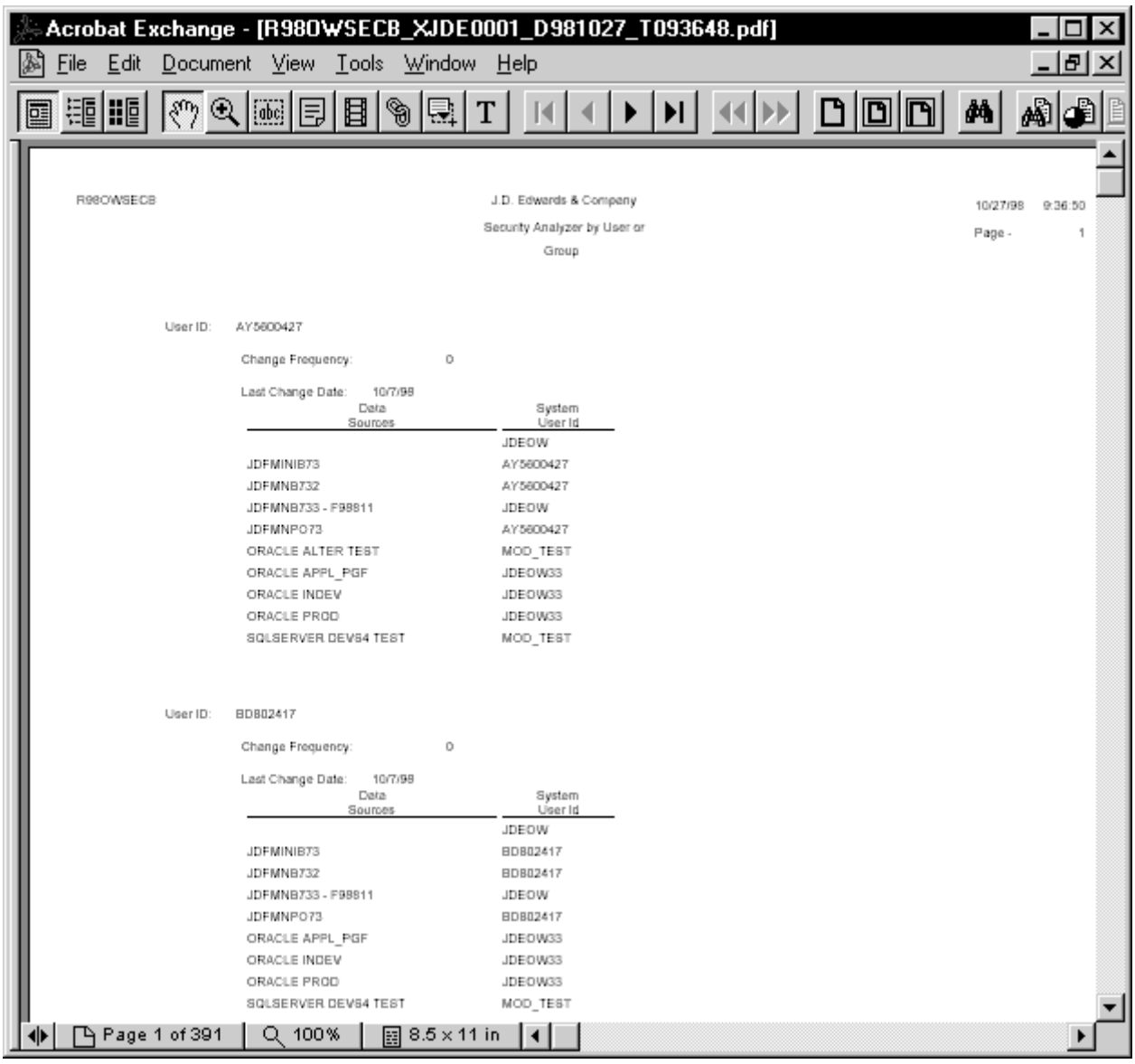
   By default, the XJDE0001 version has the processing option for this report set to 1. This option causes the report to be generated by User ID. To generate a report by Group, you can prompt for processing options and change the value to 2.



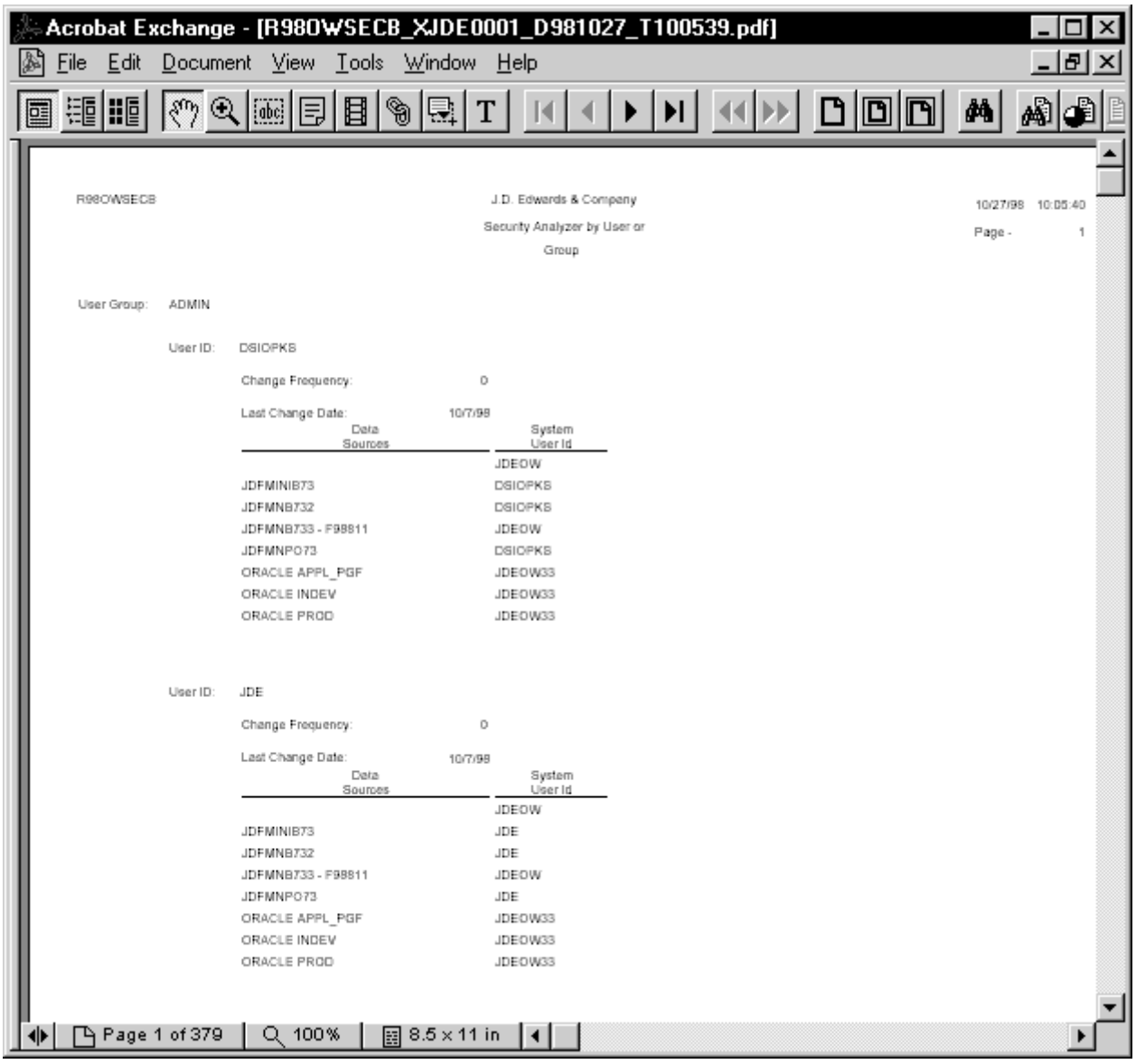
3. On the Versions Prompting form, click Submit.

Example: Security Analyzer by User or Group Report (R98OWSECA)

The following example shows an excerpt from the report with processing option 1 selected. This option setting prints the security analyzer report sorted by User ID, then by User Group, and finally by Data Source.



The following example shows an excerpt from the report with processing option 2 selected. This option setting prints the security analyzer report sorted by User Group, then User ID, and finally by Data Source.



Setting Up Unified Logon

For configurations that use a Windows NT enterprise server, to set up unified logon, you only need to modify the [SECURITY] section of the jde.ini file. When a user signs on, these settings alert OneWorld to use unified logon.

When your enterprise server is on a non-Windows NT platform, you need to set up a Windows NT service for unified logon. This service identifies the unified logon server for OneWorld. You also need to set the unified logon settings in the [SECURITY] section of the jde.ini file.

Important: When you use unified logon, you need to use the same user ID for the Windows NT domain and OneWorld so that the records for each are in synchronization. For example, if the user ID for a user in the Windows NT domain is USER1, the user ID for OneWorld must also be USER1. If the user IDs are different, unified logon will not work for the user.

Complete the following tasks:

- Modify the jde.ini settings to enable or disable unified logon
- Reset the Environment Selection form
- Setup a service for a unified logon server
- Remove a service for a unified logon server

▶ To modify the jde.ini setting to enable or disable unified logon

Locate the jde.ini files on the server and on the workstation.

1. In the server jde.ini file, add the following settings in the [SECURITY] section:

```
[SECURITY]
SecurityMode=0, 1 or 2
```

Value	Description
0	A value of 0 for SecurityMode will accept only users set up for standard signon security.
1	A value of 1 for SecurityMode will accept only users set up for unified logon.
2	A value of 2 for SecurityMode will accept users set up for both unified logon and standard signon security.

2. In the workstation jde.ini file, add the following settings in the [SECURITY] section:

```
[SECURITY]
UnifiedLogon=0 or 1
UnifiedLogonServer=server_name
```

Value	Description
0	A value of 0 for UnifiedLogon disables unified logon for the workstation. This setting is the default.
1	A value of 1 for UnifiedLogon sets unified logon for the workstation.
server_name	Enter the name of the server on which the unified logon server data resides.

▶ To reset the Environment Selection form

Locate the jde.ini files on the workstation.

In the workstation jde.ini file, modify the following setting:

```
[SECURITY]
ShowUnifiedLogon=0 or 1
```

Value	Description
0	A value of 0 for ShowUnifiedLogon disables the Environment Selection form. When you click the checkbox on the Environment Selection form to set a default environment, you set this value to 0.
1	A value of 1 for ShowUnifiedLogon enables the Environment Selection form. When a user signs on to OneWorld, the Environment Selection form appears to allow the user to choose an environment. This is the default setting for ShowUnifiedLogon

▶ To setup a service for unified logon

If your enterprise server is non-Windows NT, J.D. Edwards recommends that you set up services for unified logon on your deployment server. Your deployment server is always a Windows NT server.

1. Run UniLogonSetup.exe.

   The Unified Logon Server Setup form appears. On this form, you define the Windows NT service for unified logon servers. You can also remove these services on this form.

2. Complete the following fields:

   - Unified Logon Service Name:

     Enter the name for your unified logon server in this field.

   - OneWorld Port Number:

     The port number for your unified logon server should match the OneWorld port number of the OneWorld server for which you want to set up unified logon.

   - Service Executable Filename:

     Enter the directory path for the unified logon service program in this field.

   - Log Filename:

     Enter the name of the unified logon log file including the full directory path in this field.

3. The default user list contains all authenticated network users. If you need to create a custom user list, enter users or groups in the Users or User Groups box to add user information to the unified logon user list.

   Note: Generally, the default Windows NT list of authenticated network users lists users by group.

4. Click the Install Service button to save the service information for the unified logon server.

▶ To remove a service for unified logon

1. Run UniLogonSetup.exe.

   The Unified Logon Server Setup form appears.

2. From the Unified Logon Service Name drop-down listbox, choose a unified logon server, then click the Uninstall Service button.

Vocabulary Overrides

Vocabulary Overrides is an application that you can use to change the text that displays on forms and reports. You can specify both form columns and row headings, provide customization for multiple languages and industries, and retain your overrides with your next OneWorld software update.

Because the Vocabulary Overrides application (P9220) affects the use interface throughout OneWorld, it is very important that you secure this application from most of your users. When you work with Vocabulary Overrides for an interactive or batch application, Vocabulary Overrides simulates an application check out from your Central Objects repository, just as if you checked out the application using Object Librarian. This is done so that while you are working on the application in Vocabulary Overrides no one can check the application out. After you finish creating overrides, Vocabulary Overrides removes the checkout line in Object Librarian.

Note: When the Object Librarian line is written for Vocabulary Overrides, no specifications are brought down to the requesting workstation. Instead, this workstation accesses the relational database tables directly. See the *OneWorld Development Tools* guide for information about using Object Librarian and checking applications in and out.

After you make vocabulary override changes, use an update package to push these changes to your users. See the *Package Management Guide* for information about building packages. You do *not* need to rebuild the application before building it into a package if the only change to the application is due to vocabulary overrides.

This section contains the following:

- ❑ Accessing vocabulary overrides
- ❑ Creating vocabulary overrides
- ❑ Reviewing vocabulary overrides
- ❑ Resetting vocabulary overrides

Accessing Vocabulary Overrides

You can access the Vocabulary Overrides application from either a OneWorld Explorer menu (System Administration Tools) or from within Object Librarian.

This topic contains the following:

- Accessing Vocabulary Overrides from System Administration Tools
- Accessing Vocabulary Overrides from Object Librarian

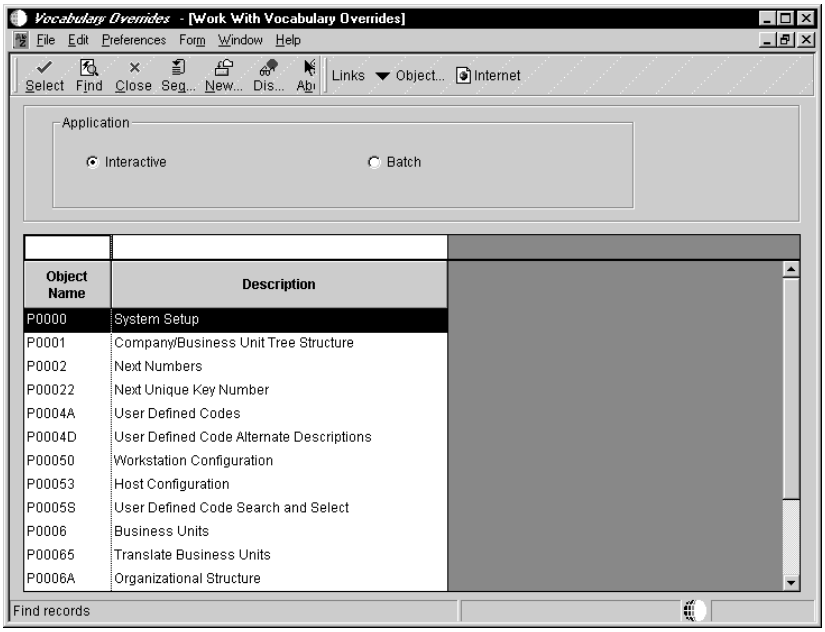
▶ To access Vocabulary Overrides from System Administration Tools

From System Administration Tools (GH9011)

1. Choose Vocabulary Overrides (P9220).

   The Work With Vocabulary Overrides form appears. Use this form to select the application for which you want to create vocabulary overrides.

2. On Work With Vocabulary Overrides, under the Application heading, click Interactive or Batch, use the query by example fields if you want to refine your search, then click Find.



The detail area lists interactive or batch applications.

3. Choose the application that you want, then click Select.

   If the application you selected is already checked out, an error message appears, which reads: “This object is currently in use by Object Librarian and, therefore, unavailable.” You will have to create vocabulary overrides for this application at another time or contact the user(s) of the application to see if they can check in or erase their checkout line.

4. If the SAR Requirement form appears, complete the following field:

   - SAR Number

     Enter a SAR number. This form appears if the system administrator set up the processing option for Vocabulary Overrides to require a SAR number for overrides.

   The Interactive Vocabulary Overrides form or the Batch Vocabulary Overrides form appears. All of the interactive forms or batch versions associated with the application you selected appear in the detail area. You can expand any row that has a plus (+) sign on the left side.

   The Vocabulary Overrides application writes a checkout line for this application in Object Librarian. This is done so that while you are working on the application in Vocabulary Overrides no one can check the application out. After you finish creating overrides, Vocabulary Overrides removes the checkout line in Object Librarian.

Field	Explanation
Object Name	The OneWorld architecture is object based. This means that discrete software objects are the building blocks for all applications, and that developers can reuse the objects in multiple applications. Each object is stored in the Object Librarian. Examples of OneWorld objects include: • Batch Applications • Interactive Applications • Business Views • Business Functions • Business Functions Data Structures • Event Rules • Media Object Data Structures
Description	The description of a record in the Software Versions Repository file. The member description is consistent with the base member description.

▶ To access Vocabulary Overrides from Object Librarian

From Cross Application Development Tools (GH902)

1. Choose Object Librarian (P9860).

   The Work With Object Librarian form appears.

2. Click Find, choose an interactive or batch application, then click Select.

   The Application Design Aid form or the Batch Application Librarian form appears.

3. From the Form menu, choose Vocabulary Overrides.

   A Vocabulary Override Warning form appears with the following text: "Warning! You are now accessing Vocabulary Overrides. This application will override currently checked in objects. You must have authority to make changes."

4. If you have authorization to make Vocabulary Override changes, click OK on the Vocabulary Override Warning.
5. If the SAR Requirement form appears, complete the following field:

   - SAR Number

     Enter a SAR number. This form appears if the system administrator set the processing option that requires a SAR number for Vocabulary Overrides.

   The Work with Vocabulary Overrides form appears.

6. Click Select.

   The Interactive Vocabulary Overrides form or the Batch Vocabulary Overrides form appears. All of the interactive forms or batch versions associated with the application appear in the detail area. You can expand any row that has a plus (+) sign on the left side.

   The Vocabulary Overrides application writes a checkout line for this application in Object Librarian. This is done so that while you are working on the application in Vocabulary Overrides no one can check the application out. After you finish creating overrides, Vocabulary Overrides removes the checkout line in Object Librarian.

Creating Vocabulary Overrides

You can create vocabulary overrides to customize your interactive and batch OneWorld applications. After you make vocabulary override changes, use an update package to push these changes to your users. For example, you could create vocabulary overrides for the Verify OCM report. After you make vocabulary override changes, you should use an Update Package to push these changes to your users.

You do *not* need to re-build the application before building it into a package if the only change to the application is due to vocabulary overrides. See the *Package Management Guide* for information about building packages.

Note: When you create a vocabulary override for a report, the override occurs at the version level. When you run the version, the vocabulary override appears on the report instead of the data dictionary description. The vocabulary override does not affect the base report specifications or any other version of the report.

Complete the following tasks:

- Create interactive vocabulary overrides
- Create batch vocabulary overrides

▶ To create an interactive vocabulary override

To access this form, see *Accessing Vocabulary Overrides* in this section.

1. To work with a language other than the domestic language, on Interactive Vocabulary Overrides, complete the following field, then click Find:

   - Language

     Enter a language code. Leave this field blank if you are creating vocabulary overrides in your domestic language.

2. Double-click the + icon on one of the forms listed in the detail area.

   The form expands, displaying the types of text that are available on that form, such as find/browse text, control text, grid column text, exit text, and text variables.

3. Double-click the + icon on one of the types of text.

   The type of text expands, displaying all of the text that you can override.

4. To create a vocabulary override, change the text in the Description column for a particular item. Click OK when you finish creating overrides.

Some descriptions for data items contain carriage returns and new-line characters. To create a vocabulary override for these descriptions (indicated with a special icon to the left of that row), choose the data item row, and from the Row menu, choose Extended Text Revision.

The Extended Text Revision form appears. Change the text in the free-form area, then click OK.

The Vocabulary Overrides application writes a checkout line for this application in Object Librarian. This is done so that while you are working on the application in Vocabulary Overrides no one can check the application out. After you finish creating overrides, Vocabulary Overrides removes the checkout line in Object Librarian.

Field	Explanation
Language	A user defined code (system 01/type LP) that specifies a language to use in forms and printed reports. Before any translations can become effective, a language code must exist at either the system level or in your user preferences.
Form Name	The unique name assigned to a form.
Description	A free-form text field for comments or memoranda.
Override	Identifies whether the default value of a particular text item has been overridden. Overrides are defined during development of interactive and batch applications when the developer determines that the data dictionary value, or current text, is not appropriate or explicit enough for the particular application. When the interactive or batch application runs, the text is changed and this flag is set. The following values are valid: 0 Indicates the text is the original value. If it is a data item releated field, then it is the same as the data dictionary. 1 Indicates the original text, often the data dictionary default, has been overridden.

Field	Explanation
Data Item	For World, the RPG data name. This data field has been set up as a 10-byte field for future use. Currently, it is restricted to 4 bytes so that, when preceded by a 2-byte table prefix, the RPG data name will not exceed 6 bytes. Within the Data Dictionary, all data items are referenced by this 4-byte data name. As they are used in database tables, a 2-character prefix is added to create unique data names in each table specification (DDS). If you are adding an error message, this field must be left blank. The system assigns the error message number using next numbers. The name appears on a successful add. You should assign error message numbers greater than 5000. Special characters are not allowed as part of the data item name, with the exception of #, @, $. You can create protected data names by using $xxx and @xxx, where you define xxx. For OneWorld, a code that identifies and defines unit of information. It is an 8-character, alphabetical code that does not allow blanks or special characters such as: % & , . +. Create new data items using system codes 55-59. The alias cannot be changed.
Text Type	A user defined name or remark. . . . . . *Form-specific information* . . . . . . Used to display the type of control with which the text is associated. For example, the text Grid Column would indicate that the text is a column header in a grid.
Visible	Indicates whether the data item or text for a data item is visible on a form or report. This is defined during development of the form or report when the developer determines that a data item is required for processing, but should not be visible on the form or report. The valid values are: 1 The data item or data item text are visible on the form or report 0 The data item or data item text are not visible on the form or report

See Also

- ❏ *Object Librarian* and *Checkout Log* in the *Development Tools Guide* for more information about how OneWorld checks objects in and out.
- ❏ *Package Management Guide* for information about deploying changed applications to users.

To create a batch vocabulary override

To access this form, see *Accessing Vocabulary Overrides* in this section.

1. To work with a language other than the domestic language, on Batch Vocabulary Overrides, complete the following field, then click Find:

   - Language

     Enter a language code. Leave this field blank if you are creating vocabulary overrides in your base (domestic) language.

2. Double-click the + icon of one of the versions listed in the detail area.

   The version expands, displaying the types of text that are available on that version, such as page header and group sections.

3. Double-click directly on the plus (+) sign icon of one of the types of text.

   The type of text expands, displaying all of the text that you can override.

4. To create a vocabulary override, change the text in the Description column for a particular item. Click OK when you finish creating overrides.

   The Vocabulary Overrides application writes a checkout line for this application in Object Librarian. This is done so that while you are working on the application in Vocabulary Overrides no one can check the application out. After you finish creating overrides, Vocabulary Overrides removes the checkout line in Object Librarian.

Field	Explanation
Language	A user defined code (system 01/type LP) that specifies a language to use in forms and printed reports. Before any translations can become effective, a language code must exist at either the system level or in your user preferences.
Version	The name given to identify a version of the software.
Section	A user defined name or remark.
Description	A free-form text field for comments or memoranda.

Field	Explanation
Override	Identifies whether the default value of a particular text item has been overridden. Overrides are defined during development of interactive and batch applications when the developer determines that the Data Dictionary value is not appropriate or explicit enough for the particular application. When the interactive or batch application runs, the text is changed and this flag is set. The following values are valid: 0 The text is the original value from Data Dictionary 1 The Data Dictionary default value has been overridden
Data Item	For World, the RPG data name. This data field has been set up as a 10-byte field for future use. Currently, it is restricted to 4 bytes so that, when preceded by a 2-byte table prefix, the RPG data name will not exceed 6 bytes. Within the Data Dictionary, all data items are referenced by this 4-byte data name. As they are used in database tables, a 2-character prefix is added to create unique data names in each table specification (DDS). If you are adding an error message, this field must be left blank. The system assigns the error message number using next numbers. The name appears on a successful add. You should assign error message numbers greater than 5000. Special characters are not allowed as part of the data item name, with the exception of #, @, $. You can create protected data names by using $xxx and @xxx, where you define xxx. For OneWorld, a code that identifies and defines unit of information. It is an 8-character, alphabetical code that does not allow blanks or special characters such as: % & , . +. Create new data items using system codes 55-59. The alias cannot be changed.
Control	A user defined name or remark.
Visible	Indicates whether the data item or text for a data item is visible on a form or report. This is defined during development of the form or report when the developer determines that a data item is required for processing, but should not be visible on the form or report. The valid values are: 1 The data item or data item text are visible on the form or report 0 The data item or data item text are not visible on the form or report

Reviewing Vocabulary Overrides

You can use Vocabulary Overrides to review every location in OneWorld where someone has overidden a data item. You can view the override locations either from a form or from a report.

▶ To review Vocabulary Overrides

From System Administration Tools (GH9011)

1. Choose Vocabulary Overrides (P9220).

   The Work With Vocabulary Overrides form appears.

2. On Work With Vocabulary Overrides, from the Form menu, choose Overrides.

   The Overridden Data Item Search form appears.

3. Complete the following, then click OK.
   - Enter a Data Item to search for

     Enter the name of a data item into the Data Item field.

   - Select a scope for the Application search

     Click on either Interactive, Batch, or Both. This will refine the search for the data item to only interactive or batch applications.

   - Output results to

     Click either Interactive Application or Printed Report. This determines how you want to view the results of your search.

   If you selected to view your search via the Interactive Application, the Data Item Locator form appears when this search is complete, which could take some time. This form displays a list of all of the applications where the data item appears.

   If you selected to view your search via the Printed Report, an Acrobat Portable Document Format (PDF) file is created, which you can view or print.

Resetting Vocabulary Overrides

You can reset vocabulary overrides to the original data dictionary definition. If you need to reset multiple vocabulary overrides to the default data dictionary definition, OneWorld provides an automated process that resets overrides at the interactive form level, the batch version level, and the interactive and batch application level. When you reset vocabulary overrides at the form level, you reset all vocabulary overrides on a specific form, for example, the Work with Addresses form (W01012B) in the Address Book application. When you reset vocabulary overrides at the application level, you reset all vocabulary overrides on all forms or versions in an entire interactive or batch application, for example, the Address Book application (P0101) or the Print Mailing Labels report (R01401).

Complete the following tasks:

- Reset a vocabulary override
- Reset all vocabulary overrides on a form (interactive)
- Reset all vocabulary overrides in a version (batch)
- Reset all vocabulary overrides in an application (interactive and batch)

Before You Begin

❑ Access the Vocabulary Overrides application. See *Accessing Vocabulary Overrides* in this section.

▶ To reset a vocabulary override

On the Work with Vocabulary Overrides form

1. Click one of the following options, then click Find:
    - Interactive
    - Batch

2. Choose an application, then click Select.

   Depending on the application type, one of the following forms appears:

   - If you chose an interactive application, the Interactive Vocabulary Overrides form appears. The forms associated with the application appear in the grid on this form.
   - If you chose a batch application, the Batch Vocabulary Overrides form appears. The versions associated with the application appear in the grid on this form.

3. Double-click the + icon in the row header for one of the forms or versions in the grid, then double-click the + icon in the row header for a type of text on the form or a type of section in the version.

   The grid expands to display the data items associated with the type of text or section.

4. Choose the data item that you want to reset, then from the Row menu, choose Reset Description.

   Note: The Reset Description menu option is inactive if a vocabulary override does not exist for the data item.

5. Click OK to return to the Work with Vocabulary Overrides form.

   If you click Cancel to return to the Work with Vocabulary Overrides form *after* you reset a vocabulary override, you *do not* cancel the action. The data item remains at the default data dictionary definition.

▶ To reset all vocabulary overrides on a form (interactive)

On the Work with Vocabulary Overrides form

1. Click the Interactive option, then click Find.
2. Choose an application, then click Select.

   The Interactive Vocabulary Overrides form appears. The grid on this form displays any forms within the application.

3. Choose a form, then from the Row menu, choose Reset by Form.

   OneWorld clears all vocabulary overrides from the form and resets the data items to the data dictionary definitions.

   Caution: When you choose the Reset by Form menu option, the decision is final. OneWorld does not provide a confirmation box or a proof mode.

▶ To reset all vocabulary overrides in a version (batch)

On the Work with Vocabulary Overrides form

1. Click the Batch option, then click Find.
2. Choose an application, then click Select.

   The Batch Vocabulary Overrides form appears. The grid on this form displays any versions for the application.

3. Choose a version, then from the Row menu, choose Reset by Version.

   OneWorld clears all vocabulary overrides from the version and resets the data items to the base definitions. If no base definition exists for a data item, OneWorld resets the data item to the default data dictionary definition.

   Caution: When you choose the Reset by Version menu option, the decision is final. OneWorld does not provide a confirmation box or a proof mode.

▶ To reset all vocabulary overrides in an application (interactive and batch)

On the Work with Vocabulary Overrides form

1. Click one of the following options, then click Find:
   - Interactive
   - Batch
2. Choose an application, then click Select.

   Depending on type of application, either the Interactive Vocabulary Overrides form or the Batch Vocabulary Overrides form appears. The grid displays forms for interactive applications and versions for batch applications.

3. From the Form menu, choose the Reset by Application menu option for interactive applications or the Reset by Batch menu option for batch applications.

   OneWorld clears all vocabulary overrides from the *entire* application and resets the data items to the base definitions. If no base definition exists for a data item, OneWorld resets the data item to the default data dictionary definition.

   Caution: When you choose either the Reset by Application or the Reset by Batch menu option, the decision is final. OneWorld does not provide a confirmation box or a proof mode.

Transaction Processing

A transaction is a logical unit of work (comprised of one or more SQL statements) performed on the database to complete a common task and maintain data consistency. Transaction statements are closely related and perform interdependent actions. Each statement performs part of the task, but all are required for the complete task.

Transaction Processing ensures that related data is added to or deleted from the database simultaneously, thus preserving data integrity in your application. In transaction processing, data is not written to the database until a commit command is issued. When this happens, data is permanently written to the database.

For example, if a transaction comprises database operations to update two database tables, either all updates will be made to both tables, or no updates will be made to either table. This guarantees that the data remains in a consistent state and the integrity of the data is maintained.

You see a consistent view of the database during a transaction. You do not see changes from other users during a transaction.

Transaction Processing ensures that transactions are:

- Atomic - Either all database changes for an entire transaction are completed or none of the changes happen.
- Consistent - Database changes transform from one consistent database state to another.
- Isolated - Transactions from concurrent applications do not interfere with each other. The updates from a transaction are not visible to other transactions executing concurrently until the transaction commits.
- Durable - Complete database operations are permanently written to the database.

Commits and Rollbacks

The scope of a transaction is defined by the beginning and the end of the transaction. The end of a transaction occurs when the transaction is committed or rolled back. If neither a commit or a rollback of a transaction occurs, the transaction is rolled back when you exit OneWorld.

Transaction processing uses commits to control database operations. Commits are commands to the database. Transactions can be auto or manual commit. For auto commits, database changes are written permanently to the database (committed) as they are executed. For manual commits, database changes are only written permanently to the database when either a commit or rollback occurs.

Commit

A commit is an explicit command to the database to permanently store the results of operations performed by a statement. This is the successful end of a transaction.

Two-Phase Commit (Manual Commit Mode)

A two-phase commit coordinate a distributed transaction. They occur only when at least one update statement has been executed to two separate data sources in the same transaction.

Rollback

A rollback is an explicit command to the database to cancel the results of operations performed by a statement. This is the unsuccessful end of a transaction.

Any failure of an insert, update, or delete within a transaction boundary will cause all record activity within that transaction to roll back. If no failures have occurred at the end of the transaction, a commit is done and the records become available to other processes.

In the case of a catastrophic failure (for example, network problems) the DBMS performs an auto-rollback. If the user clicks Cancel on a form, a rollback command is issued through a system function.

This section describes:

- Understanding OneWorld transaction processing
- Understanding Record Locking
- Setting jde.ini files for transaction processing

Understanding Transaction Processing

A OneWorld transaction is a logical unit of work (comprised of one or more SQL statements) performed on any number of databases. A single-statement transaction consists of one statement, and a multiple-statement transaction consists of more than one statement.

You can construct a transaction within a OneWorld application to group multiple database operations. The application can then request the database management system to buffer the database operations until the application executes a specific command to perform the updates requested within the transaction. Database operations that are not part of a transaction update the database immediately.

If transaction processing is on in an application, you cannot see updated records until an update has been committed. Only processes within that transaction can access records in the transaction until the transaction is complete.

The OneWorld Application Design tool allows you to enable an application for transaction processing and to define what database operations comprise a transaction. Not all transactions or applications must be enabled. Enable transaction or applications appropriately according to your database configuration.

If transaction processing is on for database operations against tables that reside in DB2, then those tables must be journalled or you may have problems. Journalling can increase performance overhead due to the additional processing required. Contact your DB2 administrator if you have problems with this.

General messages and errors for transaction processing are written to the jde.log or jdedebug.log.

Data Interdependence

Data interdependence refers to the data elements that make a transaction complete. For example, a voucher has records in both the F0411 and F0911 tables. Because there is data interdependence between the two tables, the transaction is incomplete when there is data in one table and not the other.

Transaction Boundaries

Data interdependence is defined by a transaction boundary. A transaction boundary encompasses all of the data elements that comprise a transaction. A transaction boundary might include only the data elements on a single form. When a transaction includes data from another form, the transaction boundary must be extended to include the data on that form.

Transaction Processing Scenarios

The typical flow for a transaction is as follows:

1. Application starts and calls runtime engine
2. Runtime engine initializes the transaction
3. Runtime engine opens a view
4. Runtime engine performs database operations
5. Runtime engine commits database operations

If you want two connected forms to be included in the same transaction boundary you must turn on transaction processing for the parent form and designate “Include in parent” on interconnect to the second form. You do not need to turn on transaction processing for the second form because your choice on your interconnect form will override your choice on the called form.

The following table outlines the relationship between two forms and the boundaries that exist in each scenario. Transaction boundaries are defined through form interconnections and business function interconnections. In the example below, the OK button on Form1 invokes Form2. You can change the transaction boundaries by changing TP On and TP Off. The table explains what would happen if you defined your transaction boundary in various ways.

Scenario		TP On	TP Off	Form, BSFN Interconnect, Table I/O	Comment
A	Form1		X		All forms use Auto Commit.
	Form2		X		
B	Form1		X	X	Because neither form uses Manual Commit the Include in Parent flag on Form Interconnect Properties is ignored.
	Form2		X		All forms use Auto Commit.
C	Form1	X			Form1 (parent) uses Manual Commit mode, and Form2 (child) uses Auto Commit.
	Form2		X		Because the Include in Parent flag is Off, the transaction boundary does not extend to include Form2 (child).

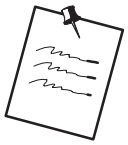
Scenario		TP On	TP Off	Form, BSFN Interconnect, Table I/O	Comment
D	Form1	X		X	Even though transaction processing flag is Off for Form2 (child), the Include in Parent flag is On.
	Form2		X		The transaction boundary extends to include Form2 (child).
E	Form1		X		Because the Include in Parent flag is Off, Form1 (parent) and Form2 (child) operate as independent entities.
	Form2	X			Form1 operates in Auto Commit mode and Form2 operates in Manual Commit mode.
F	Form1		X	X	An odd case. Because transaction processing is Off for Form1 (parent), the transaction boundary does not extend to the child, even though the Include in Parent flag is On for Form2 (child).
	Form2	X			Form2 (child) is in Manual Commit mode and the interconnect is ignored.
G	Form1	X			Transaction processing is On for both forms.
	Form2	X			Because the Include in Parent flag is Off, each form is a transaction boundary and a commit is issued for each.
H	Form1	X		X	Transaction processing is On for both forms. However, because the Include in Parent flag is On the transaction processing on Form2 is ignored.
	Form2	X			The transaction boundary encompasses both forms. Form2 is a child of Form1.

Transaction Processing and Business Functions

An application or batch process establishes the primary transaction boundary. If you have a business function that calls another business function, the database operations in the function being called are still grouped within the boundaries of the parent application.

Master business functions should not define their own boundaries. You may require two or more master business functions to create one logical transaction, so the calling application should define the boundaries.

If your application calls several business functions and you need the business functions included in the transaction boundary, you must enable transaction processing. If you need to roll back database operations for the business function if there is a failure, you must designate “Include in Transaction” on the business function interconnect.



When you use business functions within a transaction you must be careful not to cause a deadlock. If you split the logic for manipulating a table between two functions, you may cause a deadlock if you include one function in the transaction, but not the other. If you have a business function that selects records for information and also updates or inserts data to other tables, you may want to split the business function apart.

Transaction Processing in Remote Business Functions

In a transaction-enabled application, if a server business function has modified a record and a client business function outside the transaction attempts to access the record, the client function will be locked out until the server business function has committed. Database changes performed by server-side business functions will not be seen by the client application until the data is committed. If a server business function fails to commit or a user cancels a transaction on a server business function, the business function's transactions roll back.

Transaction Processing System Functions

There are several transaction processing system functions available. There may be situations where you need to use system function for additional transaction processing functionality.

For example, you have two forms, FormA and FormB and FormA has transaction processing enabled and calls FormB with the "Include in Parent" option on for the *Post OK Button is Clicked* event. Since FormB inherits FormA's transaction boundaries, if a user cancels on FormB, the following occurs: FormB's entries will not be written, control is returned to FormA, and FormA's entries are written and committed. If you want to prevent commitment of FormA's entries in this situation, you use the Rollback Transaction system function.

You can use the following system functions to define transaction boundaries in a batch process:

- *Begin Transaction* to define where the transaction begins
- *Commit Transaction* to define where the transaction ends
- *Rollback Transaction* to rollback a transaction

Refer to the Online APIs for more information about specific system functions.

Understanding Record Locking

OneWorld does not implement any data-locking techniques. It relies on the native locking strategy of the vendor database management system. This improves performance by reducing duplication of efforts.

There are some specific situations when the vendor database does not automatically lock as needed. In these situations, you can use explicit instructions to OneWorld to control data-locking. For example, you can use record locking to ensure the integrity of the Next Numbers facility.

OneWorld data locking may be accomplished using one of the following methods:

- Optimistic Locking

  You can use optimistic locking (sometimes referred to as "record change detection") to prevent a user from updating a record if it has changed between the time the user inquired on the record and when they want to update the record.

- Pessimistic Locking

  You can use record locking to prevent attempts to update the same record at the same time. The record is locked before it is updated.

Optimistic Locking

You can turn on "record change detection" within the workstation jde.ini file. Refer to *Setting jde.ini Files for Transaction Processing* in the *System Administration Guide* for more information. This type of database locking prevents a user from updating a record that changes during the time the user is inquiring on it. If the record has changed, the user must select the record again and then make the change. This functionality is available for business functions, table I/O and named event rules.

The following example illustrates "record change detection". For example, suppose two users are working within the Address Book application:

Time	Action
10:00	User A selects Address Book record "1001" to inspect it.

Time	Action
10:05	User B selects Address Book record “1001” to inspect it. Both users now have Address Book record “1001” open.
10:10	User B updates a field in Address Book record “1001” and clicks OK. OneWorld updates Address Book record “1001” with the update that User B made.
10:15	User A updates a field in Address Book record “1001” and clicks OK. OneWorld does not update Address Book record “1001,” and the system displays a message informing User A that the record has changed during the time User A was viewing it. For User A to change the record, User A must re-select it and perform the update.

When “record change detection occurs”, OneWorld displays the following message:

> “The record you are updating has been changed by another user. Please reselect your data.”

Pessimistic Locking

Pessimistic locking is sometimes referred to as simply “record locking.” You can use “record locking” to prevent multiple users or applications from trying to update the same record at the same time. For example, suppose a user enters a transaction that uses Next Numbers. When he clicks OK, the Next Number function selects the appropriate Next Number record, validates that this number is not already in the transaction file, and then updates the Next Number record by incrementing the number. If another process tries to access the same Next Number record before the first process has successfully updated the record, the Next Number function will wait until the record is unlocked, and then complete the second process.

Record locking in OneWorld is implemented by calling published JDEBase APIs. When you use “record locking”, you should consider the time it takes to select and update a record, because the record is locked until the update is complete. Transaction processing uses a special set of locking APIs. A locked record may or may not be part of a transaction. Record locking APIs are independent of the transaction and its boundaries. They always lock whether you are in manual or auto commit mode.

Records that are updated using record locking APIs (for example JDB_FetchForUpdate or JDB_UpdateCurrent) within a transaction boundary are locked from the time the record is selected for update until the commit or rollback occurs. Records within the transaction boundary that are updated without using record locking APIs, are locked from the time of the update until the commit or rollback occurs. This is also true if you use a business function to define and activate transaction processing.

Using Pessimistic Record Locking Within a Transaction Boundary

You may need to use record locking in conjunction with transaction processing. For example, if you want records locked between the read and the update you must use record locking.

Business Functions and Pessimistic Record Locking

You may want to use record locking in a business function if the business function updates a table. The table being updated should have a high potential for record contention with another user or job. Remember that you should keep records locked for as short a time as possible. Try to ensure that the select or fetch for an update occurs as close to the update as possible.

The following pessimistic record locking APIs can be called from business functions:

- JDB_SelectAllForUpdate
- JDB_SelectForUpdate
- JDB_SelectKeyedForUpdate
- JDB_SelectKeyedPositionForUpdate
- JDB_FetchForUpdate
- JDB_FetchKeyedForUpdate
- JDB_UpdateCurrent
- JDB_DeleteCurrent

See Also

- The online *Published APIs*

Setting the jde.ini for Transaction Processing and Lock Manager

You must modify the enterprise server and workstation jde.ini files to enable transaction processing.

For each OneWorld workstation, you must enable transaction processing by changing settings in the workstation jde.ini file. You should make these changes on the deployment server to the resident jde.ini file that is delivered to workstations through package deployment, and then deploy a package with the changed jde.ini file.

This chapter contains the following topics:

- Understanding concurrent release support
- Understanding transaction processing and lock manager logging
- Setting the jde.ini for transaction processing and lock manager

Understanding Concurrent Release Support

Prior to release B73.3, transaction processing was primarily controlled by settings in the [TP MONITOR ENVIRONMENT] section of the jde.ini. On the server, this section contains nine settings. On the workstation, this section contains seven settings.

For release B73.3, the [TP MONITOR ENVIRONMENT] section has been removed from the jde.ini and replaced with the [LOCK MANAGER] section. This new section contains three settings in both the server and client jde.ini files.

Settings that used to be in the [TP MONITOR ENVIRONMENT] section were removed because they were either obsolete or assigned an internal default value.

The following shows why the settings in the [TP MONITOR ENVIRONMENT] section have been eliminated for release B73.3:

Setting	Reason for eliminating
Status	Obsolete. As of release B732.2, the TP monitor is always set to ON.

LogPath	Assigned the base directory from the jde.ini file.
LogStatements	Obsolete. Statements are always be logged.
LogBufferSize	An internal default is used (1 MB).
DisplayServerErrorMsg	Obsolete. The client always displays server error messages.
ServerRetryInterval	An internal default interval is used.
RegistryCleanupInterval	An internal default interval is used.
RegistryRecordLifeSpan	An internal default span is used.
ServerTimeout	This value is provided by JDENET settings.

