



eaDirect™ Production Guide

eaDirect is a member of the eaSuite™ product line

V3.0
Document ID: DIGN-05-3.0-01
Data Published: 5.29.02

© 1997–2002 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, eaProcess, eaService, 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 Online Account Management and the eaSuite™7
 - About This Guide9
 - Related Documentation 10
 - Contacting edocs Technical Support 11

- 1 Overview of the Application Setup Process 13**
 - Before Getting Started 13
 - The Application Setup Process 14
 - What Jobs Do I Need to Create? 15
 - Why Do I Need an Indexer Job? 16
 - Other Production Jobs 17
 - EmailNotification Job..... 17
 - Purge App Job17
 - Purge Logs Job 17
 - HTML Output and XML Output Jobs..... 18
 - Detail Extractor Job..... 18
 - What is a View? 19
 - Dynamic Web views: HTML, CSV, XML, Chart, XSLT, and XML Query..... 19
 - Email Notification views 24
 - HTML Output views..... 25
 - Detail Extractor views..... 25
 - What Files Do I Need to Publish? 26
 - Files you publish with job configurations..... 26
 - Publishing your application's dynamic Web views (version sets)..... 27

- 2 Setting Up a New Application and Jobs 29**
 - Logging Into the eaDirect Command Center 29
 - Creating a New Application..... 31

Creating and Configuring an Indexer Job.....	32
Task 1: Scanner	38
Task 2: Indexer	39
Task 3: IXLoader.....	41
Task 4: AutoIndexVolAccept.....	46
Creating and Configuring an EmailNotification Job	47
Task 1: IVNScanner.....	53
Task 2: MailNotification	55
Creating and Configuring a Purge App Job.....	57
Task 1: PurgeIndexData	59
Task 2: PurgeEmailData	60
Task 3: PurgeActivityData.....	60
Creating and Configuring a Purge Logs Job	60
PurgeLogs Task.....	62
Creating and Configuring an HTML Output Job	62
Task 1: Scanner	71
Task 2: Indexer	71
Task 3: StaticHtmlFormatter	71
Creating and Configuring an XML Output Job.....	72
Task 1: Scanner	79
Task 2: Indexer	80
Task 3: XMLFormatter	80
Creating and Configuring a Detail Extractor Job	81
Task 1: IVNScanner.....	86
Task 2: StatementsToIR	88
Task 3: DXLoader	89
Publishing Your Application's Dynamic Web Views	90
3 Managing the Live Production Process.....	95
General Production Monitoring Activities	95
Scheduling Jobs	97
Monitoring Production Jobs	101
Viewing job status	103
Viewing and verifying task status.....	104
Canceling and retrying failed jobs.....	106
Viewing Message Logs.....	107

Viewing Job Output.....	110
Changing a Job Configuration	111
Deleting a Job.....	112
Monitoring System Services	113
4 Publishing and Using Version Sets.....	115
When to Publish New Version Sets	115
Publishing a New Version Set.....	116
How eaDirect Determines which Version Set to Use for Live Retrieval	135
Browsing for Version Sets.....	136
Maintaining Old Version Sets for Historical Statements	137
Viewing the Contents of a Design File	138
Deleting a Version Set	140
5 Reviewing System Activity.....	143
Viewing Production Statistics	143
User Statistics	143
Bill Statistics	147
Database Administration.....	151
Purging Historical Application Data; Purge App Job.....	151
Purging Historical Log Data; Purge Logs Job.....	152
Other System Administration Activities	152
Changing the eaDirect Administrator's Password.....	152
Deleting an Old Application.....	154
Appendix A: Troubleshooting	157
Version Sets	157
Production Jobs.....	158
System	159
Appendix B: Error Messages.....	161
eaDirect Error Messages	161
Appendix C: Glossary	185
eaDirect Terms and Acronyms.....	185
Online Glossaries and Databases.....	204
Index	205

Preface

About Online Account Management and the eaSuite™

Online Account Management solutions provide organizations and their business and consumer customers with the capability to service every aspect of their account relationship over the Internet. The account encapsulates all pertinent customer information, along with the activity and charges associated with the exchange of goods or services between a customer and an organization, and as such serves as the foundation for the relationship.

Whether an organization provides physical goods or services, the account relationship is one-time or recurring; or the customer is a large business with a global presence or is an individual consumer, the needs remain the same. Today, both organizations and their customers are demanding the ability to efficiently and conveniently manage, analyze accounts and have visibility into account initiations and modifications, detailed account history, orders, disputes and payments.

eaSuite – the Proven Platform for Online Account Management

edocs has developed the most comprehensive and sophisticated software platform and services for Online Account Management. Organizations that deploy edocs Online Account Management solutions increase productivity and improve profitability by:

- Reducing support and processing costs
- Improving operational effectiveness
- Increasing customer satisfaction

- Enhancing marketing effectiveness

The eaSuite has been deployed at organizations worldwide in some of the most complex business and rigorous technical environments, and has proven to be the industry's most scalable, standards-based platform with a total cost of ownership lower than any competing packaged or custom developed solution.



eaDirect™ is the platform product of the eaSuite that creates the core infrastructure of enterprise Online Account Management solutions. Organizations use eaDirect to access customer account data and develop ways of presenting the data for viewing, analysis and modification online. By providing business and consumer customers with online account information and self-service capabilities, eaDirect enables organizations to reduce support and processing costs, improve operational effectiveness, and increase customer satisfaction.

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 Automatic Clearing House (ACH) and credit card payments, and payments via various payment processing service providers.

eaService™ reduces call center management costs and increases customer satisfaction by providing critical account data access to customer service representatives (CSRs). With eaService, a contact center is able to provide better service by more efficiently resolving customer account inquiries at the point of customer contact. Through its intuitive browser interface, eaService allows CSRs to take advantage of what are generally thought of as customer-facing online capabilities of an Online Account Management system, to draw out account information.

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 Online Account Management experience. eaMarket leverages the transactional account data that is the foundation of an Online Account Management system to present personalized relevant marketing and customer service messages to customers.

eaPost® is the account content distribution system that provides an organization's customers with the convenience of All Portals Addressable™ access to account information. eaPost handles all the complexities of securely distributing summary account information to any endpoint while also bringing customers back the organization's Website to manage and control their online experience.

About This Guide

This guide is intended for system administrators or other IT professionals responsible for setting up and running a live eaDirect application in a J2EE environment. It describes the general process and specific procedures required to:

- Set up a new eaDirect application and the associated jobs using the eaDirect Command Center.
- Produce regular online statements electronically and manage the ongoing live production process.

This guide assumes you have:

- Installed eaDirect in your J2EE environment.
- Designed and developed the necessary application files (using DefTool and Composer in a Windows NT or 2000 environment).

This guide does *not* describe general UNIX or Windows system administration. See the appropriate UNIX or Windows user documentation.

Related Documentation

The following online Help is available in the eaDirect Command Center:

Online Help	How to Access
Comprehensive Command Center Help	Click Help on the Command Center menu.
Screen-level Command Center Help	Click the Help button on a screen for details about that particular screen. Click Help Contents there to access the complete Production Help.
A PDF version of this guide	A PDF version of this document is located on your product CD.

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 Production Guide	How to set up and run a live eaDirect application in a J2EE environment.
eaDirect User's Guide	What you need to know to design an application to present your account information online; how to use DefTool and Composer to create the application.

Add-on eaSuite products eaPay, eaPost, eaMarket, and eaXchange provide their own documentation.

Contacting edocs Technical Support

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 8:00 am to 8:00 pm 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

Whether you report a problem online, by email or by telephone, 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.

Preface

- Do you have an idea on how to improve eaDirect? Don't hesitate to e-mail your idea to support@edocs.com.

Overview of the Application Setup Process



Before Getting Started

Your eaDirect application was created by a project team in your organization. An eaDirect application consists of various design files used in a live production environment with your eaDirect software to enable Web users to view statements online.

During the Mastering process, your project team evaluated your organization's online presentation needs along with your data input files. They created an application that would deliver the specific data you want customers to see presented exactly the way you want them to see it. They used the eaDirect GUI design tools, DefTool and Composer, to create these files.

You must use the eaDirect Command Center to set up and configure your application to prepare it for implementation in a live production environment.

Before setting up your application in the Command Center, you must:

- Become familiar with the eaDirect design files created for your application and the particular account information they are intended to provide the user. Creating and configuring the correct production jobs with the appropriate configuration settings requires a thorough understanding of your application. You must understand exactly what each .DDF, .ALF, and template file in your application has been designed to present. This chapter describes how each type of design file is used during production to create a particular view of statement data online.

- Work with your project team to establish what jobs you need to define and which job configuration settings you need for your application to work as intended by your design team. Review the job configuration options described in Chapter 3.

The Application Setup Process

The process of setting up a new application in the eaDirect Command Center requires three general steps. If you have multiple applications, it is best to set up one application at a time.

To set up a new eaDirect application, you must:

1. **Create a new application.** This short step requires you to define, or name, the application in the eaDirect Command Center.
2. **Create and configure the associated production jobs.** To implement your application in a live environment, you must configure various production jobs. This chapter describes the types of jobs you must create to make your statement data available for online presentment.

For each job, you must choose the configuration options that will enable your eaDirect application to function as intended; see Chapter 3 for a description of all configuration options. For some jobs, you must also publish associated version sets when you configure the job.

3. **Publish dynamic Web view files (version sets).** This chapter describes what views are and how eaDirect uses dynamic Web views to extract and present statements online. Dynamic Web views are discussed further in “Dynamic Web views: HTML, CSV, XML, Chart, XSLT, and XML Query” on Page 19.

Once you have defined your eaDirect application, created and configured jobs, and published the dynamic Web views, you can proceed to Chapter 4 to set up a schedule for each job and begin live production. Note that eaDirect does not automatically schedule jobs to run; you must manually specify job schedules for production; see Chapter 4.

What Jobs Do I Need to Create?

Each eaDirect application requires certain batch jobs run on a recurring basis to make statement data available for online viewing.

The specific number and type of production (batch) jobs you need to create and configure depends on the number and type of views the project team has developed in your application.

You must create and configure the following jobs for an application:

- **An Indexer job, if your application uses live retrieval** to index the data file in preparation for live statement viewing on the web. Applications using live retrieval must have an Indexer job. An Indexer job is also necessary to enable you to generate email output.
- **An EmailNotification job, if your application has an email notification view** to send an email message to enrolled users informing them that a statement awaits them online.
- **A Purge App job** to periodically delete old data references from the index, email, report activity, detail, annotations, and dispute tables.
- **A Purge Logs job** to periodically delete old data from the logs table in the database.

Each job type is described in more detail in this chapter.



Caution

HTML Output or XML Output jobs are necessary only if you plan to provide users access to static output; not live retrieval.

A Detail Extractor job is necessary only if you plan to upload recurring table data from the input file to a database table for merging using additional application functionality.

Why Do I Need an Indexer Job?

Each application that uses live statement retrieval requires an Indexer job to prepare the input data for dynamic viewing.

An Indexer job:

- Enables Web users to view statements using live retrieval.
- Indexes data fields such as customer name, amount due, and due date for display on a page listing historical statements available for viewing (sometimes called a history list).
- Enables you to generate email notifications (which you create in a separate job).
- Enables you to extract recurring table data from the input file and load it into a database table (which you load using a separate job and additional application support to use the data).

By scheduling the Indexer job to run automatically on a regular interval coinciding with the generation of your statement data file, you enable your latest statements to be routinely available for on-demand Web presentation.

The Indexer job extracts important data about your data file, called metadata, and places it in the database. When a user clicks a link to their statement, the browser uses index data in the database, the data input file, and the dynamic view files to present the user's statement.

The Indexer job also extracts data about any fields you choose to index, such as customer name, amount due, and due date, to the database. Indexing these fields makes them available for display in a historical list of bills.

The Indexer job consists of separate tasks that run sequentially; see “Creating and Configuring an Indexer Job” on Page 32 for details.

Other Production Jobs

EmailNotification Job

You must set up and configure an EmailNotification job if your application is designed to generate an email notification message. An email message informs a user that an online statement is available, and can include a direct link to dynamically view the statement.

In a live environment, you run an EmailNotification job after the Indexer job because it consists of individual tasks that depend on the successful completion of the Indexer job.

For details on how to set up an EmailNotification job, see “Creating and Configuring an EmailNotification Job” on Page 47.

Purge App Job

Purge App is a system maintenance job that removes index, email, reporting, detail, annotations, and dispute data from the database. Configure this job and run it periodically to free up space on your database server and to limit user access to historical statements.

For more information about the Purge App job and other system maintenance activities, see “Database Administration” on Page 151.

For details on setting up a Purge App job, see “Creating and Configuring a Purge App Job” on Page 57.

Purge Logs Job

Purge Logs is a system maintenance job that removes historical information from the log table in the system database (for all applications). Configure this job and run it periodically to free up space on your database server.

For more information about the Purge Logs job and other system maintenance activities, see “Database Administration” on Page 151.

For details on setting up a Purge Logs job, see “Creating and Configuring a Purge Logs Job” on Page 60.

HTML Output and XML Output Jobs

HTML Output and XML Output jobs create an HTML or XML output file with extracted data for each primary key. If you plan to make statements available for live retrieval on the Web, you do not need to create an HTML Output or XML Output job.

Configure an HTML Output job only if you plan to limit access to your database and let users view a static HTML output file only. Static HTML output files may be necessary if you partner with thin or thick consolidators for statement presentment. The choice to use static output would be the result of specific performance and security issues.

Configure an XML Output job if you plan to generate a static XML output file for loading into another database.

For details on setting up an HTML Output job, see “Creating and Configuring an HTML Output Job” on Page 62.

For details on setting up an XML Output job, see “Creating and Configuring an XML Output Job” on Page 72.

Detail Extractor Job

Set up and configure a Detail Extractor job only if you intend to upload data from your input file to a database table. You can use the uploaded data in any way, merging it with online statements, performing data mining, etc. You must create any applications needed to extract and use the data, however.

A Detail Extractor job is not required for presenting basic print statements on the Web using eaDirect.

In a live environment, you run the Detail Extractor job after the Indexer job; it only processes a data input file that the Indexer job has successfully indexed.

For details on setting up a Detail Extractor job, see “Creating and Configuring a Detail Extractor Job” on Page 81.

What is a View?

A **view** is a set of design files that results in a particular presentation of statement data.

A view can enable a user to dynamically display formatted statements live on the Web, generate email notifying users that an online statement is available, or to present other account data in various formats.

Dynamic Web views: HTML, CSV, XML, Chart, XSLT, and XML Query

A dynamic Web view is a set of design files that dynamically present a particular view of statement data to a user online. The design files identify which data to extract and how to display the data to the user.

An eaDirect application can have one or more views, customized for an organization's online presentment needs. Multiple views can present different levels of statement information such as a summary page and statement detail pages.

When an enrolled user clicks a link to view their statement online, eaDirect uses the view files along with the application's data input file and index data from the database (generated by the Indexer job) to dynamically present the statement on the Web.

A typical **dynamic HTML Web view** consists of a pair of DDF and ALF files, and one or more associated HTML templates:

- **DDF** – A DDF is a Data Definition File, which contains the rules for finding and extracting data from your application's input data source. This DDF file is used during live statement retrieval. Your project team creates DDF files using eaDirect's DefTool.

- **ALF** – An ALF is an Application Logic File, which contains the rules for presenting the data extracted from the data input source in a template on the Web, in email, etc. The ALF can also contain business logic (conditional statements that consider current statement data) to display alternate messages or advertisements for marketing or business purposes. Your project team creates ALF files using eaDirect's Composer tool.
- **HTML Templates** – Customized HTML templates format and present the extracted data for viewing in a browser. A view can have multiple templates associated with it. Your project team creates HTML files (using any variety of methods), which they then manipulate appropriately for online presentation using eaDirect's Composer tool.

In addition to HTML-formatted views, eaDirect application views can dynamically present data in one of these formats:

- **CSV** – (Comma Separated Values) To display data in spreadsheet format. A dynamic CSV view consists of a DDF file, created expressly for the view, and a TOK file.
- **XML** – To display data in XML format. A Dynamic XML view consists of a DDF file created expressly for the view.
- **Chart** – To present data in one of several chart formats. A dynamic chart view consists of a Properties file.
- **XSLT** – To convert statement data into XML, which enables you to present the data in a variety of formats, including CSV, HTML, WML, VXML, and QIF. An XSLT view consists of an XSLT file and an application DDF.
- **XML Query** – To access a view of dispute, annotation, and detail data from the database. The Detail Extractor job extracts this information from the data input file and loads it into the database. An XML view consists of a DDF file.

When setting up a new eaDirect application, you make the dynamic Web view files available to an application by "publishing" them using the eaDirect Publisher tool, described in the next section.

When publishing a dynamic view, you give it a name, such as AccountSummary, CallDetail, etc. You can use the same view names in multiple applications. Each would represent entirely different views, using different design files in each application. (You could also have multiple versions of the same application, for example, NatlWireless version 1.0 and NatlWireless version 2.0, which would have views with the same names.)

Sample eaDirect application

Your eaDirect installation includes the sample application called “NatlWireless,” designed to simulate a telecommunication provider’s application. NatlWireless uses a sample ASCII data file called NatlWireless.txt.

National Wireless sample files are located in the
<root>:\EDCSbd\samples\NatlWireless directory.

Sample dynamic HTML views

NatlWireless provides two dynamic HTML views to simulate dynamic statements, called HtmlDetail and NW_LocSummary.

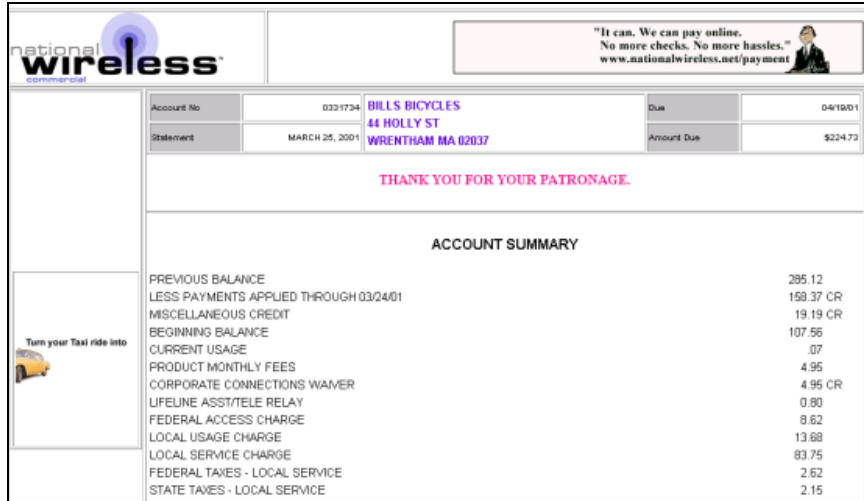
HtmlDetail - Summary statement view - The main dynamic HTML view consists of the following files:

- NatlWireless.DDF
- NatlWireless.ALF
- NatlWireless.DDF

Note that this view uses NatlWireless.DDF, which you would also publish with the application’s Indexer job.

This view presents the primary page of the NatlWireless application:

Overview of the Application Setup Process

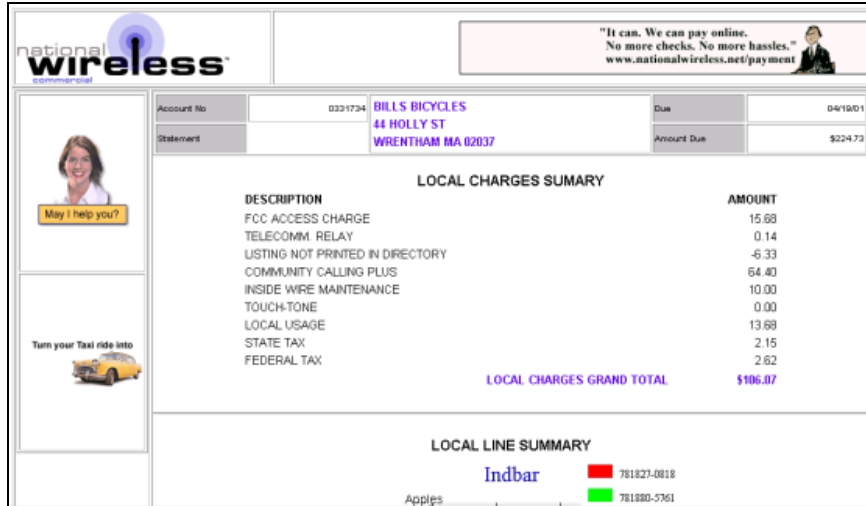


NW_LocSummary - Detail statement view – A second dynamic HTML view used with the NatlWireless statement consists of the following files:

- NW_LocSummary.DDF
- NW_LocSummary.ALF
- NW_LocSummary.DDF

Notice that this view uses a different DDF file from the previous view. The detail statement view uses the index data generated by the NatlWireless.DDF published for this application's Indexer job, but uses a custom DDF for data extraction.

The composed view presents the following page of local call detail for NatlWireless:



Sample dynamic CSV, XML, XSLT, and XMLQuery views

In addition to the HTML view files, NatlWireless provides the following sample dynamic views:

Job Type	View Name	NatlWireless Sample Files
CSV	User-provided	NatlWireless.DDF NatlWireless.TOK
XML	User-provided	NatlWireless.DDF
XSLT	SummaryInfo	NatlWireless.DDF XSLTDownload/summary_info_csv.XSL
XMLQuery	AnnotationQuery	XMLQuery\annot_sql.XML
	DetailQuery	XMLQuery\detail_sql.XML
	DisputeQuery	XMLQuery\dispute_sql.XML

For a complete listing of National Wireless sample files, see the SDK Module *Customizing and Deploying edocs Sample Web Applications*.

Email Notification views

eaDirect lets you communicate with users by creating and sending an email notification to enrolled users, typically to let them know that an online statement is available for viewing.

An email notification view consists of an ALF/DDF pair, plus one or more HTML files designed specifically to generate a particular email message.

To compose and send an email notification message, you use an EmailNotification job, which uses an associated email notification view along with the user's enrollment data, input data file, and index data from the database (generated by the Indexer job) to generate the email.

Example

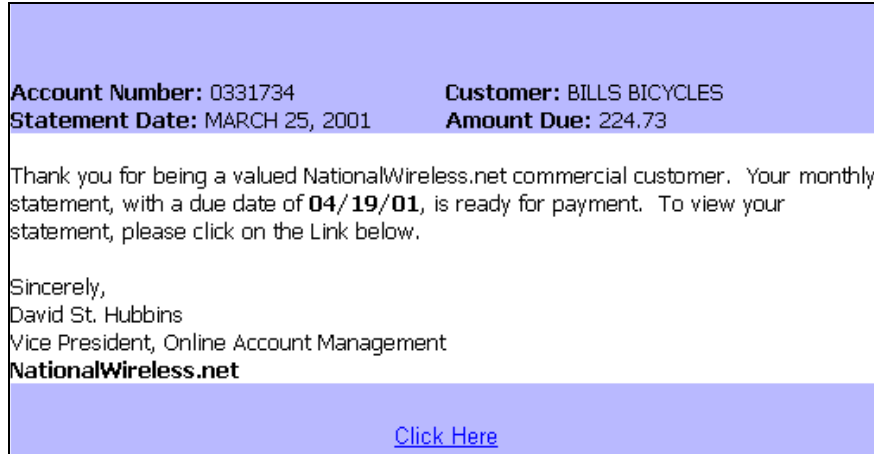
The NatlWireless sample application contains a sample email notification view, consisting of the following files, to generate a sample email for telecommunication customers:

- NatlWireless.DDF
- NW_Email.ALF
- NW_Email.HTM
- NW_EmailAlternate.HTM

Note that this view uses NatlWireless.DDF, the same DDF used by the dynamic HTML summary view and for the Indexer job.

It also contains an auxiliary HTML template called NWEmailAlternate.HTM. Conditional business logic in the ALF file determines which template eaDirect uses to generate the email notification.

A composed email notification using the NW_Email.HTM template looks like this:



HTML Output views

If you generate static HTML-formatted output using an HTML Output job, your application uses a view containing an ALF/DDF pair and one or more corresponding HTML templates.

Detail Extractor views

A Detail Extractor job extracts a single table or group of data and uploads it into a database table. The view files you publish to extract a single view of recurring table data includes a DDF file, a database table schema XML file, and a statement XSLT style sheet.

eaDirect provides the following sample files with the NatlWireless application to demonstrate publishing and using a view called “dtlextr” for the Detail Extractor job.

- NatlWireless.DDF
- DetailExtractor\summary_info.XML
- DetailExtractor\summary_info.XML

What Files Do I Need to Publish?

Setting up a new application requires you to publish application design files. "Publishing" identifies the design files an application uses and lets you move the files from the design environment to your application server.

You must publish

- Certain files along with each job configuration, at the time you configure the job
- Your application's dynamic Web view files, which you can do after setting up and configuring your application and jobs

You use eaDirect's Publisher tool to publish design files.

Files you publish with job configurations

When you create and configure eaDirect batch jobs, you publish the following files as part of (during) the job configurations:

Batch Job Type	Files you are instructed to publish as part of the job configuration:
Indexer	A DDF file for the Indexer task.
Email	The email notification view: a DDF, an ALF, and one or more HTML files
HTML Output	A DDF file for the Indexer task, and the HTML Output view files: a DDF, ALF, and one or more HTML templates
XML Output	A DDF file (for use by both the Indexer and XMLFormatter tasks in this job)
Detail Extractor	A DDF file, database table schema XML file, and an XSLT style sheet for use by the StatementsToIR task

Chapter 3 guides you through the process of creating and configuring jobs, and publishing the required design files.

Publishing your application's dynamic Web views (version sets)

After creating and configuring your application jobs, you publish the files that make up each dynamic Web view in your application.

"Publishing" a dynamic Web view identifies each file belonging to a particular view, and lets you name the view. It also lets you move the files from the design environment to your application server. When a user clicks a link to their statement ("running" a dynamic Web view), the link identifies the set of design files to use to extract and present the user's statement data.

Publishing adds a timestamp to the set of view files. A **version set** is a dated set of design files. Publishing a view is also called "**creating a version set**."

The following table shows the type of files that make up each type of dynamic Web view:

Dynamic Web view format	Files you publish in a version set for this view format:
HTML	DDF ALF One or more HTML files
CSV	DDF TOK
XML	DDF
Chart	A Properties file
XSLT	XSL DDF
XML Query	XML

See "Publishing your application's dynamic Web views (version sets)" on Page 27 for instructions.

Overview of the Application Setup Process

For information about publishing in a live production environment, see “Publishing and Using Version Sets” on Page 115.

Setting Up a New Application and Jobs

2

Logging Into the eaDirect Command Center

You use the Command Center to set up, configure, and manage your applications.

During live production, you use the Command Center to schedule and run production tasks, monitor system activity, and perform other system administration activities.

The Command Center is a secure application that requires you to log in with an administrator's ID and password. Both the ID and password are determined during eaDirect installation and configuration. If you forget the Command Center password, contact your system administrator or the person who installed eaDirect.

Always log out of the Command Center after completing a session. By logging out, you help maintain the security of the eaDirect production environment and minimize the chance an application or job can be accidentally corrupted or destroyed.

To change the administrator's password, see “Changing the eaDirect Administrator's Password” on Page 152.

To log into the Command Center:

1. Verify that the Web server and the database server are both running.
2. Launch your Internet browser.

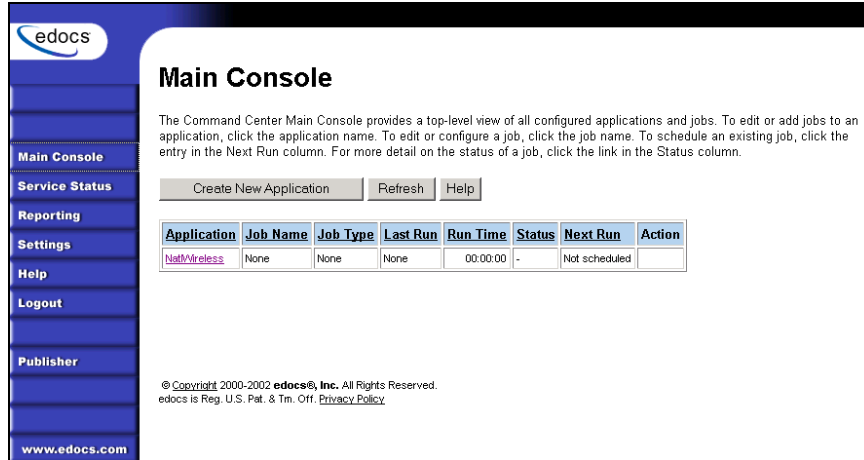
3. Enter the URL for the eaDirect Command Center servlet configured when eaDirect was installed, such as <http://dusky:7001/eaDirect>.



