



What's New in eaSuite 3.4

V3.4
Document ID: DIGN-07-3.4-02
Date Published: 2.14.03

© 1997–2003 edocs® Inc. All rights reserved.

edocs, Inc., Two Apple Hill, 598 Worcester Road, Natick, MA 01760

The information contained in this document is the confidential and proprietary information of edocs, Inc. and is subject to change without notice.

This material is protected by U.S. and international copyright laws. edocs and eaPost are registered in the U.S. Patent and Trademark Office.

No part of this publication may be reproduced or transmitted in any form or by any means without the prior written permission of edocs, Inc.

eaSuite, eaDirect, eaPay, eaCare, eaAssist, eaMarket, and eaXchange are trademarks of edocs, Inc.

All other trademark, company, and product names used herein are trademarks of their respective companies.

Printed in the USA.

Table of Contents

- Preface 7**
 - About Customer Self-Service and eaSuite™ 7
 - Related Documentation 10
 - If You Need Help 11
- 1 New Features 13**
 - New in eaCare 13
 - Reporting and Analytics with eaCare 13
 - Enterprise Profile Management (EPM) with eaCare 14
 - New in eaDirect 14
 - Running Jobs Concurrently 14
 - Native XML Input with edocs XML Services 15
 - Defining Extended Functions For Group Fields 15
 - Positioning Starting Columns For DB Extract Files 16
 - DDN to Datasource Mapping 16
 - Sub-account Indexing 16
 - Cut, Copy, and Paste in DefTool 17
 - Custom Post-Conversion Builder 17
- 2 Reporting and Analytics Module 19**
 - Creating and Configuring a Report Job 19
 - Installing and Configuring Reporting 19
 - Creating An Indexer Job For Reporting 20
 - Creating And Configuring A Report Job 26
 - Indexing The Generated Report Data 30
 - Testing Reporting 31
- 3 EPM Console 35**
 - Using the EPM Console For Enterprise Profile Management 35
 - Setting Up the Hierarchy Console 36
 - Adding Attributes to the CDA Schema 37

Deploying the Hierarchy Web Application	39
Indexing Hierarchy Sample Data	40
Working with Hierarchies.....	40
SuperAdmin and Admin Roles and Privileges	40
Logging In to the Hierarchy Console.....	40
Navigating the Hierarchy Console	44
Creating Hierarchy Folders	46
Editing Hierarchy Folders.....	47
Deleting Hierarchy Folders	48
Enrolling Users in the Hierarchy	48
Assigning Accounts Into Hierarchy Folders	51
Assigning Users Into Hierarchy Folders.....	52
Searching for Accounts in the Hierarchy.....	53
Viewing Accounts For a User in the Hierarchy	53
Customizing the Hierarchy Console	54
Application Programming Interfaces (APIs)	54
com.edocs.hierarchy	54
com.edocs.hierarchy.app	54
com.edocs.hierarchy.render	54
com.edocs.hierarchy.taglib	55
4 Running Jobs Concurrently	57
Overview of Concurrency	57
Summary Job Information on the Main Console	58
Managing Individual Job Instances	59
Number of Rows On the Task Status Page.....	61
Canceling and Retrying Failed Jobs	63
Configuring and Scheduling Concurrent Jobs.....	65
Multiple Statement Processing.....	67
5 Native XML Input with edocs XML Services.....	69
About Native XML Input	69
Overview of XML Input Workflow	70
About XML Input Files for eaDirect	71
Enabling Your Web Application for XML Input	71
Creating and Configuring an XML Loader job	72
Task 1: Scanner	77
Task 2: XMLLoader.....	78

Task 3: XIR Loader	78
Task 4: AutoIndexVolAccept.....	79
Creating and Configuring an XML Email Notification Job	79
Task 1: IVNScanner.....	83
Task 2: XMLMailNotification	84
Index	85

About Customer Self-Service and eaSuite™

eaSuite

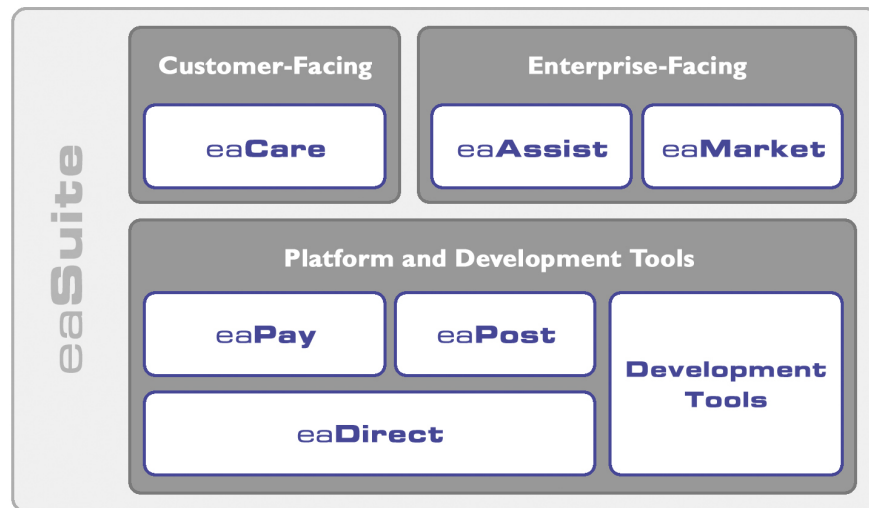
edocs has developed the industry's most comprehensive software and services for deploying Customer Self-Service solutions. eaSuite™ combines electronic presentment and payment (EPP), order management, knowledge management, personalization and application integration technologies to create an integrated, natural starting point for all customer service issues. eaSuite's unique architecture leverages and preserves existing infrastructure and data, and offers unparalleled scalability for the most demanding applications. With deployments across the healthcare, financial services, energy, retail, and communications industries, and the public sector, eaSuite powers some of the world's largest and most demanding customer self-service applications. eaSuite is a standards-based, feature rich, and highly scalable platform, that delivers the lowest total cost of ownership of any self-service solution available.

eaSuite is designed to support how organizations approach designing and deploying Customer Self-Service applications:

Customer-Facing Solutions present customers with the sophisticated functionality to meet customers' self-service needs. eaSuite offers a full set of capabilities to enable the range of business and consumer customer service activities, along with the flexibility to completely customize the solution to meet vertical industry and specific company requirements.

Enterprise-Facing Solutions empower employees within an organization and external partners to leverage the edocs platform to facilitate self-service and to support assisted service. Customer service representatives (CSRs), sales agents, account managers, marketing managers, broker-dealers and channel partners all play a role in delivering customer service, creating content, accessing information and performing activities for the benefit of customers.

Platform and Development Tools are designed to meet the rigorous infrastructure demands of the most technologically advanced organizations. These components of the eaSuite power edocs solutions with the functionality and development tools necessary to make account data available, and to create the customer- and enterprise-facing applications that enable customer self-service.



eaAssist

eaAssist™ reduces interaction costs and increases customer satisfaction by enabling enterprise agents – customer service representatives (CSRs), sales agents, broker-dealers and others – to efficiently access critical account data and service-related information to effectively service customers. Through its browser interface designed especially for the enterprise agent, eaAssist enables agents to take advantage of customer-facing online capabilities to provide better service by more efficiently resolving customer account inquiries at the point of customer contact.

eaMarket

eaMarket™ is the personalization, campaign and content management solution that enables organizations to increase revenue and improve customer satisfaction by weaving personalized marketing and customer service messages throughout the Customer Self-Service experience. The transactional account data that provides the foundation for a Customer Self-Service solution – such as transaction activity, service or usage charges, current task and prior service history – bring valuable insight into customers and can help optimize personalized marketing and customer service campaigns. eaMarket leverages that data to present relevant marketing and customer service messages to customers.

eaDirect

eaDirect™ is the core infrastructure of enterprise Customer Self-Service solutions for organizations large and small with special emphasis on meeting the needs of organizations with large numbers of customers, high data volumes and extensive integration with systems and business processes across the enterprise. Organizations use eaDirect with its data access layer, composition engine, and security, enrollment and logging framework to power complex Customer Self-Service applications.

eaPay

eaPay™ is the electronic payment solution that decreases payment processing costs, accelerates receivables and improves operational efficiency. eaPay is a complete payment scheduling and warehousing system with real-time and batch connections to payment gateways for Automated Clearing House (ACH) and credit card payments, and payments via various payment processing service providers.

eaPost

eaPost® is the account content distribution system that handles all the complexities of enrollment, authentication and secure distribution of summary account information to any endpoint, while also bringing customers back the organization's Website to manage and control their self-service experience.

Development Tools

eaSuite Development Tools™ are visual development applications that provide intuitive graphical user interface (GUI) environments for designing and developing Customer Self-Service solutions. The Development Tools encompass data management, workflow authoring, rules management and accounts receivable integration, as well as a full Software Developers Kit for custom application development.

Related Documentation

A PDF version of this guide is also available.

Online	How to Access
A PDF of this guide	A PDF of this guide is available on the eaDirect CD-ROM.

This guide is part of the eaDirect documentation set. For more information about implementing your eaDirect application, see one of the following guides:

Print Document	Description
eaDirect Installation and Configuration Guide	How to install eaDirect and configure it in a distributed environment.
eaDirect User's Guide	How to create data extraction and design files for an eaDirect application using the DefTool and Composer composition tools.
eaDirect Production Guide	How to set up and run a live eaDirect application in a J2EE environment.
eaDirect Developer's Guide	Provided with the eaDirect Software Developer's Kit (SDK), describes eaDirect application server components and related applications; defines the public interfaces for customizing and extending the functionality of an eaDirect application.

If You Need Help

Technical support is available to customers who have valid maintenance and support contracts with edocs. Technical support engineers can help you install, configure, and maintain your edocs application.

To reach the U.S. Service Center, located in Natick, MA (Monday through Friday 8:00am to 8:00pm EST):

- Telephone: 508.652.8400
- Toll Free: 877.336.3362
- E-support: support.edocs.com (This requires a one-time online registration)
- E-mail: support@edocs.com

When you report a problem, please be prepared to provide us the following information:

- What is your name and role in your organization?
- What is your company's name?
- What is your phone number and best times to call you?
- What is your e-mail address?
- In which edocs product did a problem occur?
- What is your Operating System version?
- What were you doing when the problem occurred?
- How did the system respond to the error?
- If the system generated a screen message, please send us that screen message.
- If the system wrote information to a log file, please send us that log file.

If the system crashed or hung, please tell us.

New Features



This document supplements the currently published guides for the eaSuite. For more information, please see the *eaDirect User's Guide*, *Production Guide*, and *Software Developer's Kit (SDK)*.

New in eaCare

Reporting and Analytics with eaCare

- Create pre-configured reports for large B2B volumes.
- Create telecommunications reports from indexed data with the new Report job.
- The Report job lets you create “Top 20” reports on data such as your most expensive calls, longest calls, most expensive called numbers, most called numbers, or most called countries, as well as reports summing cost by call types or time periods.

Benefits

- Establishes a reporting framework with a library of re-useable reports.
- Provides a pre-configured reporting model that scales and performs for huge B2B data volumes.
- Supports a generic syntax for telecommunications data and a method for configuring reports (XML template).

Enterprise Profile Management (EPM) with eaCare

- New Web console interface for creating and managing organizations.
- Manage structures of accounts and users via a Web Console, create groupings of sites, departments or regions.
- Pinpoint users and accounts with search and return lists.
- Includes user navigational access control in one console for easy management.
- Supports upload and export of data for updates and reuse.

Benefits

- Allows operations staff, customer service representatives or customer organizations themselves to create and manage their organizational hierarchy.
- Visual and intuitive Web console supports a broad spectrum of skills, using a flexible and open format that can be customized.
- Provides one Web console for a view of the organizational structure of accounts and users.
- Supports custom mappings and attributes with full create, update and maintenance of mappings, useful for extensions like Device IDs and tracking.

New in eaDirect

Running Jobs Concurrently

- Schedule multiple thread-safe jobs to run concurrently within an application, leveraging computing power to complete jobs faster.

- Specify the maximum number of concurrent instances to allow for each job: 5, 10, 15, or 20.
- Start up to 20 instances of a job in the same scheduled run, then monitor and manage instances on the Task Status page.
- Process multiple statements at one time with the new **Report** job. The **StatementScanner** task runs a job instance for each statement in the input file, optimizing processing time for a file with many documents.

Benefits

- Enhances flexibility for configuration improvements, especially for multiple application environments seeking maximum indexing performance.
- Provides hardware investments with new options for maximizing usage of existing hardware while preserving the existing scalability model.