During a transition period, both sections will be supported concurrently. There are three possible scenarios during this transition period:

- **The [LOCK MANAGER] section does not exist.** In this scenario, OneWorld checks for settings in the [TP MONITOR ENVIRONMENT] section.
- **Both [LOCK MANAGER] and [TP MONITOR ENVIRONMENT] sections exist.** In this scenario, OneWorld uses the settings in the [LOCK MANAGER] section.
- **Neither section exists.** In this scenario, transaction processing cannot be started and a failure occurs.

Understanding Transaction Processing Logging

The two-phase commit coordinator acts in two phases. In the first phase, it instructs the Log Manager to flush the logs for each data source to hard disk. This makes a permanent backup storage. The logs contain every database operation that was carried out.

This action ensures that if any of the data sources fail to commit after the others have committed, all databases can be returned to a consistent state by referring to the contents of the logs. If all the logs for each of the data sources are flushed successfully, then the second phase begins.

In the second phase, the coordinator instructs each of the data sources to commit its respective transaction. If any of the data sources fails to commit, a commit log report is generated from the logs that were generated in phase one (this is written to the directory specified in the LOGPATH in the jde.ini which contains a listing by data source of all the SQL statements that were part of the transaction). Also contained in the commit log are details as to which data sources passed and which ones did not.

This information will help the database administrator to manually synchronize the data sources so that they are all in a consistent state. The log report is only generated when at least one data source fails to commit. If all data sources successfully commit, then no log report is generated, and all the logs from phase one are deleted by the Log Manager.

Setting the jde.ini for Transaction Processing and Lock Manager

As previously explained, during a transitional period both the [TP MONITOR ENVIRONMENT] and [LOCK MANAGER] sections will be supported.

If you are using a release of OneWorld prior to B73.3, enter settings for the [TP MONITOR ENVIRONMENT] section. If you are using release B73.3 or higher, enter settings for the [LOCK MANAGER] section.

Settings for the TP MONITOR ENVIRONMENT section (pre B73.3)

The following tasks describe how to enter settings for the [TP MONITOR ENVIRONMENT] section of the jde.ini file, both for the server and for the workstation. These settings will be used only if you are using a OneWorld release prior to B73.3.

▶ To enter TP MONITOR ENVIRONMENT settings for the server

1. Locate your enterprise server's jde.ini file.
2. Using an ASCII editor such as Notepad, view the jde.ini file to ensure accuracy of the following settings:

```
[TP MONITOR ENVIRONMENT]
Status=status value
LogPath=log path
LogStatements=log on/off value
LogBufferSize=log buffer value
RequestedService=service value
Server=server name
ServerTimeout=timeout value
AvailableService=service value
RegistryCleanupInterval=cleanup value
RegistryRecordLifeSpan=life span value
RegistryRecordLifeSpan=lifespan value
LogServices=service value
```

See the following table for explanations of the variables above.

Setting	Value
Status	Indicates whether transaction processing is on or off. The transaction processing monitor should usually be on unless you are not using transaction processing in any of your applications, or if you want to temporarily disable transaction processing, such as for testing. Valid values are ON and OFF.
LogPath	This setting specifies directory in which the transaction logs are placed. This path should correspond to the location of your jde.log and ldedebug.log. For example, a UNIX path might be: /u10/owdevel/tc283984/b73.2 For more information about logging, see *Understanding Transaction Processing Logging*.
LogStatements	This setting specifies whether the transaction monitor should keep a log of every operation performed within a transaction. Valid values are ON and OFF.
LogBufferSize	This setting indicates the number of bytes set aside to hold the operations being logged in memory before they are copied to disk. This value is a OneWorld internal default value and should not be changed.
RequestedService	This setting specifies the service that the client requests from the server. Valid values are: • TS Time stamp service is requested • NONE No service is requested
Server	Specifies the server that is hosting the TMS. For example, a server name might be: intelnta
ServerTimeout	The timeout in seconds for all the network operations. This value can be adjusted based on the network traffic. It is necessary for the workstation jde.ini file. It is also necessary for the server jde.ini file in case a batch job is running on the server. This value is a OneWorld internal default value and should not be changed.

Setting	Value
AvailableService	Indicates the server that this Transaction Management Server is offering. When the Transaction Manager on the workstation is initialized, it queries the TMS for this value. This is called TM-TMS handshaking. If this value is the same as the one that the workstation has in its jde.ini file, then that service will be invoked at the appropriate times when OneWorld is running. Valid values are: • TS Record Change Detector (Timestamp Service) • NONE No service is available
RegistryCleanup Interval	The period after which all the expired records are deleted from the TMS record registry. This interval is specified in minutes. This value is a OneWorld internal default value and should not be changed.
RegistryRecordLife Span	The maximum period during which a record can exist in the TMS record registry. After this period the record will expire and will be deleted. This life span is specified in minutes. This value is a OneWorld internal default value and should not be changed.
LogServices	This setting turns on the Trace Log for TMS, and supplements the Jde.log file. Valid values are: • 1 Tracing for TMS is On • 0 Tracing for TMS is OFF The default value for this setting is 0. You should turn tracing for TMS on only after all other debugging methods have been exhausted.

▶ To enter TP MONITOR ENVIRONMENT settings for the workstation

Be sure to enable transaction processing on the server before enabling it on the workstation. If you try to set up the workstation's jde.ini file before you have set up the server's jde.ini, you could be requesting a service on the server that is not yet available, which will generate an error.

1. Locate the jde.ini file that will be sent to the workstation as part of a package installation. This file is located on the OneWorld deployment server in the release share path:

```
\\B xxx\CLIENT\MISC\jde.ini
```

   where *xxx* is the installed release level of OneWorld. For example, B732.

2. Using an ASCII editor such as Notepad, view the jde.ini file to ensure accuracy of the following settings:

```
[TP MONITOR ENVIRONMENT]
Status=status value
LogPath=log path
LogStatements=log on/off value
LogBufferSize=log buffer value
RequestedService=service value
Server=server name
ServerTimeout=timeout value
```

See the following table for explanations of the variables above.

Setting	Value
Status	Indicates whether transaction processing is on or off. The transaction processing monitor should usually be on unless you are not using transaction processing in any of your applications, or if you want to temporarily disable transaction processing, such as for testing. Valid values are ON and OFF.
LogPath	This setting specifies directory in which the transaction logs are placed. This path should correspond to the location of your jde.log and ldedebug.log. For example, a UNIX path might be: /u10/owdevel/tc283984/b73.2 For more information about logging, see *Understanding Transaction Processing Logging*.
LogStatements	This setting specifies whether the transaction monitor should keep a log of every operation performed within a transaction. Valid values are ON and OFF.
LogBufferSize	This setting indicates the number of bytes set aside to hold the operations being logged in memory before they are copied to disk. This value is a OneWorld internal default value and should not be changed.
RequestedService	This setting specifies the service that the client requests from the server. Valid values are: • TS Time stamp service is requested • NONE No service is requested

Setting	Value
Server	Specifies the server that is hosting the TMS. For example, a server name might be: intelnta
ServerTimeout	The timeout in seconds for all the network operations. This value can be adjusted based on the network traffic. It is necessary for the workstation jde.ini file. It is also necessary for the server jde.ini file in case a batch job is running on the server. This value is a OneWorld internal default value and should not be changed.

The last three lines pertain to record change detection and must be set if you want the workstation to perform "record is changed" database locking (as explained under the *Record is Changed* topic).

Instead of deploying a package, you could also manually copy the jde.ini file to all workstations.

Settings for the LOCK MANAGER section (B73.3 and higher)

The following tasks describe how to enter settings for the [LOCK MANAGER] section of the jde.ini file, both for the server and for the workstation. These settings will be used even if you entered values for the [TP MONITOR ENVIRONMENT] section.

▶ To enter LOCK MANAGER settings for the server

1. Locate your enterprise server jde.ini file.
2. Using an ASCII editor such as Notepad, view the jde.ini file to ensure accuracy of the following settings:

```
[LOCK MANAGER]
Server=server name
AvailableService=available server service
RequestedService=client service request
```

See the following table for explanations of the variables.

Setting	Value
Server	This setting indicates the lock manager server to be used to process records. The value for this setting is the name of the server acting as the lock manager. For example, a server name might be: intelnta If the client is used as a server (such as in cases where batch applications are running on the workstation,) this setting must match the same entry in the workstation jde.ini file's [LOCK MANAGER] section.
`AvailableService`	This setting indicates the available service of the server. Valid values are: TS Time stamp service is available NONE No service is available This setting applies only to servers.
`RequestedService`	This setting indicates the type of service that the client requests from the server. Valid values are: TS Time stamp service is requested NONE No service is requested

▶ To enter LOCK MANAGER settings for the workstation

1. Locate your workstation jde.ini file.
2. Using an ASCII editor such as Notepad, view the jde.ini file to ensure accuracy of the following settings:

```
[LOCK MANAGER]
Server=server name
RequestedService=client service request
```

See the following table for explanations of the variables.

Setting	Value
Server	This setting indicates the lock manager server to be used to process records. The value for this setting is the name of the server acting as the lock manager. For example, a server name might be: intelnta If the client is used as a server (such as in cases where batch applications are running on the workstation,) this setting must match the same entry in the workstation jde.ini file's [LOCK MANAGER] section.
`RequestedService`	This setting indicates the type of service that the client requests from the server. Valid values are: TS Time stamp service is requested NONE No service is requested

The Scheduler Application

There might be times when you want to run batch jobs that take up a great deal of machine resources or that require users to be signed off after normal working hours. You might also want the flexibility to run jobs at scheduled intervals during the day or even periodically throughout the month or year.

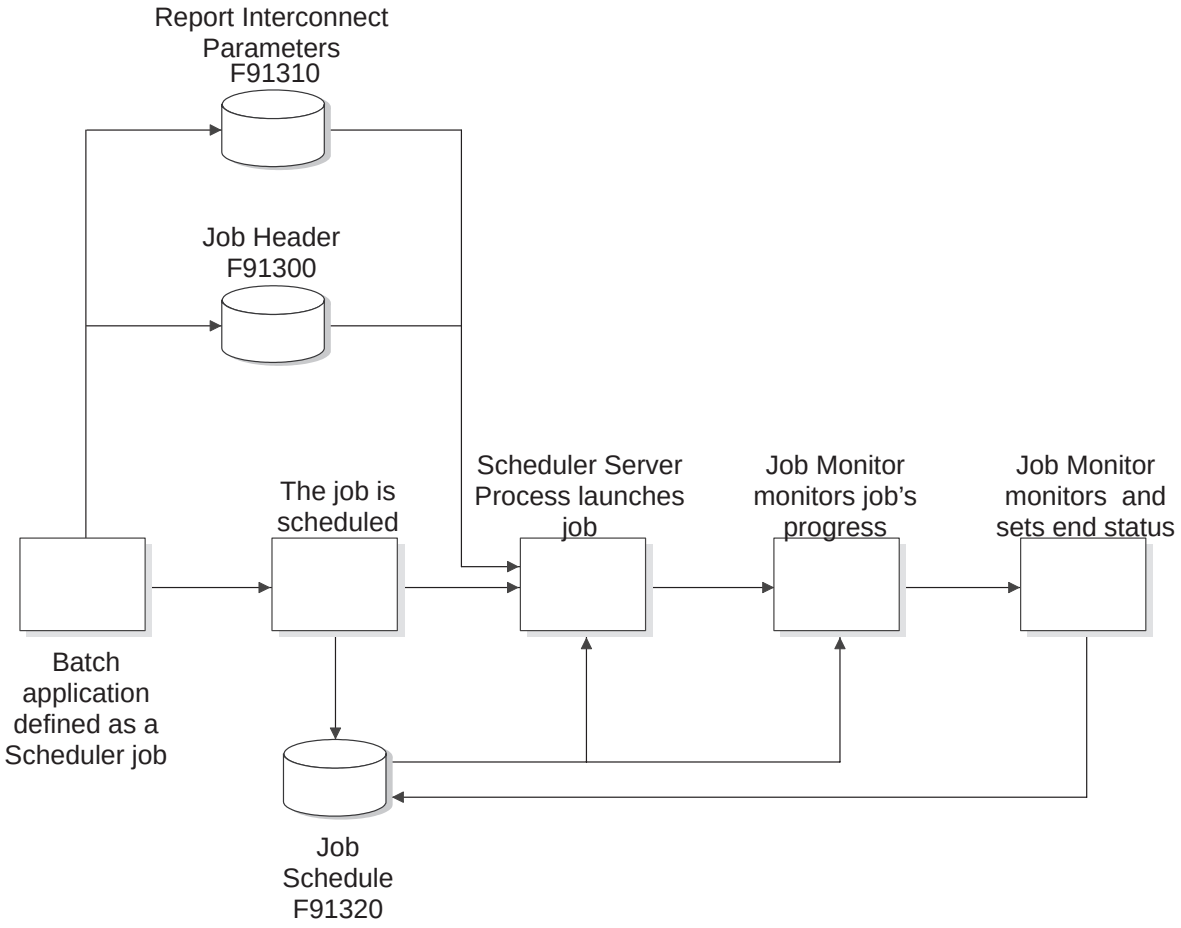
The Scheduler application enables you to schedule batch jobs to run after hours or periodically throughout the day, if necessary, according to a schedule you define. You can schedule jobs by time, daily, weekly, monthly, yearly, or based a specified period. You can also set up the scheduler to restart a job in the event of a job failure.

You can even specify the server on which you want the job to run, as well as the time zone, whether it is in your city or in some other country. The Scheduler system uses a modified version of Universal Coordinated Time (UTC), which counts the number of minutes, not seconds.

Included in this section are the following topics:

- ❑ Scheduling jobs
- ❑ Working with job properties
- ❑ Working with the job schedule
- ❑ Understanding the Scheduler server
- ❑ Working with the Scheduler server
- ❑ Modifying daylight savings rules
- ❑ Running Scheduler reports

The following graphic illustrates the scheduling process.



When you define a scheduled job, the parameters of that job are stored in the Job Master table (F91300).

After the job is scheduled, the system writes records to the Job Schedule table (F91320), indicating each time that the job should be launched. As the job runs, the Job Monitor monitors the job's progress.

When the job ends, the Job Monitor assigns an End status to the job and updates the job's record in the Job Schedule table (F91320) to indicate that the job either ended successfully or in error.

Scheduling Jobs

When you schedule a batch process to run through the Scheduler, you can also add a recurrence pattern to the job, which means that the job will restart at the intervals you define: once a week, once a month, once a year, and so on. You schedule jobs in the local time of the server on which the job will run.

This chapter contains the following topics:

- ❑ Scheduling a job
- ❑ Scheduling a recurring job
- ❑ Revising a scheduled job
- ❑ Entering Scheduler processing options
- ❑ Reviewing all jobs or local jobs

Before You Begin

- ❑ To use a server's time zone, you must first specify the time zones you want to use. To do this, copy the Scheduler processing options (version ZJDE001 on the Work with Versions form) and modify them according to your needs. If you use more than one time zone you should modify the processing options to display the Work with Versions form each time you invoke the Schedule Jobs application. That way you can choose each time zone in which you want to schedule the job.

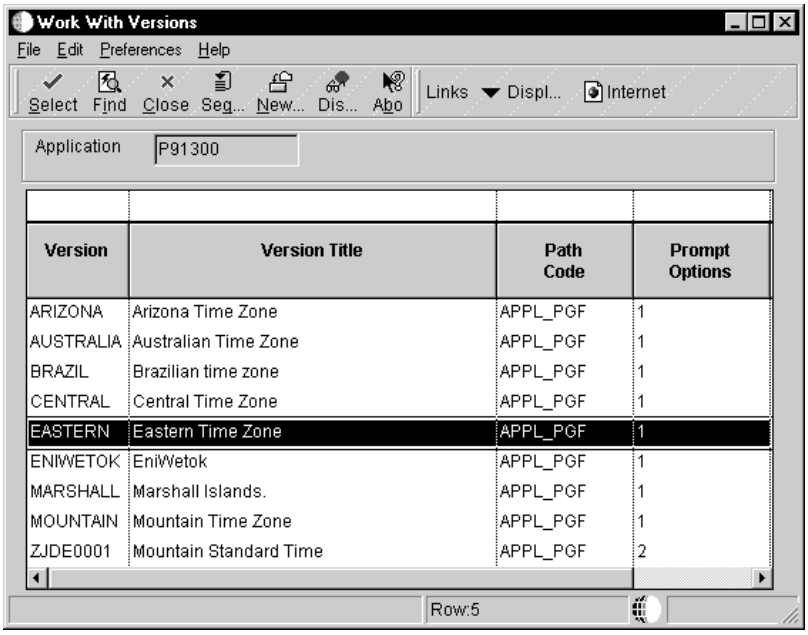
Scheduling a Job

When you schedule a job you choose a time zone in which the job will be run, and then define the parameters of the job. For example, if you want to submit a job from your workstation in Germany that will run on a server in Australia, you would choose the time zone for Australia so that the job will run at the intended time.

To schedule a job

From the System Administration Tools menu (GH9011)

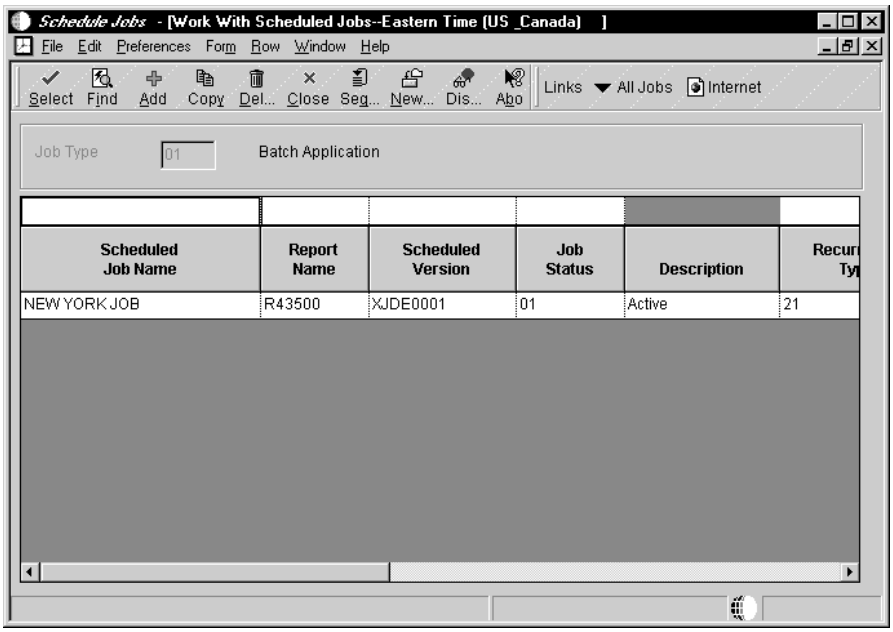
1. Choose Scheduler (GH9015), and then choose Schedule Jobs (P91300). The Work with Versions form appears.



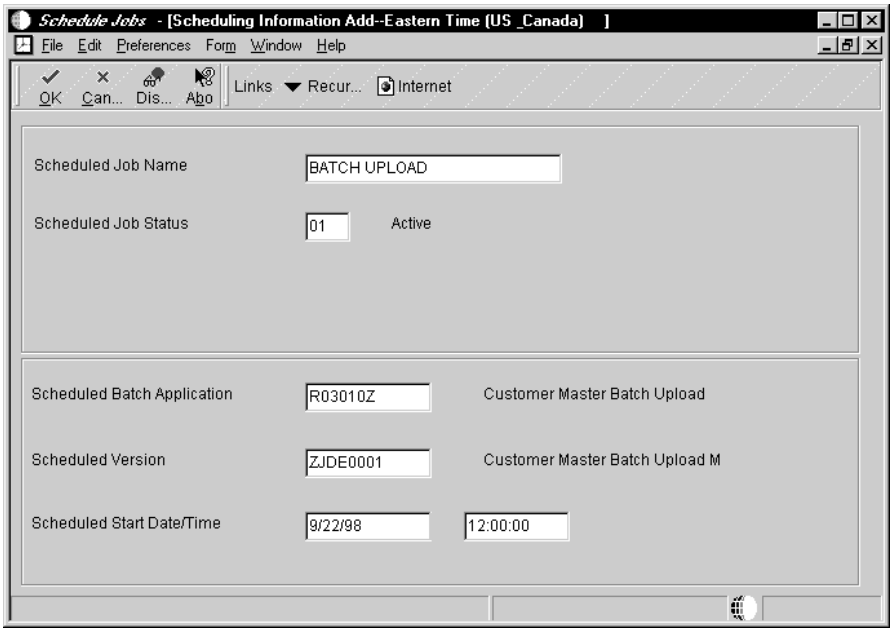
   If you use only one time zone you might not be prompted to select a version. In this case the Work with Versions form does not appear and you can skip the following step.

   When you select version ZJDE001, you will be able to enter Scheduler processing options. For more information about entering processing options, see *Entering Scheduler Processing Options*.

2. On the Work with Versions form, choose the time zone in which the scheduled job will run. For example, you might choose the Eastern Time Zone to run jobs in Eastern Standard Time (EST). The Work with Schedule Jobs form appears.



3. On Work with Scheduled Jobs, click Add. The Scheduling Information Add form appears.



4. On Schedule Information Add, complete the following fields:
   - Scheduled Job Name
   - Scheduled Job Status
   - Scheduled Batch Application

- Scheduled Version
- Scheduled Start Date/Time

5. Click OK.

Field	Explanation
Scheduled Job Name	A name that uniquely identifies a scheduled job to the system and the user. Use this name to indicate what the job function is. For example Monthly Close or Nightly Back Up.
Scheduled Job Status	The current status of the scheduled job. As long as the status is active the job will continue to be evaluated to be submitted to run. Once the scheduled end date for the job has been reached the status will be change to not active. The status may be marked as not active or suspended at any other time to stop the scheduler from considering the job for submission. It may then be reactivated so that the scheduler will begin including the job again. The job may only be reactivated if the calculation of the end date produces a date in the future.
Scheduled Start Date/Time	The next date on which the scheduled job will be submitted.
Scheduled Batch Application	The object name of the report that is submitted by the scheduler.
Scheduled Version	The version of the report scheduled to run. A version identifies a specific set of data selections and sequencing settings used by the batch job.

See Also

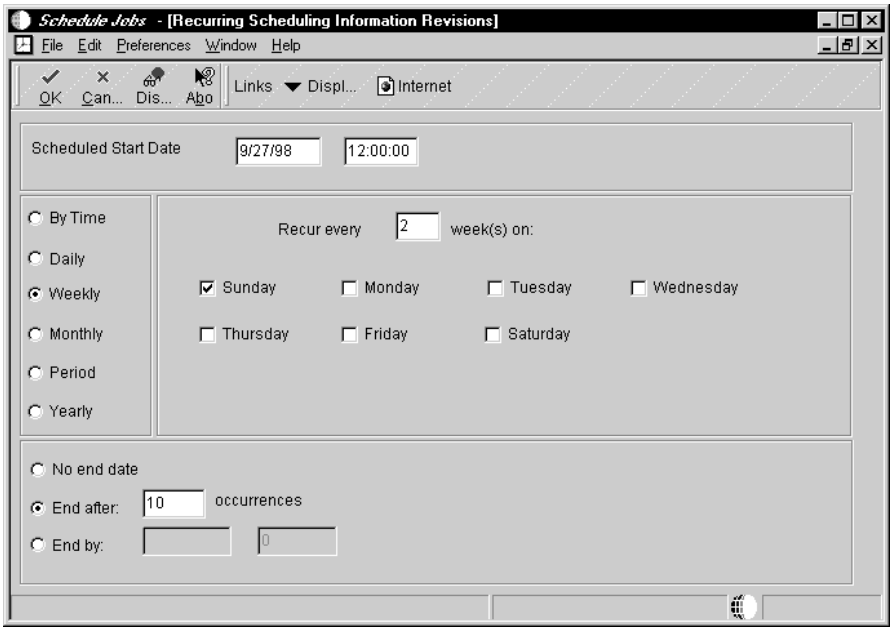
- *Scheduling a Recurring Job*
- *Working with Job Properties*
- *Entering Scheduler Processing Options*

Scheduling a Recurring Job

There might be jobs you need to run more than once. In these situations you can set recurrence frequency for a scheduled job so that it runs hourly, weekly, biweekly, monthly, and so on. You can also specify how many times you want the job run before it ends, or you can define a date after which the job will no longer run.

▶ To schedule a recurring job

1. On the Schedule Information Add form, choose Recurrence from the Row menu. The Recurring Scheduling Information Revisions form appears.



2. On Recurring Scheduling Information Revisions, choose one of the following options and complete the accompanying fields that appear after you choose an option:

   - By time
     - Every *n* minutes OR
     - Every *n* hours

     Run the job at the specified time interval. (For example, run the job every 40 minutes or every 8 hours.)

   - Daily
     - Every *n* days OR
     - Every weekday

     Run the job at the specified interval of days or every weekday. (For example, run the job every 7 days or every weekday.)

   - Weekly
     - Recur every *n* weeks on
     - Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday

     Run the job at the specified weekly interval on the specified day of the week. (For example, run the job every 2 weeks on Monday.)

- Monthly
  - Day *n* of every *x* month(s) OR
  - The [first, last, and so on] [day, Sunday, Friday, and so on] of every *n* month(s)

  Run the job on the specified day of every *x*th month, or on a specific day of every *n*th month. (For example, run the job on day 1 of every 4 months, or on the last Friday of every second month.)

- Period
  - Day *n* of every *x* period(s) OR
  - The [first, last, and so on] [day, Sunday, Friday, and so on] of every *n* period(s)
  - Company

  Run the job on the specified day of every *x*th period, or on a specific day of every *n*th period. Also enter the company code associated with that fiscal period. (For example, run the job on day 10 of every 2 periods, or on the last day of every third period.)

- Yearly
  - Every [month name] [date] OR
  - The [first, last, and so on] [day, Sunday, Friday, and so on] of [month name] OR
  - Day *x* of the [first, last, and so on] period OR
  - The [first, last, and so on] [day, Sunday, Friday, and so on] of the [first, last, and so on] period

  Run the job at the specified time of the year. For example, you might want to run the job on the last day of December of each year. (For example, run the job every January 1, or on the first Monday of June, or on day 15 of the last period, or on the second Tuesday of the fourth period.)

3. Specify when you want the Scheduler to stop submitting the job by choosing one of the following options:
   - No end date
   - End after *x* occurrences

     Where *x* is the number of occurrences of the job that can run before it expires.

   - End by

     Enter the month, day, and year on which you want the job to expire.

4. Click OK.

Field	Explanation
Scheduled Start Date	The next date on which the scheduled job will be submitted.
By Time	The type of interval for which the scheduled job should run, such as daily, weekly, monthly or yearly.
Daily	The type of interval for which the scheduled job should run, such as daily, weekly, monthly or yearly.
Weekly	The type of interval for which the scheduled job should run, such as daily, weekly, monthly or yearly.
Monthly	The type of interval for which the scheduled job should run, such as daily, weekly, monthly or yearly.
Period	The type of interval for which the scheduled job should run, such as daily, weekly, monthly or yearly.
Yearly	The type of interval for which the scheduled job should run, such as daily, weekly, monthly or yearly.
No end date	These options limit the time the job will be considered active by the scheduler. You can select either the number of times this job is allowed to run or a date and time for which the job will no longer run. The job will always be considered active if the No end date option is selected.
End after:	These options limit the time the job will be considered active by the scheduler. You can select either the number of times this job is allowed to run or a date and time for which the job will no longer run. The job will always be considered active if the No end date option is selected.
End by:	These options limit the time the job will be considered active by the scheduler. You can select either the number of times this job is allowed to run or a date and time for which the job will no longer run. The job will always be considered active if the No end date option is selected.

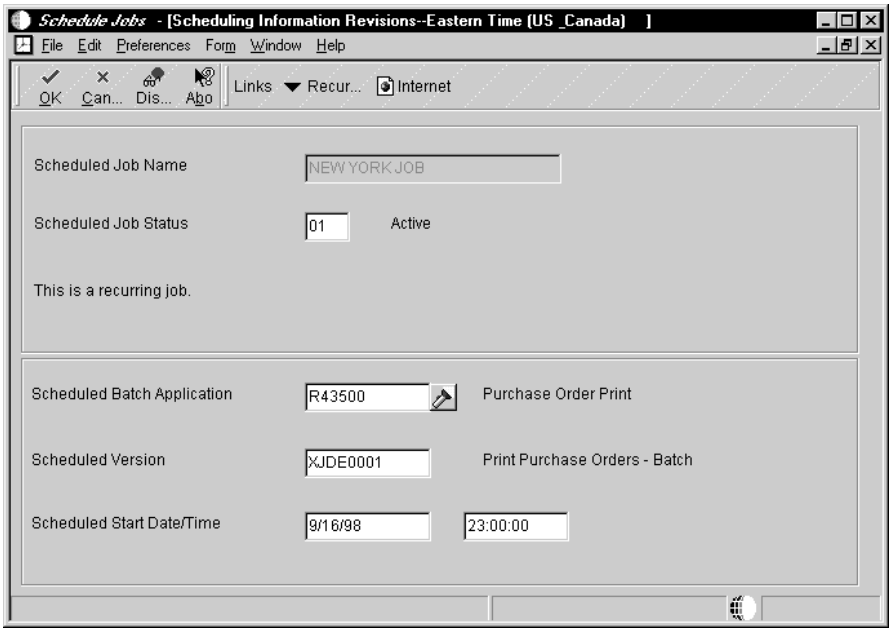
Revising a Scheduled Job

There might be times when you want to revise the information for a job. For example, you might want to change the job status to Active or Not Active. Or, you might want to enter a new batch process as the scheduled job, or change the job start date and time.

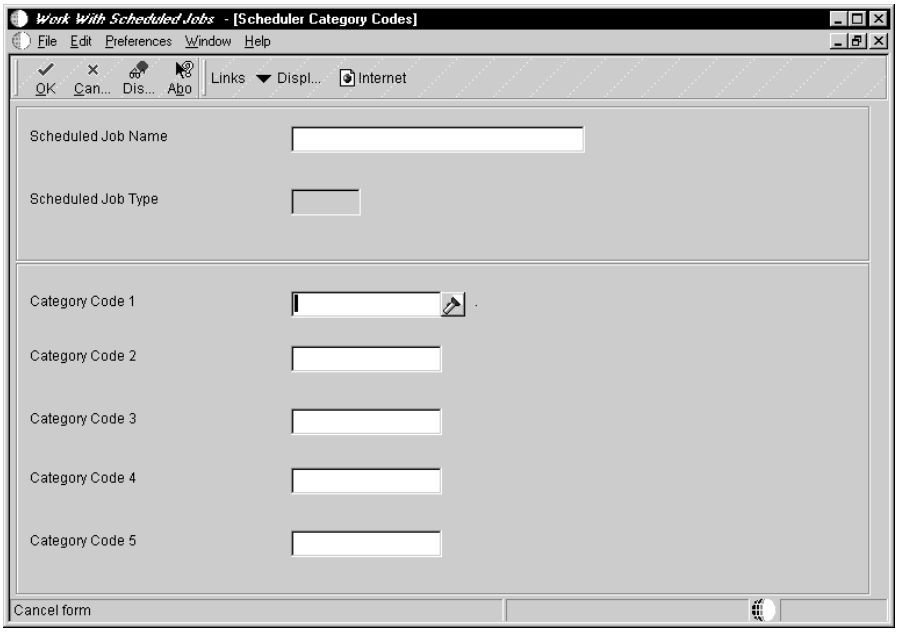
▶ To revise a scheduled job

From the Job Scheduler menu (GH9015)

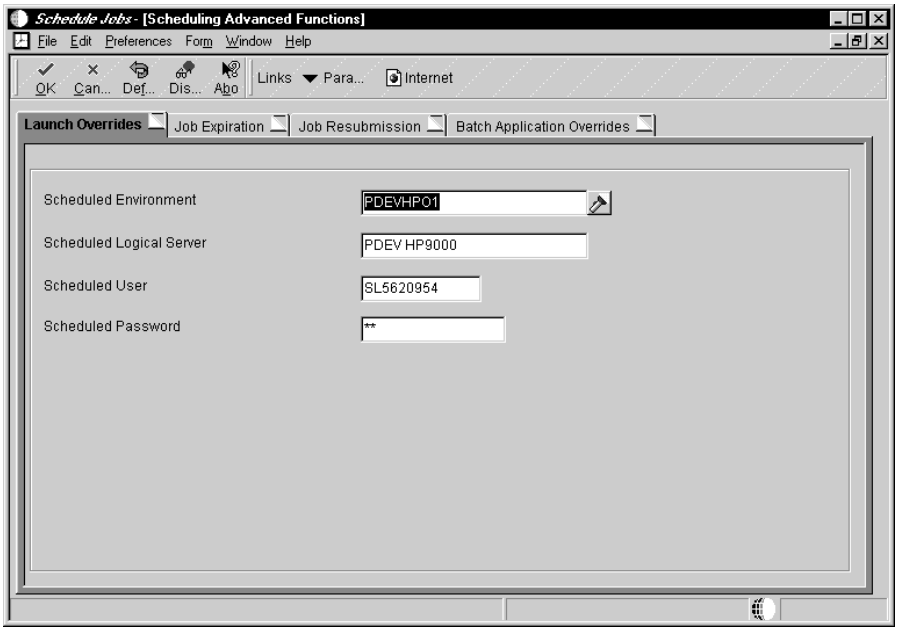
1. Choose Schedule Jobs.
2. On Work with Versions, choose the time zone in which the job will run and click Select.
3. On Work with Scheduled Jobs, choose the job you want to revise and then choose Job Revisions from the Row menu.



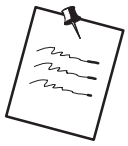
4. On Scheduling Information Revisions, modify the following fields as necessary and click OK:
   - Scheduled Job Status
   - Scheduled Batch Application
   - Scheduled Version
   - Scheduled Start Date/Time
5. If you want to remove recurrence from a scheduled job, choose Remove Recurrence from the Form menu.
6. If you want to add category codes to the scheduled job, choose Category Codes from the Form menu. The Scheduler Category Codes form appears.



7. If you want to revise this job's advanced functions, choose Advanced Functions from the Form menu. The Advanced Functions form appears.



For more information about entering advanced functions, see *Working With Job Properties.*



You can also activate or inactivate a job by choosing the job on Work with Scheduled Jobs and choosing Change Status from the Row menu.

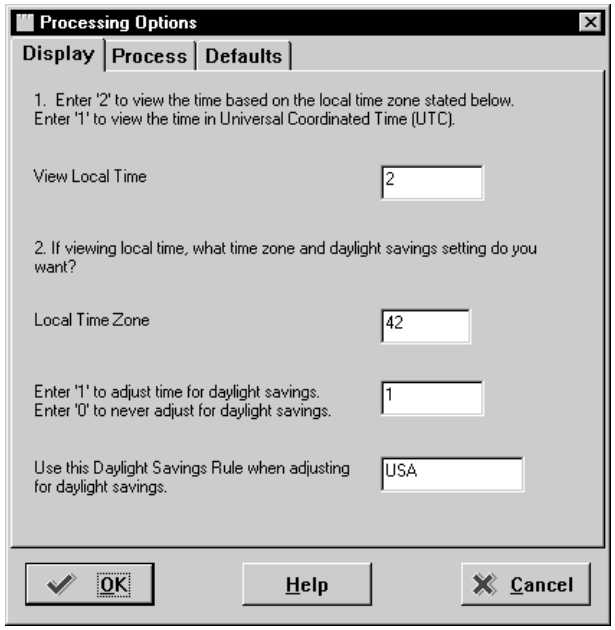
Entering Scheduler Processing Options

The Scheduler processing options enable you to specify preferences and defaults for time zone, job recurrence, and job type. You can enter or change processing options int two ways:

- Select version XJDE0001 from the Work with Versions form
- From the Job Scheduler menu (GH9015), highlight Schedule Jobs, click the right mouse button, and then select Prompt for Values

The Scheduler processing options form has three tabs that allow you to change different parameters:

- Display: Enter time zone information
- Process: Enter information about schedule records
- Defaults: Enter information about job type, job occurrences, and job resubmissions



Enter ’2’ to view the time based on the local time zone stated below. Enter ’1’ to view the time in Universal Coordinated Time (UTC).

View Local Time

This field enables you to specify how time is displayed. Enter 2 to view time based on the local time zone you enter in the View Local Time field.

Enter 1 to view time according to Universal Coordinated Time, formerly known as Greenwich Mean Time, which is the time at the Greenwich meridian (zero degrees longitude).

If viewing local time, what time zone and daylight savings setting do you want?

Local Time Zone

The remaining fields apply only if you view local time.

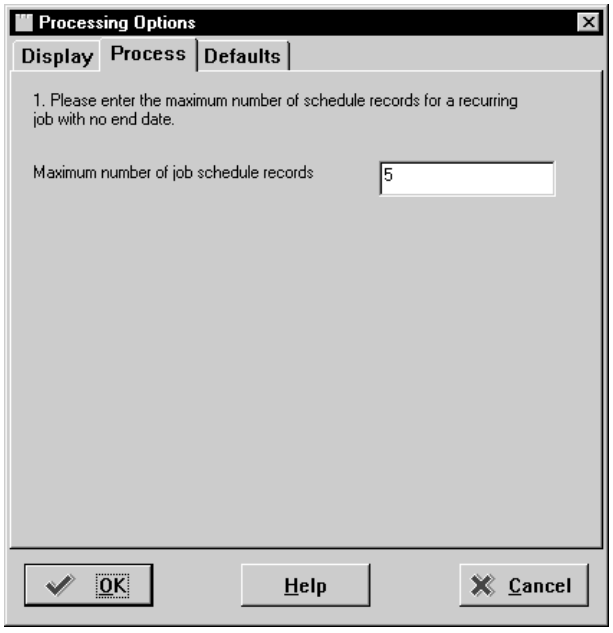
In the Local Time Zone field, use the visual assist to locate the local time zone for your area.

Enter '1' to adjust time for daylight savings. Enter '0' to never adjust for daylight savings.

Enter 1 to have OneWorld automatically adjust the time when daylight savings time goes into effect. If you do not want OneWorld to make this adjustment, enter 0.

Use this Daylight Savings Rule when adjusting for daylight savings.

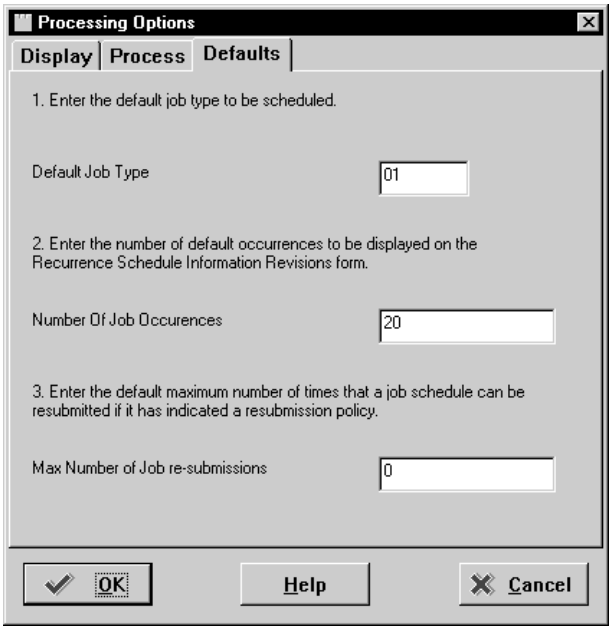
If you elect to adjust for Daylight Savings Time, specify the rule to use for making the adjustment. For more information about Daylight Savings Time rules, see *Modifying Daylight Savings Rules*.



Please enter the maximum number of schedule records for a recurring job with no end date.

Maximum number of job schedule records.

The number you enter at this field determines the maximum number of schedule records allowed for each recurring job that has no end date.



Enter the default job type to be scheduled.

Default Job Type

Use the visual assist to select the default job type, such as batch application or workflow process.

Enter the number of default occurrences to be displayed on the Recurrence Schedule Information Revisions form.

Number of Job Occurrences

The number you enter will be the default number that appears on the Recurrence Schedule Information Revisions form for the number of job occurrences. This setting is a default value only; you can always override the default value.

Enter the default maximum number of times that a job schedule can be resubmitted if it has indicated a resubmission policy.

Max Number of Job resubmissions

The number you enter will be the default value for the number of times a job can be resubmitted. This setting is a default value only; you can always override the default value.

Reviewing all Jobs or Local Jobs

If necessary, you can review all jobs in all time zones, or local jobs only. Depending on the view you are currently using, the system protects the other choice. For example, if you are currently viewing local jobs, the system would protect the Local Jobs choice and allow you to choose only All Jobs.

▶ To review all jobs or local jobs

From the Work with Scheduled Jobs form

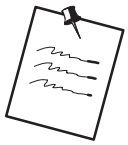
From the Form menu choose Display, then choose All Jobs or Local Jobs.

Field	Explanation
Scheduled Job Status	The current status of the scheduled job. As long as the status is active the job will continue to be evaluated to be submitted to run. Once the scheduled end date for the job has been reached the status will be change to not active. The status may be marked as not active or suspended at any other time to stop the scheduler from considering the job for submission. It may then be reactivated so that the scheduler will begin including the job again. The job may only be reactivated if the calculation of the end date produces a date in the future.
Scheduled Batch Application	The object name of the report that is submitted by the scheduler.
Scheduled Version	The version of the report scheduled to run. A version identifies a specific set of data selections and sequencing settings used by the batch job.
Scheduled Start Date/Time	The next date on which the scheduled job will be submitted.

Working with Job Properties

Use Advanced Functions to override the job properties, such as the location at which the job will run and the environment in which it will run. You can also use Advanced Functions to specify whether you want the system to resubmit a job if it ends in error, or if you want to change job expiration specifications.

You can even define whether to log errors to the jde.log or jdedebug.log and whether you want the system to resubmit a job or override printer locations and job queues. You can also add a report interconnect to a job if you want to pass parameters to it.



To restore the defaults in the advanced functions application, click the default button on Scheduling Advanced Functions.

The following topics are described:

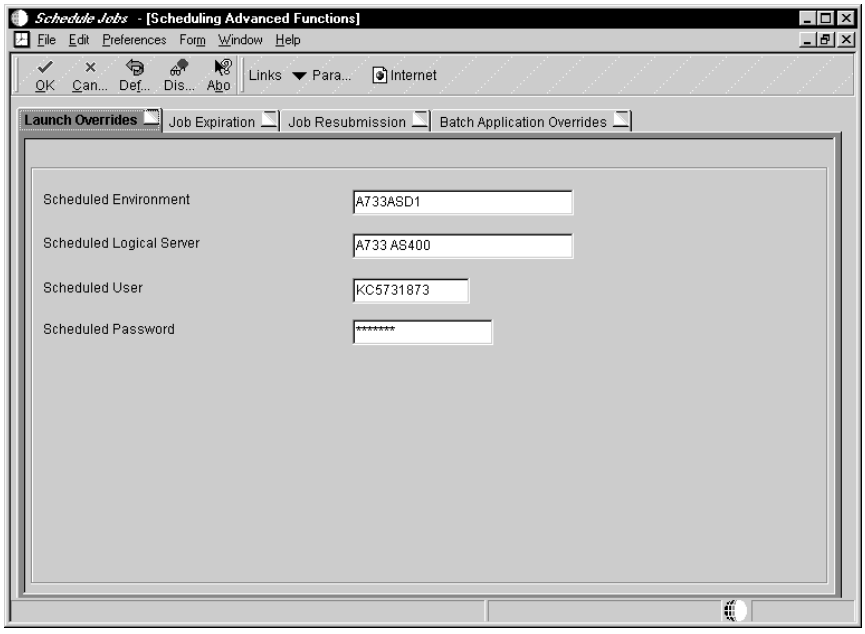
- Overriding the environment
- Overriding job expiration specifications
- Defining when the Scheduler will resubmit a job
- Overriding batch application specifications
- Adding values to a report interconnect

Overriding the Environment

There might be situations where you need to override the environment. For example, you would want to override the environment if the environment in which you want to run the job is not available or if it is different from the environment you are logged into when you scheduled the job.

