
JD Edwards EnterpriseOne Condition-Based Maintenance 9.0 Implementation Guide

September 2008

Copyright © 2003-2008, Oracle and/or its affiliates. All rights reserved.

Trademark Notice

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

License Restrictions Warranty/Consequential Damages Disclaimer

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

Subject to patent protection under one or more of the following U.S. patents: 5,781,908; 5,828,376; 5,950,010; 5,960,204; 5,987,497; 5,995,972; 5,987,497; and 6,223,345. Other patents pending.

Warranty Disclaimer

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

Restricted Rights Notice

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are “commercial computer software” or “commercial technical data” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

Hazardous Applications Notice

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Third Party Content, Products, and Services Disclaimer

This software and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third party content, products and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third party content, products or services.

Contains GNU libgmp library; Copyright © 1991 Free Software Foundation, Inc. This library is free software which can be modified and redistributed under the terms of the GNU Library General Public License.

Includes Adobe® PDF Library, Copyright 1993-2001 Adobe Systems, Inc. and DL Interface, Copyright 1999-2008 Datalogics Inc. All rights reserved. Adobe® is a trademark of Adobe Systems Incorporated.

Portions of this program contain information proprietary to Microsoft Corporation. Copyright 1985-1999 Microsoft Corporation.

Portions of this program contain information proprietary to Tenberry Software, Inc. Copyright 1992-1995 Tenberry Software, Inc.

Portions of this program contain information proprietary to Premia Corporation. Copyright 1993 Premia Corporation.

This product includes code licensed from RSA Data Security. All rights reserved.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>).

This product includes cryptographic software written by Eric Young (ey@cryptsoft.com).

This product includes software written by Tim Hudson (tjh@cryptsoft.com). All rights reserved.

This product includes the Sentry Spelling-Checker Engine, Copyright 1993 Wintertree Software Inc. All rights reserved.

Open Source Disclosure

Oracle takes no responsibility for its use or distribution of any open source or shareware software or documentation and disclaims any and all liability or damages resulting from use of said software or documentation. The following open source software may be used in Oracle's JD Edwards EnterpriseOne products and the following disclaimers are provided:

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>). Copyright (c) 1999-2000 The Apache Software Foundation. All rights reserved. THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Contents

General Preface

About This Documentation Preface	ix
JD Edwards EnterpriseOne Application Prerequisites.....	ix
Application Fundamentals.....	ix
Documentation Updates and Downloading Documentation.....	x
Obtaining Documentation Updates.....	x
Downloading Documentation.....	x
Additional Resources.....	x
Typographical Conventions and Visual Cues.....	xi
Typographical Conventions.....	xii
Visual Cues.....	xii
Country, Region, and Industry Identifiers.....	xiii
Currency Codes.....	xiv
Comments and Suggestions.....	xiv
Common Fields Used in Implementation Guides.....	xiv

Preface

JD Edwards EnterpriseOne Condition-Based Maintenance Preface.....	xvii
JD Edwards EnterpriseOne Products.....	xvii
JD Edwards EnterpriseOne Application Fundamentals.....	xvii
Common Fields Used in this Implementation Guide.....	xvii

Chapter 1

Getting Started with JD Edwards EnterpriseOne Condition-Based Maintenance.....	1
JD Edwards EnterpriseOne Condition-Based Maintenance Overview.....	1
JD Edwards EnterpriseOne Condition-Based Maintenance Integrations.....	1
JD Edwards EnterpriseOne Condition-Based Maintenance Implementation.....	2
Global Implementation Steps.....	2
Condition-Based Maintenance Implementation Steps.....	2

Chapter 2

Setting Up Condition-Based Maintenance.....	5
Understanding Condition-Based Maintenance Setup.....	5

Setting Up Sequences for Notification and Investigation Messages.....	5
Understanding Sequencing for Notification and Investigation Messages.....	6
Form Used to Set Up Sequences for Notification and Investigation Messages.....	6
Setting Processing Options for the Condition-Based Maintenance Message Sequences Program (P1315).....	6
Setting Up Message Sequences.....	7
Setting Up Rules for Notification and Investigation Messages.....	8
Understanding Rules for Notification and Investigation Messages.....	8
Forms Used to Set Up Rules for Notification and Investigation Messages.....	8
Setting Processing Options for the Condition-Based Maintenance Message Rules Program (P1316).....	8
Setting Up Message Rules.....	8
Setting Up Sequences for Alert Action Rules.....	10
Understanding Sequencing for Alert Action Rules.....	10
Form Used to Set Up Sequences for Alert Action Rules.....	10
Setting Up Sequences for Alert Action Rules.....	10
Setting Up Rules for Alert Actions.....	11
Understanding Alert Action Rules.....	11
Forms Used to Set Up Rules for Alert Actions.....	12
Setting Processing Options for the Condition-Based Maintenance Alert Action Rules Program (P1318).....	12
Setting Up Alert Action Rules.....	12

Chapter 3

Working with Condition-Based Alerts.....	17
Understanding Condition-Based Alerts.....	17
Entering Condition-Based Alerts.....	19
Understanding Condition-Based Alert Entry.....	19
Form Used to Enter Condition-Based Alerts Manually.....	19
Setting Processing Options for the Condition-Based Alerts Workbench Program (P1310).....	20
Setting Processing Options for the Condition-Based Alerts Revisions Program (P1311).....	22
Entering Condition-Based Alerts Manually.....	23
Responding to Condition-Based Alerts.....	25
Understanding Responses to Condition-Based Alerts.....	25
Forms Used to Respond to Condition-Based Maintenance Alerts.....	27
Entering Responses to Condition-Based Alerts.....	27
Initiating an Investigation Message.....	27
Creating a Work Order.....	28
Updating the Preventive Maintenance Schedule.....	29

Processing Condition-Based Alerts.....	29
Setting Processing Options for the Condition-Based Alerts Processor Program (R1312).....	29
Reviewing Condition-Based Alerts.....	30
Completing Condition-Based Alerts.....	31
Understanding Condition-Based Alerts Completion.....	31
Forms Used to Complete Condition-Based Alerts.....	32

Appendix A

Delivered Workflow for JD Edwards EnterpriseOne Condition-Based Maintenance.....	33
Delivered Workflow for Condition-Based Maintenance.....	33
Condition-Based Maintenance Notification Process.....	33
Condition-Based Maintenance Investigation Process.....	34

Glossary of JD Edwards EnterpriseOne Terms.....	35
--	-----------

Index	51
--------------------	-----------

About This Documentation Preface

JD Edwards EnterpriseOne implementation guides provide you with the information that you need to implement and use JD Edwards EnterpriseOne applications from Oracle.

This preface discusses:

- JD Edwards EnterpriseOne application prerequisites.
- Application fundamentals.
- Documentation updates and downloading documentation.
- Additional resources.
- Typographical conventions and visual cues.
- Comments and suggestions.
- Common fields in implementation guides.

Note. Implementation guides document only elements, such as fields and check boxes, that require additional explanation. If an element is not documented with the process or task in which it is used, then either it requires no additional explanation or it is documented with common fields for the section, chapter, implementation guide, or product line. Fields that are common to all JD Edwards EnterpriseOne applications are defined in this preface.

JD Edwards EnterpriseOne Application Prerequisites

To benefit fully from the information that is covered in these books, you should have a basic understanding of how to use JD Edwards EnterpriseOne applications.

You might also want to complete at least one introductory training course, if applicable.

You should be familiar with navigating the system and adding, updating, and deleting information by using JD Edwards EnterpriseOne menus, forms, or windows. You should also be comfortable using the World Wide Web and the Microsoft Windows or Windows NT graphical user interface.

These books do not review navigation and other basics. They present the information that you need to use the system and implement your JD Edwards EnterpriseOne applications most effectively.

Application Fundamentals

Each application implementation guide provides implementation and processing information for your JD Edwards EnterpriseOne applications.

For some applications, additional, essential information describing the setup and design of your system appears in a companion volume of documentation called the application fundamentals implementation guide. Most product lines have a version of the application fundamentals implementation guide. The preface of each implementation guide identifies the application fundamentals implementation guides that are associated with that implementation guide.

The application fundamentals implementation guide consists of important topics that apply to many or all JD Edwards EnterpriseOne applications. Whether you are implementing a single application, some combination of applications within the product line, or the entire product line, you should be familiar with the contents of the appropriate application fundamentals implementation guides. They provide the starting points for fundamental implementation tasks.

Documentation Updates and Downloading Documentation

This section discusses how to:

- Obtain documentation updates.
- Download documentation.

Obtaining Documentation Updates

You can find updates and additional documentation for this release, as well as previous releases, on Oracle's PeopleSoft Customer Connection website. Through the Documentation section of Oracle's PeopleSoft Customer Connection, you can download files to add to your Implementation Guides Library. You'll find a variety of useful and timely materials, including updates to the full line of JD Edwards EnterpriseOne documentation that is delivered on your implementation guides CD-ROM.

Important! Before you upgrade, you must check Oracle's PeopleSoft Customer Connection for updates to the upgrade instructions. Oracle continually posts updates as the upgrade process is refined.

See Also

Oracle's PeopleSoft Customer Connection, http://www.oracle.com/support/support_peoplesoft.html

Downloading Documentation

In addition to the complete line of documentation that is delivered on your implementation guide CD-ROM, Oracle makes JD Edwards EnterpriseOne documentation available to you via Oracle's website. You can download PDF versions of JD Edwards EnterpriseOne documentation online via the Oracle Technology Network. Oracle makes these PDF files available online for each major release shortly after the software is shipped.

See Oracle Technology Network, <http://www.oracle.com/technology/documentation/psftent.html>

Additional Resources

The following resources are located on Oracle's PeopleSoft Customer Connection website:

Resource	Navigation
Application maintenance information	Updates + Fixes
Business process diagrams	Support, Documentation, Business Process Maps

Resource	Navigation
Interactive Services Repository	Support, Documentation, Interactive Services Repository
Hardware and software requirements	Implement, Optimize + Upgrade; Implementation Guide; Implementation Documentation and Software; Hardware and Software Requirements
Installation guides	Implement, Optimize + Upgrade; Implementation Guide; Implementation Documentation and Software; Installation Guides and Notes
Integration information	Implement, Optimize + Upgrade; Implementation Guide; Implementation Documentation and Software; Pre-Built Integrations for PeopleSoft Enterprise and JD Edwards EnterpriseOne Applications
Minimum technical requirements (MTRs)	Implement, Optimize + Upgrade; Implementation Guide; Supported Platforms
Documentation updates	Support, Documentation, Documentation Updates
Implementation guides support policy	Support, Support Policy
Prerelease notes	Support, Documentation, Documentation Updates, Category, Release Notes
Product release roadmap	Support, Roadmaps + Schedules
Release notes	Support, Documentation, Documentation Updates, Category, Release Notes
Release value proposition	Support, Documentation, Documentation Updates, Category, Release Value Proposition
Statement of direction	Support, Documentation, Documentation Updates, Category, Statement of Direction
Troubleshooting information	Support, Troubleshooting
Upgrade documentation	Support, Documentation, Upgrade Documentation and Scripts

Typographical Conventions and Visual Cues

This section discusses:

- Typographical conventions.
- Visual cues.
- Country, region, and industry identifiers.
- Currency codes.

Typographical Conventions

This table contains the typographical conventions that are used in implementation guides:

Typographical Convention or Visual Cue	Description
Bold	Indicates PeopleCode function names, business function names, event names, system function names, method names, language constructs, and PeopleCode reserved words that must be included literally in the function call.
<i>Italics</i>	Indicates field values, emphasis, and JD Edwards EnterpriseOne or other book-length publication titles. In PeopleCode syntax, italic items are placeholders for arguments that your program must supply. We also use italics when we refer to words as words or letters as letters, as in the following: Enter the letter <i>O</i> .
KEY+KEY	Indicates a key combination action. For example, a plus sign (+) between keys means that you must hold down the first key while you press the second key. For ALT+W, hold down the ALT key while you press the W key.
Monospace font	Indicates a PeopleCode program or other code example.
“ ” (quotation marks)	Indicate chapter titles in cross-references and words that are used differently from their intended meanings.
. . . (ellipses)	Indicate that the preceding item or series can be repeated any number of times in PeopleCode syntax.
{ } (curly braces)	Indicate a choice between two options in PeopleCode syntax. Options are separated by a pipe ().
[] (square brackets)	Indicate optional items in PeopleCode syntax.
& (ampersand)	When placed before a parameter in PeopleCode syntax, an ampersand indicates that the parameter is an already instantiated object. Ampersands also precede all PeopleCode variables.

Visual Cues

Implementation guides contain the following visual cues.

Notes

Notes indicate information that you should pay particular attention to as you work with the JD Edwards EnterpriseOne system.

Note. Example of a note.

If the note is preceded by *Important!*, the note is crucial and includes information that concerns what you must do for the system to function properly.

Important! Example of an important note.

Warnings

Warnings indicate crucial configuration considerations. Pay close attention to warning messages.

Warning! Example of a warning.

Cross-References

Implementation guides provide cross-references either under the heading “See Also” or on a separate line preceded by the word *See*. Cross-references lead to other documentation that is pertinent to the immediately preceding documentation.

Country, Region, and Industry Identifiers

Information that applies only to a specific country, region, or industry is preceded by a standard identifier in parentheses. This identifier typically appears at the beginning of a section heading, but it may also appear at the beginning of a note or other text.

Example of a country-specific heading: “(FRA) Hiring an Employee”

Example of a region-specific heading: “(Latin America) Setting Up Depreciation”

Country Identifiers

Countries are identified with the International Organization for Standardization (ISO) country code.

