IBM Unica Detect
Version 8 Release 5
July 31, 2012

Installation Guide

Note
Before using this information and the product it supports, read the information in "Notices" on page 79.

This edition applies to version 8, release 5, modification 0 of IBM Unica Detect (product number 5725-D16) and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Chapter 1. Contacting IBM Unica technical support

If you encounter a problem that you cannot resolve by consulting the documentation, your company's designated support contact can log a call with IBM® Unica® technical support. Use the information in this section to ensure that your problem is resolved efficiently and successfully.

If you are not a designated support contact at your company, contact your IBM Unica administrator for information.

Information to gather

Before you contact IBM Unica technical support, gather the following information:

- A brief description of the nature of your issue.
- Detailed error messages you see when the issue occurs.
- Detailed steps to reproduce the issue.
- Related log files, session files, configuration files, and data files.
- Information about your product and system environment, which you can obtain as described in "System information."

System information

When you call IBM Unica technical support, you might be asked to provide information about your environment.

If your problem does not prevent you from logging in, much of this information is available on the About page, which provides information about your installed IBM Unica applications.

You can access the About page by selecting **Help > About**. If the About page is not accessible, you can obtain the version number of any IBM Unica application by viewing the `version.txt` file located under the installation directory for each application.

Contact information for IBM Unica technical support

For ways to contact IBM Unica technical support, see the IBM Unica Product Technical Support website: (http://www.unica.com/about/product-technical-support.htm).

Chapter 2. Planning Your IBM Unica Detect Installation

This chapter provides a broad overview of the setup and configuration of IBM Unica Detect, to help you plan your installation and deployment.

About Detect

Detect allows your business to trigger targeted actions in response to complex patterns of activity discovered in incoming streams of transactions. Detect allows you to:

- Make decisions on incoming data quickly
- Configure the system to receive any kind of transactional input and to access existing static data
- Create and modify the central rules that control the triggering behavior using an intuitive web browser-based interface

About the Detect functional units

From the point of view of machine resources, Detect can be grouped into functional units. You can modify which components run on which unit but the relationships among the components makes certain configurations more efficient than others. In this list, the key components are grouped by the unit on which they most commonly reside:

- **Web Server Unit** — Enables web access to the user interface pages for system configuration and trigger building, testing, and deployment.
- **Cluster Unit** — Provides system processing. Each cluster unit has its own Feeder, Outcome Listener, and Engine components to eliminate possible input/output bottlenecks. Within each cluster unit, one or more engines process transactions using rules to search for matches to defined triggers and make heavy use of memory and CPU. The Feeder reads transactions from the data feed files and makes heavy use of input/output resources.You can set up Detect to run with multiple cluster units installed on multiple machines, as described in "About scaling with multiple cluster units" on page 5.
- **Database Unit** — Holds the Detect database. The database contains metadata for the system such as configuration information, rules, logging info, system state settings, and customer state. The database unit usually contains the outcomes that result from the triggers fired by the engine. The database unit makes heavy use of disk storage.

About the system components

This section lists the key Detect**process components**. Note that they are grouped by the functional unit (Web Server unit, Cluster unit, or Database unit) on which they most commonly reside.

Web Server unit

- **Microsoft Internet Information Server (IIS)**—The web server supported by Detect.
- **Enterprise Management Controller (EMC)**—The main process for Detect. EMC starts up all the other processes and implements the user interface for running the system. It starts and stops Detect batch execution, reports run status back to

the user interface, and monitors the TSM status of the running system. The EMC must always reside on the Web Server machine.

- **Trigger Set Manager (TSM)**—The TSM monitors the system; controlling the Feeder, Engine, and Listener for one workspace per run. The TSMs can be located on any machine.

Cluster unit

- **System Wrapper**—Implements ramp up and Outcome Management Tool (OMT) when invoked from the command line starting the main system.
- **Feeder**—Implements the functionality of the feeder. The feeder is the system that sends transactions to the rule engine.
- **Engine**—Acts as the main functional component of Detect. The engine sends the outcome to the Queue. It reads the rules from the database and receives transactions from the feeder. It matches the transactions with the rules and, when there is a match that triggers a rule, it sends the outcomes back to the database.

  Because the engine can run as multiple processes to improve efficiency, you can also configure multiple engines on one cluster unit machine. There can be as many engines on a cluster unit as are useful for your implementation. The optimum number is typically determined through testing. After the system rules have been configured, it is helpful to do test runs with increasing numbers of engines, noting the number of engine instances that results in a minimum in processing time. Too few engines leaves spare processing power on the unit. Too many engines robs enough resources from the other components (Feeder, Listener, and queue service) so that they in turn become the rate-limiting step, slowing overall throughput.