▶ To override the environment

1. On the Work with Scheduled Jobs form, choose the job.
2. Choose Advanced Functions from the Row menu. The Scheduling Advanced Functions form appears.



3. On Scheduling Advanced Functions, complete the following fields on the Launch Overrides tab as necessary :

   - Scheduled Environment
   - Scheduled Logical Server
   - Scheduled User
   - Scheduled Password

     Enter the scheduled user's password in this field.

Field	Explanation
Scheduled Environment	Indicates the environment in which the scheduled job will run.
Scheduled Logical Server	The OneWorld logical server against which the job is submitted.
Scheduled User	Indicates the user ID for which the the job will start.
Scheduled Password	Indicates the password required to start the job. This will be encrypted before it is stored the in Job Schedule Master table.

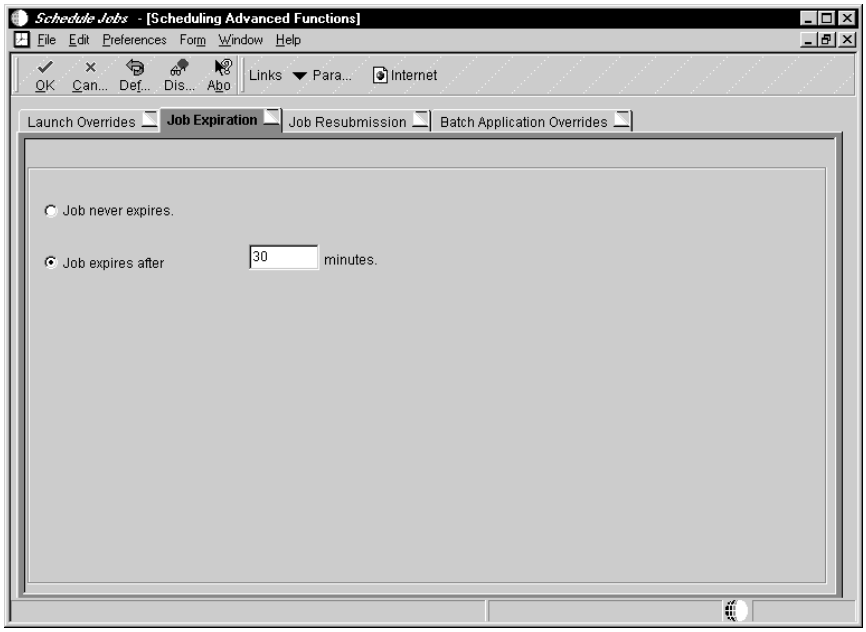
Overriding Job Expiration Specifications

Job expiration specifications ensure that servers do not become overloaded with unexpired jobs. If necessary, you can override job expiration specifications so that the job will never expire, or will expire after a certain number of minutes.

For example, suppose you schedule a job to run at midnight and another job for 1:00 a.m., but the server goes down and probably will not come back up again before the jobs are scheduled to run. In this case, you can specify that the first job you scheduled at midnight will expire in 30 minutes (12:30), so that if the server does not come back up within 30 minutes, the job will expire.

▶ To override job expiration specifications

1. On the Work with Scheduled Jobs form, choose the job, and then choose Advanced Functions from the Row menu.
2. On the Scheduling Advanced Functions form, click the Job Expiration tab.



3. Indicate whether you want the job to never expire, or expire after a certain period of time has elapsed.
4. Click OK.

Field	Explanation
Job never expires	This field indicates the number of minutes after the scheduled start time that this job expires. Once the job expires, it will not be run.

Field	Explanation
Job expires after	This field indicates the number of minutes after the scheduled start time that this job expires. Once the job expires, it will not be run.

Defining When the Scheduler Resubmits a Job

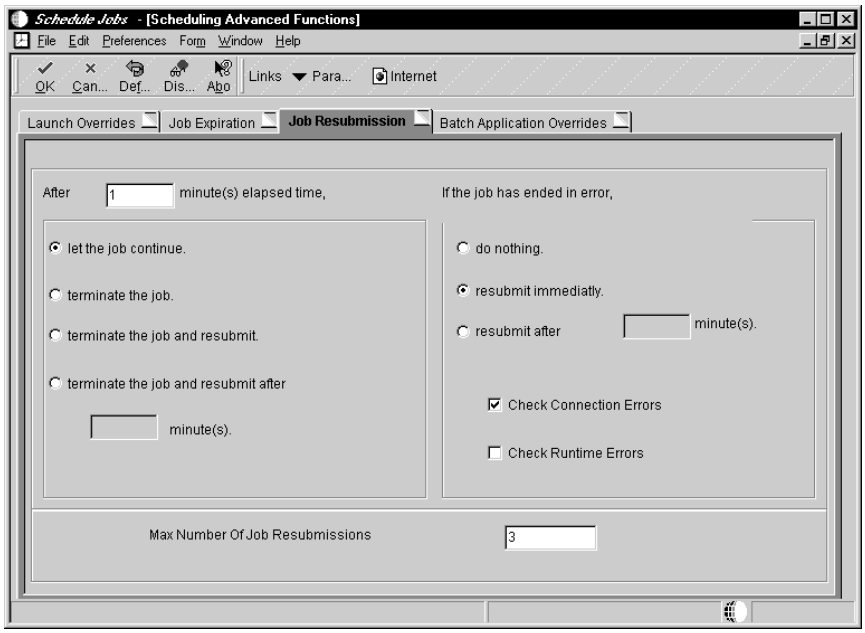
You can define when the Scheduler should resubmit jobs. This feature is useful if a job ends in error, for example, because the Scheduler will submit the job again after a certain period of time.

To help avoid taking up system resources, you can limit the number of times a job can be resubmitted.

You can also have the Scheduler check for connection errors or run-time errors when the job runs. Connection errors occur when the system fails to connect to the server to submit the job. Run-time errors occur when the server on which the job is running places the job in an error state. You can set up the system to monitor for both cases.

▶ To define when the system resubmits jobs

1. On the Work with Scheduled Jobs form, choose the job and then choose Advanced Functions from the Row menu.
2. On Scheduling Advanced Functions, click the Job Resubmission tab.



3. Specify the number of minutes that elapse before the job continues or terminates, and then choose one of the following options:
   - Let the job continue
   - Terminate the job
   - Terminate the job and resubmit
4. If you want to terminate the job and resubmit it after a certain period of time, choose the following option and enter the number of minutes you want to elapse before the system resubmits the job:
   - Terminate the job and resubmit after *x* minute(s).
5. Choose one of the following options for when the job ends in error:
   - Do nothing
   - Resubmit immediately
   - Resubmit after *x* minute(s)
6. Specify whether you want the system to check for connection or run-time errors, or both.
7. Specify the maximum number of times you want the job to be resubmitted.
   - Max Number of Job Resubmissions
8. Click OK.

Field	Explanation
let the job continue	Indicates how to handle a job that is in process too long.
terminate the job	Indicates how to handle a job that is in process too long.
terminate the job and resubmit	Indicates how to handle a job that is in process too long.
terminate the job and resubmit after	Indicates how to handle a job that is in process too long.
do nothing	Indicates the method for resubmitting a job if the job has an error status.
resubmit immediatly	Indicates the method for resubmitting a job if the job has an error status.
resubmit after	Indicates the method for resubmitting a job if the job has an error status.
resubmit after	Indicates the method for resubmitting a job if the job has an error status.
Check Connection Errors	Indicates the method for resubmitting a job if the job has an error status.

Field	Explanation
Max Number Of Job Resubmissions	This is the maximum number of times that a job schedule instance can be re-submitted. For example, a job is scheduled to be submitted at 12AM on April Second. The scheduler is instructed to re-submit this job if it ever ends in error. However, this field is set to three. That means the job can only be re-submitted three times. The fourth time this job ends in error, the scheduler will no longer submit it.

Overriding Batch Application Specifications

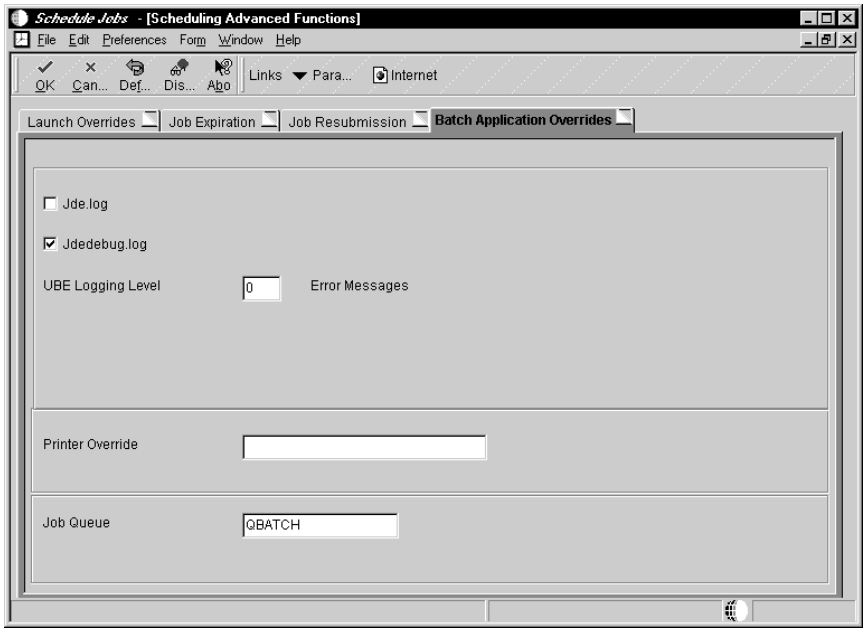
You can specify if you want errors written to the jde.log. If you want errors written to the jdedebug.log, you can set the trace level to determine what types of errors to include in the log.

You can also override the printer at which a report is printed. This feature is useful if a particular printer is down or if you want to print a report to a printer other than the default.

Also, you can override the queue to which the output of a submitted job is sent. If you want to pass parameters to a particular batch job, you can attach a report interconnect through Batch Application Specifications as well.

▶ To override batch application specifications

1. On the Work with Scheduled Jobs form, choose the job and then choose Advanced Functions from the Row menu.
2. On Advanced Functions, click the Batch Application Overrides tab.



3. Choose one or more of the following options:

   - jde.log
   - jdedebug.log

     If you chose jdedebug.log, you must also choose the jde.log option.

   - UBE Logging Level

     If you chose jdedebug.log, you can set a trace level to log certain levels of errors.

   - Printer Override

     Enter the name of the printer to which you want to print the report the job generates.

   - Job Queue

     Enter the name of the job queue to which you want the job output sent.

4. Click OK.

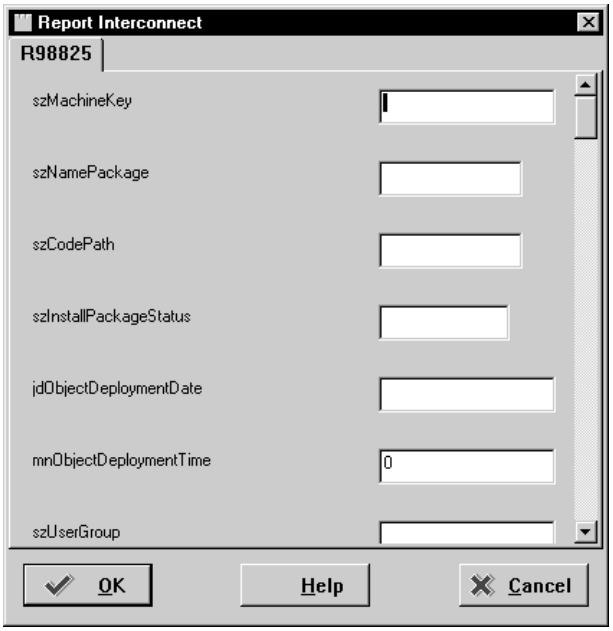
Adding Values to a Report Interconnect

You can add values to be passed through a report interconnect into a batch process when that batch process is launched. The batch process must first contain a report interconnect.

To add values to a report interconnect

1. On the Work with Scheduled Jobs form, choose the job and then choose Advanced Functions from the Row menu.
2. On Advanced Functions, choose Parameters from the Form menu.

   The system displays the Report Interconnect form with the parameters for that particular batch process.



3. Enter the values you want to pass to the batch process when the process runs.
4. Click OK.

Field	Explanation
Jde.log	When the batch job is run on a server, this field allows you to indicate if JDE logging should be enabled for the execution. If the server is already set to perform JDE logging, it occurs regardless of how this field is set.
Jdedebug.log	When the batch job runs on a server, this field indicates whether tracing is enabled for execution of the job. If the server is already set to perform tracing, it occurs regardless of how this field is set.

Field	Explanation
UBE Logging Level	Indicates the type of error logging that occurs when the batch job runs. The following list describes the different levels: 0 Error Messages 1 Informative Messages and Log Entry 2 Section Level Messages 3 Object Level Messages 4 Event Rule Messages 5 Database Mapping Messages 6 UBE Internal Function Calls, Textout Values
Printer Override	The default printer device.
Job Queue	The job queue to which the job was submitted. On the AS/400 this is an actual system job queue. On other systems it is a OneWorld logical queue.

See Also

- *Creating a Report Interconnect* in the *OneWorld Development Tools Guide*.
- *Debug Tracing* in the *OneWorld Development Tools Guide* for information about setting trace levels.

Working with the Job Schedule

When you schedule a job that includes a recurrence pattern, the system creates a set of schedule records, or instances, for the job in the Job Schedule table (F91320). The Job Schedule table indicates the times and dates the job will run. You can review these instances and their statuses, and also change the scheduled job information. For example, you can change the location at which you want a job to process, delete a job instance, or override any advanced functions.

Because the job schedule table is also used for audit information, you can modify or delete only jobs that have not yet run.

This chapter describes the following:

- ❏ Reviewing all job schedules
- ❏ Changing the launch status of a job
- ❏ Viewing the job detail
- ❏ Setting the job status manually

Reviewing All Job Schedules

When you review all job schedules, you can view all instances of jobs that have been launched. You can even revise a job by choosing a job instance, and then choosing Revise Job from the Row menu.

See *Revising a Scheduled Job* for more information about revising the job schedule.

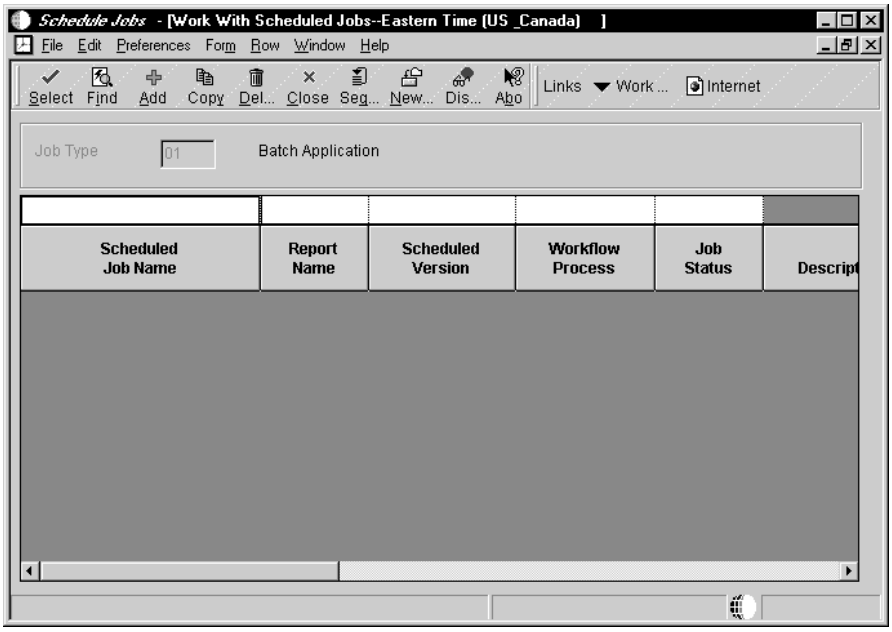
You can filter the job instances you want and review by launch date, start date, and time. For example, you can review all job instances by today's date by entering that date in the Scheduled Start Date field. Or, you can review all job instances that were launched on a certain date by entering that date in the Job Launch Date field.

You can also filter job instances by scheduled job name, launch status, report name, or scheduled version.

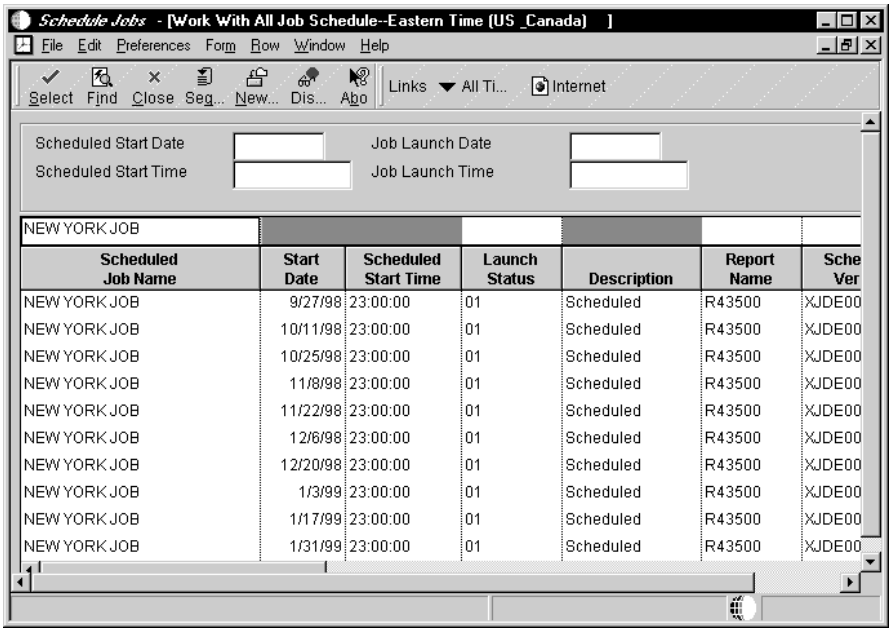
▶ To review all job schedules

From the Job Scheduler menu (GH9015)

1. Choose Schedule Jobs. The Work with Scheduled Jobs form appears.



2. On Work with Scheduled Jobs, choose All Schedules from the Form menu. The Work with All Job Schedules form appears.



3. On Work with All Job Schedules, do the following:
    - To filter by start date or start time, complete the following fields:
        - Scheduled Start Date
        - Scheduled Start Time

- To filter by launch date or launch time, complete the following fields:
  - Job Launch Date
  - Job Launch Time
- To filter by job name, launch status, report name, or scheduled version, complete the following fields:
  - Scheduled Job Name
  - Launch Status
  - Report Name
  - Scheduled Version
- To view all scheduled jobs in all time zone, choose All Time Zones from the Form menu.

  Alternatively, choose Local Time Zone from the Form menu to view all scheduled jobs in the local time zone.

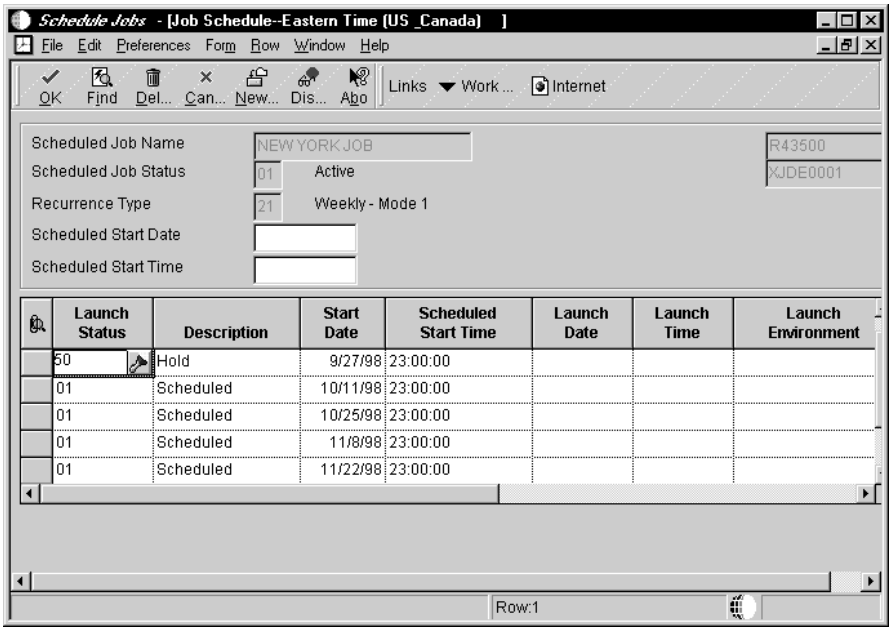
Changing the Job Launch Status

You can change the launch status for a job. For example, you might need to put a job on hold or reschedule a job.

▶ To change the job launch status

From the Job Scheduler menu (GH9015)

1. Choose Schedule Jobs. The Work with Version form appears.
2. On Work with Versions, choose the time zone in which the job will run and then click Select. The Work with Scheduled Jobs form appears.
3. On Work with Scheduled Jobs, find and choose the job you want to work with.
4. From the Row menu, choose Job Schedule. The Job Schedule form appears.



5. On Job Schedule, choose the job instance whose launch status you want to change, and then enter a new status in the Launch Status field.

   Enter 1 for Scheduled or 50 for Hold.

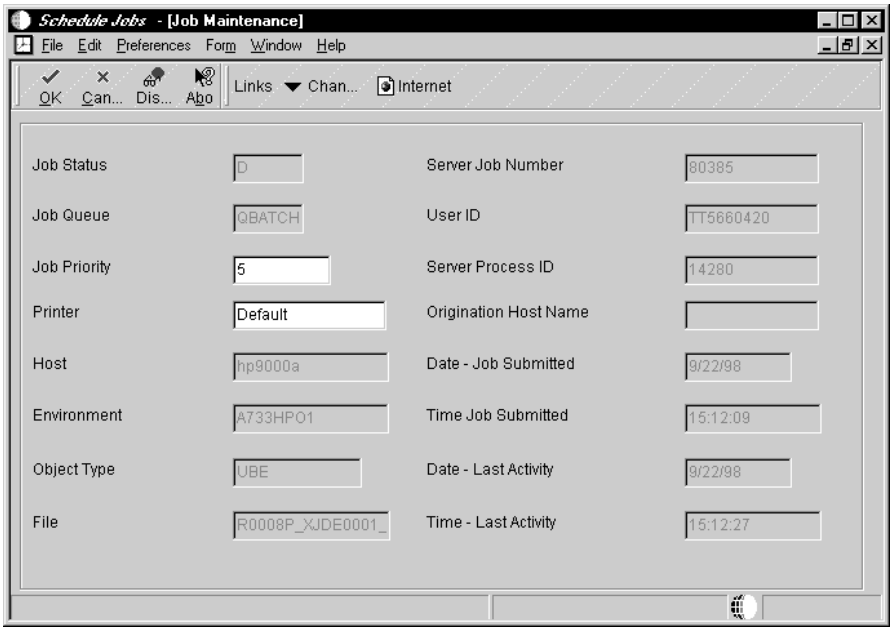
6. Click OK.

Viewing the Job Detail

You can view the job detail if you want to review the job queue, the priority in which it will run, the location of the report printer, and other details of the job. From this form you can also change the job priority or the location at which the report will print.

▶ To view the job detail

1. On the Job Schedule form, choose the job and then choose View Detail from the Row menu. The Job Maintenance form appears.



2. On Job Maintenance, complete the following fields, if necessary:
   - Priority
   - Printer
3. Click OK.

Setting the Job Status Manually

As a system administrator, you can change the status for jobs if the Scheduler is not updating the launch status, or if the Job Monitor is disabled.

If you need to kill a job, choose Work with Servers from the Form menu on the Work with Scheduled Jobs form.

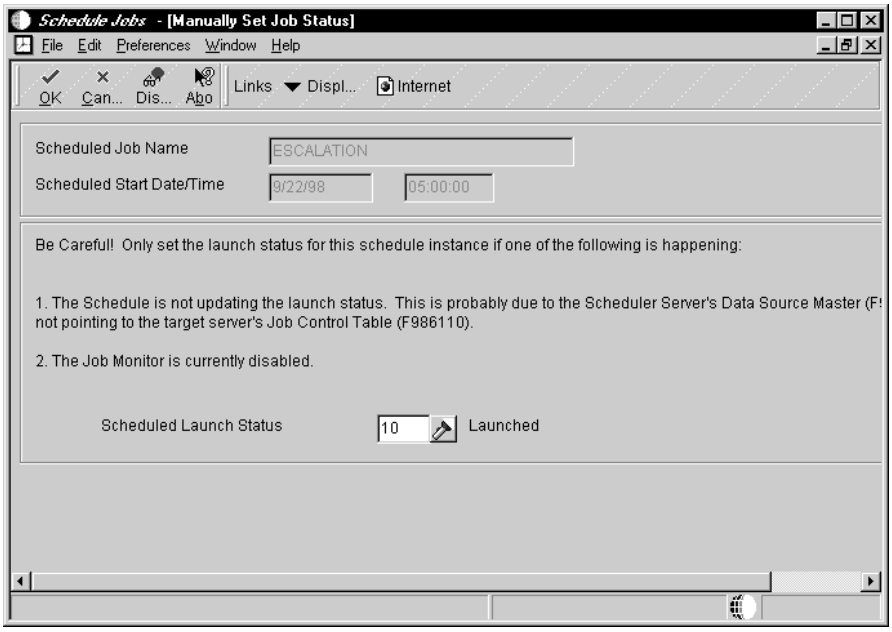
You should secure other users out of the Set Status option. Only the OneWorld administrator should have access to this option. For more information about security, see the *Security* section.

▶ To set the job status manually

On the Work with Scheduled Jobs form

1. Choose the job and then choose Job Schedule from the Row menu. The Job Schedule form appears.

2. On Job Schedule, choose the job instance for which you want to manually set the job status, and then choose Set Status from the Row menu. The Manually Set Job Status form appears.



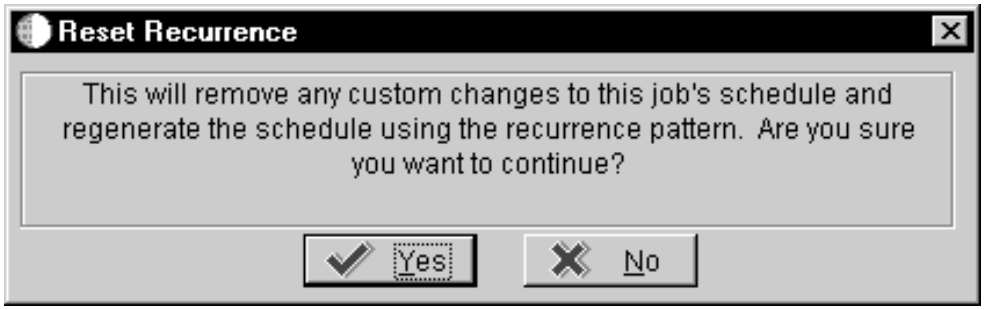
3. On Manually Set Job Status, complete the following field:
   - Scheduled Launch Status
4. Click OK.

Resetting the Job Schedule

If you make custom changes to a job's schedule and then change your mind, you can remove those changes and regenerate the job schedule using the recurrence pattern. The job schedule will be reset to the way it was before you made custom changes.

To reset the job schedule

1. From the Job Schedule form, choose Reset Schedule from the Form menu. A warning message appears.



2. Click Yes to confirm resetting the job’s schedule.

Understanding the Scheduler Server

The Scheduler server is a process that performs two distinct functions: it launches all jobs at the scheduled times, and monitors each job's progress and ending state. These functions are started by a JDENET message as defined in the following kernel type in the jde.ini file:

```
[JDENET_KERNEL_DEF10]
dispatchDLLName=jdekrnl.dll
dispatchDLLFunction=_JDEK_DispatchScheduler@24
maxNumberOfProcesses=1
beginningMsgTypeRange=2001
endingMsgTypeRange=2256
newProcessThresholdRequests=0
numberOfAutoStartProcesses=1
```

The Scheduler launches batch processes in a server/environment/user combination based on the information in the Job Master table (F91300). Once the Scheduler is started, JDENET keeps it in a kind of "wait" state by calling the Scheduler dispatch function every minute with an "idle" message. This idle message allows the Scheduler process to check whether it should launch a job or monitor its jobs that are running. In addition, JDENET will also send the Scheduler any message sent from the workstation (for example, that the new job schedules have been added).

Following are explanations of each of the elements of the Scheduler server.

Control Record

A control record is a job record in the Schedule Job Master table (F91300). It is named *SCHEDULE and is hidden from the user. The *SCHEDULE record contains information about the state of the Scheduler processes on the server, and it is the method of communicating to those processes.

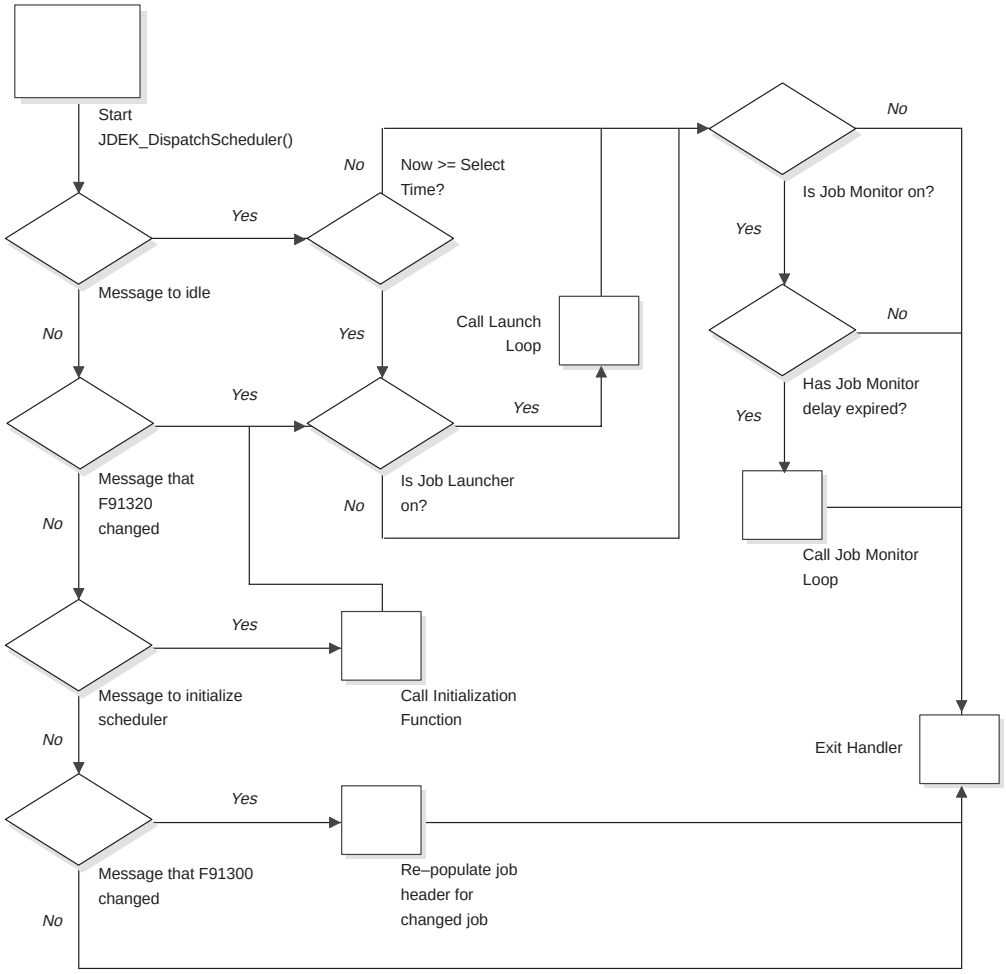
For example, when the launch loop starts on the server it will set a flag in this record to indicate that it is up and running. You can end the launch loop by toggling the corresponding end process flag (such as Job Launcher Status or Job Monitor status) from the Form menu on Scheduler Server Control. The next time the launch loop fetches the control record, it will find the flag, reset both flags, and end.

If the system does not find the control record when it is fetched, a function will be called to recreate it. In addition, if the record is found to be corrupt, the above function is called to recreate it as well. The sleep times for the job monitor will be reset to 15 minutes and the audit information in this record will be updated with the user ID set to SCHEDULER.

Dispatch Function

The dispatch function handles the incoming message from the workstation and starts the requested process. The JDENet process either sends a message to initialize the Scheduler, signals that the F91320 has changed, or gives an idle message. The idle message is sent every minute unless one of the other messages is sent. When the idle message is sent, the dispatch function checks to see if the launch loop or job monitor needs to be called. If neither do, then control is given back to JDENet.

The following illustration shows the flow of the dispatch function.

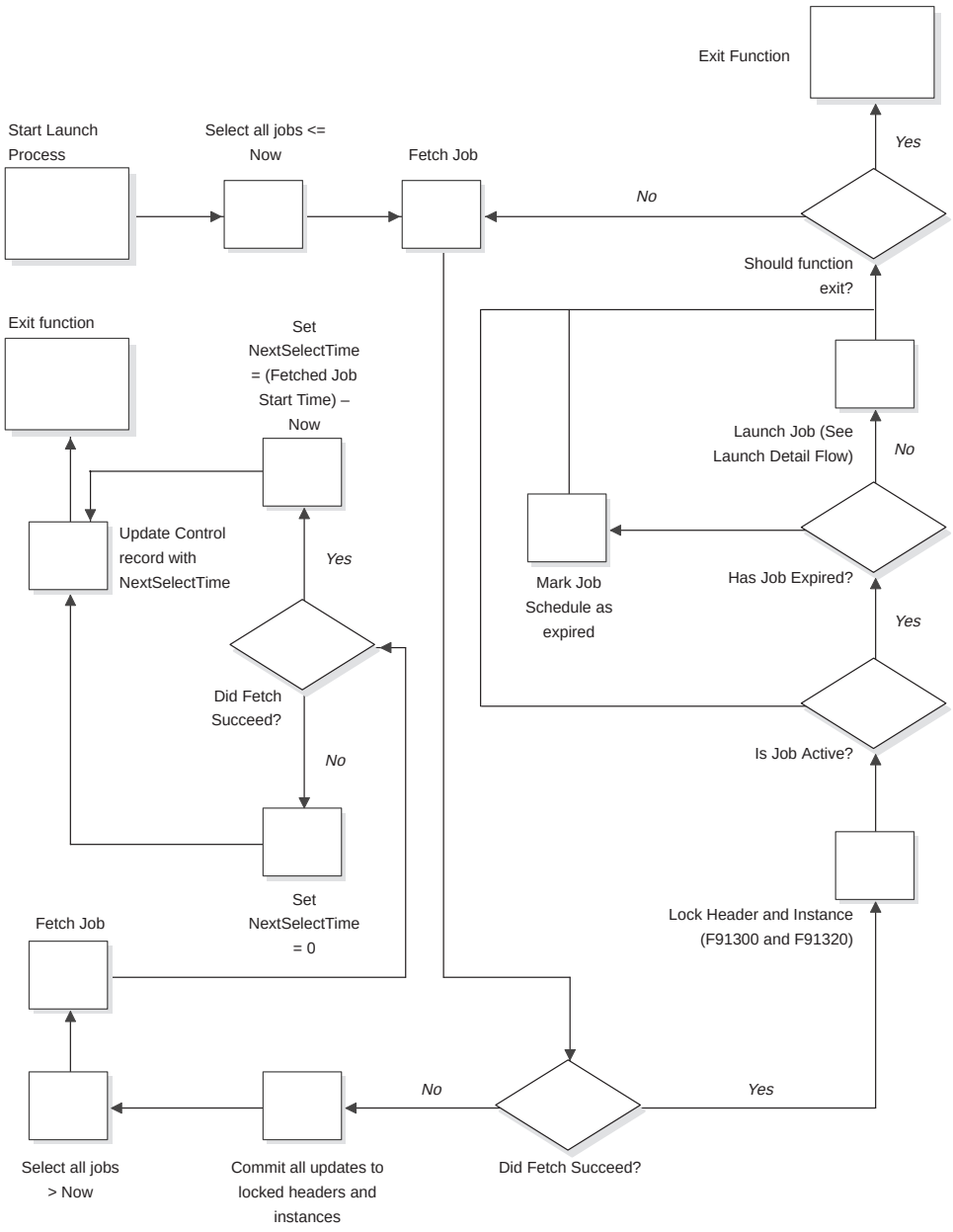


Launch Loop

This function selects all the jobs up to the current time. It then loops through the selected records and launches the active jobs if they have not expired. After launching all current jobs, the launch loop fetches all future jobs sorted by start time. If the fetch succeeds, the next select time (NST) will be set to the difference between the current time and the start of the next job. If the fetch fails, the NST will be set to zero, which indicates that this function should be run the next time any record is added to or updated by the Job Schedule table (F91320). In addition to launching jobs, the launch loop also checks the control record periodically to see if it should exit.

The launch loop also looks for updates of all the schedule instances (F91320 records) and job headers (F91300 records) it fetches. After the launch loop has processed these records, it then commits any changes and unlocks all the records.

The following illustration shows the launch loop process flow.

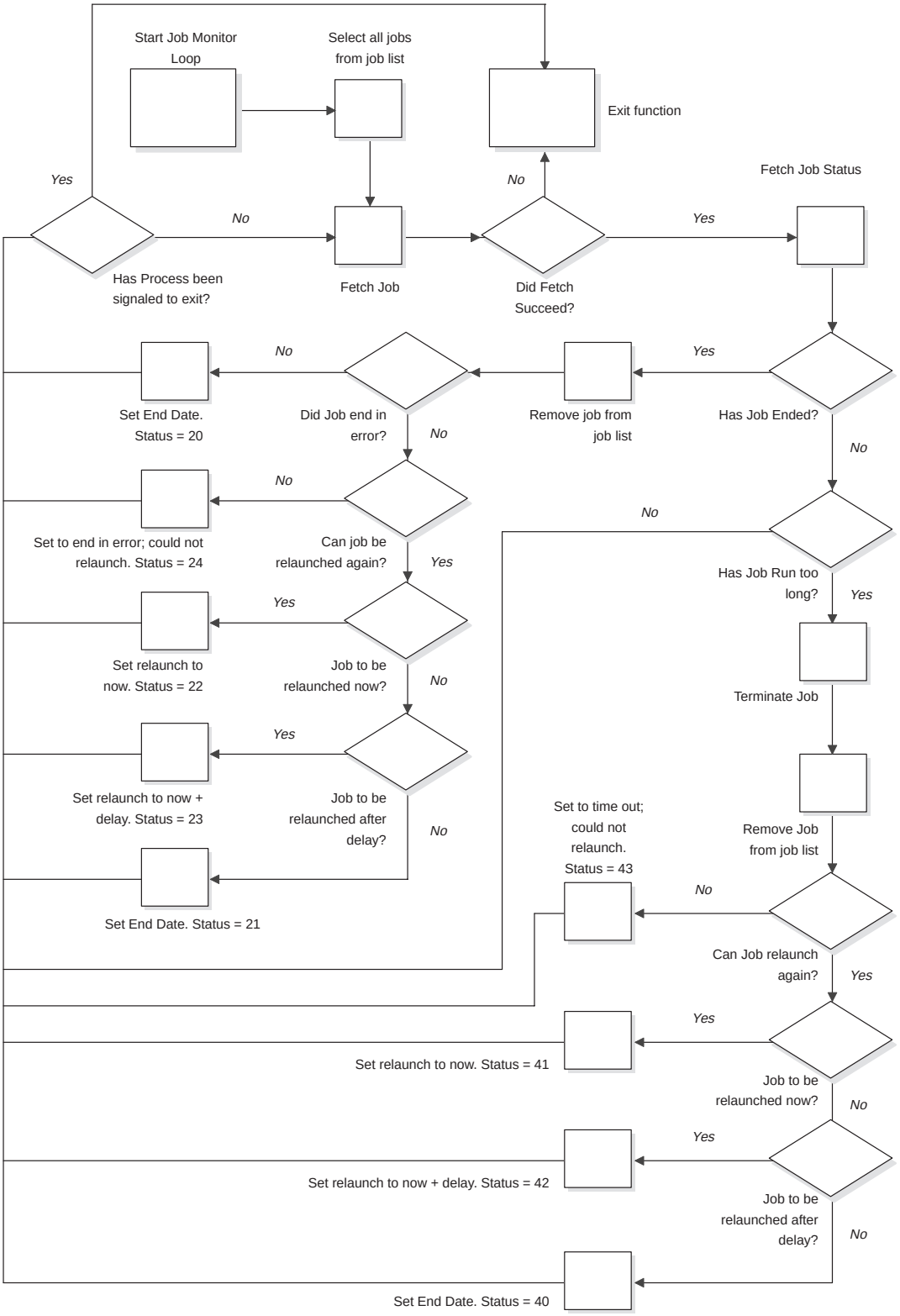


Job Monitor Loop

The job monitor loop monitors the end-states of the launched jobs and relaunches those that end in error, if requested to do so by the user. This loop cycles through the internal job list that the job launch loop populates. In addition, it terminates jobs that run too long, if requested to do so. A job cannot be relaunched for more times than specified in the job's F91300 record.

Like the launch loop, the job monitor loop periodically fetches the control record to see if it should end.

The following illustration shows the process flow of the job monitor loop.


Start Job Monitor Loop
Select all jobs from job list
Exit function
Yes
No
No
Yes
Fetch Job Status
Has Process been signaled to exit?
Fetch Job
Did Fetch Succeed?
No
Yes
Set End Date. Status = 20
Did Job end in error?
Remove job from job list
Has Job Ended?
No
No
Yes
No
Set to end in error; could not relaunch. Status = 24
Can job be relaunched again?
Has Job Run too long?
Yes
Yes
Yes
Set relaunch to now. Status = 22
Job to be relaunched now?
Terminate Job
No
Yes
Set relaunch to now + delay. Status = 23
Job to be relaunched after delay?
Remove Job from job list
No
Set to time out; could not relaunch. Status = 43
Set End Date. Status = 21
No
Can Job relaunch again?
Yes
Yes
Set relaunch to now. Status = 41
Job to be relaunched now?
No
Yes
Set relaunch to now + delay. Status = 42
Job to be relaunched after delay?
No
Set End Date. Status = 40

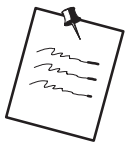
Working with the Scheduler Server

You can stop, reset, restart, and refresh the Scheduler server. For example, if the server goes offline it needs to be reset. You can also modify the server and monitor sleep time, specifying how many seconds you want JDENet to wait until it checks to see if it needs to initialize, or "wake up," the Scheduler server.

There may also be times when you need to turn on or off the Job Launcher or Job Monitor, such as in situations where you must take down the servers that you submit jobs to and want to avoid unnecessary connection errors when jobs are submitted.

You can also change the jde.ini file to enable the Scheduler to restart automatically by changing the `numberOfAutoStartProcesses` line. If you enable this feature and the server the Scheduler server is running on comes down, the Scheduler server automatically restarts when the server comes back up instead of having to be restarted manually. When the Scheduler server restarts, the Scheduler checks the control table (F91320) to determine if it should restart on that particular server. If not, the Scheduler shuts down. See *jde.ini* in this guide for more information about the autostart function.