4. On the Login Administrator page, enter the administrator's ID and password. The default ID is **admin** and the password is **edocs**.

If you can't access the Login Administrator page or eaDirect does not recognize the ID and password, consult your system administrator or the person who installed eaDirect.

5. Click **submit**. eaDirect displays the Command Center Main Console:



To log out of the Command Center:

- Click **Logout** on the Main Console.

Creating a New Application

To create a new application:

1. Go to the eaDirect Command Center.

Click **Create New Application** from the Main Console. eaDirect displays the Create New Application screen:

The screenshot shows the 'edocs' logo in the top left corner. Below it is a vertical navigation menu with the following items: Main Console, Service Status, Reporting, Settings, Help, Logout, and Publisher. The main content area is titled 'Create New Application:' and contains a text input field labeled 'Application Name:'. Below the input field are two buttons: 'Create Application and Continue' and 'Help'. At the bottom of the page, there is a copyright notice: '© Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)'.

2. Enter the name of the application. The first character in the name must be an alpha (because database tables will also be given this name). The rest of the name can contain alphanumerics and underscores, but no spaces.
3. Click **Create Application and Continue**. eaDirect displays the Create New Job screen.
4. Proceed with the following instructions to create and configure an Indexer job for your application.

Creating and Configuring an Indexer Job

You must create and configure an Indexer job if you plan to use live retrieval with your application.

Creating and configuring an Indexer job requires you to:

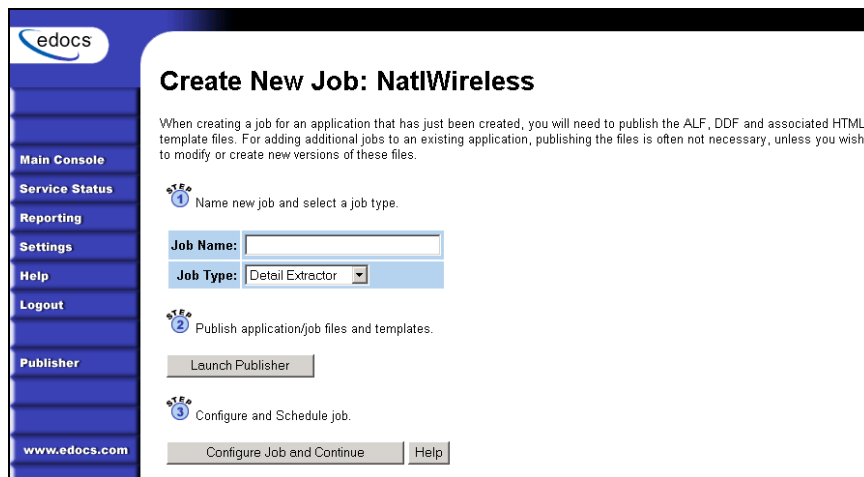
- Specify configuration settings for the four production tasks that run sequentially as part of the Indexer job: Scanner, Indexer, IXLoader, and AutoIndexVolAccept.
- Publish the DDF file created expressly for your Indexer job.

Review all the task and field configuration settings (in this section) to determine which options to use in your application.

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

To create and configure an Indexer job:

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:



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**.

5. Click **Create**. Publisher displays the following screen:

edocs

Select a Version Set Type

Dynamic Web Views

Job Type	Number of Auxiliary Files
HTML	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 Output	0

Browse

Create

Fetch

Delete

Help

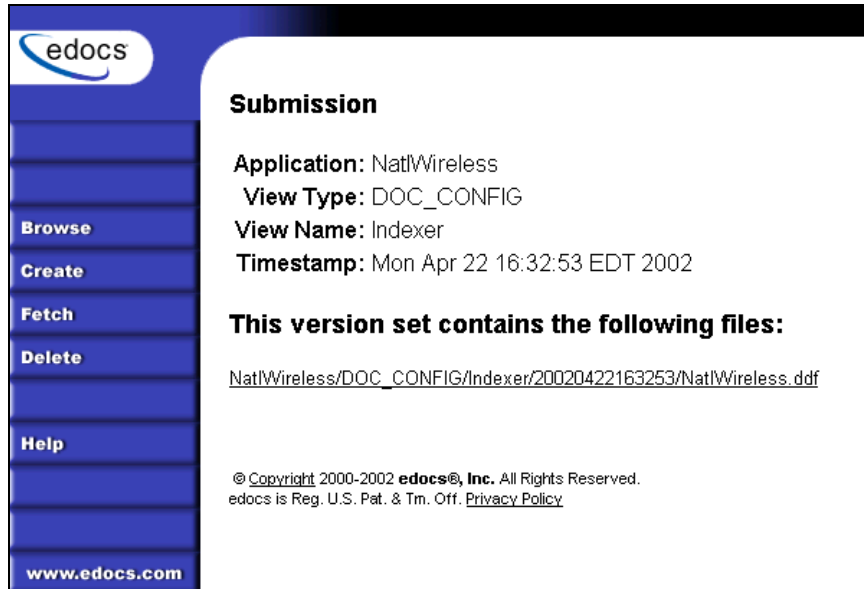
www.edocs.com

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

The screenshot shows the edocs web interface. On the left is a blue sidebar with the edocs logo at the top and a menu with buttons for Browse, Create, Fetch, Delete, and Help. The main content area is white and titled 'Create a version set for Indexer'. It contains the following fields and controls:

- Application:** A drop-down menu currently showing 'NatlWireless'.
- View Name:** A text input field containing 'Indexer'.
- DDF File:** A text input field followed by a 'Browse...' button.
- Below the DDF File field are three buttons: 'Submit', 'Clear', and 'Help'.
- At the bottom of the main area is a copyright notice: '© Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)'.
- At the bottom of the sidebar is the URL 'www.edocs.com'.

7. Select the application name from the drop-down list.
8. Browse and select the DDF file for the Indexer job.
9. Click **submit**. Publisher displays the Submission screen with details about the DDF file:



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



12. Specify the configuration parameters for each of the four tasks that run as part of the Indexer job. Carefully read the descriptions of each task and field (in the section below) to choose the appropriate values for your application.
13. When finished entering configuration parameters, click **Submit Changes and Schedule**. eaDirect asks “OK to submit this configuration?”
14. Click **OK**. eaDirect submits the job configuration parameters and displays the Schedule screen. You can specify the Indexer job schedule later.
15. 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). Scanner finds multiple files, it processes them one at a time.

Scanner Task Input:	Scanner Task Output:
<ul style="list-style-type: none"> Data source in input directory 	<ul style="list-style-type: none"> Data file moved to the data file path location Timestamp added to the data file name

Scanner Task Configuration (Indexer job)	
Field	What to enter
Input File Path	Specify the input file data directory where Scanner can find the application's data input file. For UNIX systems, you must place the input file on the same file system as the data file. This file path can be on a NFS Mount. Only use a symbolic link on the same file system. The default is EDX_HOME/Input/Application Name.
Input File Name	Specify the name of your application's data input file. You can use wildcards (*) in the file name.
Output File Path	Specify the name of the application data directory where you want Scanner to move the data input file. The default is EDX_HOME/Data/Application Name.

Task 2: Indexer

The Indexer task uses the data file (from the data directory), the published Indexer DDF file, and index field list (from the job configuration), and places information about the data fields to be indexed into an Intermediate Representation (.IR) file.

Indexer Task Input:	Indexer Task Output:
<ul style="list-style-type: none"> • Data source file in the output file path • Most recent DDF published in AppProfiles directory (not configurable) • The list of fields to index (configurable) 	<ul style="list-style-type: none"> • An .IR file containing the following metadata, in the data file path: • The primary key • The date/time the file is processed, from the application server clock • A data file ID • Byte offset information (position within the data file where customer's information is located) • The number of pages of a customer's statement • The selected index field list <p>The first row of the IR file contains the following header fields: <u>Z_Primary_Key</u>, <u>Z_DocDate</u>, <u>Z_Doc_ID</u>, plus the index field names.</p>

Indexer Task Configuration (Indexer job)	
Field	What to enter/select
DDF Path	(Not an editable field.) The directory path and name of the DDF file this Indexer job uses appear in this field. (This is the DDF you specify when publishing the Indexer version set.)
Doc Date	The date you're indexing the data input file (not editable).
Index Field List	<p>eaDirect displays all fields defined in the Indexer DDF.</p> <p>Select the fields you want to index, such as customer name, amount due, and due date for display on a page listing historical statements available for viewing (sometimes called a history list).</p> <p>You may need to select <i>additional</i> fields to index to:</p> <ul style="list-style-type: none"> • Display those values to the customer in a summary page • Display those values to the customer service representative (CSR) in a history list/summary view • Build additional CSR functionality to search using those values • Use those values to create an output file for paper-based statement validation (an alternative to eaDirect's online verification functionality) • Allow access to those values by other systems, such as system integrators building a comprehensive Web site <p>To select a field, highlight the field name and click the right arrow button. Use the scroll bar to view more fields. To unselect a field, highlight the field name and click the left arrow button.</p>



Caution

Be sure to select fields for indexing. If you don't select any fields, eaDirect indexes all fields in the DDF, causing all the fields to appear on the application summary page. Also, do not select fields followed by an asterisk (*); these are length 255k, and can cause job failure. Select these fields for Indexing only if you are knowingly using them with short data that will not cause job failure.

Task 3: IXLoader

The IXLoader task uses the database loader to load data from the .IR 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. IXLoader also creates the .LOG, .CTL, and .CMD files.

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

IXLoader Task Configuration (Indexer job)	
Field	What to enter/select
Skip Rows	<p>Enter the number of rows of data at the beginning of your .IR file that you do not want loaded into the database, if any. These rows could contain unwanted data extracted from the beginning of your input file.</p> <p>The default (0) loads all rows from the .IR file into the index database table (except the first row which always contains header fields).</p>
Split Size	<p>The Split Size parameter lets you specify whether to use a single index database table or multiple tables to store Indexer data.</p> <p>To store the data generated by Indexer in one database table, use the default (0) in the Split Size field.</p>
Split Size <i>(continued)</i>	<p>If you prefer to use multiple database tables to store Indexer data, you must carefully calculate the proper Split Size value (number of rows) appropriate for your application.</p> <p>The value you specify in the Split Size field is <i>not</i> the maximum, or actual, number of records you want per table. The value should be based on the typical number of documents in your IR file as well as the maximum number of rows you want in each table.</p> <p>See “How Split Size Works” (following this table) for a sample calculation.</p>

IXLoader Task Configuration (Indexer job)					
Field	What to enter/select				
Optional Field Count	<p>Enter the number of additional fields your application uses, if any. eaDirect adds this number of generically defined columns to the database table.</p> <p>This option is intended for applications with custom summary pages that use generic fields to store data such as a flag indicating whether a user has made a payment. It can also be used with customized verification processes. You can specify up to 16 optional fields.</p> <p>Use the default value (0), if you are not using extra fields.</p>				
Load Method	Choose one of the following database load methods:				
	<table border="1"> <tbody> <tr> <td>Direct load</td> <td>Stores data directly to the database. This option locks the index table, loads the information, and ends the call to the database. No sharing of tables is allowed during this process. A direct load is the fastest method.</td> </tr> <tr> <td>Conventional load</td> <td>Uses Insert statements, one row at a time. Performs multiple selects and inserts on the table at once, but does not lock up the database and lock out Web users. A conventional load is slower than a direct load.</td> </tr> </tbody> </table>	Direct load	Stores data directly to the database. This option locks the index table, loads the information, and ends the call to the database. No sharing of tables is allowed during this process. A direct load is the fastest method.	Conventional load	Uses Insert statements, one row at a time. Performs multiple selects and inserts on the table at once, but does not lock up the database and lock out Web users. A conventional load is slower than a direct load.
Direct load	Stores data directly to the database. This option locks the index table, loads the information, and ends the call to the database. No sharing of tables is allowed during this process. A direct load is the fastest method.				
Conventional load	Uses Insert statements, one row at a time. Performs multiple selects and inserts on the table at once, but does not lock up the database and lock out Web users. A conventional load is slower than a direct load.				

How Split Size Works

Specifying a positive Split Size value does not actually split an index table; it determines a point over which eaDirect creates a new table and begins adding new records there. When the number of rows in the database table plus the number of new rows coming from the IR table would exceed the Split Size, eaDirect creates a new table.

Indexer job checks the number of records in the current database table and adds to it the number of records to be loaded from the IR file, and

- If the total number of records is less than the split size, it proceeds to add all of the new records to the same table.
- If the total number of records is greater than the split size, Indexer creates a new table and adds the new records to it. The value you specify in the Split Size field should be a figure below the maximum number of rows you want in the table.

Example:

If you have 70,000 records in the IR file and you want to keep the maximum index table size to 140,000 records, you could specify the split size to be 200,000. Each month you run Indexer, it needs to add 70,000 rows to the database.

First month: Indexer finds the number of records in the database table is 0. It calculates $0 + 70,000 = 70,000$, which is less than the split size of 200,000, so Indexer adds the 70,000 rows to the new (empty) table.

Second month: Indexer checks the number of rows in the current table and finds 70,000. It calculates $70,000 + 70,000 = 140,000$, which is less than the split size of 200,000, so it adds the 70,000 rows to the table.

Third month: Indexer finds there are 140,000 records in the table. It calculates $140,000 + 70,000 = 210,000$, which is greater than the split size of 200,000, so it creates a new database table and adds the rows there.

Each month thereafter: Indexer creates a new table whenever adding the new records would cause the table to exceed 200,000, which is now every other month. Each database table grows to a maximum of 140,000 records.

The following table shows an example of how the split table process works:

Indexer Job Run	Number of rows Indexer finds in the current table	Create new Index database table? (Would adding the data from this run cause table size to exceed split size of 200,000?)	# Rows added to current table	# Rows added to a new table if created	Total # rows in current table after Indexer runs
Month 1	0 in NatlWireless_1	No 0+70,000 <200,000	70,000		70,000 in NatlWireless_1
Month 2	70,000 in NatlWireless_1	No 70,000+70,000 <200,000	70,000		140,000 in NatlWireless_1
Month 3	140,000 in NatlWireless_1	Yes 140,000+70,000 >200,000 Create NatlWireless_2	0	70,000	70,000 in NatlWireless_2
Month 4	70,000 in NatlWireless_2	No 70,000+70,000 <200,000	70,000		140,000 in NatlWireless_2
Month 5	140,000 in NatlWireless_2	Yes 140,000+70,000 >200,000 Create NatlWireless_3	0	70,000	70,000 in NatlWireless_3
Month 6	70,000 in NatlWireless_3	No 70,000+70,000 <200,000	70,000		140,000 in NatlWireless_3
Month 7	140,000 in NatlWireless_3	Yes 140,000+70,000 >200,000 Create NatlWireless_4	0	70,000	70,000 in NatlWireless_4

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.

The verification process lets someone inside your organization see the data, but not customers. Using the internal verification process, you can mark the volume as approved, making it available for users. A volume is a data input file that has been successfully processed by the Indexer job and referenced in the volumes table.

AutoIndexVolAccept Task Input:	AutoIndexVolAccept Task Output:
<ul style="list-style-type: none">• Settings from the Indexer job configuration	<ul style="list-style-type: none">• If AutoAccept, fills in the Date Accepted field and the volumes table.• If Intercept to Verify, does not fill in a date field and it is up to your internal verification process to assess the date. The user cannot view their data until this process is complete

AutoIndexVolAccept Task Configuration (Indexer job)	
Field	What to enter/select
Action on Index Volume	Choose one of the following options:
	<p>Auto Accept (Default) Choose this option if you are not using a custom verification/audit application. Auto Accept automatically makes the index references immediately available (viewable) to the user.</p> <p>Auto Accept fills in the Date Accepted field in the volumes table. A date in this field means the volume has been approved.</p>
	<p>Intercept to Verify Choose this option if you use an internal verification process, which lets you perform quality control on the data then accept or reject it.</p> <p>You can accept or reject the entire set of data, but not individual documents. Users cannot access this data until the verification process is complete.</p>

Creating and Configuring an EmailNotification Job

You create an EmailNotification job to create and send an email notification message to enrolled users.

Creating and configuring an EmailNotification job requires you to:

- Specify configuration settings for the two production tasks that run sequentially as part of the job: IVNScanner and MailNotification
- Publish the email view files (DDF, ALF, and HTML templates) created for the intended email message

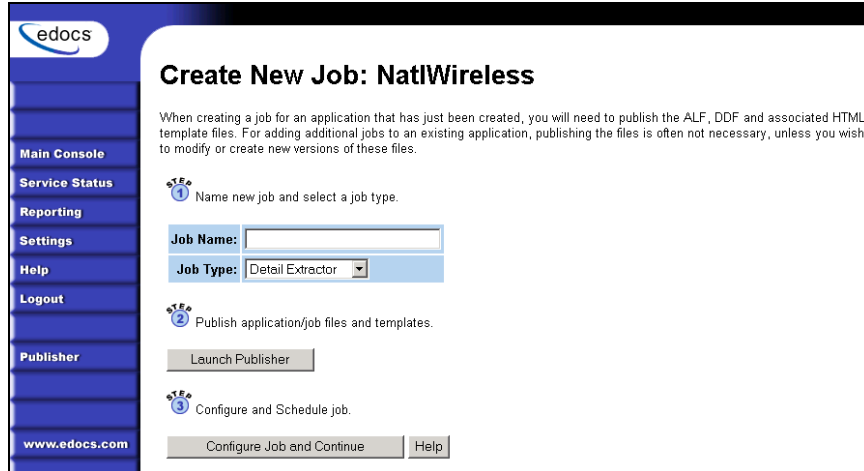
Review all the task and field configuration settings in this section to determine which options to use in your application.

To create and configure an EmailNotification job

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

Job Name	Job Type	Last Run	Status	Next Run	Delete?
Indexer	Indexer	None	Not scheduled	Not scheduled	<input type="checkbox"/>

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



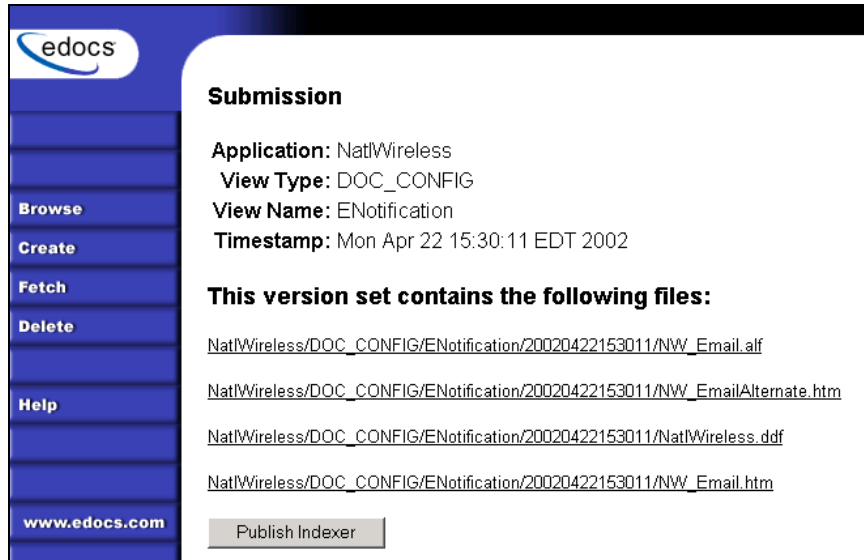
3. Enter a meaningful name for the job. The job name must start with an alpha character. The rest of the characters can be alphanumeric and can contain underscores, but no spaces.
4. Select the Email Notification job type from the drop-down menu.
5. Click **Launch Publisher**.
6. Click **Create**.
7. Under Batch Jobs, next to EmailNotification, click the number of auxiliary HTML templates the view uses. (The number of HTML templates *in addition to* the default template. If the view uses one HTML file, click **0**, if it uses two HTML templates, click **1**, etc.) Publisher displays the Create a Version Set for Email Notification screen:

The screenshot shows a web interface for edocs. On the left is a blue sidebar with navigation buttons: Browse, Create, Fetch, Delete, and Help. The main content area is titled 'Create a version set for Email Notification'. It contains the following fields and controls:

- Application:** A drop-down menu with 'Please select' as the current selection.
- View Name:** A text field containing 'ENotification'.
- DDF File:** A text input field with a 'Browse...' button to its right.
- ALF File:** A text input field with a 'Browse...' button to its right.
- HTML Template:** A text input field with a 'Browse...' button to its right.
- Auxiliary File:** A text input field with a 'Browse...' button to its right.
- Below the input fields are three buttons: 'Submit', 'Clear', and 'Help'.

At the bottom of the main content area, there is a copyright notice: '© Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)'.

8. Select the application name from the drop-down list. Select the DDF, ALF, and HTML files in the version set.
9. Click **submit**. Publisher displays the Submission screen showing the version set you just published:



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

edocs

Application: NatlWireless Job: EmailNotification

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: IVNScanner

Index Volume Status: Accepted

Scan Starting From (Number of Days): 7

Task 2: MailNotificationTask

Base URL: http://doc_hughes/eaSample/User?app=UserMain&jsp=/user/jsp/Detail.jsf

SMTP Hosts: SMTP host names (comma separated)

Return Address: Return email address

Subject Text: Email subject text

Administrator's Addresses: Administrator email addresses (comma separated)

Administrator Subject Text: Email Notification Error

Max Number of Retries: 3

12. Specify the configuration parameters for each EmailNotification 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 EmailNotification job later; see “Scheduling Jobs” on Page 97.
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.

IVNScanner looks for a date processed in the Date Accepted (or Date Rejected) column in the volumes table. A volume is a data input file that has been successfully processed by the Indexer job and referenced in the volumes table. The Indexer job must run before EmailNotification. The EmailNotification job processes one indexed volume at a time until IVNScanner finds no more newly indexed volumes listed in the volumes table.

If you selected “Intercept to Verify” in the Action on Index Volume option when configuring the AutoIndexVolAccept task in the Indexer job, you must use your customized verification/audit application to either accept or reject the indexed data before the EmailNotification job can process email for that data. (You can optionally choose to send email when the volume is rejected; this is useful in a test environment only.)

IVNScanner Task Input:	IVNScanner Task Output:
<ul style="list-style-type: none"> • A date processed value in the Date Accepted or the Date Rejected columns in the volumes table 	<ul style="list-style-type: none"> • (None)

IVNScanner Task Configuration (MailNotification job)		
Field	What to enter/select	
Index Volume Status	Specifies that the job can proceed to create and send email to customers when a date appears in either the Date Accepted or Date Rejected column in the volumes table.	
	Accepted	(Default) Choose this option if you do not have custom verification/audit application or if you have one and you want to send out email only after a volume has been approved. If you choose this option, IVNScanner looks for a date in the Date Accepted column; if it contains a date, it proceeds to generate email.
	Rejected	Use this setting to send email if a volume has been rejected in the custom verification application (use this option for testing purposes only). If you choose this option, IVNScanner looks for a date in the Date Rejected column; if it finds a date, it proceeds to generate email.
Scan 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.	

Task 2: MailNotification

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

The configuration settings you must specify for this task include the Base URL, the mail server name, the return address, the administrator's address, subject lines, etc., and how eaDirect handles delivery errors.

You also specify enrollment model settings to tell eaDirect where to look for email enrollment information. If you use eaDirect enrollment, by default, the system uses the email addresses found in the eaDirect database. If you use a customized enrollment application, you must specify the custom Enroll Model file name.

MailNotification Task Input:	MailNotification Task Output:
<ul style="list-style-type: none"> • Index data • Data file • Email version set (.DDF, .ALF, .HTML, auxiliary files) • Task configuration settings • Recipient email information 	<ul style="list-style-type: none"> • Emails sent to the specified SMTP for delivery • Flag added to the database indicating "sent" status flag for reporting purposes

MailNotification Task Configuration	
Field	What to enter/select
Base URL	Specify the Web address where the user can view their statement after receiving an email (this is usually included as a link in the email message).
SMTP Hosts	Specify the mail server name. You can specify multiple host systems, separated by commas.
Return Address	Specify the "from" address that appears in the notification. The user can reply to this address.
Subject Text	Enter the text to appear in the subject line in email notifications.
Administrator's Addresses	Specify the address to receive email if there's a problem passing it to the SMTP host or if it is not working properly for some other reason. Enter multiple addresses separated by commas (the mail goes to all addresses).
Administrator Subject Text	Enter the text to use in the subject line of messages sent to the administrator addresses.
Max Number of Retries	Specify the number of times eaDirect should try sending the mail to the mail server in the event of an error (the server could be busy or down) before the job stops.
Retry Period (min)	Specify how frequently eaDirect should retry sending email if it fails. The default is 60 minutes.
Enroll Model	Specify the name of the enrollment resolver your application uses.
Access Type	<p>This field provides a way to pass an extra parameter to one of edocs enrollment applications (edocs or CDA).</p> <p>Leave this field blank if you do not have customizations to either edocs application, or if you use a customized enrollment application. Consult the programmer if your application requires passing an extra parameter.</p>

Creating and Configuring a Purge App Job

Create and configure the Purge App job to periodically remove index, email, reporting data, and detail database tables (if any) from your application database tables. Running this job frees up space on your database server.

You may also need to run a Purge App job to eliminate index references to historical statement data to prevent it from being included in summary page information.

Caution Once you run a Purge App job, the data deleted will no longer be available for statistical reporting.

When you configure a Purge App job, you specify settings for the three production tasks that run sequentially in the Purge App job:

- Task 1: PurgeIndexData – Purges records from the index, detail, annotations, and dispute tables that have a date processed older than the number of days you specify.
- Task 2: PurgeEmailData - Purges information from the dynamic mail queue table. This table grows rapidly if you send email.
- Task 3: PurgeActivityData - Purges user activity data.

Review the task and field configuration settings in this section to determine which options to set for your application. Note that you do not need to publish files for a Purge App job.

To create and configure a Purge App job:

1. On the Main Console, click the application name, listed under Applications in the table. The Edit Application screen appears.
2. Click **Add New Job**. eaDirect displays the Create New Job screen:

Setting Up a New Application and Jobs

edocs

Create New Job: NatlWireless

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.

1 Name new job and select a job type.

Job Name:

Job Type:

2 Publish application/job files and templates.

3 Configure and Schedule job.

3. Enter a meaningful name for the Purge App 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 job type Purge App. (You do not publish files for a Purge App job.)
5. Click **Configure Job and Continue**. eaDirect displays the Purge App job configuration screen:

edocs

Application: NatlWireless Job: PurgeApp

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.

Task 1: PurgeIndexData

Purge Prior To (Number of Days):

Task 2: PurgeEmailData

Purge Prior To (Number of Days):

Task 3: PurgeRptActivityData

Purge Prior To (Number of Days):

- For each task (described in the next section), enter a value appropriate for your application and database:

0	Purges all information to date
1	Purges all information up to midnight of the previous day
>1	Purges all information up to midnight of the day specified (limit is 3650 days, or ten years)
-1	Do not purge. Choose this option if you do not want to purge the task data from the Index table when you run the Purge App job.

- 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 Purge App job later.
- Click **Main Console**.

Task 1: PurgeIndexData

PurgeIndexData purges records from the index, detail, annotation, and dispute tables with a date processed outside the range you specify here.

The range you specify matches a list of indexed documents based on the date they were processed. For each document, PurgeIndexData purges the index references, detail tables (loaded by the Detail Extractor job), and the related dispute and annotations information, regardless of their state or the date the information was submitted.

You have the option not to purge data with the Purge App job.

Note that data input files remain in the data directory. You must manually remove them as needed.

Task 2: PurgeEmailData

Purges information from the dynamic mail queue table with a date processed outside the range you specify here. This table grows rapidly if you send email. This task purges records from the email table with a date processed that falls outside the range you specify here.