Native XML Input with edocs XML Services

- eaDirect lets you extract and publish your data as XML using schemas, stylesheets, and other XML design files. Native XML input completes the end-to-end workflow by accepting input data in XML format and processing it with XSD schemas, attributes, and XSL stylesheets.

Benefits

- This scalable, persistent edocs solution lets you manage large volumes of XML data in real time, while saving system costs of processing and composing raw XML.

Defining Extended Functions For Group Fields

- You can now use Composer to create field elements that sort, filter, page, combine, calculate arithmetic, and analyze business conditions on group fields. For more information, see the *eaDirect User Guide*.

Benefits

- Provides additional composition flexibility and more options when working with grouped data structures.

Positioning Starting Columns For DB Extract Files

- When using DefTool to create a new DDF for a DB Extract input file, you can now specify a starting column position if the page delimiter is a string. For more information on positioning columns and pages in a DDF, see the *eaDirect User Guide*.

Benefits

- Provides additional flexibility when defining DB extract data sources.

DDN to Datasource Mapping

- Specify a custom datasource EJB for each eaDirect application directly in the Command Center, without hand-editing deployment descriptors and redeploying your web application.

Benefits

- Enhances the usability and scalability by allowing developers to have a different mapping for each web application.

Sub-account Indexing

- Increased flexible optimization controls specifically for massive B2B data volumes.
- Sub-account indexing speeds access through large data volumes directly to specific account sub-structures.

- The Development Environment and Command Center updated for new configuration support.

Benefits

- Increased control for configuring data sources in multiple application environments; configure applications for B2B or B2C data sources or mix configurations for consolidated applications.
- Delivers additional reporting options for cross-invoice reporting, summary reports, and trending at the summary level.

Cut, Copy, and Paste in DefTool

- You can now cut, copy, and paste Page styles, Fields (including group fields), Markers, Tables and columns, and Groups within a DDF or to another DDF.

Benefits

- Using cut, copy, and paste saves time and reduces errors when building multiple application views.

Custom Post-Conversion Builder

- Development Environment updated with new formatting specification builder.
- Default list of conversions still provided, but now it's easy to add your own.
- Create your new conversion routines with the "conversion builder."
- Add, save and re-use conversion routines.

Benefits

- Frees application developers to create their own format conversions.

New Features

- Especially useful for internationalization of applications where presentation requirements vary widely.

Creating and Configuring a Report Job

The Report job creates preconfigured telecommunication reports from indexed data for the following criteria:

- Top 20 Most Expensive Calls
- Top 20 Longest Calls
- Top 20 Most Expensive Called Numbers
- Top 20 Most Called Numbers
- Top 20 Most Called Countries
- Cost Sum by Call Types
- Cost Sum by Months
- Cost Sum by Time Periods

Installing and Configuring Reporting

Perform the following steps to install and configure your system to test reporting:

1. The reportSample EAR provides a way to view reports. If you haven't already installed ear-reportSample.ear, then do that first. Refer to the eaDirect Installation and Configuration Guide for your platform for more information about installing reportSample.

2. Create a new application to index the sample data. Sample DDFs, ALFs and templates are provided using NationalWireless data under the EDCSbd/samples/NW_Reports and EDCSbd/reports directories.
3. Create an Indexer job, and publish a DDF that contains the details on which your report will be based. Some of those details will be selected by the report description XML file (RDX) for the report data.

You also need to publish an XML dynamic web view using the same DDF, because the Report job translates the data into XML format as part of creating the reports.

4. Run the Report job, which creates a subdirectory for each of the eight report types and puts data into them that will be used to create reports.
5. Create new applications for each reporting type. Then, under each application, run the Indexer job to index the data for each report type, in the appropriate directory. This allows reportSample to display the reports to the user(s).

Creating An Indexer Job For Reporting

1. The Create New Job screen appears automatically after you create a new application. Otherwise click the application name on the Main Console, then click the **Add New Job** button. eaDirect displays the Create New Job screen:

Create New Job: myReport

When creating a job for an application that has just been created, you will need to publish the ALF, DDF and associated HTML template files. For adding additional jobs to an existing application, publishing the files is often not necessary, unless you wish to modify or create new versions of these files.

STEP 1 Name new job and select a job type.

Job Name:	<input type="text" value="Indexer"/>
Job Type:	<input type="text" value="Indexer"/> ▼

STEP 2 Publish application/job files and templates.

STEP 3 Configure and Schedule job.

© Copyright 1997-2002 edocs®, Inc. All Rights Reserved.
edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)

2. Enter a meaningful name for the Indexer job. The job name can contain alphanumeric characters and underscores, but no spaces. The Indexer job name cannot start with a numeric.
3. Select the Indexer job type from the drop-down menu.
4. Click **Launch Publisher**.

Select a Version Set Type	
Dynamic Web Views	
Job Type	Number of Auxiliary Files
HTML	0 1 2 3 more
XS	0 1 2 3 more
CSV	0
XML	0
CHART	0
XSLT	0
XMLQuery	0
Batch Jobs	
Job Type	Number of Auxiliary Files
Detail Extractor	0
Email Notification	0 1 2 3 more
HTML Output	0 1 2 3 more
Indexer	0
Report	0
XML Email Notification	0
XML Loader	0
XML Output	0

- Under Batch Jobs, next to Indexer, click [0](#) (Number of Auxiliary files). Publisher displays the Create a Version Set For Indexer screen:

Create a version set for Indexer

Application:

View Name: Indexer

DDF File:

© [Copyright](#) 1997-2002 edocs®, Inc. All Rights Reserved.
edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)

6. Select the application name from the drop-down list. For example, **myReport**.
7. Browse and select the DDF file for the Indexer job. The sample DDF file is *EDCSbd/samples/NW_Reports/NW_Reports.ddf*.
8. Click **Submit**. Publisher displays the Submission screen with details about the DDF file:

Submission

Application: esReport
View Type: DOC_CONFIG
View Name: Indexer
Timestamp: Fri Dec 13 13:07:40 GMT-05:00 2002

This version set contains the following files:

esReport/DOC_CONFIG/Indexer/20021213130740/NW_Reports.ddf

© Copyright 1997-2002 edocs®, Inc. All Rights Reserved.
edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)

9. Under Dynamic Web Views, next to XML, click 0 (Number of Auxiliary files). Publisher displays the Create a Version Set For XML screen:

Create a version set for XML

Application: ▾
View Type: XML
View Name:
DDF File:

© Copyright 1997-2002 edocs®, Inc. All Rights Reserved.
edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)

10. Select the application name from the drop-down list.

11. Enter **XmlDetail** for the View Name.
12. Browse and select a DDF file for the Indexer job. Enter the path to the same DDF entered for the Version Set for Indexer. For example, *EDCSbd/samples/NW_Reports/NW_Reports.ddf*.
13. Click **Submit**. Publisher displays the Submission screen with details about the DDF file.
14. Close the Publisher window.
15. At the Create New Job screen in Command Center, click **Configure Job and Continue**. eaDirect displays the job configuration screen:

Application: myReport Job: Indexer

From this screen, all parameters of the selected job can be modified. To edit parameters, change the entries in the desired fields and click the Submit Changes and Schedule button. To Reset the fields or for Help click the appropriate button at the of the screen.

Submit Changes and Schedule
Refresh
Reset
Help

Task 1: Scanner

Input File Path:	/export/spare/EDCSbd/Input/myReport/
Input File Name:	*. *
Output File Path:	/export/spare/EDCSbd/Data/myReport/

Task 2: Indexer

DDF Path:	/export/spare/EDCSbd/AppProfiles/myReport/DOC_CONFIG/Indexer/200212131806/NW_Reports.ddf									
Doc Date:	----Today's Date----									
Index Field List:	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="border: 1px solid gray; padding: 2px;">AmtDue</td> <td rowspan="4" style="border: none; text-align: center; vertical-align: middle;"> <div style="display: flex; justify-content: center; gap: 5px;"> >> << </div> </td> <td style="border: 1px solid gray; padding: 2px;">----- your selections here-----</td> </tr> <tr> <td style="border: 1px solid gray; padding: 2px;">CustName</td> <td style="border: 1px solid gray; padding: 2px;"></td> </tr> <tr> <td style="border: 1px solid gray; padding: 2px;">DueDate*</td> <td style="border: 1px solid gray; padding: 2px;"></td> </tr> <tr> <td style="border: 1px solid gray; padding: 2px;">StatementDate</td> <td style="border: 1px solid gray; padding: 2px;"></td> </tr> </table>	AmtDue	<div style="display: flex; justify-content: center; gap: 5px;"> >> << </div>	----- your selections here-----	CustName		DueDate*		StatementDate	
AmtDue	<div style="display: flex; justify-content: center; gap: 5px;"> >> << </div>	----- your selections here-----								
CustName										
DueDate*										
StatementDate										

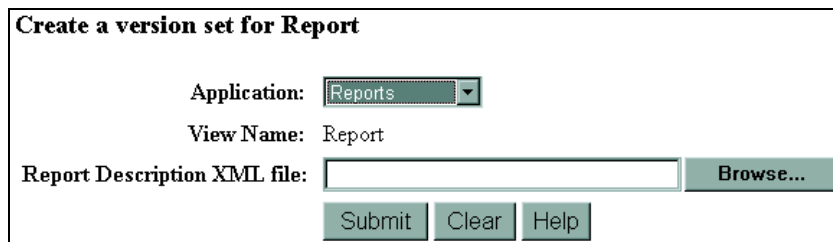
Task 3: IXLoader

16. For Task 2, select all the Index Fields. The fields that you select to be included in the Report output must also be defined in the RDX file.
17. When finished entering configuration parameters, click **Submit Changes and Schedule**. eaDirect asks “OK to submit this configuration?”
18. Click **OK**. eaDirect submits the job configuration parameters and displays the Schedule screen. Before you schedule the Indexer job, be sure to copy the *NationalWireless.txt* file to the input directory. For example:

```
cd /opt/EDCSbd/samples/NW_Reports/datafile
cp NationalWireless.txt /opt/EDCSbd/Input/myReport
```
19. Click **Main Console**.

Creating And Configuring A Report Job

1. Create a job for your application, select the Job Type **Report**.
2. Click Launch Publisher, and click **Create**. Under Batch Jobs, next to Report, click 0 (Number of Auxiliary files). Publisher displays the Create a Version Set For Report screen:



Create a version set for Report

Application: Reports

View Name: Report

Report Description XML file: Browse...

Submit Clear Help

3. Browse to find the sample report description XML, select it, and click Submit to publish the XML file as a version set. For example, */opt/EDCSbd/samples/NW_Reports/nwReportDescription.xml*.
4. Close the Publisher browser window.
5. Submit the job, which brings up the job configuration screen.

Application: myReport Job: Report

From this screen, all parameters of the selected job can be modified. To edit parameters, change the Changes and Schedule button. To Reset the fields or for Help click the appropriate button at the top of

Submit Changes and Schedule Refresh Reset Help

Task 1: StatementScanner

Index Volume Status: Accepted

Scan IVN Starting From (Number of Days): 7

Account Resolver: JNDI name of the Account Resolver

Account Attribute:

Task 2: ReportTask

Report Description XML File: /export/spare/EDCSbd/AppProfiles/myReport/DOC_CONFIG/Report

XML View Name: XmlDetail

Report File Path: /export/spare/EDCSbd/ Input/

Task 3: ReportCollector

Output File Path: No user configuration necessary

Task 1: StatementScanner

The Statement Scanner scans the index volume table to create a list of statements (documents) being indexed and passes the doc id of the document to the Report task if the account is valid (enrolled).

StatementScanner Task Configuration (Report job)	
Field	What to enter
Index Volume Status	Specifies that the job can proceed when a date appears in either the Date Accepted or Date Rejected column in the volumes table.

StatementScanner Task Configuration (Report job)	
Field	What to enter
	<p>Accepted</p> <p>(Default) Choose this option if you do not have custom verification/audit application or if you have one and you want to run the job only after a volume has been approved.</p> <p>If you choose this option, IVNScanner looks for a date in the Date Accepted column; if it contains a date, it proceeds to run the job.</p>
	<p>Rejected</p> <p>Use this setting to run the job even if a volume has been rejected in the custom verification application (use this option for testing purposes only).</p> <p>If you choose this option, IVNScanner looks for a date in the Date Rejected column; if it finds a date, it proceeds to run the job.</p>
Scan IVN starting from (number of days)	Specify how many previous days' volumes to scan for; IVNScanner selects any volumes indexed on or between the current date and the number of days ago you specify. (7 is the default)
Account Resolver	Specifies the Resolver used for user account information. For example, <i>edx/reportSample/ejb/CDAAccountResolver</i> .
Account Attribute	Not used.