JDENet handles the calls to initialize the Scheduler Server. As explained in *Understanding the Scheduler Server*, the JDENet process either sends a message to initialize the Scheduler to launch a job if it receives a message that the F91320 has changed, or it sends an idle message if no change is detected. For faster response time, you can lower the number of seconds you want JDENet to wait until it checks to see if there has been a change to F91320.



This application is for OneWorld administrators only. You should secure users out of this application. For more information about security, see the *Security* section.

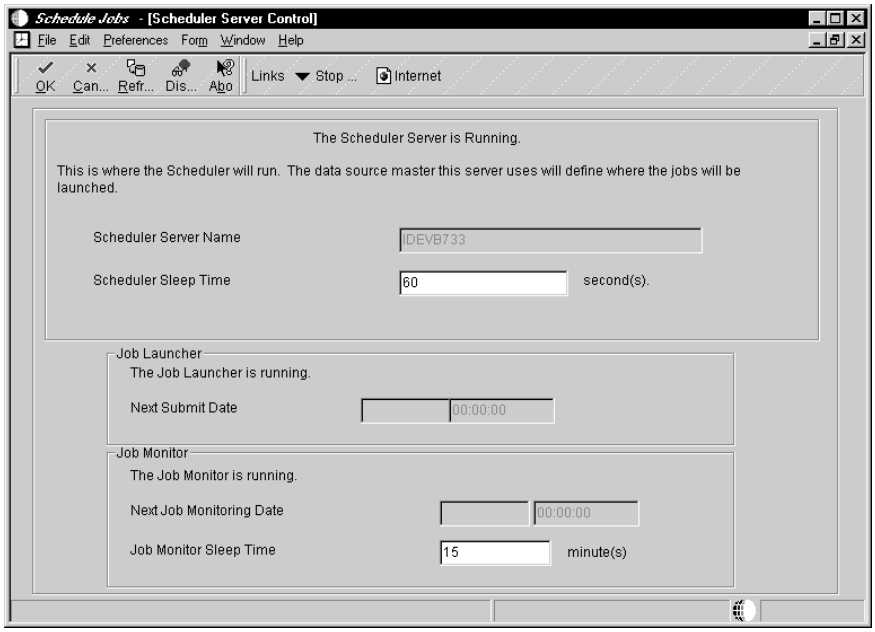
Complete the following tasks:

- Stop or restart the Scheduler server
- Pause the Job Launcher or Job Monitor
- Reset the Scheduler server
- Refresh the Scheduler server settings
- Modify the Scheduler server and monitor sleep time

▶ To stop or restart the Scheduler server

From the Job Scheduler menu (GH9015)

1. Choose Schedule Jobs (P91300). The Work with Versions form appears.
2. On Work with Versions, choose the time zone in which the scheduled jobs run, and then click Select.
3. On Work with Scheduled Jobs, choose Scheduler Server from the Form menu. The Scheduler Server Control form appears.



4. On Scheduler Server Control, do one of the following:
   - To stop the server, choose Stop Scheduler from the Form menu.
   - To restart the server, choose Start Scheduler from the Form menu.
5. Click OK.

▶ To pause the job launcher or job monitor

There might be times when you want to pause the job launcher or job monitor, such as when you want to take down the servers that you submit jobs to and want to avoid server connection errors that might occur while those servers are down.

When you pause the job launcher, the Scheduler stops looking at the F91320 table for jobs to launch. When you pause the job monitor, the Scheduler stops monitoring the status of launched jobs.

On Scheduler Server Control

1. To pause the job launcher, choose Pause Job Launcher from the Form menu.
2. To pause the job monitor, choose Pause Job Monitor from the Form menu.
3. Click OK.

To reset the Scheduler server

You reset the Scheduler server after you change the status of the Job Monitor or Job Launcher. For example, if you change the status of the Job Monitor, you would choose Reset to refresh the settings on the server.

On Scheduler Server Control

1. Choose Reset from the Form menu.
2. Click OK.

To refresh the Scheduler server settings

When you refresh the Scheduler server settings, it causes the server to refresh its cache of launched jobs and close and restart all environment and table handles. That is, it is a kind of internal refresh of the server's internal structures. You might want to refresh the Scheduler server settings if you had to restart the server.

On Scheduler Server Control

1. Choose Refresh from the Form menu.
2. Click OK.

To modify the Scheduler server and monitor sleep time

Sleep time is the time in which the Scheduler server or monitor is idle.

On Scheduler Server Control

1. Complete the following fields as necessary:
   - Scheduler Sleep Time
   - Job Monitor Sleep Time
2. Click OK.

Field	Explanation
Scheduler Sleep Time	This field indicates the number of seconds the scheduler server will sleep (or idle). For example, if this field was set to 60 seconds, the Scheduler Server will wake up every 60 seconds to check to see if it needs to launch or monitor jobs. The default is 60 and it must be greater than zero.

Field	Explanation
Job Monitor Sleep Time	This field indicates the number of minutes the job monitor will pause between job status checks.

Modifying Daylight Savings Rules

Daylight savings rules tell the system how each locale implements its daylight savings time. The Scheduler uses these rules along with the time zone information to determine when jobs should be run on a particular server.

You can add a new daylight savings rule or modify an existing one.

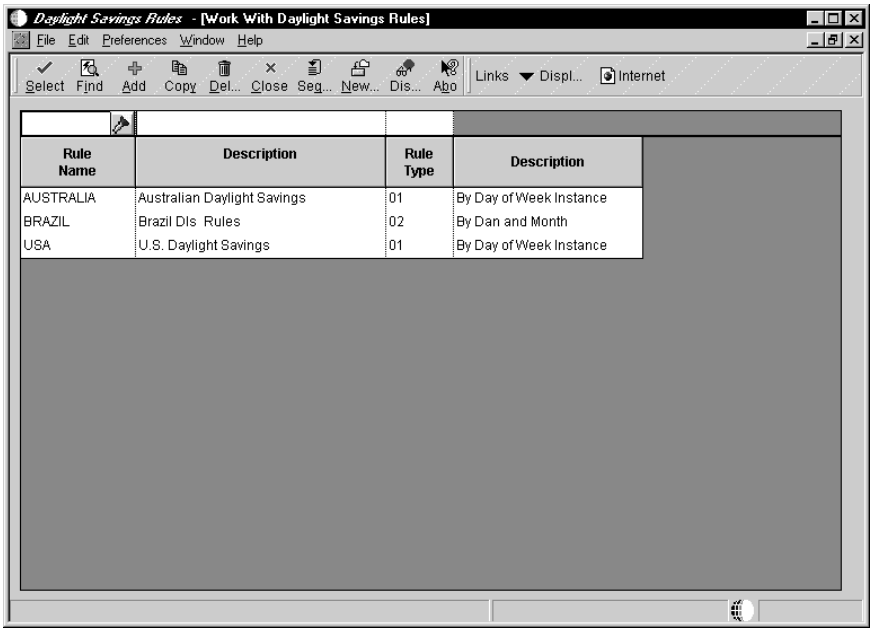
Complete the following tasks:

- Add a new daylight savings rule
- Modify an existing daylight savings rule

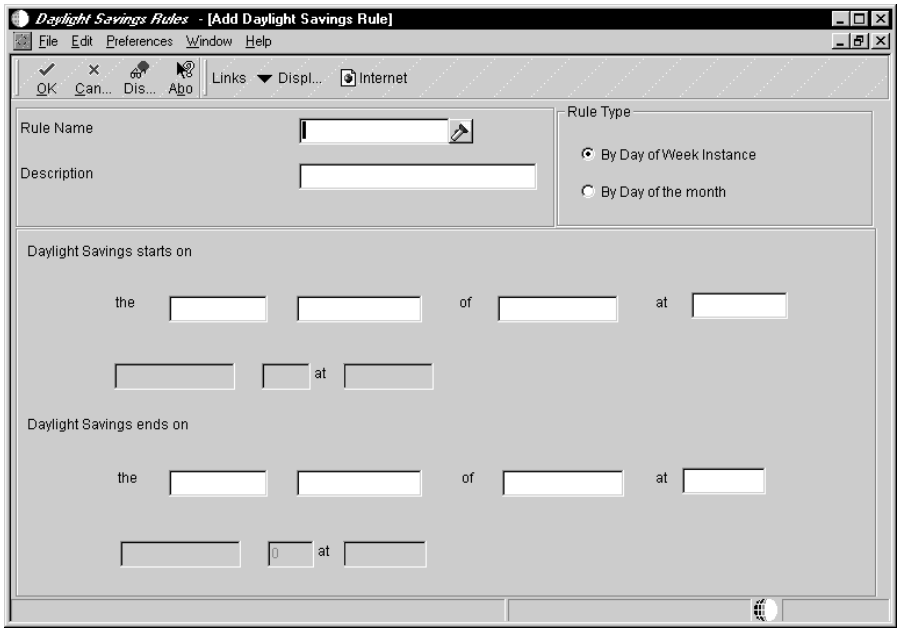
▶ To add a new daylight savings rule

From the Job Scheduler menu (GH9015)

1. Choose Daylight Savings Rules (P00085). The Work with Daylight Savings Rules form appears.



2. On Work with Daylight Savings Rules, click Add. The Add Daylight Savings Rule form appears.



3. On Add Daylight Savings Rule, complete the following fields:
   - Rule Name
   - Description
4. Choose one of the following rule types:
   - By day of the week instance
   - By day of the month
5. Enter the dates for which daylight savings time begins and ends and click OK.

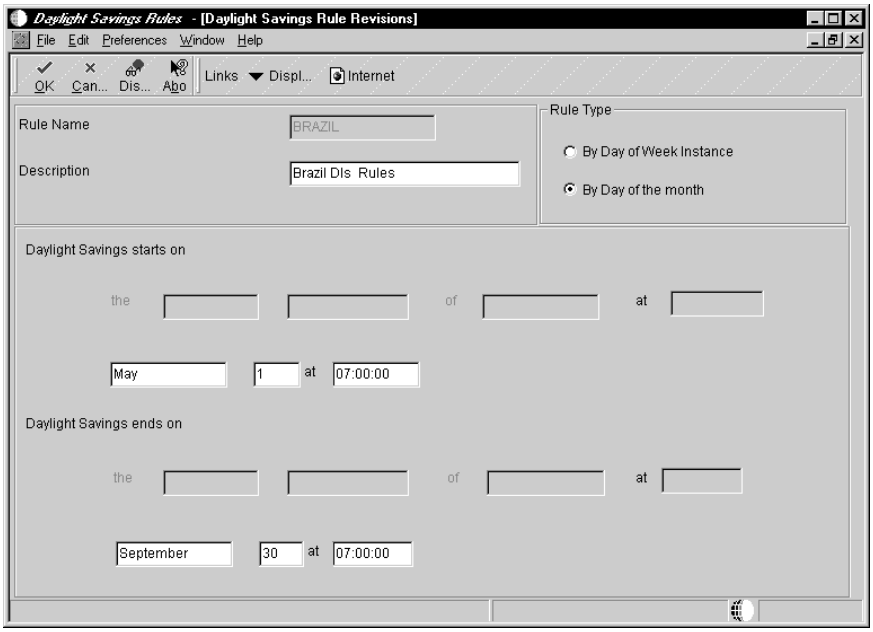
Field	Explanation
Rule Name	Unique name identifying a daylight savings rule. Daylight savings rules to adjust time for a geographic and political locale.
Description	A description, remark, name, or address.
By Day of Week Instance	A code that indicates the method that is used to determine a daylight savings rule. By Day of Week Instance indicates that daylight savings will start and stop on a certain day of the week for a certain month, such as the first Sunday of April to the first Sunday of October By Day and Month indicates that daylight savings will start and stop on a certain day of a certain month, such as April 3 to October 10

Field	Explanation
By Day of the month	A code that indicates the method that is used to determine a daylight savings rule. By Day of Week Instance indicates that daylight savings will start and stop on a certain day of the week for a certain month, such as the first Sunday of April to the first Sunday of October By Day and Month indicates that daylight savings will start and stop on a certain day of a certain month, such as April 3 to October 10

▶ To modify an existing daylight savings rule

From the Job Scheduler menu (GH9015)

1. Choose Daylight Savings Rules (P00085).
2. Choose the rule you want to modify, and then click Select. The Daylight Savings Rule Revisions form appears.



3. On Daylight Savings Rule Revisions, add a new description, if necessary, in the Description field.
4. Choose one of the following rule types:
   - By day of the week instance
   - By day of the month
5. Modify the dates and times for which daylight savings time begins and ends, and then click OK.

Running Scheduler Reports

Run the Scheduled Jobs report when you want to review a summary of scheduled jobs and their status. You can run this report based on Universal Coordinated Time or by local time through the processing options. You can also adjust for daylight savings time.

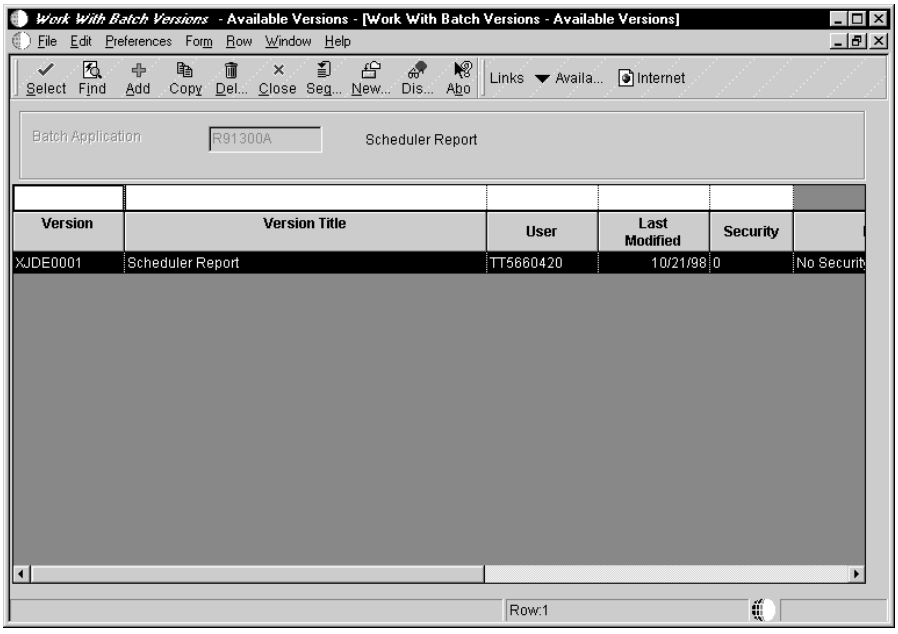
If you want to purge records from the Job Schedule table (F91320), run the Scheduler Purge Program. You can run the purge program in proof mode and final mode.

The following task describes how to run reports. The procedure is the same regardless of which report you run.

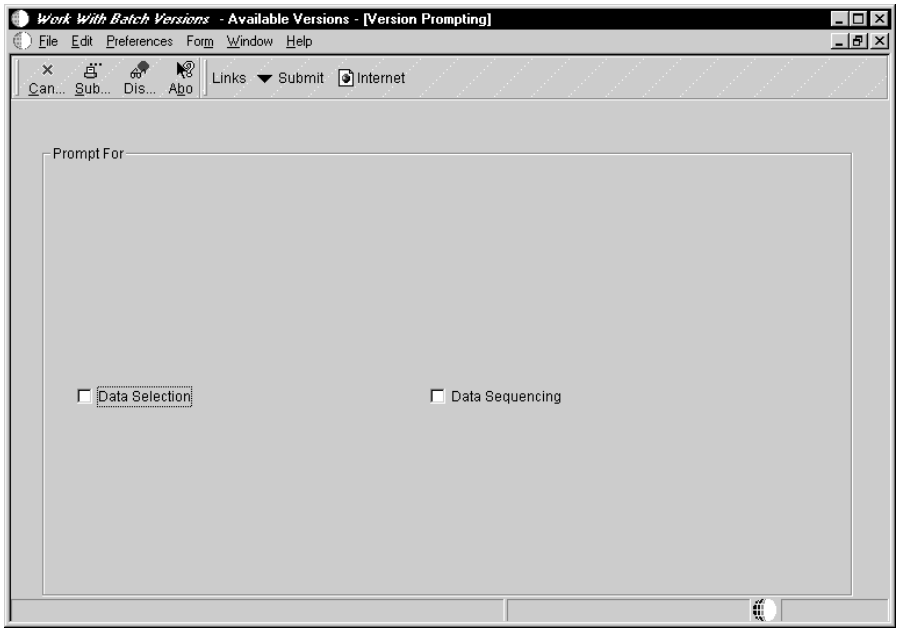
▶ To print the Scheduled Jobs or Purge Scheduled Jobs report

From the Job Scheduler menu (GH9015)

1. Choose Print Scheduled Jobs (R91300A) or Purge Scheduled Jobs (R91300B). The Work with Batch Versions form appears.



2. On the Work with Batch Versions form, choose a version in the detail area and then click Select. The Version Prompting form appears.



3. On Version Prompting, choose any of the following, if necessary, and then click the Submit button:
   - Data Selection
   - Data Sequencing
4. On Report Output Destination, choose one of the following and then click OK:
   - On Screen
   - To Printer

See Also

- *Submitting and Printing Reports* in the *Enterprise Report Writing Guide*.

Media Objects and Imaging

OneWorld's Media Objects and Imaging features allow you to attach useful information to an application, including information that might currently exist as a paper-based document. The Media Objects feature allows you to attach the information to OneWorld applications. The Imaging feature, within Media Objects, gives you flexibility to create a more efficient method of information storage.

What Are Media Objects?

Use Media Objects to link information to applications, either to individual rows in a grid or to a form. The following list describes the types of information that you can attach to a grid row or form:

Text

Media Objects provides a word processor that lets you create a text-only attachment. For example, you could use a text attachment to provide specific instructions for a form or additional information about a record.

Image

Images include files such as Windows bitmaps, Graphics Interchange Format (GIF) files, and Joint Photographic Experts Group (JPG) files. These files might represent electronically created files as well as scanned images of paper-based documents.

Object Linking and Embedding (OLE)

Media Objects can be files that conform to the OLE standard. OLE allows you to create links between different programs. Using these links, you can create and edit an object from one program in a different program. OneWorld provides the links you need to attach OLE objects.

You attach OLE media objects at the base form level. Media objects attached at this level are attached to a form and not to any data that might appear in the form. You can attach media objects to a grid row or a form, but the files themselves exist in separate directories. The only file information included with the application to which the OLE links is the path to the supporting file.

You can only use OLE objects that you properly register and install as OLE objects through OneWorld.

OneWorld Shortcuts	A OneWorld shortcut is a link that opens a OneWorld application. Within media objects, you can only attach OneWorld shortcuts, that is, you cannot attach Windows shortcuts to media objects.

System administrators can also set up templates. A template might include attachments of its own, such as images and shortcuts. For example, you can create a letterhead and a standard form for a memo. Also, you might create a shortcut to include in the template to provide access to an application that uses data specific to the information that you add to the template.

What Is Imaging?

The imaging capabilities available in OneWorld allow you to link to a third-party imaging product. Imaging systems allow you to scan and electronically store paper-based information. For example, this information might include documents such as sales orders, purchase orders, vendor invoices, and product schematics. OneWorld imaging integration includes a Media Objects viewer and a third-party product that provides scanning and searching interfaces to allow you to find and display images. OneWorld's implementation of imaging also provides a view of integrated images using the viewer of the native imaging product.

Complete the following tasks:

- ❑ Enabling OneWorld to use media objects
- ❑ Working with media object queues
- ❑ Working with media objects
- ❑ Setting up imaging

Enabling OneWorld to Use Media Objects

To use media objects, OneWorld requires a set of event rules to process the media objects. This processing includes:

- Keeping track of where the media object files are stored
- Keeping track of which media objects are attached to which OneWorld objects (rows, forms, and reports)
- Indicating which OneWorld objects have attachments
- Creating or viewing attachments

OneWorld provides standard processing for media objects which allows you to to bypass all event rules that are required to implement media objects. All of the required information is gathered for a form in Form Design Aid and does not require you to define any event rules. Standard processing does the following:

- Standardizes the usage of media objects across forms
- For any grid, places a binder clip icon on the row header if a media object is defined for that row
- For a form, places an icon in the status bar if a media object is defined for the form
- Allows you to attach documents to the form or to a row in the grid
- Allows you to double click on the binder clip in a row to view media objects for that row
- Allows you to click on the binder clip in the status bar to view media objects for the form

If you choose not to use standard processing for a form, you can still develop your own system for handling media objects using existing event rules or event rules you develop.

OneWorld uses the Media Objects Storage table (F00165) to store link records for media objects and imaging. You must define your media object data structure using a unique key structure so that the Media Objects Storage table can store data correctly. The layout of the this table is as follows:

GT || F4211Keys || The media object text

GT (generic text) is the naming convention when defining a media object data structure. The F4211Keys portion is what the system uses to access the unique media object attachment for that particular record. The keys here typically match what the unique key would be in the F4211 for each detail line. The media object text portion is the actual text attachment that would store information typed in by the user.

Before You Begin

- ❑ In order to see the media object binder clip column on a form, you must turn off the Hide Row Numbers option in the Grid properties for the form.

▶ To use standard processing for media objects

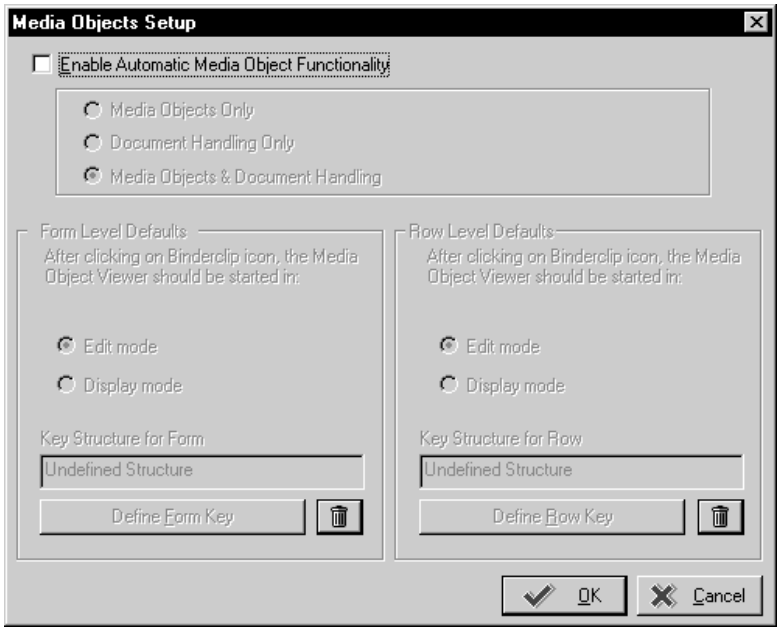
Use this procedure if you are creating new forms to enable processing for media objects, or if you want to change an existing form that is not currently enabled.

From Form Design Aid

1. Choose the grid portion of an existing or newly created form and verify that the Hide Row Numbers option is selected.

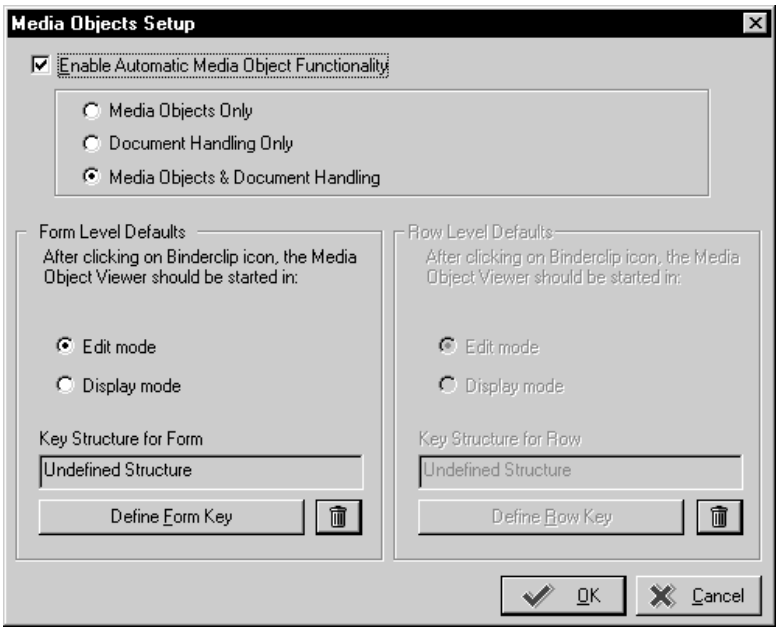
   This enables you to see the media object binder clip column on a form.

2. From the form menu, choose Media Objects Setup.



3. Click the Enable Automatic Media Object Functionality option.

   This enables imaging and opens the other fields on the form.



4. On Media Objects Setup, click one of the following:

   - Media Objects Only

     Use the Media Objects Only option if you do not want to interface with third-party products including imaging. If you select this option, you will only be able to use media objects that are defined to and supported from within OneWorld.

   - Document Handling Only

     Use the Document Handling Only option if you are developing a form that is enabled for media objects via functionality in event rules, and you wish to bypass standard processing.

   - Media Objects & Document Handling

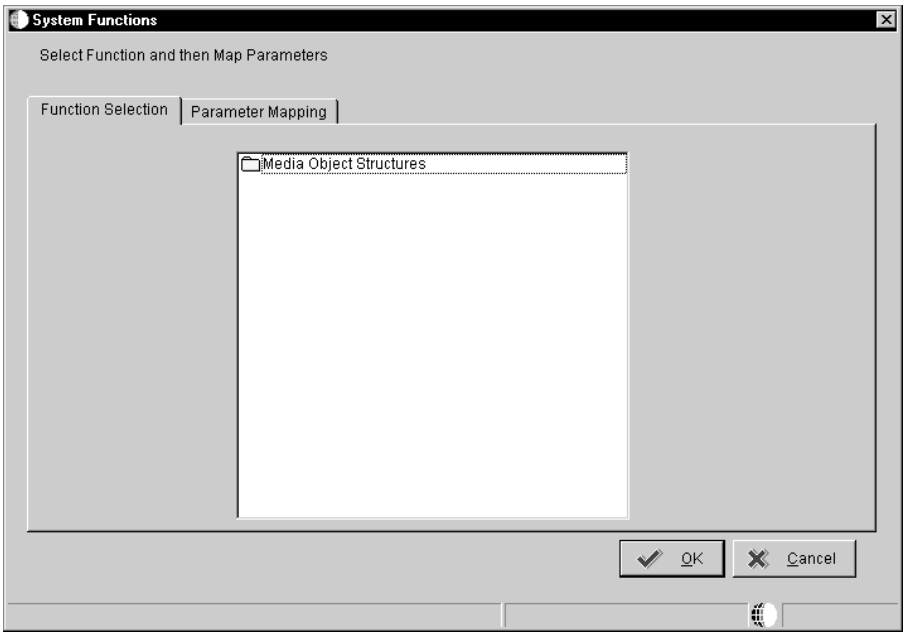
     If you wish to enable standard processing later, you must delete all of the event rules for media objects and select the Media Objects & Document Handling option.

5. Click Edit mode or Display mode.

   Edit mode allows the user to make changes, while display mode is read-only.

6. Click Define Form Key.

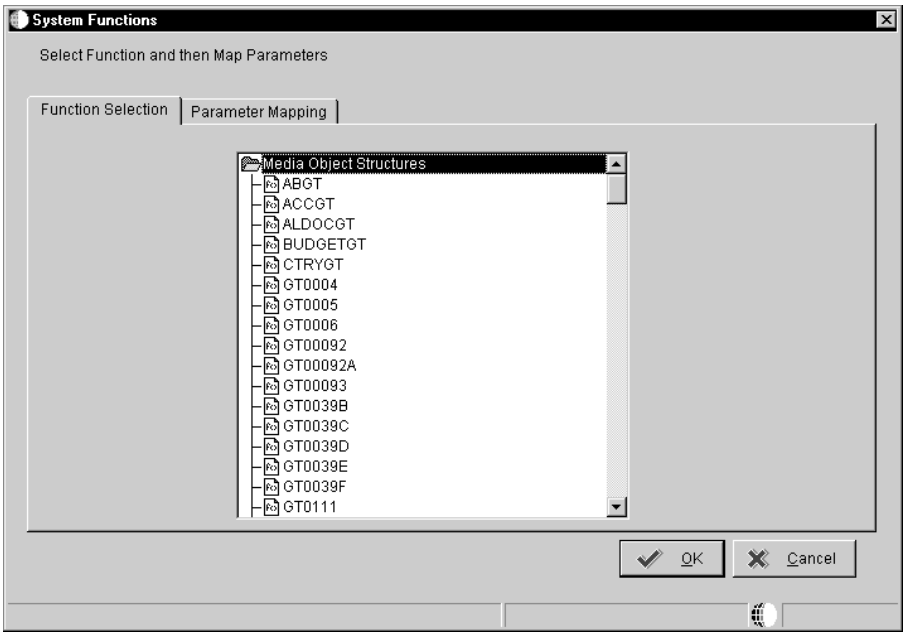
   The System Functions form appears. This form is identical to the parameter definition form used to define system functions in Event Rules, except only the Media Object Structures header is displayed.



The key structure is used to store link records in the Media Objects Storage file. These links are critical to proper functioning of Imaging.

7. Double-click on the Media Object Structures folder.

   A list of all of the currently defined data structures for Media Objects appears.



8. Select the appropriate structure and define it as described in the *OneWorld Tools Guide*.

Working with Media Object Queues

OneWorld media object queues allow the storage location of media objects to be tracked by reference as opposed to physical network location. This allows for easy administration of media location. For example, the location for media objects on your server can change, and the change only has to be reflected in one place in OneWorld.

You must define a media object queue to identify the pointer to the location where the actual image files or OLE objects reside. For OLE objects, the name of the queue must be OLEQUE. The media object queues must reside on a Windows NT machine. A valid network-qualified path must exist for this machine. You can use the deployment server or any other Windows NT machine. If you use the deployment server to store media object queues, remember that you must make the machine available for use daily by OneWorld applications.

Media object queues provide the system administrator the ability to easily manage the storage of media objects in a OneWorld configuration. Within OneWorld, you must set up media object queues in order to use images that are outside the imaging product's domain (for example, scanned images). You can set up media object queues for the following types of objects:

- Image objects (actual files)
- OLE objects (links to files)
- URLs (Internet addresses)

What Are Image Media Objects?

Image media objects are individual files that are accessed and viewed using a third-party imaging product. These objects are stored in locations defined with a name and a network-qualified path. For example, if all the images for financial applications are stored in a directory on the network called \\server1\financials\images, an image media object queue could be defined as follows:

- Path: \\server1\financials\images
- Name: FIN_IMAGES.BMP

What Are OLE Media Objects?

OLE media objects are individual objects that are created and viewed using a OLE-compliant application outside of OneWorld. In OneWorld, the OLE object attached to a row or form is actually a link to the OLE object that resides in a media object queue. The distinction between OLE objects and non-OLE objects is important because, other than graphic files, you cannot attach non-OLE objects from OneWorld if they are not compliant. Examples of valid OLE objects are Microsoft Windows OLE-compliant applications such as Word, Excel, Powerpoint, and Visio. Other examples might include sound or video files (.wav or .avi extensions).

What Are URL Media Objects?

URL media objects are Internet addresses that point to Web sites that are identified by industry standard uniform resource locations (URLs). When defined in the media object table, users can click on these addresses and connect to Internet locations.

What Are the Media Object Tables?

Media object queues typically represent network directory locations for OneWorld media object files, such as OLE objects and images. The two media object tables are Media Object Queues (F98MOQUE) and Imaging Constants (F98101).

The media object queues are stored in the Media Object Queues table which, along with the Imaging Constants table, should be located in the system data source. The Media Object Queues table contains the associated key value of the data record the media object is attached to, the image reference, and the OLE reference. The image reference and the OLE reference are queue names. The queue name is used to access the Media Object Queue table for the location of the OLE object or image.

What Are OneWorld Text Items?

Text items that you create using the OneWorld media objects word processor do not require Media Object Queues. The Media Objects Storage table (F00165) contains the associated key value of the data record to which the text media object is attached, and the text itself. Text items that originate from applications external to OneWorld (for example, Microsoft Word or Wordpad) must be stored as OLE objects.

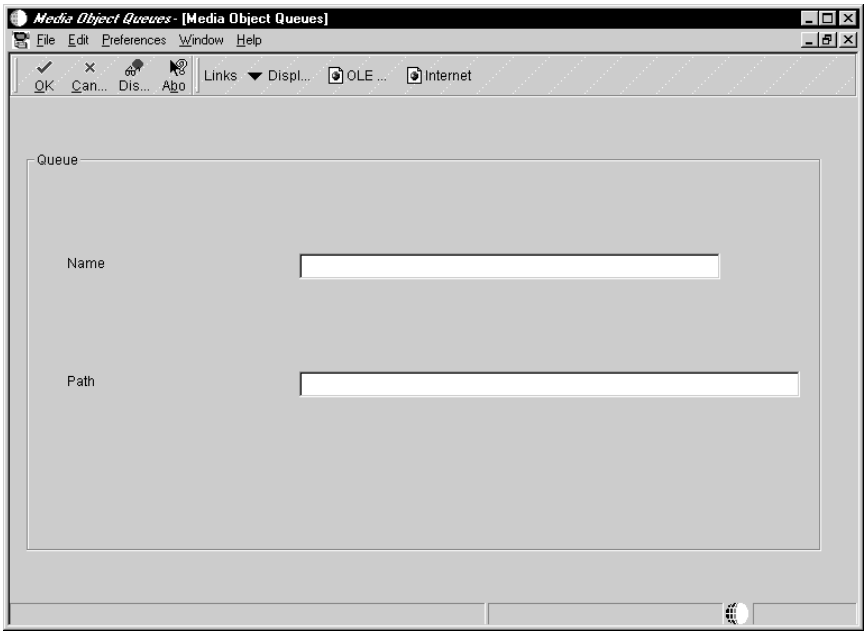
Working with media object queues includes the following tasks:

- Add a media object queue
- Define the location of a media object queue
- Delete a media object queue

To add a media object queue

On System Administration Tools (GH9011)

1. Choose Media Object Queues (P98MOQUE).
2. Click Add.



3. On the Media Object Queues form, complete the following fields:
   - Name
   - Path

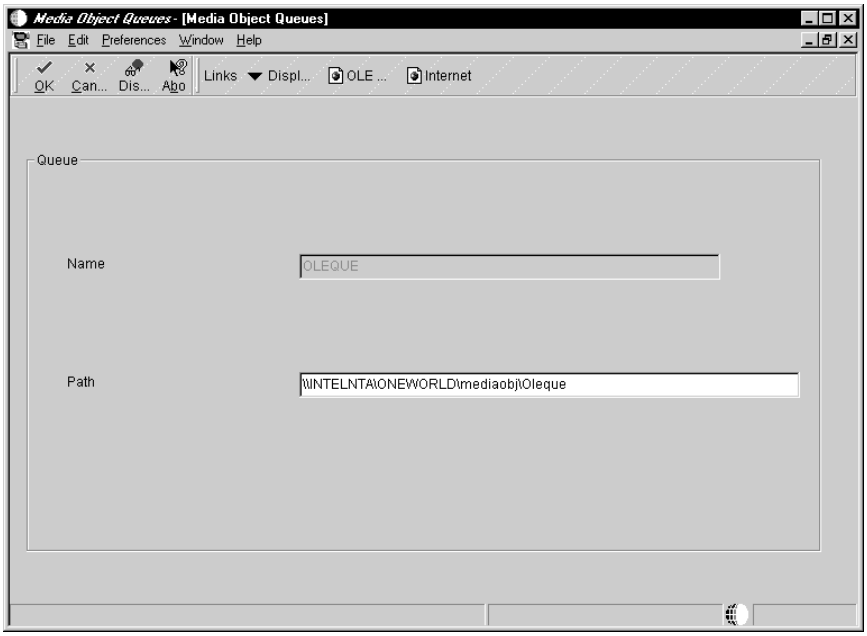
Field	Explanation
Name	Identifies the name of a media object queue. This queue name is the first half of a properly defined media object queue, where the second half is the media object queue path. OLEQUE is a reserved queue name for OneWorld media objects. You must use this name as the default queue name in the OLE object attachment mode of Media Objects. It is mandatory that this queue name be defined in order to use OLE object attachments. The queue path and queue name are stored in the F98MOQUE table. The location of this table is controlled by OCM. The system reads the F98MOQUE table to determine name of the queue and the location of its associated OLE objects, images, or URLs.
Path	Identifies a network-qualified or local path that points to the actual location of OLE objects, images, or URLs. This is the second half of a properly defined Media Object queue, where the first half is the Media Object queue name. For example, a valid queue path for a network location might be \\server1\share3\images\financial. The queue path and queue name are stored in the F98MOQUE table. The location of this table is controlled by OCM. The system reads the F98MOQUE table to determine name of the queue and the location of its associated OLE objects, images, or URLs.

▶ To define the location a media object queue

On System Administration Tools (GH9011)

1. Choose Media Object Queues (P98MOQUE).
2. If an OLE queue does not exist, click Add.
3. Complete the following fields:
   - Name
   - Path

4. If you want to change an existing media object que, click Find to display a list of queue names and their paths.
5. Choose the queue name you wish to modify and click Select.



6. Change the information in the Path field to reflect the new location.

▶ To delete a media object queue

On System Administration Tools (GH9011)

1. Choose Media Object Queues (P98MOQUE).
2. Choose the queue name you wish to delete and click Select.
3. From the Form menu, choose Delete.

   Deleting a media object queue deletes the definition of the queue, not the associated path and objects themselves.

Working with Media Objects

You can use the Media Objects feature to add text, graphics, and other objects to forms and records. For example, you can use a note to explain special circumstances surrounding a journal entry. Or you can attach drawings, animations, and other types of objects to forms and records. A pop-up menu provides access to established templates for attachments and an option to set the properties for the Media Objects form.

When you attach a media object to a form, the attachment might not be available if you access different data on the form. For example, if you attach a media object to a detail form that contains data for Order Number 2002, this attachment does not appear on the detail form that appears when you access data for Order Number 3003. The base form, which in this case is a detail form, is the same for both Order Numbers, but the data associated with the form is specific to each Order Number. The Order Number represents the key to the location where an attachment is stored.

OneWorld supports the attachment of OLE objects. OLE allows you to create links between different programs. Using these links, you can save an object from one program in a different program. OneWorld provides the links you need to attach OLE objects. You can attach OLE objects as media objects and at the base form level. When you attach an object at the base level of the form, you attach the object to the form and not to any data that might appear in the form.

If attachments exist for a form, when you open the form, a paper clip icon automatically appears at the right of the status bar. For an OLE object attached at the base form level, a document icon appears at the right of the status bar.

Initially, grid rows do not indicate whether attachments exist for the corresponding records. You can perform a search on every record that OneWorld displays or search on each record individually to determine whether attachments exist for records.

The Text feature includes a word processor that lets you create, view, edit, and delete notes. Also, when you type a Web address, for example, http://www.jdedwards.com, the text becomes an active link to the Web site at the address.

When you create a text attachment, you can also set up templates. You can use templates to create a format for a frequently used media object.

This topic contains the following:

- Checking for attachments
- Attaching media objects to forms and rows

- ❑ Deleting media objects
- ❑ Working with templates
- ❑ Working with the properties of media objects
- ❑ Attaching OLE objects at the base form level

Checking for Attachments

When the detail area is initially loaded, OneWorld displays records based on a user's search criteria. Users are not able to see if any attachments might exist in the detail area. To verify whether attachments exist, the user must press the Checking for Attachments icon (magnifying glass/paper clip) at the top left portion of the row header column. However, each time the user pages down to see additional detail information in the detail area, Oneworld only refreshes the data. Therefore, the user must again click the Checking for Attachments magnifying glass/paper clip icon for new data.

An alternative, but less efficient, method to check for attachments is to take the mouse pointer and "hover" over a particular row header. To "hover" means to locate the mouse pointer at a row header and wait a second. If an attachment exists, OneWorld displays the attachment icon.

Attaching Media Objects to Forms and Rows

Use the attachments feature to attach text, photos, drawings, spreadsheets, video images, application shortcuts, and sounds to forms and to grid rows. For example, you might attach the image of an invoice to a data entry record, attach a legal document to a record that describes a contractual agreement, or attach text that describes a process on a form.

The attachments feature is not available on all forms. On some forms you might be able to attach media objects to rows but not to the form itself.

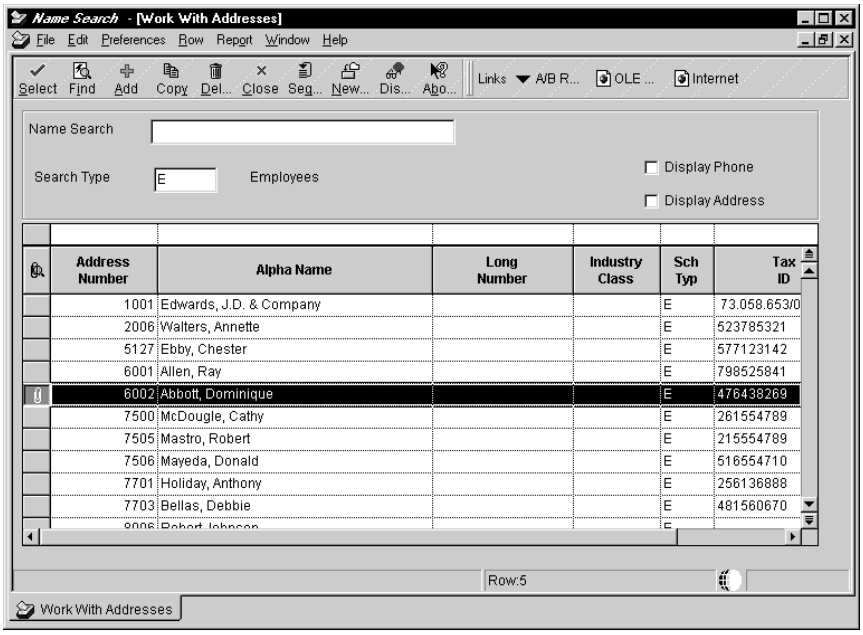
Complete the following tasks:

- Attach text
- Attach an image
- Attach an OLE object
- Attach a shortcut

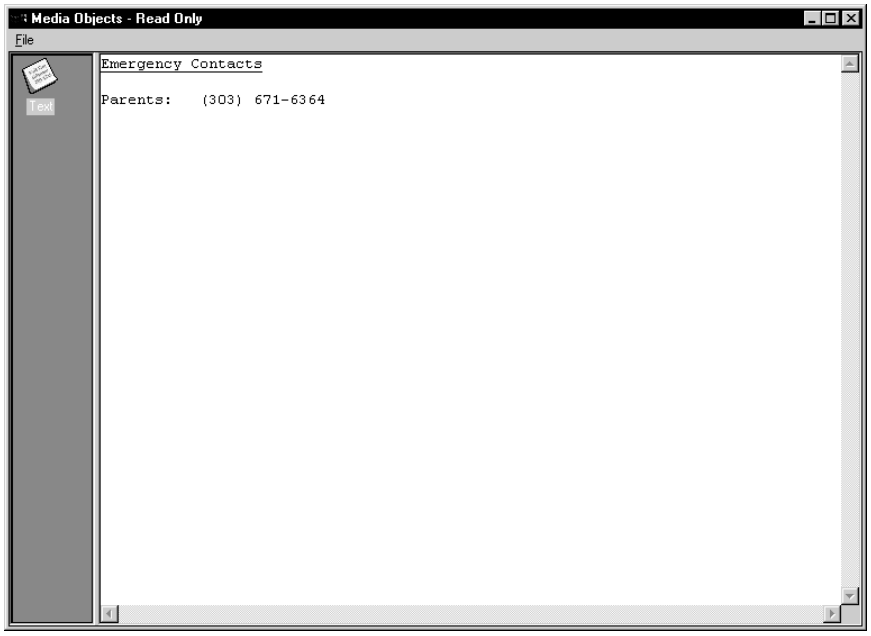
To attach text

When you create a text attachment, you can format the paragraphs and run a spell check. OneWorld also supports the attachment of OLE objects.