You have the option not to purge email data with the Purge App job.

Task 3: PurgeActivityData

PurgeActivityData purges user activity data with a date processed outside the range you specify here. This task purges records from the activity table with a date processed that falls outside the range you specify here.

You have the option not to purge report data when you run the Purge App job.

Creating and Configuring a Purge Logs Job

Create and configure the Purge Logs job to periodically remove historical information from the log table in the system database. Purge Logs removes data for all applications you have. Running this job frees up space on your database server.

When you configure a Purge Logs job, you specify settings for the Purge Logs production task. Note that you do not need to publish files for a Purge Logs job.

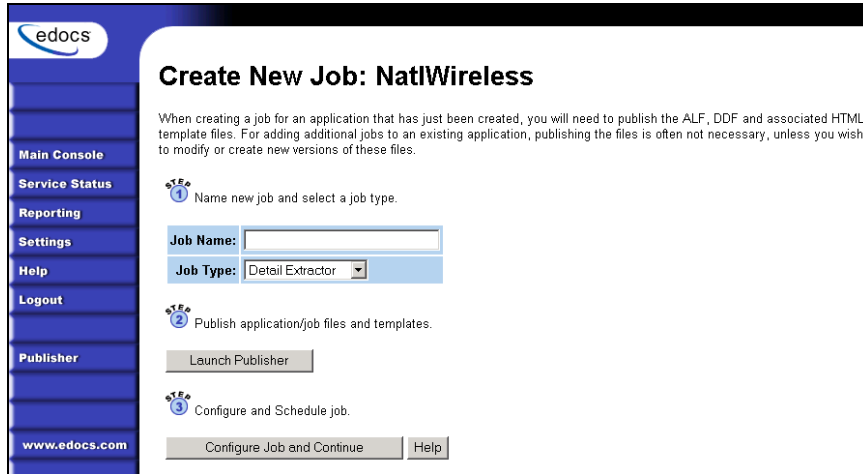


Caution Purge Logs removes data globally (for all applications you have). Once you run a Purge Log job, the data deleted will no longer be available for inclusion on View Log reports.

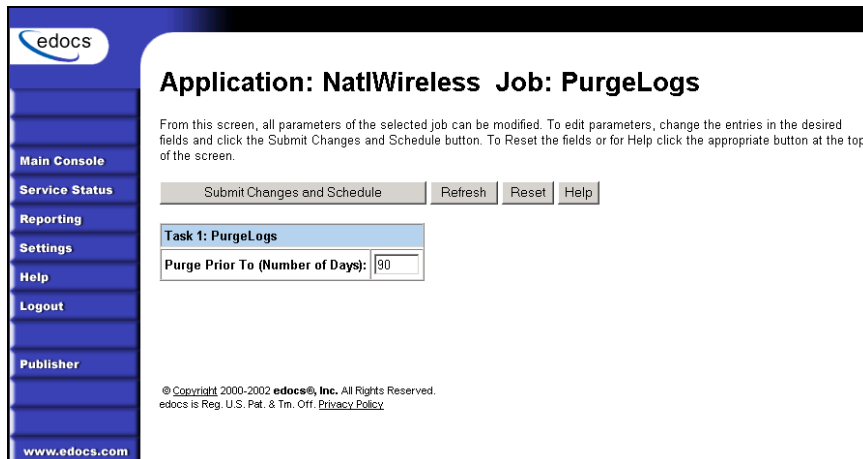
To create and configure a Purge Logs job:

1. On the Main Console, click the application name, listed under Applications 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 Purge Logs 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 job type Purge Logs.
5. Click **Configure Job and Continue**. eaDirect displays the Purge Logs configuration screen:



6. Specify the configuration parameters for the PurgeLogs task.
7. When finished, click **Submit Changes and Schedule**. eaDirect submits the job configuration parameters and displays the Schedule screen. You can schedule the Purge Logs job later.
8. Click **Main Console**.

PurgeLogs Task

The PurgeLogs task purges records from the logs table. It removes records with a date processed outside the range you specify here.

PurgeLogs Task Configuration	
Field	What to enter/select
Purge Prior to (Number of Days)	Specify one of the following values to determine what to purge:
	0 Purges all information to date
	1 Purges all information up to midnight of the previous day
	>1 Purges all information up to midnight of the day specified

Creating and Configuring an HTML Output Job

An HTML Output job creates a static HTML output file for each primary key. Configure an HTML Output job only if you plan to limit access to your database and let users view a static HTML output file only. Static HTML output files may be necessary if you partner with thin or thick consolidators for statement presentment.



Caution

An HTML Output job is necessary only if you plan to provide users access to static HTML output. Use an Indexer job with dynamic HTML Web views to provide live retrieval of HTML-formatted statements.

Creating and configuring an HTML Output job requires you to:

- Specify configuration settings for the three production tasks that run sequentially as part of the Indexer job: Scanner, Indexer, and HTMLFormatter.
- Publish the DDF file for the application's Indexer task (if you don't already have an Indexer job defined for the application).
- Publish the HTML Output view files (DDF, ALF, and HTML templates) defined for the application.

Review the task and field configuration settings in this section to determine which options to set for your application.

To create and configure an HTML Output job:

1. On the Main Console, click the application name, listed under Applications in the table. The Edit Application screen appears.
2. Click **Add New Job**. eaDirect displays the Create New Job screen:

Setting Up a New Application and Jobs

The screenshot shows the 'edocs' web interface for creating a new job. The page title is 'Create New Job: NatlWireless'. A sidebar on the left contains navigation links: Main Console, Service Status, Reporting, Settings, Help, Logout, Publisher, and www.edocs.com. The main content area has a heading 'Create New Job: NatlWireless' and a paragraph explaining that new jobs require publishing ALF, DDF, and HTML template files. Below this are three numbered steps: 1. Name new job and select a job type. This step includes a 'Job Name' text input field and a 'Job Type' dropdown menu currently set to 'Detail Extractor'. 2. Publish application/job files and templates. This step includes a 'Launch Publisher' button. 3. Configure and Schedule job. This step includes a 'Configure Job and Continue' button and a 'Help' button.

3. Enter a meaningful name for the HTML Output job. The job name must start with an alpha character. The rest of the name can be alphanumeric and can have underscores, but no spaces.
4. Select job type HTML Output.
5. Click **Launch Publisher**. eaDirect launches Publisher.
6. Click **Create**. eaDirect displays the following screen:

The screenshot shows the edocs interface with a sidebar on the left containing navigation buttons: Browse, Create, Fetch, Delete, Help, and www.edocs.com. The main content area is titled 'Select a Version Set Type' and is divided into two sections: 'Dynamic Web Views' and 'Batch Jobs'.

Dynamic Web Views

Job Type	Number of Auxiliary Files
HTML	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 Output	0

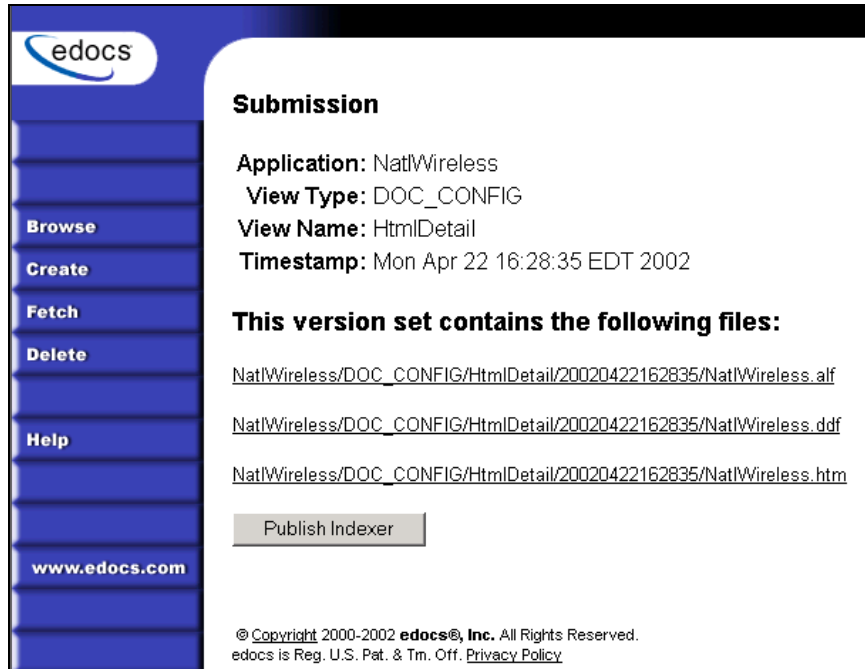
- Under Batch Jobs, next to HTML Output, click the number of auxiliary HTML templates the view uses (the number in addition to the default template). Publisher displays the Create a Version Set for HTML Output screen:

The screenshot shows the edocs web interface. On the left is a blue sidebar with the edocs logo at the top and a menu with items: Browse, Create, Fetch, Delete, Help, and www.edocs.com at the bottom. The main content area is titled 'Create a version set for HTML Output'. It contains the following fields and controls:

- Application:** A drop-down menu with 'Please select' as the current selection.
- View Name:** A text field containing 'HtmlDetail'.
- DDF File:** A text input field with a 'Browse...' button to its right.
- ALF File :** A text input field with a 'Browse...' button to its right.
- HTML Template:** A text input field with a 'Browse...' button to its right.
- Below the input fields are three buttons: 'Submit', 'Clear', and 'Help'.

At the bottom of the main content area, there is a copyright notice: '© Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)'.

8. Select the application name from the drop-down list.
9. Select the DDF, ALF, and HTML files in the HTML Output version set.
10. Click **submit**. Publisher displays the Submission screen with details about the version set:



11. If you *don't* already have an Indexer job configured for this application, click **Publish Indexer** to publish a DDF for the Indexer task that runs as part of the HTML Output job (If you already have an Indexer job for this application, go to Step 14.) eaDirect displays the Create a Version Set for Indexer screen:

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

- Application:** A dropdown menu currently showing "NatlWireless".
- View Name:** A text input field containing "Indexer".
- DDF File:** A text input field followed by a "Browse..." button.
- Below the DDF File field are three buttons: "Submit", "Clear", and "Help".

At the bottom of the main content area, there is a copyright notice: "© Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)".

12. Specify the DDF file for the Indexer task.
13. Click **submit**. eaDirect displays the Submission screen for the DDF:

The screenshot shows a web interface for edocs. On the left is a vertical blue navigation menu with buttons for 'Browse', 'Create', 'Fetch', 'Delete', and 'Help'. The main content area is titled 'Create a version set for Indexer'. It contains a form with the following fields: 'Application' (a dropdown menu showing 'NatlWireless'), 'View Name' (a text input field containing 'Indexer'), and 'DDF File' (a text input field with a 'Browse...' button to its right). Below the 'DDF File' field are three buttons: 'Submit', 'Clear', and 'Help'. At the bottom of the main content area, there is a copyright notice: '© Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. Privacy Policy'. The edocs logo is in the top left corner, and the website address 'www.edocs.com' is at the bottom left of the page.

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

Setting Up a New Application and Jobs

The screenshot shows the configuration interface for the 'Application: NatlWireless Job: HTMLOutput'. The interface includes a sidebar with navigation options: Main Console, Service Status, Reporting, Settings, Help, Logout, and Publisher. The main content area contains instructions and three task configuration sections.

Application: NatlWireless Job: HTMLOutput

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: c:\EDCSbd\input\NatlWireless\
Input File Name: *.alp
Output File Path: c:\EDCSbd\Data\NatlWireless\

Task 2: Indexer

DDF Path: c:\EDCSbd\AppProfiles\NatlWireless\DOC_CONFIG/Indexer\20020422163253\NatlWireless.ddf
Doc Date: ----Today's Date----
Index Field List: CurrentCharges* CustName LateFee* AmountDue LastStmIndicator* PymtTx* StatementDate FirstStmIndicator*

Task 3: StaticHtmlFormatter

DDF Path: c:\EDCSbd\AppProfiles\NatlWireless\DOC_CONFIG/HtmlDetail\20020422162835\NatlWireless.ddf

16. Specify the configuration parameters for each of the three tasks that run as part of the HTML Output job. Carefully read the descriptions of each task and field (below) to choose the values appropriate for your application and job.
17. When finished entering configuration parameters, click **Submit Changes and Schedule**. eaDirect submits the job configuration parameters and displays the Schedule screen. You can specify the HTML Output job schedule later.
18. Click **Main Console**.

Task 1: Scanner

See page 38 for a description of Scanner task parameters.

Task 2: Indexer

See page 39 for a description of Indexer task parameters.

Task 3: StaticHtmlFormatter

The StaticHtmlFormatter task uses the data file and most recent HTML version set published (DDF, ALF, and HTML's) to generate an HTML-formatted static output file for each primary key in the data file.

StaticHtmlFormatter Task Input:	StaticHtmlFormatter Task Output:
<ul style="list-style-type: none">• Data input file• Most recent HTML version set published (.DDF, .ALF, & HTMLs)	<ul style="list-style-type: none">• A static HTML file for each primary key in the /Output/Application Name directory

StaticHtmlFormatter Task Configuration	
Field	What to enter/select
DDF Path	(Not an editable field.) Displays the full pathname of the DDF published with the version set for this job.
ALF Path	(Not an editable field.) Displays the full pathname to the ALF file you published with the version set for the job.
HTML Template Path	(Not an editable field.) Displays the full pathname to the HTML template you published with the version set for this job.
Output Path	Enter the full pathname where you want eaDirect to place the Static HTML-formatted output. The default directory is \$EDX_HOME/Output/Application Name.

Creating and Configuring an XML Output Job

An XML Output job creates a static XML output file. Configure an XML Output job if you plan to generate a static XML output file for loading into another database.



Caution An XML Output job is necessary only if you plan to generate static XML output. Use an Indexer job with dynamic XML Web views to provide users with live retrieval of XML-formatted statements.

Creating and configuring an XML Output job requires you to:

- Specify configuration settings for the three production tasks that run sequentially as part of the Indexer job: Scanner, Indexer, and XMLFormatter.
- Publish a DDF file for XML output (if you don't already have an Indexer job defined for this application.)
- Publish the same DDF file for use by the Indexer task.

Review the task and field configuration settings in this section to determine which options to set for your application.

You must publish the DDF file created for your XML Output job.

To create and configure an XML Output job:

1. On the Main Console, click the application name, listed under Applications in the table. The Edit Application screen appears.
2. Click **Add New Job**.
3. Enter a meaningful name for the XML Output job. The job name must start with an alpha character. The rest of the name can be alphanumeric and can have underscores, but no spaces.
4. Select job type XML Output.
5. Click **Launch Publisher**. eaDirect launches Publisher.
6. Click **Create**. Publisher displays the following screen:

The screenshot shows the edocs application interface. On the left is a blue sidebar with navigation buttons: 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.

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

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 Output	0

7. Under Batch Jobs, Number of Auxiliary files, click the 0 next to the XML Output job type. Publisher displays the Create a Version Set for XML Output screen:

The screenshot shows a web application interface with a blue sidebar on the left containing navigation buttons: Browse, Create, Fetch, Delete, and Help. The main content area is titled "Create a version set for XML Output". It features a form with the following fields and controls:

- Application:** A drop-down menu with "Please select" as the current selection.
- View Name:** A text input field containing "XmlDetail".
- DDF File:** A text input field followed by a "Browse..." button.
- Below the DDF File field are three buttons: "Submit", "Clear", and "Help".

At the bottom of the main content area, there is a copyright notice: "© Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)". The sidebar also includes the "edocs" logo at the top and the website address "www.edocs.com" at the bottom.

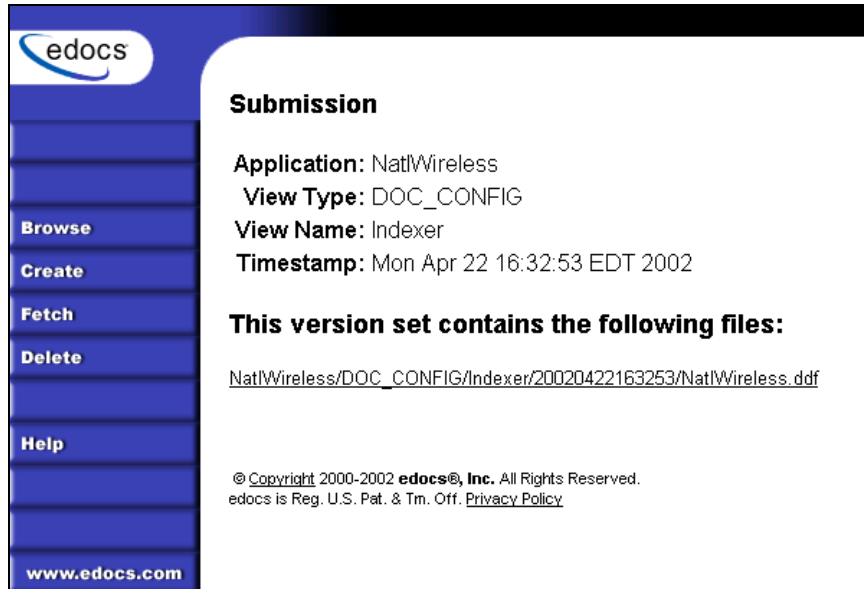
8. Select the application from the drop-down list.
9. Select the DDF file for the XML Output job.
10. Click **submit**. Publisher shows the Submission screen for the XML Output DDF:



11. If you *don't* already have an Indexer job configured for this application, click **Publish Indexer** to publish a DDF file for the Indexer task that runs as part of the XML Output job. (If you already have an Indexer job for this application, go to Step 12.) eaDirect displays the Create a Version Set for Indexer screen:

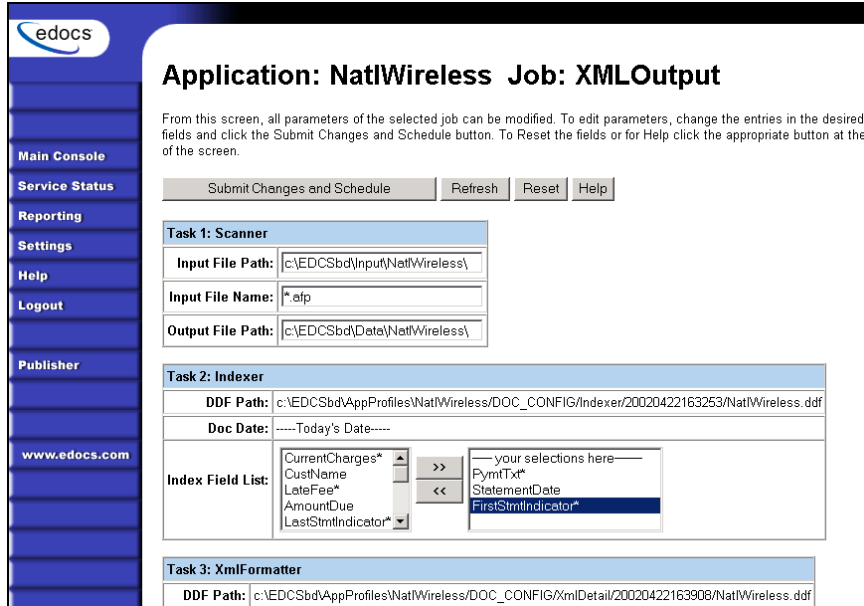
The screenshot shows the edocs web interface. On the left is a blue sidebar with the edocs logo at the top and a menu with the following items: Browse, Create, Fetch, Delete, Help, and www.edocs.com at the bottom. The main content area is white and titled "Create a version set for Indexer". It contains the following form elements: "Application:" with a dropdown menu showing "NatlWireless"; "View Name:" with a text input field containing "Indexer"; "DDF File:" with a text input field and a "Browse..." button; and three buttons: "Submit", "Clear", and "Help". At the bottom of the main area, there is a copyright notice: "© Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)".

12. Select the DDF file to use for the Indexer task and click **submit**. eaDirect displays the DDF for Indexer task Submission screen:



13. Close Publisher.

At the Create New Job screen in Command Center, click **Configure Job and Continue**. eaDirect displays the XMLOutput job configuration screen:



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

Task 1: Scanner

See page 38 for a description of Scanner task parameters.

Task 2: Indexer

See page 39 for a description of Indexer task parameters.

Task 3: XMLFormatter

The XMLFormatter task uses the data file and the DDF published with the version set for the XML Output job to generate a static, XML-formatted output file for each primary key in the data file.

XMLFormatter Task Input:	XMLFormatter Task Output:
<ul style="list-style-type: none"> • Data input file • Most recent DDF published for the XML Output job 	<ul style="list-style-type: none"> • A static XML file in the /Output/Application Name directory

XMLFormatter Task Configuration	
Field	What to enter/select
DDF Path	Enter the full pathname to the DDF File. You must specify the same DDF file that you publish for the XML Output job Indexer task.
Output Path	Enter the full pathname where you want eaDirect to place the Static XML-formatted output. The default directory is \$EDX_HOME/Output/Application Name.

Creating and Configuring a Detail Extractor Job

A Detail Extractor job lets you to use XSL to upload recurring detail data from your input file to the eaDirect database. You can then use the uploaded data any way you want. For example, you could upload the contents of telephone call detail, merge it into your online statements, and let your Web users track line item disputes, add annotations, etc. You must write separate applications to retrieve the data and perform any functionality with the data.

You must configure and run a separate Detail Extractor job for each set of data (table or group of tables) you want to upload to a database table.

Creating and configuring a Detail Extractor job requires you to:

- Specify configuration settings for the three production tasks that run sequentially as part of the Detail Extractor job: IVNScanner, StatementsToIR, and DXLoader.
- Publish the Detail Extractor view files, which include a DDF file with the rules for extracting the table data, database table schema XML, and a statement XSLT style sheet to organize the extracted data the database table. You can use the editor of your choice to create the XML and XSLT files.

You can use the same DDF file for multiple Detail Extractor jobs; define all tables and groups you plan to upload to a database in the DDF.

Review the task and field configuration settings in this section to determine which options to set for the Detail Extractor job.

To create and configure a Detail Extractor job:

1. On the Main Console, click the application name, listed under Applications in the table. The Edit Application screen appears.
2. Click **Add New Job**. eaDirect displays the Create New Job screen.

Setting Up a New Application and Jobs

edocs

Create New Job: NatlWireless

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.

1 Name new job and select a job type.

Job Name:

Job Type:

2 Publish application/job files and templates.

3 Configure and Schedule job.

3. Enter a meaningful name for the Detail Extractor job. The job name must start with an alpha character. The rest of the name can be alphanumeric and can have underscores but no spaces. (Using the name of the table defined in the DDF for the job name can help you remember which data the job uploads.)
4. Select job type **Detail Extractor**.
5. Click **Launch Publisher**. eaDirect launches Publisher.
6. Click **create**. Publisher displays the following screen:

The screenshot shows the edocs application interface. On the left is a blue sidebar with navigation buttons: 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

Job Type	Number of Auxiliary Files
HTML	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 Output	0

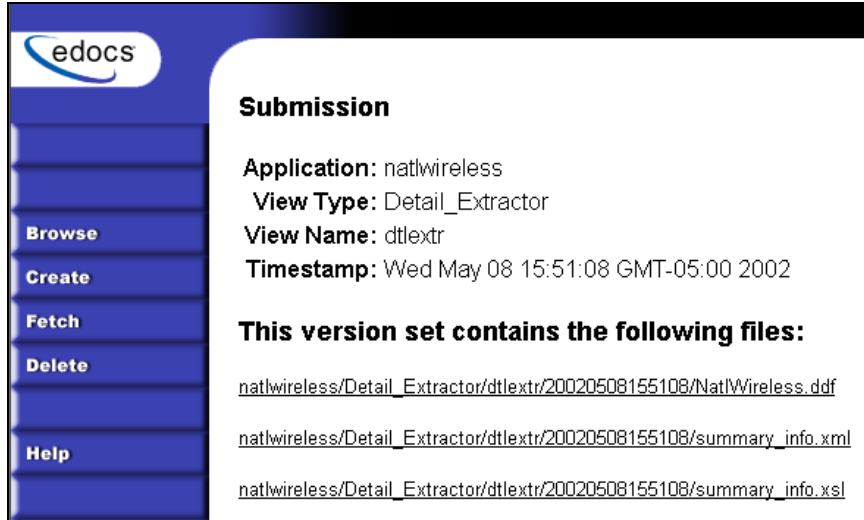
- Under Batch Jobs, Number of Auxiliary files, click the 0 next to the Detail Extractor job type. Publisher displays the Create a Version Set for Detail Extractor screen:

The screenshot shows the edocs web interface. On the left is a blue navigation sidebar with buttons for Browse, Create, Fetch, Delete, and Help. The main content area is titled "Create a version set for Detail Extractor". It contains the following fields and controls:

- Application:** A drop-down menu with "Please select" as the current selection.
- View Name:** A text input field.
- DDF File:** A text input field with a "Browse..." button to its right.
- Database Table XML File:** A text input field with a "Browse..." button to its right.
- Statement XSLT File:** A text input field with a "Browse..." button to its right.
- At the bottom of the form are three buttons: "Submit", "Clear", and "Help".

At the bottom of the page, there is a copyright notice: "© Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)". The website address "www.edocs.com" is visible in the bottom left corner of the page.

8. Select the application from the drop-down list.
9. Specify a view name for the job that uniquely identifies the view of table data the job extracts. (Using the name of the table defined in the DDF for the view name can help you remember which view maps to which table.) Do not reuse the name of a view given to another Detail Extractor job.
10. Select the DDF file for the Detail Extractor job.
11. Select the database table XML file you created for the view.
12. Select the statement XSLT style sheet.
13. Click **submit**. Publisher shows the Submission screen for the Detail Extractor DDF:



- Close Publisher. At the Create New Job screen in Command Center, click **Configure Job** and **Continue**. eaDirect displays the Detail Extractor job configuration screen:



15. Specify the configuration parameters for each of the three tasks that run as part of the Detail Extractor job. Carefully read the descriptions of each task and field (below) to choose the values appropriate for your application and job.
16. When finished entering configuration parameters, click **Submit Changes and Schedule**. eaDirect submits the job configuration parameters and displays the Schedule screen. You can specify the Detail Extractor job schedule later. (You must schedule Detail Extractor to run after the Indexer job; Detail Extractor only uploads data for statements that have been indexed).
17. Click **Main Console**.

Task 1: IVNScanner

The IVNScanner task determines whether index data has been verified before letting the Detail Extractor job continue processing. This task is primarily intended for applications with a customized verification/audit application.

IVNScanner looks for a date processed in the Date Accepted (or Date Rejected) column in the volumes table. A volume is a data file that has been successfully processed by the Indexer job and referenced in the volumes table. The Indexer job must run before Detail Extractor. The Detail Extractor job processes one indexed volume at a time until IVNScanner finds no more newly indexed volumes listed in the volumes table.

If you selected “Intercept to Verify” in the Action on Index Volume option when configuring the AutoIndexVolAccept task in the Indexer job, you must use your customized verification/audit application to either accept or reject the indexed data before the Detail Extractor job can upload data from the indexed data input file. (You can optionally choose to run the job when the volume is rejected; this is useful in a test environment only.)

IVNScanner Task Input:	IVNScanner Task Output:
<ul style="list-style-type: none"> A date processed value in the Date Accepted or the Date Rejected columns in the volumes table 	<ul style="list-style-type: none"> (None)

IVNScanner Task Configuration	
Field	What to enter/select
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.
	<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, the Detail Extractor job proceeds.</p>
	<p>Rejected</p> <p>Use this setting to continue the job 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, the job proceeds.</p>

Scan 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.
-------------------------------------	--

Task 2: StatementsToIR

The StatementsToIR task uses the DDF, database table schema XML, and the XSLT style sheet view files published with the job to extract the recurring detail content from the data input file and place the data into an Intermediate Representation (IR) file.

StatementsToIR Task Input:	StatementsToIR Task Output:
<ul style="list-style-type: none"> • Data source file in the output file path • Most recent DDF published in AppProfiles directory (not configurable) • The database table schema XML and the XSLT style sheet files published with the view 	<ul style="list-style-type: none"> • An .IR file containing the extracted data, in the data file path

StatementsToIR Task Configuration	
Field	What to enter/select
View Name	The name of the view you published for this job. The Detail Extractor view consists of a DDF, a database table schema XML, and an XSLT style sheet.
Enroll Model	Specify the name of the enrollment resolver your application uses.