  The engine can be explicitly run but usually is run at specified intervals (for example, nightly) by a scheduler. When Detect starts up, it reads a set of configuration parameters from the Detect database. These parameters specify where the system can find all of its components as well as the rules that define the patterns to match, and where to look for the incoming data.

  The engine analyzes data from any or all of batch transactions and customer profiles. During a run, the engine uses the business rules to analyze the incoming data for specified patterns. When the engine finds matching patterns in the data, the triggers in the rule fire and the resulting action sends the specified outcome to the listener and then to an Outcome database. The contents of the Outcome database are used for reporting, or can be fed into other marketing and sales systems used by the organization.
- **Listener**—Picks outcome messages sent by the engine to Queue, and sends them to the database.
- **Queue**—Refers to Microsoft's Message Queue (MSMQ) service. The system uses the MSMQ for sending transaction and messages between the subsystems. MSMQ must be installed on every cluster unit.

Database unit

For performance reasons, the database unit generally does not run Detect components although there is no reason that it cannot if it makes sense in your application. The system needs to know where the database unit is configured.

About configuration options

Detect is installed on each of the machines that run Detect components. You can specify the components that run on a particular machine using the Configuration Utility.

Configuration options are very flexible. You can set up all of these components on a single machine or use a configuration of multiple machines to optimize system performance. As the processing requirements of your application increase, so can the number of cluster units included in your configuration. Also, you may decide to combine two functional units onto one machine. The final decisions on how to set up your system should take into account your system resources as well as the throughput needed by your system.

About scaling with multiple cluster units

Most enterprise Detect implementations use a single cluster unit, but customers with large transaction volumes or tight processing windows may deploy multiple clusters units on multiple machines. In a Detect environment with multiple cluster units, the cluster units typically share a single database and a single web server unit consisting of a Trigger Set Manager (TSM), an Enterprise Management Controller (EMC), and a single IIS instance.

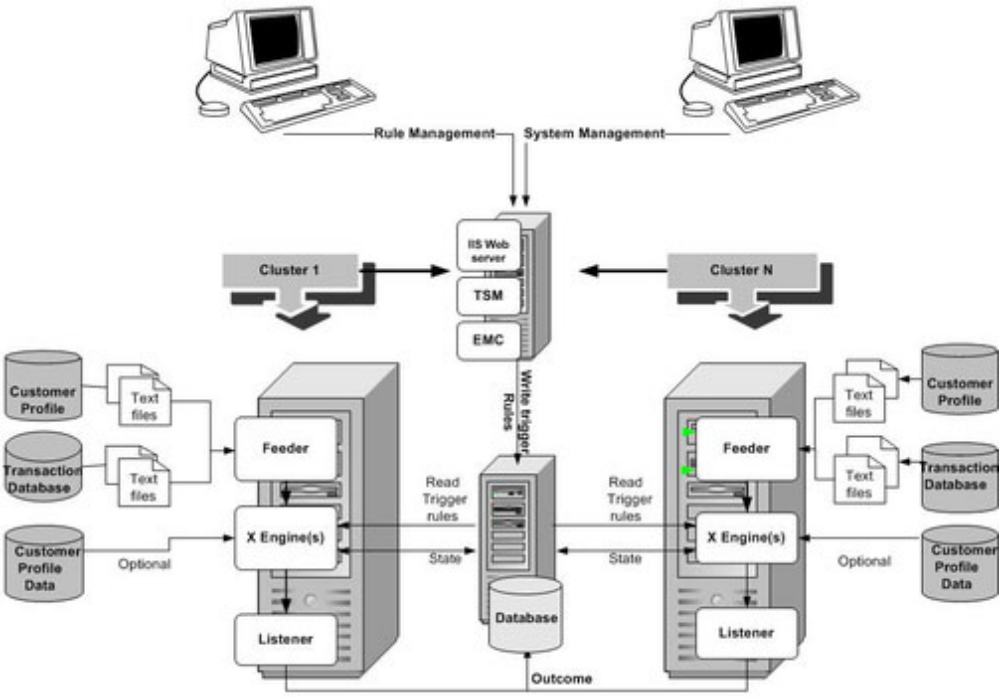
For details on installing multiple cluster units, see Chapter 10, "Configuring Multiple Cluster Units," on page 43.