On a form where attachments are available



1. Do one of the following:
   - To attach text to a form, from the Form menu, choose Attachments.

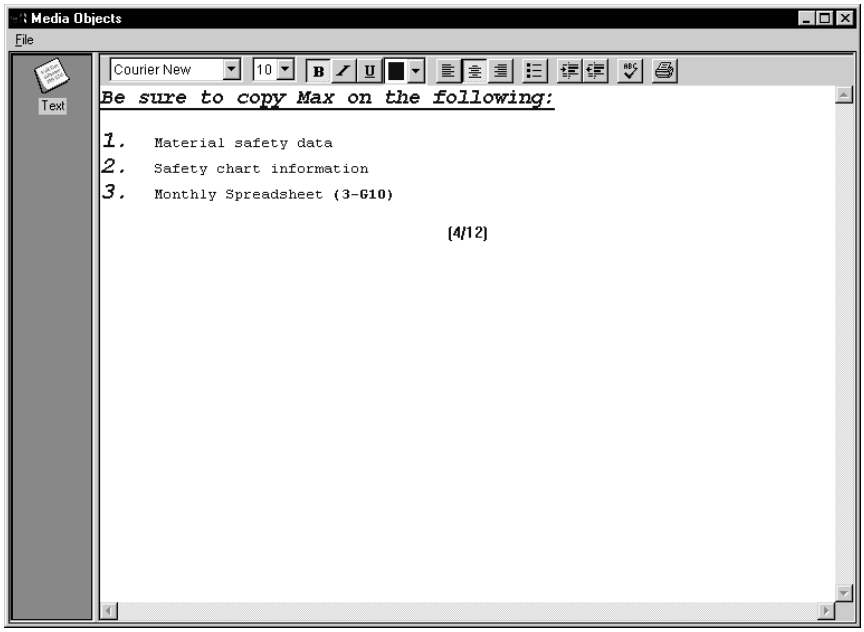


- To attach text to a grid row, choose the row, then from the Row menu, choose Attachments.

The Media Objects workspace appears.

The Media Objects workspace is split into two panels. The left panel is the icon panel and the right panel is the viewer panel. Icons for any objects previously attached appear in the icon panel.

2. Do one of the following:
   - From the File menu, choose Add, then Text.
   - In the icon panel, click the right mouse button, choose Add, then Text from the pop-up menu.



3. In the viewer panel, type the desired text.

   You can use the formatting tools at the top of the viewer panel to format the text of your note.

4. When you finish, from the File menu, choose Save & Exit.

▶ To attach an image

On a form where attachments are available

1. Do one of the following:
   - To attach an image to a form, from the Form menu, choose Attachments.
   - To attach an image to a grid row, choose the row, then from the Row menu, choose Attachments.

     The Media Objects workspace is split into two panels. The left panel is the icon panel and the right panel is the viewer panel. Icons for any objects previously attached appear in the icon panel.

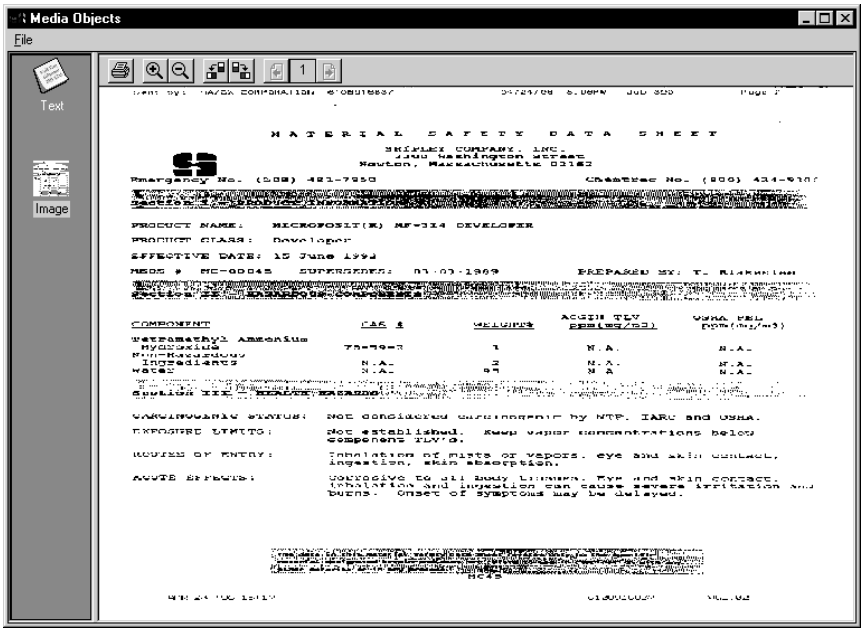
2. On Media Objects, do one of the following:
   - From the File menu, choose Add, then Image.
   - In the icon panel, click the right mouse button, choose Add, then Images from the pop-up menu.

3. Complete the following options:
   - Queue Name
   - Files of Type
   - Preview

   Toggle this option to display or hide a preview of each image listed.

4. Choose an image, then click OK.

   The image appears in the viewer panel.



5. When you finish, from the File menu, choose Save & Exit.

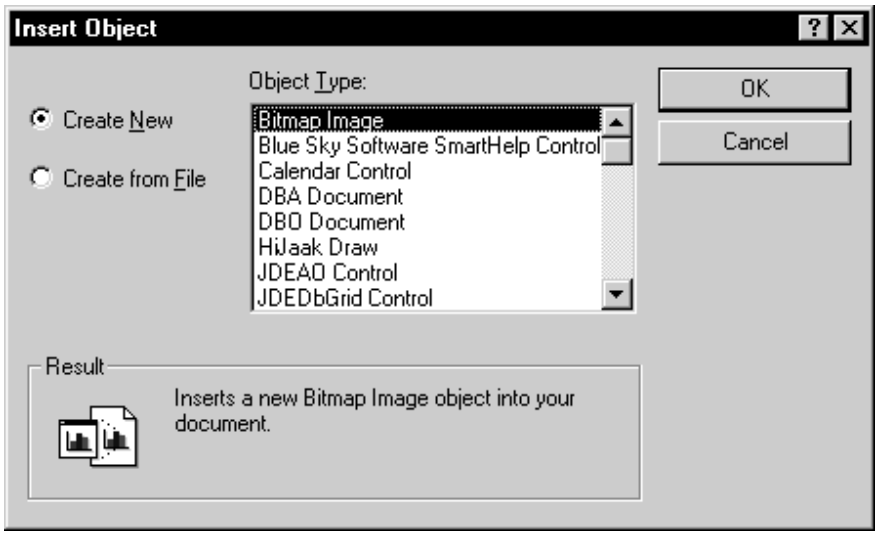
Field	Explanation
Queue Name	The name of the directory where the image file exists.
Files of Type	The list of file extensions that the system supports. For example, file types might include .bmp for a Windows bitmap, .gif for a graphics interchange format file, and .jpg for a joint photographic experts group file.

To attach an OLE object

On a form where attachments are available

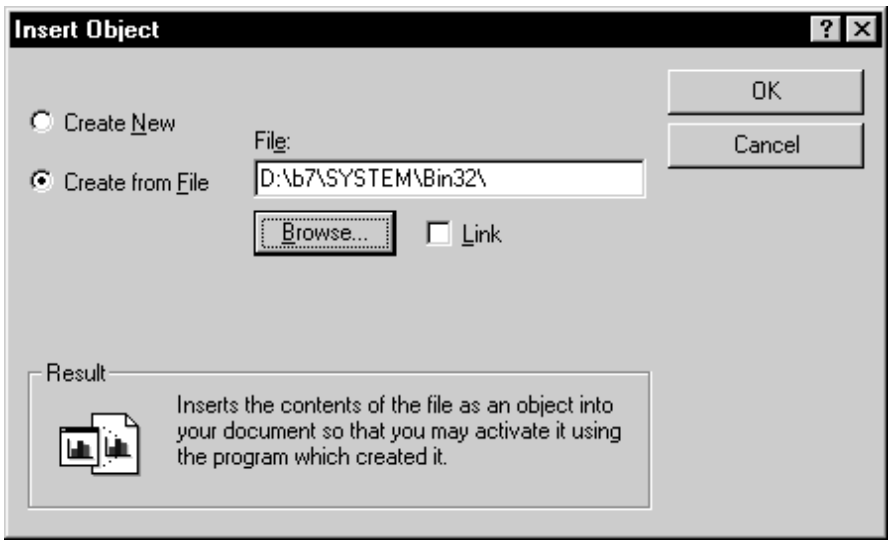
1. Do one of the following:
   - To attach an OLE object to a form, from the Form menu, choose Attachments.
   - To attach an OLE object to a grid row, choose the row, then from the Row menu, choose Attachments.

     The Media Objects workspace is split into two panels. The left panel is the icon panel and the right panel is the viewer panel. Icons for any objects previously attached appear in the icon panel.
2. On Media Objects, do one of the following:
   - From the File menu, choose Add, then OLE.
   - In the icon panel, click the right mouse button, choose Add, then OLE from the pop-up menu.



3. On Insert Object, to create a new object, click Create New, and choose an object type from the list of supported types.

   The object types you see depend on the products available to you.
4. Create your object, then go to Step 7.



5. To attach an existing object, click Create from File and complete the following fields:

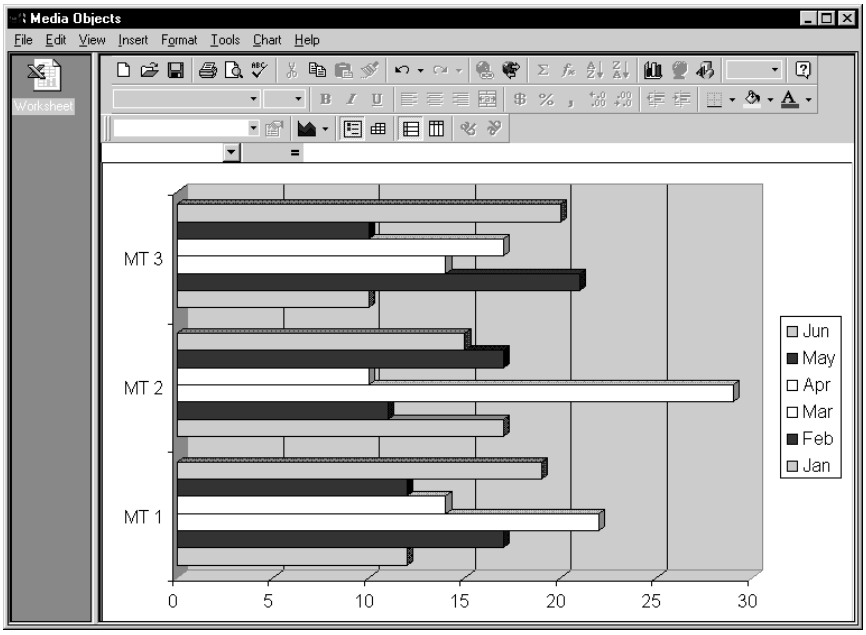
   - File

     Enter the name of the object you want to attach. Click the Browse button to search your system for available files.

   - Link

     Turn this option on to create a link between OneWorld and the attached object. Turn this option off to embed the object in the OneWorld row or form.

   The menu bar displays the menus for the application from which you call the object. For example, if you select a Microsoft Excel document, the menus for Excel appear on the menu bar.



6. Edit the object in the viewer panel to your preference.
7. When you finish, from the File menu, choose Save & Exit.

▶ To attach a shortcut

Include a shortcut to provide access directly from a record to an associated OneWorld application. The shortcut must already exist before you can attach it.

On a form where attachments are available

1. Do one of the following:
   - To attach a shortcut to a form, from the Form menu, choose Attachments.
   - To attach a shortcut to a grid row, choose the row, then from the Row menu, choose Attachments.
2. On Media Objects, do one of the following:
   - From the File menu, choose Add, then choose Shortcut.
   - In the icon panel, click the right mouse button, choose Add, then Shortcut from the pop-up menu.
3. On Open, browse through the available shortcut files, then choose the appropriate shortcut.

Your shortcut appears in the viewer panel.

4. When you finish, from the File menu, choose Save & Exit.

Deleting Media Objects

When you no longer need an attachment, use the Delete feature on Media Objects to remove the object. When you delete text, the text is permanently erased. When you delete images, OLE objects, and shortcuts you remove the attachment of the file to the record or form. The object itself is not deleted.

▶ To delete an object

On a form where attachments are available

1. Do one of the following:
   - To delete an attachment to a form, from the Form menu, choose Attachments.
   - To delete an attachment to a grid row, choose a row with a paper clip icon, then from the Row menu, choose Attachments.
2. Choose the appropriate icon in the icon panel, then choose Delete from the File menu.

   The icon disappears from the icon panel.

3. When you finish, from the File menu, choose Save and Exit.

Working with Templates

A template is a special kind of document that provides basic tools for shaping a final document. For departments, templates can provides means to help users format and standardize text attachments by reusing existing definitions. For example, a template can give Customer Support users a head start on developing support call information. The Media Objects Templates form allows you to attach create, modify, and delete templates.

Complete the following tasks:

- Attach a template
- Create a template
- Modify a template
- Delete a template
- Delete a template on Media Objects

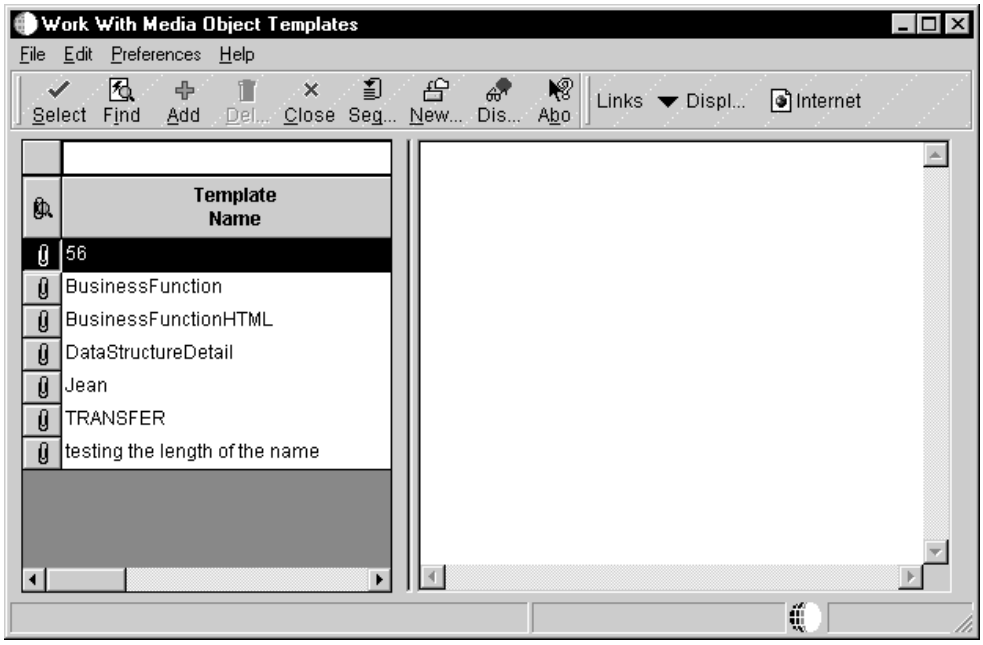
▶ To attach a template

On a form where attachments are available

1. Do one of the following:
   - To attach a template, from the Form menu, choose Attachments.
   - To attach a template to a grid row, choose a row with a paper clip icon, then from the Row menu, choose Attachments.
2. On Work With Media Objects, in the icon panel, click the right mouse button and choose Templates from the pop-up menu.
3. On Work With Media Objects Templates, click Find.

   You can use the query by example line to refine your search.

   If Find does not display any templates, then none exist. See *To create a template* for instructions for creating one.

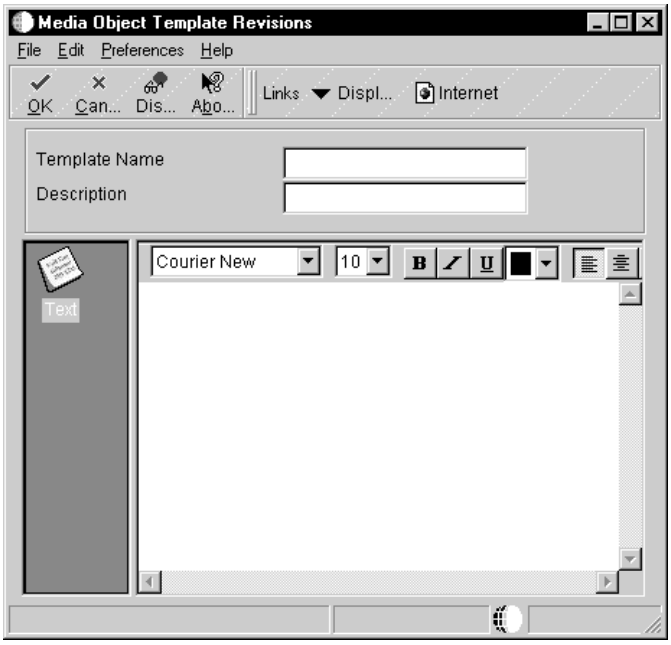


4. To preview a template, double-click the paper clip icon in the row header.
5. Choose the grid row for the template you want to attach, then click Select.
6. OneWorld adds a text media object using the template you selected. You can edit the text in the workspace.
7. When you finish, from the File menu, choose Save & Exit.

▶ To create a template

On a form where attachments are available

1. Do one of the following:
   - To create a template, from the Form menu, choose Attachments.
   - To create a template for a grid row, choose a row with a paper clip icon, then from the Row menu, choose Attachments.
2. On Work With Media Objects, in the icon panel, click the right mouse button, then choose Templates from the pop-up menu.
3. On Work With Media Objects Templates, click Add.



4. On Modify Template, complete the following fields, then enter your template information into the workspace:
   - Template Name
   - Description

▶ To modify a template

On a form where attachments are available

1. Do one of the following:
   - To modify a template, from the Form menu, choose Attachments.
   - To modify a template for a grid row, choose a row with a paper clip icon, then from the Row menu, choose Attachments.
2. On Media Objects, in the icon panel, click the right mouse button, then choose Templates from the pop-up menu.
3. On Media Objects Templates, click Find.

   You can use the Query by Example line to refine your search.

4. Choose the grid row for the template you want to modify, then click Select.
5. On Modify Template, modify the template as necessary, then click OK.

▶ To delete a template

On a form where attachments are available

1. Do one of the following:
   - To delete a template, from the Form menu, choose Attachments.
   - To delete a template for a grid row, choose a row with a paper clip icon, then from the Row menu, choose Attachments.
2. On Media Objects, in the icon panel, click the right mouse button, then choose Templates from the pop-up menu.
3. On Media Objects Templates, click Find.

   You can use the Query By Example line to refine your search.

4. Choose the grid row for the template you want to delete, click Delete, then on Confirm Delete, click OK.

▶ To delete a template on Media Objects

On a form where attachments are available

1. Do one of the following:
   - To delete a template on Media Objects, from the Form menu, choose Attachments.
   - To delete a template for a grid row on Media Objects, choose a row with a paper clip icon, then from the Row menu, choose Attachments.
2. On Media Objects, in the icon panel, choose the text icon for the template, then choose Delete from the File menu.

   The template and the text icon disappear.

Working with the Properties of Media Objects

Each type of object has unique properties. You can view an object's properties in the pop-up menu that appears when you click the right mouse button in the icon panel on Media Objects. This pop-up menu provides you with the option to view and, for some objects, to change the properties of an object.

Complete the following tasks:

- Set Media Objects properties
- Set text properties
- Set image properties
- View OLE properties
- Set shortcut properties

▶ To set Media Objects properties

This procedure describes how to set properties that affect how the Media Objects feature handles all media objects.

On Media Objects

1. In the icon panel, click the right mouse button, then choose Properties.
2. On Properties, review the following on the Key Information tab:
    - Technical information about the key for the form.
3. Click the Flags tab to review the following information:
    - Allow Text Items
    - Allow Image Items
    - Allow OLE Items
    - Allow RTF Text
    - Show Text Item on Open
    - Read Only

▶ To set text properties

On Media Objects

1. In the icon panel, click the right mouse button over a text icon, then choose Properties from the pop-up menu.
2. On the text properties form, review the following fields on User Audit Information:
    - Created by
    - Date Created
    - Time Created
    - Updated By

- Date Updated
- Time Updated

3. Click the Printing Information tab, then do the following if necessary:
   - Click the Check to print before report item option
   - Complete the Effective From field
   - Complete the Effective To field

▶ To set image properties

On Media Objects

1. In the icon panel, click the right mouse button over an image icon, then choose Properties from the pop-up menu.
2. On the image properties form, review the following fields:
   - File Name
   - Queue Name
   - Queue Path
3. To give the image a title, complete the following field:
   - Description

▶ To view OLE properties

On Media Objects

1. In the icon panel, click the right mouse button over a shortcut icon, then choose Properties from the pop-up menu.
2. On the OLE properties form, review the following fields:
   - File Name
   - Queue Name
   - Queue Path

▶ To set shortcut properties

On Media Objects

1. In the icon panel, click the right mouse button over an OLE object icon then choose Properties from the pop-up menu.

2. On the shortcut properties form, review the following fields on the General tab:
    - Menu Name
    - Selection
    - Icon File
    - Icon Index
3. Do the following if necessary:
    - Click the Colors tab to set the color for the shortcut hypertext.
    - Click the Fonts tab to set font properties such as size, family, bold, italics, underline, and strikeout.

Attaching OLE Objects at the Base Form Level

At the base level of a standard form, you can attach OLE objects using the OLE Objects button on the Links toolbar. Menu bars and toolbars appear on all standard forms. When you attach an OLE object at the base level of a form, rather than associating the attachment with a record, the OLE object attaches only to the form. No matter what record appears on the form, the OLE object you attach using the OLE Objects button will always appear when you open the form.

Complete the following tasks:

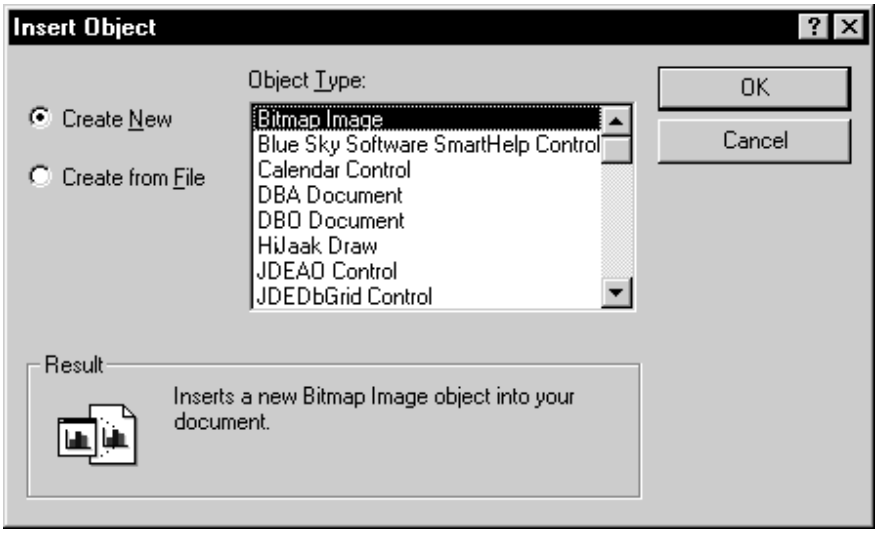
- Attach OLE objects at the base form level
- Delete OLE objects at the base form level

▶ To attach OLE objects at the base form level

On any standard form

1. Do one of the following:
    - Click the OLE Objects button on the Links toolbar.
    - From the Preferences menu, choose OLE Objects.

2. On the OLE Objects form, do one of the following:
   - From the File menu, choose Add Object
   - In the icon panel, click the right mouse button, then choose Add Object



3. On Insert Object, to create a new object, choose one from the list of supported types.

   The object types you see depend on the products available to you.

4. Create your object, then go to Step 7.

5. To attach an existing object, click Create from File and complete the following fields:

   - File

     Enter the name of the object you want to attach. Click the Browse button to search your system for available files.

   - Link

     Turn this option on to create a link between OneWorld and the attached object. Turn this option off to embed the object in the OneWorld row or form.

   The menu bar displays the menus for the application from which you call the object. For example, if you select a Microsoft Excel document, the menus for Excel appear on the menu bar.

6. Edit the object in the viewer panel to your preference.
7. When you finish, from the File menu, choose Save & Exit.

▶ To delete OLE objects at the base form level

On any standard form

1. Do one of the following:
   - Click the OLE Objects button on the Links toolbar
   - From the Preferences menu, choose OLE Objects
   - Click the document icon at the right of the status bar
2. On the OLE Objects form, choose the object and do one of the following:
   - From the File menu, choose Delete Object
   - In the icon panel, click the right mouse button, then choose Delete Object
3. When you finish, from the File menu, choose Save & Exit.

Setting Up Imaging

One way to attach images to OneWorld forms and grid rows is to use the Image function of the Media Object feature. However, this solution is not designed for use with sophisticated document handling systems. Currently, OneWorld directly supports products from two imaging vendors:

- Optika Imaging Systems, Inc.
- Lava

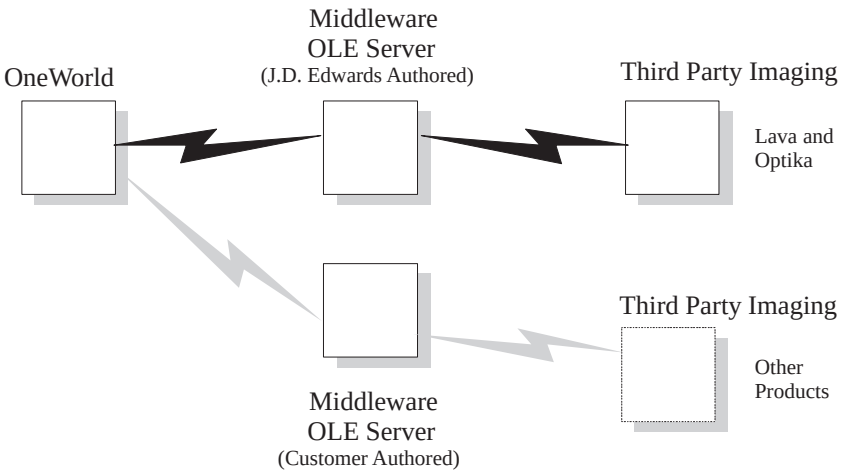
OneWorld uses the OLE client/server model to interface with third-party document handling systems. This includes the OLE client interface and the OLE server. For the currently supported imaging systems, OneWorld meets these minimum design goal tasks:

- Search. The search mechanism locates a document stored in an indexing system of a document handling system. The search mechanism navigates the storage structures of the document handling system so that the user can easily find a particular document or set of documents.
- Link. Upon a successful search operation, the link mechanism returns the unique document identifier to OneWorld. This identifier is stored with the transaction.
- View. The view mechanism passes the unique document identifier to a document viewing mechanism so that the user can view the document.

Customers with requirements for third-party imaging systems other than those currently supported can design custom OLE automation servers for interfacing purposes. The OLE server can be written in any OLE-compliant language. J.D. Edwards has a published set of APIs to enable you to develop compatible middleware applications. The Published APIs are in a Windows help file which is installed with OneWorld.

Typical Flow for Imaging Systems

This illustration shows how OneWorld supports third-party imaging products through a middleware OLE server layer. Customers can also create their own OLE servers to support additional imaging systems.



Cnc/Imaging

Typical Use for Imaging

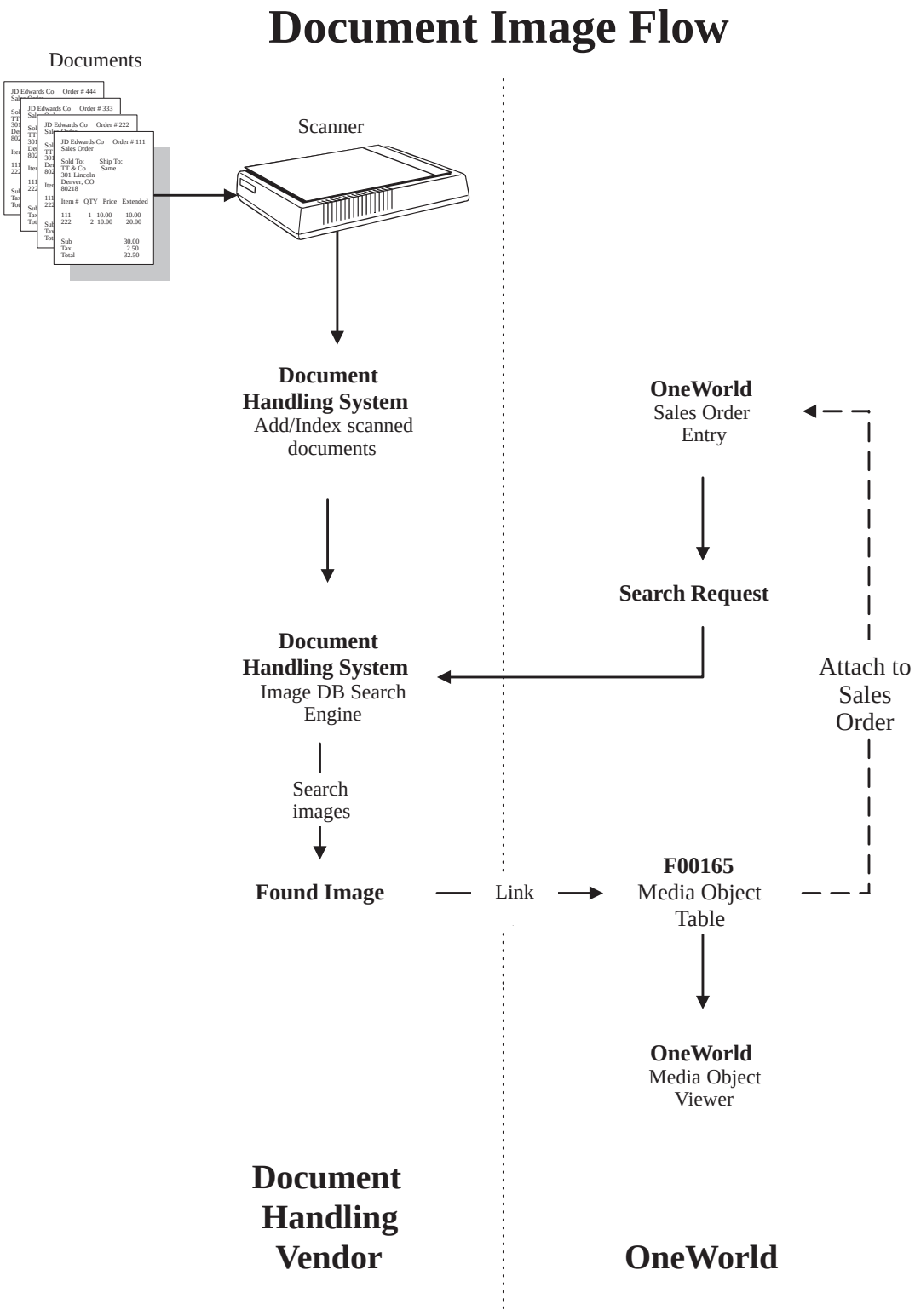
You can use imaging with a document handling system. With this system you can automatically scan and catalog documents. The system indexes the images so that you can recall them based on certain sets of criteria. For example, you might index images according to type, department, and date. You can recall, view, and analyze an image at any given time. In a transaction entry scenario, scan a paper-based file when the document enters the mail room so that a data entry clerk can recall the image to use as a source document.

OneWorld has the ability to retrieve and view documents based on selection criteria that a user defines. A linking system ties the OneWorld transaction to the document for later retrieval and reference. You can attach a OneWorld transaction identifier with the scanned image in the document handling system to allow a user to directly access an application from the image.

You set up imaging by enabling the imaging at the system level. For an imaging system to be enabled, it must have a registration record in the Imaging Constants (F98101) table.

Imaging Process Flow

This diagram provides a graphical example of a typical imaging scenario:

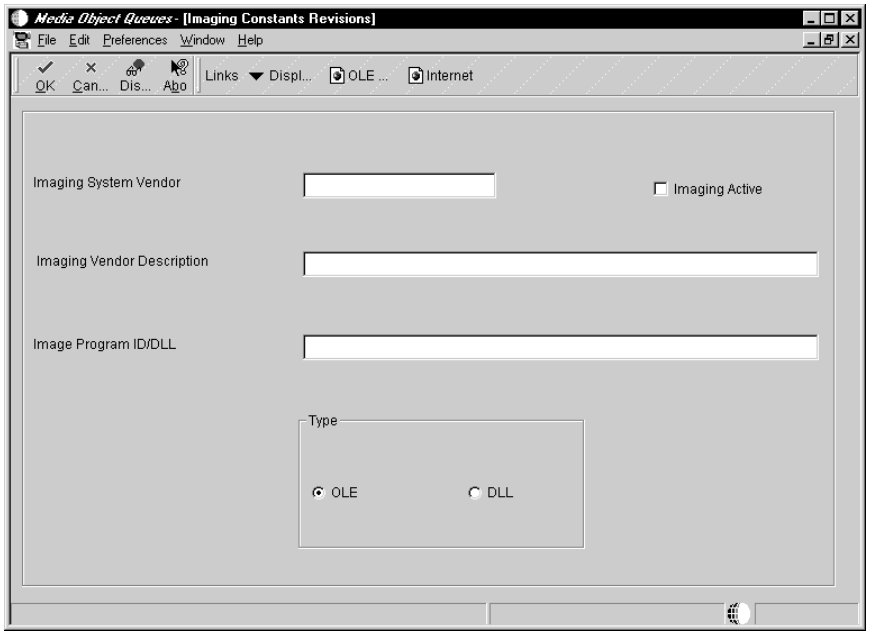


To enable media objects

On System Administration Tools (GH9011)

1. Choose Media Object Queues (P98MOQUE).
2. On Work With Media Object Queues, from the Form menu, choose Imaging.

   This calls the Imaging Constants application (P98101).



3. On Imaging Constants Revisions, complete the following fields:
   - Imaging System Vendor
   - Imaging Vendor Description
   - Image Program ID/DLL
   - Imaging Active
   - OLE

Field	Explanation
Imaging System Vendor	The name of the imaging system vendor you are using on your system.
Imaging Vendor Description	The description of the imaging system vendor you are using on your system.

Field	Explanation
Image Program ID/DLL	If the image type is an OLE, enter the Program ID that will be used to uniquely identify the imaging system in the system registry. If the image type is a DLL, enter the imaging system DLL name.
Imaging Active	Indicates whether an imaging sytem is currently active for the system.
OLE	Indicates the type of interface used by the imaging system. OLE Use the OLE type for 16-bit imaging systems. These systems typically use an OLE server with an executable (.exe) or library (.dll) file type. The OLE server must conform to the interface defined by the OneWorld published API. DLL Use the DLL type for 32-bit imaging systems.
DLL	Indicates the type of interface used by the imaging system. OLE Use the OLE type for 16-bit imaging systems. These systems typically use an OLE server with an executable (.exe) or library (.dll) file type. The OLE server must conform to the interface defined by the OneWorld published API. DLL Use the DLL type for 32-bit imaging systems.

Universal Table Browser

If you want to view the data in tables in different databases, you can use the OneWorld tool called Universal Table Browser. This tool lets you verify the existence of data in a table as well as determine the table's structure. The Universal Table Browser uses JDEBASE APIs to retrieve data from the database, making it independent of the database you access.

This section contains the following:

- Viewing the data in tables

Viewing the Data in Tables

If you want to view the data in tables in different databases, you can use the OneWorld tool called Universal Table Browser. This tool lets you verify the existence of data in a table as well as determine the table's structure. The Universal Table Browser uses JDEBASE APIs to retrieve data from the database, making it independent of the database you access.

Note: All column and row security that you set up through Security Workbench apply to the Universal Table Browser. See *Working With Security Workbench*.

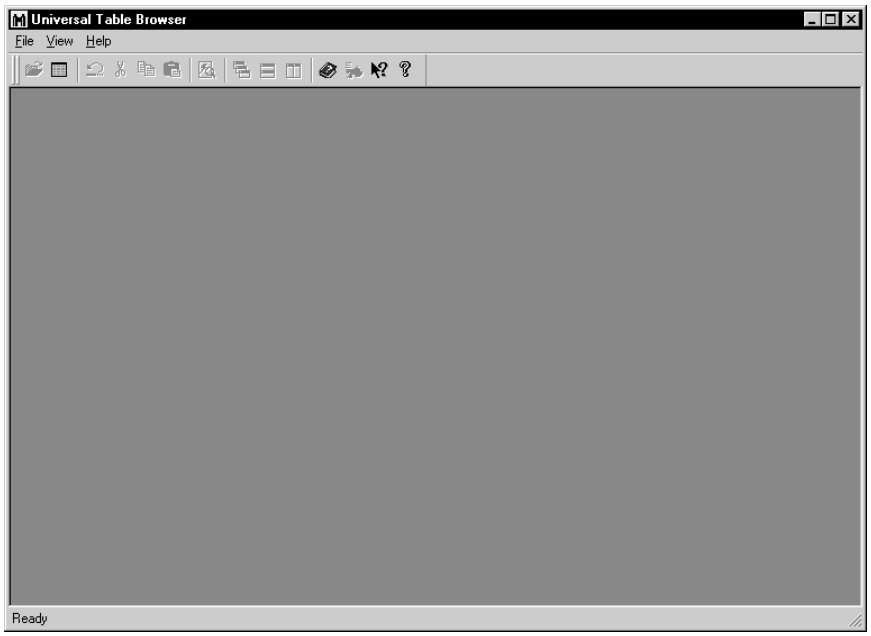
Complete the following tasks:

- View the data in tables
- View column properties in a table

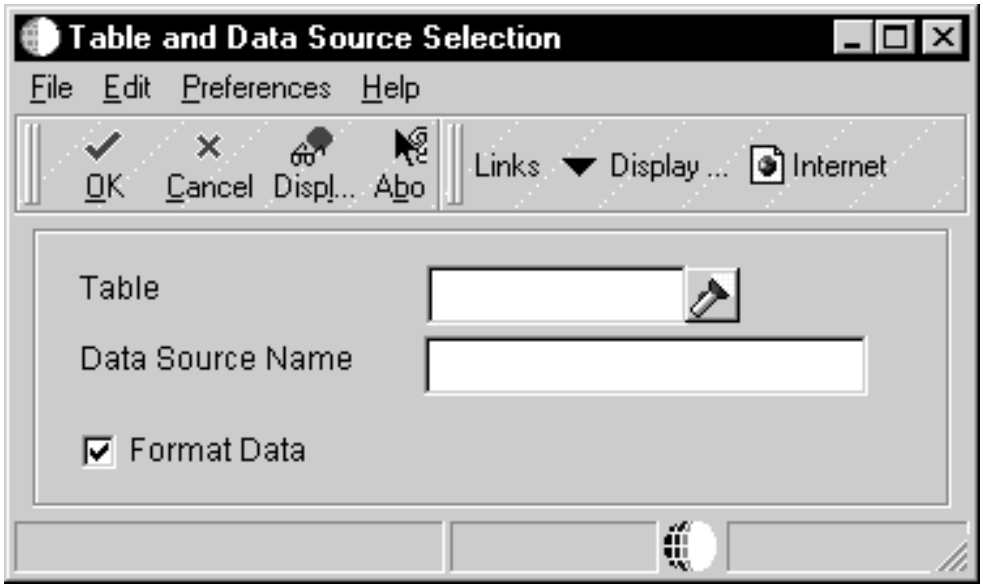
▶ To view the data in tables

On Cross Application Development Tools (GH902)

1. Choose Universal Table Browser.



2. On Universal Table Brower, choose Open Table from the File menu.



3. Complete the following required fields:
   - Table
   - Data Source Name
4. Complete the following optional field:
   - Format Data

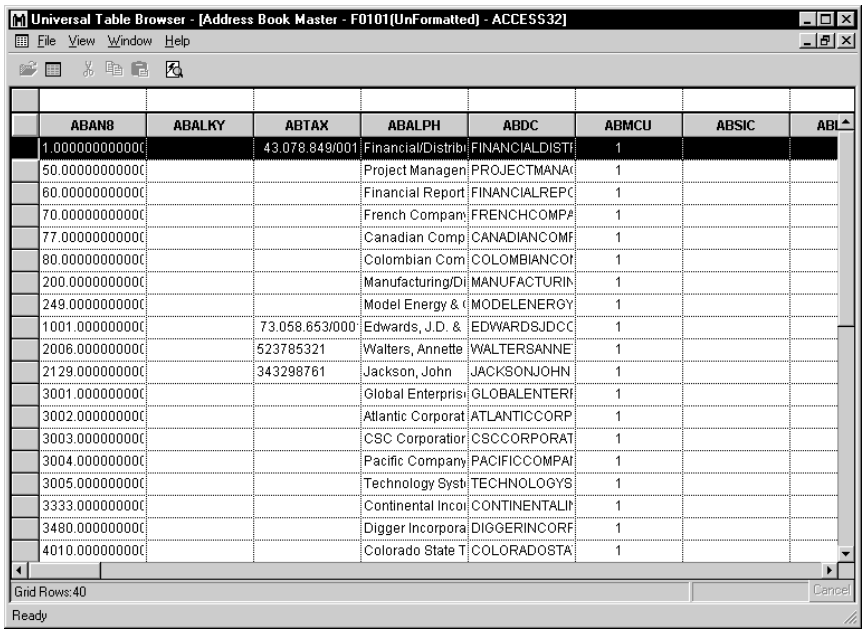
The Query By Example (QBE) function is operable as in any other OneWorld application. For example, you can enter “>50” in the ABAN8 column QBE to display records with an Address Book Number greater than 50. You could also enter “F*” in the ABALPH column QBE to display records with an Alpha Name that begins with the letter F.

The column sequence and column width functions are operable as in any other OneWorld application. You can rearrange the columns. For example, you might want to move a column that you use often from the end to the front, or move a column next to an associated column. You can also size the columns.

Field	Explanation
Object Name	The OneWorld architecture is object based. This means that discrete software objects are the building blocks for all applications, and that developers can reuse the objects in multiple applications. Each object is stored in the Object Librarian. Examples of OneWorld objects include: • Batch Applications • Interactive Applications • Business Views • Business Functions • Business Functions Data Structures • Event Rules • Media Object Data Structures . . . . . *Form-specific information* . . . . . Indicates the name of the OneWorld table. For example, F0101 is the Address Book master. You can use the visual assist to enable the search and select form and locate a table.
Data Source Name	A valid data source in which the table resides. This default value is obtained from the OCM settings in the environment to which the user is signed on. Use the visual assist to enable the data search form and select any OneWorld data source.
Format Data	Indicates whether you want the Universal Table Browser to format portions of the data (default) or whether you want to view raw data. Formatted. The Universal Table Browser displays the data according to the specifications of the OneWorld data dictionary item. For example, assume that the data item PROC is a numeric field of size 15, with 4 display decimals. For a value of 56.2185, the Universal Table Browser displays a formatted value (using the data dictionary editing) as 56.2185, even though this value is stored in the database as 562185. Nonformatted. The Universal Table Browser displays the data according to the specification of the database and the data item type (such as numeric) from which the data came. For example, assume the table data item, PROC, is a numeric field stored in the database. Depending on the database, this field might default to a size of 32 with a precision of 15 being a numeric data type. Because OneWorld does not store the decimals in the database, a value 56.2185 would be stored and displayed in the database as 562185.000000000000000.