Output File Path	Specify the name of the application data directory where you want to create the IR file. The default is your data output directory (which you specify in the configuration of your Indexer job's Scanner task).
------------------	---

Task 3: DXLoader

The DXLoader task creates a new detail database table and uploads the detail data in the .IR file to the table using the database platform's SQL loader.

DXLoader Task Input:	DXLoader Task Output:
<ul style="list-style-type: none"> .IR file 	<ul style="list-style-type: none"> A database table with content .LOG, .CTL, and .CMD files (creates a .bad file if the SQLLDR or BCP fail)

DXLoader Task Configuration		
Field	What to enter/select	
Load Method	Choose one of the following database load methods:	
	Direct load	Stores data directly to the database. This option locks the Detail table, loads the information, and ends the call to the database. No sharing of tables is allowed during this process. A direct load is the fastest method.
	Conventional load	Uses Insert statements, one row at a time. Performs multiple selects and inserts on the table at once, but does not lock up the database and lock out Web users. A conventional load is slower than a direct load.

Publishing Your Application's Dynamic Web Views

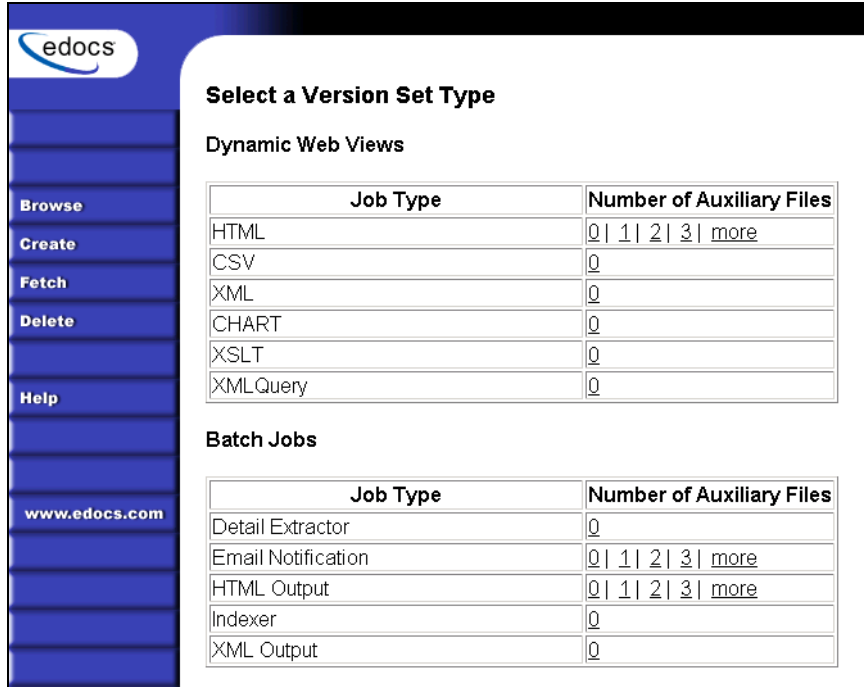
After creating and configuring your application jobs, you must publish each dynamic Web view in your application.

"Publishing" a dynamic Web view identifies each file belonging to a particular view, and lets you give the view a name. When a user clicks a link to their statement ("running" a dynamic Web view), the link identifies the set of design files to use to extract and present the user's statement data.

Publishing dates the set of view files with a timestamp. Publishing a view is also called "creating a version set." A version set is a dated set of design files.

To publish a dynamic Web view for an application:

1. Click Publisher on the Command Center Main Console
2. Click **create**. Publisher displays the following screen:



- For a dynamic HTML version set, under Dynamic Web Views, next to HTML, click the number of auxiliary HTML templates this view uses (in addition to one); if the view uses one HTML file, click 0. If the view uses two HTML template files, then click 1, etc.

To publish a CSV, XML, Chart, XSLT, or XML Query view, under Dynamic Web Views, Number of Auxiliary Files, click the 0 next to the job type. Publisher displays the Create a Version Set screen for the view type you're publishing:

4. Select the application name from the drop-down list.
5. Enter the view name; this is the specific name the Web browser looks for in the code. This view name is customized in your JSP and HTML pages; consult your design team for details.
6. Select the names of the design files to publish. You publish the following files for each type of dynamic Web view:

HTML	A DDF, ALF, and one or more HTML files
CSV	A DDF and a TOK file
XML	A DDF file
Chart	A Properties file
XSLT	An XSLT file and an application DDF
XML Query	A DDF file

7. Click Submit. Publisher displays the Submission screen showing the files you published:

edocs

Submission

Application: NatlWireless
View Type: HTML
View Name: HTMLDetail
Timestamp: Mon Apr 22 16:56:53 EDT 2002

This version set contains the following files:

[NatlWireless/HTML/HTMLDetail/20020422165653/NatlWireless.alf](#)
[NatlWireless/HTML/HTMLDetail/20020422165653/NatlWireless.ddf](#)
[NatlWireless/HTML/HTMLDetail/20020422165653/NatlWireless.htm](#)

[www.edocs.com](#)

© Copyright 2000-2002 edocs®, Inc. All Rights Reserved.
edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)

8. If you have any additional dynamic Web views for your application, click **Create** and continue to publish version sets for those views.

Once your application and jobs are defined and you've published your dynamic Web views, you can proceed to schedule jobs for live production.

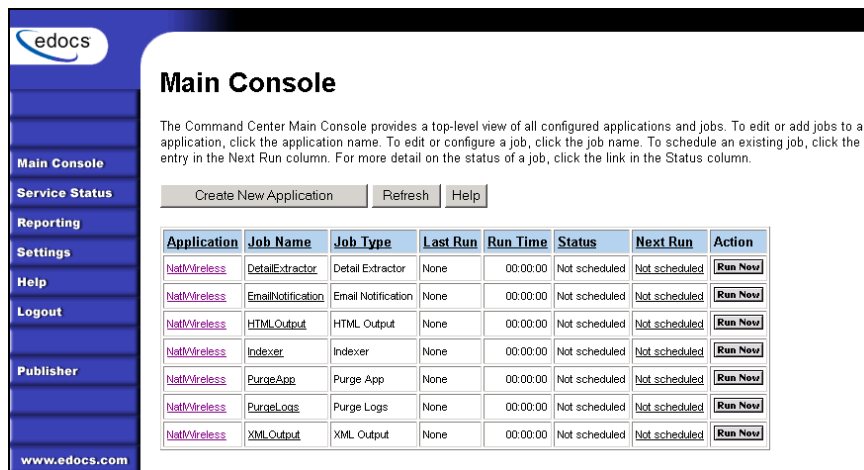
Managing the Live Production Process

3

General Production Monitoring Activities

Once you have set up and configured an application and its jobs, you use the eaDirect Command Center to schedule jobs, manage the production process on a daily basis, and to perform administrative activities related to your eaDirect application.

The Command Center Main Console provides a high-level status of all activity related to eaDirect jobs in the production environment, and is the first screen you see when you log into the Command Center:



The screenshot shows the 'Main Console' interface for edocs. It features a left-hand navigation menu with options like 'Main Console', 'Service Status', 'Reporting', 'Settings', 'Help', 'Logout', and 'Publisher'. The main content area displays a table of configured applications and jobs. Above the table are buttons for 'Create New Application', 'Refresh', and 'Help'. The table has columns for Application, Job Name, Job Type, Last Run, Run Time, Status, Next Run, and Action.

Application	Job Name	Job Type	Last Run	Run Time	Status	Next Run	Action
NatVireless	DetailExtractor	Detail Extractor	None	00:00:00	Not scheduled	Not scheduled	Run Now
NatVireless	EmailNotification	Email Notification	None	00:00:00	Not scheduled	Not scheduled	Run Now
NatVireless	HTML_Output	HTML Output	None	00:00:00	Not scheduled	Not scheduled	Run Now
NatVireless	Indexer	Indexer	None	00:00:00	Not scheduled	Not scheduled	Run Now
NatVireless	PurgeApp	Purge App	None	00:00:00	Not scheduled	Not scheduled	Run Now
NatVireless	PurgeLogs	Purge Logs	None	00:00:00	Not scheduled	Not scheduled	Run Now
NatVireless	XML_Output	XML Output	None	00:00:00	Not scheduled	Not scheduled	Run Now

You also use the Main Console to schedule, control, and monitor all eaDirect production jobs, including:

- Setting and changing job schedules

- Monitoring the status of jobs and individual production tasks
- Starting a job
- Viewing job output
- Publishing new version sets
- Monitoring system services

Keeping your eaDirect applications running efficiently in an ongoing, live production environment requires regular monitoring and maintenance. Here are a few of the system monitoring activities you want to perform on a regular basis:

Daily application monitoring tasks:

- Check the Command Center Status screen to monitor the state of production jobs. See “Monitoring Production Jobs” on Page 101 for details on verifying job completion.
- Check the administrator email accounts for any eaDirect administrator alert mail. eaDirect generates administrator email if there's a problem passing email notifications to the SMTP host or if email notification is not working properly for some other reason.

Weekly (or more often) application monitoring tasks:

- Check message log report messages: Activity, Error, and Warning Logs. See “Viewing Message Logs” on Page 107.

General system maintenance activities:

- Run activity reports to review application usage statistics. See “Reviewing System Activity” on Page 143.
- Maintain the eaDirect database. See “Database Administration” on Page 151.”

Scheduling Jobs

You must manually schedule jobs to run in a live production environment. eaDirect does not automatically schedule any jobs. The frequency with which you choose to run a job depends on both the job type and your organization's presentment needs. Consider all jobs and system events in planning your schedule.

You can schedule a job to run a simple weekly or monthly basis, or establish a more complex timetable. Review the available scheduling parameters and carefully choose the combination of options that yield the particular schedule you need.

You can change a job schedule anytime, *except while the job is processing*.



If you try to save schedule changes while a job is running in the production queue (job status says "Processing"), eaDirect ignores the new scheduling parameters.


Here is a general idea of how often you might want to run various jobs:

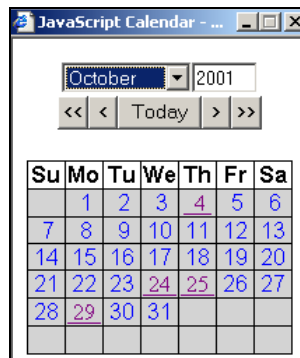
Job Type	How Often to Run
Indexer	<p>Schedule the Indexer job on a regular basis that corresponds with the generation of a new input data file, such as a monthly account statement.</p> <p>You can schedule an Indexer job to rerun frequently over a certain period of time to allow for the expected arrival of a data input file at an unspecific point. (Only when the Scanner task finds a data input file does the job proceed.)</p>
EmailNotification	Schedule EmailNotification to briefly lag the Indexer job so you can verify its completion before notifying users by email that a statement is available.
Purge Apps	Schedule Purge Apps to run as often as necessary to clear space on your database server and limit user access to historical statements.
Purge Logs	Schedule Purge Logs to run as often as necessary to clear

Job Type	How Often to Run
	space on your database server.
HTML Output; XML Output	<p>Schedule an HTML Output or XML Output job on a regular basis that corresponds with the generation of a new input data file, such as a monthly account statement.</p> <p>You can schedule HTML Output or XML Output to rerun frequently over a certain period of time to allow for the expected arrival of a data input file at an unspecified point. (The Scanner task runs as part of both job types, and only when it finds a data input file does the job proceed.)</p>

To set or change a job schedule:

1. At the eaDirect Command Center, locate the job you want to schedule or reschedule 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.)

- Specify a valid start date for the schedule to take effect. Click  Popup Calendar to select dates quickly:



- Specify the schedule window parameters and any repeating parameters, if necessary. These options are described in the following table. To clear the screen to reenter all parameters, click **Clear Schedule**.

Job Schedule Parameters	
Field	Use to...
<i>Schedule Date</i>	
Schedule Date	Specify the date the job schedule goes into effect
<i>Schedule Window</i>	
Start Time	Time of day (hour and minutes) to run the job
Try Once	Run the job once on date and time specified only
Try every ... minutes until ... time of day	Rerun the job at the specified interval (in minutes) until the end time
<i>Repeating</i>	
Do not repeat this event	Run the job as specified in the previous fields and do not repeat it
Repeat every ...	Run the job on the daily or weekly frequency specified: every, every other, every third, or every fourth day, on the selected day of week, all week days, or on both weekend days
Repeat on day ... of the month every ...	Run the job on the numeric day of the month specified, on every month, every other month, or every 3, 4, 6, or 12 months
Forever	Run the schedule continuously (no end date)
Until ...	Run the schedule up until the end date, then stop

- 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 run the job immediately, click **Run Now**. (You can also choose this button on the Main Console.)

Monitoring Production Jobs

Use the Command Center Main Console to monitor the state of all production jobs for your applications.

Regularly check the status of jobs and tasks to track:

- Whether a job has completed successfully
- Which tasks completed successfully
- Why a job failed

You can use the Command Center to correct problems and restart or cancel failed jobs. The Main Console displays the following information for each application and job:

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, EmailNotification, Purge Apps, Purge Logs, HTML Output, XML Output, or Detail Extractor.
Last Run	Date and time the job last ran.
Run Time	Elapsed time the scheduled job has been running in hours, minutes, and seconds.
Status	Current execution state of the job.
Next Run	Date and time the job is scheduled to run next.
Action	Displays a button that lets you take action on that job. The Run Now button lets you run the job immediately, overriding scheduling parameters; the Retry button lets you rerun a failed job.

Note that the Main Console does not show any activity until you create one or more applications and jobs.

To list jobs for a particular application only:

- On the Main Console, click the name of the application in the Application column. The Edit Application page appears, showing only those jobs defined for the selected application.

To sort jobs listed on the Main Console by application:

- Click the word **Application** in the column header.

To sort jobs on the Main Console by job name (alphabetically), job type, last run, run time, status, or next run:

- Click the column header.

To display the current information on the Main Console:

- Click the **Refresh** button.

Viewing job status

The status of each eaDirect production job appears on the Command Center Main Console. A job can be in one of the following states:

Job Status	Description
Canceled	Job run failed and was canceled
Cancelled, 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.
Done	Job has completed successfully
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.
Failed	Job failed
Not started	Job has not begun executing
Not scheduled	Job has not been scheduled to run
Reprocessing	Job is currently executing after being marked for reprocessing
Processing	Job is currently executing
No operation	Scanner task cannot locate the input data file for the scheduled job
No operation, recurring	Previous job run resulted in a "No operation" status, but the job has been scheduled to run again

Viewing and verifying task status

Each production jobs consist of several individual tasks that work together to generate job output. eaDirect also assigns a status to the individual tasks that run when the job runs.

Every configured task must complete successfully before eaDirect sets job status to **Done** on the Main Console.

If any of the production tasks is unable to complete, the job fails, and the status changes to **Failed**. eaDirect displays all failed jobs in red on the Main Console. If a job fails, you can run it again.

To view task status detail for a job:

- Click the status of the job in the Main Console **Status** column. The Task Status screen appears showing the status of each production task in the job.

To change the display order of tasks (processing order remains unchanged), click **Task**. To change the display order of information in the Last Run and Status columns, click **Last Run** or **Status**. Click the links again to restore each display to its original order.

Click **Refresh** to display an updated task status.

You have the option of rerunning or canceling a failed job. Click **Retry Failed Job**, or **Cancel Failed Job**.

To verify that a task completed successfully

- In addition to checking the individual task status on the Task Status screen, you can check for individual task output to determine whether a task completed successfully, as described in the following table:

Task	How to verify task completion
Scanner	Check the job's status window. Verify that the data file was moved from the input directory to the data directory and renamed.
Indexer	Check the job status window. Verify the .IR file is in the /Data/Application Name directory.
IXLoader Task	Check the job status window. Verify the appearance of .CTL, .CMD, and .LOG files. Check for Index references (one per primary key) within the index table in database.
AutoIndexVolAccept	Check the job status window.
IVNScanner	Check the job status window.
Mail Notification	Check the job status window. Verify emails sent.
HTMLFormatter	Check the job status window. Verify the appearance of static HTML output files.
XMLFormatter	Check the job status window. Verify the appearance of static XML output.
StatementsToIR	Check the job status window. Verify the .IR file in the /Data/Application Name directory.
DXLoader	Check the job status window. Verify the existence of a populated database table.

Canceling and retrying failed jobs

Retry running a failed job if you want to rerun it from the point where it failed. If you want to restart the entire job, cancel the job and run it again.

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

- On the Main Console, click the **Retry** button for the failed job, or click the status of a job in the Status column to display the Task Status screen, and click **Retry Failed Job**.

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

To cancel a *failed* job:

- On the Main Console, click the status of a job in the Status column to display the Task Status screen, and click **Cancel Failed Job**.

eaDirect changes the job status to Canceled, which remains until the next time the job is scheduled to run again.

Viewing Message Logs

eaDirect maintains logs of all activities that occur and messages generated during production. Review these logs on a regular, ongoing basis to monitor jobs in your production environment.

You can create and view a report showing any of the following types of log messages generated over a select time period:

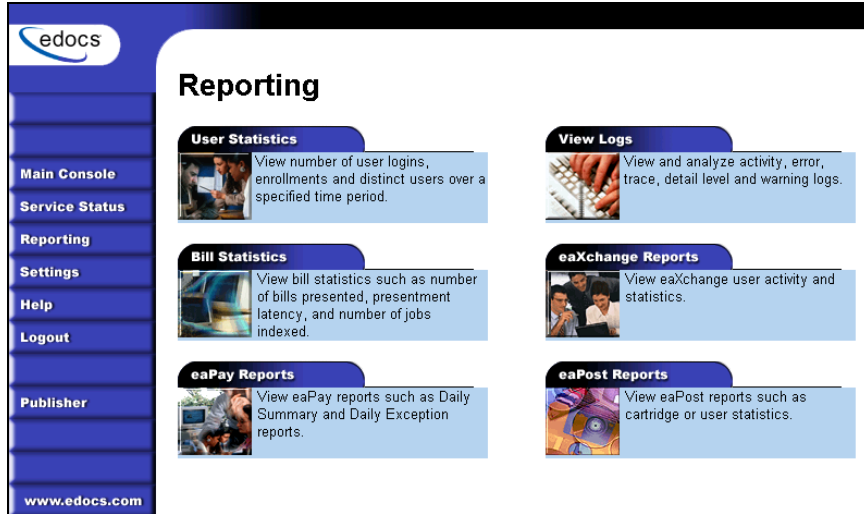
- **Error** – Error log
- **Information** – Activity log
- **Warning** – Warning log

eaDirect log reports display the following information:

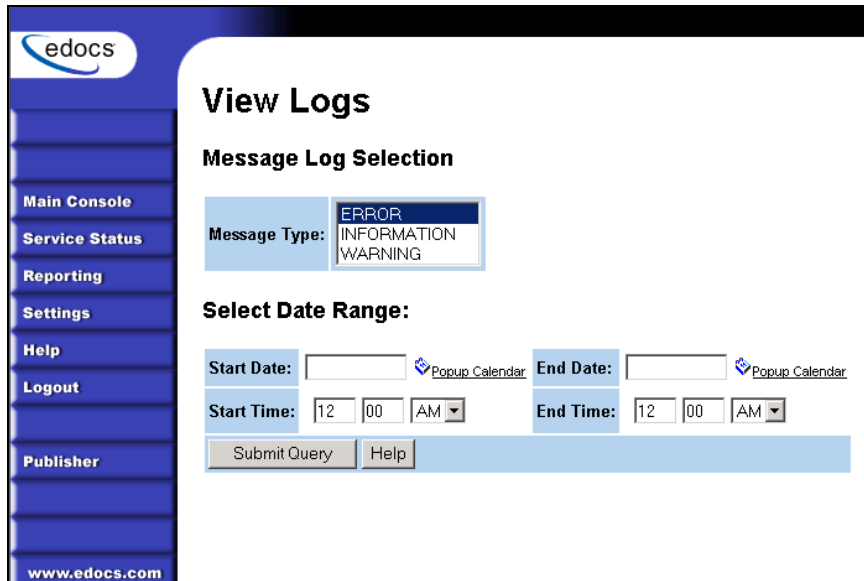
Log Report Column	Description
Timestamp	The date and time the message was created in the log
SourceHost	Name of the server that generated the error message or where the production activity occurred
Message ID	A code identifying the eaDirect task where the error occurred and the level of error See Appendix B – Error Messages on Page 161 for a description of each message ID.
Message	Message text

To view production log messages:


1. Click **Reporting** on the Command Center menu. The Reporting and Logs Screen appears:

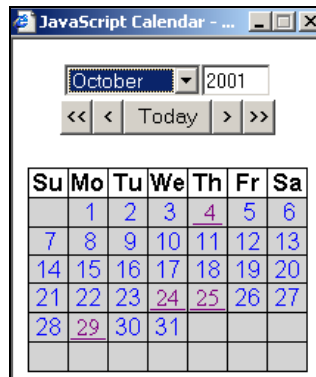


2. Click the **View Logs** tab to display the View Logs screen:



3. Select the type of message log to view.

4. Enter a start date and end date range to search. Click  **Popup Calendar** to select dates quickly:



5. Enter a start time and end time to search.
6. Click **submit Query**.
7. eaDirect displays the log messages of the selected type generated during the selected date and time range.

Timestamp	SourceHost	Message ID	Message
Thu Sep 20 16:55:45 EDT 2001	doc_poe/172.25.11.95	MON0003	Monitor created
Fri Sep 21 15:43:23 EDT 2001	doc_poe/172.25.11.95	SCH0003	PWC Scheduler started
Fri Sep 21 15:56:32 EDT 2001	doc_poe/172.25.11.95	SCH0003	PWC Scheduler started
Mon Sep 24 18:39:24 EDT 2001	doc_poe/172.25.11.95	MON0003	Monitor created
Mon Sep 24 18:39:27 EDT 2001	doc_poe/172.25.11.95	MON0003	Monitor created
Mon Sep 24 19:28:16 EDT 2001	doc_poe/172.25.11.95	ISC0004	IVNScanner.setTaskParams(EmailJob, Nat/Wireless, 1): {Job Name=EmailJob, DDN=Nat/Wireless, Task Name=IVNScanner, Task Order=1, Config Params={Param Name=Index Volume Status, values={Value=Accepted, Selected=true, Key=true}}{Value=Reject Selected=false, Key=false}, Required=true, Read Only=false, Multiples=false}
Mon Sep 24 19:28:16 EDT 2001	doc_poe/172.25.11.95	PTK0001	Created and Starting
Mon Sep 24 19:37:14 EDT 2001	doc_poe/172.25.11.95	PTK0001	Created and Starting
Mon Sep 24 19:37:34 EDT 2001	doc_poe/172.25.11.95	PTK0001	Created and Starting
Mon Sep 24 19:37:39 EDT 2001	doc_poe/172.25.11.95	ISC0004	IVNScanner.setTaskParams(EmailJob, Nat/Wireless, 1): {Job Name=EmailJob, DDN=Nat/Wireless, Task Name=IVNScanner, Task Order=1, Config Params={Param Name=Index Volume Status, values={Value=Accepted, Selected=true, Key=true}}{Value=Reject Selected=false, Key=false}, Required=true, Read Only=false, Multiples=false}
Mon Sep 24 19:37:39 EDT 2001	doc_poe/172.25.11.95	PTK0001	Created and Starting
Mon Sep 24 19:38:59 EDT 2001	doc_poe/172.25.11.95	ISC0004	IVNScanner.setTaskParams(EmailNotification, Nat/Wireless, 1): {Job Name=EmailNotification, DDN=Nat/Wireless, Task Name=IVNScanner, Task Order=1, Config Params={Param Name=Index Volume Status,

8. To select different log information to view, click **Reselect Log View**.

Viewing Job Output

You can view application output online to verify that your application and system are working properly.

You must first enroll through the eaDirect Enrollment Module; enrolling identifies each user to an online billing service provider. After you've enrolled successfully, you can log into the Enrollment Module and view a statement.

To enroll for online statement viewing:

1. Run the Indexer job and publish the design files you want to view in live retrieval.

2. After the Indexer job executes successfully, connect to the eaDirect Enrollment Module; enter the following URL in your browser, substituting your server and application names:
<http://<YourMachineName>:<PortNumber>/eaSample/User?app=UserMain&jsp=/user/jsp/HistoryList.jsp&ddn=<ApplicationName>>

(For testing purposes, you can view the NatlWireless application.)
eaDirect displays an enrollment login page.
3. Click the **Enroll Now** link. An empty subscription page appears.
4. Enter the customer number. (If you are viewing the National Wireless sample application, you can use sample customer account numbers 0331734, 4191463, or 8611250.)
5. Click **submit** to save the subscription information or **Reset** to clear the text fields.
6. Return to the Enrollment login page and enter your username (Subscriber ID) and password. This is the same username and password combination you entered during enrollment.
7. Click **submit**. A sample statement summary page appears.
8. To view details, click **view**.
9. Return to the statement summary page and click **Logout**.

Changing a Job Configuration

You change a job configuration any time, *except while the job is processing*.



If you try to save configuration changes while a job is running in the production queue (job status is "Processing") eaDirect ignores the new job configuration parameters.

To edit a job configuration:

1. On the Main Console, click the name of the job you want to reconfigure. eaDirect displays the job configuration screen.
2. Enter your changes. If you want to clear all current job parameters, click **Reset**.
3. Click **Submit Changes and Schedule**.
4. Click **OK**. eaDirect displays the Schedule screen, where you can edit the job schedule, if needed. See “Scheduling Jobs” on Page 97 for details.

Deleting a Job

You can delete a job you no longer need in an application. Deleting a job removes the job configuration and schedule in the Command Center.

Deleting a job does not remove data associated with the job that is already in the database. (Use Purge App and Purge Logs jobs to purge data.)

Make sure you really want to delete the job, and not make one of the following changes instead:

- To cancel and retry a *failed* job, see “Canceling and retrying failed jobs” on Page 106.
- To **change a job** configuration, see “Changing a Job Configuration” on Page 111.
- To **change a job schedule**, see “Scheduling Jobs” on Page 97.

To delete a job:

1. On the Main Console, click the name of the application. eaDirect displays the Edit Application screen, which lists the application jobs.

edocs

Edit Application: NatlWireless

Add New Job Delete Marked Jobs Help

Job Name	Job Type	Last Run	Status	Next Run	Delete?
DetailExtractor	Detail Extractor	None	Not scheduled	Not scheduled	<input type="checkbox"/>
EmailNotification	Email Notification	None	Not scheduled	Not scheduled	<input type="checkbox"/>
HTMLOutput	HTML Output	None	Not scheduled	Not scheduled	<input type="checkbox"/>
Indexer	Indexer	None	Not scheduled	Not scheduled	<input type="checkbox"/>
PurgeApp	Purge App	None	Not scheduled	Not scheduled	<input type="checkbox"/>
PurgeLogs	Purge Logs	None	Not scheduled	Not scheduled	<input type="checkbox"/>
XMLOutput	XML Output	None	Not scheduled	Not scheduled	<input type="checkbox"/>

www.edocs.com

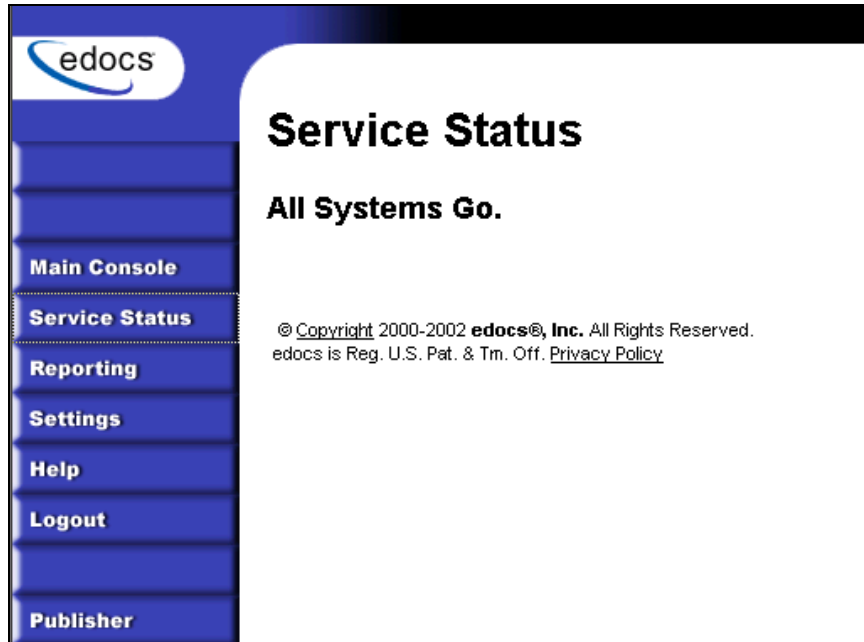
2. Click the box in the "Delete?" column for the job.
3. Click the **Delete Marked Jobs** button.
4. Click **OK** when asked if you are sure you want to delete the marked job. eaDirect removes the job from the application.