Task 2: ReportTask

The Report task uses the report description XML file specified here to extract data from the XML view named XmlDetail that was created when running the Indexer job. It then writes report files, one for each docid in the eight subdirectories created for the eight report types.

ReportTask Configuration (Report job)	
Field	What to enter
Report description XML	Contains the path to the last Report Description XML file published for your application. This field is read-only.
XML View name	The name of the view that you published for this job when configuring the Indexer job. The default is XmlDetail .
Report file directory	The path to the directory where the Report job writes the reports. Directories are created for each report. For this release there are eight directories, as described in the following table.

Subdirectory Name	Description
<applicationName>_r1	Top 20 most expensive calls
<applicationName>_r2	Top 20 longest calls
<applicationName>_r3	Top 20 most expensive called numbers
<applicationName>_r4	Top 20 most called numbers
<applicationName>_r5	Top 20 most called states
<applicationName>_r6	Cost sum by call type
<applicationName>_r7	Cost sum by month
<applicationName>_r8	Cost sum by time period

The reports in the RDX file map to report names in the same way as the preceding table. For example, **r1** maps to **Top 20 most expensive calls**.

Task 3: Collector

The report task generates one file for each docid in each of the report subdirectories. For each report type, the Collector task combines the reports for each docid into a single file.

No configuration is necessary for this task.

Submit and schedule the job. Be sure you have enabled concurrency for the eaDirect scheduler.



There must be at least one user enrolled to the application for the Report job to run successfully. Otherwise, the job will report a No-op in the Statement Scanner.

Indexing The Generated Report Data

The data generated by the Report job must be indexed so that the user can see it. You can use the sample DDFs with your ALF and HTML files to index the report data that was generated by the Report job. Create an application for each report type, and index the generated data so that it can be viewed with reportSample.

The following steps show how to index the generated report data for the "Top 20 most expensive calls" report:

1. Create a new application for <applicationName>_r1. For example, **myReport_r1**.
2. Create an Indexer job for that application. Then publish, use the following sample information.

For HTML under Dynamic Web View, choose 1 and enter the following information:

Resource	Value
view type	HTML
view name	HtmlDetail
DDF	<i>/opt/EDCSbd/samples/reports/reports_ddf/r1.ddf</i>
ALF	<i>/opt/EDCSbd/samples/NW_Reports/NW_Reports.alf</i>
HTML	<i>/opt/EDCSbd/samples/NW_Report/NW_Reports.htm</i>

For Indexer under Batch Jobs:

Resource	Value
DDF	<i>/opt/EDCSbd/samples/reports/reports_ddf/r1.ddf</i>

3. Close the Publisher, return to the Indexer job and check the following information

Resource	Value
Input File Path	<i>/opt/EDCSbd/Input/myReport_r1</i>
Input File Name	<i>*.txt</i>
Output File Path	Keep the default

Select the fields you want to index. In order to use report sample ear to view the NationalWireless data, you must at least choose the `AuxiField2` (doc id) filed in the index field list. Then save the job and schedule it.

4. Repeat the previous steps for the remaining report directories, using the DDF, ALF and HTML for the appropriate report directory (for example, **r2** for the second report).

For report 6 (**r6**), there are **two** HTML (*.htm*) files that must be published (the other reports only have one HTML file).

Testing Reporting

If you have used the sample DDFs and data files to install and configure the reporting feature, then you can use the following procedure to view the reports using the reportSample application.

1. Log on to the reportSample application, using one of the accounts from the file that you indexed for the Report job. Be sure to use the application name that you used to run the Report job as the DDN. The URL is *http://host:port/reportSample&ddn=applicationName*. For example:

`http://myHost:7001/reportSample?ddn=myReport`

edocs® ONLINE ACCOUNT MANAGEMENT & BILLING									
Account Summary					Edit Profile			Logout	
Account History... Click icons to view account detail and pay.									
View	Reports	Z_DOC_DATE	FIDSVERSION	AuxField01	AuxField02	AuxField03	AuxField04	AuxField05	AuxField06
	Report	01-06-2003	101	0331734	ivn-1/po-0/bc-16025/pc-8/dd-20030103	MARCH 25, 2001	04/19/01	224.7	



Be sure you select your main application for the DDN. If you choose one of the sub-applications as the DDN, you will not be able to view the report data. For example: use **myReport** as the DDN, but **NOT myReport_r1**.

- Click on the Report link to view the reports.

edocs® ONLINE ACCOUNT MANAGEMENT & BILLING	
Account Summary	Logout
© Copyright 1997-2003 edocs®, Inc. All Rights Reserved. Report specifications are Reg. U.S. Pat. & Tm. Off. Privacy Policy	
Select a call type	<input type="text" value="All call types"/>
Select a report type	<input type="text" value="Top 20 most expensive calls"/>
<input type="button" value="submit"/>	

- Select the call type and report type to view the reports. For example, select:

Parameter	Value
Call Type	Long Distance Peak time
Report Type	Top 20 Most Expensive Calls

Which will create a report similar to this:

101									
0407200									
0407200									
ivm-3/pc-16025/bc-22109/pc-10/dd-20030106									
MARCH 25, 2001									
04/19/01									
451.04									
BOBBYS BUILDING SERVICES									
Detail Fields									
01/25		11:58	803 738- 3769	COLUMBIA	NV	D	26.9	2.39	
01/31		04:04	262 367- 7216	HARTLAND	WI	D	24.7	2.20	
02/16		02:57	262 781- 6949	BEDFORD	WI	D	21.8	1.94	
02/15		02:15	803 738- 3769	COLUMBIA	NV	D	20.5	1.82	
01/25		11:33	704 555- 1212	DIR ASST		D	1.0	.99	
			704						

The eaSuite includes powerful new tools for customer self-care with eaCare. This chapter provides a summary of the eaSuite features that support enterprise profile management (EPM) for B2B hierarchies:

- The Hierarchy Console web application
- The APIs provided with the Hierarchy Console to customize the console

Using the EPM Console For Enterprise Profile Management

The EPM Console provides an intuitive user interface for administrators to organize and edit account information. Its flexible format is open for customization and import/export aware through edocs Common Directory Access (CDA). Use the Hierarchy Console to map accounts and subaccounts into account groups that can control access privileges, statement viewing, and other online account management.



The Hierarchy application supports a flexible B2B hierarchy. When changes occur within your organization, your Admin can make the corresponding changes in the Hierarchy application.



The Hierarchy Console does not perform any calculations directly. Rather, it passes data and metadata to other components that calculate statement data for presentment.

The Hierarchy Console itself does not control access to statements. Rather, an administrator can assign a user or account to a folder in the hierarchy, so that anyone with access privileges to that folder may use eaDirect to view statements for users in that folder.

Setting Up the Hierarchy Console

This procedure assumes you have installed eaDirect and the web application archive `ear-hierarchySample.ear`.

To set up the Hierarchy Console:

1. Create the CDA schema with the script `create_hierarchy_schema`. This creates the root node `o=Hierarchy`.
2. Deploy the sample EAR file `ear-hierarchySample.ear`.
3. Index the HierarchySample data by creating an `eaDirect` application and running the Indexer job on the sample data file.
4. Navigate to the Hierarchy Console at your application server and port, for example: `http://dusky:7001/hierarchySample/user/jsp/index.jsp`



Caution

The `o=Hierarchy` name is the *context root* of the `hierarchySample` web application. You cannot change this name without making a deployment descriptor change to `ejb-enrollment-hm.jar`. In `META-INF/ejb-jar.xml`, change the `contextRoot` value to your new context root. For more information on customizing application context, see the **SDK User Management Framework**.

5. Enroll an administrative user as a SuperAdmin with root level privileges.

For details on creating folders and enrolling users, see [Working with Hierarchies](#).

Adding Attributes to the CDA Schema

In this step, you set your classpath and run the CDA client to create the CDA schema for the Hierarchy Console. This procedure assumes you have created an `eaDirect` database as part of your installation.

You need classpath, environment, and password information from your database installation for this procedure. Write down your information in this table, and contact your system administrator if you need help.

Variable	Default/Example Values	Your Values Here
Login user	oracle	
Login password	edocs	
EDX_HOME	/opt/EDCSbd/	

Variable	Default/Example Values	Your Values Here
WL_HOME or WS_HOME	/opt/BEA/wlserver6.1	
JAVA_HOME	/export/home/bean/jdk131	
Server name	dusky	
Database port	1521	
Database instance	edx0	
DBA name	edx_dba	
DBA password	edx	

To create the default CDA schema for the hierarchy Console:

1. Set your environment with the shell script `setEnv.sh`, for example:

```
. $WL_HOME//config/mydomain/setEnv.sh
```

2. Log in as the edocs user (**nobody**) and set your CLASSPATH to include your edocs, application server, and Java home directories. For example:

```
CLASSPATH=$EDX_HOME/lib/edx_common.jar:$WL_HOME/lib/weblogic.jar:$WL_HOME /jdk131
```



Tip

If your JDK is set elsewhere in your environment, you may not need to include it in your classpath. If you receive errors while running the script, set your classpath to include your JDK.

3. Export your CLASSPATH.
4. Navigate to the sample directory for `create_hierarchy_schema` at `<EDX_HOME>/samples/hierarchySample/schema`
5. To start the CDA client and create the default CDA schema for the hierarchy, edit this command for your own environment with the values for your database server name and port and oracle username, password, and database instance.

```
java -Djdbc.drivers=oracle.jdbc.driver.OracleDriver
com.edocs.jndi.cda.cli.Main
jdbc:oracle:thin:@dusky:1521:edx0 edx_dba edx <
create_hierarchy_schema
```

6. Run your edited command. If the script executes successfully, you will see a series of caret prompts, like this: >>>>>>>>>>
7. If your application server is running, stop and start the server to pick up the changes to the CDA schema.
8. If you run the `create_hierarchy_schema` script again, or if you run another database script such as `create_training_schema`, you may need to remove or comment out any schema binding conflicts. For example, both these scripts create an `sb role` attribute, and the second script will throw a `SchemaBind` exception.



The application server caches schema information. If you make changes while the application server is running, you must stop and start your application server to pick up the changes to the CDA schema. If you receive errors enrolling a SuperAdmin, stop and start the app server before enrolling users.

By default, the `create_hierarchy_schema` script pre-populates the Hierarchy Console with a root level organization “Hierarchy.” You then use the Hierarchy Console to create your own hierarchy subfolders under this root level. You can then map account and user information from subdocuments into your hierarchy to group and organize accounts.

You must first create a root-level hierarchy folder (or run `create_hierarchy_schema`) and enroll at least one SuperAdmin user with root level privileges before you can assign users and accounts to folders. For more information, see [Creating Hierarchy folders](#).

Deploying the Hierarchy Web Application

Start your application server if it is not already running, and deploy the sample EAR file

```
<EDX_HOME>/samples/hierarchySample/J2EEApps/weblogic/ear
-hierarchySample.ear
```

Indexing Hierarchy Sample Data

Your eaDirect installation places a sample DDF, ALF, and dataset for the Hierarchy Console in

```
<EDX_HOME>/samples/NW_Subdoc
```

Use the files in this folder to create an eaDirect application and run the Indexer job.

The Indexer job indexes all **subdocuments** defined in your DDF. For more information, see **Subdocument Indexing**.

Working with Hierarchies

SuperAdmin and Admin Roles and Privileges

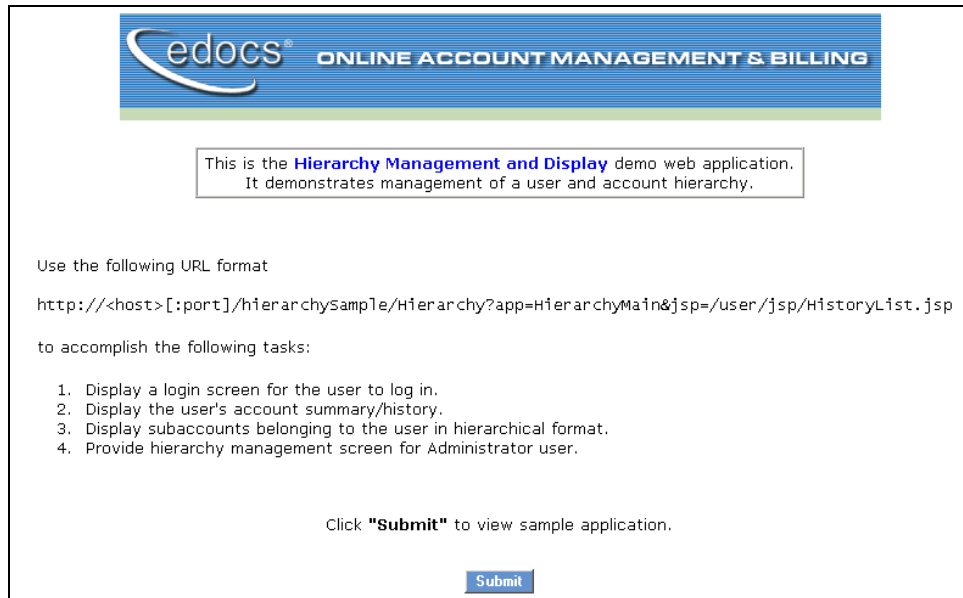
A **SuperAdmin** *registers* **primary accounts** with an **organization**. When a SuperAdmin logs in, the left pane of the Hierarchy Console presents a list of *unregistered* accounts containing all primary accounts for the DDN, for example IBM. These primary accounts can then be associated with an organization within the hierarchy, for example Northeast or Central.