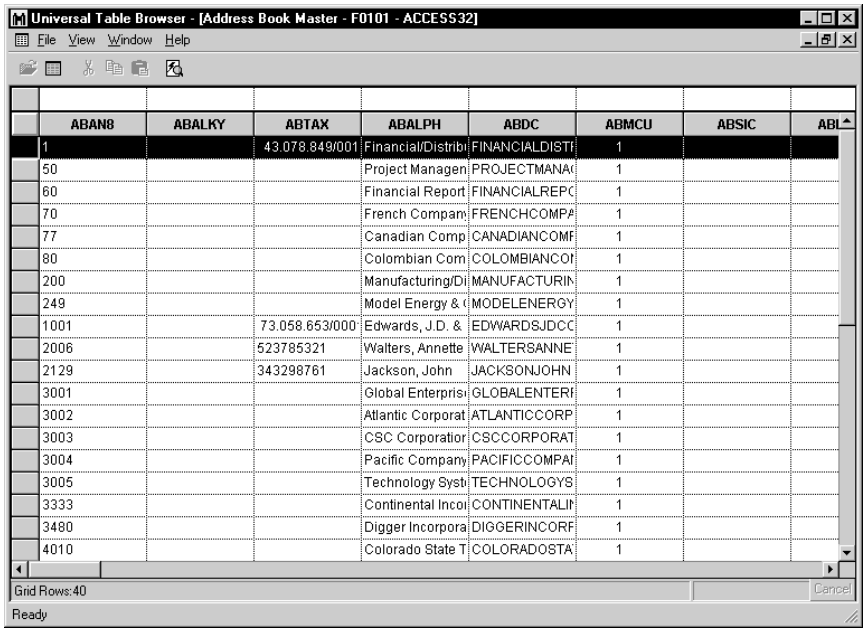
Example: Universal Table Browser (Unformatted Data)

In this example, a database table is shown as it was opened with the Format Data option turned off. Notice the structure of the information in the ABAN8 column of table F0101. This information is not formatted and is shown exactly as it is stored in the database.



Example: Universal Table Browser (formatted data)

In this example, a database table is shown as it was opened with the Format Data option turned on. Notice that the structure of the information in the ABAN8 column of table F0101 is formatted using the data dictionary specifications.



▶ To view column properties in a table

On Universal Table Browser

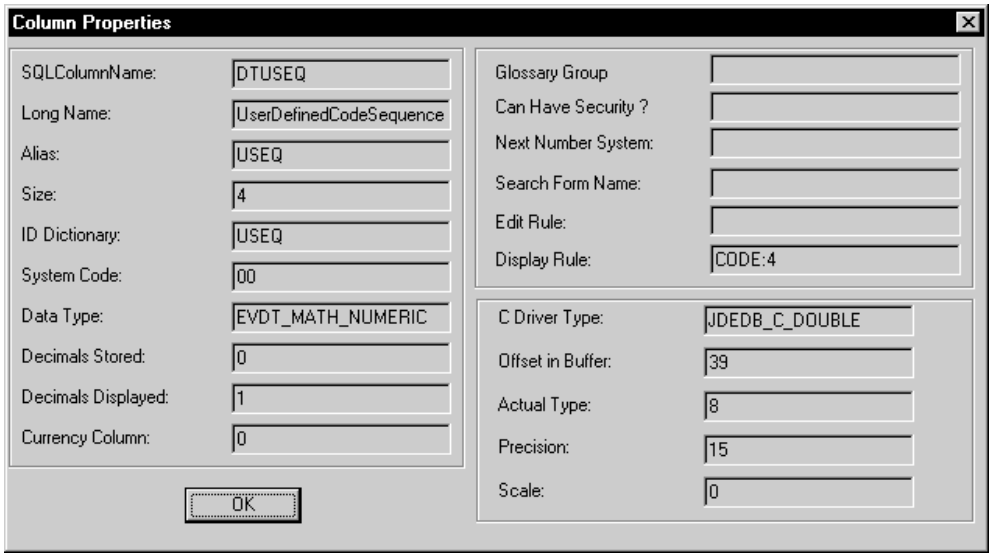
1. View a table as described in *Viewing the data in tables.*
2. Right-click on a desired column and choose Column Properties.

   If you are viewing a formatted table, the data dictionary properties are displayed in the upper-right portion of the Column Properties form.

   If you are viewing an unformatted table, the data dictionary properties are not displayed.

Example: Column Properties

In this example, the column properties are shown for the OneWorld data dictionary item USEQ. The SQL database name for this OneWorld item is DTUSEQ.



OneWorld Naming Conventions

The following list provides information about the naming conventions J.D. Edwards suggests that you use when you set up your configuration. You should use alphanumeric characters for your names. Depending on your server platform, some characters might not be allowed.

Path Codes

The naming conventions for a path code are as follows:

- Limited to 10 characters
- Letters must be uppercase only

Data Sources

The naming conventions for a data source are as follows:

- Limited to 30 characters
- Case sensitive
- Space sensitive

Specific exceptions for the Client Access data source are as follows:

Data Source Name:

- Limited to 32 characters
- Must begin with an alphabetic character
- You cannot use the following characters: { } [ ] ( ) ? * = ! @ ;

Note: You must type the data source name before you can use the Client Access ODBC driver to access AS/400 data.

Data Source Description:

- Limited to 80 characters

Package Names

The naming conventions for a package are as follows:

- Limited to 10 characters
- Uppercase only
- You cannot use the following characters: / \ : * ? " < > |

Server Names

The naming conventions for a server depend on the specific platform. For example, an HP9000 and an AS/400 allow you to enter different characters when you define the server name. OneWorld also limits the amount of characters you can use to name a server to 15 characters, regardless of the platform.

Workstation Names

The naming conventions for a workstation are as follows:

- Limited to 15 characters
- Each workstation requires a unique name
- When you add a workstation to a Windows NT Server domain, you must use the name created for the computer by the network administrator

  If the workstation name does not have a computer account in the domain, you cannot sign on to the domain or access any domain user accounts.

The jde.ini File

This section provides a listing of the settings within the jde.ini file (INI file for AS/400). This file provides the runtime settings required for OneWorld initialization. Specific versions of the file must reside on every OneWorld workstation and enterprise server in the installation.

This section describes the following tasks:

- ❏ Understanding workstation jde.ini settings
- ❏ Understanding AS/400 server INI settings
- ❏ Understanding UNIX server jde.ini settings (HP9000, RS/6000, or Sun Enterprise servers)
- ❏ Understanding Windows NT server jde.ini settings (Intel or Compaq AlphaServer)

How to use this section

To make finding descriptions faster, the sections, such as [CLUSTER], are alphabetized. The settings within the sections are in the order they appear in the jde.ini file.

Locating the jde.ini File

You can locate the jde.ini file (INI file for AS/400) in various places, depending on your OneWorld platform.

- For workstations, see *Working with the Workstation jde.log* in *Troubleshooting the Workstation* in the *Server and Workstation Administration Guide.*
- For enterprise servers, see *Working with the Enterprise Server jde.log* in *Troubleshooting the Enterprise Server* in the *Server and Workstation Administration Guide.*

Understanding Workstation jde.ini Settings

This section details the settings found in the client-side OneWorld workstation jde.ini file. Information is organized by section, for example [DEBUG]. Sections are alphabetized, but settings found within sections are listed in the order they are found in the software.

[DB SYSTEM SETTINGS]

Setting	Value	Purpose
Version=	43	A version number to prevent mismatch of jde.ini file with running version of OneWorld.
Default User=	JDE	The user account name for the database bootstrap tables.
Default Env=	A733CLA	The default environment on the workstation and/or the enterprise server.
Default PathCode=	PROD	The name of a subdirectory under \b7 that OneWorld uses to find specifications to dispay signon informatmation before an environment is selected.
Base Datasource=	Access 32	Data source representing database from which logon information is retrieved.
Object Owner=		Owner of system database tables.
Server=	*server name*	Server on which database resides
Database=	Access 32	The name of database where the system tables reside.
Load Library=	JDBODBC.DLL (non-Oracle), (default) JDBOCI73.DLL (Oracle only), JDBOCI80.DLL (Oracle only)	The JDE driver used to access the database holding the system tables. This depends on the database to be used and the type of system running OneWorld.
Decimal Shift=	N (default) or Y	A flag to indicate if decimal shifting is used for numeric data.
Julian Dates=	N (default) or Y	A flag to indicate if dates are stored in Julian or database-specific format.
Use Owner=	N (default) or Y	A flag to indicate that tables names are to qualified by owner.
Secured=	N (default) or Y	Indicates whether or not this is a secured database requiring a user and password login.

Setting	Value	Purpose
Type=	A (default), O, S, I	A single character denoting the type of database holding the system tables. These can be O (Oracle), A (MS Access), S (SQL Server), or I (Client Access, AS/400).
LibraryList=		AS/400 only. Database server holding the system tables.

[DB SYSTEM SETTINGS - SECONDARY]

This section is used for workstations only. The settings are used for a secondary data source to start OneWorld, should the primary data source be unavailable. These settings should be the same as the values in the Data Source Master table (F98611) for the secondary data source.

Setting	Typical Value	Purpose
Base Datasource=	Access32	The data source representing the database from which logon information is retrieved.
Object Owner=		The database owner of the system tables.
Server=	*server name*	The server on which the database holding the system tables resides.
Database=	Access32	The name of the database holding the system tables
Load Library=	JDBODBC.DLL (default)	The JDE driver used to access the database holding the system tables.
Decimal Shift =	N (default) or Y	A flag to indicate if decimal shifting is used for numeric data.
Julian Dates=	N (default) or Y	A flag to indicate if dates are stored in Julian or database-specific formats.
Use Owner=	N (default) or Y	A flag to indicate that tables names are to qualified by owner.
Secured=	N (default) or Y	A flag to indicate whether or not this is a secured database requiring user and password login.
Type=	A (default), O, S, or I	A single character denoting the type of database holding the system tables. These can be: O (Oracle), A (MS Access), S (SQL Server), or I (Client Access, AS/400).
Library List=		AS/400 only. Database server holding the system tables
Library=		AS/400 only. Database library holding the system tables.

[DEBUG]

Setting	Typical Value	Purpose
TAMMultiUserOn=	0	
Output=	None	Controls the status of the jdedebug file. Valid values are: NONE. No trace information is written to jdedebug.log. FILE. Database and runtime trace information is written to the file specified by the DebugFile= parameter in the [DEBUG] section. EXCFILE. Runtime trace information is written to the file specified by the DebugFile= parameter in the [DEBUG] section. BOTH. Trace information is written to both jde.log and jdedebug.log.
ServerLog=	0 (default) or 1	0 disables workstation requesting business function JDE.LOG and JDEDEBUG.OG entries from the server. 1 enabes workstation requesting business function JDE.LOG and JDEDEBUG.LOG entries from the server.
LEVEL=	BSFN,EVENTS	This parameter controls the debug level. You can specify any combination of allowable values using commas as delimiters. The default setting is LEVEL=BSFN,EVENTS. Valid values are: EVENTS Traces the start and stop of events. BSFN Traces when business functions are entered and when they return. SF_*x* Traces when system functions execute. The *x* variable is any allowable system function value. Valid values are listed below: SF_GRID SF_PARENT_CHILD SF_GENERAL SF_MESSAGING SF_WORKFLOW SF_WORKFLOW_ADMIN SF_MEDIA_OBJ SF_CONTROL For example, LEVEL=SF_CONTROL. In addition, you can specify multiple system functions by separating them with commas. For example, LEVEL=SF_GRID,SF_CONTROL.
DebugFile=	c:\jdedebug.log	Location and name of the jdedebug.log file
JobFile=	c:\jde.log	Location and name of jde.log file

[EVEREST]

Setting	Typical Value	Purpose
ShowAlias=	0 (default for PROD packages) 1 (default for APPL packages)	Enables (1) or disables (0) the ability to right-click on a Data Dictionary item and display its alias.

[INSTALL]

Setting	Typical Value	Purpose
DefaultSystem=	system	The name of the subdirectory under b7 that contains the OneWorld foundation code and tools.
ClientPath=	OneWorld Client Install	The name of the directory on the deployment server that contains the Workstation Installation program and other files used during deployment.
PackagePath=	package	The name of the subdirectory on the deployment server under a path code that contains the packages built for that path code.
DataPath=	data	The name of the subdirectory on the deployment server under the path code that contains the Access database delivered for all packages for that path code.
HOSTS=	hosts	The name of the directory on the deployment server that contains all types of hosts files. Used in host configuration generate app.
HP9000=	hp9000	The name of the directory on the deployment server that contains HP9000 files. Used in host configuration generate application.
RS6000=	rs6000	The name of the directory on the deployment server that contains RS/6000 files. Used in host configuration generate application.
AS400=	as400	The name of the directory on the deployment server that contains AS400 files. Used in host configuration generate application.
SUN=	sun	The name of the directory on the deployment server that contains AS400 files. Used in host configuration generate application.
LocalCodeSet=	WE_ISO88591	A setting used to determine alternate language usage. See “National Language Support” in the *OneWorld Upgrade Guide* (B73.3) for other values.

[INTERACTIVE RUNTIME]

J.D. Edwards strongly encourages you to not change any of the settings in this section.

Setting	Value	Purpose
DefaultMailServer=	C:\Program Files\Microsoft Exchange\ Exchng32.exe	Indicates the default mail server. When the user takes the e_mail exit from a OneWorld form, OneWorld launches the default mail program specified here. Examples of mail servers include Microsoft Exchange Server and Lotus Domino Mail Server.
DefaultBrowser=	Default.htm	This entry specifies the default web page to load when a web browser is invoked from a OneWorld application.
PWndBackground=	onemenu.bmp	
BITMAP_StatusBar=	stbr1.bmp	This key specifies the bitmap that shows up in the status bar of OneWorld forms.
ANIMATION_StatusBar=	oneworld.avi	This key is also obsolete and will be deleted in the near future
OBJECTQUEUE=		This key specifies a path to store OLE structured storage files for OLE objects that are created from OW grids
FONT_FaceName=	Arial	Specifies the Font name for the OW forms and controls excluding the Grid and Parent Child control
FONT_Height=	-12	Specifies the height of the above font
FONT_Weight=	400	Specifies the weight of the font
COLOR_GridBackground=	255,255,128	Specifies the background color of the grid in RGB (Red Green Blue) format
COLOR_GridForeground=	0,0,0	Specifies the foreground color of the grid in RGB (Red Green Blue) format
COLOR_GridDesktop=	64,128,128	Specifies the desktop color of the grid in RGB (Red Green Blue) format
COLOR_GridLines=	0,0,0	Specifies the grid lines color of the grid in RGB (Red Green Blue) format
CACHE_UseCache=	1	Key that instructs OW to use form caching
CACHE_MaximumNodes=	50	Indicates the maximum number of nodes to store in the cache
GRID_GridIntegralRows=	1	Indicates whether the grid should display integral number of grid lines. That is, no rows should be cut off
GRID_FONT_FaceName=	Arial	Same as FONT_ above: specifies font name for the Grid and Parent Child control.
GRID_FONT_Height=	-12	Same as FONT_ above: for Grid and Parent Child control text
GRID_FONT_Weight=	400	Same as FONT_ above: for Grid and Parent Child control text.

Setting	Value	Purpose
EXCEPTION_Enabled=	TRUE	Indicates whether structured exception handling is enabled
Initial_Language_Code=		
TextLimit=	80	Used to calculate the average character width when displaying text for controls in the Form Design Aid (FDA)

[JDE_CG]

Setting	Typical Value	Purpose
STDLIBDIR=	$(COMP)\VC98\lib	The path to the lib directory used by the MSVC compiler. This value is updated by a workstation installation based on the user's deployment preferences.
TPLNAME=	EXEFORM2	
ERRNAME=	CGERR	
TARGET=	Debug (default) Release	Used by the code generator and global build program to determine the type of build. Customer should only build under release, as there will be conflicts with the release build of the tools if they build under debug.
INCLUDES=	$(COMP)\VC98\include; $(SYSTEM)\include; $(SYSTEM)\cg; $(APP)\include; $(SYSTEM)\includev	The path to the include (header files) directory used by the MSVC compiler. This value is updated by a workstation installation based on the user's deployment preferences.
LIBS=	$(COMP)\VC98\lib; $(SYSTEM)\lib32; $(APP)\lib32; $(SYSTEM)\libv32	The path to the library directory used by the MSVC compiler and OneWorld Foundation. This value is updated by a workstation installation based on the user's deployment preferences.
MAKEDIR=	$(COMP)\VC98\bin; $(COMP)\Common\MSDev98\Bin	The path to the make directory used by the MSVC compiler. This value is updated by a workstation installation based on the user's deployment preferences.
USER=	*user name*	User ID who performed the workstation installation.

[JDEMAIL]

Setting	Typical Value	Purpose
mailProfile=	*Default Profile*	The name of the profile to be used for external mail systems to be accessed through OneWorld Work Center. Examples of external mail servers include Microsoft Exchange Server and Lotus Domino Mail Server.
mailServer=	owsmtp.jdedwards.com	The domain name of the SMTP server to be accessed for sending server mail messages.

[JDENET]

Setting	Typical Value	Purpose
netPgmName=	jdenet_n.exe	The full path of the JDENET_N executable.
krnlPgmName=	jdenet_k.exe	The full path of the JDENET_K executable.
serviceNameListen=	6005	
serviceNameConnect=	6005	
maxNetProcesses=	1	Defines the maximum number of JDENET_N processes that can be running. You can increase the value for a server that is expecting heavy JDENET message flow.
maxNetConnections=	50	Specifies the total number of connections that all JDENET_N processes can handle. For workstations, JDENET connections are mostly outgoing workstation to server connection. You can increase the value for a workstation that is expecting to connect to a larger number of servers at the same time.
maxKernelProcesses=	11	Specifies the maximum number of JDENET_K processes that can be running. The value should be greater than all the values added together in maxNumberOfProcesses of all the dedicated kernels on this server.
maxKernelRanges=	9	Specifies the number of dedicated servers and types. If this value is less than the value specified by maxKernelProcesses, then only the number of processes specified by this value are loaded into the workstation's memory and any other defined sections are not recognized by this workstation.
maxLenInlineData=	1024	For JDENET internal use only.
maxLenFixedData=	4096	For JDENET internal use only.
maxFixedDataPackets=	1024	For JDENET internal use only.
connectTimeout=	30	Number of seconds to wait before a connection request expired. Increase the value if a workstation is experiencing connection failure.
singleProcess=	1	Set to 1 on OneWorld workstation, set to 0 on OneWorld server.
netTrace=	0	
enableMsgPriority=	0	Set to 1 to enable priority messaging, set to 0 to disable priority messaging. This feature is not currently being used by OneWorld and should be set to 0, because it can cause and overall decrease in performance.

Setting	Typical Value	Purpose
enableGDMsgs=	0	Set to 1 to enable Guaranteed Delivery Messages, set to 0 to disable Guaranteed Delivery Messages. This feature is only used by Data Replication, so only enable it when using DR.
gdmRetryInterval=	0	To enable the sending of Guaranteed Delivery Messages, the following settings must be in the [JDENET] section. If the gdmRetryInterval is set to "0", the messages will only be retried at startup, otherwise they will be retried according to the number of seconds specified by the setting. For performance reasons, setting the retry interval to "0" is recommended usage on the workstation. Because workstations are not restarted as often, "0" is not a viable option for the servers, thus a high setting of around 1000 or so is recommended on a server. EnableGDMsgs=1 gdmRetryInterval=0
newProcessThreshold Connects=	0	Number of active connections JDENET_N process can handle before starting a new JDENET_N process. This setting is only used when the number of active JDENET_N processes is less than the value set in maxNetProcesses.
kernelDelay=	0	Number of seconds JDENET_K process will sleep when started. Use for debugging purpose and should be 0 for production.
netBroadcastAddress=	10.255.255.255	OneWorld uses the INI setting to broadcast datagram control messages A datagram broadcast to IP address 10.255.255.255 will reach all hosts in the network with a 10.XXX.XXX.XXX IP address. INADDR_BROADCAST is the default broadcast address if no INI setting is found. A datagram broadcast to address INADDR_BROADCAST will reach only hosts in the same physical network.

[JDENET_KERNEL_DEF1]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 1 kernel for internal and testing processes.

Setting	Value	Purpose
dispatchDLLName=	jdenet.dll	Identifies the name of the JDENET service program.
dispatchDLLFunction=	_JDENET_Dispatch Message@28	The name of the JDENET function for handling JDENET messages. The dispatchDLLName and dispatchDLLFunction entries are platform specific.
maxNumberOfProcesses=	2	Max number of dedicated kernel processes can be running for this dedicated server process.
beginningMsgTypeRange=	0	Beginning message type ID for this dedicated server process.
endingMsgTypeRange=	255	Ending message type ID for this dedicated server
newProcessThresholdRequests=	2	Number of outstanding requests each dedicated server process can have before starting a new dedicated server process. This setting is only used when active dedicated server process running for this dedicated server process is less than the value set in maxNumberOfProcesses.

[JDENET_KERNEL_DEF2]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 2 kernel to process OneWorld batch process (UBE) pass-through requests.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchUBEMessage@28	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	256	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	511	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF3]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 3 kernel to process data replication requests.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_DispatchRepMessage@28	Same as JDE_KERNEL_DEF1.

maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	512	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	550	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF4]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 4 kernel to process security server requests.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchSecurity@28	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	551	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	600	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF5]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 5 kernel to processes transaction manager and lock manager requests.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_TM_DispatchTransactionMessage@28	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	601	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	650	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF6]

This section defines JDENET internal dedicated server processes. It should not be changed. Oneworld uses the Type 6 kernel to process requests for remote Master Business Function (MBF). These requests are also referred to as CallObject requests using the jdeCallObject application programming interface (API).

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchCallObjectMessage@28	Same as JDE_KERNEL_DEF1.

maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	901	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	1156	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF7]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 7 kernel to process JDBNet server-to-server requests.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchJDBNETMessage@28	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	1201	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	1456	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF8]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 8 kernel to process package installation requests.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchPkgInstallMessage@28	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	1501	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	1756	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF9]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 9 kernel to process requests for the OneWorld Server Administration Workstation (SAW) application.

Setting	Typical Value	Purpose
dispatchDLLName=	jdesaw.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchSAWMessage@28	Same as JDE_KERNEL_DEF1.

maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	2001	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	2256	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF10]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 10 kernel to process requests for the OneWorld Scheduler application.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchScheduler@28	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	2501	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	2756	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF11]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 11 kernel to process package build requests.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchPkgBuildMessage@28	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	3001	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	3256	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.

[LOCK MANAGER]

Setting	Typical Value	Purpose
Server=	*server name*	This setting indicates the lock manager server to be used to process records. The value for this setting is the name of the server acting as the lock manager.
RequestedService=	NONE	This setting indicates the type of service that the workstation requests from the server. The service that is currently being provided by servers is time stamping (TS) only.

[NETWORK QUEUE SETTINGS]

Setting	Typical Value	Purpose
UBEQueue=	QBATCH	The batch name that the client submits for the UBE or package installation to the server.
UBEPriority=	5	The priority set when the UBE is submitted. For workstations, valid values are 1 to 5, where 1 is the highest priority setting. The priority setting is relative to other UBE jobs submitted by OneWorld.
PrintImmediate=	FALSE (default) or TRUE	OneWorld servers hold the UBE spool files submitted from a OneWorld workstation unless the jde.ini file on the workstation has the PrintImmediate=TRUE setting (this is case sensitive) in the [NETWORK QUEUE SETTINGS] section. On the AS/400, the spool file is created with the HOLD(*YES) attribute as a default. If the setting, PrintImmediate=TRUE is set in the jde.ini file on the workstation, upon submission of the UBE to the OneWorld server the spool file is released once the it is placed on the appropriate outqueue and closed.
SaveOutput=	TRUE (default) or FALSE	Lists whether the user wants to save the log files generated by the UBE.
InstallSpecs=	Y	Lists whether the user wants to install specifications when submitting UBEs.
JDENETTimeout=	60	The time out value for clients to attempt to connect to the server. Listed in seconds.

[OBJECT LIBRARIAN]

Setting	Value	Purpose
OLTLogMode=	YES (default) or NO or APPEND	This setting specifies if and how the Object Librarian Transaction log (OLT.log) is generated. It has three options, YES, NO and APPEND. If the value is YES, the OLT.log is generated for each transaction. If the log exists before a transaction, its contents are overwritten. If the value is NO, no OLT.log is generated during Object Librarian object transactions. If the value of the setting is APPEND, the information for a transaction is appended to the OLT.log. When the size of the log reaches its maximum size allowed (2 MB), the user is prompted to rename the existing file. If the user chooses not to rename it, the existing contents of the log will be overwritten by the information generated by the new transaction.
OLTLogContents=	GENERAL (default) or DETAIL	This setting specifies if detail information about specification records will be generated in the OLT.log. It has two options, GENERAL and DETAIL. If the value is GENERAL, no detail information about specification records will be generated. If the value is DETAIL, detail information will be generated.

[REPLICATION]

Setting	Typical Value	Purpose
DefaultEnvironment=	*environment name*	The Default Environment must contain a valid environment for the path code in which the publisher resides.
RepTrace=	0 (default) or 1	1 turns on replication tracing (logging), 0 indicates that replication tracing is off.
ForcedSync=	(entry does not appear initially)	Typically, OneWorld uses this setting for workstations after they initially load packages from a deployment server. The values for this setting are: • 0=off • 1=on OneWorld will force synchronization of replicated tables if the value for this setting is 1 or if this setting does not appear in the jde.ini. After OneWorld synchronizes the replicated tables on the publisher and the subscriber, OneWorld either changes the value from 1 to 0 or writes this setting into the jde.ini file with a value of 0.

[SECURITY]

Setting	Typical Value	Purpose
SecurityServer=	*server name*	
DataSource=	ORACLE PVC	
DefaultEnvironment=	*environment name*	Defines a valid environment in which the path code defines F98OWSEC.
UnifiedLogon=	0 (default), or 1	This setting specifies whether the unified logon feature is on or off. When off, Oneworld uses the standard logon functionality. Enter 0 (or leave blank) to set unified logon to off, or 1 to set it to on.
UnifiedLogonServer=	*server name*	This setting specifies where the unified logon server resides. If no server is present, OneWorld uses the same machine as the OneWorld security server.
ShowUnifiedLogon=	0 or 1 (default)	This setting determines whether the OneWorld environment selection form appears when the unified logon feature is used. Enter 0 if you do not want the environment selection form to be displayed, or 1 to display the form.

[SVR]

Setting	Typical Value	Purpose
EnvType=	1	Used by JDEKRNL
EnvironmentName=		
SpecPath=	spec	This line and all of the following in this section specify the path names so other OneWorld source programs know where to look for files. For instance, if “spec” were ever to be changed to “specifications”, changing SpecPath would allow this on the fly. This value is not updated by any program or process. The only reason to change this is aesthetic. This is the subdirectory under the path code user to store the replicated set of specification files on the workstation. This value is not updated by any program or process. There is really no good reason to ever change the name of this directory.
SourcePath=	source	On the client workstation, this is the subdirectory under the path code user to store the business function source files.
ObjectPath=	obj	On the client workstation, this is the subdirectory under the path code user to store the business function object files.
HeaderPath=	include	On the client workstation, this is the subdirectory under the path code and system directory, used to store the business function and system header files.
HeaderVPath=	includev	On the client workstation, this is the subdirectory under the system directory, used to store the foundation code header files.

Setting	Typical Value	Purpose
BinPath=	bin32	On the client workstation, this is the subdirectory under the path code and system directory, used to store the replicated set of business functions, application and foundation code dlls.
LibPath=	lib32	On the client workstation, this is the subdirectory under the path code and system directory, used to store the business function and system lib files.
LibVPath=	libv32	On the client workstation, this is the subdirectory under the path code and system directory, used to store the third party libraries.
MakePath=	make	On the client workstation, this is the subdirectory under the path code user to store the replicated set of business function make files. This value is not updated by any program or process. J.D. Edwards recommends that you not change the name of this directory.
WorkPath=	work	On the client workstation, this is the subdirectory under the path code user to store the replicated set of application temp files that are created during a build. This value is not updated by any program or process. J.D. Edwards recommends that you not change the name of this directory.
CodeGeneratorPath=	cg	On the client workstation, this is the subdirectory under the system directory, used to store the interactive application form type templates. These templates are used at runtime and are created during a build of applications.
ResourcePath=	res	On the client workstation, this is the subdirectory under the path code user to store the replicated set of bitmaps.
IconPath=	res\icons	On the client workstation, this is the subdirectory under the path code user to store the replicated set of icons.
HelpPath=	helps	This is the path to the location that stores the client-accessible set of replicated help files, if any. This path may point to a server. This path is specified in User Profiles.
TreeBmpPath=	res\treebmps	On the client workstation, this is the subdirectory under the path code user to store the tree bit map files.
ModelPath=	models	On the client workstation, this is the subdirectory under the path code user to store the models files.
LocalePath=	locale	This is the base directory for the National Language Support (NLS) conversion tables.
Iconvpath=	locale\Iconv	This is the directory for the National Language Support (NLS) conversion map.

[TAPI]

Setting	Value	Purpose
ProgID=OWTAPI.*driver*.1	The name of the driver for the third-party telephony product (instead of *driver*)	This setting identifies to OneWorld the logical name of the third-party telephony product's driver. You can obtain this information from the driver vendor or the Worldwide Customer Support administrator. A standard OneWorld distribution also includes a test driver called TestDriver. When entering this driver name the setting would look like this: ProgID=OWTAPI.TestDriver.1

[TAPI – *driver*]

Setting	Value	Purpose
Host= Port=	The host and port number for the telephony driver. For example: Host=ctiserver1.jdedwards.com Port=8001	This section is for third-party telephony drivers that have additional configuration settings. For example, you might need to specify the host and port number for the third-party telephony product. You can obtain the entries for this section from the driver vendor or the Worldwide Customer Support administrator.

[UBE]

Setting	Value	Purpose
UBEDebugLevel=	0 (default), 1, 2, 3, 4, 5, or 6	Used to specify what level of debugging information will be provided when using UBE debug logging. 0 is off, and 6 is the highest level of logging information. 0 = Error messages only 1 = Informative messages 2 = Section level messages 3 = Object level messages 4 = Event rules messages 5 = SQL statements 6 = UBE function messages
UBESaveLogFile=	0 (default), or 1	This determines whether or not UBE debug log files will be saved. 0 means they will not be saved, and 1 means the files will be saved.

Setting	Value	Purpose
UBEShowPDFLink=	0 (default) or 1	When running a report with Balance Auditor defined, if this value is 1, then a rectangle will be displayed around the link in the PDF file. NO rectangle will be displayed if this value is 0.
UBEFont_PointSize=	80	This is the font size to be used if running with a non-default language preference.

Understanding AS/400 Server INI Settings

This section details the settings found in the OneWorld AS/400 server INI. Information is organized by section, for example [DEBUG]. Sections are alphabetized, but settings found within sections are generally listed in the order they are found in the software.

[AS400]

Setting	Value	Purpose
CRTMOD=	CRTMOD MODULE(%s/%s) SRCFILE(%s/%s) SRCMBR(%s) OUTPUT(*PRINT) DBGVIEW(*NONE) OPTIMIZE(40)	The string used by the package install to compile business functions. Note that CRTMOD and CRTMOD2 are concatenated, and used by OneWorld to compile business functions.
CRTMOD2=	DEFINE(JDENV_AS400 MUTEX PRODUCTION_ VERSION NO_SIGNALS) TGTRLS(V4R2M0)	The concatenated string used by the package install for declaring additional defines for compiling business functions.
CRTSRVPGM=	CRTSRVPGM SRVPGM(%s/%s) MODULE(%s/*ALL) BNDSRVPGM(JDELIB JDEKRNL) EXPORT(*AL) OPTION(*DUPPROC *DUPVAR *UNRSLVREF) TGTRLS(V4R2M0)	The string used by the package install for binding business function modules to create the OneWorld service programs (*SRVPGM).
CRTDBPGM1=	CRTPGM PGM(%s/%s) MODULE(DBDRVAG DBDRV_AC DBDRV_CC DBDRV_CN	The concatenation of CRTDBPGM* settings are used to create the database programs JDB_*. These database programs are automatically created by OneWorld at start-up. The SENTINEL job creates them at start-up time, and monitors and creates additional programs as needed during runtime. The status of the programs and their usage is maintained in the user space JDEPGMCTL in the CONTROL library.
CRTDBPGM2=	DBDRV_CH DBDRV_CP DBDRV_RQ DBDRVSQL DBMONCTL DBDRVDLI	See Purpose for CRTDBPGM1.
CRTDBPGM3=	DBSQL DBSQL_A DBSQL_D DBSQL_I DBSQL_M DBSQL_S DBSQL_U DBSQL_X	See Purpose for CRTDBPGM1.

Setting	Value	Purpose
CRTDBPGM4=	BNDSRVPGM(JDEKRNL JDELIB JDEIPC) ACTGRP(%s) OPTION(*DUPPROC	See Purpose for CRTDBPGM1.
CRTDBPGM5=	*DUPVAR) AUT(*ALL) TGTRLS(V4R2M0)	See Purpose for CRTDBPGM1.
PrintUBEJoblog=	FALSE (default) or TRUE	If true, indicates that OneWorld always writes the AS/400 JOGLOG for the batch application (UBE) to a spool file.
PrintUBEJoblogOn Error=	FALSE (default) or TRUE	If true, indicates that OneWorld writes the AS/400 JOBLOG for the batch application (UBE) to a spool file if an error occurs, for example, if a UBE fails.

[BSFN BUILD]

Setting	Value	Purpose
Build Area=	/jdeb733_0/packages	The location on the server where the package will be built.
Optimization Flags=	(40)	Machine dependent. These compile flags are used when building business functions in "Release" mode. You should not change these flags.
DebugFlags=	*ALL	Machine dependent. These compile flags are used when building business functions in "Debug" mode. You should not change these flags.
InliningFlags=	Y (default, yes) N (no)	Yes turns on inlining on the AS400. No turns it off. This entry is blank for non-AS/400 servers.
DefineFlags=	JDENV_AS$))MUTEX PRODUCTION_VERSI ON JDBDB2400 AS400V3R6	
CompilerFlags=	*EXPMAC *NOSHOWINC -qspill=1024	This setting determines whether to compile listings when doing a server package build. If it is set to *PRINT, listings are compiled. If set to *NONE, listings are not compiled.
CompileOutput=	*NONE or *PRINT	Machine dependent. Valid compiler flags. The spill flag sets the stack space when business functions are compiled. J.D. Edwards has found that 1024 is adequate to compile the delivered business functions. While values other than the default of 1024 may be valid on various host platforms, this is the only value validated by J.D. Edwards.
OSReleaseLevel=	V4R2M0	Release level you are compiling to. You should not change these flags.

Setting	Value	Purpose
LinkFlags=	*DUPROC *DUPVAR * UNRSLVREF	Machine dependent. These flags are used when linking business functions. You should not change these flags.
LinkLibraries=	JDELIB, JDEKRNL, JDENET, JDEIPC	Libraries to which business functions are linked. (Windows NT and AS/400 servers only.)
SimultaneousBuilds=	0 (default) any integer	Indicates the number of DLLs that can be built at a time. O means that all will be built simultaneously.

[DB SYSTEM SETTINGS]

Setting	Value	Purpose
Version=	43	A version number to prevent mismatch of jde.ini file with running version of OneWorld.
Default User=	JDETL	The user account name for the database bootstrap tables.
Default Pwd=	JDETL	The user account password for the database bootstrap tables.
Default Env=	B733APP	The default data source on the workstation and/or the enterprise server.
Default PathCode=	B733APP	The subdirectory under \$PKG under which the business function code is stored.
Base Datasource=	DB2	Data source representing database from which logon information is retrieved.
Object Owner=		Owner of system database tables.
Server=	*server name*	Server on which database resides
Database=	*database name*	The name of database where the system tables reside.
Load Library=	DBDR (default)	The JDE driver used to access the database holding the system tables. This depends on the database to be used and the type of system running OneWorld.
Decimal Shift=	Y (default) or N	A flag to indicate if decimal shifting is used for numeric data.
Julian Dates=	Y (default) or N	A flag to indicate if dates are stored in Julian or database-specific format.
Use Owner=	Y or N (default)	A flag to indicate that tables names are to qualified by owner.
Secured=	Y (default) or N	Indicates whether or not this is a secured database requiring a user and password login.
Type=	I	A single character denoting the type of database holding the system tables. These can be O (Oracle), A (MS Access), S (SQL Server), or I (Client Access, AS/400).

Setting	Value	Purpose
Library=	*database library*	AS/400 only. Database library holding the system tables.
DatabaseProgamMax=	-1 (default)	AS/400 only. Maximum number of database connection programs to allow. The value -1 means no limit.
DatabaseProgramInitial=	10 (default)	AS/400 only. The number of database connection programs to start initially at OneWorld start.
DatabaseProgramThreshold=	3 (default)	AS/400 only. The threshold for starting new database connection programs. If the number of database connection programs not is use drops below this limit, start new ones.
DatabaseProgramAdditional=	10 (default)	AS/400 only. How many new database connection programs to start when the threshold number is reached.
DatabaseProgramCheckIntervalSeconds=	10 (default)	AS/400 only. How long, in seconds, until OneWorld will be put to sleep after the database connection programs are created.
Default Journal=	OW_JRNL	AS/400 only. The name of the default journal. Journaling is required on AS/400 for rollback recovery. There are two components to journaling: the journal and the journal receiver. Both before and after images of a database transaction can be recorded by journaling. This can be set to any character string 10 characters or fewer in length.
Default Journal LIBRARY=	B733SYS	AS/400 only. The library name where the journal is stored. This can be set to any valid library name.
Default Journal Receiver	OW_JRNL000	AS/400 only. The name of the journal receiver. This can be set to any character string 10 characters or fewer in length.
Default Journal Receiver LIBRARY=	B733SYS	AS/400 only. The library name where the journal receiver is stored. This can be set to any valid library name.

[DEBUG]

Setting	Typical Value	Purpose
Output=	FILE	Controls the status of the jdedebug log file. Valid values are: NONE. No trace information is written to jdedebug.log. FILE. Database and runtime trace information is written to the file specified by the DebugFile= parameter in the [DEBUG] section. EXCFILE. Runtime trace information is written to the file specified by the DebugFile= parameter in the [DEBUG] section. BOTH. Trace information is written to both jde.log and jdedebug.log.
Trace=	TRUE	Writes additional trace information to the log files to aid in debugging.
DebugFile=	JDEB733/ jdedebug	Location of the jdedebug log. J. D. Edwards ships this value set to jdedebug. There are no processes that update this value. The names of the resulting files will be *path*/jdedebug_######.log, here ###### represents the AS/400 job number associated with the job that created the file. **Note:** OneWorld does not create the path to these files. The path must exist prior to the logging process. The path resides in the Integrated File System (IFS) on the AS/400. You may use the AS/400 WRKLNK command to see a list of directories and files and navigate between the IFS directories. J.D. Edwards ships a command called DSPSTMF that allows you to view these log files. In addition, you can set up Client Access to more easily view some of the smaller log files See *Using AS/400 Integrated File System (IFS) Logging Support* for details about how to set up Client Access to view log files.
JobFile=	JDEB733/jde.log	Location of the jde log. J. D. Edwards ships this value set to the jde.log. There are no processes that update this value. Examine the log files jde.log, and jdedebug for information useful to assist in problem analysis and resolution. The names of the resulting files will be *path*/jde_######.log where ###### is the AS/400 job number associated with the job that created the file. The default value is the JDE.LOG member located in the system library. The process ID is specified as a member name in the field JDE.LOG with a # sign as the first character. For example, the member name might be #2142. See *Using AS/400 Integrated File System (IFS) Logging Support* for details about how to set up Client Access to view log files.
JDETSFile=	/JDEB733/JDETS. LOG	This specifies the location of the lock manager trace file on the AS/400.
ClientLog=	1 (default) or 0	1 enables servicing CALLOBJ server trace to workstation. 0 disables servicing CALLOBJ server trace to workstation.

Setting	Typical Value	Purpose
LogErrors=	1 (default) or 0	Action for error messages. 0 or FALSE indicates that no error messages will be written to JDE.LOG. 1 or TRUE indicates that error messages will be written to JDE.LOG.
KeepLogs=	1	1 indicates that the logs will be saved after printing. 0 indicates that the logs will not be saved.
RunBatchDelay=	0	Specifies the time that runbatch waits upon startup, in seconds. This setting allows developers to start debug on the job or process.
TAMTraceLevel=	0 (default)	Specifies the level of TAM tracing where 0 is off and 9 provides the greatest amount of tracing detail.
RepTrace=	0 (default) or 1	You can enable replication trace if you want to perform troubleshooting on your replication process. When you enable this trace, the replication process sends additional information to JDE.LOG. Do not leave replication trace on permanently, as JDE.LOG will become too large. The valid values are: 0 = OFF 1 = ON

[INSTALL]

Setting	Typical Value	Purpose
DefaultSystem=	B733SYS	The name of the OneWorld System library.
ClientPath=	B733APP	The name of a valid path code on the deployment server that contains the workstation installation program and other files used during deployment.
B733=		Should be left blank on AS/400
LocalCodeSet=	US_EBCDIC	A setting used to determine alternate language usage. See "National Language Support" in the *OneWorld Upgrade Guide* (B73.3) for other values.
WebAdmin=	1	This setting specifies whether the system generates all the Java objects for the default user. This includes overriding Java objects previously generated. If you leave this value blank, the system generates all the Java objects for the current user.
EnvCreation=	1 (default) to 5	This setting determines the number of environments that can be processed (loaded) at the same time.

[JDEIPC]

Setting	Typical Value	Purpose
maxNumberOfResources=	1000	Total number of IPC resources that are available to JDE.
startIPCKeyValue	2101	On NT this value is used just to uniquely name the IPC Shared memory. On all other systems, this is the value of the IPC Id which JDEIPC used for its shared memory. This plus the maxNumberOfResources define the range of IPC ids that JDE will used on the system. SysAdmins should ensure that this range of Ids is not used by any other software. Although JDEIPC will not used an existing Id in its range, this may not be true of other software.
avgResourceNameLength	15	JDE Internal. Increase this value if you get an IPC error 'String table full."
maxMsgqEntries=	1024	
mazMsgqBytes=	65536	
ipcTrace=	0	

[JDEMAIL]

Setting	Typical Value	Purpose
mailServer=	owsmtp.jdedwards.com	The domain name of the SMTP server to be accessed for sending server mail messages.

[JDENET]

Setting	Typical Value	Purpose
netPgmName=	/QSYS.LIB/B733SYS.LIB/JDENT_N.PGM	The full path of the JDENET_N executable.
krnlPgmName=	/QSYS.LIB/B733SYS.LIB/JDENET_K.PGM	The full path of the JDENET_K executable.
serviceNameListen=	jde_server	
serviceNameConnect=	jde_server	
maxNetProcesses=	1	Defines the maximum number of JDENET_N processes that can be running. You can increase the value for a server that is expecting heavy JDENET message flow.
maxNetConnections=	800	Total number of connections all JDENET_N processes can handle. This value is platform-specific. You can increase the value for a server that is expecting to handle larger number of workstations at the same time.