Monitoring System Services

You can check on the status of system services using the Command Center.

To view the status of system services:

1. Click **Service Status** on the Command Center menu. The Service Status page appears:



2. eaDirect indicates whether all services are running or which, if any, are missing.

If services are missing:

1. Close Command Center.
2. Shut down and restart WebLogic from your WLHome directory.

Display the Service Status again to verify that the problem has been corrected. If services are still missing, refer to your eaDirect Installation and Configuration guide.

Publishing and Using Version Sets

4

When to Publish New Version Sets

As part of the initial process of setting up your eaDirect application, you published the original design files that present an application's statements and generate email notifications.

A version set is a dated set of design files which eaDirect uses to present a user with an online view of a statement. "Publishing" a version set using the eaDirect Publisher tool identifies each design file belonging to a particular view and moves the files from the design environment to your application server. You give the view a name, and Publisher timestamps the version set for further identification.

Your organization might occasionally find it necessary to modify application design files to apply new business logic, text messages, logos, advertisements, other elements of an online statement, or to accommodate a new data input file format. To implement the new design files in your production environment, you must publish the files, creating new version sets.

It is only necessary to publish design files again for an application if you update the files for any reason. You cannot update or overwrite an existing version set; you must create and publish a new one.

If you create a new version set to accommodate changes in the application data input file, you must also publish a new Indexer version set (DDF).

When you publish a new version set for the view, eaDirect proceeds to use the new version set with Indexer jobs until you publish another one. Publishing a new version set on the same day you run the Indexer job makes it easier to keep track of when design changes take effect.

It is possible to publish multiple version sets on the same day an Indexer job is run, or on subsequent days. See “How eaDirect Determines which Version Set to Use for Live Retrieval” on Page 135 for details.

Publishing a New Version Set

Each time you publish a new version set, eaDirect:

- Creates a timestamp version directory called AppName/ViewType/ViewName/Timestamp
- Adds each file in the version set to the directory
- Creates a reference in the database for the version set

For details on what files you must publish for each type of job, see “What Files Do I Need to Publish?” on Page 26.

If you publish a new dynamic version set with changes to the DDF, you must also publish the new DDF with the Indexer job if the modifications to the DDF were to accommodate changes to the application’s data input file layout.

Note that you do not need to publish files for Purge App and Purge Logs jobs.

To publish a new dynamic version set:

- See “Publishing your application's dynamic Web views” on Page 27 for instructions.

To publish a new version set (DDF) for an Indexer job:

1. Click **Publisher** on the Command Center Main Console.
2. Click **create**. Publisher displays the Select a Version Set Type screen.

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

Dynamic Web Views

Job Type	Number of Auxiliary Files
HTML	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 Output	0

- Under Batch jobs, click 0 next to Indexer. eaDirect displays the Create a Version Set for Indexer screen:

The screenshot shows a web interface for edocs. On the left is a blue sidebar with the edocs logo at the top and a list of navigation buttons: Browse, Create, Fetch, Delete, Help, and www.edocs.com at the bottom. The main content area is white and titled 'Create a version set for Indexer'. It contains the following fields and controls:

- Application:** A drop-down menu with 'NatlWireless' selected.
- View Name:** A text input field containing 'Indexer'.
- DDF File:** A text input field followed by a 'Browse...' button.
- Below the DDF File field are three buttons: 'Submit', 'Clear', and 'Help'.
- At the bottom of the main area is a copyright notice: '© Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)'.

4. Select the application from the drop-down list.
5. Specify the DDF file to publish for the Indexer job.
6. Click **submit**. Publisher displays the Submission screen with details about the DDF:

edocs

Submission

Application: NatlWireless
View Type: DOC_CONFIG
View Name: Indexer
Timestamp: Mon Apr 22 16:32:53 EDT 2002

This version set contains the following files:

[NatlWireless/DOC_CONFIG/Indexer/20020422163253/NatlWireless.ddf](#)

© Copyright 2000-2002 edocs®, Inc. All Rights Reserved.
edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)

[www.edocs.com](#)

To publish a new EmailNotification version set:

1. Click Publisher on the Command Center Main Console.
2. Click **create**. Publisher displays the Select a Version Set Type screen.

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

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

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 Output	0

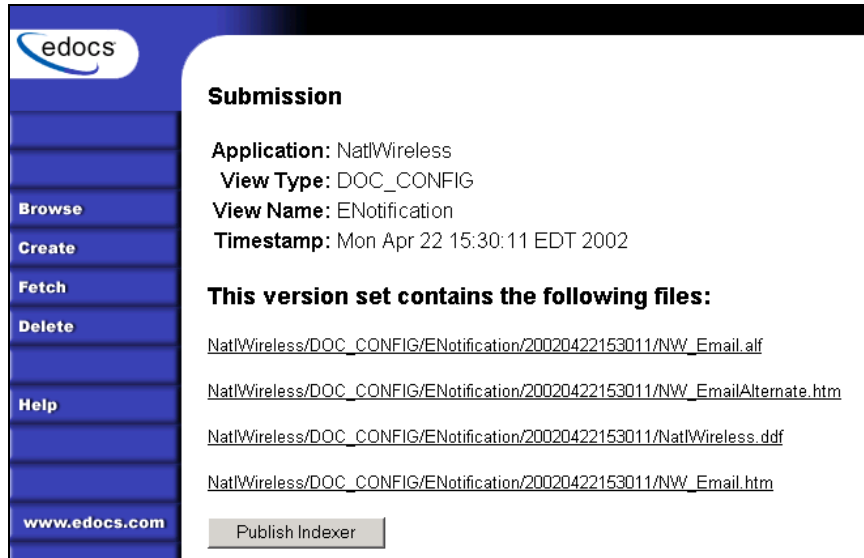
- Under Batch Jobs, next to EmailNotification, click the number of auxiliary HTML templates the view uses (the number in addition to the default template). Publisher displays the Create a Version Set for Email Notification screen:

The screenshot shows a web interface for edocs. On the left is a blue sidebar with buttons: Browse, Create, Fetch, Delete, Help, and www.edocs.com. The main content area is titled 'Create a version set for Email Notification'. It contains a form with the following fields and controls:

- Application:** A drop-down menu with 'Please select' as the current selection.
- View Name:** A text field containing 'ENotification'.
- DDF File:** A text input field followed by a 'Browse...' button.
- ALF File:** A text input field followed by a 'Browse...' button.
- HTML Template:** A text input field followed by a 'Browse...' button.
- Auxiliary File:** A text input field followed by a 'Browse...' button.
- At the bottom of the form are three buttons: 'Submit', 'Clear', and 'Help'.

At the bottom of the page, there is a copyright notice: © Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)

4. Select the application from the drop-down menu.
5. Specify the DDF, ALF, and HTML files in the EmailNotification version set.
6. Click **submit**. Publisher displays the Submission screen with details about the version set:



The screenshot shows the 'edocs' interface. On the left is a blue navigation menu with buttons for 'Browse', 'Create', 'Fetch', 'Delete', 'Help', and 'www.edocs.com'. The main content area is titled 'Submission' and displays the following information:

- Application:** NatlWireless
- View Type:** DOC_CONFIG
- View Name:** ENotification
- Timestamp:** Mon Apr 22 15:30:11 EDT 2002

Below this, it states: **This version set contains the following files:**

- [NatlWireless/DOC_CONFIG/ENotification/20020422153011/NW_Email.aif](#)
- [NatlWireless/DOC_CONFIG/ENotification/20020422153011/NW_EmailAlternate.htm](#)
- [NatlWireless/DOC_CONFIG/ENotification/20020422153011/NatlWireless.ddf](#)
- [NatlWireless/DOC_CONFIG/ENotification/20020422153011/NW_Email.htm](#)

At the bottom of the main content area is a 'Publish Indexer' button.

To publish a new HTML Output version set:

1. Click **Publisher** on the Command Center Main Console.
2. Click **Create**. Publisher displays the Select a Version Set Type screen.

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

Dynamic Web Views

Job Type	Number of Auxiliary Files
HTML	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 Output	0

- Under Batch Jobs, next to HTML Output, click the number of auxiliary HTML templates the view uses (the number in addition to the default template). Publisher displays the Create a Version Set for HTML Output screen:

The screenshot shows the edocs web interface. On the left is a blue navigation menu with the following items: edocs, Browse, Create, Fetch, Delete, and Help. The main content area is titled "Create a version set for HTML Output". It contains the following fields and controls:

- Application:** A drop-down menu with "Please select" as the current selection.
- View Name:** A text field containing "HtmlDetail".
- DDF File:** A text field with a "Browse..." button to its right.
- ALF File:** A text field with a "Browse..." button to its right.
- HTML Template:** A text field with a "Browse..." button to its right.
- Below the fields are three buttons: "Submit", "Clear", and "Help".

At the bottom of the form area, there is a copyright notice: "© Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)". The URL "www.edocs.com" is displayed at the bottom left of the page.

4. Select the application from the drop-down menu.
5. Specify the DDF, ALF, and HTML files in the HTML Output job version set.
6. Click **submit**. Publisher displays the Submission screen with details about the version set:

edocs

Submission

Application: NatlWireless
View Type: DOC_CONFIG
View Name: HtmlDetail
Timestamp: Mon Apr 22 16:28:35 EDT 2002

This version set contains the following files:

[NatlWireless/DOC_CONFIG/HtmlDetail/20020422162835/NatlWireless.alf](#)
[NatlWireless/DOC_CONFIG/HtmlDetail/20020422162835/NatlWireless.ddf](#)
[NatlWireless/DOC_CONFIG/HtmlDetail/20020422162835/NatlWireless.htm](#)

[www.edocs.com](#)

© Copyright 2000-2002 **edocs**®, Inc. All Rights Reserved.
edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)

7. Click **Publish Indexer** to publish a DDF file for the Indexer task that runs as part of the HTML Output job. eaDirect displays the Create Version Set for Indexer screen:

The screenshot shows a web interface for 'edocs'. On the left is a vertical blue navigation menu with buttons for 'Browse', 'Create', 'Fetch', 'Delete', and 'Help'. The main content area is titled 'Create a version set for Indexer'. It contains the following fields and controls:

- Application:** A drop-down menu currently showing 'NatlWireless'.
- View Name:** A text input field containing 'Indexer'.
- DDF File:** A text input field followed by a 'Browse...' button.
- Below the DDF File field are three buttons: 'Submit', 'Clear', and 'Help'.

At the bottom of the main content area, there is a copyright notice: © Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#). The URL 'www.edocs.com' is visible at the bottom of the navigation menu.

7. Select the application from the drop-down list.
8. Specify the DDF file for the HTML Output job.
9. Click **submit**. Publisher displays the Submission screen with details about the DDF:

The screenshot shows the edocs web interface. On the left is a blue sidebar with the edocs logo at the top and a list of navigation buttons: Browse, Create, Fetch, Delete, Help, and www.edocs.com at the bottom. The main content area is white and displays the following information:

Submission

Application: NatlWireless
View Type: DOC_CONFIG
View Name: Indexer
Timestamp: Mon Apr 22 16:32:53 EDT 2002

This version set contains the following files:

[NatlWireless/DOC_CONFIG/Indexer/20020422163253/NatlWireless.ddf](#)

© Copyright 2000-2002 edocs®, Inc. All Rights Reserved.
edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)

To publish a new DDF file for XML Output:

1. Click Publisher on the Command Center Main Console.
2. Click **create**. Publisher displays the Select a Version Set Type screen.

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

Dynamic Web Views

Job Type	Number of Auxiliary Files
HTML	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 Output	0

- Under Batch Jobs, next to XML Output, click 0. Publisher displays the Create a Version Set for XML Output screen:

The screenshot shows a web interface for edocs. On the left is a vertical navigation menu with buttons for Browse, Create, Fetch, Delete, and Help. The main content area is titled "Create a version set for XML Output". It contains the following fields and controls:

- Application:** A drop-down menu currently showing "Please select".
- View Name:** A text field containing "XmlDetail".
- DDF File:** A text input field followed by a "Browse..." button.
- Below the DDF File field are three buttons: "Submit", "Clear", and "Help".

At the bottom of the main content area, there is a copyright notice: "© Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)".

4. Select the application from the drop-down menu.
5. Specify the DDF file for the XML Output job.
6. Click **submit**. Publisher displays the Submission screen with details about the DDF:

The screenshot shows the edocs web interface. On the left is a blue navigation sidebar with the edocs logo at the top and buttons for Browse, Create, Fetch, Delete, and Help. The main content area is white and titled 'Submission'. It displays the following information: Application: NatWireless, View Type: DOC_CONFIG, View Name: XmlDetail, and Timestamp: Mon Apr 22 16:39:08 EDT 2002. Below this, it states 'This version set contains the following files:' followed by a file path: NatWireless/DOC_CONFIG/XmlDetail/20020422163908/NatWireless.ddf. A 'Publish Indexer' button is located below the file path. At the bottom of the page, there is a copyright notice: '© Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. Privacy Policy'. The website address 'www.edocs.com' is visible in the bottom left corner of the page.

7. Click **Publish Indexer** to publish the DDF for the Indexer task that runs as part of the XML Output job. eaDirect displays the Create Version Set for Indexer screen:

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

- Application:** A dropdown menu currently showing "NatlWireless".
- View Name:** A text input field containing "Indexer".
- DDF File:** A text input field followed by a "Browse..." button.
- Below the DDF File field are three buttons: "Submit", "Clear", and "Help".

At the bottom of the main content area, there is a copyright notice: "© Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)".

8. Specify the DDF file for the XML Output job's Indexer task.
9. Click **submit**. Publisher displays the Submission screen with details about the DDF:

edocs

Submission

Application: NatlWireless
View Type: DOC_CONFIG
View Name: Indexer
Timestamp: Mon Apr 22 16:32:53 EDT 2002

This version set contains the following files:

[NatlWireless/DOC_CONFIG/Indexer/20020422163253/NatlWireless.ddf](#)

© Copyright 2000-2002 **edocs**, Inc. All Rights Reserved.
edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)

[www.edocs.com](#)

To publish a new Detail Extractor version set:

1. Click Publisher on the Command Center Main Console.
2. Click **create**. Publisher displays the Select a Version Set Type screen.

edocs

Select a Version Set Type

Dynamic Web Views

Job Type	Number of Auxiliary Files
HTML	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 Output	0

Browse
Create
Fetch
Delete
Help

www.edocs.com

- Under Batch Jobs, next to Detail Extractor, click 0. Publisher displays the Create a Version Set for Detail Extractor screen:

edocs

Create a version set for Detail Extractor

Application:

View Name:

DDF File:

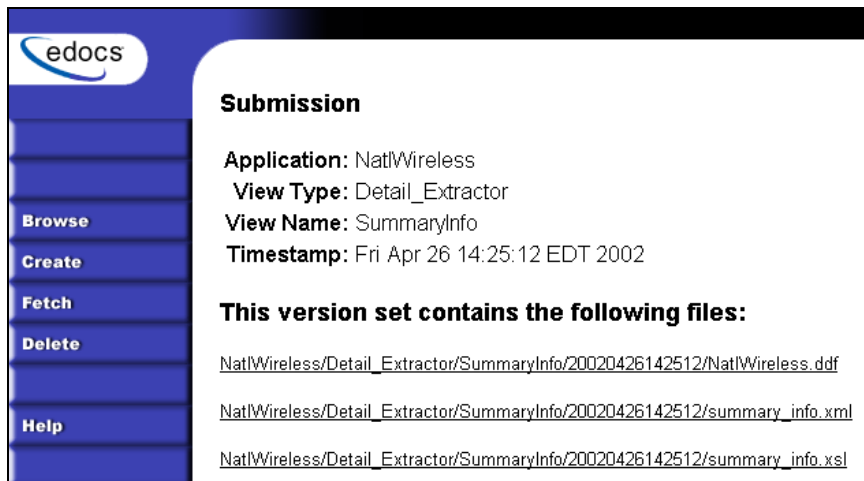
Database Table XML File:

Statement XSLT File:

© Copyright 2000-2002 edocs®, Inc. All Rights Reserved.
edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)

Browse
Create
Fetch
Delete
Help

4. Select the application from the drop-down list.
5. Enter a name for the view. (It can be helpful to use the DDF name as the view name.)
6. Specify the DDF file for the Detail Extractor's StatementsToIR task.
7. Specify the database table schema XML file.
8. Specify the statement XSLT style sheet.
9. Click **submit**. Publisher displays the Submission screen with details about the files you just published:



How eaDirect Determines which Version Set to Use for Live Retrieval

If you've published multiple version sets of a dynamic view, eaDirect must determine which associated version set to use for live statement retrieval.

When a user retrieves a statement, eaDirect determines which published version set to use according to the following rules:

1. **Last version set published on the day the data file was indexed** - eaDirect first looks for and uses a version set published the same day the data file was indexed.

If more than one version set was published the day the index was created, eaDirect uses the set with the latest timestamp before midnight (up to and including 11:59:59) on that day. (eaDirect does not remember the time of day the index was created, only the date, therefore the time Indexer ran is not a factor in determining which version set to use; eaDirect always uses the version set published latest in the day.)

2. **Most recent version set published before the day the data file was indexed** - If eaDirect does not find a version set published the same day the data file was indexed, it looks for the most recent version set published before the date the data file was indexed.
3. **Next version set published after the day the data file was indexed** - If eaDirect does not find a version set published on or before the date the data file was indexed, it looks for the next version set published after 00:00:00 the next day, looking forward for the first available version set.

Example: One Version Set Published on the Same Day Indexer Ran

If you publish a dynamic web view version set on 8/7 at 10:00 am, and run the Indexer job at 11:00 am, when a user retrieves the 8/7 statement the next day, eaDirect uses the version set published on 8/7 at 10:00 am.

Example: Multiple Version Sets Published on the Same Day

If you publish a version set on 8/7 at 10:00 am, run the Indexer job at 11:00 am, then publish another version set at 3:00 pm, the next day when a user retrieves the 8/7 statement, eaDirect uses the 3:00 pm version set, since it was the latest set published on the day the index was created. The time of day the Indexer job ran does not determine which version set eaDirect uses; it uses the latest set created on that day.

Examples: Multiple Version Sets Published on Different Days

A. If a user retrieves a statement with an index date of 10/30, and the associated version sets are dated 10/25, 10/26, 10/31 eaDirect uses the 10/26 version set.

B. If a user retrieves a statement with an index date of 10/30, and the only associated version sets are dated 11/01 and 11/08, eaDirect uses the 11/01 set.

Browsing for Version Sets

Publisher lets you browse the production environment to display a list of version sets for a view. You must specify the application name, job type, view type, and view name.

Note that although you can see a list of version sets, there is no immediate way of knowing what is different about each set other than the timestamp. Some version sets could extract new data, or have changes to business logic, text, advertisements, etc. You can look at a file directly to determine what it does; see “Viewing the Contents of a Design File” on Page 138.

To browse for a list of the version sets published for a job:

1. Click **Browse** on the Publisher menu. The Browse Version History screen appears.
2. From the drop-down menus, choose the application, job type of the version sets you want to view, view type, and view name.

- Click **Browse Version History**. If the search is successful, Publisher displays the Version History screen listing each file in the version sets as links. If more than one Version Set is found, eaDirect uses alternating colors to differentiate them in the list. To view the content of a design file, click the design file name link on the Version History screen:

Version History

Application	View Type	View Name	Timestamp	File
NatlWireless/HTML/HTMLDetail/20020422165653/				NatlWireless.alf
NatlWireless/HTML/HTMLDetail/20020422165653/				NatlWireless.ddf
NatlWireless/HTML/HTMLDetail/20020422165653/				NatlWireless.htm
NatlWireless/HTML/HTMLDetail/20020422171714/				NatlWireless.alf
NatlWireless/HTML/HTMLDetail/20020422171714/				NatlWireless.ddf
NatlWireless/HTML/HTMLDetail/20020422171714/				NatlWireless.htm

© Copyright 2000-2002 edocs®, Inc. All Rights Reserved.
edocs is Reg. U.S. Pat. & Tm. Off. [Privacy Policy](#)

If the search is unsuccessful, Publisher displays the message “There is no version set available to browse.”

Maintaining Old Version Sets for Historical Statements

eaDirect maintains a copy of each version set you publish; it does not replace or delete version sets. This lets you maintain a chronological history of application design modifications over time.

Maintaining old version sets also creates a “snapshot” of a statement, freezing the original presentation to enable users to view it unchanged in the future. When a user retrieves an old, or historical statement, eaDirect displays the data for that month’s statement using the version set that was in effect when the data file was indexed.

Caution Leave old version sets in place to enable the display of historical statements with the original formatting.

For example, if on June 20th, a user retrieves a statement for April 20th, that statement would appear just as it did when originally viewed in April. No data or design changes made to the application after April 20th would apply to the April statement, regardless of when it is viewed. As long as you leave the original version set in place, the original format and content of the April statement remain unchanged.

Because eaDirect looks for a version set using the logic described above, it means you cannot make adjustments to a version set published and used with an old statement. For example, if you have version 2.0 of an application in production, and found issues with your version 1.0 application statements, you cannot make adjustments to the version 1.0 version set retroactively. You would only have viewing access to the “snapshot in time” of the version 1.0 statements. If you were to delete the version set published for version 1.0, eaDirect would look for another version set in accordance with the logic described in this chapter, and inevitably use a version set not intended for that statement, likely displaying it improperly.

Viewing the Contents of a Design File

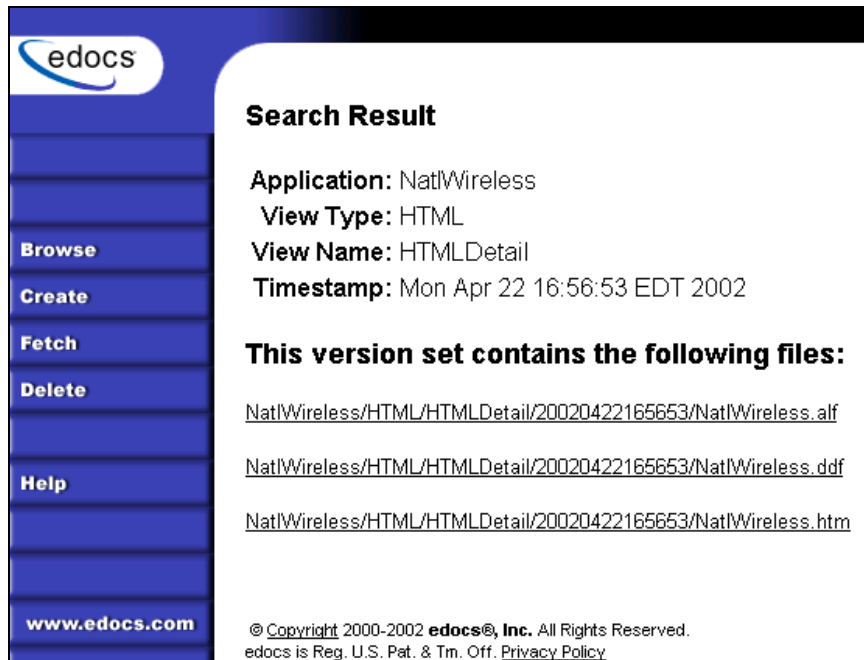
You can use the Command Center to display the ASCII content of eaDirect ALF or XML design files, or display an HTM file as a static HTML page. (To view the content of a DDF file you must use eaDirect’s DefTool.)

Viewing the contents of a design file lets you determine

- What it is designed to extract or present.
- What modifications were made between different versions of a file (between version sets), such as changes to text, business logic, advertisements, etc.

To view the content of an eaDirect design file:

1. Browse for the design file using Publisher. See “Browsing for Version Sets” on Page 136.
2. Click the name of the design file you want to view. eaDirect displays the Search Result screen listing the files in the version set:



3. Click the name of the file you want to view. eaDirect displays the file contents:

```

<?xml version="1.0" ?>
<!DOCTYPE ALF (View Source for full doctype...)>
- <ALF>
  <VERSION>2.20</VERSION>
  <DATA_GROUP>NatlWireless</DATA_GROUP>
  <DDF>C:\EDCSbd\Samples\NatlWireless\NatlWireless.ddf</DDF>
  <SWITCH />
- <HOME>
  <DefaultTemplate>Default_Template</DefaultTemplate>
</HOME>
- <TEMPLATES>
  - <Template Name="Default_Template">
    - <SECTIONS>
      - <Section Name="HeaderMsgTxt" Promotional="N">
        - <Statement>
          - <Condition>
            <![CDATA[ (FirstStmIndicator) ]]>
          </Condition>
          <Action1>NewCustMsg</Action1>
        - <Action2>
          - <Statement>
            - <Condition>
              <![CDATA[ (LastStmIndicator) ]]>
            </Condition>
            <Action1>LastStmMsg</Action1>

```

Deleting a Version Set

In a normal production environment, you would *never* want to delete a version set. If you accidentally delete a version set, there is *no* recovery plan in place.

If you delete the only dynamic Web view version set associated with an indexed data file, you would no longer be able to view those documents.

If there are several version sets of a view and you delete the latest one, eaDirect uses the view published previous to the version set you deleted, which can result in the online statement being formatted incorrectly.

The only possible reasons you might want to delete a version set is when you know you made a mistake in publishing it, you are archiving old data (which removes references to the statements in the index) and the version set is associated with that old data, or you are working in a test environment.

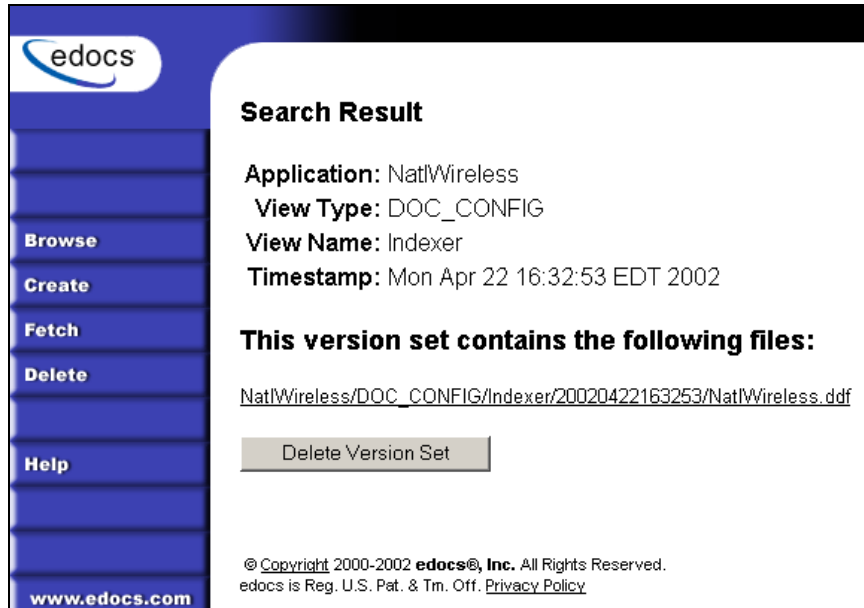
Be extremely careful if you choose to delete a version set for any reason.

Deleting a version set deletes the version set files and any references to the version set in the database.

To delete a version set:

1. Click **Delete** on the Publisher menu. The Delete a Version Set screen appears:

2. Choose the application and job type associated with the version set you want to delete.
3. Click **Yes** if you want to delete the latest version set, or enter a date to display the last version set published before that day.
4. Click **Search**. If Publisher finds the version set, it displays the list of the design files in the set.