An Admin *assigns* **subaccounts** into one or more organizations within the hierarchy. When an Admin logs in, the left pane of the Hierarchy Console presents a list of *unassigned* subaccounts for the Admin's organization(s), from all primary accounts that the SuperAdmin has registered with that organization. For example, these subaccounts could be the telephone numbers of IBM employees in the Northeast, which the Admin could then assign to the Boston or New York organizations.

Logging In to the Hierarchy Console

To bring up the Hierarchy login screen, use the following URL format for your own host and port:

`http://<host>[:port]/hierarchySample/user/jsp/index.jsp`



The screenshot shows a web page with a blue header containing the 'edocs' logo and the text 'ONLINE ACCOUNT MANAGEMENT & BILLING'. Below the header is a white box with a black border containing the text: 'This is the **Hierarchy Management and Display** demo web application. It demonstrates management of a user and account hierarchy.' Below this box, the text reads: 'Use the following URL format' followed by the URL `http://<host>[:port]/hierarchysample/Hierarchy?app=HierarchyMain&jsp=/user/jsp/HistoryList.jsp`. Below the URL, it says 'to accomplish the following tasks:' followed by a numbered list: 1. Display a login screen for the user to log in. 2. Display the user's account summary/history. 3. Display subaccounts belonging to the user in hierarchical format. 4. Provide hierarchy management screen for Administrator user. At the bottom of the page, it says 'Click **"Submit"** to view sample application.' and there is a blue 'Submit' button.

Clicking the Submit button brings you to the login screen.

edocs® ONLINE ACCOUNT MANAGEMENT & BILLING

Enroll Now

User Login... Enter your username and password, then click "Submit".

Username:	SU
Password:	*****
DDN:	NW_Subdoc

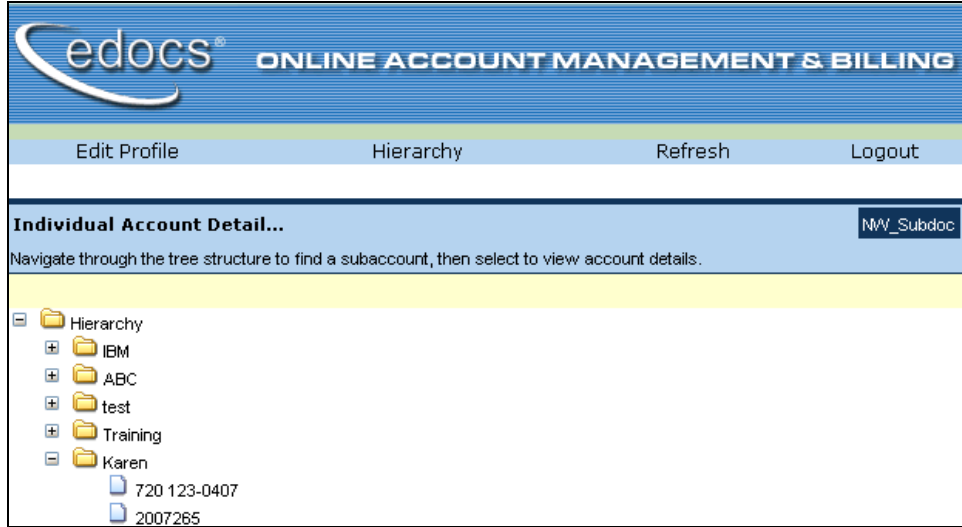
Submit **Reset**

© Copyright 1997-2002 edocs®, Inc. All Rights Reserved.
edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)

From this screen, you can log in as an existing user or enroll a new user. For enrollment, see [Enrolling Users in the Hierarchy](#).

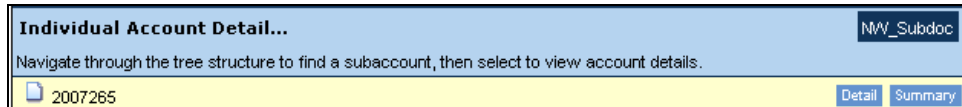
When a user logs in, she sees her assigned level in the hierarchy and may navigate down through any subtrees to view any accounts for which she has privileges.

For example, a **SuperAdmin** user has privileges to view all organizations in the Individual Account Detail pane, and may register primary accounts to these **organizations**. In this example, the SuperAdmin user **su** has logged in to the Hierarchy Console and can see the root level folder **Hierarchy** and all folders below it.



You must enroll at least one SuperAdmin user before enrolling users at lower levels in the hierarchy. For more information, see [Enrolling Users in the Hierarchy](#).

Select an account from the tree. It appears as a highlighted bar at the top of the Console pane.



Click the **Summary** button to view account data for this account.

Account History for 2007265... Click the view icon to display account detail.							
View	DocumentDate	CustName	StatementDate	AmntDue	State	CustType	DueDate
	10-10-2002	BLACK & BROWN CUSTOM LOG CABIN	MARCH 25, 2001	186.67	NM	B2B	04/19/01 TO
	10-10-2002	BLACK & BROWN CUSTOM LOG CABIN		186.67	NM		04/19/01 TO
	09-27-2002	BLACK & BROWN CUSTOM LOG CABIN		186.67	NM		04/19/01 TO

From the login screen, click **Hierarchy** in the navigation bar to go to the Hierarchy Console.

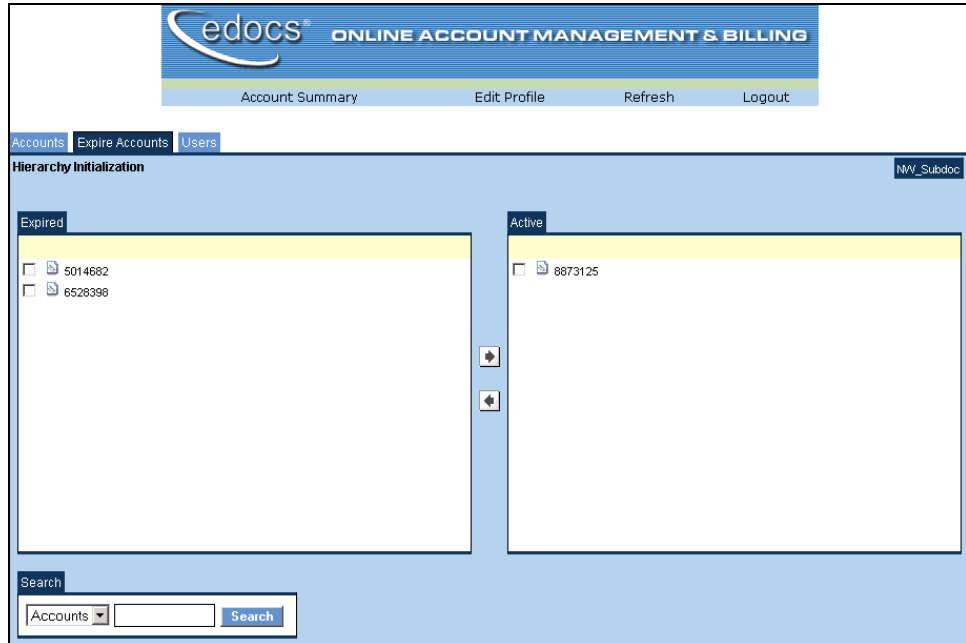
Navigating the Hierarchy Console

The Hierarchy Console has two panes, each containing a directory tree of nested folders, users, and accounts. The left pane has three tabs: **Accounts**, **Expire Accounts**, and **Users**.

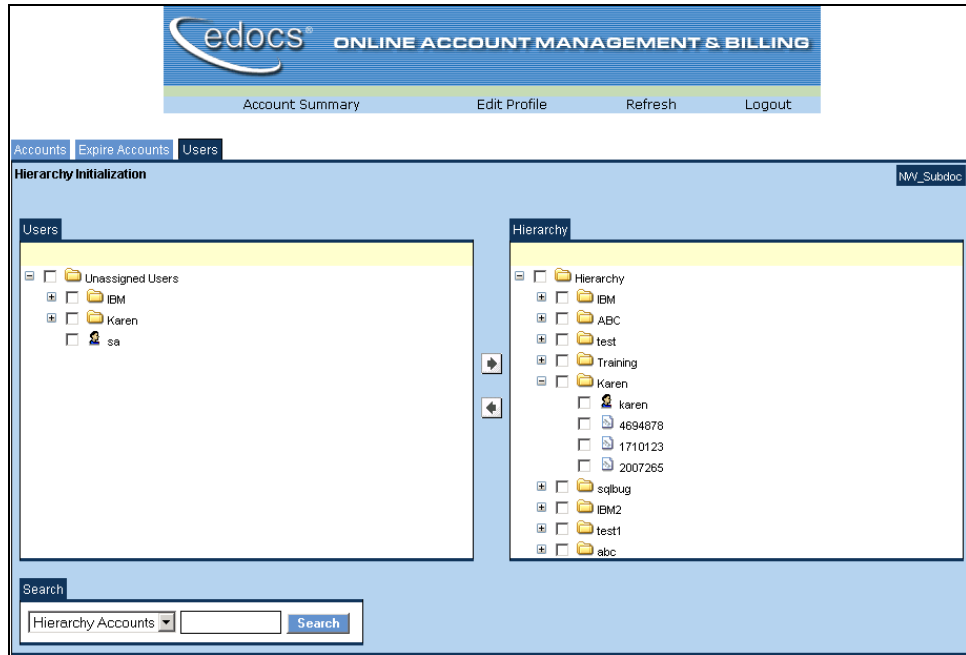
Click the **Accounts** tab to show all accounts not assigned to a folder in the hierarchy. To assign accounts, see [Assigning accounts into hierarchy folders](#).



Click the **Expire Accounts** tab to show a list of expired and active accounts. To change the status of an account, select its checkbox and use the left or right arrow.



Click the **Users** tab to show all users (including any Admins and SuperAdmins) not assigned to a folder in the hierarchy. To assign users, see [Assigning users into hierarchy folders](#).



In the right pane, the **Hierarchy** tree presents a directory of **nodes** in the hierarchy as folders. Clicking a folder expands it to show all accounts and users assigned to that node in the hierarchy.

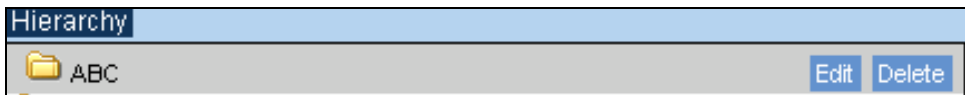
Creating Hierarchy Folders

An Admin or SuperAdmin may create folders at any level in the hierarchy for which they have privileges. A SuperAdmin must create at least a top-level folder for each organizational hierarchy before anyone can assign users or accounts.