Region Identifiers

Regions are identified by the region name. The following region identifiers may appear in implementation guides:

- Asia Pacific
- Europe
- Latin America
- North America

Industry Identifiers

Industries are identified by the industry name or by an abbreviation for that industry. The following industry identifiers may appear in implementation guides:

- USF (U.S. Federal)

- E&G (Education and Government)

Currency Codes

Monetary amounts are identified by the ISO currency code.

Comments and Suggestions

Your comments are important to us. We encourage you to tell us what you like, or what you would like to see changed about implementation guides and other Oracle reference and training materials. Please send your suggestions to your product line documentation manager at Oracle Corporation, 500 Oracle Parkway, Redwood Shores, CA 94065, U.S.A. Or email us at appsdoc@us.oracle.com.

While we cannot guarantee to answer every email message, we will pay careful attention to your comments and suggestions.

Common Fields Used in Implementation Guides

Address Book Number	Enter a unique number that identifies the master record for the entity. An address book number can be the identifier for a customer, supplier, company, employee, applicant, participant, tenant, location, and so on. Depending on the application, the field on the form might refer to the address book number as the customer number, supplier number, or company number, employee or applicant ID, participant number, and so on.
As If Currency Code	Enter the three-character code to specify the currency that you want to use to view transaction amounts. This code enables you to view the transaction amounts as if they were entered in the specified currency rather than the foreign or domestic currency that was used when the transaction was originally entered.
Batch Number	Displays a number that identifies a group of transactions to be processed by the system. On entry forms, you can assign the batch number or the system can assign it through the Next Numbers program (P0002).
Batch Date	Enter the date in which a batch is created. If you leave this field blank, the system supplies the system date as the batch date.
Batch Status	<p>Displays a code from user-defined code (UDC) table 98/IC that indicates the posting status of a batch. Values are:</p> <p><i>Blank:</i> Batch is unposted and pending approval.</p> <p><i>A:</i> The batch is approved for posting, has no errors and is in balance, but has not yet been posted.</p> <p><i>D:</i> The batch posted successfully.</p> <p><i>E:</i> The batch is in error. You must correct the batch before it can post.</p>

P: The system is in the process of posting the batch. The batch is unavailable until the posting process is complete. If errors occur during the post, the batch status changes to *E*.

U: The batch is temporarily unavailable because someone is working with it, or the batch appears to be in use because a power failure occurred while the batch was open.

Branch/Plant	Enter a code that identifies a separate entity as a warehouse location, job, project, work center, branch, or plant in which distribution and manufacturing activities occur. In some systems, this is called a business unit.
Business Unit	Enter the alphanumeric code that identifies a separate entity within a business for which you want to track costs. In some systems, this is called a branch/plant.
Category Code	Enter the code that represents a specific category code. Category codes are user-defined codes that you customize to handle the tracking and reporting requirements of your organization.
Company	Enter a code that identifies a specific organization, fund, or other reporting entity. The company code must already exist in the F0010 table and must identify a reporting entity that has a complete balance sheet.
Currency Code	Enter the three-character code that represents the currency of the transaction. JD Edwards EnterpriseOne provides currency codes that are recognized by the International Organization for Standardization (ISO). The system stores currency codes in the F0013 table.
Document Company	<p>Enter the company number associated with the document. This number, used in conjunction with the document number, document type, and general ledger date, uniquely identifies an original document.</p> <p>If you assign next numbers by company and fiscal year, the system uses the document company to retrieve the correct next number for that company.</p> <p>If two or more original documents have the same document number and document type, you can use the document company to display the document that you want.</p>
Document Number	Displays a number that identifies the original document, which can be a voucher, invoice, journal entry, or time sheet, and so on. On entry forms, you can assign the original document number or the system can assign it through the Next Numbers program.
Document Type	<p>Enter the two-character UDC, from UDC table 00/DT, that identifies the origin and purpose of the transaction, such as a voucher, invoice, journal entry, or time sheet. JD Edwards EnterpriseOne reserves these prefixes for the document types indicated:</p> <p><i>P</i>: Accounts payable documents.</p> <p><i>R</i>: Accounts receivable documents.</p> <p><i>T</i>: Time and pay documents.</p> <p><i>I</i>: Inventory documents.</p> <p><i>O</i>: Purchase order documents.</p> <p><i>S</i>: Sales order documents.</p>

Effective Date

Enter the date on which an address, item, transaction, or record becomes active. The meaning of this field differs, depending on the program. For example, the effective date can represent any of these dates:

- The date on which a change of address becomes effective.
- The date on which a lease becomes effective.
- The date on which a price becomes effective.
- The date on which the currency exchange rate becomes effective.
- The date on which a tax rate becomes effective.

Fiscal Period and Fiscal Year

Enter a number that identifies the general ledger period and year. For many programs, you can leave these fields blank to use the current fiscal period and year defined in the Company Names & Number program (P0010).

G/L Date (general ledger date)

Enter the date that identifies the financial period to which a transaction will be posted. The system compares the date that you enter on the transaction to the fiscal date pattern assigned to the company to retrieve the appropriate fiscal period number and year, as well as to perform date validations.

JD Edwards EnterpriseOne Condition-Based Maintenance Preface

This preface discusses:

- JD Edwards EnterpriseOne products.
- JD Edwards EnterpriseOne application fundamentals.
- Common fields used in this implementation guide.

JD Edwards EnterpriseOne Products

This implementation guide refers to these JD Edwards EnterpriseOne products from Oracle:

- JD Edwards EnterpriseOne Condition-Based Maintenance.
- JD Edwards EnterpriseOne Capital Asset Management.

JD Edwards EnterpriseOne Application Fundamentals

Additional, essential information describing the setup and design of your system appears in a companion volume of documentation called *JD Edwards EnterpriseOne Financial Management Application Fundamentals Implementation Guide*.

Customers must conform to the supported platforms for the release as detailed in the JD Edwards EnterpriseOne minimum technical requirements. In addition, JD Edwards EnterpriseOne may integrate, interface, or work in conjunction with other Oracle products. Refer to the cross-reference material in the Program Documentation at <http://oracle.com/contracts/index.html> for Program prerequisites and version cross-reference documents to assure compatibility of various Oracle products.

Common Fields Used in this Implementation Guide

Alert Level

Enter a code to indicate the functional status of a piece of equipment, as assessed by a condition-based maintenance system. For maintenance schedules that are condition-based, the preventive maintenance system uses this code to determine whether maintenance is due. Enter a value from user-defined code (UDC) table 13/AL (Alert Level).

This field enables you to further define the message rule to determine who gets notification or investigation messages for each alert level.

Alert Status	Enter a value from UDC 13/AS (Alert Status) that indicates the status of a condition-based alert.
Investigation Recipient	Enter an address book number to identify the recipient of an alert investigation request. This number can be a single person or a distribution list. The system verifies this number against the JD Edwards EnterpriseOne Address Book system.
Investigation Structure Type	Enter a code to identify a type of organizational structure that has its own hierarchy in the JD Edwards EnterpriseOne Address Book system (for example, email). Values are defined in UDC 01/TS (Parent/Child Structure Type).
Notification Recipient	Enter an address book number to identify the recipient of an alert notification message. This number can be a single person or a distribution list. The system verifies this number against the JD Edwards EnterpriseOne Address Book system.
Notification Structure Type	Enter a code to identify a type of organizational structure that has its own hierarchy in the JD Edwards EnterpriseOne Address Book system (for example, email). Values are defined in UDC 01/TS (Parent/Child Structure Type).
Service Type	<p>Enter a value from UDC 12/ST (Service Types) that indicates the condition-based maintenance service or preventive maintenance service to be performed. Examples of codes include TIRES for replace tires, ENGINE for engine overhaul, CLUTCH for adjust clutch and REBUILD for rebuild equipment.</p> <p>The system uses this code, in conjunction with the equipment number, to obtain the default rules from the Maintenance Rules table (F1393) when creating a work order and to update the Maintenance Schedule (F1207) table with the alert level from the condition based alert.</p>

CHAPTER 1

Getting Started with JD Edwards EnterpriseOne Condition-Based Maintenance

This chapter discusses:

- JD Edwards EnterpriseOne Condition-Based Maintenance overview.
- JD Edwards EnterpriseOne Condition-Based Maintenance integrations.
- JD Edwards EnterpriseOne Condition-Based Maintenance implementation.

JD Edwards EnterpriseOne Condition-Based Maintenance Overview

Unplanned downtime can create a huge dent in a company's productivity and profitability. That is why it is so important to keep equipment in top working condition. Traditional maintenance programs are no longer sufficient; companies must be able to find and fix problems before they cause equipment breakdowns.

JD Edwards EnterpriseOne Condition-Based Maintenance (CBM) from Oracle enables you to make maintenance decisions based on actual condition, rather than time or usage interval. You can identify equipment problems early, when they are less costly to correct, and perform maintenance only when needed, thereby increasing asset utilization, extending equipment life, and reducing maintenance costs. JD Edwards EnterpriseOne CBM enables you to quickly respond to and resolve equipment issues instead of waiting for the equipment to fail. JD Edwards EnterpriseOne CBM:

- Uses alerts generated from realtime equipment monitoring devices that assess equipment conditions against predefined, normal operation parameters.
- Provides realtime alerts (using pager, email, or other messaging systems) when equipment is operating outside of normal bounds so appropriate maintenance action can be taken.
- Automatically initiates appropriate maintenance action, including investigation, creation of a work order, and maintenance schedule updates.

JD Edwards EnterpriseOne Condition-Based Maintenance Integrations

The JD Edwards EnterpriseOne CBM system integrates with the JD Edwards EnterpriseOne Capital Asset Management (CAM) system from Oracle using these modules:

- JD Edwards EnterpriseOne Work Order module.

- JD Edwards EnterpriseOne Preventive Maintenance module.

The JD Edwards EnterpriseOne CBM system works with other JD Edwards EnterpriseOne systems to ensure that all information is fully integrated. We discuss integration considerations in the implementation chapters in this implementation guide. Supplemental information about third-party application integrations is located on the Customer Connection website.

JD Edwards EnterpriseOne CAM

The JD Edwards EnterpriseOne CAM system enables you to maintain and service equipment using the right mix of reactive, preventive, and predictive maintenance. This enables you to provide the required equipment reliability at the lowest possible cost.

JD Edwards EnterpriseOne Condition-Based Maintenance Implementation

This section provides an overview of the steps that are required to implement JD Edwards EnterpriseOne CBM.

In the planning phase of the implementation, take advantage of all JD Edwards EnterpriseOne sources of information, including the installation guides and troubleshooting information. A complete list of these resources appears in the preface in *About This Documentation* with information about where to find the most current version of each.

When determining which electronic software updates (ESUs) for JD Edwards EnterpriseOne CBM, use the EnterpriseOne and World Change Assistant. EnterpriseOne and World Change Assistant, a Java-based tool, reduces the time required to search and download ESUs by 75 percent or more and enables you to install multiple ESUs at one time.

See *JD Edwards EnterpriseOne Tools 8.98 Software Update Guide*

Global Implementation Steps

This table lists the suggested global implementation steps for JD Edwards EnterpriseOne CBM:

Step	Reference
1. Set up companies, fiscal date patterns, and business units.	<i>JD Edwards EnterpriseOne Financial Management Application Fundamentals 9.0 Implementation Guide</i> , "Setting Up Organizations"
2. Set up accounts, and the chart of accounts.	<i>JD Edwards EnterpriseOne Financial Management Application Fundamentals 9.0 Implementation Guide</i> , "Setting Up Bank Accounts" and <i>JD Edwards EnterpriseOne Financial Management Application Fundamentals 9.0 Implementation Guide</i> , "Creating the Chart of Accounts"
3. Enter address book records.	<i>JD Edwards EnterpriseOne Address Book 9.0 Implementation Guide</i> , "Entering Address Book Records"

Condition-Based Maintenance Implementation Steps

This table lists the suggested application-specific implementation steps for JD Edwards EnterpriseOne CBM:

Step	Reference
1. Set up equipment master records.	<i>JD Edwards EnterpriseOne Capital Asset Management 9.0 Implementation Guide, "Setting Up Equipment"</i>
2. Set up preventive maintenance schedules.	<i>JD Edwards EnterpriseOne Capital Asset Management 9.0 Implementation Guide, "Setting Up Preventive Maintenance"</i>
3. Set up work orders.	<i>JD Edwards EnterpriseOne Capital Asset Management 9.0 Implementation Guide, "Setting Up Work Orders"</i>
4. Set up sequences for notification and investigation messages.	<u>Chapter 2, "Setting Up Condition-Based Maintenance," Setting Up Sequences for Notification and Investigation Messages, page 5</u>
5. Set up rules for notification and investigation messages.	<u>Chapter 2, "Setting Up Condition-Based Maintenance," Setting Up Rules for Notification and Investigation Messages, page 8</u>
6. Set up sequences for alert action rules.	<u>Chapter 2, "Setting Up Condition-Based Maintenance," Setting Up Sequences for Alert Action Rules, page 10</u>
7. Set up alert action rules.	<u>Chapter 2, "Setting Up Condition-Based Maintenance," Setting Up Rules for Alert Actions, page 11</u>

CHAPTER 2

Setting Up Condition-Based Maintenance

This chapter provides an overview of JD Edwards EnterpriseOne Condition-Based Maintenance (CBM) setup and discusses how to:

- Set up sequences for notification and investigation messages.
- Set up rules for notification and investigation messages.
- Set up sequences for alert action rules.
- Set up rules for alert actions.

Understanding Condition-Based Maintenance Setup

Before you can use the features of JD Edwards EnterpriseOne CBM, you need to define the information that is critical for system processes.