The screenshot shows the edocs web interface. On the left is a blue sidebar with the edocs logo at the top and a menu with buttons for Browse, Create, Fetch, Delete, and Help. The main content area is white and displays search results for a version set. The results include the application name (NatWireless), view type (DOC_CONFIG), view name (Indexer), and timestamp (Mon Apr 22 16:32:53 EDT 2002). Below this, it states 'This version set contains the following files:' followed by a file path: NatWireless/DOC_CONFIG/Indexer/20020422163253/NatWireless.ddf. A 'Delete Version Set' button is positioned below the file path. At the bottom of the page, there is a copyright notice: '© Copyright 2000-2002 edocs®, Inc. All Rights Reserved. edocs is Reg. U.S. Pat. & Tm. Off. Privacy Policy'.

5. Click **Delete Version Set**.

eaDirect prompts you to verify that you want to delete the version set. Click **OK**. eaDirect lets you know if the deletion was successful.

Viewing Production Statistics

The eaDirect Command Center lets you create and view reports showing statistical information about individual user activity and the general level of statement presentment requests your applications receive.

This section describes how to use the Command Center to create each type of eaDirect report:

- User Statistics
- Bill Statistics

Note that data from versions of eaDirect prior to 3.0 are not included in the User Statistics and Bill Statistics reports.

User Statistics

You can create and view reports showing detailed information about each time an enrolled user views a statement online. eaDirect lets you search the production database for information about one or all enrolled users, for one or all applications you have, over a particular period of time.

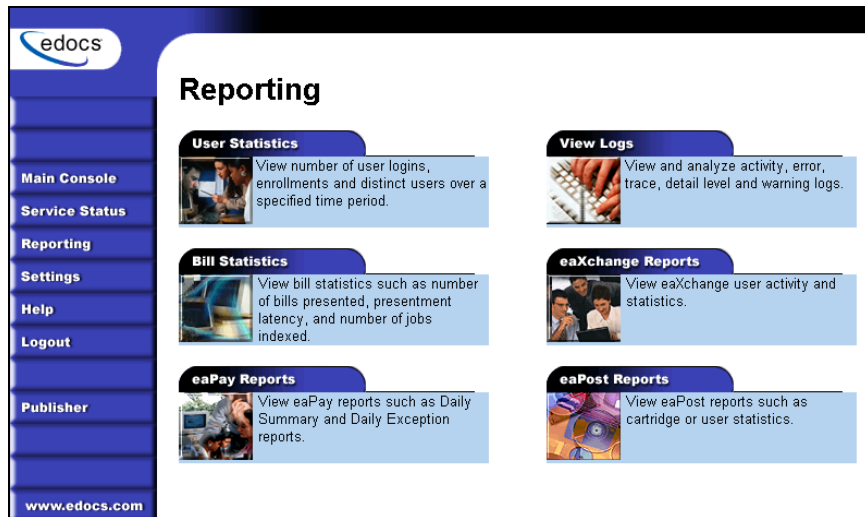
User Statistics reports let you monitor the specific statement viewing activities and habits of individual users.

eaDirect displays the following information on a User Statistics report:

User Statistic	Description
User ID	The ID of the user who viewed their statement (the name of the user who enrolled)
Processed	The time Indexer processed the statement data
Notification	Indicates whether the enrolled user received an email notification regarding the statement
Login	Date and time the user logged in
Bill Viewed	Date and time the user requested to view the statement

To view user statistics:

1. Click **Reporting** on the Command Center menu. The Reporting screen appears:




2. Click the **User Statistics** tab. The User Statistics screen appears:

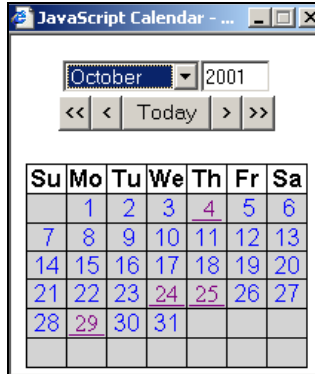
3. Select an application from the drop-down menu.
4. To search for history and statistics on a particular user, enter the user's ID. To see information about all users who have viewed statement during particular time period, leave the User ID field blank.



If you don't enter a user ID, the search can take a long time to finish and result in a large data set if you have a large volume of users and traffic. Set a small date and time range to narrow the search.

5. Specify a start and end date range to search. Click  **Popup Calendar** to select dates quickly:

Reviewing System Activity



6. Specify a start time and end time range to search.
7. Click **Submit Query**.
8. eaDirect displays the application's user statistics for the selected date and time range.

User Statistics: NatlWireless

UserID	Processed	Notification	Login	Bill Viewed
0331734	None	None	Wed Sep 26 17:01:52 EDT 2001	None
0331734	Wed Sep 26 16:57:30 EDT 2001	None	Wed Sep 26 17:01:52 EDT 2001	Wed Sep 26 17:02:08 EDT 2001
0331734	Wed Sep 26 16:57:30 EDT 2001	None	Wed Sep 26 17:01:52 EDT 2001	Wed Sep 26 17:04:11 EDT 2001
0331734	None	None	Wed Sep 26 17:14:21 EDT 2001	None
0331734	Wed Sep 26 16:57:30 EDT 2001	None	Wed Sep 26 17:14:21 EDT 2001	Wed Sep 26 17:14:32 EDT 2001
0331734	Wed Sep 26 16:57:30 EDT 2001	None	Wed Sep 26 17:14:21 EDT 2001	Wed Sep 26 17:14:47 EDT 2001
0331734	Wed Sep 26 16:57:30 EDT 2001	None	Wed Sep 26 17:14:21 EDT 2001	Wed Sep 26 17:16:27 EDT 2001
8611250	None	None	Thu Sep 27 14:06:39 EDT 2001	None
8611250	Wed Sep 26 16:57:30 EDT 2001	None	Thu Sep 27 14:06:39 EDT 2001	Thu Sep 27 14:06:54 EDT 2001
8611250	Wed Sep 26 16:57:30 EDT 2001	None	Thu Sep 27 14:06:39 EDT 2001	Thu Sep 27 14:08:55 EDT 2001
8611250	None	None	Thu Sep 27 14:13:51 EDT 2001	None
8611250	Wed Sep 26 16:57:30 EDT 2001	None	Thu Sep 27 14:13:51 EDT 2001	Thu Sep 27 14:14:00 EDT 2001

Copyright © 2000, 2001 edocs, Inc. All rights reserved. [Privacy Policy](#)

9. Click **Search Again** to perform another search for user statistics.

Bill Statistics

You can create and view reports showing summary information about how many times users log in and view their statements. eaDirect lets you search the production database for information about statement creation and presentment in one or all applications you may have over a particular time period.

Bill Statistics reports let you monitor the general access volume your applications receive.

You can display a standard ASCII-formatted report or view the statistics graphically in chart format.

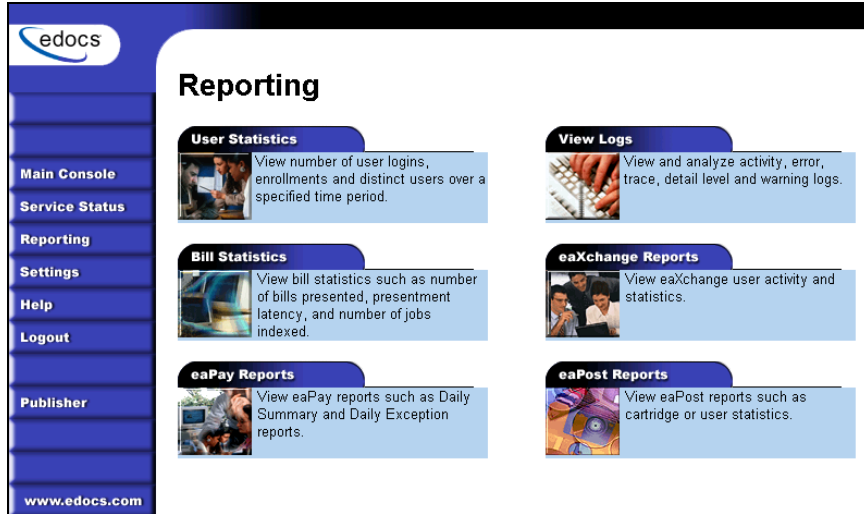
eaDirect displays the following information on a standard Bill Statistics report:

Statistic	Description
Number of Logins	The total number of times users logged in during the selected time period
Number of Unique Users	The number of users who logged in during the selected time period (does not count repeated logins by the same user)
Number of Bills Processed	The number of statements processed during all Indexer tasks run during the selected time period.
Number of Bills Viewed	The number of statements viewed during the selected time period (does not count repeated views)

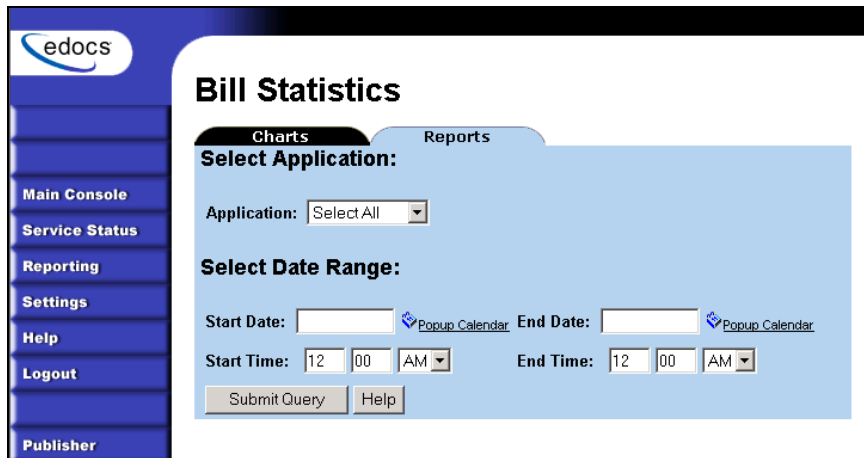
You can optionally display this report as a bar chart.

To view statement statistics:

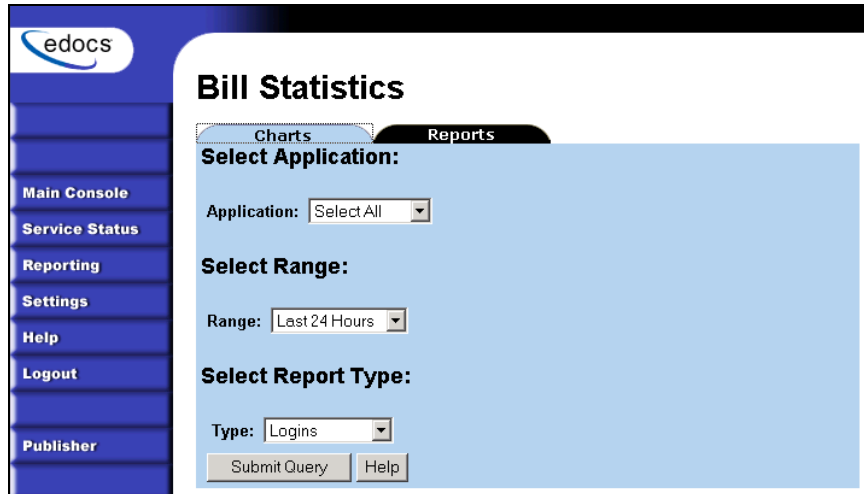
1. Click **Reporting** on the Command Center menu. eaDirect displays the Reporting screen:




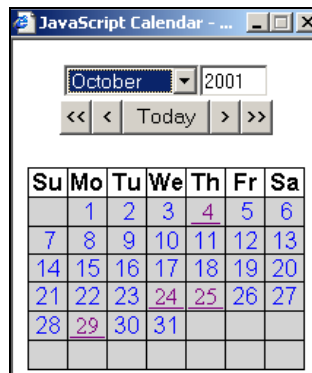
2. Click the **Bill Statistics** tab to display the Bill Statistics screen. The standard Reports option tab appears by default:



3. If you want the report in the form of a chart, click the **Charts** tab:



4. Select an application from the drop-down menu.
5. For standard reports, enter a start and end date, and a start and end time range to search. Click  **Popup Calendar** to select dates quickly. For a chart, select a date range to search from the drop-down list. Your search can be as recent as 24 hours, or as far back as 12 months. Select a report type; you can view number of user logins, unique users, bills processed, or bills viewed.



6. Click **Submit Query** to start the search.

7. eaDirect displays the application's statistics report or chart for the selected dates:

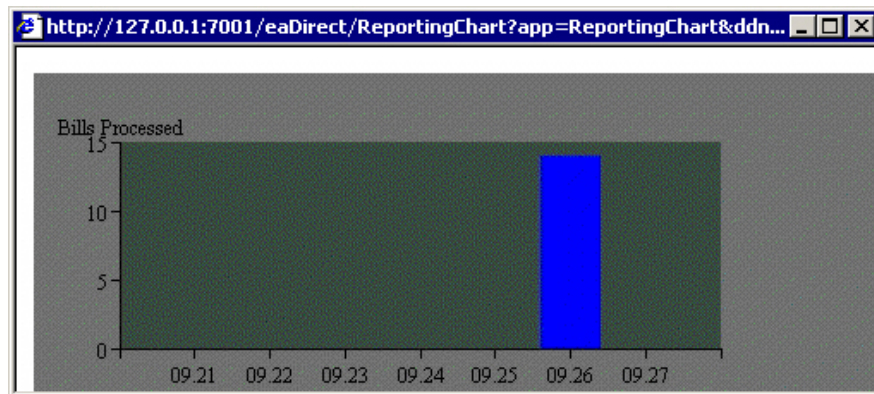
Bill Statistics: NatlWireless

Start Time: Tue Sep 25 00:00:00 EDT 2001

End Time: Fri Sep 28 00:00:59 EDT 2001

Search Again Help

Report	Results
# of Logins	4
# of Unique Users	2
# of Bills Processed	14
# of Bills Viewed	8



8. Click **Search Again** to perform another search for statement statistics.

Database Administration

Running an eaDirect application in a live production environment can generate a large volume of historical data in an application's database. You are responsible for monitoring and maintaining your own database server.

It is a good idea to monitor your database server on a weekly or other regular basis to:

- Check database utilization; to periodically eliminate older application data and free up space on your database server, create, configure, and run the Purge App and Purge Logs jobs, described here.
- Check memory utilization on SQL server/swap (paging) file utilization. When peak number of Commit Charge gets to 10% of limit, then it is advisable to increase the size of paging file, or install more RAM.
- Backup

Purging Historical Application Data; Purge App Job

Create and configure the Purge App job to periodically remove index, email, reporting, detail, annotation, and dispute data from the database. Running the Purge App job frees up space on your database server.

If your application calls for a certain number of month's statements to be included in a statement summary page, you must regularly run a Purge App job to eliminate index references to old statement data to prevent it from being included in summary page indexed fields.

If you upload detail to the database using the Detail Extractor job, or track annotations and disputes, run Purge App to remove this data as well.

To prevent older statements from getting included in the a summary screen, you must

- Run a Purge Apps job to remove the index references to the data input files that fall outside a specified date range (to limit user access to historical statements).

- Remove the data input file (if not accessing from primary storage/backup location).

You can also use Purge Logs to clean up a testing or QA machine that has logged data you don't want or need.

For details, see “Creating and Configuring a Purge App Job” on Page 57.

Purging Historical Log Data; Purge Logs Job

Create and configure a Purge Logs job to periodically remove old messages from the log tables in the database (for all applications). Over time, log tables can grow significantly in size. Running a Purge Logs job frees up space on your database server.

You can also use Purge Logs to clean up a testing or QA machine that has logged data you don't want or need.

For details, see “Creating and Configuring a Purge Logs Job” on Page 60.

Other System Administration Activities

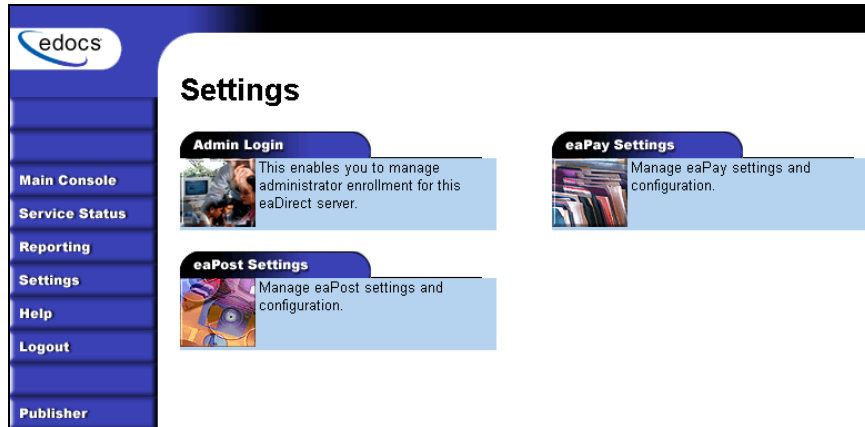
Changing the eaDirect Administrator's Password

You can change the eaDirect Administrator's password, which you use to log into the eaDirect Command Center, at any time.

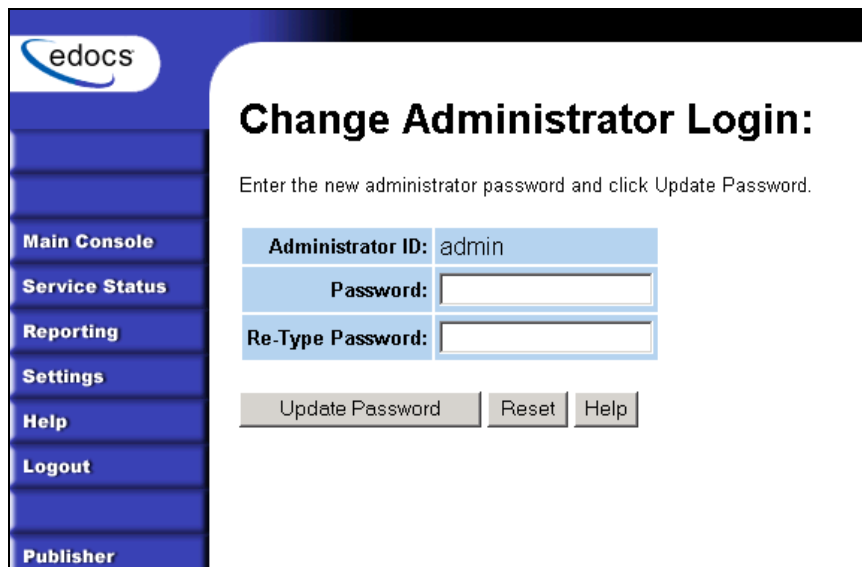
It's a good idea to periodically change the password to ensure system security.

To change the eaDirect Administrator's password:

1. On the Command Center Main Console, click **settings**. eaDirect displays the Settings screen:



2. Click the **Admin Login** tab to display the Change Administrator Login page.



3. Enter the new password in the Password field.
4. Enter the password again in the Re-Type Password field to confirm. Click Reset to clear the fields, if necessary.

5. Click **Update Password**.

Deleting an Old Application

If you have an old or unusable application, you can use the Command Center to delete it from your system.

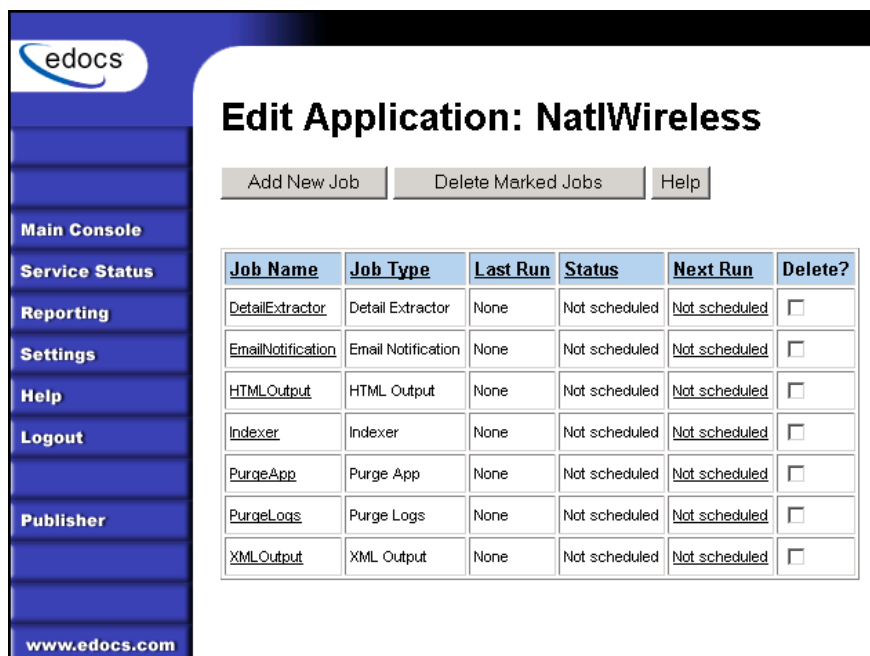
If you are testing in a quality assurance or development environment, you might want to remove an unneeded application.

You would not normally need to delete an application from a production environment, unless you misconfigured an application or set up the training application.

Note that deleting an application only removes it from use; it does not delete any associated data in the database for jobs that were run. To delete data from the database, run Purge App and Purge Logs jobs. See “Database Administration” on Page 151.

To delete an old application:

1. Delete all jobs associated with an application. You can't delete an application until you've deleted all the related jobs.
2. At the Main Console, click the name of the application you want to delete. eaDirect displays the Edit Application screen:



3. If any jobs still exist, click the box in the "Delete?" column for each job, then click **Delete Marked Jobs**. Click **OK** when asked if you are sure you want to delete the marked jobs.
4. Click the **Remove Application** button (which only appears when no jobs are listed).
5. Click **OK** when asked if you are sure you want to delete the application. eaDirect deletes the application.

Appendix A: Troubleshooting

Version Sets

Version Set Problem	Solution
You are defining a job, and a file path reads “Unable to Determine File Path”	A version set has not been defined; publish a version set.
The Indexer version set is incorrect and the problem affects indexed values.	Delete the incorrect index references. Fix the DDF problem. Publish a new (correct) version set. Run Indexer against the new version set.
The Indexer version set is incorrect and the problem does not affect indexed values.	Fix the DDF problem. Publish a new (correct) version set.
You deleted the most recent version set, then index, and the job used an old version set.	System looks for and uses most recent version set. If no version set of required type exists, eaDirect generates an error. Publish a new version set.
You published the wrong or incorrect files.	Delete the incorrect version set. Publish a new version set. Consider the timing of publishing a new version set and ongoing job processing.
You published a version set to the wrong application.	Delete the incorrect version set. Publish the version set to the correct application. Consider the timing of publishing a new version set and ongoing job processing.

Version Set Problem	Solution
You published the wrong type of version set (i.e. static HTML job instead of dynamic web view)	Delete the incorrect version set. Publish the correct version set.
The view name as you defined when publishing the version set is not the exact call from the JSP.	Delete the incorrect version set. Publish a version set with the correct view name.
Auxiliary file(s) are missing.	Delete the version set and republish with all files or binary FTP/Copy file into correct directory.
When defining a version set, you selected a wrong file as required file.	Delete the version set and republish with the correct files. Or, if the wrong file and correct file have the same name, binary FTP copy the correct file over the wrong file in the version set directory.

Production Jobs

Production Problem	Solution
A lot of jobs are failing.	Check the connections between your database and application servers.

System

System Problem	Solution
You have trouble logging in for the first time after installing eaDirect.	Check the connections between your database and application servers.
System services are missing.	Close Command Center and shut down and restart WebLogic from your WLHome directory.

Appendix B: Error Messages

eaDirect Error Messages

APP is com.edocs.services.application.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
APP0001	Error	Error intializing app stored procedure call strings	Call for support
APP0002	Error	Unable to interpret the docid: {0}	Call for support
APP0003	Exception	Exception caught: {0}	Call for support

DFE is com.edocs.tasks.statements2ir.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
DFE0001	Exception	StatementsToIR.getTaskParams: Failed to get task config params - {0}	Call for support
DFE0002	Exception	StatementsToIR.getTaskParams: Failed to get task config params - {0}	Call for support
DFE0003	Exception	StatementsToIR.getTaskParams: Failed to get task config params - {0}	Call for support
DFE0004	Error	StatementsToIR.setTaskParams({0}, {1}, {2}): {3}	Call for support
DFE0005	Exception	StatementsToIR.setTaskParams: Failed to set task config params - {0}	Call for support
DFE0006	Exception	StatementsToIR.setTaskParams: Failed to	Call for support

Appendix B: Error Messages

DFE is com.edocs.tasks.statements2ir.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
		set task config params - {0}	
DFE0007	Exception	StatementsToIR.setTaskParams: Failed to set task config params - {0}	Call for support
DFE0008	Exception	StatementsToIR.isTaskConfigValid: Failed while validating config params - {0}	Call for support
DFE0009	Exception	StatementsToIR.isTaskConfigValid: Failed while validating config params - {0}	Call for support
DFE0010	Exception	StatementsToIR.isTaskConfigValid: Failed while validating config params - {0}	Call for support
DFE0011	Exception	StatementsToIR.isTaskConfigValid: Failed while validating config params - {0}	Call for support
DFE0012	Error	StatementsToIR.processTask({0}, {1})	Call for support
DFE0013	Error	StatementsToIR.processTask({0}, {1}) - Done; ivn: {2}; view: {3}; output path: {4}; validator: {5}; ir file: {6}	Call for support
DFE0014	Exception	StatementsToIR.processTask: failed to process - {0}	Call for support
DFE0015	Exception	StatementsToIR.processTask: failed to process - {0}	Call for support
DFE0016	Exception	StatementsToIR.processTask: failed to process - {0}	Call for support
DFE0017	Error	StatementsToIR.getPreviousTaskParam({0}, {1}): Failed to access {2} from previous task	Call for support
DFE0018	Error	StatementsToIR.transform: No statements for IVN {0} matched the validation	Call for support

DFE is com.edocs.tasks.statements2ir.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
		criteria.	
DFE0019	Error	StatementsToIR.isValidPath: The output path is not valid. Output paths that contain only white space or are null are not allowed.	Call for support
DFE0020	Error	StatementsToIR.isValidPath: The output path, {0}, is not a directory.	Call for support
DFE0021	Error	StatementsToIR.isValidPath: The task does not have write access to the output directory, {0}, and cannot generate the IR file.	Call for support
DFE0022	Exception	StatementsToIR.isValidPath: An exception was thrown while validating the output path, {0}	Call for support
DFE0023	Error	StatementsToIR.transform: Accounts were processed for Index Volume Number {0} but there is no content in the IR file. There is mostly likely a problem with the Statement XSL stylesheet published in the Detail Extractor view.	Call for support
DFE0024	Error	StatementsToIR.getViewNames: Encountered an exception while accessing the view list for application {0} - {1}	Call for support

Appendix B: Error Messages

DXL is com.edocs.tasks.dxloader.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
DXL0001	Error	DXTTableManager.removeFailedLoad: Failed to get the last loaded table name for index volume number {0}. Exception is {1}	Call for support
DXL0002	Warning	DXTTableManager.removeFailedLoad: Could not expire detail catalog entry for index volume number {0} and table {1}. Exception is {2}	Call for support
DXL0003	Warning	DXTTableManager.removeFailedLoad: Could not drop detail database table for index volume number {0} and table {1}. Exception is {2}	Call for support
DXL0004	Error	DXTTableManager.createTable: Failed to create an entry in the detail catalog table for index volume number, {0}, and view name, {1}. Exception is {2}	Call for support
DXL0005	Error	DXTTableManager.createTable: Creating database table {0} failed for index volume number {1}. Exception is {2}	Call for support
DXL0006	Information	DXTTableManager.removeFailedLoad: No database table to remove for index volume number {0} and view name {2}.	Call for support
DXL0007	Error	DXTTableManager.createTable: The index volume number, {0}, has already been processed for view name, {1}. This job cannot be completed.	Call for support