To create a folder in the Hierarchy Console:

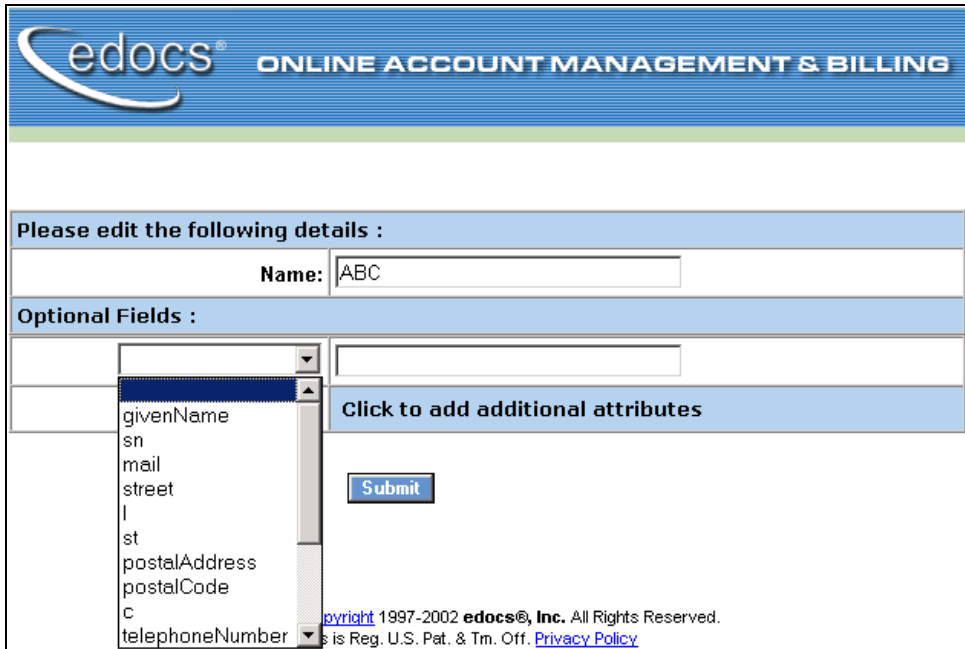
1. Bring up the hierarchy console and log in with your username, password, and DDN to go to the login page.
2. You will see the context root folder **Hierarchy**, created by the script **create_hierarchy_schema**. Within this root folder, you may then create subfolders for your Organizational Units, (ex. ABC, Gateway).

3. Clicking a folder highlights it at the top of the page with Edit and Delete buttons.



Editing Hierarchy Folders

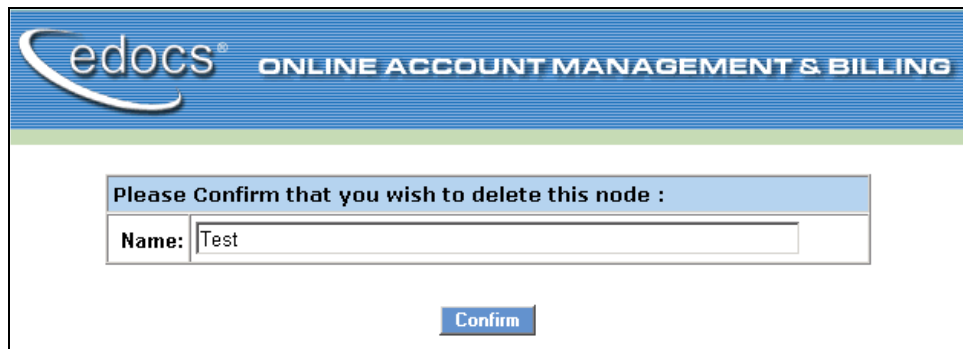
Highlight the folder and click **Edit**. You can add additional attributes to a folder through this screen.

A screenshot of the "edocs ONLINE ACCOUNT MANAGEMENT & BILLING" web application. The page has a blue header with the "edocs" logo and the text "ONLINE ACCOUNT MANAGEMENT & BILLING". Below the header is a section titled "Please edit the following details :". Under this section, there is a "Name:" label followed by a text input field containing "ABC". Below that is a section titled "Optional Fields :". To the left of this section is a vertical list of attributes: givenName, sn, mail, street, l, st, postalAddress, postalCode, c, and telephoneNumber. A dropdown menu is open over this list, showing the selected attribute "givenName". To the right of the list is a text input field. Below the input field is a button labeled "Click to add additional attributes". At the bottom of the form is a "Submit" button. At the very bottom of the page, there is a copyright notice: "Copyright 1997-2002 edocs®, Inc. All Rights Reserved. is Reg. U.S. Pat. & Tm. Off. Privacy Policy".

Deleting Hierarchy Folders

An Admin or SuperAdmin may delete folders at any level in the hierarchy for which they have privileges. You may not delete a folder containing accounts or subfolders (children or leaf nodes). Before you can delete a folder, you must unassign all accounts and users and delete any subfolders.

Highlight the folder and click **Delete**. A confirmation screen appears. Click **Confirm** to delete the folder.



The screenshot shows a confirmation dialog box within the edocs interface. The header of the interface reads "edocs® ONLINE ACCOUNT MANAGEMENT & BILLING". The dialog box contains the text "Please Confirm that you wish to delete this node :". Below this text is a form field labeled "Name:" with the value "Test" entered. At the bottom of the dialog box is a blue button labeled "Confirm".

Enrolling Users in the Hierarchy

Before you enroll users, you should create folders for any Organizations to which you plan to assign your users. You will use the enrollment screen to create a new Organization.

To enroll a new user from the login screen, click **Enroll Now**. The enrollment page appears.

The screenshot shows the edocs ONLINE ACCOUNT MANAGEMENT & BILLING interface. At the top, there is a blue header with the edocs logo and the text "ONLINE ACCOUNT MANAGEMENT & BILLING". Below the header, a text box states: "This **sample** demonstrates the flexibility of the **User Management Framework JNDI Implementation** by maintaining a hierarchical user enrollment schema." Below this, a paragraph explains: "When enrolling you must select the organization of which you are a member. The list of active organizations is provided below." A link labeled "here" is provided to create a new organization. The main part of the form is titled "Enter your subscription details:" and contains several input fields: Username, Password, Re-Type Password, Description, User Role (dropdown menu with "User" selected), Organization (dropdown menu with "NatlWireless" selected), Email Address, and Assign User into Hierarchy (dropdown menu with "No" selected). At the bottom of the form are "Submit" and "Reset" buttons.

This table shows the allowable values for each enrollment field.

Field Name	Allowable Values	Status	Notes
Username	User-defined	Required	Enter the username you wish to create for this user. This is the primary account key.

Field Name	Allowable Values	Status	Notes
Password	User-defined	Required	Enter the password you wish to create for this user. Retype the password for security.
Description	User-defined	Optional	Enter an optional description of this user, for example their title or name.
User Role	SuperAdmin Admin User <i>(Names are for this sample and may be customized in a custom schema)</i>	Required	Select the level of the hierarchy you want this user to see. A SuperAdmin can see all folders from the root level. An Admin can see all folders at and below their assigned folder. A User can see only their own account information.
Organization	Hierarchy North South IBM Coke	Required	Select your DDN or use the Click Here link to create a new organization.
Email Address	User-defined	Optional	Enter the e-mail address for this user.
Assign User into Hierarchy	Yes/No	Required	Yes automatically adds this user to the organization (folder) you define. This allows this user to see the Hierarchy tab in the Console at the folder level you define on enrollment. No leaves this user unassigned. Later, a SuperAdmin or Admin for this hierarchy can assign unassociated users to a folder.

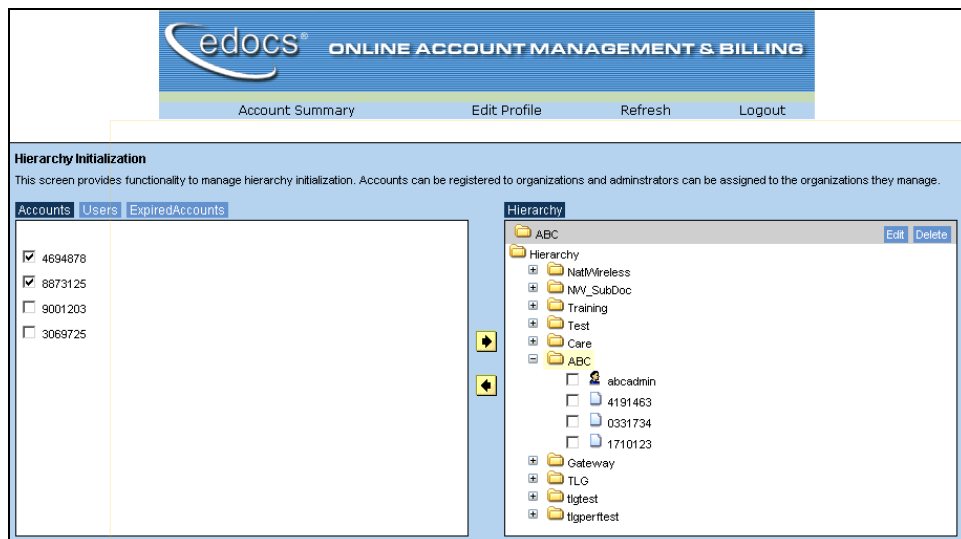
Assigning Accounts Into Hierarchy Folders

Admins and SuperAdmins may assign some or all accounts to a folder at any level of the hierarchy *except the context root*. A folder may contain multiple accounts, but an account may be assigned to only one folder.

Admins and SuperAdmins may also remove an account from a folder. This does not remove the account from the system, but simply means the account is no longer associated with that folder. The dissociated account reappears in the list of unassigned accounts.

To assign an account to a folder in the hierarchy:

1. Log in as the SuperAdmin or Admin and click the Hierarchy link. The Hierarchy Console displays all folders for which you have privileges.
2. Click the **Accounts** tab in the left pane to see a list of unassigned accounts.



3. In the left pane, select the checkbox for the account(s) you want to assign.
4. In the right pane, select the folder to which you are assigning them. The folder will highlight as shown.

5. Click the **right arrow**. The account(s) will appear within the assigned folder.

Assigning Users Into Hierarchy Folders

Admins and SuperAdmins may assign users to a folder at any level of the hierarchy, or assign a user directly to a single account. A folder may contain multiple users, but each user may be assigned to only one folder.

Admins and SuperAdmins may also remove a user from a folder. This does not remove the user from the system, but simply means the user is no longer associated with that folder.

To assign a user to a folder in the hierarchy:

1. Log in as the SuperAdmin or Admin and click the Hierarchy link to go to the Hierarchy Console. You will see all folders for which you have privileges.
2. Click the **Users** tab in the left pane to see a list of unassigned users.



3. In the left pane, select the checkbox for the user(s) you want to assign.
4. In the right pane, select the folder to which you are assigning them. The folder will highlight as shown.
5. Click the **right arrow**. The user(s) will appear within the assigned folder.

Searching for Accounts in the Hierarchy

To find an account in the hierarchy, use the Search box at the bottom of the Hierarchy Console screen. Select **Hierarchy Accounts** from the drop-down menu and enter an account number, then click **Search**. The Console displays the name of the Organizational Unit (ou) folder to which the account is assigned, in this example **Training**.

Search			
Hierarchy Accounts	8611250	Search	Clear
Hierarchy Context	Account Context	Account Number	Description
accountContext=8611250,ou=Training	8611250		

Viewing Accounts For a User in the Hierarchy

To view a list of accounts:

1. Enroll a User with the Hierarchy sample enrollment page (not the eaSample page).
2. Log in as that User and click the Account Summary link to go to the Individual Account Detail page.
3. The Individual Account Detail will display a list of available accounts, but not the Hierarchy assignment panes.

Customizing the Hierarchy Console

The Hierarchy Console **must be customized** with the `com.edocs.hierarchy` API to support advanced features, including:

- Paying statements directly through the Hierarchy Console (viewing is controlled by each web application, paying requires a payment solution)
- Customizing the Hierarchy Console user interface
- Searching the hierarchy
- Support for multiple DDNs, for example to group the accounts of a single customer with ten accounts across three DDNs.

For custom solutions integrating the Hierarchy Console, please consult your edocs Professional Services representative.

Application Programming Interfaces (APIs)

`com.edocs.hierarchy`

Provides an interface to define the values of constants for hierarchy fields and utility classes for creating a hierarchy as a directory information tree (DIT).

`com.edocs.hierarchy.app`

Provides servlet classes to extend and override the example servlets in `com.edocs.app` for user login and content access to account data mapped to the hierarchy.

`com.edocs.hierarchy.render`

Provides an interface and classes to design and display the user interface of the hierarchy console as HTML.

com.edocs.hierarchy.taglib

Provides a custom tag library of JSP tags for presenting hierarchy data.

For more information on Hierarchy APIs, please see the eaDirect Application Programming Interface Specification (Javadoc).

For example implementations of these APIs in the Hierarchy Console web application, please see the sample JSP pages in **ear-hierarchySample.ear**.

Running Jobs Concurrently

4

Overview of Concurrency

eaDirect lets you configure the Scheduler to enable multiple instances of an application job to run in parallel. The Command Center lets you monitor and manage the individual job instances to keep your production environment running efficiently.

Until now, eaDirect ran all jobs sequentially, requiring one job instance to complete before another the job could start again. If an application job processed large or multiple input files, repeating the job sequentially may not have allowed enough time to complete the job before another input file appeared in the input directory.