In addition to the setup topics that are included in this guide, refer to topics in these guides for additional setup and update options:

See Also

JD Edwards EnterpriseOne Capital Asset Management 9.0 Implementation Guide, "Setting Up Work Orders"

JD Edwards EnterpriseOne Capital Asset Management 9.0 Implementation Guide, "Setting Up Preventive Maintenance"

JD Edwards EnterpriseOne Capital Asset Management 9.0 Implementation Guide, "Working with Preventive Maintenance Schedules"

JD Edwards EnterpriseOne Application Fundamentals Interoperability 9.0 Implementation Guide, "Processing Interoperability for Condition-Based Maintenance"

JD Edwards EnterpriseOne Tools 8.98 Workflow Tools Guide

Setting Up Sequences for Notification and Investigation Messages

This section provides an overview of sequencing for notification and investigation messages and discusses how to:

- Set processing options for the Condition-Based Maintenance Message Sequences program (P1315).

- Set up message sequences.

Understanding Sequencing for Notification and Investigation Messages

Use the Condition-Based Maintenance Message Sequences program to define the search sequences that the system uses to select the message rule that determines the alert message recipients. The hierarchy determines the order in which the system searches for message recipients.

You can use a processing option to specify whether to enter message sequences for a notification process or for an investigation process. Notification messages warn or alert managers (such as production, shipping, and plant supervisors) of a problem. Investigation messages request that a plant technician investigate the problem and report findings back to the message originator.

Observe these guidelines:

- Set up the hierarchy from specific to general.
- For performance reasons, set up only the necessary message sequences.
- The workflow process CBMNOTIFY supports sending a notification message to an individual recipient or a distribution list.
- The workflow process CBMINVEST supports sending an investigation message to an individual recipient or a distribution list.

See Also

[Appendix A, "Delivered Workflow for JD Edwards EnterpriseOne Condition-Based Maintenance," page 33](#)

JD Edwards EnterpriseOne Tools 8.98 Workflow Tools Guide

Form Used to Set Up Sequences for Notification and Investigation Messages

Form Name	FormID	Navigation	Usage
Condition-Based Maintenance Message Sequences	W1315A	<ul style="list-style-type: none"> • Condition-Based Maintenance Setup (G13CBM41), Notification Message Sequences • Condition-Based Maintenance Setup (G13CBM41), Investigation Message Sequences 	Set up message sequences.

Setting Processing Options for the Condition-Based Maintenance Message Sequences Program (P1315)

Processing options enable you to specify the default processing for programs and reports.

Defaults

- 1. Message Rules Type** Specify the message rules type. Values are listed in user-defined code (UDC) 13/TY (Message Rules Type).

Setting Up Message Sequences

Access the Condition-Based Maintenance Message Sequences form.

Notification Message Sequences - Condition-Based Maintenance Message Sequences

OK Find Delete Cancel Tools

Message Rules Type Notification Rules

Records 1 - 3 Customize Grid

	Sequence Number	Customer	Site	Business Unit	Location	Product Family	Product Model	Equipment Number	Inventory Number
1	1.00	N	N	N	N	N	N	Y	N
2	2.00	N	N	Y	N	N	N	N	N
3									

Condition-Based Maintenance Message Sequences form

- Sequence Number** Enter a number to specify the order of a group of records on the form.
- Customer** Enter *Y* or *N* to indicate whether the customer value is used as part of the criteria to search for a message recipient.
- Site** Enter *Y* or *N* to indicate whether the site value is used as part of the criteria to search for a message recipient.
- Business Unit** Enter *Y* or *N* to indicate whether the business unit value is used as part of the criteria to search for a message recipient.
- Location** Enter *Y* or *N* to indicate whether the location value is used as part of the criteria to search for a message recipient.
- Product Family** Enter *Y* or *N* to indicate whether the product family is used as part of the criteria to search for a message recipient.
- Product Model** Enter *Y* or *N* to indicate whether the product model value is used as part of the criteria to search for a message recipient.
- Equipment Number** Enter *Y* or *N* indicate whether the equipment number value is used as part of the criteria to search for a message recipient.
-
- Note.** This field heading changes according to which number is primary in the Equipment Constants program (that is, Equipment Number, Unit Number, or Serial Number).
-
- Inventory Number** Enter *Y* or *N* to indicate whether the inventory number value is used as part of the criteria to search for a message recipient.

Setting Up Rules for Notification and Investigation Messages

This section provides an overview of rules for notification and investigation messages and discusses how to:

- Set processing options for the Condition-Based Maintenance Message Rules program (P1316).
- Set up message rules.

Understanding Rules for Notification and Investigation Messages

Use the Condition-Based Maintenance Message Rules program (P1316) to define the search criteria and related message recipients for notification and investigation of alerts. The system uses information from the asset master and the alert to search through the rules and identify the message recipients.

Forms Used to Set Up Rules for Notification and Investigation Messages

Form Name	FormID	Navigation	Usage
Condition-Based Maintenance Message Rules Revisions	W1316B	Condition-Based Maintenance Setup (G13CBM41), Condition-Based Maintenance Message Rules Click Add on the Work With Condition-Based Maintenance Message Rules form. Select a sequence option on the Condition-Based Maintenance Sequences Search and Select form.	Set up message rules.
Condition-Based Maintenance Sequences Search and Select	W1315B	Click Add on the Work With Condition-Based Maintenance Message Rules form.	Select a sequence option for which you want to create a message rule.

Setting Processing Options for the Condition-Based Maintenance Message Rules Program (P1316)

Processing options enable you to specify the default processing for programs and reports.

Versions

- 1. Equipment Search/Select (P17012S) Version** Specify the version of the Equipment Search/Select program (P17012S) that the system uses. If you leave this processing option blank, the system uses ZJDE0001.

Setting Up Message Rules

Access the Condition-Based Maintenance Message Rules Revisions form.

Condition-Based Maintenance Message Rules - Condition-Based Maintenance Message Rules Revisions

OK Cancel Tools

Message Rules Type Notification Rules

Customer Number

Site Number

Business Unit

Location

Product Model

Product Family

Equipment Number

Inventory Number

Alert Level

Notification Recipient

Notification Structure Type

Condition-Based Maintenance Message Rules Revisions form

Message Rules Type	Enter a code to indicate the message rules type. Values are listed in UDC 13/TY (Message Rules Type). The system displays either the notification or investigation fields based on the message rule type.
Customer Number	Enter a number to identify an entry in the JD Edwards EnterpriseOne Address Book system, such as employee, applicant, participant, customer, supplier, tenant, or location.
Site Number	Enter an address book number for the lessor, renter, or lending institution.
Business Unit	<p>Enter an alphanumeric code to identify a separate entity within a business for which you want to track costs. For example, a business unit might be a warehouse location, job, project, work center, branch, or plant.</p> <p>You can assign a business unit to a document, entity, or person for purposes of responsibility reporting. For example, the system provides reports of open accounts payable and accounts receivable by business unit to track equipment by responsible department.</p> <p>Business unit security might prevent you from viewing information about business units for which you have no authority.</p>
Location	Enter the current physical location of an asset. The location must have a valid business unit or job number in the Business Unit Master table (F0006).
Product Model and Product Family	<p>Enter a code to classify an inventory item into a model or group for customer service.</p> <p>Examples include <i>Laser Printer</i>, <i>InkJet</i>, or <i>Fax</i>.</p>
Equipment Number	Enter a numeric value up to eight digits that uniquely identifies an asset.

Inventory Number

Enter an inventory item number. The system provides three separate item numbers plus an extensive cross-reference capability to other item numbers to accommodate substitute item numbers, replacements, bar codes, customer numbers, supplier numbers, and so forth. The item numbers are:

Item Number (short): An eight-digit, computer-assigned item number.

2nd Item Number: A 25-digit, free-form, user-defined alphanumeric item number.

3rd Item Number: A 25-digit, free-form, user-defined alphanumeric item number.

Note. The system displays fields based on the message rule type and enables fields based on the sequence rule that is selected.

Setting Up Sequences for Alert Action Rules

This section provides an overview of sequencing for alert action rules and discusses how to set up sequences for alert action rules.

Understanding Sequencing for Alert Action Rules

Use the Condition-Based Maintenance Alert Action Sequences program (P1317) to define the search sequences that the system uses to determine the action to take to respond to the alert message. The hierarchy determines the order in which the system searches for alert actions.

Observe these guidelines:

- Set up the hierarchy from specific to general.
- For performance reasons, set up only the necessary alert actions.

Form Used to Set Up Sequences for Alert Action Rules

Form Name	FormID	Navigation	Usage
Condition-Based Maintenance Alert Action Sequences	W1317A	Condition-Based Maintenance Setup (G13CBM41), Alert Action Rule Sequences	Set up sequences for Alert Action Rules.

Setting Up Sequences for Alert Action Rules

Access the Condition-Based Maintenance Alert Action Sequences form.

The screenshot shows the 'Alert Action Rule Sequences - Condition-Based Maintenance Alert Action Sequences' form. It includes a toolbar with icons for OK, Find, Delete, Cancel, and Tools. Below the toolbar is a grid titled 'Records 1 - 2' with columns: Sequence Number, Customer, Site, Business Unit, Location, Product Family, Product Model, Equipment Number, and Inventory Number. The first record shows the following values: Sequence Number: 1.00, Customer: N, Site: N, Business Unit: N, Location: N, Product Family: N, Product Model: N, Equipment Number: Y, Inventory Number: N.

Condition-Based Maintenance Alert Action Sequences form

Sequence Number	Enter a number to specify the order of a group of records on the form.
Customer	Enter <i>Y</i> or <i>N</i> to indicate whether the customer value is used as part of the criteria to search for alert actions.
Site	Enter <i>Y</i> or <i>N</i> to indicate whether the site value is used as part of the criteria to search for alert actions.
Business Unit	Enter <i>Y</i> or <i>N</i> to indicate whether the business unit value is used as part of the criteria to search for alert actions.
Location	Enter <i>Y</i> or <i>N</i> to indicate whether the location value is used as part of the criteria to search for alert actions.
Product Family	Enter <i>Y</i> or <i>N</i> to indicate whether the product family is used as part of the criteria to search for alert actions.
Product Model	Enter <i>Y</i> or <i>N</i> to indicate whether the product model value is used as part of the criteria to search for alert actions.
Equipment Number	Enter <i>Y</i> or <i>N</i> to indicate whether the equipment number value is used as part of the criteria to search for alert actions.
	<hr/> Note. This field heading changes according to which number is primary in the Equipment Constants program (that is, Equipment Number, Unit Number, or Serial Number). <hr/>
Inventory Number	Enter <i>Y</i> or <i>N</i> to indicate whether the inventory number value is used as part of the criteria to search for alert actions.

Setting Up Rules for Alert Actions

This section provides an overview of alert action rules and discusses how to:

- Set processing options for the Condition-Based Maintenance Alert Action Rules program (P1318).
- Set up alert action rules.

Understanding Alert Action Rules

Use the Condition-Based Maintenance Alert Action Rules program (P1318) to define the search criteria and related actions for alerts. Alert action rules enable you to set up rules based on equipment coding to define what action is required when the alert is created. Options include sending a notification email, sending an investigation email, creating a work order, and updating the alert level on a PM schedule. The system performs the action when you run the Condition-Based Alerts Processor program (R1312).

Forms Used to Set Up Rules for Alert Actions

Form Name	FormID	Navigation	Usage
Condition-Based Maintenance Alert Action Rules Revisions	W1318B	Condition-Based Maintenance Setup (G13CBM41), Alert Action Rules Click Add on the Work With Condition-Based Maintenance Alert Action Rules form. Select a sequence and click Select on the Condition-Based Maintenance Sequences Search and Select form.	Set up alert action rules.

Setting Processing Options for the Condition-Based Maintenance Alert Action Rules Program (P1318)

Processing options enable you to specify the default processing for programs and reports.

Versions

Equipment Search/Select (P17012S) Version Specify the version of the Equipment Search/Select program (P17012S) that the system uses. If you leave this processing option blank, the system uses the ZJDE0001 version.

Setting Up Alert Action Rules

Access the Condition-Based Maintenance Alert Action Rules Revisions form.

Alert Action Rules - Condition-Based Maintenance Alert Action Rules Revisions

OK Cancel Tools

Customer Number

Site Number

Business Unit

Location

Product Model

Product Family

Equipment Number

Inventory Number

Measurement Location

Alert Level

☐ Send Notification Message

Automated Response Type * -- Select One --

Model Work Order

Model Work Order

Service Type

Condition-Based Maintenance

Service Type

Condition-Based Maintenance Alert Action Rules Revisions form

The fields that are available for entry are determined by the Alert Action Rule Sequence that is selected.

Customer Number

Enter a number to identify an entry in the JD Edwards EnterpriseOne Address Book system, such as employee, applicant, participant, customer, supplier, tenant, or location.

Site Number

Enter an address book number for the lessor, renter, or lending institution.

Business Unit

Enter an alphanumeric code to identify a separate entity within a business for which you want to track costs. For example, a business unit might be a warehouse location, job, project, work center, branch, or plant.

You can assign a business unit to a document, entity, or person for purposes of responsibility reporting. For example, the system provides reports of open accounts payable and accounts receivable by business unit to track equipment by responsible department.

Business unit security might prevent you from viewing information about business units for which you have no authority.

Location

Enter the current physical location of an asset. The location must have a valid business unit or job number in the Business Unit Master table (F0006).

Product Model and Product Family

Enter a code to classify an inventory item into a model or group for customer service.

Examples include *Laser Printer*, *InkJet*, and *Fax*.

Equipment Number

Enter a numeric value up to eight digits that uniquely identifies an asset.