IAC is com.edocs.tasks.autoaccept.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
IAC0001	Information	AutoIndexVolAccept.processTask({0}, {1})	None
IAC0002	Information	AutoIndexVolAccept.processTask({0}, {1}): Accepting index volume: {3}	None
IAC0003	Error	AutoIndexVolAccept.processTask: failed to process - {0}	Call for support
IAC0004	Error	AutoIndexVolAccept.processTask: failed to process - {0}	Call for support
IAC0005	Error	AutoIndexVolAccept.processTask: failed to process - {0}	Call for support
IAC0006	Error	AutoIndexVolAccept.getTaskParams: Failed to get task config params - {0}	Call for support
IAC0007	Error	AutoIndexVolAccept.getTaskParams: Failed to get task config params - {0}	Call for support
IAC0008	Error	AutoIndexVolAccept.getTaskParams: Failed to get task config params - {0}	Call for support
IAC0009	Information	AutoIndexVolAccept.setTaskParams({0}, {1}, {2}): {3}	None
IAC0010	Error	AutoIndexVolAccept.setTaskParams: Failed to set task config params - {0}	Call for support
IAC0011	Error	AutoIndexVolAccept.setTaskParams: Failed to set task config params - {0}	Call for support
IAC0012	Error	AutoIndexVolAccept.setTaskParams: Failed to set task config params - {0}	Call for support
IAC0013	Error	AutoIndexVolAccept.isTaskConfigValid: Failed while validating config params - {0}	Call for support
IAC0014	Error	AutoIndexVolAccept.isTaskConfigValid: Failed while validating config params - {0}	Call for support

Appendix B: Error Messages

IAC is com.edocs.tasks.autoaccept.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
IAC0015	Error	AutoIndexVolAccept.isTaskConfigValid: Failed while validating config params - {0}	Call for support
IAC0016	Error	AutoIndexVolAccept.isTaskConfigValid: Failed while validating config params - {0}	Call for support

IDX is com.edocs.tasks.indexer.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
IDX001	Error	Failed to load indexer library - {0}	Call for support
IDX002	Error	Indexer.getTaskParams: Failed to get field list - {0}	Call for support
IDX003	Error	Indexer.setTaskParams: Failed to set indexer config. - {0}	Error
IDX004	Error	Indexer.isTaskConfigValid: Failed while validating config. - {0}	Error
IDX005	Information	Indexer.processTask({0}, {1})	Information
IDX006	Error	Indexer.processTask: ddf file path invalid for job Instance: {0}, Task Order: {1}	Error
IDX007	Error	Indexer.processTask: Failed to access data file: {0} for insatnce: {1}, task order {2}	Error
IDX008	Information	Indexer.processTask: Created ir file: {0} of size {1}	Information
IDX009	Error	Indexer.processTask: Invalid count of documents processed: {0}	Error
IDX0010	Error	Indexer.processTask: ir file: {0} not accessible	Error
IDX0011	Error	Indexer.processTask: Failed to process - {0}	Error

ISC is com.edocs.tasks.ivnscanner.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
ISC0001	Error	IVNScanner.getTaskParams: Failed to get task config params – {0}	Call for support
ISC0002	Error	IVNScanner.getTaskParams: Failed to get task config params – {0}	Call for support
ISC0003	Error	IVNScanner.getTaskParams: Failed to get task config params – {0}	Call for support
ISC0004	Error	IVNScanner.setTaskParams({0}, {1}, {2}): {3}	None
ISC0005	Error	IVNScanner.setTaskParams: Failed to set task config params - {0}	Call for support
ISC0006	Error	IVNScanner.setTaskParams: Failed to set task config params - {0}	Call for support
ISC0007	Error	IVNScanner.setTaskParams: Failed to set task config params - {0}	Call for support
ISC0008	Error	IVNScanner.isTaskConfigValid: Failed while validating config params - {0}	Call for support
ISC0009	Error	IVNScanner.isTaskConfigValid: Failed while validating config params - {0}	Call for support
ISC0010	Exception	IVNScanner.isTaskConfigValid: Failed while validating config params - {0}	Call for support
ISC0011	Exception	IVNScanner.isTaskConfigValid: Failed while validating config params - {0}	Call for support
ISC0012	Error	IVNScanner.processTask({0}, {1})	None
ISC0013	Information	IVNScanner.processTask({0}, {1}) - Done; ivn: {2}; file: {3}; ir file: {4}	None
ISC0014	Error	IVNScanner.processTask: failed to process - {0}	Call for support

Appendix B: Error Messages

ISC is com.edocs.tasks.ivnscanner.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
ISC0015	Error	IVNScanner.processTask: failed to process - {0}	Call for support
ISC0016	Error	IVNScanner.processTask: failed to process - {0}	Call for support
ISC0017	Error	IVNScanner.processTask: The number of days to scan must be a positive integer. It cannot be {0}	Call for support

IXL is com.edocs.tasks.ixloader.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
IXL0001	Information	IXLoader.processTask: Loaded index summary - {0}	Call for support
IXL0002	Error	IXLoader.configOK - missing param: {0}, for job {1}, ddn {2}, task order {3}	Call for support

LDR is com.edocs.tasks.ixloader.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
LDR0001	Exception	Loader.getTaskParams: Failed to get task config params - {0}	Call for support
LDR0002	Error	Loader.configOK - missing param: {0}, for job {1}, ddn {2}, task order {3}	Call for support
LDR0003	Information	Loader.setTaskParams({0}, {1}, {2}): {3}	None
LDR0004	Exception	Loader.setTaskParams: Failed to set task config params - {0}	Call for support
LDR0005	Error	Loader.isTaskConfigValid: Config never set for job: {0}, ddn: {1}	Call for support

LDR is com.edocs.tasks.ixloader.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
LDR0006	Exception	Loader.isTaskConfigValid: Failed while validating config params - {0}	Call for support
LDR0007	Information	Loader.processTask({0}, {1})	Call for support
LDR0008	Error	Loader.processTask: ir file name: {0} is inaccessible	Call for support
LDR0009	Error	Loader.processTask: Invalid db connect info for db: {0}	Call for support
LDR0011	Error	Loader.processTask: The SQL loader did not generate a log file so it most likely failed to start for control file: {0}	Call for support
LDR0012	Error	Loader.processTask: The SQL loader encountered problems, see {0}	Call for support
LDR0013	Exception	Loader.processTask: failed to process - {0}	Call for support
LDR0014	Exception	Loader.processTask: failed to process - {0}	Call for support
LDR0015	Exception	Loader.processTask: failed to process - {0}	Call for support
LDR0016	Error	Loader.processTask: There are no records to process for {0}. Either the IR file has no records or the total number of records is less than or equal to the number of rows to skip.	Call for support
LDR0017	Error	Loader.getPreviousTaskParam({0}, {1}): Failed to access {2} from previous task	Call for support
LDR0018	Error	SQLServerTableLoader.modifyIRFile: The first identifiable data record in IR file {0} is an empty line. No data records will be copied to the last position in the file.	Call for support

Appendix B: Error Messages

LDR is com.edocs.tasks.ixloader.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
LDR0019	Information	Loader.processTask: Loaded {0} rows successfully.	None
LDR0021	Error	TableLoader.executeTableLoad: Exception thrown while clean up temporary files... {0}	Call for support
LDR0022	Information	SQLServerTableLoader.cleanUp: Successfully removed temporary file {0}	None
LDR0023	Warning	SQLServerTableLoader.cleanUp: Unable to clean up temporary file {0}	Call for support
LDR0024	Error	SQLLoader Log: Found an error in {0} on line {1}. The line is... {2}. Please examine the sql loader log file for more details on how to correct the problem.	Call for support
LDR0025	Exception	TableLoader.validateTableLoad: Encountered a problem accessing the log file created by the sql loader... {0}	Call for support
LDR0026	Exception	TableLoader.validateTableLoad: Failed to access log file created by the sql loader... {0}	Call for support
LDR0027	Information	SQLServerTableLoader.cleanUp: Cannot remove temporary ir file {0} because it does not exist.	None
LDR0028	Information	SQLLoader Potential Resolution: {0}	None
LDR0029	Error	Loader.processTask: The IR file, {0}, is invalid: {1}.	Call for support

MAI is com.edocs.services.mailer.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
MAI0001	Error	Error intializing mailer purge stored procedure call strings	Call for support

MGR is com.edocs.services.merger.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
MGR0001	Error	InvalidAppException occurred while trying to access the application path: {0}	Call for support
MGR0002	Exception Error	{0} occurred while trying to access the application path: {1}	Call for support
MGR0003	Exception	Exception occurred while trying to access the input stream: {0}	Call for support
MGR0004	Exception	Exception occurred while reading versioning information: {0}	Call for support
MGR0005	Exception	C++ merger code threw an exception: {0}	Call for support
MGR0007	Error	UnsatisfiedLinkError, Check LD_LIBRARY_PATH	Call for support
MGR0008	Exception Error	Unable to deserialize the parameter properties object: {0} Unable to interpret the docid: {0}	Call for support
MGR0009	Exception	Exception occurred while application information: {0}	Call for support
MGR0010	Exception	Exception occurred while reading versioning information: {0}	Call for support
MGR0011	Exception	C++ merger code threw an exception: {0}	Call for support
MGR0013	Error	UnsatisfiedLinkError, Check LD_LIBRARY_PATH	Call for support

Appendix B: Error Messages

MGR is com.edocs.services.merger.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
MGR0014	Error	Unable to interpret the docid: {0}	Call for support
MGR0015	Error	Exception occurred while preparing to retrieve the document: {0}	Call for support
MGR0016	Error	Exception occurred while reading versioning information: {0}	Call for support
MGR0017	Error	C++ code threw an exception: {0}	Call for support

MNS is com.edocs.tasks.mns.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
MNS0001	Information	Started	None
MNS0002	Exception	Exception caught: {0}	Call for support
MNS0003	Information	Finished Processing	None
MNS0004	Error	Something is very wrong. Mail servers might not be working.	Call for support
MNS0005	Error	Total Accounts = {0}, Total Tried = {1}, Total Emails Sent = {2}	None
MNS0006	Exception	Exception caught: {0}	Call for support
MNS0007	Exception	Exception caught: {0}	Call for support
MNS0008	Exception	Exception caught: {0}	Call for support
MNS0009	Exception	Exception caught: {0}	Call for support
MNS0010	Exception	Exception caught: {0}	Call for support
MNS0011	Information	Created and starting...	None
MNS0012	Exception	Exception caught: {0}	Call for support
MNS0013	Exception	Exception caught: {0}	Call for support
MNS0014	Exception	Exception caught: {0}	Call for support

MNS is com.edocs.tasks.mns.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
MNS0015	Exception	Exception caught: {0}	Call for support
MNS0016	Exception	Exception caught: {0}	Call for support
MNS0017	Exception	Exception caught: {0}	Call for support
MNS0018	Exception	Exception caught: {0}	Call for support
MNS0019	Exception	Exception caught: {0}	Call for support
MNS0020	Error	Error initializing stored procedures call strings	Call for support
MNS0021	Information	Failed to send mails, tried retry number of times, Perhaps Mail servers are not working.	Check your mail server
MNS0022	Exception	Exception caught: {0}	Call for support

MON is com.edocs.services.monitor.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
MON0001	Exception	Exception occurred while looking up EJB: {0}	Call for support
MON0002	Exception	Exception occurred while looking up EJB: {0}	Call for support
MON0003	Information	Monitor created	None
MON0004	Exception	Exception occurred while looking up EJB: {0}	Call for support
MON0005	Information	Monitor removed	None

Appendix B: Error Messages

PDB is com.edocs.pwc.db.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
PDB0001	Error	Error initializing stored procedure call strings	Call for support

PTK is com.edocs.pwc.tasks.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
PTK0001	Information	Created and starting	None
PTK0002	Exception	Exception caught: {0}	Call for support
PTK0003	Exception	Exception caught: {0}	Call for support
PTK0004	Exception	Exception caught: {0}	Call for support
PTK0005	Exception	Exception caught: {0}	Call for support
PTK0006	Exception	Exception caught: {0}	Call for support
PTK0007	Exception	Exception caught: {0}	Call for support
PTK0008	Exception	Exception caught: {0}	Call for support
PTK0009	Exception	Exception caught: {0}	Call for support
PTK0010	Exception	Exception caught: {0}	Call for support
PTK0011	Exception	Exception caught: {0}	Call for support
PTK0012	Exception	Exception caught: {0}	Call for support
PTK0013	Exception	Exception caught: {0}	Call for support
PTK0014	Exception	Exception caught: {0}	Call for support
PTK0015	Exception	Exception caught: {0}	Call for support
PTK0016	Information	Finished processing task	None

PUR is com.edocs.tasks.purge.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
PUR0001	Information	{0} purged {1} records from the database	Call for support
PUR0002	Information	The task configuration for {0} has a negative purge age of {1}. No purging will occur.	Call for support
PUR0003	Error	The following exception was caught during {0} {1}	Call for support

SCH is com.edocs.pwc.scheduler.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
SCH0001	Information	Starting JobProcessor for ({0}) job instance: {1}	Call for support
SCH0002	Exception	Exception in processJobs: {0}	Call for support
SCH0003	Information	PWC Scheduler started	Call for support
SCH0004	Exception	Reprocessing of jobs failed: {0}	None
SCH0005	Exception	Processing of jobs failed: {0}	None
SCH0006	Exception	Non-recoverable error in PWC Scheduler!: {0}	None
SCH0007	Information	Starting job instance thread for: {0}	Call for support
SCH0008	Error	Job instance initialization failed for: {0} {1}	Call for support
SCH0009	Error	Failed to get Job/Schedule Object for job instance: {0} {1}	Call for support
SCH0010	Error	Failed to update failed job status for: {0} {1}	None
SCH0011	Information	Job instance: {0}; Starting task: {1}; order: {2}	Call for support

Appendix B: Error Messages

SCH is com.edocs.pwc.scheduler.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
SCH0012	Information	Job instance: {0}; Done task: {1}; order: {2}	Call for support
SCH0013	Information	Job instance: {0}; Starting task: {1}; order: {2}	Call for support
SCH0014	Error	Failed to update failed job status for: {0} {1}	None
SCH0015	Error	Failure in job processor thread for: {0} {1}	Call for support
SCH0016	Error	Failed to update failed job status for: {0} {1}	None
SCH0017	Error	Status forced to "Failed" from {0} for hung job instance: {1}	None
SCH0018	Error	PWC Scheduler unable to force "Fail" hung job instances: {0}	Call for support
SCH0019	Error	Failed to define next schedule for job instance: {0}	None
SCH0020	Information	Done job instance thread for: {0}	None

SCN is com.edocs.tasks.scanner.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
SCN0001	Error	Scanner.getTaskParams: Failed to get task config params - {0}	Call for support
SCN0002	Error	Scanner.setTaskParams - missing param: {0}, for job {1}, ddn {2}, task order {3}	Call for support
SCN0003	Information	Scanner.setTaskParams({0}, {1}, {2}): {3}	Call for support
SCN0004	Error	Scanner.setTaskParams: Failed to set task config params - {0}	Call for support
SCN0005	Error	Scanner.isTaskConfigValid: Failed to find	Call for support

SCN is com.edocs.tasks.scanner.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
		input directory: {0}	
SCN0006	Error	Scanner.isTaskConfigValid: Failed to create output directory: {0}	Call for support
SCN0007	Error	Scanner.isTaskConfigValid: Failed while validating config params - {0}	Call for support
SCN0008	Information	Scanner.processTask({0}, {1})	None
SCN0009	Error	Scanner.processTask: Failed to create ddn volume - {0}	Call for support
SCN0010	Error	Scanner.processTask: Failed caused by an invalid input file: {0}	Call for support
SCN0011	Information	Scanner.processTask({0}, {1}): Attempting to move {2} -> {3}	None
SCN0012	Error	Scanner.processTask({0}, {1}) - attempt to move: {2} -> {3} failed	Call for support
SCN0013	Information	Scanner.processTask({0}, {1}) - attempt to move: {2} -> {3} succeeded	None
SCN0014	Error	Scanner.processTask: failed to process - {0}	Call for support
SCN0015	Information	Scanner.processTask: {0} : file length is 0.	None

SCT is com.edocs.tasks.shellcmd.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
SCT0001	Error	Exception caught: {0}	Call for support
SCT0002	Error	Shell Command unable to finish: {0}	Call for support
SCT0003	Information	ShellCmdTask: About to execute the following shell command: {0}	None
SCT0004	Information	ShellCmdTask: Return Value = {0}	None

Appendix B: Error Messages

SCT is com.edocs.tasks.shellcmd.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
SCT0005	Information	ShellCmdTask: Shell output: {0} DVN: {1}	None
SC0006	Error	ShellCmdTask: processTask: Failed to create ddn volume - {0}	Call for support
SCT0007	Error	ShellCmdTask: processTask: Unable to set task output: {0}	Call for support
SCT0008	Information	ShellCmdTask: DVN changed. Shell output: {0} DVN: {1}	None

SDF is com.edocs.services.statements2ir.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
SDF0001	Error	DetailTransformer.<constructor>: The application name is null.	Call for support
SDF0002	Error	DetailTransformer.<constructor>: The detail view is null.	Call for support
SDF0003	Error	DetailTransformer.<constructor>: The content writer is null.	Call for support
SDF0005	Information	DetailTransformer.transform: Cannot extract statements for a null index volume number.	Call for support
SDF0006	Information	DetailTransformer.process: The view type is null.	Call for support
SDF0007	Information	DetailTransformer.process: The view name is null.	Call for support
SDF0009	Information	DetailTransformer.process: The account resolver is not specified so all accounts will be extracted	Call for support
SDF0010	Information	DetailTransformer.process: Failed to create the XSLTMerger	Call for support
SDF0011	Information	XML2Table.getTable: Failed to parse the	Call for support

SDF is com.edocs.services.statements2ir.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
		Detail Extractor database table definition file. The error is as follows: {0}	
SDF0012	Information	DetailTransformer.process: Failed to find the XSLT Template	Call for support
SDF0013	Information	DetailTransformer.process: Failed to find the DDF	Call for support
SDF0014	Information	DetailTransformer.getResolver: Failed to create the account resolver : {0}	Call for support
SDF0015	Information	DetailTransformer.writer: Failed during XSLT merger : {0}	Call for support
SDF0016	Information	DetailTransformer.writer: Data authorization exception : {0}	Call for support
SDF0017	Error	DetailTransformer.writer: Failed to read the result of the XSLT merger : {0}	Call for support
SDF0018	Error	DetailTransformer.writer: There is no content in the IR file for Index Volume Number {0}. The most likely scenario is that no accounts matched the validation criteria.	Call for support
SDF0019	Error	DetailTransformer.transform: Failed while transforming the XML to the output format : {0}	Call for support
SDF0020	Error	IRContentWriter.getHeader: The database table XML file is null or empty	Verify that the published Detail Extractor view is valid Call for support
SDF0021	Information	IRContentWriter.endProcessing: File is 0 length and will be deleted.	Call for support

Appendix B: Error Messages

SDF is com.edocs.services.statements2ir.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
SDF0022	Error	IRContentWriter.endProcessing: {0} statement(s) processed.	Call for support
SDF0023	Information	IRContentWriter.endProcessing: File is null and will be deleted.	Call for support
SDF0024	Error	XML2TableHandler.error: {0}	Call for support
SDF0025	Error	XML2Table.getTable: Failed to find a SAX Parser using the 'com.edocs.xml.sax.parser' key.	Call for support
SDF0026	Error	XML2Table.getTable: Failed to create a SAX Parser using the 'com.edocs.xml.sax.parser' key.	Call for support
SDF0027	Error	XML2Table.getTable: Errors found while parsing the create table XML. XML name: {0}.	Call for support
SDF0028	Error	DetailView.getVersionObj: Error retrieving version reader : {0}.	Call for support
SDF0029	Error	DetailView.getVersionObj: Error retrieving version set : {0}.	Call for support
SDF0030	Error	DetailTransformer.getAcctList: Error retrieving account list for index volume number {0}. Exception is {1}	Call for support
SDF0031	Error	XML2TableHandler.error: There is a good chance that the Detail Extractor database table description XML file does not have the following XMLSchema declaration. xmlns:xsi="http://www.w3.org/2000/10/XM LSchema-instance" xsi:noNamespaceSchemaLocation="edx-DE- table.xsd".	Consult the eaDirect Detail Extractor xml sample files for the correct syntax Call for support
SDF0032	Error	DetailTransformer.getAcctList: There are no	Call for support

SDF is com.edocs.services.statements2ir.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
		accounts listed for index volume number {0}.	
SDF0033	Error	XML2Table.getTable: The XML reader {0} doesn't support one of the following features 'http://xml.org/sax/features/validation', 'http://xml.org/sax/features/namespace', 'http://apache.org/xml/features/validation/schema'.	Call for support

SHF is com.edocs.tasks.StaticHtmlFormatter.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
SHF0001	Information	StaticHtmlFormatter task began processing	None
SHF0002	Error	StaticHtmlFormatter failed in the C++ code	Call for support
SHF0003	Error	StaticHtmlFormatter finished processing {0} files in time(hrs:min:sec) - {1}	None
SHF0004	Error	Exception caught: {0}	Call for support
SHF0005	Exception	Exception caught: {0}	Call for support
SHF0006	Exception	Exception caught: {0}	Call for support
SHF0007	Exception	Exception caught: {0}	Call for support
SHF0008	Error	Unable to read DDF/ALF/Template paths for this task because the version set is either missing or corrupt.	Call for support

VRS is com.edocs.services.versioning.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
VRS0001	Exception	Exception occurred while creating IVersionSetReader object locally: {0}	Call for support

Appendix B: Error Messages

VRS is com.edocs.services.versioning.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
VRS0002	Exception	Exception occurred whilst creating IVersionedReader remote object: {0}	Call for support
VRS0003	Exception	Unable to instantiate remote IVersionedReader interface: {0}	Call for support
VRS0004	Exception	Exception occurred whilst creating IVersionSetWriter object locally: {0}	Call for support
VRS0005	Exception	Exception occurred whilst creating IVersionSetWriter remote object: {0}	Call for support
VRS0006	Exception	Unable to instantiate remote IVersionSetWriter interface: {0}	Call for support
VRS0007	Exception	Exception occurred whilst creating IVersionedObj object locally: {0}	Call for support
VRS0008	Exception	Exception occurred whilst creating IVersionedObj remote object: {0}	Call for support
VRS0009	Exception	Unable to instantiate remote IVersionedObj interface: {0}	Call for support
VRS0010	Exception	Unable read data required for instantiating remote version interface implementations	Call for support

XML is com.edocs.tasks.xml.LogMsgCatalog			
Message ID	Severity	Message Text	User Action
XML0001	Information	XMLFormatter task began processing	None
XML0002	Error	XMLFormatter failed in the C++ code. Return value from C++ code was: {0}	Call for support
XML0003	Information	XML Formatter task finished processing a total of {0} documents.	None
XML0004	Exception	Exception caught: {0}	Call for support
XML0005	Exception	Exception caught: {0}	Call for support
XML0006	Exception	Exception caught: {0}	Call for support
XML0007	Error	Unable to read DDF path for this task because the version set is either missing or corrupt.	Call for support

Appendix C: Glossary

eaDirect Terms and Acronyms

This glossary defines eaDirect terms as well as technology and general business terms related to electronic presentment and payment.

Account Number	The identification of an individual document in a source file. The account number indicates the identity of the individual or business associated with the account information.
ACH	Automated Clearing House. A network for Electronic Funds Transfer, governed by the National ACH Association (NACHA) and the U.S. Federal Reserve Bank. The ACH Network is used for all kinds of EFT transactions, including direct deposit of paychecks and monthly debits for routine payments to vendors. The ACH network is separate and distinct from the various bankcard networks that process credit card transactions. ACH transactions are conducted among participating financial institutions by the transfer of ACH files in batch mode, which can take up to 72 hours before the money is actually transmitted. In eaPay, an ACH return file is sent if there are insufficient funds in the account or if there are other errors or problems with the transaction. (Term used with eaPay.)
Addenda Record	ACH record type data needed to completely identify an account holder or to provide information concerning a transaction that carries supplemental payment. (Term used with eaPay.)

- AIX** Advanced Interactive Executive. IBM's version of UNIX, which runs on 386 and higher PCs, RS/6000 workstations and 390 mainframes. It is based on AT&T's UNIX System V with Berkeley extensions.
- ALF** Application Logic File. An ALF contains the rules to present the data extracted from the original data source in a customized HTML format. An ALF is created during application design and development using the eaDirect Composer tool. The ALF (which is in XML format) is then used along with the DDF during live statement retrieval to display extracted account data on the Web, in email, etc. The ALF also contains business logic (conditional statements that consider current statement data) for marketing and other customization purposes. EaMarket adds enhanced marketing promotion capabilities to eaDirect. In addition to dynamic statement presentment, ALFs are also developed and used for the composition of static output, most commonly notification emails. An application can have multiple ALFs for use in presenting different statement views. See also **View**.
- Anchor** An additional (and optional) condition placed on a DDF extraction rule to ensure the proper data is extracted or located. Anchors can be defined on any DDF extraction rule, including fields, markers and page styles. If an anchor is defined, the system looks for the anchor pattern first and then the defined element. Anchors are useful when the data to be located is too generic to be detected by a pattern or it is found within a large range of rows and columns. See also **Table Anchor**.
- API** Application program interface. A set of routines, protocols, and tools for building software applications. A good API makes it easier to develop a program by providing all the building blocks. A programmer puts the blocks together.
- Application** A customized set of eaDirect files (DDFs, ALFs, Statement HTML templates, Group HTML templates and images) created exclusively to extract and present online statements for a service provider. A custom application must be designed for each input data source. Each application is given a unique name. A single service provider can have more than one application if they present data from input data sources with different formats or content.

Application Name	The name of a particular application. eaDirect expects a DDF and ALF named after the application when dynamically composing the first statement view. The main application directory must also be named after the application. See also DDN .
ASP	<p>Active server page. A Web server technology from Microsoft that allows for the creation of dynamic, interactive sessions with the user. An ASP is a Web page containing HTML and embedded programming code written in VBScript or Jscript. It was introduced with Version 3.0 of Microsoft's Internet Information Server (IIS). When an ASP page is requested by the browser, IIS executes the embedded program. ASPs are Microsoft's alternative to CGI scripts and JavaServer Pages (JSPs), which allow Web pages to interact with databases and other programs. Third party products add ASP capability to non-Microsoft Web servers.</p> <p>ASP can also refer to Application Service Provider. See Service Provider.</p>
Authentication	The process by which a Web application (such as an eaDirect application) verifies the identity of an individual Web user. eaDirect applications must verify the identity of an enrolled user before they can view account information.
Business Logic	<p>In eaDirect, the conditional statements added to an ALF during application design and development. Conditional statements are written to create customized statements based on the particular financial and personal activity of each statement recipient. Business logic lets you display alternative messages or implement targeted advertising campaigns under specified conditions.</p> <p>Also refers to the Java code that implements the functionality of a business application. In the EJB model, this logic is implemented by the methods of an enterprise bean.</p>
Category	Subdirectory of a library. The lowest level of storage; a category has no subdirectories. A library can have one or more categories. (Term used with eaMarket.)

CCD	Cash Concentration and Disbursement. An automated corporate payment used primarily for the intra-company concentration or disbursement of funds. EaPay supports this type of ACH payment. (Term used with eaPay.)
Clicks	A subset of impressions and show the number of times the promotion was clicked. (Term used with eaMarket.)
Column	A unique data field that appears within a table. The columns of a table are usually related to each other in some logical way. Related table column occurrences usually appear on the same row, although this is not required. For example, in a table of telephone call detail, there could be columns for “Number Dialed,” “Call Duration,” and “Cost of Call.” (In previous releases of eaDirect, columns have been referred to as record fields.)
Composer	The Windows-based, GUI tool used to create the rules for mapping data extracted from an input file to an HTML template for display on the Web, email, etc. The Composer displays the HTML template in WYSIWYG format and lets designers drag and drop placeholders for data elements into the template. The Composer also lets designers define business rules (build conditions) for marketing purposes. The main output of the Composer is an Application Logic File (ALF). The secondary output of the Composer is HTML templates, edited to include placeholder tags for the placement of extracted data elements and the execution of business logic statements. Composer is used along with DefTool to design and develop an eaDirect application.
Container	A Java entity that provides life cycle management, security, deployment, and runtime services to components. Each type of container (EJB, Web, JSP, servlet, applet, and application client) also provides component-specific services.
Content	What is to be shown for a promotion, such as GIF, JPEG, HTML and URLs. (Term used with eaMarket.)
CORBA	Common Object Request Broker Architecture. An architecture that enables pieces of programs, called objects, to communicate with one another independent of which programming language they were written in or what operating system they’re running on.