Setting	Typical Value	Purpose
newProcessThreshold Connects=	100	Number of active connections JDENET_N process can handle before starting a new JDENET_N process. This setting is only used when the number of active JDENET_N processes is less than the value set in maxNetProcesses.
netShutdownInterval=	15	
connectTimeout=	10	Number of seconds to wait before a connection request expired. Increase the value if the workstation is experiencing connection failure.
maxKernelProcesses=	12	Maximum number of JDENET_K processes that can be running. The value should be greater than all the values add together in maxNumberOfProcesses for all of the dedicated servers as defined in the [JDENET_KERNEL_DEFx sections.
maxKernelRanges=	12	Number of dedicated servers and types.
kernelDelay=	0	Number of seconds JDENET_K process will sleep when started. Use for debugging purpose and should be 0 for production.
maxLenInlineData=	1024	For JDENET internal use only.
maxLenFixedData=	4096	For JDENET internal use only.
maxFixedDataPackets=	1024	For JDENET internal use only.
singleProcess=	0	Set to 1 on OneWorld workstation. Set to 0 on OneWorld server.
netTrace=	0	
enableMsgPriority=	0	Set to 1 to enable priority messaging, set to 0 to disable priority messaging. This feature is not currently being used by OneWorld and should be set to 0, because it can cause and overall decrease in performance.
enableGDMsgs=	0	Set to 1 to enable Guaranteed Delivery Messages, set to 0 to disable Guaranteed Delivery Messages. This feature is only used by Data Replication, so only enable it when using DR.

Setting	Typical Value	Purpose
gdmRetryInterval=	60	To enable the sending of Guaranteed Delivery Messages, the following settings must be in the [JDENET] section. If the gdmRetryInterval is set to "0", the messages will only be retried at startup, otherwise they will be retried according to the number of seconds specified by the setting. For performance reasons, setting the retry interval to "0" is recommended usage on the workstation. Because workstations are not restarted as often, "0" is not a viable option for the servers, thus a high setting of around 1000 or so is recommended on a server. EnableGDMsgs=1 gdmRetryInterval=0
krnlCoreDump=	0	Setting this value to 1 causes the kernel job to write a JOBLOG and perform a DSPJOB if it crashes. This is very useful for debugging purposes.

[JDENET_KERNEL_DEF1]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 1 kernel for internal and testing processes.

Setting	Value	Purpose
dispatchDLLName=	JDENET	Identifies the name of the JDENET service program.
dispatchDLLFunction=	JDENET_Dispatch Message	The name of the JDENET function for handling JDENET messages. The dispatchDLLName and dispatchDLLFunction entries are platform specific.
maxNumberOfProcesses=	1	Max number of dedicated kernel processes can be running for this dedicated server.
beginningMsgTypeRange=	0	Beginning message type ID for this dedicated server.
endingMsgTypeRange=	255	Ending message type ID for this dedicated server
newProcessThresholdRequests=	1	Number of outstanding requests each dedicated kernel process can have before starting a new dedicated kernel process. This setting is only used when active dedicated kernel process running for this dedicated server is less than the value set in maxNumberOfProcesses.
numberOfAutoStartProcesses=	0	Indicates the number of processes to automatically start for this kernel type when the main JDENET process start up on a server. If this number is 0, then no kernel processes will start. THis number must be less than the maximum of processes (in setting maxNumberofProcesses) allowed for this kernel type. If the number is greater than the maximum allowed, then only the maximum number of processes will start.

[JDENET_KERNEL_DEF2]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 2 kernel to process OneWorld batch process (UBE) pass-through requests.

Setting	Typical Value	Purpose
dispatchDLLName=	JDEKRNL	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchUBEMessage	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	256	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	511	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	1	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF3]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 3 kernel to process data replication requests.

Setting	Typical Value	Purpose
dispatchDLLName=	JDEKRNL	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	DispatchRepMessage	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	512	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	550	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	1	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF4]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 4 kernel to process security server requests.

Setting	Typical Value	Purpose
dispatchDLLName=	JDEKRNL	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchSecurity	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	551	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	580	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	1	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF5]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 5 kernel to processes transaction manager and lock manager requests.

Setting	Typical Value	Purpose
dispatchDLLName=	JDEKRNL	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	TM_DispatchTransactionManager	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	601	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	650	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF6]

This section defines JDENET internal dedicated server processes. It should not be changed. Oneworld uses the Type 6 kernel to process requests for remote Master Business Function (MBF). These requests are also referred to as CallObject requests using the jdeCallObject application programming interface (API).

Setting	Typical Value	Purpose
dispatchDLLName=	JDEKRNL	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchCallObjectMessage	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	10	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	901	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	1156	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF7]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 7 kernel to process JDBNet server-to-server requests.

Setting	Typical Value	Purpose
dispatchDLLName=	JDEKRNL	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchJDBNETMessage	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	1201	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	1756	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	1	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF8]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 8 kernel to process package installation requests.

Setting	Typical Value	Purpose
dispatchDLLName=	JDEKRNL	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchPkgInstallMessage	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	1501	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	1756	Same as JDE_KERNEL_DEF1.

newProcessThresholdRequests=	1	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF9]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 9 kernel to process requests for the OneWorld Server Administration Workstation (SAW) application.

Setting	Typical Value	Purpose
dispatchDLLName=	JDESAW	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchSAWMessage	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	2001	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	2256	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	1	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF10]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 10 kernel to process requests for the OneWorld Scheduler application.

Setting	Typical Value	Purpose
dispatchDLLName=	JDEKRNL	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchScheduler	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	2501	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	2756	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	1	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF11]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 11 kernel to process package build requests.

Setting	Typical Value	Purpose
dispatchDLLName=	JDEKRNL	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchPkgBuildMessage	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	3001	Same as JDE_KERNEL_DEF1.

endingMsgTypeRange=	3256	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF12]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 12 kernel to support a single, dedicated kernel job for UBE subsystem requests.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekern.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchUBESBSMessage	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1. Do not set higher than 1.
beginningMsgTypeRange=	3501	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	3756	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.

[LOCK MANAGER]

Setting	Typical Value	Purpose
AvailableService=	NONE	This setting indicates the service that the lock manager server is offering. It is also used to indicate whether the lock manager server is on or off. Valid values are TS and NONE.
RequestedService=	NONE	This setting indicates the type of service that the workstation requests from the server. The service that is currently being provided by servers is time stamping (TS) only. Valid values are TS and NONE.

[NETWORK QUEUE SETTINGS]

Setting	Typical Value	Purpose
DefaultPrinterOUTQ=	QGPL/ONEWORLD_A	The default printer to which batch applications will be routed.

[SECURITY]

Setting	Typical Value	Purpose
DataSource=		
User=	JDESVR	
Password=	JDESVR	

DefaultEnvironment=	P733ASD1	Defines a valid environment in which the path code defines F98OWSEC.
SecurityServer=	*security server name*	
ServerPswdFile=	TRUE	The setting of this parameter determines whether OneWorld uses special password handling for batch reports running on the server. Set the value to TRUE to instruct OneWorld to enable special handling of passwords. Set the value to FALSE to disable special handling. When OneWorld runs a batch report on the server, it runs the report using a string of line commands and parameters that includes the "user password". Under some operating systems, it is possible to query the status of a job and view the parameters that were used to start the process. As a security measure, you can enable special handling by OneWorld. When enabled, OneWorld does not include the "user password" in the parameter list for a batch process. Instead, it includes the name of a file that contains the "user password". OneWorld instructs the operating system to destroy this file as soon as the batch report reads the password.
History=	0	
SecurityMode=	0 (default), 1, or 2	This setting controls whether OneWorld uses the standard logon procedure, unified logon, or both. Enter 0 to accept only the standard logon, 1 to accept only the unified logon, or 2 to accept both.
AllowedUsers=	*group or user names*	This setting for the unified logon feature enables you to specify users or groups who are allowed to use OneWorld. If no users or groups are specified, all users who have logged on to the proper domains are authenticated by the unified logon server.

[SVR]

Setting	Typical Value	Purpose
SpecPath=		This line and all of the following in this section specify the path names so other OneWorld source programs know where to look for files. This value is not updated by any program or process. The only reason to change this is aesthetic. This is the subdirectory under the path code user to store the replicated set of specification files on the workstation.
PackedSpecPath=	/b733APP	

[TCENGINE]

Setting	Typical Value	Purpose
TraceLevel=	0 (default)	The level of table conversion logging to perform. Valid values are 0-9**,** where 9 generates the most logging, and 0 generates no logging.
StopAfterRow=	0 (default)	The number of rows to process during table conversion. This setting is useful for debugging. the value 0 indicates the table conversion processes all rows. Provide a positive number to stop after processing that number of rows.
ForceRowByRow=	0 (default) or 1	0 allows inserts from selects. 1 forces table conversions to convert one row at a time, regardless of whether an insert could be used.

[UBE]

Setting	Typical Value	Purpose
UBEDebugLevel=	0 (default), 1, 2, 3, 4, 5, or 6	Used to specify what level of debugging information will be provided when using UBE debug logging. 0 is off, and 6 is the highest level of logging information. 0 = Error messages only 1 = Informative messages 2 = Section level messages 3 = Object level messages 4 = Event rules messages 5 = SQL statements 6 = UBE function messages
UBESubsystemLimit	3	Used to specify the number of subsystem jobs per report version.
UBEPrintDataItems	0 (default), or 1	Used to specify whether to print the associated data item description in the pdf file as meta data for third party vendors. 0 is no, and 1 is yes.

Setting	Typical Value	Purpose
Default Printer Queue Value= (for example, QPGL/ONEWORLD_A)	POSTSCRIPT-PRINTER,PAGESIZE (250,250),DRAWER (LETTER),ORIENT(L)	Used to specify the default printer used for printing batch application reports.
WebServer	1	This setting specifies whether the system enables the UBE feature from the web server and identifies the OneWorld kernel as a web kernel to meet the special needs of the web. If you leave this value blank, the calls from the business functions or the error message handling from the kernel will not work properly.

[WORLD ENVIRONMENT MAP]

Setting	Value	Purpose
OneWorldEnvironmentName= (for example, APPLJDEDC2)	*WorldEnvironment Name* (for example, QA73COMP)	The string used by special business function code to set up WorldSoftware library lists from within OneWorld to assist in calling WorldSoftware from OneWorld. As of B73.3, the functions associated with these settings might not be utilized by application developers.

Understanding UNIX Server jde.ini Settings (HP9000, RS/6000, or Sun servers)

This section details the settings found in the OneWorld UNIX server jde.ini file. Some settings may differ between server platforms. Information is organized by section, such as [DEBUG]. Sections are alphabetized, but settings found within sections are generally listed in the order they are found in the software.

[BSFN BUILD]

Setting	Value	Purpose
Build Area=	/usr/oneworld/b733/packages	The location on the server where the package will be built.
Optimization Flags=	+02 (default for HP9000) -02 (default for RS/6000 and Sun)	Machine dependent. These compile flags are used when building business functions in "Release" mode. You should not change these flags.
DebugFlags=	-g -y -D_DEBUG _DJDEDEBUG (default for HP9000) -g -qfulpath -qdbextra -D_DEBUG -DJDEDEBUG (default for RS/6000) -g -D_DEBUG -DJDEDEBUG (default for Sun)	Machine dependent. These compile flags are used when building business functions in "Debug" mode. You should not change these flags.
InliningFlags=	blank (default)	Yes turns on inlining on the AS400. No turns it off. This entry is blank for non-AS/400 servers.
DefineFlags=	-DKERNEL -DPRODUCTION_VERSION -DNATURAL_ALIGNMENT -D_HPUX-SOURCE (default for HP9000) -DKERNEL -DPRODUCTION_VERSION -DNATURAL_ALIGNMENT (default for RS/6000) -DKERNEL -DPRODUCTION_VERSION -DNATURAL_ALIGNMENT -D_SUN-SOURCE (default for Sun)	

Setting	Value	Purpose
CompilerFlags=	-Aa +w1 +z -c (default for HP9000) -qalign=natural -qflag=I:I -c (default for RS/6000) -qspill=1024 -misalign –KPIC (default for Sun)	Machine dependent. Valid compiler flags. The spill flag sets the stack space when business functions are compiled. J.D. Edwards has found that 1024 is adequate to compile the delivered business functions.
OSReleaseLevel=	+DAportable –q32 (for AIX)	Release level you are compiling to. You should not change these flags.
LinkFlags=	-b -z (default for HP9000) -bl:/<your system directory>/bin32/functlist.imp -bM:SRE -bexpall -brtl -lc -bnentry -L. L/usr/<your system directory>/lib -ljdelib -ljdekrnl -ljdenet -bloadmap:loadmap (default for RS/6000) -G –L$(ORACLE_HOME)/lib (default for Sun)	Machine dependent. These flags are used when linking business functions. You should not change these flags.
LinkLibraries=	blank (default)	Libraries to which business functions are linked. (Windows NT and AS/400 servers only.)
SimultaneousBuilds=	0 (unlimited) (default) any integer (number of simultaneous builds)	Indicates the number of DLLs that can be built at a time. O means that all will be built simultaneously.

[CLUSTER]

Setting	Typical Value	Purpose
Primary Node=	*server name*	When clustering is used with OneWorld, this setting specifies a primary server where OneWorld will run, or a floating IP address name. This setting is delivered commented out.

[DB SYSTEM SETTINGS]

Setting	Value	Purpose
Version=	43	A version number to prevent mismatch of jde.ini file with running version of OneWorld.
Default User=	JDESVR	The user account name for the database bootstrap tables.
Default Pwd=		The user account password for the database bootstrap tables.

Setting	Value	Purpose
Default Env=	P733HPO1 (default for HP9000) P733RSO1 (default for RS/6000)	The default data source on the workstation and/or the enterprise server.
Default PathCode=	PROD	The subdirectory under \$PKG under which the business function code is stored.
Base Datasource=	ORACLE SVR	Data source representing database from which logon information is retrieved.
Object Owner=	JDESVR	Owner of system database tables.
Server=	*server name*	Server on which database resides
Database=	hp9000adevl	The name of database where the system tables reside.
Load Library=	libora73.sl (default for HP9000) libora80.so (default for RS/6000)	The JDE driver used to access the database holding the system tables. This depends on the database to be used and the type of system running OneWorld.
Decimal Shift=	Y (default) or N	A flag to indicate if decimal shifting is used for numeric data.
Julian Dates=	Y (default) or N	A flag to indicate if dates are stored in Julian or database-specific format.
Use Owner=	Y (default) or N	A flag to indicate that tables names are to qualified by owner.
Secured=	Y (default) or N	Indicates whether or not this is a secured database requiring a user and password login.
Type=	O (default), A, S, I	A single character denoting the type of database holding the system tables. These can be O (Oracle), A (MS Access), S (SQL Server), or I (Client Access, AS/400).
LibraryList=	blank (default)	AS/400 only. Database server holding the system tables.
TriggerLibrary=	JDBTRIG (default)	AS/400 only. Database library holding the system tables.

[DEBUG]

Setting	Typical Value	Purpose
Output=	FILE	Controls the status of the jdedebug file. Valid values are: NONE. No trace information is written to jdedebug.log. FILE. Database and runtime trace information is written to the file specified by the DebugFile= parameter in the [DEBUG] section. EXCFILE. Runtime trace information is written to the file specified by the DebugFile= parameter in the [DEBUG] section. BOTH. Trace information is written to both jde.log and jdedebug.log.
Trace=	TRUE	Writes additional trace information to the log files to aid in debugging.
ClientLog=	1 (default) or 0	1 enables servicing of business functions JDE.LOG and JDEDEBUG.LOG entries from the server to the workstation. 0 disables this service.
DebugFile=	jdedebug.log	Location and name of the jdedebug.log file
JobFile=	jde.log	Location and name of jde.log file
LogErrors=	1	
JDETSFile=	JDETS.log	Specifies the location of the lock manager trace file.
RepTrace=	1	Enables replication of log messages.
KeepLogs=	1	Keeps UBE logs in the PrintQueue directory for all UBEs.

[INSTALL]

Setting	Typical Value	Purpose
DefaultSystem=	system	The name of the subdirectory under b7 that contains the OneWorld foundation code and tools.
ClientPath=	client	The name of the directory on the deployment server that contains the Workstation install program and other files used during deployment.
PackagePath=	package	The name of the subdirectory on the deployment server under a path code that contains the packages built for that path code.
DataPath=	data	The name of the subdirectory on the deployment server under the path code that contains the Access database delivered for all packages for that path code.
B733=	/usr/oneworld/b733	Base path of the OneWorld installation.

Setting	Typical Value	Purpose
LocalCodeSet=	WE_ISO88591	A setting used to determine alternate language usage. See "National Language Support" in the *OneWorld Upgrade Guide* (B73.3) for other values.
WebAdmin=	1	This setting specifies whether the system generates all the Java objects for the default user. This includes overriding Java objects previously generated. If you leave this value blank, the system generates all the Java objects for the current user.
EnvCreation=	1 (default) to 5	This setting determines the number of environments that can be processed (loaded) at the same time.

[JDEIPC]

Setting	Typical Value	Purpose
ipcTrace=	0	Set to 1 to enable IPC logging messages. Caution: this setting can cause the log files to grow very fast.
maxNumberOfSemaphores =	200 (default for HP9000 and Sun)	Not delivered for the RS/6000.
startIPCKeyValue	7001	Delivered commented out. On NT this value is used just to uniquely name the IPC Shared memory. On all other systems, this is the value of the IPC Id which JDEIPC uses for its shared memory. This plus the maxNumberOfResources define the range of IPC ids that JDE will used on the system. SysAdmins should ensure that this range of Ids is not used by any other software. Although JDEIPC will not used an existing Id in its range, this may not be true of other software.

[JDEMAIL]

Setting	Typical Value	Purpose
mailServer=	owsmtp.jdedwards.com	The domain name of the SMTP server to be accessed for sending server mail messages.

[JDENET]

Setting	Typical Value	Purpose
netPgmName=	jdenet_n	The full path of the JDENET_N executable.
krnlPgmName=	jdenet_k	The full path of the JDENET_K executable.
serviceNameListen=	jde_server	Port number or service name used by OneWorld to communicate with clients and other servers.

Setting	Typical Value	Purpose
serviceNameConnect=	jde_server	Port number or service name used by OneWorld to communicate with clients and other servers.
maxNetProcesses=	1	Defines the maximum number of JDENET_N processes that can be running. You can increase the value for a server that is expecting heavy JDENET message flow.
maxNetConnections=	1250 (default for HP9000) 800 (default for RS/6000)	Total number of connections all JDENET_N processes can handle. This value is platform-specific. You can increase the value for a server that is expecting to handle larger number of workstations at the same time.
maxKernelProcesses=	12	Max number of JDENET_K processes can be running. The value should be greater than all the values add together in maxNumberOfProcesses of all the dedicated servers.
maxKernelRanges=	12	Number of dedicated server types
maxLenInlineData=	1024	For JDENET internal use only.
maxLenFixedData=	4096	For JDENET internal use only.
maxFixedDataPackets=	1024	For JDENET internal use only.
connectTimeout=	5 (default for HP9000) 4 (default for RS/6000)	Number of seconds to wait before a connection request expired. Increase the value if a workstation is experiencing connection failure.
singleProcess=	0	Set to 1 on OneWorld client. Set to 0 on OneWorld server.
netTrace=	1	Enables JDENET log messages.
enableMsgPriority=	0	Set to 1 to enable priority messaging, set to 0 to disable priority messaging. This feature is not currently being used by OneWorld and should be set to 0, because it can cause and overall decrease in performance.
enableGDMsgs=	0	Set to 1 to enable Guaranteed Delivery Messages, set to 0 to disable Guaranteed Delivery Messages. This feature is only used by Data Replication, so only enable it when using DR.

Setting	Typical Value	Purpose
gdmRetryInterval=	1000	To enable the sending of Guaranteed Delivery Messages, the following settings must be in the [JDENET] section. If the gdmRetryInterval is set to "0", the messages will only be retried at startup, otherwise they will be retried according to the number of seconds specified by the setting. For performance reasons, setting the retry interval to "0" is recommended usage on the workstation. Because workstations are not restarted as often, "0" is not a viable option for the servers, thus a high setting of around 1000 or so is recommended on a server. EnableGDMsgs=1 gdmRetryInterval=0
newProcessThreshold Connects=	0	Number of active connections JDENET_N process can handle before starting a new JDENET_N process. This setting is only used when the number of active JDENET_N processes is less than the value set in maxNetProcesses.
netAdministrator=	0	
kernelDelay=	0	Number of seconds JDENET_K process will sleep when started. Use for debugging purpose and should be 0 for production.
krnlCoreDump=	1	Not delivered with the RS/6000
netCoreDump=	1	Not delivered with the RS/6000

[JDENET_KERNEL_DEF1]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 1 kernel for internal and testing processes.

Setting	Value	Purpose
dispatchDLLName=	libjdenet.sl (default for HP9000) libjdenet.so (default for RS/6000 and Sun)	Identifies the name of the JDENET service program.
dispatchDLLFunction=	JDENET_Dispatch Message	The name of the JDENET function for handling JDENET messages. The dispatchDLLName and dispatchDLLFunction entries are platform specific.
maxNumberOfProcesses=	1	Max number of dedicated kernel processes can be running for this dedicated server.
beginningMsgTypeRange=	0	Beginning message type ID for this dedicated server.

endingMsgTypeRange=	255	Ending message type ID for this dedicated server
newProcessThresholdRequests=	0	Number of outstanding requests each dedicated kernel process can have before starting a new dedicated kernel process. This setting is only used when active dedicated kernel process running for this dedicated server is less than the value set in maxNumberOfProcesses.
numberOfAutoStartProcesses=	0	Indicates the number of processes to automatically start for this kernel type when the main JDENET process start up on a server. If this number is 0, then no kernel processes will start automatically. This number must be less than the maximum of processes (in setting maxNumberofProcesses) allowed for this kernel type. If the number is greater than the maximum allowed, then only the maximum number of processes will start.

[JDENET_KERNEL_DEF2]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 2 kernel to process OneWorld batch process (UBE) pass-through requests.

Setting	Typical Value	Purpose
dispatchDLLName=	libjdeknet.sl (default for HP9000) libjdekrnl.so (default for RS/6000) libjdeknet.so (default for Sun)	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchUBEMessage	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	256	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	511	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF3]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 3 kernel to process data replication requests.

Setting	Typical Value	Purpose
dispatchDLLName=	libjderepl.sl (default for HP9000) libjdekrnl.so (default for RS/6000) libjderepl.so (default for Sun)	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	DispatchRepMessage	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	512	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	550	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF4]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 4 kernel to process security server requests.

Setting	Typical Value	Purpose
dispatchDLLName=	libjdeknet.sl (default for HP9000) libjdekrnl.so (default for RS/6000) libjdeknet.so (default for Sun)	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchSecurity	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	551	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	580	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF5]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 5 kernel to processes transaction manager and lock manager requests.

Setting	Typical Value	Purpose
dispatchDLLName=	libtransmon.sl (default for HP9000) libjdekrnl.so (default for RS/6000) libtransmon.so (default for Sun)	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	TM_DispatchTransactionManager	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	This value should always be 1 for the transaction manager server.
beginningMsgTypeRange=	601	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	650	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF6]

This section defines JDENET internal dedicated server processes. It should not be changed. Oneworld uses the Type 6 kernel to process requests for remote Master Business Function (MBF). These requests are also referred to as CallObject requests using the jdeCallObject application programming interface (API).

Setting	Typical Value	Purpose
dispatchDLLName=	libjdeknet.sl (default for HP9000) libjdekrnl.so (default for RS/6000) libjdeknet.so (default for Sun)	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchCallObjectMessage	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	901	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	1156	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF7]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 7 kernel to process JDBNet server-to-server requests.

Setting	Typical Value	Purpose
dispatchDLLName=	libjdeknet.sl (default for HP9000) libjdekrnl.so (default for RS/6000) libjdeknet.so (default for Sun)	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchJDBNETMessage	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	1201	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	1456	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF8]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 8 kernel to process package installation requests.

Setting	Typical Value	Purpose
dispatchDLLName=	libjdeknet.sl (default for HP9000) libjdekrnl.so (default for RS/6000) libjdeknet.so (default for Sun)	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchPkgInstall Message	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	1501	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	1756	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF9]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 9 kernel to process requests for the OneWorld Server Administration Workstation (SAW) application.

Setting	Typical Value	Purpose
dispatchDLLName=	libjdesaw.sl (default for HP9000) libjdesaw.so (default for RS/6000) libjdesaw.so (default for Sun)	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchSAWMessage	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	2000 (default for HP9000) 2001 (default for RS/6000)	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	2256	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF10]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 10 kernel to process requests for the OneWorld Scheduler application.

Setting	Typical Value	Purpose
dispatchDLLName=	libjdeschr.sl (default for HP9000) libjdekrnl.so (default for RS/6000) libjdeschr.so (default for Sun)	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchScheduler	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	2501	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	2756	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	1	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF11]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 11 kernel to process package build requests.

Setting	Typical Value	Purpose
dispatchDLLName=	libjdeknet.sl (default for HP9000) libjdekrnl.so (default for RS/6000) libjdeknet.so (default for Sun)	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchPkgBuildMessage	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	3001	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	3256	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF12]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 12 kernel to support a single, dedicated kernel job for UBE subsystem requests.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekern.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchUBESBSMessage	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1. Do not set higher than 1.
beginningMsgTypeRange=	3501	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	3756	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.

[LOCK MANAGER]

Setting	Typical Value	Purpose
Server=	*server name*	This setting indicates the lock manager server to be used to process records. The value for this setting is the name of the server acting as the lock manager.
AvailableService=	NONE	This setting indicates the service that the lock manager server is offering. It is also used to indicate whether the lock manager server is on or off. Valid values are TS and NONE.
RequestedService=	NONE	This setting indicates the type of service that the workstation requests from the server. The service that is currently being provided by servers is time stamping (TS) only. Valid values are TS and NONE.

[NETWORK QUEUE SETTINGS]

Setting	Typical Value	Purpose
UBE Semaphore Key=	3600	
DefaultPrinterOUTQ=	*printer name*	The default printer to which batch applications will be routed.

[SECURITY]

Setting	Typical Value	Purpose
User=	JDESVR	OneWorld user
Password=	JDESVR	
DefaultEnvironment=	P733HPO1 (default for HP9000) P733RSO1 (default for RS/6000)	Defines a valid environment in which the path code defines F98OWSEC.
DataSource=	ORACLE PVC B733	The name of the OneWorld data source that contains the security tables.
SecurityServer=	*server name*	Name of the server that provides security services. Usually the current host.
ServerPswdFile=	TRUE	The setting of this parameter determines whether OneWorld uses special password handling for batch reports running on the server. Set the value to TRUE to instruct OneWorld to enable special handling of passwords. Set the value to FALSE to disable special handling. When OneWorld runs a batch report on the server, it runs the report using a string of line commands and parameters that includes the "user password". Under some operating systems, it is possible to query the status of a job and view the parameters that were used to start the process. As a security measure, you can enable special handling by OneWorld. When enabled, OneWorld does not include the "user password" in the parameter list for a batch process. Instead, it includes the name of a file that contains the "user password". OneWorld instructs the operating system to destroy this file as soon as the batch report reads the password.
History=	0	
SecurityMode=	0 (default), 1, or 2	This setting controls whether OneWorld uses the standard logon procedure, unified logon, or both. Enter 0 to accept only the standard logon, 1 to accept only the unified logon, or 2 to accept either, based on the setting in the workstation jde.ini file.
AllowedUsers=	*group or user names*	This setting for the unified logon feature enables you to specify users or groups who are allowed to use OneWorld. If no users or groups are specified, all users who have logged on to the proper domains are authenticated by the unified logon server.

[SVR]

Setting	Typical Value	Purpose
EnvType=	1	Used by JDEKRNL
EnvironmentName=	P733HPO1 (default for HP9000) P733RSO1 (default for RS/6000)	
SpecPath=	spec	This line and all of the following in this section specify the path names so other OneWorld source programs know where to look for files. For instance, if "spec" were ever to be changed to "specifications", changing SpecPath would allow this on the fly. This value is not updated by any program or process. The only reason to change this is aesthetic. This is the subdirectory under the path code user to store the replicated set of specification files on the workstation. This value is not updated by any program or process. There is really no good reason to ever change the name of this directory.
SourcePath=	source	
ObjectPath=	obj	
HeaderPath=	include	
HeaderVPath=	includev	
BinPath=	bin32	
LibPath=	lib32	
LibVPath=	libv32	
MakePath=	make	
WorkPath=	work	
CodeGeneratorPath=	cg	
ResourcePath=	res	
HelpPath=	helps	
NextIDPath=	nextid	
LibraryListName=	P733HPO1 (default for HP9000) P733RSO1 (default for RS/6000)	

[TAM]

Setting	Value	Purpose
TAMTraceLevel	0	

[UBE]

Setting	Value	Purpose
UBESubsystemLimit	3	Used to specify the number of subsystem jobs per report version.
UBEPrintDataItems	0 (default), or 1	Used to specify whether to print the associated data item description in the pdf file as meta data for third party vendors. 0 is no, and 1 is yes.
UBEDebugLevel	0 (default), 1, 2, 3, 4, 5, or 6	Used to specify what level of debugging information will be provided when using UBE debug logging. 0 is off, and 6 is the highest level of logging information. 0 = Error messages only 1 = Informative messages 2 = Section level messages 3 = Object level messages 4 = Event rules messages 5 = SQL statements 6 = UBE function messages
WebServer	1	This setting specifies whether the system enables the UBE feature from the web server and identifies the OneWorld kernel as a web kernel to meet the special needs of the web. If you leave this value blank, the calls from the business functions or the error message handling from the kernel will not work properly.

Understanding Windows NT Server jde.ini Settings

This section details the settings found in the OneWorld Windows NT server jde.ini file. Information is organized by section, such as [DEBUG]. Sections are alphabetized, but settings found within sections are generally listed in the order they are found in the software. For cases when defaults for Intel and Alpha differ, the two values are labeled. When the order of entries within a section differ between the two systems, the items are in the order found in the Intel jde.ini file.

[BSFN BUILD]

Setting	Value	Purpose
Build Area=	Z: BUILDS\BDEV\packages	The location on the server where the package will be built
OptimizationFlags=	/Gz /O1 /MD /W4 /GX /Gy /Fp$(PRECOMPHDR)	Machine dependent. These compile flags are used when building business functions in "Release" mode. You should not change these flags.
DebugFlags=	/Gz /O1 /Zi /MDd /Yd /W4 /GX /Gy /Fp$(PRECOMPHDR)	Machine dependent. These compile flags are used when building business functions in "Debug" mode. You should not change these flags.
InliningFlags=		Yes turns on inlining on the AS400. No turns it off. This entry is blank for non-AS/400 servers.
DefineFlags=	/D "WIN32" /D "_DEBUG" /D "WINDOWS" /D "IAMASERVER" /D "KERNAL"	
CompilerFlags=	/nologo /c -qspill=1024	Machine dependent. Valid compiler flags. The spill flag sets the stack space when business functions are compiled. J.D. Edwards has found that 1024 is adequate to compile the delivered business functions.
OSReleaseLevel=	5.0	Release level you are compiling to. You should not change these flags.
LinkFlags=	/DLL /DEBUG /SUBSYSTEM:windows /out:$(DLLTARGET) /PDB:$(PDB) /IMPLIB:$(LIBRARY) /FORCE:MULTIPLE /FORCE:UNRESOLVED /INCREMENTAL:YES /VERBOSE	Machine dependent. These flags are used when linking business functions. You should not change these flags.

Setting	Value	Purpose
LinkLibraries=	jdekrnl.lib, jdel.lib, jdenet.lib, or jdeipc.lib	Libraries to which business functions are linked. (Windows NT and AS/400 servers only.)
SimultaneousBuilds=	0	Indicates the number of DLLs that can be built at a time. O means that all will be built simultaneously.

[BSFN Builder]

Setting	Value	Purpose
User=	JDE	(Not used for Alpha)
Pwd=	JDE	(Not used for Alpha)
PathCode=	nextiddev	(Not used for Alpha)
BFDir=	BSFNERR	
Build Area=	z: builds\B733 (default for Intel) g: builds\B733 (default for Alpha)	
DBSFNFlags=	/Gz /O1 /Zi /MDd /Yd /W4 /GX /Gy /Fp$(PRECOMPHDR) /D “WIN32” /D “_DEBUG” /D “WINDOWS” /D “IAMASERVER” /D “KERNAL” /nologo /c	Machine dependent. Flags used to create optimized code.
RBSFNFlags=	/Gz /O2 /MD /W4 /GX /Gy /Fp$(PRECOMPHDR) /D “WIN32” /D “NDEBUG” /D “_WINDOWS” /D “IAMASERVER” /D “KERNAL” /nologo /c	Machine dependent. Flags used to create debug builds.

Setting	Value	Purpose
DLinkFlags=	/DLL /DEBUG /SUBSYSTEM:windows /out:$(DLLTARGET) /PDB:$(PDB) /IMPLIB:$(LIBRARY) /FORCE:MULTIPLE /FORCE:UNRESOLVED /INCREMENTAL:YES /VERBOSE	Machine dependent.
RLinkFlags=	/DLL /DEBUG /SUBSYSTEM:windows /out:$(DLLTARGET) /PDB:$(PDB) /IMPLIB:$(LIBRARY) /FORCE:MULTIPLE /FORCE:UNRESOLVED /VERBOSE /MAP:$(MAPTARGET) /OPT:REF	Machine dependent.

[DB SYSTEM SETTINGS]

Setting	Value	Purpose
Version=	43	A version number to prevent mismatch of jde.ini file with running version of OneWorld.
ODBCCursors=	1	
CursorType=	3	
Default User=		The user account name for the database bootstrap tables.
Default Pwd=		The user account password for the database bootstrap tables.
Default Env=	P733HPO1	The default data source on the workstation and/or the enterprise server.
Default PathCode=	PROD	The subdirectory under \$PKG under which the business function code is stored.
Base Datasource=	ORACLE SVR	Data source representing database from which logon information is retrieved.
Object Owner=	JDESVR4	Owner of system database tables.
Server=	hp9000a	Server on which database resides
Database=	hp9000adev1	The name of database where the system tables reside.
Load Library=	JDBOCI73.DLL	The JDE driver used to access the database holding the system tables. This depends on the database to be used and the type of system running OneWorld.
Decimal Shift=	Y	A flag to indicate if decimal shifting is used for numeric data.

Setting	Value	Purpose
Julian Dates=	Y	A flag to indicate if dates are stored in Julian or database-specific format.
Use Owner=	Y	A flag to indicate that tables names are to qualified by owner.
Secured=	Y	Indicates whether or not this is a secured database requiring a user and password login.
Type=	O (default), A, S, I	A single character denoting the type of database holding the system tables. These can be O (Oracle), A (MS Access), S (SQL Server), or I (Client Access, AS/400).

[DEBUG]– Intel

Setting	Typical Value	Purpose
LogErrors=	1	
Output=	FILE	Controls the status of the jdedebug file. Valid values are: NONE. No trace information is written to jdedebug.log. FILE. Database and runtime trace information is written to the file specified by the DebugFile= parameter in the [DEBUG] section. EXCFILE. Runtime trace information is written to the file specified by the DebugFile= parameter in the [DEBUG] section. BOTH. Trace information is written to both jde.log and jdedebug.log.
Trace=	TRUE	Writes additional trace information to the log files to aid in debugging.
DebugFile=	z:\jbuilds\b733\log\jdedebug.log	Location and name of the jdedebug.log file
JobFile=	z:\jbuilds\b733\log\jde.log	Location and name of jde.log file
ClientLog=	1 (default) or 0	1 enables servicing of business functions JDE.LOG and JDEDEBUG.LOG entries from the server to the workstation. 0 disables this service.
JDETSFile=		Specifies the location of the lock manager trace file.
KeepLogs=	1	

Setting	Typical Value	Purpose
RepTrace=	1	
TAMTraceLevel=	0	

[DEBUG]- Alpha

Setting	Typical Value	Purpose
DebugFile=	g:\jbuilds\b733\log\jdedebug.log	Location and name of the jdedebug.log file
JobFile=	g:\jbuilds\b733\log\jde.log	Location and name of jde.log file
LogErrors=	1	
Output=	FILE	Controls the status of the jdedebug file. Valid values are: NONE. No trace information is written to jdedebug.log. FILE. Database and runtime trace information is written to the file specified by the DebugFile= parameter in the [DEBUG] section. EXCFILE. Runtime trace information is written to the file specified by the DebugFile= parameter in the [DEBUG] section. BOTH. Trace information is written to both jde.log and jdedebug.log.
ClientLog=	1	
JDETSFile=		Specifies the location of the lock manager trace file.
Trace=	TRUE	
KeepLogs=	1	
RepTrace=	1	
TAMTraceLevel=	0	

[INSTALL]

Setting	Typical Value	Purpose
B733=	z:\builds\b733 (default for Intel) g: \builds\bdev (default for Alpha)	Same as JDE_KERNEL_DEF1.
LocalCodeSet=	WE_ISO88591	A setting used to determine alternate language usage. See "National Language Support" in the *OneWorld Upgrade Guide* (B73.3) for other values.

Setting	Typical Value	Purpose
WebAdmin=	1	This setting specifies whether the system generates all the Java objects for the default user. This includes overriding Java objects previously generated. If you leave this value blank, the system generates all the Java objects for the current user.
EnvCreation=	1 (default) to 5	This setting determines the number of environments that can be processed (loaded) at the same time.

[JDE_CG]

Setting	Typical Value	Purpose
TARGET=	RELEASE	Used by the code generator and global build program to determine the type of build. Customer should only build under release, as there will be conflicts with the release build of the tools if they build under debug.
INCLUDES=	c:\msdev\devstudio\vc\include;c:msdev\devstudio\sharedide\include;$(SYSTEM)\INCLUDE;$(SYSTEM)\INCLUDEV;$(SYSTEM)\CG;$(APP)\INCLUDE;	The path to the include directory (header files) used by the MSVC compiler. This value is updated by a workstation installation based on the user's deployment preferences.
LIBS=	c:\msdev\devstudio\vc\lib:c:\msdev\devstudio\sharedide\lib;$(SYSTEM)\LIB32;$(SYSTEM)\LIBV32;$(APP)\LIB32;	The path to the directory for libraries used by the MSVC and OneWorld Foundation Code. This value is updated by a workstation installation based on the user's deployment preferences.
MAKEDIR=	c:\msdev\devstudio\vc\bin;c:\msdev\devstudio\sharedide\bin	The path to the make directory used by the MSVC compiler. This value is updated by a workstation installation based on the user's deployment preferences.
STDLIBDIR=	c:\msdev\devstudio\vc\lib;c\msdev\devstudio\sharedide\lib	The path to the lib directory used by the MSVC compiler. This value is updated by a workstation installation based on the user's deployment preferences.

[JDEIPC]

Setting	Typical Value	Purpose
ipcTrace=	0	
;CLSID=		
;startIPCKeyValue	7001	On Windows NT this value is used just to uniquely name the IPC Shared memory. On all other systems, this is the value of the IPC ID which JDEIPC used for its shared memory. This plus the maxNumberOfResources define the range of IPC IDs that JDE will used on the system. System administrators should ensure that this range of IDs is not used by any other software. Although JDEIPC will not used an existing ID in its range, this may not be true of other software.

[JDEMAIL]

Setting	Typical Value	Purpose
mailServer=	owsmtp.jdedwards.com	The domain name of the SMTP server to be accessed for sending server mail messages.

[JDENET]

Setting	Typical Value	Purpose
netPgmName=	jdenet_n	The full path of the JDENET_N executable.
krnlPgmName=	jdenet_k	The full path of the JDENET_K executable.
serviceNameListen=	jde_server	
serviceNameConnect=	jde_server	
maxNetProcesses=	1	Defines the maximum number of JDENET_N processes that can be running. You can increase the value for a server that is expecting heavy JDENET message flow.
maxNetConnections=	400	Total number of connections all JDENET_N processes can handle. This value is platform-specific. You can increase the value for a server that is expecting to handle larger number of workstations at the same time.
maxKernelProcesses=	12	Max number of JDENET_K processes can be running. The value should be greater than all the values add together in maxNumberOfProcesses of all the dedicated servers.
maxKernelRanges=	12	Number of dedicated servers and types
maxLenInlineData=	1024	For JDENET internal use only.
maxLenFixedData=	4096	For JDENET internal use only.
maxFixedDataPackets=	1024	For JDENET internal use only.
connectTimeout=	4	Number of seconds to wait before a connection request expired. Increase the value if a workstation is experiencing connection failure.
singleProcess=	0	Set to 1 on OneWorld workstation. Set to 0 on OneWorld server.
netTrace=	0	
enableMsgPriority=	0	Set to 1 to enable priority messaging, set to 0 to disable priority messaging. This feature is not currently being used by OneWorld and should be set to 0, because it can cause and overall decrease in performance.

Setting	Typical Value	Purpose
enableGDMsgs=	0	Set to 1 to enable Guaranteed Delivery Messages, set to 0 to disable Guaranteed Delivery Messages. This feature is only used by Data Replication, so only enable it when using DR.
newProcessThreshold Connects=	0	Number of active connections JDENET_N process can handle before starting a new JDENET_N process. This setting is only used when the number of active JDENET_N processes is less than the value set in maxNetProcesses.
kernelDelay=	0	Number of seconds JDENET_K process will sleep when started. Use for debugging purpose and should be 0 for production.
gdmRetryInterval=	1000	To enable the sending of Guaranteed Delivery Messages, the following settings must be in the [JDENET] section. If the gdmRetryInterval is set to "0", the messages will only be retried at startup, otherwise they will be retried according to the number of seconds specified by the setting. For performance reasons, setting the retry interval to "0" is recommended usage on the workstation. Because workstations are not restarted as often, "0" is not a viable option for the servers, thus a high setting of around 1000 or so is recommended on a server. EnableGDMsgs=1 gdmRetryInterval=0
netBroadcastAddress=	INADDR_BROADCAST	OneWorld uses the INI setting to broadcast datagram control messages A datagram broadcast to IP address 10.255.255.255 will reach all hosts in the network with a 10.XXX.XXX.XXX IP address. INADDR_BROADCAST is the default broadcast address if no INI setting is found. A datagram broadcast to address INADDR_BROADCAST will reach only hosts in the same physical network.
;ServiceControlRefresh=	5	
netAdministrator=	0	
singleProcess=	0	

[JDENET_KERNEL_DEF1]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 1 kernel for internal and testing processes.

Setting	Value	Purpose
dispatchDLLName=	jdenet.dll	Identifies the name of the JDENET service program.
dispatchDLLFunction=	_JDENET_Dispatch Message@28 (default for Intel) JDENET_Dispatch Message (default for Alpha)	
maxNumberOfProcesses=	1	Max number of dedicated kernel processes can be running for this dedicated server.
beginningMsgTypeRange=	0	Beginning message type ID for this dedicated server.
endingMsgTypeRange=	255	Ending message type ID for this dedicated server
newProcessThresholdRequests=	2	Number of outstanding requests each dedicated kernel process can have before starting a new dedicated kernel process. This setting is only used when active dedicated kernel process running for this dedicated server is less than the value set in maxNumberOfProcesses.
numberOfAutoStartProcesses=	0	Indicates the number of processes to automatically start for this kernel type when the main JDENET process start up on a server. If this number is 0, then no kernel processes will start. THis number must be less than the maximum of processes (in setting maxNumberofProcesses) allowed for this kernel type. If the number is greater than the maximum allowed, then only the maximum number of processes will start.