Inventory Number	<p>Enter an inventory item number. The system provides three separate item numbers plus an extensive cross-reference capability to other item numbers to accommodate substitute item numbers, replacements, bar codes, customer numbers, supplier numbers, and so forth. The item numbers are:</p> <p><i>Item Number (short):</i> An eight-digit, computer-assigned item number.</p> <p><i>2nd Item Number:</i> A 25-digit, free-form, user-defined alphanumeric item number.</p> <p><i>3rd Item Number:</i> A 25-digit, free-form, user-defined alphanumeric item number.</p>
Measurement Location	<p>Enter a value from UDC 13/LC (Measurement Location) that indicates a measurement location on a piece of equipment. This field enables you to further define the alert action rule to determine whether alert notifications are sent, and what response to use for the incoming alert. If you have not defined an alert action rule that is specific to a particular measurement location, leave this field blank to define an alert action rule that you can use for any measurement location.</p>
Alert Level	<p>Enter a value from UDC 13/LC (Alert Level) that indicates the functional status of a piece of equipment, as assessed by a condition-based maintenance system. This field enables you to further define the alert action rule to determine whether alert notifications are sent, and what response to use for the incoming alert. If you have not defined an alert action rule that is specific to an alert level, leave this field blank to define an alert action rule that you can use for any alert level.</p>
Send Notification Message	<p>Select this option to specify whether a notification message is sent when a condition-based alert is entered into the system.</p>
Automated Response Type	<p>Enter a value that indicates the type of automatic response that is required when a condition-based alert is entered in the Condition-Based Alerts table (F1310). Values are listed in UDC table 13/AR (Automatic Response Type). Values include:</p> <p><i>1 No Automatic Response:</i> This option allows the user to review the alert and manually trigger a response from the Condition-Based Alerts Workbench program (P1310).</p> <p><i>2 Create Investigation Message:</i> This option automatically creates an investigation message using information from the condition-based alert record.</p> <p><i>3 Create WO from Model:</i> This option automatically creates a work order using the model work order information from the condition-based alert record.</p> <p><i>4 Update PM Schedule:</i> This option automatically updates the Maintenance Schedule File table (F1207) using the asset number and service type information from the condition-based alert record.</p>

Note. To perform the automated response-type processing, you must run the Condition-Based Alerts Processor program (R1312).

Model Work Order

Model Work Order

Enter a number that identifies an original document. The system uses the information from the model work order to create the work order for the condition-based maintenance alert.

CHAPTER 3

Working with Condition-Based Alerts

This chapter provides an overview of condition-based alerts and discusses how to:

- Enter condition-based alerts.
- Respond to condition-based alerts.
- Complete condition-based alerts.

Understanding Condition-Based Alerts

In contrast to frequency-based preventive maintenance, condition-based maintenance uses upper and lower limits, tolerances, and other conditions that might cause an equipment or system warning or alarm to provide a system response to these conditions based on user, vendor, or manufacturer criteria.

The JD Edwards EnterpriseOne Condition-Based Maintenance (CBM) programs provide both real-time and batch connections to respond to alerts from equipment monitoring devices. For example, a wholesale food distributor monitors refrigeration equipment within cold storage and freezer rooms. If the temperature rises above a maximum level, the system provides an alert that can generate an investigation message, create a work order to service the equipment, or update a preventive maintenance schedule.

Dynamic systems (electrical, hydraulic, mechanical, or thermal) possess normal signatures when operating correctly. A subtle signature change might mean the onset of a failure mode. The small differences between normal and abnormal signatures are often hidden by noise in the system. However, modern transducers and associated signal-analysis techniques can now discriminate between truly random variations and significant trends. With equipment sensors that provide operating information, increasingly available technology supports realtime, on-board monitoring of equipment.

JD Edwards EnterpriseOne CBM enables you to:

- Identify changes in the condition of a machine that will indicate a potential failure.
- Identify physical characteristics that collectively indicate the current condition of a machine.
- Measure, analyze, and report data to recognize trends.

Normally, an alarm condition generates a maintenance action, such as the creation of a work order to inspect, repair, or replace the cause of the alarm. Other alarms might send a pager alert, email message, or other electronic alert to the proper technician, supervisor, or individual who needs instant notification of critical equipment status.

JD Edwards EnterpriseOne CBM is also useful for budgeting considerations, such as planning for costly engine maintenance only when certain conditions occur.

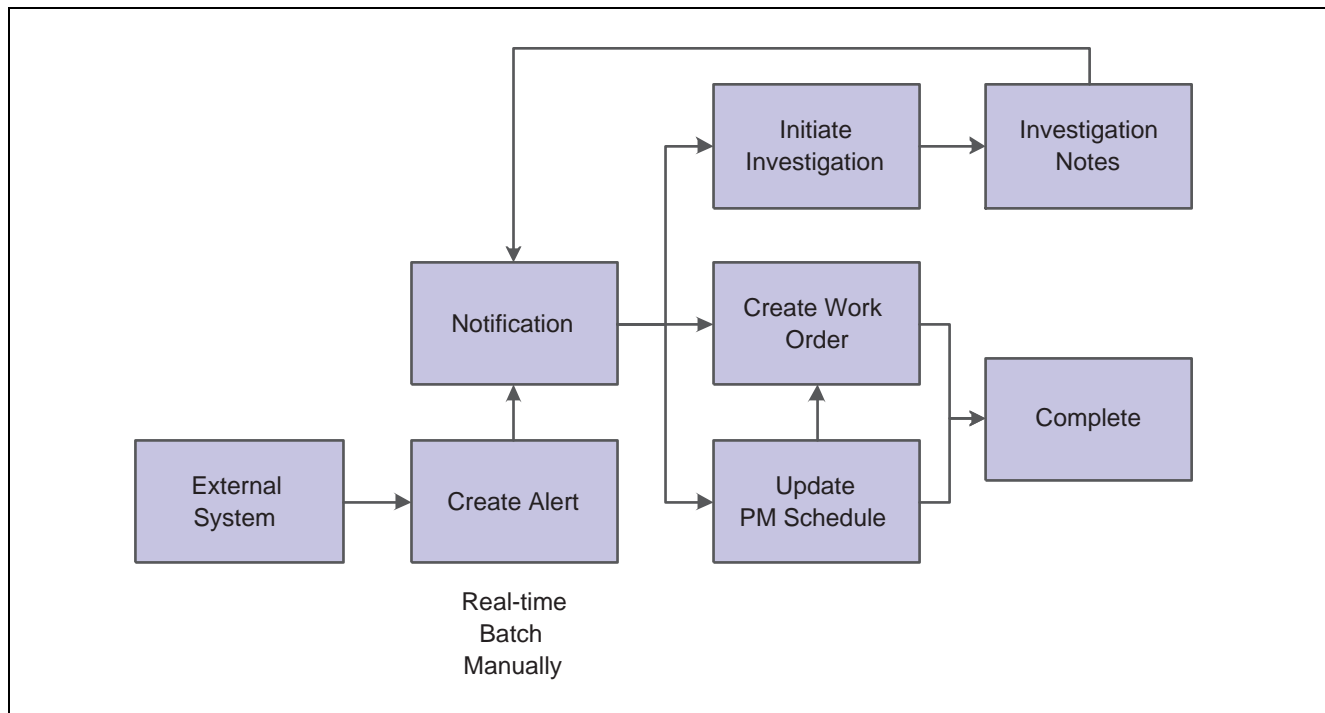
Note. JD Edwards EnterpriseOne CBM supports receiving alerts and responding to them, but does not support receiving readings and analyzing them to determine whether an alert condition exists.

Process Flow for Condition-Based Maintenance

These steps provide an example of the typical flow of events and processes within JD Edwards EnterpriseOne CBM:

1. An external system determines whether an equipment reading is at an alert status.
2. The external system sends alert details to the JD Edwards EnterpriseOne CBM system.
3. The JD Edwards EnterpriseOne CBM system imports the alert details to the condition-based alerts table.
4. The JD Edwards EnterpriseOne CBM system sends a notification message to the maintenance coordinator.
5. The maintenance coordinator reviews the message about the equipment in alert.
6. The maintenance coordinator sends an investigation message to the responsible technician
Alternatively, the equipment alert causes the system to automatically send an investigation message to the responsible technician
7. The technician reports back to the maintenance coordinator with the results of the investigation.
8. The maintenance coordinator uses the investigation results to determine whether to create a work order.
9. If necessary, the maintenance coordinator creates a work order or updates the preventive maintenance schedule to create a work order.
Alternatively, the equipment alert causes the system to automatically create a work order to respond to the alert condition.

This flowchart illustrates the JD Edwards EnterpriseOne CBM process:



Condition-Based Maintenance process

Entering Condition-Based Alerts

This section provides an overview of condition-based alert entry and discusses how to:

- Set processing options for the Condition-Based Alerts Workbench program (P1310).
- Set processing options for the Condition-Based Alerts Revisions program (P1311).
- Enter condition-based alerts manually.

Understanding Condition-Based Alert Entry

Several methods are available to enter condition-based alerts into the system. An external system can send condition-based alerts into the JD Edwards EnterpriseOne system, or you can manually enter alerts from within JD Edwards EnterpriseOne.

You can use these methods to enter alerts:

- Electronically enter alerts in real time, following the XPI model.

The system receives the alert information using the notification event. The system sends information about the successful or unsuccessful alert creation using the response event. You can manage these electronic alerts from the Condition-Based Alerts Workbench program (P1310).

- Electronically download alerts in batch.

The system imports information from the interoperability table to the application tables and performs the same validation and business processes that occur as if you enter alerts manually. You can manage these electronic alerts from the Condition-Based Alerts Workbench program.

- Enter alerts manually using the Condition-Based Alerts Workbench program.

The Condition-Based Alerts Workbench program also enables you to perform various alert tasks, such as:

- Adding important alert information, including media objects that contain time stamps.

You can enter text that describes the issue and actions being taken, and the system enters a time stamp (date, time, and user information) for each entry.

- Sending a notification alert message manually or automatically to those who need instant notification of critical equipment status, such as technicians, supervisors, or a distribution list.
- Responding to the alerts.

Alert responses are discussed separately.

Form Used to Enter Condition-Based Alerts Manually

Form Name	FormID	Navigation	Usage
Condition-Based Alerts Revisions	W1311B	Daily Condition-Based Maintenance Processing (G13CBM10), Condition-Based Alerts Workbench Click Add on Work With Condition-Based Alerts.	Enter condition-based alerts manually.

Setting Processing Options for the Condition-Based Alerts Workbench Program (P1310)

Processing options enable you to specify the default processing for programs and reports.

Defaults

1. Business Unit, 2. Location, 3. Customer Number, 4. Site Number, 5. Manager, 6. Technician, 7. Alert Status, 8. Alert Level, 10. Measurement Location, 13. Equipment Status, 14. Company, 15. Product Family, and 16. Product Model

Specify the business unit or job, location, customer number, site number, manager number, technician number, alert status, alter level, measurement status, equipment status, company, product family, or product model that the system uses to search for condition-based alerts.

9. Automated Response Type

Specify the automated response type that the system uses to search for condition-based alerts. Values are listed in user-defined code (UDC) 13/AR (Automated Response Type).

11. Event Date From and 12. Event Date Thru

Specify the beginning event date or the ending event date in a range of event dates. The system uses this date when searching for condition-based alerts.

17. Notification Workflow Process and 18. Investigation Workflow Process

Specify the condition-based alerts workflow process that the system uses for notification or investigation. The system uses these values when displaying the notification message information or the investigation message information from workflow. If you leave this processing option blank, the system uses the CBMNOTIFY workflow process.

Note. If you leave the Investigation Workflow processing option blank, the system uses the CBMINVEST workflow process.

Categories

1. Major Accounting Class

Specify the category code that classifies assets into accounting classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/C1 (Major Accounting Class).

2. Major Equipment Class

Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/C2 (Major Equipment Class).

3. Manufacturer

Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/C3 (Manufacturer).

4. Model Year

Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/C4 (Model Year).

5. Usage Miles or Hours

Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/C5 (Usage Miles or Hours).

- | | |
|--|--|
| 6. Category Code - F/A 6 | Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/C6 (Equipment Code). |
| 7. Category Code - F/A 7 | Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/C7 (Category Code 7). |
| 8. Category Code - F/A 8 | Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/C8 (Division). |
| 9. Category Code - F/A 9 | Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/C9 (Category Code 9). |
| 10. Category Code - F/A 10 (Rate Group) | Specify the category code that groups similar items for billing. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/C0 (Rate Group). |
| 11. Category Code - F/A 11 | Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/F1 (Category Code 11). |
| 12. Category Code - F/A 12 | Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/F2 (Category Code 12). |
| 13. Category Code - F/A 13 | Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/F3 (Category Code 13). |
| 14. Category Code - F/A 14 | Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/F4 (Category Code 14). |
| 15. Category Code - F/A 15 | Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/F5 (Category Code 15). |
| 16. Category Code - F/A 16 | Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/F6 (Category Code 16). |
| 17. Category Code - F/A 17 | Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/F7 (Category Code 17). |
| 18. Category Code - F/A 18 | Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/F8 (Category Code 18). |
| 19. Category Code - F/A 19 | Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/F9 (Category Code 19). |

- 20. Category Code - F/A 20** Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/F0 (Category Code 20).
- 21. Category Code - F/A 21** Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/21 (Category Code 21).
- 22. Category Code - F/A 22** Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/22 (Category Code 22).
- 23. Category Code - F/A 23** Specify the category code that classifies assets into groups or classes. The system uses this code when searching for condition-based alerts. Enter a value from UDC 12/23 (Category Code 23).

Versions

- 1. Condition-Based Alerts Revisions (P1311) Version** Specify the version of the Condition-Based Alerts Revisions program (P1311). If you leave this processing option blank, the system uses ZJDE0001.
- 2. Work with Failure Analysis (P17766) Version** Specify the version of the Work with Failure Analysis program (P17766). If you leave this processing option blank, the system uses ZJDE0001.
- 3. Work Order Revisions (P17714) Version** Specify the version of the Work Order Revisions program (P17714). If you leave this processing option blank, the system uses ZJDE0003.
- 4. PM Backlog (P12071) Version** Specify the version of the Preventive Maintenance Backlog program (P12071). If you leave this processing option blank, the system uses ZJDE0001.
- 5. Equipment Search/Select (P17012S) Version** Specify the version of the Equipment Search/Select program (P17012S). If you leave this processing option blank, the system uses ZJDE0001.