CSR	Customer Service Representative. A CSR can be employed by the service provider or other entity.
CTE, CTX	Corporate Trade Exchange. An ACH Standard Entry Class code type used for corporate to corporate payments. CTX can transmit up to 9,999 addenda records in ANSI X12 syntax (EDI). (Term used with eaPay.)
Data Store	Physical location where the content of a promotion is stored. A data store can be a database or file system and there can be one or many. (Term used with eaMarket.)
DDF	Data Definition File. The DDF contains the rules for finding and extracting data from an application's input data source. An application designer creates a DDF file using eaDirect's DefTool during the initial system design. It is created exclusively for use with a service provider's particular data source format and application. DDFs are used on a regular basis during statement live retrieval to extract account data for display on the Web. The DDF is also used by the Indexer task to prepare a data source and the entire eaDirect application for live statement retrieval.
DDF Namespace	The collection of data elements defined for extraction in a DDF, including field, table, and group names. Most data elements are explicitly named, but some could be implicitly named (for example, the rows within a detail block might not have individual names). The contents of an application DDF Namespace appear in the left pane of the Composer screen, within the Definition tab. A designer can click and drag elements to the HTML view on the right.
DDN	Data Definition Name. The name of a particular application (internal to eaDirect). EaDirect expects a DDF and ALF named after the DDN when dynamically composing the first statement view. The main application directory must also be named after the DDN. See also Application Name .

DefTool	The Windows-based, GUI tool used to create the rules for finding and extracting data from a data source. DefTool's interface lets designers view the data source content in digitized format and graphically identify the elements to extract. The output of DefTool is a Data Definition File (DDF). DefTool is used along with the Composer tool to design and develop an eaDirect application.
Detail Extractor	A type of eaDirect batch job used to extract and upload data from the data input file to an eaDirect database table. This data requires custom application functionality to extract the data from the database to merge with statements or use in any other way. For the Detail Extractor job you publish a DDF file, a database table schema XML file, and an XSLT style sheet.
Deployment	The process by which software is installed into an operational environment.
Distributed Application	An application made up of distinct components running in separate runtime environments, usually (in the eaDirect environment) on similar platforms connected via a network. Typical distributed applications are two-tier (client/server), three-tier (client/middleware/server), and n-tier (client/multiple middleware/multiple servers).
DOCID	An API-level string created by eaDirect to uniquely identify a particular document, or statement.
Document	A single statement or set of account data in an input data source. An input data source typically consists of many individual documents. Also used to describe a Web page composed by eaDirect to display a single user's statement or view of account data.
Document Style	A delimiter used to locate the first page of each statement in a data source. The Document Style data string is defined during application design to enable eaDirect to identify a new statement in the data source. If the Document Style is found on a page, it indicates with certainty that this is the first page of the next statement (delimiting statements in the data source). The Document Style definition consists of a name, pattern, and the row and column coordinates within which eaDirect must search for the start of the identifying data string.

Dynamic Paging	The ability to dynamically spread repeating data across multiple HTML pages including the ability to navigate the resulting pages. (Dynamic paging does not correspond to physical pages identified in the source data file.)
Dynamic Web View	A view used to create a dynamic statement presentation on the Web. Dynamic view types are: HTML, CSV, XML, Chart, XSLT, XML Query. You publish dynamic Web views.
eaDirect	edocs flagship software product; the foundation of eaSuite. EaDirect provides secure online account management and presentment systems capable of using a variety of input data formats.
eaMarket	The eaSuite software product that enhances eaDirect's capability to integrate marketing and customer service messages throughout the online account management experience.
eaPay	The eaSuite software product that provides electronic payment capabilities to an eaDirect system and lets the end-user manage and track payments. EaPay supports a wide range of electronic payment methods including CheckFree, credit card and ACH transactions.
eaPost	The eaSuite software product that handles all aspects of distributing summary account information to any end-point. Service providers can plug-in new cartridges to reach new destinations.
eaSuite	The complete set of edocs online presentment products, including eaDirect and add-on products: eaMarket, eaXchange, eaPay, and eaPost.
EBPP	Electronic bill presentment and payment.
EJB	JavaSoft Enterprise JavaBeans. A specification defining a component architecture for building distributed, object-oriented business applications in Java. Enterprise JavaBeans encapsulate business logic. EJB developers can develop their own business components (such as invoices, bank accounts, and shipping routes), called enterprise beans, or purchase them from third-party vendors. Enterprise JavaBeans run in a special environment called an EJB container.

EJB Container	A container that implements the EJB component contract of the J2EE architecture. This contract specifies a runtime environment for enterprise beans that includes security, concurrency, life cycle management, transaction, deployment, and other services. An EJB container is provided by an EJB or J2EE server.
EJB Server	Software that provides services to an EJB container. For example, an EJB container typically relies on a transaction manager that is part of the EJB server to perform the two-phase commit across all the participating resource managers. The J2EE architecture assumes that an EJB container is hosted by an EJB server from the same vendor, so does not specify the contract between these two entities. An EJB server can host one or more EJB containers.
Email Notification	A type of eaDirect batch job used to dispatch secure email notifications to enrolled users after a statement has been processed. You must publish a view with each EmailNotification job.
Enrollment For Payment	The process by which an end-user becomes enrolled in a service provider's system for the purpose of paying bills. The enrollment process involves providing user-profile information that uniquely identifies the individual to an online billing service provider.
Enrollment For Presentation	The process by which a Web user becomes enrolled in a service provider's system for the purpose of viewing interactive or notification-based statement presentation. The enrollment process involves providing user-profile information that uniquely identifies the individual to an online billing service provider.
Enterprise Bean	A component that implements a business task or business entity; either an entity bean or a session bean. Also called EJB.
ERP	Enterprise Resource Planning. An integrated information system that serves all departments in an enterprise. ERP evolved from the manufacturing industry, and implies the use of packaged rather than proprietary software, although ERP modules can possibly be modified using a vendor's proprietary tools or a programming language (standard or proprietary).

Field	An independent data element within a statement that stands apart from any repeating line item detail. It is often used to isolate account information such as an account number, customer name, and summary financial information, such as total amount due.
FIDS	Formatted Input Data Stream. A type of data file created for an alternative method of generating a DDF automatically. A FIDS file contains extracted source data and related metadata. An eaDirect product extension uses the FIDS to automatically generate extraction rules (the DDF) for an application.
FTP	File Transfer Protocol. A protocol used on the Internet for sending files.
Group	<p>A collection of tables that repeat together within a document. A group can be used to present related data elements together in moderately complex applications containing multiple or combined statement information. For example, a group could be used to consolidate shipping information with related detail in a table that repeats for each ship-to address. A group can contain tables and/or groups of tables.</p> <p>Also a unit that reviews and approves content before it goes into a promotion. (Term used with eaMarket)</p>
HTML Output	A type of eaDirect batch job used to create static HTML-formatted email and output files. You publish a DDF file for the Indexer task and a view for this job type.
IIOB	Internet Inter-ORB Protocol. A protocol used for communication between CORBA object request brokers in a TCP/IP environment.
IIS	Internet Information Server. Microsoft's Web server that runs on, and is bundled with, Windows NT and Windows 2000. IIS is limited to the Windows NT platform, but is relatively easy to administer since it is tightly integrated with the operating system. (Netscape's Web servers run on all major platforms, including Windows NT, OS/2, and UNIX.)
Impressions	The number of times the page containing the promotion was accessed. (Term used with eaMarket.)

Indexer	A type of eaDirect batch job used to prepare a data file to be accessed by other production tasks. Indexer extracts meta-data from the application's DDF and stores the content in partitioned database index tables. Indexing an input file enables sub-second statement-retrieval response time across data sets of millions of electronic documents while maintaining the core data in the original input format. You publish a DDF file for the Indexer task, and dynamic Web views which depend on the Indexer job.
Input file (also Data File)	A service provider's file of account or statement data which eaDirect extracts and presents on the Web. An input file consists of many individual documents. Also called a source file, eaDirect supports a wide variety of print formats, database extracts, and intermediate document composition formats. Each eaDirect application must be custom-designed for its unique input file format and content.
ISAPI	Internet Server API. A programming interface on IIS. Using ISAPI function calls, Web pages can run programs written as DLLs on the server, typically to access a database. IIS comes with a DLL that lets embedded queries access ODBC-compliant databases. ISAPI is an alternative to using CGI scripts on Microsoft Web servers. The counterpart to ISAPI on the client side is WinInet.
Java API	Java Application Programming Interface. Prewritten code organized into packages of similar topics. For instance, the Applet and AWT packages include classes for creating fonts, menus, and buttons.
J2EE	Java 2.0 Enterprise Edition
J2EE Application	Any deployable unit of J2EE functionality. This can be a single module or a group of modules packaged into an .ear file with a J2EE application deployment descriptor. J2EE applications are typically engineered to be distributed across multiple computing tiers.
J2EE Server	The runtime portion of a J2EE product. A J2EE server provides Web and/or EJB containers.
J2SE	Java 2.0 Standard Edition

JDBC	Java Database Connectivity. An industry standard for database independent connectivity between the Java platform and a wide range of databases. The JDBC interface provides a call-level API for SQL-based database access.
JDK	Java Development Kit. A software development environment for writing applets and applications in the Java programming language.
JMS	Java Messaging Service. An API for use with enterprise messaging systems such as IBM, MQ Series, TIBCO Rendezvous, etc.
JNDI	Java Naming & Directory Interface. A set of APIs that assists with the interfacing to multiple naming and directory services.
JNI	Java Native Interface. A programming interface (API) in Sun's Java Virtual Machine used for calling native platform elements such as GUI routines. RNI (Raw Native Interface) is the JNI counterpart in Microsoft's Java Virtual Machine.
Job	An eaDirect production function designed to produce a particular type of output in an application. All the different types of jobs defined for an application contribute to the process of extracting and presenting cyclical account data for viewing online. Jobs can be scheduled to run automatically on specific days, times, and intervals. See also Job Type .
Job Type	The particular function or purpose of an eaDirect job. Job types include Indexer, Email Notification, Purge App, Purge Logs, XML Output, HTML Output, and Detail Extractor.
JSP	JavaServer Page. An extensible Web technology that uses template data, custom elements, scripting languages, and server-side Java objects to return dynamic content to a client. Typically the template data is HTML or XML elements, and in many cases the client is a Web browser. (JSPs are converted to Java Servlets by the servlet engine.)
JTA	Java Transaction API.

JTS	Java Transaction Services. A way to maintain control of distributed transactions to protect them from sharing violations or other failures. WebLogic uses JTS internally for its transaction processing functionality, but does not export any of its JTS implementation as a public API.
JVM	Java Virtual Machine. An abstract computing machine, or virtual machine, JVM is a platform-independent programming language that converts Java byte code into machine language and executes it. Most programming languages compile source code directly into machine code that is designed to run on a specific microprocessor architecture or operating system, such as Windows or UNIX. A JVM - a machine within a machine - mimics a real Java processor, enabling Java bytecode to be executed as actions or operating system calls on any processor regardless of the operating system. For example, establishing a socket connection from a workstation to a remote machine involves an operating system call. Since different operating systems handle sockets in different ways, the JVM translates the programming code so that two machines on different platforms can connect.
LDAP	Lightweight Directory Access Protocol. A protocol used to access a directory listing. LDAP support is being implemented in Web browsers and e-mail programs that can query an LDAP-compliant directory. LDAP is expected to provide a common method for searching e-mail addresses on the Internet, eventually leading to a global white pages. LDAP is a sibling protocol to HTTP and FTP and uses the ldap:// prefix in its URL.
Library	Storage area within a data store. There can be one or more storage areas per data store. (Term used with eaMarket.)
Marker	A set of delimiters defining where eaDirect can find and extract table or group data from source file. Markers are defined using the DefTool during the application design and development process.

Metadata	Descriptive information about data elements; data about data. In eaDirect, metadata is the term used to describe the key data elements used to positively identify a statement within a data source during live retrieval. The metadata is extracted from the input data source and preserved in partitioned index tables during the Indexing job to prepare the data source to be accessed by other production tasks.
MICR	Magnetic Ink Character Recognition, sometime used to refer to the account and transit number at the bottom of a check.
NACHA	National Automated Clearing House Association. The organization that governs and sets the operational rules for all participants of the ACH Network (Term used with eaPay.)
NSAPI	Netscape Server API. A programming interface on Netscape's Web Server. Using NSAPI function calls, Web pages can invoke programs on the server, typically to access data in a database. NSAPI is an alternative to using CGI scripts on Netscape Web servers.
OAM	Online account management. The ability to provide live account statement views to customers and manage customer relationships on the Web. Online account management is made available by a service provider. For the customer, OAM can include the ability to make payments online and/or receive statement summaries on a selected payment consolidation portal.
ODBC	Open Database Connectivity. A standard database access method developed by Microsoft to make it possible to access any data from any application, regardless of which DBMS is handling the data. ODBC inserts a middle layer, called a database driver, between an application and the DBMS. The purpose of this layer is to translate the application's data queries into commands the DBMS understands, which requires that both the application and DBMS be ODBC-compliant.
ODFI	Originating Depository Financial Institution. Institution responsible for the origination of ACH transactions; the institution could deposit items directly with an ACH operator. Typically, the ODFI is the online service provider's bank. (Term used with eaPay.)

ORB	Object Request Broker. A component in the CORBA programming model, ORB acts as the middleware between clients and servers. In the CORBA model, a client can request a service without knowing what servers are attached to the network. The various ORBs receive the requests, forward them to the appropriate servers, then send the results back to the client.
Page Style	A definition created in the DDF to identify a statement page (subsequent to the Document Style) that contains a finite subset of the statement's data. A statement can have many pages consisting of one page style or they can have multiple page styles. Examples of different page styles include a summary page, line item detail pages, a cover letter, an ID card, or other types of information formatted differently (and therefore contain separate subsets of data). The page Style definition consists of a name, pattern and the row and column coordinates within which eaDirect must search for the start of the data string that differentiates the page Style from others in the statement.
Pattern	The nature and format of a data element to be extracted from an input file. A pattern must be defined for each data element to be extracted from an input file. For example, dollar amounts vary in length (number of digits) across statements, yet have the same general pattern. All addresses vary in length, yet consistently contain both numbers and letters, etc. Patterns are specified in all DDF definitions (page styles, fields, table columns and markers, etc.) as Regular Expressions. To extract a piece of data from an input file, eaDirect looks to the DDF file for the extraction rules (the page style, search area and pattern) defined for the data. When eaDirect finds a piece of data on that style page, within the defined search area, matching the defined pattern (called pattern matching), it extracts the data.
Payee	A business entity capable of processing or accepting payments. (Term used with eaPay.)
PPD	Prearranged Payment or Deposit. Automated consumer payment application by which a consumer can authorize debits or credits to his/her account by a company or financial. eaPay supports this type of ACH payment. (Term used with eaPay.)

Primary Key	The mandatory field defined on the document style page in the DDF, used to identify the individual or business associated with a statement in the input data source. The primary key is often the account number, however, in the case of statements for recipients with multiple account numbers, another data element must be selected as the Primary Key. Each Primary Key Field instance must be unique to the recipient.
Production	Running eaDirect application jobs on a regular basis to provide current account information on demand to Web users.
Profile	Defines the business logic or market segmentation behind a promotion (the basis on which a promotion occurs). Profiles are created using Composer in eaDirect. (Term used with eaMarket.)
Promotion	The specifics of a marketing campaign (the "who, what, when, where and why"). (Term used with eaMarket.)
Publishing	To identify a set of new or updated eaDirect design files (typically a DDF, ALF, and HTML templates) for use by an application in a live production environment. The set of files published is called a view. The production process uses the published view files along with input data to enable end-users to view their account information on the Web.
Purge App	The eaDirect batch job used to remove an application's index, email, reporting, detail, annotations, and dispute data from the database up to a date specified.
Purge Logs	The eaDirect batch job used to remove all rows in the .LOG table up to the date specified.
RDFI	Receiving Depository Financial Institution. The receiving financial institution is an ACH transaction's final destination. Typically, the customers' banks are the RDFIs. (Term used with eaPay.)

Regular Expression	<p>In eaDirect, a programming expression containing the search pattern, or rules, for locating and extracting a data element from an input file. A regular expression is a syntactical combination of symbols that represent a value, and consists of at least one operand (a value) and can have one or more operators (actions).</p> <p>Successful regular expressions, also known as regexes, take advantage of the known characteristics of a data element, such as the general content and relative position of characters (numbers, decimal points, letters, etc.) while accounting for expected unknowns or variables such as exact length. A regular expression can be fairly general or more specific, depending on how exact the pattern needs to be to reliably extract the correct data.</p>
RMI	<p>Java Remote Method Invocation. A distributed object model for Java program to Java program, in which the methods of remote objects written in the Java programming language can be invoked from other Java virtual machines, possibly on different hosts.</p>
Schedule	<p>In eaDirect, when a job runs; in eaMarket, when a promotion runs.</p>
Search Area	<p>The area on a document page in an input data source, identified by start and end coordinates of rows and columns, where eaDirect looks for the start of a particular data element for identification (page styles and markers) or extraction (fields and table columns). The search area for each field must be defined in the DDF during application development. It is not possible to identify the precise location where data starts and ends in an input file, but it is possible to identify an area, relative to the page, where a data element can be found, along with the pattern of data expected.</p>
Section	<p>Piece of screen "real estate" on the HTML template that can be associated with conditions, to allow content of the section to vary based on data. (Term used with eaMarket.)</p>
Service Provider	<p>Business or other entity that provides online account management using one or more eaDirect applications.</p>
Source file	<p>See Input file.</p>

SWIFT	Society for Worldwide Interbank Financial Transfers. (Term used with eaPay.)
Table	A group of related data elements, defined as columns, which could iterate within the statement. Since table data is extracted together from the input data source, it is formatted and presentment together as well. Tables are generally built to extract and present iterative or transactional information, such as telephone or credit card transaction detail. Tables can also be built for multi-row data elements, such as customer addresses and messages. Tables are built using DefTool, and consist of a table anchor, marker pair, and one or more columns.
Table Anchor	A required element of a table, which, when detected during extraction of the table, confirms that data exists on relevant rows. The table anchor can be an internal anchor (a column) or an external anchor (a data string that resides on every row where there is table data and is not extracted as part of the table). See also Anchor .
Task List	Identifies promotions that are pending approval. (Term used with eaMarket.)
User ID	The identification a Web-user enters to log into an eaDirect application.
Version Set	<p>A set of design files (typically a DDF/ALF pair and all statement and group HTML templates) published for use in data processing and live retrieval of statement views on a service provider's online account management site. A version set is a dated instance of a view.</p> <p>A new version sets can be published when the view design files are updated for ongoing presentment and marketing purpose.</p>
View	A set of design files (typically a DDF/ALF pair and all statement and group HTML templates) which result in a particular Web view of a document. An application can have multiple views for different presentment purposes, such as generating dynamic or static HTML-formatted output and email notification. Different views can also be created to show different levels of information in a document, such as statement summary information, statement detail, etc. See also Version Set .

View Name	The name identifying a particular application view.
View Type	The type of output, or view, a version set produces: HTML, CSV, XML, Chart, XSLT, or XML Query.
VM	Virtual Machine. A self-contained operating environment that behaves as if it is a separate computer. For example, Java applets run in a Java Virtual Machine that has no access to the host operating system. Advantages to this design include system independence and security.
Volume	A data input file that has been successfully processed by the Indexer job and referenced in the volumes table. A date processed value in the Date Accepted or the Date Rejected column indicates the volume status (primarily intended for systems using an internal verification process).
XML	Extensible Markup Language. An open standard for identifying data from the W3C. XML is used to define data elements on Web pages and B2B documents. It uses a similar tag structure as HTML; however, HTML defines <i>how</i> elements are displayed and XML defines <i>what</i> those elements contain. HTML uses predefined tags, but XML allows tags to be defined by the page developer. XML can identify virtually any data items, such as product, sales rep and amount due, enabling Web pages to function like database records. By providing a common method for identifying data, XML supports B2B transactions and is expected to become the dominant format for electronic data interchange. XML is a subset of SGML (Standard Generalized Markup Language), while HTML is an application of SGML.
XML Output	A type of eaDirect batch job used to generate XML-formatted email and output files. You publish a DDF file with an XML Output job (for use by both the Indexer and XMLFormatter tasks).
XSL	Extensible Style sheet Language. A style sheet format for XML documents. It is the XML counterpart to the Cascading Style Sheets (CSS) language in HTML, although XML supports CSS1 and CSS2 as well. XSLT (XSL Transformations) are extensions to XSL for converting XML documents into XML or other document types and can be used independently of XSL.

Yield

The ratio of clicks to impressions. For example, 25 clicks and 100 impressions shows a yield of 25%. (Term used with eaMarket.)

Online Glossaries and Databases

Site Name/Type of Information

TechEncyclopedia

Technical and Internet terminology

<http://techweb.com/encyclopedia/>

Webopedia

Online dictionary and search engine for computer and Internet technology

<http://webopedia.com/>

What Is?

IT-specific encyclopedia and database

<http://whatis.techtarget.com/>

Sun Microsystems

Glossary of Java-related technology terms

<http://java.sun.com/docs/glossary.html>

Tradition/Axone Financial Terminology Database

Financial terminology

<http://tradition.axone.ch/>

Encarta

General dictionary and encyclopedia

<http://encarta.msn.com/>

Bea Systems

Bea document archive

<http://edocs.bea.com/index.html>

Index

A

- Access Type, 56
- Action on Index Volume, 47
- Activity log, 107
- Administrator's ID and password, 29, 152
- ALF, 20, 138
- Application
 - deleting, 154
 - development, 13
 - process of setting up new, 13
- Application Logic File, 20

B

- Bill statistics, 147
- Browsing version sets, 136

C

- Changing
 - Administrator's password, 152
 - job configuration, 111
 - job schedule, 97
- Chart view, 19, 90

- Command Center
 - logging in and out, 29
 - Main Console, 95
 - using, 13
- Composer, 13
- Configuring tasks
 - DXLoader, 89
 - Indexer, 39
 - IVNScanner, 53, 86
 - IXLoader, 41
 - MailNotification, 55
 - PurgeActivityData, 60
 - PurgeEmailData, 60
 - PurgeIndexData, 59
 - PurgeLogs, 62
 - Scanner, 38
 - StatementsToIR, 88
 - StaticHtmlFormatter, 71
 - XMLFormatter, 80
- Creating a new application, 31
- Creating and configuring jobs
 - Detail Extractor, 18, 81

- EmailNotification, 17, 47
- HTML Output (static), 18, 62
- Indexer, 16, 32
- Purge App, 17, 57, 151
- Purge Logs, 17, 60, 152
- XML Output (static), 18
- CSV view, 19, 90

D

- Data
 - extraction, 19
 - index, 16, 39
 - input file, 39
 - moving the data input file, 38
 - presentation, 20
 - purging, 57, 60, 151, 152
- Data Definition File, 19
- Database
 - administration, 151
 - cleaning up server, 57, 60
- DDF, 19, 26, 27, 138
- DDF Path, 40
- DefTool, 13
- Deleting
 - application, 154
 - job, 112
 - version set, 140
- Design files, 13, 19

- Detail data, purging, 57, 151
- Detail Extractor
 - job, 18, 81
 - publishing version sets for, 116, 132
 - views, 25
- Detail page views, 19
- Doc Date, 40
- DXLoader task, 89
- Dynamic Web views
 - HTML, CSV, XML, Chart, XSLT, XML Query, 19, 90
 - publishing, 27, 115

E

- eaDirect, 8
- eaMarket, 9
- eaPay, 8
- eaPost, 9
- eaService, 8
- eaSuite, 7
- edocs Technical Support, 11
- Email data, purging, 57, 151
- EmailNotification
 - job, 17, 47
 - publishing version sets for, 116, 119
 - views, 24
- Enroll Model, 56, 88
- Error

log, 107
 Extraction rules, 19

F

Failed jobs, 103, 106
 Fields to index, 40
 Files
 design, 21
 publishing, 21, 26

G

Glossary, 185

H

Help, 10
 Historical statements, 137
 HTML
 Output job, 18, 62
 Output views (static), 25
 view, 19, 90
 HTML Output
 publishing version sets for, 116, 122

I

Index data
 creating, 16, 39
 purging, 57, 151
 Index Field List, 40
 Index Volume Status, 54, 87
 Indexer
 job, 16, 32

publishing version sets for, 116
 task, 39

Information log, 107
 Input File Path, 38
 Intermediate Representation (IR) file, 39, 88
 IVNScanner task, 53, 86
 IXLoader task, 41

J

J2EE, 9
 Jobs
 changing a configuration, 111
 deleting, 112
 Detail Extractor, 18, 81
 EmailNotification, 17, 47
 failed, 103, 106
 HTML Output (static), 18, 62
 Indexer, 16, 32
 Purge App, 17, 57
 Purge Logs, 17, 60
 scheduling, 97
 status, 101, 103
 Type, 101
 viewing output, 110
 which to create, 15
 XML Output (static), 18

L

Last Run, 101

- Live retrieval, 16, 32, 115, 135
 - Load Method, 43, 89
 - Log data, purging, 60, 152
 - Logs (Error, Activity, Information), 107
- M**
- MailNotification task, 55
 - Message
 - logs, 107
 - Metadata, 39
 - Missing services, 114
 - Monitoring System Services, 113
- N**
- New application
 - creating, 31
 - process of setting up, 13
 - Next Run, 101
 - No operation, 103
- O**
- Old version sets, 137
 - Optional Field Count, 43
 - Output File Path, 38, 89
- P**
- Password, changing Administrator's, 152
 - Presentation rules, 20
 - Production
 - reports, 143
 - scheduling, 97
 - troubleshooting, 157
- Publishing**
- design files, 21
 - dynamic Web views, 27, 90, 115
 - files with job
 - configurations, 26
 - version sets, 116
 - what to publish, 26
- Purge App job, 17, 57, 151
- Purge Logs job, 17, 60, 152
- PurgeActivityData task, 60
- PurgeEmailData task, 60
- PurgeIndexData task, 59
- PurgeLogs task, 62
- Purging**
- Index, email, reporting, and detail data, 57, 151
 - log data, 60, 152
- R**
- Recurring, 103
 - Removing data from the database
 - Index, email, reporting, and detail data, 57, 151
 - log data, 60, 152
 - Reporting data, purging, 57, 151

- Reports, 143
 - Reprocessing, 103
 - Rules
 - for extracting data, 19
 - for presenting data, 20
 - Run Time, 101
- S**
- Scanner task, 38
 - Scheduling jobs, 97
 - Skip Rows, 42
 - Split Size, 42, 44
 - StatementsToIR task, 88
 - Static HTML Output
 - job, 62
 - views, 25
 - StaticHtmlFormatter task, 71
 - Statistics
 - bill, 147
 - user, 143
 - Status, 101
 - job, 103
 - task, 104
 - Summary page views, 19
 - Support, 11
 - System Services, 113
- T**
- Task status, 104
 - Technical support, 11
 - Terms and acronyms, 185
 - Troubleshooting, 157
- U**
- User statistics, 143
 - Using version sets, 115
- V**
- Version sets, 27, 115
 - browsing, 136
 - deleting, 140
 - historical, 137
 - how eaDirect selects
 - for live retrieval, 135
 - publishing, 116
 - viewing file contents, 138
 - Viewing job output, 110
 - Views
 - about, 19
 - Detail Extractor, 25
 - detail page, 19
 - dynamic HTML, CSV, XML, Chart, XSLT, XML Query, 19
 - EmailNotification, 24
 - publishing, 90
 - Static HTML Output, 25
 - summary page, 19
- W**
- Web views. *See Views.. See Views.*
- X**
- XML
 - Output job, 18
 - view, 19, 90

XML Output
publishing version sets
for, 116, 127
XML Query
view, 19, 90

XMLFormatter task, 80
XSLT
view, 19, 90
XSLT style sheet, 25, 27,
81, 88