Running jobs in parallel enables you to leverage machine power to process a large amount of data in less time. You can configure the maximum number of concurrent job instances to allow for each job: 5, 10, 15, or 20.

Concurrency is available with thread-safe jobs only. **PurgeApp** and **PurgeLogs** are not thread-safe and can run only one occurrence at a time.

Concurrency also makes it possible for the new Report job to process statements in parallel with the StatementScanner task. See “Multiple Statement Processing” on Page 67 for more information.

Summary Job Information on the Main Console

For each eaDirect application, the Main Console lists each configured job type alphabetically. Although there can be multiple instances of an individual job for an application, the Main Console can display only one, so it chooses a representative job instance. eaDirect sorts all job instances first by status “ranking” and then by last run time in reverse chronological order. It selects the top-most instance from that list as the representative instance.

The Main Console displays the following information:

Command Center Main Console	
Column	Description
Application	Name of the eaDirect application.
Job Name	Name of the job.
Job Type	The purpose of the batch job: Indexer, Email Notification, Purge Apps, etc.
Last Run	Date and time the representative job instance ran.
Run Time	Elapsed time the representative job instance has been running in hours, minutes, and seconds.
Status	Current execution state of the representative job instance.
Next Run	Date and time the job is scheduled to run next. (This applies only to the job and not a particular instance.)
Action	Displays a button that lets you take action on that job. The Run Now button lets you run the job once immediately, overriding the scheduling parameters (except concurrency parameters). The Retry button lets you retry all failed instances of the job.

Jobs can have the following status, shown here in the order used for ranking purposes:

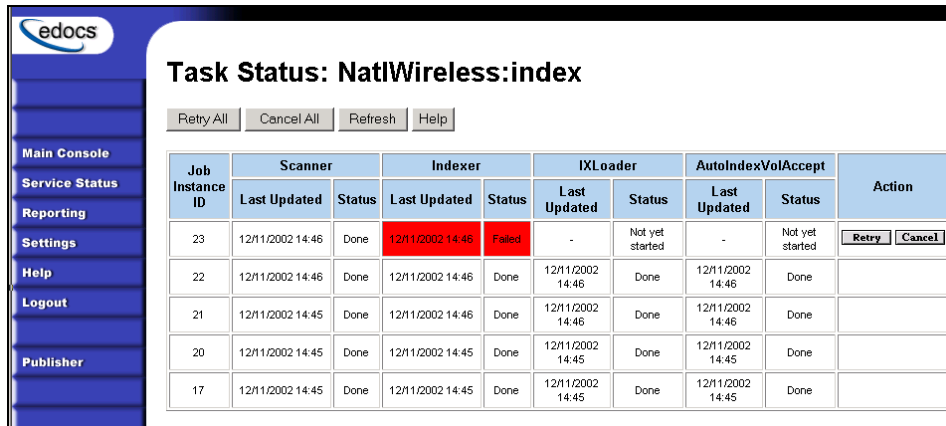
Job Status (In Ranking Order)	Description
Failed	Job failed
Processing	Job is currently executing
Reprocessing	Job is currently executing after a user manually selected it for reprocessing using the Retry or Retry All button
Reprocess	A user has manually selected the job for reprocessing using the Retry or Retry All button, but the job has not yet begun
No operation	Job/Task did nothing as resources were not ready yet, for example, if the Scanner task found no file in the input directory.
Done	Job has completed successfully
Canceled	Job run failed and was canceled
Not yet started	Job has not begun executing
Done, recurring	Job completed successfully and has been scheduled to run again, or the job has processed one data file and is looking at the input directory to see whether there are any more data files to process in this run
No operation, recurring	Previous job run resulted in a "No operation" status, but the job has been scheduled to run again
Canceled, recurring	Job was canceled and is now looking at the input directory to see whether there are any more data files to process in this run
Not scheduled	Job has not been scheduled to run

Managing Individual Job Instances

You can monitor and manage individual job instances using the Command Center Task Status page.

Running Jobs Concurrently

The Task Status page displays each instance of a job started during the most recent scheduled run in reverse chronological order (youngest first), along with the status of each task in the instance:



The screenshot shows the 'Task Status: NatlWireless:index' page. It features a navigation sidebar on the left with options like 'Main Console', 'Service Status', 'Reporting', 'Settings', 'Help', 'Logout', and 'Publisher'. The main content area has a title 'Task Status: NatlWireless:index' and buttons for 'Retry All', 'Cancel All', 'Refresh', and 'Help'. Below this is a table with columns for Job Instance ID, Scanner, Indexer, IXLoader, and AutoIndexVolAccept, each with sub-columns for 'Last Updated' and 'Status'. The table lists five job instances (IDs 17, 20, 21, 22, 23) in reverse chronological order. Instance 23 is the most recent and shows a 'Failed' status for the Indexer task, with 'Not yet started' for IXLoader and AutoIndexVolAccept. Instances 20, 21, and 22 show 'Done' for all tasks. Instance 17 also shows 'Done' for all tasks. Each row has 'Retry' and 'Cancel' buttons in the Action column.

Job Instance ID	Scanner		Indexer		IXLoader		AutoIndexVolAccept		Action
	Last Updated	Status	Last Updated	Status	Last Updated	Status	Last Updated	Status	
23	12/11/2002 14:46	Done	12/11/2002 14:46	Failed	-	Not yet started	-	Not yet started	<input type="button" value="Retry"/> <input type="button" value="Cancel"/>
22	12/11/2002 14:46	Done	12/11/2002 14:46	Done	12/11/2002 14:46	Done	12/11/2002 14:46	Done	
21	12/11/2002 14:45	Done	12/11/2002 14:46	Done	12/11/2002 14:46	Done	12/11/2002 14:46	Done	
20	12/11/2002 14:45	Done	12/11/2002 14:45	Done	12/11/2002 14:45	Done	12/11/2002 14:45	Done	
17	12/11/2002 14:45	Done	12/11/2002 14:45	Done	12/11/2002 14:45	Done	12/11/2002 14:45	Done	

The Task Status page identifies each job instance by a Job Instance ID. The Task Status page displays the following information:

Command Center Task Status Page	
Column	Description
Job Instance ID	A number uniquely identifying each job instance.
Last Updated	The time the task status last updated.
Status	Current execution state of the task. Task Status can be: Processing, Failed, Reprocessing, Reprocess, No operation, Canceled, Not yet started, or Done.
Action	Displays a button that lets you take action on that job instance or on all instances. The Retry button lets you retry that instance; the Retry All button lets you retry all failed instances of the job. The Cancel button lets you cancel that instance; the Cancel All button lets you cancel all failed instances of the job.

Number of Rows On the Task Status Page

The Task Status page displays up to the last N job instances that have Done, Canceled, or No operation status (where N is the maximum number of concurrent instances allowed for the job), plus any instances in Processing, Failed, Reprocessing, or Reprocess status.

If you are not using concurrency ($N=1$), the Task Status page shows up to five rows of job instances in Done, Canceled, or No operation status, plus any instances in Processing, Failed, Reprocessing, or Reprocess status.

When a scheduled run completes, the completed rows remain in view on the Task Status page until a new schedule begins. At this point, the Task Status page begins displaying the instances generated by the new schedule instead. The only exception is that any instances from the previous schedule still in Processing, Failed, Reprocessing, or Reprocess states remain even if a new schedule has begun. eaDirect removes those instances from the Task Status page once processing is complete, or in the case of a failed instance, once you cancel or retry it successfully.

Schedules can overlap if a second schedule begins before the current run completes. Another scheduled run can begin only if:

Running Jobs Concurrently

- The first run is not using the maximum number of instances (if enough “resource” is available). For example, if the first run has 3 instances in Processing and the maximum allowed is 10, the next run can start up to 7 new instances.
- No job instances in the first are in the Failed state.

Overlapping schedules mean that instances from both schedules could appear on the Task Status page. You can tell from the Last Updated field to which schedule the instance belongs.

The number of rows that appear on the Task Status page at any given time depends on the point of progress of the job plus:

- Whether you have enabled concurrency for the job (if the maximum number of instances specified in the schedule is >1).
- The maximum number of concurrent jobs you allow. This number is also the maximum number of Done, Canceled, or No operation jobs that can appear on the Task Status. If you are not using concurrency, the Task Status shows a maximum of 5 job instances in Done, Canceled, or No operation.
- For jobs that scan for an input file, such as Indexer, the number of input files placed in the input directory.

For jobs that process multiple statements in parallel with the **StatementScanner** task, such as the Report job, the number of statements to process up to the maximum number of instances.

- Whether the job schedule overlaps due to a long lasting run.

Examples: Task Status for the Indexer Job

Concurrency Enabled?	Number of Input Files	Maximum Instances	Number of Rows that Can Appear on the Task Status Page
No	3	1	1, 2, or 3. Each job instance must complete before another can appear. If all have been completed, 3 Done instances will show; by default the Task Status page can show up to 5 completed instances when the job runs sequentially (not concurrently).
Yes	3	10	1, 2, or 3. Three job instances can run concurrently and appear at once on the Task Status screen. (Up to 10 rows could appear at once if the job schedules overlap and more input files subsequently appear in the input directory during the second scheduled run.)
Yes	10	3	1-6. As few as one row with status in Processing or, if processing is complete, 3 rows all in Done status can appear. For example, at some point you could see 6 rows; the first instance might be in Processing, the second, third, and fourth in Done, the fifth in Failed, and the sixth in Processing, with the rest not started and therefore not shown.

Canceling and Retrying Failed Jobs

You can use the Main Console to cancel or retry a job, and the Task Status page to retry or cancel a failed instance of a job.

If one instance fails, other instances that have started continue to completion, but eaDirect does not start any new instances.

Retry running a failed job or job instance if you want to start it from the point where it failed. If you want to restart a job or instances of a job, cancel and run it again.

If the task has not been started, the Last Update field shows “-“ and Status shows Not Yet Started.

To retry a *failed job* before its next scheduled run time:

- On the Main Console, click the **Retry** button for the failed job. Or, on the Task Status page, click **Retry All**, which retries all failed instances of the job.

eaDirect immediately restarts all failed instances at the task that failed in each instance, and changes the job status from **Failed** to **Reprocess**.

To retry a *failed job instance*:

- On the Task Status page, click the **Retry** button in the Action section next to the failed instance.

eaDirect immediately restarts the failed job instance at the task that failed and changes the instance status from **Failed** to **Reprocess**.

To cancel all instances of a *failed job*:

- On the Task Status page, click the **Cancel All** button.

eaDirect cancels all failed instances of the job and changes the job status to Canceled, which remains until the next time the job is scheduled to run again.

To cancel a *failed job instance*:

- On the Task Status page, click **Cancel** in the Action section next to the failed instance.

eaDirect cancels the failed job instance and changes the job instance status to **Canceled**, which remains until the next time the job is scheduled to run again.

Configuring and Scheduling Concurrent Jobs

By default, eaDirect runs a single thread of each application job. To enable a job to run multiple threads, you must configure the job schedule and specify the maximum number of instances to allow at once: 5, 10, 15, or 20.

To configure a job to enable multiple instances to run:

1. At the eaDirect Command Center, locate the application and job you want to schedule to run multiple instances of, and click its status in the Next Run column. The Schedule screen appears. (If you just completed configuring a job, the Schedule screen appears automatically.)

edocs

Schedule Job: NatlWireless:Index

Schedule Date:
Start Date: [Popup Calendar](#)

Schedule Window:
Start Time: 12 am : 00

Try once
 Try every 5 minutes until 12 am : 00

Recurring:
 Do not repeat this event
 Repeat Every Day
 Repeat on day 1 of the month every month
 Forever
 Until [Popup Calendar](#)

Concurrency:
 Do not run multiple job instances, only one at a time
 Run maximum number 5 of concurrent job instances

Save Schedule Clear Schedule Run Now Help

2. Under the Concurrency section, select the **Run maximum number of (5) job concurrent instances** option.
3. To change the maximum number of instances to allow, click the dropdown box and choose 5, 10, 15, or 20.
4. Specify any other changes you need to make to the schedule parameters, if any. To clear the screen to reenter all parameters, click **Clear Schedule**.
5. When finished setting the schedule, click **Save Schedule**. eaDirect saves this schedule and adds the job to the production queue, overriding any scheduling parameters you set for a single execution of the job.

To configure a job to run a single instance:

- Follow the procedure above to configure multiple job instances, except choose the **Do not run multiple job instances, only one at a time** option.



Click the Run Now button on the Main Console to run just one instance of the job immediately, overriding the scheduling parameters (except concurrency parameters; if you saved the schedule to run multiple occurrences of a job, the Run Now button uses multiple occurrences instead of one).