Setting Processing Options for the Condition-Based Alerts Revisions Program (P1311)

Processing options enable you to specify the default processing for programs and reports.

Defaults

- 1. Send Notification Message** Specify a default value for the Send Notification Message field when adding a condition-based alert record. Values are:
- Blank: Use the alert action rules to determine the value.
- 0: Do not send notification messages.
- 1: Send notification messages.
- 2. Automated Response Type** Enter a value from UDC 13/AR (Automated Response Type) that specifies the default value for the Automated Response Type field when you add a condition-based alert record. If you leave this processing option blank, the system will attempt to retrieve the value from the alert action rules.
- 3. Notification Workflow Process and 4. Investigation Workflow Process** Specify the condition-based alerts workflow process that the system uses for notification or investigation. The system uses these values when displaying the notification message information or the investigation message information

from workflow. If you leave this processing option blank, the system uses the CBMNOTIFY workflow process.

Note. If you leave the Investigation Workflow processing option blank, the system uses the CBMINVEST workflow process.

Process

1. Condition-Based Alerts Processor (R1312) Version

Specify the version of the Condition-Based Alerts Processor program (R1312) that the system uses when processing the manual responses of condition-based alerts. If you leave this processing option blank, the system uses XJDE0001.

Versions

1. Condition-Based Alerts Processor (R1312) Version

Specify the version of the Condition-Based Alerts Processor program (R1312) that the system uses when processing the automated responses of condition-based alerts. If you leave this processing option blank, the system does not process the automated responses of condition-based alerts.

Note. If you leave this processing option blank, you must run a version of the Condition-Based Alerts Processor program (R1312) to be able to complete the automated response-type processing.

2. Work with Failure Analysis (P17766) Version

Specify the version that the system uses for the Work with Failure Analysis program (P17766). If you leave this processing option blank, the system uses ZJDE0001.

3. Work Order Revisions (P17714) Version

Specify the version that the system uses for the Work Order Revisions program (P17714). If you leave this processing option blank, the system uses ZJDE0003.

4. PM Backlog (P12071) Version

Specify the version of the Preventive Maintenance Backlog program (P12071) that the system uses. If you leave this processing option blank, the system uses ZJDE0001.

5. Equipment Search/Select (P17012S) Version

Specify the version of the Equipment Search/Select program (P17012S). If you leave this processing option blank, the system uses ZJDE0001.

Entering Condition-Based Alerts Manually

Access the Condition-Based Alerts Revisions form.

Condition-Based Alerts Workbench - Condition-Based Alerts Revisions

OK Cancel Form Tools

Alert Details **Response Details**

Equipment Number: 24900 Forklift

Measurement Location:

Description: Oil Leak/Engine Malfunction

Alert Level: 2 Alarm

Alert Status: 1 Open

Event Date / Time: 03/12/04 13:05:00

Notification

☒ Send Notification Message

Notification Recipient: 4800 Josephson, Michael

Notification Structure Type:

Text

Courier New 10 B I U

Condition-Based Alerts Revisions form

Equipment Number

Enter an identification code to represent an asset. You enter the identification code in one of these formats:

- 1: Asset number (a computer-assigned, eight-digit, numeric control number).
- 2: Unit number (a 12-character alphanumeric field).
- 3: Serial number (a 25-character alphanumeric field).

Every asset has an asset number. You can use unit number and serial number to further identify assets. If this is a data entry field, the first character that you enter indicates whether you are entering the primary (default) format that is defined for the system, or one of the other two formats. A special character (such as / or *) in the first position of this field indicates which asset number format you are using. You assign special characters to asset number formats on the fixed assets system constants form.

Measurement Location

Enter a value from UDC 13/LC (Measurement Location) that indicates a measurement location on a piece of equipment. You can use this code to indicate where a measurement is taken or where an alert originates on a piece of equipment.

Event Date / Time

Enter the date on which an event occurred, such as the date of a test reading or the date of a condition-based alarm.

Send Notification Message

Enter a code to specify whether a notification message is sent when a condition-based alert is entered into the system. Values are:

1 or Y: Send.

0 or N: Do not send.

This field can be populated from the processing options or from the information setup in the alert action rules.

The system will use the notification message rules to determine who receives the message, unless you enter an address book number in the Notification Recipient field.

Responding to Condition-Based Alerts

This section provides an overview of responses to condition-based alerts and discusses how to:

- Enter responses to condition-based alerts.
- Initiate an investigation message.
- Create a work order.
- Update the preventive maintenance schedule.
- Process condition-based alerts.
- Set processing options for the Condition-Based Alerts Processor program (R1312).
- Review condition-based alerts.

Understanding Responses to Condition-Based Alerts

When the system receives an equipment alert, you can send an investigation message to one person (such as a plant technician or supervisor) or to a distribution list to begin the investigation process. Then, you can generate a work order or update the preventive maintenance schedule, based on the setup parameters for condition-based maintenance. You can respond to alerts manually by using the Condition-Based Alerts Workbench program (P1310), automatically by setting up alert action rules, or by running the Condition-Based Alerts Processor program (R1312).

These responses are available for condition-based alerts:

- Initiate an investigation request.

The system sends a boilerplate message through workflow. The investigation message appears in the employee Work Center or in an email, requires a response for completing the investigation, and notifies the originator for follow-up action.

- Create a work order to inspect, repair, or replace the cause of the alert.

The system uses information from the model work order, maintenance rules, alert action rules, and processing options of the Condition-Based Alerts Processor (R1312) to create the new work order.

- Update the preventive maintenance schedule.

The system updates the alert level from the condition-based alert to a PM schedule that is set up as condition-based. You then use the preventive maintenance system to inspect, repair, or replace the cause of the alert.

To process condition-based alerts according to their automated response type values and the alert action rules, you must run the Condition-Based Alerts Processor program (R1312). For example, if the automated response type on an alert contains a value of 3 (Create W.O. from Model), R1312 creates a work order. You can set up multiple alert action rules to determine how the system responds to an alert.

The R1312 program enables you to:

- Create an investigation message.
- Create a work order from a model.
- Update the Maintenance Schedule File table (F1207).
- Run the Update PM Schedule Status program (R12807).

The report includes the equipment number, description, alert level, automated response type, and results (such as to whom an investigation message was sent or the work order number that was created).

Note. Program R1312 does not affect condition-based alerts that have an automated response type value of *I* (No Automated Response).

Use the Condition-Based Alerts Workbench program (P1310) to review condition-based alert messages. From the workbench, you can access the Process Audit Trail (P98860, Process Task Monitor), which displays the recipients of notification or investigation messages and the detail and status of workflow activities.

See Also

Chapter 3, "Working with Condition-Based Alerts," Processing Condition-Based Alerts, page 29

JD Edwards EnterpriseOne Tools 8.98 Foundation Guide

Forms Used to Respond to Condition-Based Maintenance Alerts

Form Name	FormID	Navigation	Usage
Condition-Based Alerts Revisions	W1311B	Daily Condition-Based Maintenance Processing (G13CBM10), Condition-Based Alerts Workbench Locate the applicable alert on the Work With Condition-Based Alerts form. Select Alert Revisions from the Row menu.	Enter responses to condition-based alerts.
Create Investigation Message	W1311A	Select Manual Responses from the Condition-Based Alert Revisions Form menu, and then select Investigation Msg.	Initiate an investigation message.
Create W.O. from Model	W1311E	Select Manual Responses from the Condition-Based Alert Revisions Form menu, and then select Create W.O.	Create a work order.
Update P.M. Schedule	W1311F	Select Manual Responses from the Condition-Based Alert Revisions Form menu, and then select Update P.M.	Update the preventive maintenance schedule.
Process Audit Trail	W98860A	Locate and select the applicable alert message on the Work With Condition-Based Alerts form. Select Message Review from the Row menu, and then select Notification or Investigation.	Review condition-based alerts.

Entering Responses to Condition-Based Alerts

Access the Condition-Based Alerts Revisions form.

You can determine the action that needs to be taken to resolve the alert. If you use alert action rules, you can review the system generation responses.

Initiating an Investigation Message

Access the Create Investigation Message form.

Create Investigation Message form

Note. When the system creates an investigation message, a notification recipient is required. This is the address book number that will receive the email response from the investigator once completed. If a notification email is not sent, the system will use the notification rules.

Note. If you set up the alert action rules to create an investigation message, this information will be supplied when the alert is created.

Creating a Work Order

Access the Create W.O. from Model form.

Create W.O. from Model form

Note. If you set up the alert action rules to create a work order, this information will be supplied when the alert is created.

Model Work Order

Enter a number to indicate the model work order. This document can be a voucher, a sales order, an invoice, unapplied cash, a journal entry, and so on.

Service Type

Enter a code to indicate the condition-based maintenance service to be performed. Values are in UDC 12/ST (Service Types). Examples of codes include TIRES for replace tires and ENGINE for engine overhaul.

The system uses this code, in conjunction with the equipment number, to obtain the default rules from the Maintenance Schedule File (F1207) table when creating a work order.

Planned Start Date

Enter the date when the item or line of work is to start.

Updating the Preventive Maintenance Schedule

Access the Update P.M. Schedule form.

Update P.M. Schedule form

Note. If you set up the alert action rules to update the PM schedule, this information will be supplied when the alert is created.

Service Type

Enter a code to indicate the condition-based maintenance service to be performed.

You can then use the PM Update Status program (R12807) to create a work order for the service type.

Processing Condition-Based Alerts

Select Daily Condition-Based Maintenance Processing (G13CBM10), Condition-Based Alerts Processor.

Setting Processing Options for the Condition-Based Alerts Processor Program (R1312)

Processing options enable you to specify the default processing for programs and reports.

Process

1. Work Center Or Report Specify where the system writes errors. Values are:

Blank: Write errors to the Work Center.

- I*: Write errors on the report.
- 2. Job Status Message Recipient** Specify the address book number of the recipient of job status messages that result from the Condition-Based Alerts Processor program (R1312). If you leave this processing option blank, the system uses the address book number of the current user.
- 3. Update PM Schedule (R12807) Version** Specify which version of the Update PM Schedule Status program (R12807) the system uses to process PM schedules that have been updated as a result of a condition-based alert. If you leave this processing option blank, the system does not process PM schedules.
- 4. Work Order Cross Reference** Specify the value for the work order cross-reference. Values are:
Blank: Immediate parent of the equipment.
I: Equipment's top-level parent.
2: Value from the model work order.
- 5. Create Individual Resource Assignments** Specify whether the system automatically creates individual resource assignments that are based on the Assigned To (ANP) address book number on the new work order. The system verifies that the address book number appears in the Resource Master table (F48310). This processing option applies only when the condition-based alerts process creates a work order. Values are:
Blank: Do not create.
I: Create.
- 6. Work Order Parts Detail (P17730) Version** Specify the version that the system uses for the Work Order Parts Detail program (P17730). If you leave this processing option blank, the system uses ZJDE0002. This processing applies when parts detail is attached to the new work order.
- 7. Work Order Labor Detail (P17732) Version** Specify the version that the system uses for the Work Order Labor Detail program (P17732). If you leave this processing option blank, the system uses ZJDE0002. This processing option applies when labor detail is attached to the new work order.

Versions

- 1. Condition-Based Alerts Revisions (P1311) Version** Specify the version that the system uses for the link to the Condition-Based Alerts Revisions program (P1311) that is provided in the investigation message. If you leave this processing option blank, the system uses ZJDE0001.

Reviewing Condition-Based Alerts

Access the Process Audit Trail form.

Condition-Based Alerts Workbench - Process Audit Trail

Select Close Row Tools

Process: CBMNOTIFY CBM Notification Process

Originator: 6085156 Susan Powers

Version: 1

Process Key: [3]

Records 1 - 4 [Customize Grid](#)

	Task Description	Resource	Status	Start Date	Start Time	End Date
<input checked="" type="radio"/>	START		Completed	03/12/04	13:52:12	03/12
<input type="radio"/>	Send to Single Recipient		Completed	03/12/04	13:52:13	03/12
<input type="radio"/>	Sent Notification - Single Recipient	Josephson, Michael	Completed	03/12/04	13:52:13	03/12
<input type="radio"/>	END		Completed	03/12/04	13:52:14	03/12

Process Audit Trail form

Completing Condition-Based Alerts

This section provides an overview of condition-based alerts completion and lists the forms that are used to complete condition-based alerts.

Understanding Condition-Based Alerts Completion

After you have responded to a condition-based alert, you can change the alert status to a *complete* or *closed* status when the work is completed. You can complete the alerts using:

- Work order activity rules.

If you are using work orders, you can use the work order activity rules to complete or close the condition-based alert according to the work order status.

- PM Backlog.

To complete alerts through the PM backlog, you must set the processing options for completion or cancelation and update the PM status.

- Condition-Based Alerts Workbench.

You can complete alerts manually by changing the Alert Status to a complete or closed status.

See Also

JD Edwards EnterpriseOne Capital Asset Management 9.0 Implementation Guide, "Setting Up Work Orders"

JD Edwards EnterpriseOne Capital Asset Management 9.0 Implementation Guide, "Working with Preventive Maintenance Schedules," Changing the Status of PMs to Complete

Forms Used to Complete Condition-Based Alerts

Form Name	FormID	Navigation	Usage
Work Order Revisions	W17714A	<p>Work Order (G1316), Work Order Entry</p> <p>Locate and select a work order, and then click Select on the Work With Work Orders form.</p> <p>Change the work order status on the Planning tab of the Work Order Revisions form.</p>	Complete a work order, which then completes the condition-based alert through work order activity rules.
PM Backlog	W12071A	Daily Condition-Based Maintenance Processing (G13CBM10), PM Backlog	Complete a condition-based alert and change status of PM to <i>Complete</i> .
Work With Condition-Based Alerts	W1310A	<p>Daily Condition-Based Maintenance Processing (G13CBM10), Condition-Based Alerts Workbench</p> <p>Enter a closed status in the Alert Status field on the Alerts tab.</p>	Complete a condition-based alert.