[JDENET_KERNEL_DEF2]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 2 kernel to process OneWorld batch process (UBE) pass-through requests.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchUBEMessage@28 (default for Intel) JDEK_DispatchUBEMessage (default for Alpha)	
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	256	
endingMsgTypeRange=	511	
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF3]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 3 kernel to process data replication requests.

Setting	Typical Value	Purpose
dispatchDLLName=	Jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchRepMessage@28 (default for Intel) DispatchRepMessage (default for Alpha)	
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	512	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	550	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF4]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 4 kernel to process security server requests.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchSecurity@28 (default for Intel) _JDEK_DispatchSecurity (default for Alpha)	
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	551	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	580	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF5]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 5 kernel to processes transaction manager and lock manager requests.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchTransaction Manager@28 (default for Intel) TM_DispatchTransaction Manager (default for Alpha)	
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	601	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	650	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF6]

This section defines JDENET internal dedicated server processes. It should not be changed. Oneworld uses the Type 6 kernel to process requests for remote Master Business Function (MBF). These requests are also referred to as CallObject requests using the jdeCallObject application programming interface (API).

Setting	Typical Value	Purpose
dispatchDLLName=	Jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchCallObject Message@28 (default for Intel) JDEK_DispatchCallObject Message (default for Alpha)	
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	901	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	1156	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF7]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 7 kernel to process JDBNet server-to-server requests.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchJDBNET Message@28 (default for Intel) JDEK_DispatchJDBNETMessage (default for Alpha)	
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	1201	Same as JDE_KERNEL_DEF1.
EndingMsgTypeRange=	1456	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF8]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 8 kernel to process package installation requests.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.

dispatchDLLFunction=	_JDEK_DispatchPkgInstall Message@28 (default for Intel) JDEK_DispatchPkgInstall Message (default for Alpha)	
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	1501	Same as JDE_KERNEL_DEF1.
EndingMsgTypeRange=	1756	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF9]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 9 kernel to process requests for the OneWorld Server Administration Workstation (SAW) application.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchSAWMessage@ 28 (default for Intel) JDEK_DispatchSAWMessage (default for Alpha)	
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	2001	Same as JDE_KERNEL_DEF1.
EndingMsgTypeRange=	2256	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF10]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 10 kernel to process requests for the OneWorld Scheduler application.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchScheduler@28 (default for Intel) JDEK_DispatchScheduler (default for Alpha)	
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	2501	Same as JDE_KERNEL_DEF1.
EndingMsgTypeRange=	2756	Same as JDE_KERNEL_DEF1.

newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	1	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF11]

Setting	Typical Value	Purpose
dispatchDLLName=	jdekrnl.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	_JDEK_DispatchPkgBuild Message@28 (default for Intel) JDEK_DispatchPkgBuild Message (default for Alpha)	
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1.
beginningMsgTypeRange=	3001	Same as JDE_KERNEL_DEF1.
EndingMsgTypeRange=	3256	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.
numberOfAutoStartProcesses=	0	Same as JDE_KERNEL_DEF1.

[JDENET_KERNEL_DEF12]

This section defines JDENET internal dedicated server processes. It should not be changed. OneWorld uses the Type 12 kernel to support a single, dedicated kernel job for UBE subsystem requests.

Setting	Typical Value	Purpose
dispatchDLLName=	jdekern.dll	Same as JDE_KERNEL_DEF1.
dispatchDLLFunction=	JDEK_DispatchUBESBSMessage	Same as JDE_KERNEL_DEF1.
maxNumberOfProcesses=	1	Same as JDE_KERNEL_DEF1. Do not set higher than 1.
beginningMsgTypeRange=	3501	Same as JDE_KERNEL_DEF1.
endingMsgTypeRange=	3756	Same as JDE_KERNEL_DEF1.
newProcessThresholdRequests=	0	Same as JDE_KERNEL_DEF1.

[LOCK MANAGER]

Setting	Typical Value	Purpose
Server=	*server name*	This setting indicates the lock manager server to be used to process records. The value for this setting is the name of the server acting as the lock manager.
AvailableService=	NONE	This setting indicates the service that the lock manager server is offering. It is also used to indicate whether the lock manager server is on or off. Valid values are TS and NONE.
RequestedService=	NONE	This setting indicates the type of service that the workstation requests from the server. The service that is currently being provided by servers is time stamping (TS) only. Valid values are TS and NONE.

[NETWORK QUEUE SETTINGS]

Setting	Typical Value	Purpose
QEnv=	P733HPO1	
QUser=	JDESVR	
QPassword=	JDESVR	
QueueDelay=	30	
UBEPriority=	5	The priority set when the UBE is submitted. For servers, valid values are 0 to 5, where 0 is the highest priority setting. The priority setting is relative to other UBE jobs submitted by OneWorld.
JDENETTimeout=	60	
UBEQueues=	1 (default for Intel) 2 (default for Alpha)	
UBEQueue1=	QBATCH	
UBEQueue2=	RBATCH	
PackageQueues=	1	
PKGQueue1=	QBATCH	
SpcInstallQueues	1	This setting is necessary for manual server specification installations. It specifies the number of allowable specification installation queues. Valid values are from 0 to any number.
SpcQueue1	QBATCH	This setting specifies the name of the queue used for specification installation. This queue name is associated with the first jdequeue process that is created upon the startup of OneWorld on the server.
KillImmediate=	1	

[SECURITY]

Setting	Typical Value	Purpose
SecurityServer=	*server name*	
User=	JDESVR	
Password=	JDESVR	
DefaultEnvironment=	P733HPO1	Defines a valid environment in which the path code defines F98OWSEC.
DataSource=	ORACLE PVC B733	
History=	0	
SecurityMode=	0 (default), 1, or 2	This setting controls whether OneWorld uses the standard logon procedure, unified logon, or both. Enter 0 to accept only the standard logon, 1 to accept only the unified logon, or 2 to accept both.
AllowedUsers=	*group or user names*	This setting for the unified logon feature enables you to specify users or groups who are allowed to use OneWorld. If no users or groups are specified, all users who have logged on to the proper domains are authenticated by the unified logon server.

[SVR]

Setting	Typical Value	Purpose
SpecPath=	spec	This line and all of the following in this section specify the path names so other OneWorld source programs know where to look for files. For instance, if "spec" were ever to be changed to "specifications", changing SpecPath would allow this on the fly. This value is not updated by any program or process. The only reason to change this is aesthetic. This is the subdirectory under the path code user to store the replicated set of specification files on the workstation. This value is not updated by any program or process. There is really no good reason to ever change the name of this directory.
SourcePath=	source	
ObjectPath=	obj	
HeaderPath=	include	
HeaderVPath=	includev	
BinPath=	bin32	
LibPath=	lib32	
LibVPath=	libv32	
MakePath=	make	
WorkPath=	work	

[UBE]

Setting	Value	Purpose
UBEDebugLevel=	0 (default), 1, 2, 3, 4, 5, or 6	Used to specify what level of debugging information will be provided when using UBE debug logging. 0 is off, and 6 is the highest level of logging information. 0 = Error messages only 1 = Informative messages 2 = Section level messages 3 = Object level messages 4 = Event rules messages 5 = SQL statements 6 = UBE function messages
Exception=	1	Enable Exception Handling on Windows NT Client/Server.
UBESubsystemLimit	3	Used to specify the number of subsystem jobs per report version.
UBEPrintDataItems	0 (default), or 1	Used to specify whether to print the associated data item description in the pdf file as meta data for third party vendors. 0 is no, and 1 is yes.
WebServer	1	This setting specifies whether the system enables the UBE feature from the web server and identifies the OneWorld kernel as a web kernel to meet the special needs of the web. If you leave this value blank, the calls from the business functions or the error message handling from the kernel will not work properly.

Glossary

Glossary

AAI. See automatic accounting instruction.

action message. With OneWorld, users can receive messages (system-generated or user-generated) that have shortcuts to OneWorld forms, applications, and appropriate data. For example, if the general ledger post sends an action error message to a user, that user can access the journal entry (or entries) in error directly from the message. This is a central feature of the OneWorld workflow strategy. Action messages can originate either from OneWorld or from a third-party e-mail system.

ActiveX. A computing technology, based on object linking and embedding, that enables Java applet-style functionality for Web browsers as well as other applications. (Java is limited to Web browsers at this time.) The ActiveX equivalent of a Java applet is an ActiveX control. These controls bring computational, communications, and data manipulation power to programs that can "contain" them. For example, certain Web browsers, Microsoft Office programs, and anything developed with Visual Basic or Visual C++.

alphanumeric character. A combination of letters, numbers, and symbols used to represent data. Contrast with numeric character and special character.

API. See application programming interface.

applet. A small application, such as a utility program or a limited-function spreadsheet. It is generally associated with the programming language Java, and in this context refers to Internet-enabled applications that can be passed from a Web browser residing on a workstation.

application. In the computer industry, the same as an executable file. In OneWorld, an interactive or batch application is a DLL that contains programming for a set of related forms that can be run from a menu to perform a business task such as Accounts Payable and Sales Order Processing. Also known as system.

application developer. A programmer who develops OneWorld applications using the OneWorld toolset.

application programming interface (API). A software function call that can be made from a program to access functionality provided by another program.

application workspace. The area on a workstation display in which all related forms within an application appear.

audit trail. The detailed, verifiable history of a processed transaction. The history consists of the original documents, transaction entries, and posting of records, and usually concludes with a report.

automatic accounting instruction (AAI). A code that refers to an account in the chart of accounts. AAIs define rules for programs that automatically generate journal entries, including interfaces between Accounts Payable, Accounts Receivable, Financial Reporting, General Accounting systems. Each system that interfaces with the General Accounting system has AAIs. For example, AAIs can direct the General Ledger Post program to post a debit to a specific expense account and a credit to a specific accounts payable account.

batch header. The information that identifies and controls a batch of transactions or records.

batch job. A task or group of tasks you submit for processing that the system treats as a single unit during processing, for example, printing reports and purging files. The computer system performs a batch job with little or no user interaction.

batch processing. A method by which the system selects jobs from the job queue, processes them, and sends output to the outqueue. Contrast with interactive processing.

batch server. A server on which OneWorld batch processing requests (also called UBEs) are run instead of on a client, an application server, or an enterprise server. A batch server typically does not contain a database nor does it run interactive applications.

batch type. A code assigned to a batch job that designates to which J.D. Edwards system the associated transactions pertain, thus controlling which records are selected for processing. For

example, the Post General Journal program selects for posting only unposted transaction batches with a batch type of O.

batch-of-one immediate. A transaction method that allows a client application to perform work on a client workstation, then submit the work all at once to a server applicationfor further processing. As a batch process is running on the server, the client application can continue performing other tasks. See also direct connect, store and forward.

BDA. See Business View Design Aid.

binary string (BSTR). A length prefixed string used by OLE automation data manipulation functions. Binary Strings are wide, double-byte (Unicode) strings on 32-bit Windows platforms.

Boolean Logic Operand. In J.D. Edwards reporting programs, the parameter of the Relationship field. The Boolean logic operand instructs the system to compare certain records or parameters. Available options are:

EQ	Equal To.
LT	Less Than.
LE	Less Than or Equal To.
GT	Greater Than.
GE	Greater Than or Equal To.
NE	Not Equal To.
NL	Not Less Than.
NG	Not Greater Than.

browser. A client application that translates information sent by the World Wide Web. A client must use a browser to receive, manipulate, and display World Wide Web information on the desktop. Also known as a Web browser.

BSTR. See binary string.

business function. An encapsulated set of business rules and logic that can normally be reused by multiple applications. Business functions can execute a transaction or a subset of a transaction (check inventory, issue work orders, and so on). Business functions also contain the APIs that allow them to be called from a form, a database trigger, or a non-OneWorld application. Business functions can be combined with other business functions, forms, event rules, and other components to make up an application. Business functions can be created through event rules or third-generation languages, such as C. Examples of business functions include Credit Check and Item Availability.

business function event rule. Encapsulated, reusable business logic created using through event rules rather than C programming. Contrast with embedded event rule. See also event rule.

business view. Used by OneWorld applications to access data from database tables. A business view is a means for selecting specific columns from one or more tables whose data will be used in an application or report. It does not select specific rows and does not contain any physical data. It is strictly a view through which data can be handled.

Business View Design Aid (BDA). A OneWorld GUI ool for creating, modifying, copying, and printing business views. The tool uses a graphical user interface.

category code. In user defined codes, a temporary title for an undefined category. For example, if you are adding a code that designates different sales regions, you could change category code 4 to Sales Region, and define E (East), W (West), N (North), and S (South) as the valid codes. Sometimes referred to as reporting codes.

central objects. Objects that reside in a central location and consist of two parts: the central objects data source and central C components. The central objects data source contains OneWorld specifications, which are stored in a relational database. Central C components contain business function source, header, object, library, and DLL files and are usually stored in directories on the deployment server. Together they make up central objects.

check-in location. The directory structure location for the package and its set of replicated objects. This is usually \\deploymentserver\release\path_code\package\ packagename. The sub-directories under this path are where the central C components (source, include, object, library, and DLL file) for business functions are stored.

child. See parent/child form.

client/server. A relationship between processes running on separate machines. The server process is a provider of software services. The client is a consumer of those services. In essence, client/server provides a clean separation of function based on the idea of service. A server can service many clients at the same time and regulate their access to shared resources. There is a many-to-one relationship between clients and a server, respectively. Clients always initiate the dialog by requesting a service. Servers passively wait for requests from clients.

CNC. See configurable network computing.

configurable client engine. Allows user flexibility at the interface level. Users can easily move columns, set tabs for different data views, and size grids according to their needs. The configurable client engine also enables the incorporation of Web browsers in addition to the Windows 95- and Windows NT-based interfaces.

configurable network computing. An application architecture that allows interactive and batch applications, composed of a single code base, to run across a TCP/IP network of multiple server platforms and SQL databases. The applications consist of reusable business functions and associated data that can be configured across the network dynamically. The overall objective for businesses is to provide a future-proof environment that enables them to change organizational structures, business processes, and technologies independently of each other.

constants. Parameters or codes that you set and the system uses to standardize information processing by associated programs. Some examples of constants are: validating bills of material online and including fixed labor overhead in costing.

control. Any data entry point allowing the user to interact with an application. For example, check boxes, pull-down lists, hyper-buttons, entry fields, and similar features are controls.

core. The central and foundation systems of J.D. Edwards software, including General Accounting, Accounts Payable, Accounts Receivable, Address Book, Financial Reporting, Financial Modeling and Allocations, and Back Office.

custom gridlines. A grid row that does not come from the database, for example, totals. To display a total in a grid, sum the values and insert a custom gridline to display the total. Use the system function Insert Grid Row Buffer to accomplish this.

data dictionary. The OneWorld method for storing and managing data item definitions and specifications. J.D. Edwards has an active data dictionary, which means it is accessed at runtime.

data mart. Department-level decision support databases. They usually draw their data from an enterprise data warehouse that serves as a source of consolidated and reconciled data from around the organization. Data marts can be either relational or multidimensional databases.

data replication. In a replicated environment, multiple copies of data are maintained on multiple machines. There must be a single source that "owns" the data. This ensures that the latest copy of data can be applied to a primary place and then replicated as appropriate. This is in contrast to a simple copying of data, where the copy is not maintained from a central location, but exists independently of the source.

data source. A specific instance of a database management system running on a computer. Data source management is accomplished through Object Configuration Manager (OCM) and Object Map (OM).

data structure. A group of data items that can be used for passing information between objects, for example, between two forms, between forms and business functions, or between reports and business functions.

data warehouse. A database used for reconciling and consolidating data from multiple databases before it is distributed to data marts for department-level decision support queries and reports. The data warehouse is generally a large relational database residing on a dedicated server between operational databases and the data marts.

data warehousing. Essentially, data warehousing involves off-loading operational data sources to target databases that will be used exclusively for decision support (reports and queries). There are a range of decision support environments, including duplicated database, enhanced analysis databases, and enterprise data warehouses.

database. A continuously updated collection of all information a system uses and stores. Databases make it possible to create, store, index, and cross-reference information online.

database driver. Software that connects an application to a specific database management system.

database server. A server that stores data. A database server does not have OneWorld logic.

DCE. See distributed computing environment.

default. A code, number, or parameter value that is assumed when none is specified.

detail. The specific pieces of information and data that make up a record or transaction. Contrast with summary.

detail area. A control that is found in OneWorld applications and functions similarly to a spreadsheet grid for viewing, adding, or updating many rows of data at one time.

direct connect. A transaction method in which a client application communicates interactively and directly with a server application. See also batch-of-one immediate, store and forward.

distributed computing environment (DCE). A set of integrated software services that allows software running on multiple computers to perform in a manner that is seamless and transparent to the end-users. DCE provides security, directory, time, remote procedure calls, and files across computers running on a network.

DLL. See dynamic link library.

duplicated database. A decision support database that contains a straightforward copy of operational data. The advantages involve improved performance for both operational and reporting environments. See also enhanced analysis database, enterprise data warehouse.

dynamic link library (DLL). A set of program modules that are designed to be invoked from executable files when the executable files are run, without having to be linked to the executable files. They typically contain commonly used functions.

dynamic partitioning. The ability to dynamically distribute logic or data to multiple tiers in a client/server architecture.

embedded event rule. An event rule that is specific to a particular table or application. Examples include form-to-form calls, hiding a field based on a processing option value, and calling a business function. Contrast with business function event rule. See also event rule.

employee work center. This is a central location for sending and receiving all OneWorld messages (system and user generated) regardless of the originating application or user. Each user has a mailbox that contains workflow and other messages, including Active Messages. With respect to workflow, the Message Center is MAPI compliant and supports drag and drop work reassignment, escalation, forward and reply, and workflow monitoring. All messages from the message center can be viewed through OneWorld messages or Microsoft Exchange.

encapsulation. The ability to confine access to and manipulation of data within an object to the procedures that contribute to the definition of that object.

enhanced analysis database. A database containing a subset of operational data. The data on the enhanced analysis database performs calculations and provides summary data to speed generation of reports and query response times. This solution is appropriate when external data must be added to source data, or when historical data is necessary for trend analysis or regulatory reporting. See also duplicated database, enterprise data warehouse.

enterprise data warehouse. A complex solution that involves data from many areas of the enterprise. This environment requires a large relational database (the data warehouse) that is a central repository of enterprise data, which is clean, reconciled, and consolidated. From this repository, data marts retrieve data to provide department-level decisions. See also duplicated database, enhanced analysis database.

enterprise server. A database server and logic server. See database server. Also referred to as host.

ERP. See enterprise resource planning.

event. An action that occurs when an interactive or batch application is running. Example events are tabbing out of an edit control, clicking a push button, initializing a form, or performing a page break on a report. The GUI operating system uses miniprograms to manage user activities within a form. Additional logic can be attached to these miniprograms and used to give greater functionality to any event within a OneWorld application or report using event rules.

event rule. Used to create complex business logic without the difficult syntax that comes with many programming languages. These logic statements can be attached to applications or database events and are executed when the defined event occurs, such as entering a form, selecting a menu bar option, page breaking on a report, or selecting a record. An event rule can validate data, send a message to a user, call a business function, as well as many other actions. There are two types of event rules:

1 Embedded event rules.
2 Business function event rules.

executable file. A computer program that can be run from the computer's operating system. Equivalent terms are "application" and "program.".

exit. 1) To interrupt or leave a computer program by pressing a specific key or a sequence of keys. 2) An option or function key displayed on a form that allows you to access another form.

facility. 1) A separate entity within a business for which you want to track costs. For example, a facility might be a warehouse location, job, project, work center, or branch/plant. Sometimes referred to as a business unit. 2) In Home Builder and ECS, a facility is a collection of computer language statements or programs that provide a specialized function throughout a system or throughout all integrated systems. For example, DREAM Writer and FASTR are facilities.

FDA. See Form Design Aid.

find/browse. A type of form used to:

1 Search, view, and select multiple records in a detail area.
2 Delete records.
3 Exit to another form.
4 Serve as an entry point for most applications.

firewall. A set of technologies that allows an enterprise to test, filter, and route all incoming messages. Firewalls are used to keep an enterprise secure.

fix/inspect. A type of form used to view, add, or modify existing records. A fix/inspect form has no detail area.

form. An element of OneWorld's graphical user interface that contains controls by which a user can interact with an application. Forms allow the user to input, select, and view information. A OneWorld application might contain multiple forms. In Microsoft Windows terminology, a form is known as a dialog box.

Form Design Aid (FDA). The OneWorld GUI development tool for building interactive applications and forms.

form interconnection. Allows one form to access and pass data to another form. Form interconnections can be attached to any event; however, they are normally used when a button is clicked.

form type. The following form types are available in OneWorld:

1 Find/browse.
2 Fix/inspect.
3 Header detail.
4 Headerless detail.
5 Message.
6 Parent/child.
7 Search/select.

fourth generation language (4GL). A programming language that focuses on what you need to do and then determines how to do it. Structured Query Language is an example of a 4GL.

graphical user interface (GUI). A computer interface that is graphically based as opposed to being character-based. An example of a character-based interface is that of the AS/400. An example of a GUI is Microsoft Windows. Graphically based interfaces allow pictures and other graphic images to be used in order to give people clues on how to operate the computer.

grid. See detail area.

GUI. See graphical user interface.

header. Information at the beginning of a table or form. This information is used to identify or provide control information for the group of records that follows.

header/detail. A type of form used to add, modify, or delete records from two different tables. The tables usually have a parent/child relationship.

headerless detail. A type of form used to work with multiple records in a detail area. The detail area is capable of of receiving input.

hidden selections. Menu selections you cannot see until you enter HS in a menu's Selection field. Although you cannot see these selections, they are available from any menu. They include such items as Display Submitted Jobs (33), Display User Job Queue (42), and Display User Print Queue (43). The Hidden Selections window displays three categories of selections: user tools, operator tools, and programmer tools.

host. In the centralized computer model, a large timesharing computer system that terminals communicate with and rely on for processing. In contrasts with client/server in that those users work at computers that perform much of their own processing and access servers that provide services such as file management, security, and printer management.

HTML. See hypertext markup language.

hypertext markup language. A markup language used to specify the logical structure of a document

rather than the physical layout. Specifying logical structure makes any HTML document platform independent. You can view an HTML document on any desktop capable of supporting a browser. HTML can include active links to other HTML documents anywhere on the Internet or on intranet sites.

index. Represents both an ordering of values and a uniqueness of values that provide efficient access to data in rows of a table. An index is made up of one or more columns in the table.

inheritance. The ability of a class to recieve all or parts of the data and procedure definitions from a parent class. Inheritance enhances developement through the reuse of classes and their related code.

install system code. See system code.

integrated toolset. Unique to OneWorld is an industrial-strength toolset embedded in the already comprehensive business applications. This toolset is the same toolset used by J.D. Edwards to build OneWorld interactive and batch applications. Much more than a development environment, however, the OneWorld integrated toolset handles reporting and other batch processes, change management, and basic data warehousing facilities.

interactive processing. Processing actions that occur in response to commands you enter directly into the system. During interactive processing, you are in direct communication with the system, and it might prompt you for additional information while processing your request. See also online. Contrast with batch processing.

interface. A link between two or more computer systems that allows these systems to send information to and receive information from one another.

Internet. The worldwide constellation of servers, applications, and information available to a desktop client through a phone line or other type of remote access.

interoperability. The ability of different computer systems, networks, operating systems, and applications to work together and share information.

intranet. A small version of the Internet usually confined to one company or organization. An intranet uses the functionality of the Internet and places it at the disposal of a single enterprise.

IP. A connection-less communication protocol that by itself provides a datagram service. Datagrams are self-contained packets of information that are forwarded by routers based on their address and the routing table information contained in the routers. Every node on a TCP/IP network requires an address that identifies both a network and a local host or node on the network. In most cases the network administrator sets up these addresses when installing new workstations. In some cases, however, it is possible for a workstation, when booting up, to query a server for a dynamically assigned address.

IServer Service. Developed by J.D. Edwards, this internet server service resides on the web server, and is used to speed up delivery of the Java class files from the database to the client.

ISO 9000. A series of standards established by the International Organization for Standardization, designed as a measure of product and service quality.

J.D. Edwards Database. See JDEBASE Database Middleware.

Java. An Internet executable language that, like C, is designed to be highly portable across platforms. This programming language was developed by Sun Microsystems. Applets, or Java applications, can be accessed from a web browser and executed at the client, provided that the operating system or browser is Java-enabled. (Java is often described as a scaled-down C++). Java applications are platform independent.

Java Database Connectivity (JDBC). The standard way to access Java databases, set by Sun Microsystems. This standard allows you to use any JDBC driver database.

jde.ini. J.D. Edwards file (or member for AS/400) that provides the runtime settings required for OneWorld initialization. Specific versions of the file/member must reside on every machine running OneWorld. This includes workstations and servers.

JDEBASE Database Middleware. J.D. Edwards proprietary database middleware package that provides two primary benefits:

1. Platform-independent APIs for multidatabase access. These APIs are used in two ways:
   a. By the interactive and batch engines to dynamically generate platform-specific SQL, depending on the datasource request.
   b. As open APIs for advanced C business function writing. These APIs are then

used by the engines to dynamically generate platform-specific SQL.

2. Client-to-server and server-to-server database access. To accomplish this OneWorld is integrated with a variety of third-party database drivers, such as Client Access 400 and open database connectivity (ODBC).

JDECallObject. An application programming interface used by business functions to invoke other business functions.

JDENET. J.D. Edwards proprietary middleware software. JDENET is a messaging software package.

JDENET communications middleware. J.D. Edwards proprietary communications middleware package for OneWorld. It is a peer-to-peer, message-based, socket based, multiprocess communications middleware solution. It handles client-to-server and server-to-server communications for all OneWorld supported platforms.

job queue. A group of jobs waiting to be batch processed. See also batch processing.

just in time installation (JITI). OneWorld's method of dynamically replicating objects from the central object location to a workstation.

just in time replication (JITR). OneWorld's method of replicating data to individual workstations. OneWorld replicates new records (inserts) only at the time the user needs the data. Changes, deletes, and updates must be replicated using Pull Replication.

KEY. A column or combination of columns that identify one or more records in a database table.

leading zeros. A series of zeros that certain facilities in J.D. Edwards systems place in front of a value you enter. This normally occurs when you enter a value that is smaller than the specified length of the field. For example, if you enter 4567 in a field that accommodates eight numbers, the facility places four zeros in front of the four numbers you enter. The result appears as: 00004567.

level of detail. 1) The degree of difficulty of a menu in J.D. Edwards software. The levels of detail for menus are as follows:

A Major Product Directories.
B Product Groups.
1 Basic Operations.
2 Intermediate Operations.
3 Advanced Operations.
4 Computer Operations.
5 Programmers.
6 Advanced Programmers Also known as menu levels.

2) The degree to which account information in the General Accounting system is summarized. The highest level of detail is 1 (least detailed) and the lowest level of detail is 9 (most detailed).

MAPI. See Messaging Application Programming Interface.

master table. A database table used to store data and information that is permanent and necessary to the system's operation. Master tables might contain data such as paid tax amounts, supplier names, addresses, employee information, and job information.

menu. A menu that displays numbered selections. Each of these selections represents a program or another menu. To access a selection from a menu, type the selection number and then press Enter.

menu levels. See level of detail.

menu masking. A security feature of J.D. Edwards systems that lets you prevent individual users from accessing specified menus or menu selections. The system does not display the menus or menu selections to unauthorized users.

Messaging Application Programming Interface (MAPI). An architecture that defines the components of a messaging system and how they behave. It also defines the interface between the messaging system and the components.

middleware. A general term that covers all the distributed software needed to support interactions between clients and servers. Think of it as the software that's in the middle of the client/server system or the "glue" that lets the client obtain a service from a server.

modal. A restrictive or limiting interaction created by a given condition of operation. Modal often describes a secondary window that restricts a user's interaction with other windows. A secondary window can be modal with respect to it's primary window or to the entire system. A modal dialog box must be closed by the user before the application continues.

modeless. Not restricting or limiting interaction. Modeless often describes a secondary window that does not restrict a user's interaction with other

windows. A modeless dialog box stays on the screen and is available for use at any time but also permits other user activities.

multitier architecture. A client/server architecture that allows multiple levels of processing. A tier defines the number of computers that can be used to complete some defined task.

network computer. As opposed to the personal computer, the network computer offers (in theory) lower cost of purchase and ownership and less complexity. Basically, it is a scaled-down PC (very little memory or disk space) that can be used to access network-based applications (Java applets, ActiveX controls) via a network browser.

network computing. Often referred to as the next phase of computing after client/server. While its exact definition remains obscure, it generally encompasses issues such as transparent access to computing resources, browser-style front-ends, platform independence, and other similar concepts.

next numbers. A feature you use to control the automatic numbering of such items as new G/L accounts, vouchers, and addresses. It lets you specify a numbering system and provides a method to increment numbers to reduce transposition and typing errors.

numeric character. Digits 0 through 9 that are used to represent data. Contrast with alphanumeric characters.

object. A self-sufficient entity that contains data as well as the structures and functions used to manipulate the data. For OneWorld purposes, an object is a reusable entity that is based on software specifications created by the OneWorld toolset. See also object librarian.

object configuration manager (OCM). OneWorld's Object Request Broker and the control center for the runtime environment. It keeps track of the runtime locations for business functions, data, and batch applications. When one of these objects is called, the Object Configuration Manager directs access to it using defaults and overrides for a given environment and user.

object embedding. When an object is embedded in another document, an association is maintained between the object and the application that created it; however, any changes made to the object are also only kept in the compound document. See also object linking.

object librarian. A repository of all versions, applications, and business functions reusable in building applications. It provides check-out and check-in capabilities for developers, and it controls the creation, modification, and use of OneWorld Objects. The Object Librarian supports multiple environments (such as production and development) and allows objects to be easily moved from one environment to another.

object linking. When an object is linked to another document, a reference is created with the file the object is stored in, as well as with the application that created it. When the object is modified, either from the compound document or directly through the file it is saved in, the change is reflected in that application as well as anywhere it has been linked. See also object embedding.

object linking and embedding (OLE). A way to integrate objects from diverse applications, such as graphics, charts, spreadsheets, text, or an audio clip from a sound program. See also object embedding, object linking.

object-based technology (OBT). A technology that supports some of the main principles of object-oriented technology: classes, polymorphism, inheritance, or encapsulation.

object-oriented technology (OOT). Brings software development past procedural programming into a world of reusable programming that simplifies development of applications. Object orientation is based on the following principles: classes, polymorphism, inheritance, and encapsulation.

OCM. See object configuration manager.

ODBC. See open database connectivity.

OLE. See object linking and embedding.

OneWorld. A combined suite of comprehensive, mission-critical business applications and an embedded toolset for configuring those applications to unique business and technology requirements. OneWorld is built on the Configurable Network Computing technology- J.D. Edwards' own application architecture, which extends client/server functionality to new levels of configurability, adaptability, and stability.

OneWorld application. Interactive or batch processes that execute the business functionality of OneWorld. They consist of reusable business functions and associated data that are platform

independent and can be dynamically configured across a TCP/IP network.

OneWorld object. A reusable piece of code that is used to build applications. Object types include tables, forms, business functions, data dictionary items, batch processes, business views, event rules, versions, data structures, and media objects. See also object.

OneWorld process. Allows OneWorld clients and servers to handle processing requests and execute transactions. A client runs one process, and servers can have multiple instances. OneWorld processes can also be dedicated to specific tasks (for example, workflow messages and data replication) to ensure that critical processes don't have to wait if the server is particularly busy.

OneWorld Web development computer. A standard OneWorld Windows developer computer with the additional components installed:

- JFC (0.5.1).
- Generator Package with Generator.Java and JDECOM.dll.
- R2 with interpretive and apllication controls/form.

online. Computer functions over which the system has continuous control. Users are online with the system when working with J.D. Edwards system provided forms.

open database connectivity (ODBC). Defines a standard interface for different technologies to process data between applications and different data sources. The ODBC interface is made up of a set of function calls, methods of connectivity, and representation of data types that define access to data sources.

open systems interconnection (OSI). The OSI model was developed by the International Standards Organization (ISO) in the early 1980s. It defines protocols and standards for the interconnection of computers and network equipment.

operand. See Boolean Logic Operand.

output. Information that the computer transfers from internal storage to an external device, such as a printer or a computer form.

output queue. See print queue.

package. OneWorld objects are installed to workstations in packages from the deployment server. A package can be compared to a bill of material or kit that indicates the necessary objects for that workstation and where on the deployment server the install program can find them. It is a point-in-time "snap shot" of the central objects on the deployment server.

package location. The directory structure location for the package and it's set of replicated objects. This is usually \\deployment server\release\path_code\package\ package name. The sub-directories under this path are where the replicated objects for the package will be placed. This is also referred to as where the package is built or stored.

parameter. A number, code, or character string you specify in association with a command or program. The computer uses parameters as additional input or to control the actions of the command or program.

parent/child form. A type of form that presents parent/child relationships in an application on one form. The left portion of the form presents a tree view that displays a visual representation of a parent/child relationship. The right portion of the form displays a detail area in browse mode. The detail area displays the records for the child item in the tree. The parent/child form supports drag and drop functionality.

partitioning. A technique for distributing data to local and remote sites to place data closer to the users who access. Portions of data can be copied to different database management systems.

path code. A pointer to a specific set of objects. A path code is used to locate:

1. Central Objects.
2. Replicated Objects.

platform independence. A benefit of open systems and Configurable Network Computing. Applications that are composed of a single code base can be run across a TCP/IP network consisting of various server platforms and SQL databases.

polymorphism. A principle of object-oriented technology in which a single mnemonic name can be used to perform similar operations on software objects of different types.

portability. Allows the same application to run on different operating systems and hardware platforms.

primary key. A column or combination of columns that uniquely identifies each row in a table.

print queue. A list of tables, such as reports, that you have submitted to be written to an output device, such as a printer. The computer spools the

tables until it writes them. After the computer writes the table, the system removes the table identifier from the list.

processing option. A feature of the J.D. Edwards reporting system that allows you to supply parameters to direct the functions of a program. For example, processing options allow you to specify defaults for certain form displays, control the format in which information prints on reports, change how a form displays information, and enter beginning dates.

program temporary fix (PTF). A representation of changes to J.D. Edwards software that your organization receives on magnetic tapes or diskettes.

published table. Also called a "Master" table, this is the central copy to be replicated to other machines. Resides on the "Publisher" machine. the Data Replication Publisher Table (F98DRPUB) identifies all of the Published Tables and their associated Publishers in the enterprise.

publisher. The server that is responsible for the Published Table. The Data Replication Publisher Table (F98DRPUB) identifies all of the Published Tables and their associated Publishers in the enterprise.

pull replication. One of the OneWorld methods for replicating data to individual workstations. Such machines are set up as Pull Subscribers using OneWorld's data replication tools. The only time Pull Subscribers are notified of changes, updates, and deletions is when they request such information. The request is in the form of a message that is sent, usually at startup, from the Pull Subscriber to the server machine that stores the Data Replication Pending Change Notification table (F98DRPCN).

purge. The process of removing records or data from a system table.

query by example (QBE). Located at the top of a detail area, it is used to search for data to be displayed in the detail area.

redundancy. Storing exact copies of data in multiple databases.

regenerable. Source code for OneWorld business functions can be regenerated from specifications (business function names). Regeneration occurs whenever an application is recompiled, eitherfor a new platform or when new functionality is added.

relationship. Links tables together and facilitates joining business views for use in an application or report. Relationships are created based on indexes.

release/release update. A "release" contains major new functionality, and a "release update" contains an accumulation of fixes and performance enhancements, but no new functionality.

replicated object. A copy or replicated set of the central objects must reside on each client and server that run OneWorld. The path code indicates the directory the directory where these objects are located.

run. To cause the computer system to perform a routine, process a batch of transactions, or carry out computer program instructions.

SAR. See software action request.

scalability. Allows software, architecture, network, or hardware growth that will support software as it grows in size or resource requirements. The ability to reach higher levels of performance by adding microprocessors.

search/select. A type of form used to search for a value and return it to the calling field.

selection. Found on J.D. Edwards menus, selections represent functions that you can access from a menu. To make a selection, type the associated number in the Selection field and press Enter.

server. Provides the essential functions for furnishings services to network users (or clients) and provides management functions for network administrators. Some of these functions are storage of user programs and data and management functions for the file systems. It may not be possible for one server to support all users with the required services. Some examples of dedicated servers that handle specific tasks are backup and archive servers, application and database servers.

servlet. Servlets provide a Java-based solution used to address the problems currently associated with doing server-side programming, including inextensible scripting solutions. Servlets are objects that conform to a specific interface that can be plugged into a Java-based server. Servlets are to the server-side what applets are to the client-side.

software. The operating system and application programs that tell the computer how and what tasks to perform.

software action request (SAR). An entry in the AS/400 database used for requesting modifications to J.D. Edwards software.

special character. A symbol used to represent data. Some examples are *, &, #, and /. Contrast with alphanumeric character and numeric character.

specifications. A complete description of a OneWorld object. Each object has its own specification, or name, which is used to build applications.

spool. The function by which the system stores generated output to await printing and processing.

spooled table. A holding file for output data waiting to be printed or input data waiting to be processed.

SQL. See structured query language.

static text. Short, descriptive text that appears next to a control variable or field. When the variable or field is enabled, the static text is black; when the variable or field is disabled, the static text is gray.

store and forward. A transaction method that allows a client application to perform work and, at a later time, complete that work by connecting to a server application. This often involves uploading data residing on a client to a server.

structured query language (SQL). A fourth generation language used as an industry standard for relational database access. It can be used to create databases and to retrieve, add, modify, or deleta data from databases. SQL is not a complete programming language because it does not contain control flow logic.

subfile. See detail.

submit. See run.

subscriber. The server that is responsible for the replicated copy of a Published Table. Such servers are identified in the Subscriber Table.

subscriber table. The Subscriber Table (F98DRSUB), which is stored on the Publisher Server with the Data Replication Publisher Table (F98DRPUB) identifies all of the Subscriber machines for each Published Table.

summary. The presentation of data or information in a cumulative or totaled manner in which most of the details have been removed. Many of the J.D. Edwards systems offer forms and reports that are summaries of the information stored in certain tables. Contrast with detail.

system. See application.

System Code. System codes are a numerical representation of J.D. Edwards and customer systems. For example, 01 is the system code for Address Book. System codes 55 through 59 are reserved for customer development by customers. Use system codes to categorize within OneWorld. For example, when establishing user defined codes (UDCs), you must include the system code the best categorizes it. When naming objects such as applications, tables, and menus, the second and third characters in the object's name is the system code for that object. For example, G04 is the main menu for Acounts Payable, and 04 is its system code.

system function. A program module, provided by OneWorld, available to applications and reports for further processing.

table. A two-dimensional entity made up of rows and columns. All physical data in a database are stored in tables. A row in a table contains a record of related information. An example would be a record in an Employee table containing the Name, Address, Phone Number, Age, and Salary of an employee. Name is an example of a column in the employee table.

table design aid (TDA). A OneWorld GUI tool for creating, modifying, copying, and printing database tables.

table event rules. Use table event rules to attach database triggers (or programs) that automatically run whenever an action occurs against the table. An action against a table is referred to as an event. When you create a OneWorld database trigger, you must first determine which event will activate the trigger. Then, use Event Rules Design to create the trigger. Although OneWorld allows event rules to be attached to application events, this functionality is application specific. Table event rules provide embedded logic at the table level.

TCP/IP. Transmission Control Protocol/Internet Protocol. The original TCP protocol was developed as a way to interconnect networks using many different types of transmission methods. TCP provides a way to establish a connection between end systems for the reliable delivery of messages and data.

TCP/IP services port. Used by a particular server application to provide whatever service the server is designed to provide. The port number must be readily known so that an application programmer can request it by name.

TDA. See table design aid.

Terminal Identification. The workstation ID number.Terminal number of a specific terminal or IBM user ID of a particular person for whom this is a valid profile.Header Field: Use the Skip to Terminal/User ID field in the upper portion of the form as an inquiry field in which you can enter the number of a terminal or the IBM user ID of a specific person whose profile you want the system to display at the top of the list. When you first access this form, the system automatically enters the user ID of the person signed on to the system.Detail Field: The Terminal/User ID field in the lower portion of the form contains the user ID of the person whose profile appears on the same line.A code identifying the user or terminal for which you accessed this window.

third generation language (3GL). A programming language that requires detailed information about how to complete a task. Examples of 3GLs are COBOL, C, Pascal and FORTRAN.

trigger. Allow you to attach default processing to a data item in the data dictionary. When that data item is used on an application or report, the trigger is invoked by an event associated with the data item. OneWorld also has three visual assist triggers: calculator, calendar and search form.

UDC Edit Control. Use a User-Defined Code (UDC) Edit Control for a field that accepts only specific values defined in a UDC table. Associate a UDC edit control with a database item or dictionary item. The visual assist Flashlight automatically appears adjacent to the UDC edit control field. When you click on the visual assist Flashlight, the attached search and select form displays valid values for the field. To create a UDC Edit Control, you must:

- Associate the data item with a specific UDC table in the Data Dictionary.
- Create a search and select form for displaying valid values from the UDC table.

uniform resource locator (URL). Names the address of a document on the Internet or an intranet. The following is an example of URL:http://www.jdedwards.com. This is J.D. Edwards Internet address.

user defined code (type). The identifier for a table of codes with a meaning you define for the system, such as ST for the Search Type codes table in Address Book. J.D. Edwards systems provide a number of these tables and allow you to create and define tables of your own. User defined codes were formerly known as descriptive titles.

user defined codes (UDC). Codes within software that users can define, relate to code descriptions, and assign valid values. Sometimes user defined codes are referred to as a generic code table. Examples of such codes are unit-of-measure codes, state names, and employee type codes.

valid codes. The allowed codes, amounts, or types of data that you can enter in a field. The system verifies the information you enter against the list of valid codes.

visual assist. Forms that can be invoked from a control to assist the user in determining what data belongs in the control.

vocabulary overrides. A feature you can use to override field, row, or column title text on forms and reports.

wchar_t. Internal type of a wide character. Used for writing portable programs for international markets.

web client. Any workstation that contains an internet browser. The web client communicates with the web server for OneWorld data.

web server. Any workstation that contains the IServer service, SQL server, Java menus and applications, and Internet middleware. The web server receives data from the web client, and passes the request to the enterprise server. When the enterprise server processes the information, it sends it back to the web server, and the web server sends it back to the web client.

window. See form.

workflow. According to the Workflow Management Coalition, worlflow means “the automation of a business process, in whole or part, during which documents, information, or tasks are passed from one participant to another for action, according to a set of procedural rules.”.

workgroup server. A remote database server usually containing subsets of data replicated from a master database server. This server does not

performance an application or batch processing. It may or may not have OneWorld running (in order to replicate data).

worldwide web. A part of the Internet that can transmit text, graphics, audio, and video. The World Wide Web allows clients to launch local or remote applications.

z file. For store and forward (network disconnected) user, OneWorld store and forward applications perform edits on static data and other critical information that must be valid to process an order. After the initial edits are complete, OneWorld stores the transactions in work tables on the workstation. These work table are called Z files. When a network connection is established, Z files are uploaded to the enterprise server and the transactions are edited again by a master business function. The master business function will then update the records in your transaction files.

Index

Index

A

B

C

D

E

F

G

H

I

J

L

M

N

O

P

Q

R

S

T

U

V

W