Multiple Statement Processing

eaDirect introduces parallel statement processing with the new **StatementScanner** task in the new **Report** job. StatementScanner automatically processes multiple statements from an input file at the same time. eaDirect creates an individual Report job instance for each statement in the input file it processes.

For example, if you have an input file with 1,000 statements whose accounts are resolved as valid and you set the maximum concurrency for this job at 20, scheduler can start up to 20 job instances to process 20 statements simultaneously.

You can manage the multiple instances of any job that has the StatementScanner task in much the same way you manage concurrent instances generated by other job types. See the information in this chapter about managing job instances using the Main Console and the Task Status page. Also, see the chapter about the new Report feature, including the new Report job and StatementScanner task.

Native XML Input with edocs XML Services



About Native XML Input

eaDirect can extract and publish your data as XML using schemas, stylesheets, and other XML design files. Native XML input enhances end-to-end XML support in the eaSuite by accepting your data as well-formed XML, without the need for a Data Definition File (DDF). This scalable, persistent edocs solution lets you manage large volumes of XML data in real time, while saving the system costs of processing and composing raw XML.

eaDirect has two new Command Center batch jobs and a new type of dynamic web view designed expressly to accept XML input.

The **XML Loader** job processes an XML input file, much as the Indexer job handles AFP and other print files. However, unlike the Indexer job, the XML Loader job does not extract only selected (*indexed*) data for storage in the eaDirect database. In the XML model, eaDirect applies attribute tags to the entire input file, and the modified XML file is then stored in the XML repository for “extraction” at runtime with XSL stylesheets that process only the tagged data defined in the stylesheet.

The **XML Email Notification** job processes batch email from XML data, much as the Email Notification job handles batch email for AFP and other print files. An additional job setting allows you to specify a mail auditor. Instead of a DDF, ALF, and HTML templates, the XML version set for XML email notification takes an XSL stylesheet, and creates an XML Services (XS) web view type named **XENotification**.

The content and structure of the input XML is preserved while tagging the input XML data with attribute metadata. Input data is not removed or modified by applying attributes, and the modified data is stored persistently in the XML repository.

The XS (XML Services) web view publishes XML data much like an HTML view. Instead of a DDF, ALF, and HTML templates, the XML version set for this job takes an XSL stylesheet.

This chapter describes the production steps needed to process XML input.

Overview of XML Input Workflow

1. Create an XML schema for your native XML input file.
2. Create XSL style sheets for each view to be presented.
3. Work with edocs Professional Services to edit your schema to include edocs attributes for annotation, and to customize the data source in your web application as needed to use XML.
4. Create an eaDirect application for your XML jobs.
5. Create an XML Loader job.
6. Publish at least one XML Loader batch view.
7. Publish one or more XS (XML Services) web views for XML Loader.
8. Configure and schedule your XML Loader job
9. Create an XML Email Notification job.
10. Publish a batch view (**XENotification**) for XML Email Notification.
11. Configure and schedule your XML Email Notification job.

The topics below describe these steps in more detail.

About XML Input Files for eaDirect

You will need four types of XML files for using XML input with eaDirect:

- **Data (*.xml)**: one or more valid, well-formed XML file(s), which may consist of multiple XML files each containing multiple statements.
- **Schema (*.xsd)**: a valid W3C XML schema file of your XML data file(s). DTDs must be converted into a schema for use with eaDirect.



Tip

While it is not strictly necessary, it is good practice to name each schema file with the name of the related eaDirect application.

- **Attribute (edx.xsd)**: a special type of schema file defining edocs-specific variables that identify fields in your document with metatags, for extraction at runtime.



Tip

Attribute files work closely with your input data. Consult edocs Professional Services to develop a custom edx.xsd file for your dataset.

- **Stylesheet (*.xsl)**: one or more valid, well-formed XSL stylesheets, one for each web view you wish to publish.

For information on how to prepare your XML data files, schemas, attributes and stylesheets for eaDirect, please consult your edocs Professional Services representative.

Enabling Your Web Application for XML Input

Your web application will need to point its deployment descriptors to a custom EJB defining an XML data source for eaDirect data. For information on customizing your web application, please consult your edocs Professional Services representative.

Creating and Configuring an XML Loader job

You must create and configure an XML Loader job to use XML data for live retrieval with your application. This job requires you to:

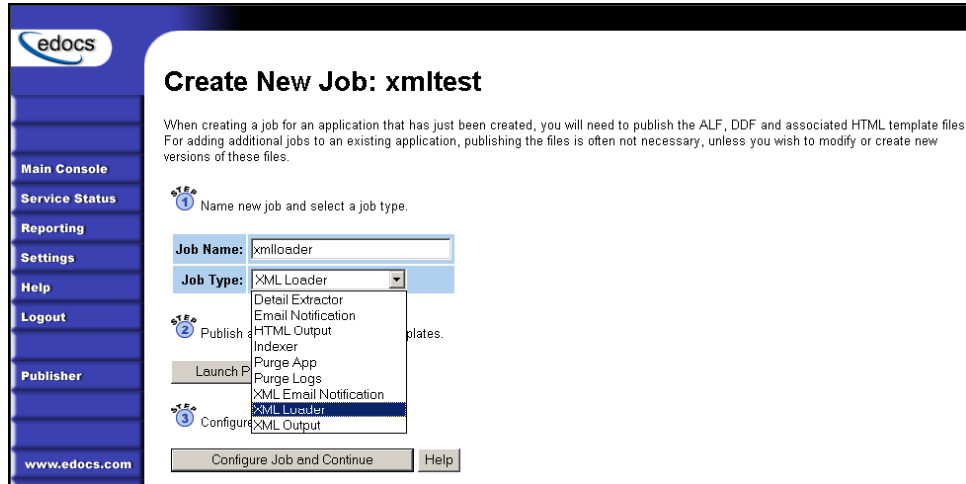
- Publish the schema and attribute files created for your XML Loader job
- Specify configuration settings for the three production tasks that run sequentially as part of the XML Loader job: Scanner, XMLLoader, and AutoIndexVolAccept

Review all the task and field configuration settings for this job to determine which options to use in your application.

Each time you run an XML Loader job, it looks for multiple data files in the input directory and processes them one at a time.

To create and configure an XML Loader job:

1. Create an XML schema for your native XML input file.
2. Create an eaDirect application for your XML input file, in this example **xmltest**.
3. The **Create New Job** screen appears automatically after you create a new application. Otherwise, click the application name on the Main Console, then click the **Add New Job** button. eaDirect displays the **Create New Job** screen:



4. Enter a meaningful name for the XML Loader job, in this example **xmlloader**. The job name may contain alphanumeric characters and underscores, but no spaces. The XML Loader job name may not start with a numeric.
5. Select the XML Loader job type from the drop-down menu.
6. Click Launch Publisher.
7. Click **Create**. Publisher displays the **Select a Version Set Type** screen:

The screenshot shows the edocs web interface. On the left is a blue navigation sidebar with buttons for Browse, Create, Fetch, Delete, Help, and the website URL www.edocs.com. The main content area is titled 'Select a Version Set Type' and contains two sections: 'Dynamic Web Views' and 'Batch Jobs'. Each section has a table with 'Job Type' and 'Number of Auxiliary Files' columns.

Dynamic Web Views

Job Type	Number of Auxiliary Files
HTML	0 1 2 3 more
XS	0 1 2 3 more
CSV	0
XML	0
CHART	0
XSLT	0
XMLQuery	0

Batch Jobs

Job Type	Number of Auxiliary Files
Detail Extractor	0
Email Notification	0 1 2 3 more
HTML Output	0 1 2 3 more
Indexer	0
XML Email Notification	0
XML Loader	0
XML Output	0

- Under Batch Jobs, next to XML Loader, click 0 (Number of Auxiliary files). Publisher displays the **Create a Version Set For XML Loader** screen:

The screenshot shows the edocs web interface. On the left is a blue sidebar with the edocs logo and a menu with buttons for Browse, Create, Fetch, and Delete. The main content area is titled "Create a version set for XML Loader" and contains the following fields and buttons:

- Application:** A drop-down menu with "xmltest" selected.
- View Name:** A text field containing "XMLLoader".
- Schema File:** A text field containing "test.xsd" and a "Browse..." button to its right.
- Attribute File:** A text field containing "edx.xsd" and a "Browse..." button to its right.
- Below the file fields are three buttons: "Submit", "Clear", and "Help".

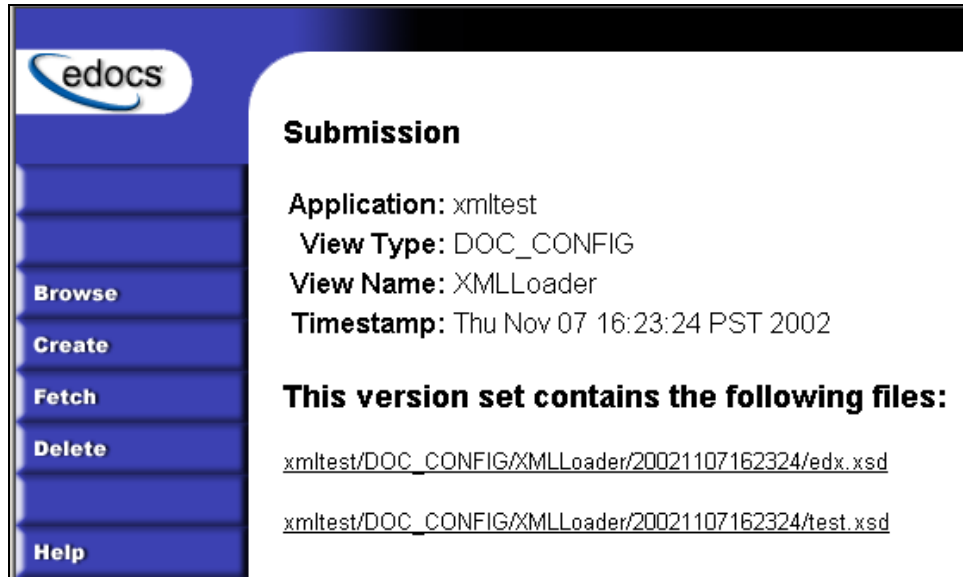
9. Select the application name from the drop-down list.
10. Browse and select the XML Schema file (for example, **test.xsd**) and Attributes file (**edx.xsd**) for the XML Loader job.



Caution

Make sure you enter the schema XSD file in the Schema field, and the attribute XSD file in the Attribute field. If you reverse these files, the job will run, but will not display correctly.

11. Click **Submit**. Publisher displays the Submission screen with details about the schema and attribute files:



12. Close the Publisher window.
13. At the **Create New Job** screen in Command Center, click **Configure Job and Continue**. eaDirect displays the job configuration screen:

Application: xml Job: xmlloader

From this screen, all parameters of the selected job can be modified. To edit parameters, change the entries in the desired fields and click the Submit Changes and Schedule button. To Reset the fields or for Help click the appropriate button at the top of the screen.

Submit Changes and Schedule Refresh Reset Help

Task 1: Scanner

Input File Path:

Input File Name:

Output File Path:

Task 2: XML Loader

Schema File Path:

Attribute File Path:

Task 3: XIRLoader

Skip Rows:

Split Size:

Optional Field Count:

Load Method:

Task 4: AutoIndexVolAccept

Action on Index Volume:

14. Specify the configuration parameters for each of the tasks that run as part of the XML Loader job. Carefully read the descriptions of each task and field (in the section below) to choose the appropriate values for your application.
15. When finished entering configuration parameters, click **Submit Changes and Schedule**. eaDirect asks "OK to submit this configuration?"
16. Click **OK**. eaDirect submits the job configuration parameters and displays the Schedule screen. You can specify the XML Loader job schedule later.
17. Click **Main Console**.

Task 1: Scanner

The Scanner task scans the input directory for new data input files. When it finds a new data file, it moves the file to the output directory and renames it, adding a timestamp (YYYYMMDDHHMMSS_filename.ext). When the Scanner finds multiple files, it processes them one at a time.



Tip

For the Input File Name in an XML Loader job, edocs recommends entering *.xml to search for all XML files in the Input directory.

For more information, see the *eaDirect Production Guide*.

Task 2: XMLLoader

The XMLLoader task uses the data file (from the data directory) and the published XMLLoader XML schema and attribute files (from the job configuration), and annotates the XML data fields to be indexed in a binary file in the edocs XML repository. It also creates an XML Intermediate Representation (.XIR) file.