APPENDIX A

Delivered Workflow for JD Edwards EnterpriseOne Condition-Based Maintenance

This appendix discusses the delivered workflow for JD Edwards EnterpriseOne Condition-Based Maintenance (CBM).

See Also

JD Edwards EnterpriseOne Tools 8.98 Workflow Tools Guide

Delivered Workflow for Condition-Based Maintenance

This section discusses the JD Edwards EnterpriseOne CBM workflow processes.

Condition-Based Maintenance Notification Process

This section discusses the JD Edwards EnterpriseOne CBM notification process workflow.

Description

Workflow Description	The system sends a workflow notification message to a responsible person or persons within the maintenance team when the system receives a condition-based alert.
Workflow Trigger	The system receives a CBM Alert that is flagged to create a notification message.
Workflow Action	The responsible person or persons within the maintenance team receive a notification message indicating that a condition-based alert has been received by the system.

Workflow Objects

System	13
Workflow Object Name	CBMNOTIFY
Object ID	N1301040
Event Descriptions / Function Name	CallCBMNotificationWorkflow
Sequence / Line Numbers	11, 11

Condition-Based Maintenance Investigation Process

This section discusses the JD Edwards EnterpriseOne CBM investigation process workflow.

Description

Workflow Description	The system can send a workflow investigation request message to a responsible person or persons within the maintenance team to investigate the condition-based alert that the system receives. When the investigation is complete, the system sends an investigation completed message to the original notification recipient.
Workflow Trigger	The system receives a CBM Alert that is flagged with an automated response type of 2 to create an investigation request message. You can also manually trigger an investigation request message from the Condition-Based Alerts Workbench (P1310).
Workflow Action	The responsible person or persons within the maintenance team receive an investigation message requesting they investigate the condition-based alert that was received in the system. The message provides a link to the Condition-Based Alerts Investigation Revisions form. On this form, the recipient can record investigation notes and indicate whether or not the investigation is complete. When the investigation is complete, the system sends another message to the original notification recipient.

Workflow Objects

System	13
Workflow Object Name	CBMINVEST
Object ID	N1301070
Event Descriptions / Function Name	CallCBMInvestigationWorkflow
Sequence / Line Number	11

Glossary of JD Edwards EnterpriseOne Terms

Accessor Methods/Assessors	Java methods to “get” and “set” the elements of a value object or other source file.
activity rule	The criteria by which an object progresses from one given point to the next in a flow.
add mode	A condition of a form that enables users to input data.
Advanced Planning Agent (APAg)	A JD Edwards EnterpriseOne tool that can be used to extract, transform, and load enterprise data. APAg supports access to data sources in the form of rational databases, flat file format, and other data or message encoding, such as XML.
alternate currency	<p>A currency that is different from the domestic currency (when dealing with a domestic-only transaction) or the domestic and foreign currency of a transaction.</p> <p>In JD Edwards EnterpriseOne Financial Management, alternate currency processing enables you to enter receipts and payments in a currency other than the one in which they were issued.</p>
Application Server	Software that provides the business logic for an application program in a distributed environment. The servers can be Oracle Application Server (OAS) or WebSphere Application Server (WAS).
as if processing	A process that enables you to view currency amounts as if they were entered in a currency different from the domestic and foreign currency of the transaction.
as of processing	A process that is run as of a specific point in time to summarize transactions up to that date. For example, you can run various JD Edwards EnterpriseOne reports as of a specific date to determine balances and amounts of accounts, units, and so on as of that date.
Auto Commit Transaction	A database connection through which all database operations are immediately written to the database.
back-to-back process	A process in JD Edwards EnterpriseOne Supply Management that contains the same keys that are used in another process.
batch processing	<p>A process of transferring records from a third-party system to JD Edwards EnterpriseOne.</p> <p>In JD Edwards EnterpriseOne Financial Management, batch processing enables you to transfer invoices and vouchers that are entered in a system other than JD Edwards EnterpriseOne to JD Edwards EnterpriseOne Accounts Receivable and JD Edwards EnterpriseOne Accounts Payable, respectively. In addition, you can transfer address book information, including customer and supplier records, to JD Edwards EnterpriseOne.</p>
batch server	A server that is designated for running batch processing requests. A batch server typically does not contain a database nor does it run interactive applications.
batch-of-one immediate	<p>A transaction method that enables a client application to perform work on a client workstation, then submit the work all at once to a server application for further processing. As a batch process is running on the server, the client application can continue performing other tasks.</p> <p>See also direct connect and store-and-forward.</p>
best practices	Non-mandatory guidelines that help the developer make better design decisions.

BPEL	Abbreviation for <i>Business Process Execution Language</i> , a standard web services orchestration language, which enables you to assemble discrete services into an end-to-end process flow.
BPEL PM	Abbreviation for <i>Business Process Execution Language Process Manager</i> , a comprehensive infrastructure for creating, deploying, and managing BPEL business processes.
Build Configuration File	Configurable settings in a text file that are used by a build program to generate ANT scripts. ANT is a software tool used for automating build processes. These scripts build published business services.
build engineer	An actor that is responsible for building, mastering, and packaging artifacts. Some build engineers are responsible for building application artifacts, and some are responsible for building foundation artifacts.
Build Program	A WIN32 executable that reads build configuration files and generates an ANT script for building published business services.
business analyst	An actor that determines if and why an EnterpriseOne business service needs to be developed.
business function	A named set of user-created, reusable business rules and logs that can be called through event rules. Business functions can run a transaction or a subset of a transaction (check inventory, issue work orders, and so on). Business functions also contain the application programming interfaces (APIs) that enable them to be called from a form, a database trigger, or a non-JD Edwards EnterpriseOne application. Business functions can be combined with other business functions, forms, event rules, and other components to make up an application. Business functions can be created through event rules or third-generation languages, such as C. Examples of business functions include Credit Check and Item Availability.
business function event rule	See named event rule (NER).
business service	EnterpriseOne business logic written in Java. A business service is a collection of one or more artifacts. Unless specified otherwise, a business service implies both a published business service and business service.
business service artifacts	Source files, descriptors, and so on that are managed for business service development and are needed for the business service build process.
business service class method	A method that accesses resources provided by the business service framework.
business service configuration files	Configuration files include, but are not limited to, interop.ini, JDBj.ini, and jdelog.properties.
business service cross reference	A key and value data pair used during orchestration. Collectively refers to both the code and the key cross reference in the WSG/XPI based system.
business service cross-reference utilities	Utility services installed in a BPEL/ESB environment that are used to access JD Edwards EnterpriseOne orchestration cross-reference data.
business service development environment	A framework needed by an integration developer to develop and manage business services.
business services development tool	Otherwise known as JDeveloper.
business service EnterpriseOne object	A collection of artifacts managed by EnterpriseOne LCM tools. Named and represented within EnterpriseOne LCM similarly to other EnterpriseOne objects like tables, views, forms, and so on.

business service framework	Parts of the business service foundation that are specifically for supporting business service development.
business service payload	An object that is passed between an enterprise server and a business services server. The business service payload contains the input to the business service when passed to the business services server. The business service payload contains the results from the business service when passed to the Enterprise Server. In the case of notifications, the return business service payload contains the acknowledgement.
business service property	Key value data pairs used to control the behavior or functionality of business services.
Business Service Property Admin Tool	An EnterpriseOne application for developers and administrators to manage business service property records.
business service property business service group	A classification for business service property at the business service level. This is generally a business service name. A business service level contains one or more business service property groups. Each business service property group may contain zero or more business service property records.
business service property categorization	A way to categorize business service properties. These properties are categorized by business service.
business service property key	A unique name that identifies the business service property globally in the system.
business service property utilities	A utility API used in business service development to access EnterpriseOne business service property data.
business service property value	A value for a business service property.
business service repository	A source management system, for example ClearCase, where business service artifacts and build files are stored. Or, a physical directory in network.
business services server	The physical machine where the business services are located. Business services are run on an application server instance.
business services source file or business service class	One type of business service artifact. A text file with the .java file type written to be compiled by a Java compiler.
business service value object template	The structural representation of a business service value object used in a C-business function.
Business Service Value Object Template Utility	A utility used to create a business service value object template from a business service value object.
business services server artifact	The object to be deployed to the business services server.
business view	A means for selecting specific columns from one or more JD Edwards EnterpriseOne application tables whose data is used in an application or report. A business view does not select specific rows, nor does it contain any actual data. It is strictly a view through which you can manipulate data.
central objects merge	A process that blends a customer's modifications to the objects in a current release with objects in a new release.
central server	A server that has been designated to contain the originally installed version of the software (central objects) for deployment to client computers. In a typical JD Edwards EnterpriseOne installation, the software is loaded on to one machine—the central server. Then, copies of the software are pushed out or downloaded to various workstations attached to it. That way, if the software is altered or corrupted through its use on workstations, an original set of objects (central objects) is always available on the central server.

charts	Tables of information in JD Edwards EnterpriseOne that appear on forms in the software.
check-in repository	A repository for developers to check in and check out business service artifacts. There are multiple check-in repositories. Each can be used for a different purpose (for example, development, production, testing, and so on).
connector	Component-based interoperability model that enables third-party applications and JD Edwards EnterpriseOne to share logic and data. The JD Edwards EnterpriseOne connector architecture includes Java and COM connectors.
contra/clearing account	A general ledger account in JD Edwards EnterpriseOne Financial Management that is used by the system to offset (balance) journal entries. For example, you can use a contra/clearing account to balance the entries created by allocations in JD Edwards EnterpriseOne Financial Management.
Control Table Workbench	An application that, during the Installation Workbench processing, runs the batch applications for the planned merges that update the data dictionary, user-defined codes, menus, and user override tables.
control tables merge	A process that blends a customer's modifications to the control tables with the data that accompanies a new release.
correlation data	The data used to tie HTTP responses with requests that consist of business service name and method.
cost assignment	The process in JD Edwards EnterpriseOne Advanced Cost Accounting of tracing or allocating resources to activities or cost objects.
cost component	In JD Edwards EnterpriseOne Manufacturing, an element of an item's cost (for example, material, labor, or overhead).
credentials	A valid set of JD Edwards EnterpriseOne username/password/environment/role, EnterpriseOne session, or EnterpriseOne token.
cross-reference utility services	Utility services installed in a BPEL/ESB environment that access EnterpriseOne cross-reference data.
cross segment edit	A logic statement that establishes the relationship between configured item segments. Cross segment edits are used to prevent ordering of configurations that cannot be produced.
currency restatement	The process of converting amounts from one currency into another currency, generally for reporting purposes. You can use the currency restatement process, for example, when many currencies must be restated into a single currency for consolidated reporting.
cXML	A protocol used to facilitate communication between business documents and procurement applications, and between e-commerce hubs and suppliers.
database credentials	A valid database username/password.
database server	A server in a local area network that maintains a database and performs searches for client computers.
Data Source Workbench	An application that, during the Installation Workbench process, copies all data sources that are defined in the installation plan from the Data Source Master and Table and Data Source Sizing tables in the Planner data source to the system-release number data source. It also updates the Data Source Plan detail record to reflect completion.
date pattern	A calendar that represents the beginning date for the fiscal year and the ending date for each period in that year in standard and 52-period accounting.

denominated-in currency	The company currency in which financial reports are based.
deployment artifacts	Artifacts that are needed for the deployment process, such as servers, ports, and such.
deployment server	A server that is used to install, maintain, and distribute software to one or more enterprise servers and client workstations.
detail information	Information that relates to individual lines in JD Edwards EnterpriseOne transactions (for example, voucher pay items and sales order detail lines).
direct connect	A transaction method in which a client application communicates interactively and directly with a server application. See also batch-of-one immediate and store-and-forward.
Do Not Translate (DNT)	A type of data source that must exist on the iSeries because of BLOB restrictions.
dual pricing	The process of providing prices for goods and services in two currencies.
duplicate published business services authorization records	Two published business services authorization records with the same user identification information and published business services identification information.
embedded application server instance	An OC4J instance started by and running wholly within JDeveloper.
edit code	A code that indicates how a specific value for a report or a form should appear or be formatted. The default edit codes that pertain to reporting require particular attention because they account for a substantial amount of information.
edit mode	A condition of a form that enables users to change data.
edit rule	A method used for formatting and validating user entries against a predefined rule or set of rules.
Electronic Data Interchange (EDI)	An interoperability model that enables paperless computer-to-computer exchange of business transactions between JD Edwards EnterpriseOne and third-party systems. Companies that use EDI must have translator software to convert data from the EDI standard format to the formats of their computer systems.
embedded event rule	An event rule that is specific to a particular table or application. Examples include form-to-form calls, hiding a field based on a processing option value, and calling a business function. Contrast with the business function event rule.
Employee Work Center	A central location for sending and receiving all JD Edwards EnterpriseOne messages (system and user generated), regardless of the originating application or user. Each user has a mailbox that contains workflow and other messages, including Active Messages.
enterprise server	A server that contains the database and the logic for JD Edwards EnterpriseOne.
Enterprise Service Bus (ESB)	Middleware infrastructure products or technologies based on web services standards that enable a service-oriented architecture using an event-driven and XML-based messaging framework (the bus).
EnterpriseOne administrator	An actor responsible for the EnterpriseOne administration system.
EnterpriseOne credentials	A user ID, password, environment, and role used to validate a user of EnterpriseOne.
EnterpriseOne object	A reusable piece of code that is used to build applications. Object types include tables, forms, business functions, data dictionary items, batch processes, business views, event rules, versions, data structures, and media objects.