Field	What to enter/select
Schema File Path	(Not an editable field.) The directory path and name of the XML Schema file. This is the XML Schema you specify when publishing the XMLLoader version set.
Attribute File Path	(Not an editable field.) The directory path and name of the XML attributes file (edx.xsd). This is the Attribute File you specify when publishing the XMLLoader version set.

Task 3: XIR Loader

The **XIRLoader** task uses the database loader to load data from the .XIR file to a database table using the script information in the .CMD file. It creates a row in the index table for each primary key. XIRLoader also creates the .LOG, .CTL, and .CMD files.

XIRLoader Task Input:	XIRLoader Task Output:
<ul style="list-style-type: none"> .XIR file Settings from the job configuration 	<ul style="list-style-type: none"> Rows added to index tables .LOG, .CTL, and .CMD files

The XIRLoader task takes the same user inputs as the IXLoader task of the Indexer job. For more information, see the *eaDirect Production Guide*.

Task 4: AutoIndexVolAccept

The AutoIndexVolAccept task determines whether the system can make the indexed data available for immediate user access or whether it must wait for you to approve the data. This task is primarily intended for eaDirect applications using a customized verification process.

For more information, see the *eaDirect Production Guide*.

Creating and Configuring an XML Email Notification Job

You create an XML Email Notification job to create and send an email notification message to enrolled users whose data is input as XML.

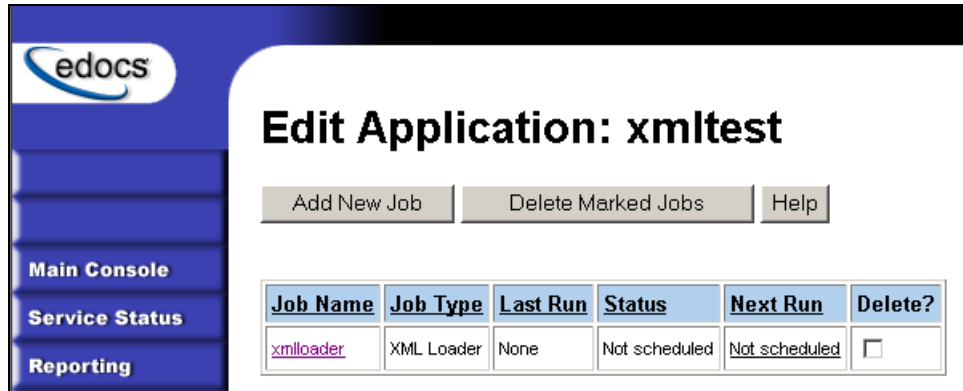
Creating and configuring an XML Email Notification job requires you to:

- Publish the XSL file created expressly for your XML Email Notification job.
- Specify configuration settings for the two production tasks that run sequentially as part of the job: **IVNScanner** and **XMLMailNotification**

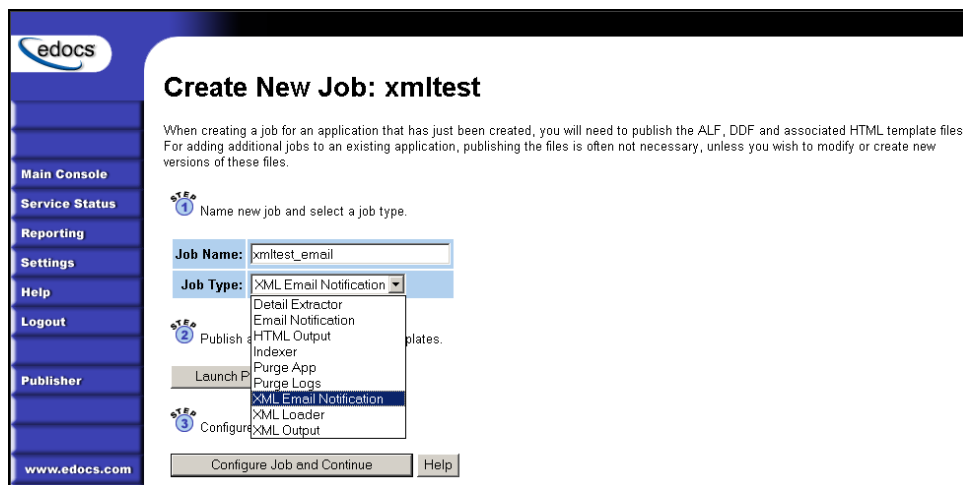
Review all the task and field configuration settings for this job to determine which options to use in your application.

To create and configure an XML Email Notification job:

1. On the Main Console, click the application name in the table. The **Edit Application** screen appears.



2. Click Add New Job. eaDirect displays the **Create New Job** screen:



3. Enter a meaningful name for the job. The job name must start with an alpha character. The rest of the characters can be alphanumeric and can contain underscores, but no spaces.
4. Select the **XML Email Notification** job type from the drop-down menu.
5. Click **Launch Publisher**.

Tip

XML Email Notification has its own XS View Name, XENotification. This name is not modifiable. To browse for an existing XML Email Notification job, open the Publisher's Browse window and use the drop-down menus to select **View Type>XS** and **View Name>XENotification**.

6. Click **Create**. Publisher displays the **Select a Version Set Type** screen:

The screenshot shows the 'Select a Version Set Type' screen. On the left is a blue navigation sidebar with buttons for 'Browse', 'Create', 'Fetch', 'Delete', and 'Help', and the URL 'www.edocs.com'. The main content area is titled 'Select a Version Set Type' and is divided into two sections:

Dynamic Web Views

Job Type	Number of Auxiliary Files
HTML	0 1 2 3 more
XS	0 1 2 3 more
CSV	0
XML	0
CHART	0
XSLT	0
XMLQuery	0

Batch Jobs

Job Type	Number of Auxiliary Files
Detail Extractor	0
Email Notification	0 1 2 3 more
HTML Output	0 1 2 3 more
Indexer	0
XML Email Notification	0
XML Loader	0
XML Output	0

7. Under **Batch Jobs**, next to **XML Email Notification**, click 0 (Number of Auxiliary files). Publisher displays the **Create a Version Set for XML Email Notification** screen:

The screenshot shows the edocs interface for creating a version set. On the left is a blue sidebar with the edocs logo and buttons for 'Browse', 'Create', and 'Fetch'. The main content area has a title 'Create a version set for XML Email Notification'. Below the title are three fields: 'Application:' with a dropdown menu showing 'xmltest', 'View Name:' with the text 'XENotification', and 'XSL File:' with a text input containing 'tiveXML\samples\testApp\test.xml' and a 'Browse...' button. At the bottom of the form are three buttons: 'Submit', 'Clear', and 'Help'.

8. Select the application name from the drop-down list. Browse to select the XSL file for this version set.
9. Click Submit. Publisher displays the Submission screen showing the version set you just published:

The screenshot shows the edocs 'Submission' screen. The left sidebar is the same as in the previous screenshot, but it now includes a 'Delete' button. The main content area has a title 'Submission'. Below the title are four lines of information: 'Application: xmltest', 'View Type: DOC_CONFIG', 'View Name: XENotification', and 'Timestamp: Thu Nov 07 16:55:00 PST 2002'. Below this information is a bold heading 'This version set contains the following files:' followed by a single file path: [xmltest/DOC_CONFIG/XENotification/20021107165500/test.xml](#).

10. Close the Publisher window.
11. At the **Create New Job** screen in Command Center, click **Configure Job and Continue**. eaDirect displays the XML Email Notification job configuration screen.

The screenshot shows the 'Application: xmltest Job: xmltest_email' configuration page. On the left is a blue navigation sidebar with the 'edocs' logo and menu items: Main Console, Service Status, Reporting, Settings, Help, Logout, and Publisher. The main content area has a title bar and a description: 'From this screen, all parameters of the selected job can be modified. To edit parameters, change the entries in the desired fields and click the Submit Changes and Schedule button. To Reset the fields or for Help click the appropriate button at the top of the screen.' Below the description are buttons for 'Submit Changes and Schedule', 'Refresh', 'Reset', and 'Help'. The configuration is divided into two tasks:

- Task 1: IVNScanner**
 - Index Volume Status: Accepted (dropdown)
 - Scan Starting From (Number of Days): 7 (text input)
- Task 2: XMLMailNotificationTask**
 - Base URL: http://Dino/eaSample/User?app=UserMain&jsp=/user/jsp/Detail.jsp (text input)
 - SMTP Hosts: SMTP host names (comma separated) (text input)
 - Return Address: Return email address (text input)
 - Subject Text: Email subject text (text input)
 - Administrator's Addresses: Administrator email addresses (comma separated) (text input)
 - Administrator Subject Text: Email Notification Error (text input)
 - Max Number of Retries: 3 (text input)
 - Retry Period (min): 60 (text input)
 - Enroll Model: JNDI name of the Account Resolver (text input)
 - Access Type: (text input)
 - Auditor Model: JNDI name of Mail Auditor (text input)

12. Specify the configuration parameters for each XML Email Notification task. Carefully read the descriptions of each task and field to choose the values appropriate for your application and job.
13. When finished entering configuration parameters, click Submit Changes and Schedule. eaDirect submits the job configuration parameters and displays the Schedule screen. You can schedule the XML Email Notification job later.
14. Click Main Console.

Task 1: IVNScanner

The IVNScanner task determines whether index data has been verified before creating and sending email to enrolled customers. This task is primarily intended for applications with a customized verification/audit application.

For more information, see the *eaDirect Production Guide*.

Task 2: XMLMailNotification

The XMLMailNotification task builds and sends email notifications *from XML data*. It uses information from the job configuration settings, the most recently published XML email version set, schema references, the data file, and the email addresses to generate email.

For information on configuring email notification, see the settings for the Email Notification job in the *eaDirect Production Guide*.

EmailNotification tasks have a new setting, **Auditor Model**. In this field, you specify the JNDI name of the Mail Auditor your system uses. Contact your system administrator to determine whether custom mail auditing has been implemented and if so, to obtain the appropriate JNDI name of your mail auditing client.

For more information on customizing your solution to audit e-mail notification, contact your edocs Professional Services representative.

Index

A

Application, 39, 54, 55,
71, 79
attributes, 14, 15, 47, 69,
70, 71, 75, 78
AutoIndexVolAccept, 72,
79

C

CDA, 35, 37, 38, 39
CDA client, 37, 38
Checking .log files for
errors, 11
Command Center, 17,
69, 76, 82
Configuring, 72, 79
Configuring tasks
IXLoader, 78
Customer Self Service, 7

D

Data Definition File, 17,
40, 69, 70
DefTool, 17
Development Tools,
eaSuite, 10
documentation, 10

E

eaAssist, 8
eaDirect, 1, 9, 10, 13, 15,
36, 37, 40, 55, 69, 70,
71, 72, 76, 77, 78, 79,
80, 82, 83, 84
eaMarket, 9
eaPay, 9
eaPost, 9
eaSample, 53
eaSuite, 7, 35
edx_common.jar, 37
Email Notification, 69,
70, 79, 80, 81, 82, 83,
84
extraction, 69, 71

F

Failed jobs, 59, 64

G

Groups, 17

H

Help, 11
documentation, 10
technical support, 11
HTML template, 69, 70

Index

- HTML templates, 69, 70
- HTML view, 70
- I**
 - Index Volume Status, 27
 - Indexer, 37, 40, 69
 - indexing, 16
 - interface, 14, 35, 54
 - IVNScanner**, 79, 83
 - IVNScanner task, 83
 - IXLoader task, 78
- J**
 - job configuration, 76, 77, 78, 82, 83, 84
 - job schedule, 77
 - Jobs, 74, 81
 - failed, 59, 64
 - status, 58
 - Type, 58
- L**
 - Last Run, 58
 - link, 50, 51, 52, 53
 - Logging into the Command Center, 42, 46
 - Login, 37
 - logs, 40, 42
- M**
 - MailNotification, 79, 84
 - Main Console, 72, 77, 79, 83
 - META-INF, 37
- N**
 - Next Run, 58
 - No operation, 59
- P**
 - Page style, 17
 - Password, 50
 - Production, 10, 78, 79, 83, 84
- R**
 - Recurring, 59
 - Reprocessing, 59
 - Run Time, 58
- S**
 - Scanner, 72, 77
 - SDK, 10
 - See Application Logic File, 40, 69, 70
 - Status, 58
- T**
 - Tables, 3, 17
 - Type, 73, 81
- U**
 - User, 37, 49, 50, 53
- V**
 - View, 81
 - Viewing, 53
- X**
 - XML, 15, 69, 70, 71, 72, 73, 74, 75, 77, 78, 79, 80, 81, 82, 83
 - XSD, 15, 75