EnterpriseOne development client	Historically called “fat client,” a collection of installed EnterpriseOne components required to develop EnterpriseOne artifacts, including the Microsoft Windows client and design tools.
EnterpriseOne extension	A JDeveloper component (plug-in) specific to EnterpriseOne. A JDeveloper wizard is a specific example of an extension.
EnterpriseOne process	A software process that enables JD Edwards EnterpriseOne clients and servers to handle processing requests and run transactions. A client runs one process, and servers can have multiple instances of a process. JD Edwards EnterpriseOne processes can also be dedicated to specific tasks (for example, workflow messages and data replication) to ensure that critical processes don’t have to wait if the server is particularly busy.
EnterpriseOne resource	Any EnterpriseOne table, metadata, business function, dictionary information, or other information restricted to authorized users.
Environment Workbench	An application that, during the Installation Workbench process, copies the environment information and Object Configuration Manager tables for each environment from the Planner data source to the system-release number data source. It also updates the Environment Plan detail record to reflect completion.
escalation monitor	A batch process that monitors pending requests or activities and restarts or forwards them to the next step or user after they have been inactive for a specified amount of time.
event rule	A logic statement that instructs the system to perform one or more operations based on an activity that can occur in a specific application, such as entering a form or exiting a field.
explicit transaction	Transaction used by a business service developer to explicitly control the type (auto or manual) and the scope of transaction boundaries within a business service.
exposed method or value object	Published business service source files or parts of published business service source files that are part of the published interface. These are part of the contract with the customer.
facility	An entity within a business for which you want to track costs. For example, a facility might be a warehouse location, job, project, work center, or branch/plant. A facility is sometimes referred to as a “business unit.”
fast path	A command prompt that enables the user to move quickly among menus and applications by using specific commands.
file server	A server that stores files to be accessed by other computers on the network. Unlike a disk server, which appears to the user as a remote disk drive, a file server is a sophisticated device that not only stores files, but also manages them and maintains order as network users request files and make changes to these files.
final mode	The report processing mode of a processing mode of a program that updates or creates data records.
foundation	A framework that must be accessible for execution of business services at runtime. This includes, but is not limited to, the Java Connector and JDBj.
FTP server	A server that responds to requests for files via file transfer protocol.
header information	Information at the beginning of a table or form. Header information is used to identify or provide control information for the group of records that follows.
HTTP Adapter	A generic set of services that are used to do the basic HTTP operations, such as GET, POST, PUT, DELETE, TRACE, HEAD, and OPTIONS with the provided URL.

instantiate	A Java term meaning “to create.” When a class is instantiated, a new instance is created.
integration developer	The user of the system who develops, runs, and debugs the EnterpriseOne business services. The integration developer uses the EnterpriseOne business services to develop these components.
integration point (IP)	The business logic in previous implementations of EnterpriseOne that exposes a document level interface. This type of logic used to be called XBPs. In EnterpriseOne 8.11, IPs are implemented in Web Services Gateway powered by webMethods.
integration server	A server that facilitates interaction between diverse operating systems and applications across internal and external networked computer systems.
integrity test	A process used to supplement a company’s internal balancing procedures by locating and reporting balancing problems and data inconsistencies.
interface table	See Z table.
internal method or value object	Business service source files or parts of business service source files that are not part of the published interface. These could be private or protected methods. These could be value objects not used in published methods.
interoperability model	A method for third-party systems to connect to or access JD Edwards EnterpriseOne.
in-your-face-error	In JD Edwards EnterpriseOne, a form-level property which, when enabled, causes the text of application errors to appear on the form.
IServer service	This internet server service resides on the web server and is used to speed up delivery of the Java class files from the database to the client.
jargon	An alternative data dictionary item description that JD Edwards EnterpriseOne appears based on the product code of the current object.
Java application server	A component-based server that resides in the middle-tier of a server-centric architecture. This server provides middleware services for security and state maintenance, along with data access and persistence.
JDBNET	A database driver that enables heterogeneous servers to access each other’s data.
JDEBASE Database Middleware	A JD Edwards EnterpriseOne proprietary database middleware package that provides platform-independent APIs, along with client-to-server access.
JDECallObject	An API used by business functions to invoke other business functions.
jde.ini	A JD Edwards EnterpriseOne file (or member for iSeries) that provides the runtime settings required for JD Edwards EnterpriseOne initialization. Specific versions of the file or member must reside on every machine running JD Edwards EnterpriseOne. This includes workstations and servers.
JDEIPC	Communications programming tools used by server code to regulate access to the same data in multiprocess environments, communicate and coordinate between processes, and create new processes.
jde.log	The main diagnostic log file of JD Edwards EnterpriseOne. This file is always located in the root directory on the primary drive and contains status and error messages from the startup and operation of JD Edwards EnterpriseOne.
JDENET	A JD Edwards EnterpriseOne proprietary communications middleware package. This package is a peer-to-peer, message-based, socket-based, multiprocess communications middleware solution. It handles client-to-server and server-to-server communications for all JD Edwards EnterpriseOne supported platforms.
JDeveloper Project	An artifact that JDeveloper uses to categorize and compile source files.

JDeveloper Workspace	An artifact that JDeveloper uses to organize project files. It contains one or more project files.
JMS Queue	A Java Messaging service queue used for point-to-point messaging.
listener service	A listener that listens for XML messages over HTTP.
local repository	A developer's local development environment that is used to store business service artifacts.
local standalone BPEL/ESB server	A standalone BPEL/ESB server that is not installed within an application server.
Location Workbench	An application that, during the Installation Workbench process, copies all locations that are defined in the installation plan from the Location Master table in the Planner data source to the system data source.
logic server	A server in a distributed network that provides the business logic for an application program. In a typical configuration, pristine objects are replicated on to the logic server from the central server. The logic server, in conjunction with workstations, actually performs the processing required when JD Edwards EnterpriseOne software runs.
MailMerge Workbench	An application that merges Microsoft Word 6.0 (or higher) word-processing documents with JD Edwards EnterpriseOne records to automatically print business documents. You can use MailMerge Workbench to print documents, such as form letters about verification of employment.
Manual Commit transaction	A database connection where all database operations delay writing to the database until a call to commit is made.
master business function (MBF)	An interactive master file that serves as a central location for adding, changing, and updating information in a database. Master business functions pass information between data entry forms and the appropriate tables. These master functions provide a common set of functions that contain all of the necessary default and editing rules for related programs. MBFs contain logic that ensures the integrity of adding, updating, and deleting information from databases.
master table	See published table.
matching document	A document associated with an original document to complete or change a transaction. For example, in JD Edwards EnterpriseOne Financial Management, a receipt is the matching document of an invoice, and a payment is the matching document of a voucher.
media storage object	Files that use one of the following naming conventions that are not organized into table format: Gxxx, xxxGT, or GTxxx.
message center	A central location for sending and receiving all JD Edwards EnterpriseOne messages (system and user generated), regardless of the originating application or user.
messaging adapter	An interoperability model that enables third-party systems to connect to JD Edwards EnterpriseOne to exchange information through the use of messaging queues.
messaging server	A server that handles messages that are sent for use by other programs using a messaging API. Messaging servers typically employ a middleware program to perform their functions.
Middle-Tier BPEL/ESB Server	A BPEL/ESB server that is installed within an application server.
Monitoring Application	An EnterpriseOne tool provided for an administrator to get statistical information for various EnterpriseOne servers, reset statistics, and set notifications.

named event rule (NER)	Encapsulated, reusable business logic created using event rules, rather than C programming. NERs are also called business function event rules. NERs can be reused in multiple places by multiple programs. This modularity lends itself to streamlining, reusability of code, and less work.
<i>nota fiscal</i>	In Brazil, a legal document that must accompany all commercial transactions for tax purposes and that must contain information required by tax regulations.
<i>nota fiscal factura</i>	In Brazil, a <i>nota fiscal</i> with invoice information. See also <i>nota fiscal</i> .
Object Configuration Manager (OCM)	In JD Edwards EnterpriseOne, the object request broker and control center for the runtime environment. OCM keeps track of the runtime locations for business functions, data, and batch applications. When one of these objects is called, OCM directs access to it using defaults and overrides for a given environment and user.
Object Librarian	A repository of all versions, applications, and business functions reusable in building applications. Object Librarian provides check-out and check-in capabilities for developers, and it controls the creation, modification, and use of JD Edwards EnterpriseOne objects. Object Librarian supports multiple environments (such as production and development) and enables objects to be easily moved from one environment to another.
Object Librarian merge	A process that blends any modifications to the Object Librarian in a previous release into the Object Librarian in a new release.
Open Data Access (ODA)	An interoperability model that enables you to use SQL statements to extract JD Edwards EnterpriseOne data for summarization and report generation.
Output Stream Access (OSA)	An interoperability model that enables you to set up an interface for JD Edwards EnterpriseOne to pass data to another software package, such as Microsoft Excel, for processing.
package	JD Edwards EnterpriseOne objects are installed to workstations in packages from the deployment server. A package can be compared to a bill of material or kit that indicates the necessary objects for that workstation and where on the deployment server the installation program can find them. It is point-in-time snapshot of the central objects on the deployment server.
package build	A software application that facilitates the deployment of software changes and new applications to existing users. Additionally, in JD Edwards EnterpriseOne, a package build can be a compiled version of the software. When you upgrade your version of the ERP software, for example, you are said to take a package build. Consider the following context: “Also, do not transfer business functions into the production path code until you are ready to deploy, because a global build of business functions done during a package build will automatically include the new functions.” The process of creating a package build is often referred to, as it is in this example, simply as “a package build.”
package location	The directory structure location for the package and its set of replicated objects. This is usually \\deployment server\release\path_code\package\package name. The subdirectories under this path are where the replicated objects for the package are placed. This is also referred to as where the package is built or stored.
Package Workbench	An application that, during the Installation Workbench process, transfers the package information tables from the Planner data source to the system-release number data source. It also updates the Package Plan detail record to reflect completion.
Pathcode Directory	The specific portion of the file system on the EnterpriseOne development client where EnterpriseOne development artifacts are stored.

patterns	General repeatable solutions to a commonly occurring problem in software design. For business service development, the focus is on the object relationships and interactions. For orchestrations, the focus is on the integration patterns (for example, synchronous and asynchronous request/response, publish, notify, and receive/reply).
planning family	A means of grouping end items whose similarity of design and manufacture facilitates being planned in aggregate.
preference profile	The ability to define default values for specified fields for a user-defined hierarchy of items, item groups, customers, and customer groups.
print server	The interface between a printer and a network that enables network clients to connect to the printer and send their print jobs to it. A print server can be a computer, separate hardware device, or even hardware that resides inside of the printer itself.
pristine environment	A JD Edwards EnterpriseOne environment used to test unaltered objects with JD Edwards EnterpriseOne demonstration data or for training classes. You must have this environment so that you can compare pristine objects that you modify.
processing option	A data structure that enables users to supply parameters that regulate the running of a batch program or report. For example, you can use processing options to specify default values for certain fields, to determine how information appears or is printed, to specify date ranges, to supply runtime values that regulate program execution, and so on.
production environment	A JD Edwards EnterpriseOne environment in which users operate EnterpriseOne software.
production-grade file server	A file server that has been quality assurance tested and commercialized and that is usually provided in conjunction with user support services.
Production Published Business Services Web Service	Published business services web service deployed to a production application server.
program temporary fix (PTF)	A representation of changes to JD Edwards EnterpriseOne software that your organization receives on magnetic tapes or disks.
project	In JD Edwards EnterpriseOne, a virtual container for objects being developed in Object Management Workbench.
promotion path	<p>The designated path for advancing objects or projects in a workflow. The following is the normal promotion cycle (path):</p> <p>11>21>26>28>38>01</p> <p>In this path, <i>11</i> equals new project pending review, <i>21</i> equals programming, <i>26</i> equals QA test/review, <i>28</i> equals QA test/review complete, <i>38</i> equals in production, <i>01</i> equals complete. During the normal project promotion cycle, developers check objects out of and into the development path code and then promote them to the prototype path code. The objects are then moved to the productions path code before declaring them complete.</p>
proxy server	A server that acts as a barrier between a workstation and the internet so that the enterprise can ensure security, administrative control, and caching service.
published business service	EnterpriseOne service level logic and interface. A classification of a published business service indicating the intention to be exposed to external (non-EnterpriseOne) systems.
published business service identification information	Information about a published business service used to determine relevant authorization records. Published business services + method name, published business services, or *ALL.

published business service web service	Published business services components packaged as J2EE Web Service (namely, a J2EE EAR file that contains business service classes, business service foundation, configuration files, and web service artifacts).
published table	Also called a master table, this is the central copy to be replicated to other machines. Residing on the publisher machine, the F98DRPUB table identifies all of the published tables and their associated publishers in the enterprise.
publisher	The server that is responsible for the published table. The F98DRPUB table identifies all of the published tables and their associated publishers in the enterprise.
pull replication	One of the JD Edwards EnterpriseOne methods for replicating data to individual workstations. Such machines are set up as pull subscribers using JD Edwards EnterpriseOne data replication tools. The only time that pull subscribers are notified of changes, updates, and deletions is when they request such information. The request is in the form of a message that is sent, usually at startup, from the pull subscriber to the server machine that stores the F98DRPCN table.
QBE	An abbreviation for <i>query by example</i> . In JD Edwards EnterpriseOne, the QBE line is the top line on a detail area that is used for filtering data.
real-time event	A message triggered from EnterpriseOne application logic that is intended for external systems to consume.
refresh	A function used to modify JD Edwards EnterpriseOne software, or subset of it, such as a table or business data, so that it functions at a new release or cumulative update level, such as B73.2 or B73.2.1.
replication server	A server that is responsible for replicating central objects to client machines.
Rt-Addressing	Unique data identifying a browser session that initiates the business services call request host/port user session.
rules	Mandatory guidelines that are not enforced by tooling, but must be followed in order to accomplish the desired results and to meet specified standards.
quote order	In JD Edwards Procurement and Subcontract Management, a request from a supplier for item and price information from which you can create a purchase order. In JD Edwards Sales Order Management, item and price information for a customer who has not yet committed to a sales order.
secure by default	A security model that assumes that a user does not have permission to execute an object unless there is a specific record indicating such permissions.
Secure Socket Layer (SSL)	A security protocol that provides communication privacy. SSL enables client and server applications to communicate in a way that is designed to prevent eavesdropping, tampering, and message forgery.
SEI implementation	A Java class that implements the methods that declare in a Service Endpoint Interface (SEI).
selection	Found on JD Edwards EnterpriseOne menus, a selection represents functions that you can access from a menu. To make a selection, type the associated number in the Selection field and press Enter.
serialize	The process of converting an object or data into a format for storage or transmission across a network connection link with the ability to reconstruct the original data or objects when needed.
Server Workbench	An application that, during the Installation Workbench process, copies the server configuration files from the Planner data source to the system-release number

	data source. The application also updates the Server Plan detail record to reflect completion.
Service Endpoint Interface (SEI)	A Java interface that declares the methods that a client can invoke on the service.
SOA	Abbreviation for <i>Service Oriented Architecture</i> .
softcoding	A coding technique that enables an administrator to manipulate site-specific variables that affect the execution of a given process.
source repository	A repository for HTTP adapter and listener service development environment artifacts.
spot rate	An exchange rate entered at the transaction level. This rate overrides the exchange rate that is set up between two currencies.
Specification merge	A merge that comprises three merges: Object Librarian merge, Versions List merge, and Central Objects merge. The merges blend customer modifications with data that accompanies a new release.
specification	A complete description of a JD Edwards EnterpriseOne object. Each object has its own specification, or name, which is used to build applications.
Specification Table Merge Workbench	An application that, during the Installation Workbench process, runs the batch applications that update the specification tables.
SSL Certificate	A special message signed by a certificate authority that contains the name of a user and that user's public key in such a way that anyone can "verify" that the message was signed by no one other than the certification authority and thereby develop trust in the user's public key.
store-and-forward	The mode of processing that enables users who are disconnected from a server to enter transactions and then later connect to the server to upload those transactions.
subscriber table	Table F98DRSUB, which is stored on the publisher server with the F98DRPUB table and identifies all of the subscriber machines for each published table.
superclass	An inheritance concept of the Java language where a class is an instance of something, but is also more specific. "Tree" might be the superclass of "Oak" and "Elm," for example.
supplemental data	<p>Any type of information that is not maintained in a master file. Supplemental data is usually additional information about employees, applicants, requisitions, and jobs (such as an employee's job skills, degrees, or foreign languages spoken). You can track virtually any type of information that your organization needs.</p> <p>For example, in addition to the data in the standard master tables (the Address Book Master, Customer Master, and Supplier Master tables), you can maintain other kinds of data in separate, generic databases. These generic databases enable a standard approach to entering and maintaining supplemental data across JD Edwards EnterpriseOne systems.</p>
table access management (TAM)	The JD Edwards EnterpriseOne component that handles the storage and retrieval of use-defined data. TAM stores information, such as data dictionary definitions; application and report specifications; event rules; table definitions; business function input parameters and library information; and data structure definitions for running applications, reports, and business functions.
Table Conversion Workbench	An interoperability model that enables the exchange of information between JD Edwards EnterpriseOne and third-party systems using non-JD Edwards EnterpriseOne tables.

table conversion	An interoperability model that enables the exchange of information between JD Edwards EnterpriseOne and third-party systems using non-JD Edwards EnterpriseOne tables.
table event rules	Logic that is attached to database triggers that runs whenever the action specified by the trigger occurs against the table. Although JD Edwards EnterpriseOne enables event rules to be attached to application events, this functionality is application specific. Table event rules provide embedded logic at the table level.
terminal server	A server that enables terminals, microcomputers, and other devices to connect to a network or host computer or to devices attached to that particular computer.
three-tier processing	The task of entering, reviewing and approving, and posting batches of transactions in JD Edwards EnterpriseOne.
three-way voucher match	In JD Edwards Procurement and Subcontract Management, the process of comparing receipt information to supplier's invoices to create vouchers. In a three-way match, you use the receipt records to create vouchers.
transaction processing (TP) monitor	A monitor that controls data transfer between local and remote terminals and the applications that originated them. TP monitors also protect data integrity in the distributed environment and may include programs that validate data and format terminal screens.
transaction processing method	A method related to the management of a manual commit transaction boundary (for example, start, commit, rollback, and cancel).
transaction set	An electronic business transaction (electronic data interchange standard document) made up of segments.
trigger	One of several events specific to data dictionary items. You can attach logic to a data dictionary item that the system processes automatically when the event occurs.
triggering event	A specific workflow event that requires special action or has defined consequences or resulting actions.
two-way authentication	An authentication mechanism in which both client and server authenticate themselves by providing the SSL certificates to each other.
two-way voucher match	In JD Edwards Procurement and Subcontract Management, the process of comparing purchase order detail lines to the suppliers' invoices to create vouchers. You do not record receipt information.
user identification information	User ID, role, or *public.
User Overrides merge	Adds new user override records into a customer's user override table.
value object	A specific type of source file that holds input or output data, much like a data structure passes data. Value objects can be exposed (used in a published business service) or internal, and input or output. They are comprised of simple and complex elements and accessories to those elements.
variance	<p>In JD Edwards Capital Asset Management, the difference between revenue generated by a piece of equipment and costs incurred by the equipment.</p> <p>In JD Edwards EnterpriseOne Project Costing and JD Edwards EnterpriseOne Manufacturing, the difference between two methods of costing the same item (for example, the difference between the frozen standard cost and the current cost is an engineering variance). Frozen standard costs come from the Cost Components table, and the current costs are calculated using the current bill of material, routing, and overhead rates.</p>

versioning a published business service	Adding additional functionality/interfaces to the published business services without modifying the existing functionality/interfaces.
Version List merge	The Versions List merge preserves any non-XJDE and non-ZJDE version specifications for objects that are valid in the new release, as well as their processing options data.
visual assist	Forms that can be invoked from a control via a trigger to assist the user in determining what data belongs in the control.
vocabulary override	An alternate description for a data dictionary item that appears on a specific JD Edwards EnterpriseOne form or report.
wchar_t	An internal type of a wide character. It is used for writing portable programs for international markets.
web application server	A web server that enables web applications to exchange data with the back-end systems and databases used in eBusiness transactions.
web server	A server that sends information as requested by a browser, using the TCP/IP set of protocols. A web server can do more than just coordination of requests from browsers; it can do anything a normal server can do, such as house applications or data. Any computer can be turned into a web server by installing server software and connecting the machine to the internet.
Web Service Description Language (WSDL)	An XML format for describing network services.
Web Service Inspection Language (WSIL)	An XML format for assisting in the inspection of a site for available services and a set of rules for how inspection-related information should be made.
web service proxy foundation	Foundation classes for web service proxy that must be included in a business service server artifact for web service consumption on WAS.
web service softcoding record	An XML document that contains values that are used to configure a web service proxy. This document identifies the endpoint and conditionally includes security information.
web service softcoding template	An XML document that provides the structure for a soft coded record.
Where clause	The portion of a database operation that specifies which records the database operation will affect.
Windows terminal server	A multiuser server that enables terminals and minimally configured computers to display Windows applications even if they are not capable of running Windows software themselves. All client processing is performed centrally at the Windows terminal server and only display, keystroke, and mouse commands are transmitted over the network to the client terminal device.
wizard	A type of JDeveloper extension used to walk the user through a series of steps.
workbench	A program that enables users to access a group of related programs from a single entry point. Typically, the programs that you access from a workbench are used to complete a large business process. For example, you use the JD Edwards EnterpriseOne Payroll Cycle Workbench (P07210) to access all of the programs that the system uses to process payroll, print payments, create payroll reports, create journal entries, and update payroll history. Examples of JD Edwards EnterpriseOne workbenches include Service Management Workbench (P90CD020), Line Scheduling Workbench (P3153), Planning Workbench (P13700), Auditor's Workbench (P09E115), and Payroll Cycle Workbench.
work day calendar	In JD Edwards EnterpriseOne Manufacturing, a calendar that is used in planning functions that consecutively lists only working days so that component and work order scheduling can be done based on the actual number of work days available. A work

	day calendar is sometimes referred to as planning calendar, manufacturing calendar, or shop floor calendar.
workflow	The automation of a business process, in whole or in part, during which documents, information, or tasks are passed from one participant to another for action, according to a set of procedural rules.
workgroup server	A server that usually contains subsets of data replicated from a master network server. A workgroup server does not perform application or batch processing.
XAPI events	A service that uses system calls to capture JD Edwards EnterpriseOne transactions as they occur and then calls third-party software, end users, and other JD Edwards EnterpriseOne systems that have requested notification when the specified transactions occur to return a response.
XML CallObject	An interoperability capability that enables you to call business functions.
XML Dispatch	An interoperability capability that provides a single point of entry for all XML documents coming into JD Edwards EnterpriseOne for responses.
XML List	An interoperability capability that enables you to request and receive JD Edwards EnterpriseOne database information in chunks.
XML Service	An interoperability capability that enables you to request events from one JD Edwards EnterpriseOne system and receive a response from another JD Edwards EnterpriseOne system.
XML Transaction	An interoperability capability that enables you to use a predefined transaction type to send information to or request information from JD Edwards EnterpriseOne. XML transaction uses interface table functionality.
XML Transaction Service (XTS)	Transforms an XML document that is not in the JD Edwards EnterpriseOne format into an XML document that can be processed by JD Edwards EnterpriseOne. XTS then transforms the response back to the request originator XML format.
Z event	A service that uses interface table functionality to capture JD Edwards EnterpriseOne transactions and provide notification to third-party software, end users, and other JD Edwards EnterpriseOne systems that have requested to be notified when certain transactions occur.
Z table	A working table where non-JD Edwards EnterpriseOne information can be stored and then processed into JD Edwards EnterpriseOne. Z tables also can be used to retrieve JD Edwards EnterpriseOne data. Z tables are also known as interface tables.
Z transaction	Third-party data that is properly formatted in interface tables for updating to the JD Edwards EnterpriseOne database.

Index

A

- additional documentation x
- alert action rules
 - setting up 11
- alert action sequences
 - setting up 10
- alerts
 - completing 31
 - entering 19
 - processing 25
 - responding to 25
 - understanding 17
- application fundamentals ix

C

- CBM, *See* condition-based maintenance
- comments, submitting xiv
- common fields xiv
- condition-based alerts, *See* alerts
- Condition-Based Alerts Processor (R1312) 25
 - processing options 29
 - reviewing audit trail 30
- Condition-Based Alerts Revisions (P1311)
 - entering alerts manually 23
 - processing options 22
- Condition-Based Alerts Revisions form 23, 27
- Condition-Based Alerts Workbench (P1310)
 - creating investigation message 27
 - creating work order 28
 - processing options 20
 - updating preventive maintenance schedule 29
- condition-based maintenance
 - alerts 17
 - implementation steps 2
 - overview 1
 - process flow 18
 - system integrations 1
 - understanding 5
- Condition-Based Maintenance Alert Action Rules (P1318)
 - processing options 12

- setting up 12
- Condition-Based Maintenance Alert Action Rules Revisions form 12
- Condition-Based Maintenance Message Rules (P1316)
 - processing options 8
 - setting up 8
- Condition-Based Maintenance Message Rules form 8
- Condition-Based Maintenance Message Sequences (P1315)
 - processing options 6
 - setting up 7
- Condition-Based Maintenance Message Sequences form 7
- contact information xiv
- Create Investigation Message form 27
- Create W.O. from Model form 28
- cross-references xiii
- Customer Connection website x

D

- documentation
 - downloading x
 - related x
 - updates x
- downloading documentation x

I

- implementation guides
 - ordering x
- investigation messages
 - rules 8
 - setting up 6, 8
- investigation workflow 34

M

- message rules
 - setting up 8
- message sequences
 - setting up 6

N

- notes xiii
- notification messages

- rules 8
- setting up 6, 8
- notification workflow 33

P

P1310 (Condition-Based Alerts Workbench)

- creating investigation message 27
- creating work order 28
- processing options 20
- updating preventive maintenance schedule 29

P1311 (Condition-Based Alerts Revisions)

- entering alerts manually 23
- processing options 22

P1315 (Condition-Based Maintenance Message Sequences)

- processing options 6
- setting up 7

P1316 (Condition-Based Maintenance Message Rules)

- processing options 8
- setting up 8

P1318 (Condition-Based Maintenance Alert Action Rules)

- processing options 12
- setting up 12

PeopleCode, typographical conventions xii

prerequisites ix

Process Audit Trail form 30

R

R1312 (Condition-Based Alerts Processor)
25

- processing options 29
- reviewing audit trail 30

related documentation x

rules, *See* alert action rules, message rules

S

setup

- alert action rules 11
- alert action sequences 10
- condition-based maintenance 5
- investigation messages 5, 6
- message rules 8
- message sequences 6
- notification messages 5, 6

suggestions, submitting xiv

T

typographical conventions xii

U

Update P.M. Schedule form 29

V

visual cues xii

W

warnings xiii

workflow 33