Typical configuration

You can install all functional units on one machine, or distribute them across multiple machines. The web server should be part of one of the cluster units, and you can distribute the other functional units (cluster units and database unit) so that they reside on their own machines. As the processing requirements of your application increase, you can add cluster units to your configuration.

Note: This document shows the units as logically distinct. However, the web server should physically reside on one of the cluster units.

The following image illustrates one of the most common and most efficient ways to set up your configuration.


Rule Management
System Management
IIS Web server
TSM
EMC
Cluster 1
Cluster N
Write trigger Rules
Customer Profile
Text files
Transaction Database
Customer Profile Data
Optional
Feeder
X Engine(s)
Listener
Read Trigger rules
State
Database
Outcome

Chapter 3. Preparing to Install IBM Unica Detect

Before you start to install IBM Unica Detect, verify that your environment meets requirements as described in this section in the *IBM Unica Detect Recommended Software Environments and Minimum System Requirements* document for this version of Detect.

- Ensure that Internet Explorer is installed on the machine(s) where you plan to use the Detect web application.
- Ensure that an Oracle or SQL Server database is available so that you can create a schema for the Detect system tables. Scripts for creating these tables are provided with your downloaded software.
- Ensure that MSMQ message queuing is installed on all of the machine(s) where you plan to install Detect components.

  To check, open **Control Panel > Add Remove Programs > Add Remove Windows Components**. Click **Application Server** and verify that **Message Queuing** is checked. Use the default settings.

  If MSMQ message queuing is not installed, follow the installation instructions provided in Appendix C, "Installing MSMQ," on page 77.

If you are upgrading

If Detect is already installed on the targeted machines, please contact IBM Unica Consulting Services before continuing. Because there are many environments and versions, each upgrade should be handled on a case-by-case basis. Appendix A, "Upgrading IBM Unica Detect," on page 65 provides some information about the requirements.

About installing tools

Detect includes several tools, including the Library Management Tools. Some of the tools are automatically installed with Detect and others you must install if you want to use them. For information on finding, and if necessary installing, those tools see the *IBM Unica Detect Administrator's Guide*.

Detect installation process overview

1. Identify machines and components. You must identify which machines you will use in the installation and which Detect components you will place on each machine. The previous chapter provides information to help you make these decisions.
2. Verify system prerequisites on all machines that will run Detect components. See Chapter 3, "Preparing to Install IBM Unica Detect."
3. Chapter 4, "Creating the IBM Unica Detect Database," on page 9.
4. Only on 64-bit machines, perform some configuration. See "Task: Configure role services and the application pool (64-bit machines only)" on page 11
5. Create a Detect system user on each machine that will run Detect components to enable communication among Detect components. See "Task: Create a Detect system user" on page 12

6. Install Detect. This procedure differs depending on whether you are installing a web server unit or a cluster unit. See "Task: Run the Detect installer on the web server unit" on page 13 and "Task: Run the Detect installer on the cluster units" on page 27
7. Set installation directory security on Detect components for the required users. You grant permissions to a different set of users depending on whether you are installing a web server unit or a cluster unit. See "Task: Set installation directory security on the web server unit" on page 15 and "Task: Set installation directory security on the cluster units" on page 28.
8. Set sharing on the installation directory for the required users. You grant access to a different set of users depending on whether you are installing a web server unit or a cluster unit. See "Task: Set sharing in the installation directory on the web server unit" on page 17 and "Task: Set sharing in the installation directory on the cluster units" on page 29.
9. Configure DCOM for the required users. The way you set DCOM identity and security differs depending on whether you are installing a web server unit or a cluster unit. See "Task: Configure DCOM settings" on page 19.
10. On the web server unit only, configure IIS and install Crystal Reports. See "To create the virtual directory" on page 21, "To set properties on the virtual directory" on page 21, and "To set the Web Service Extension (IIS6.0 only)" on page 22, and "Task: Run the Crystal Reports installer" on page 23.
11. Create the Detect system tables. On the machine that hosts the Detect database, customize and then run the database configuration scripts. See Chapter 7, "Creating the IBM Unica Detect System Tables in Oracle," on page 33 or Chapter 8, "Creating the IBM Unica Detect System Tables in SQL Server," on page 39.
12. Configure users in the Detect web application. See Chapter 9, "Creating an Initial IBM Unica Detect User," on page 41.

Note: After you complete the installation, there are some configuration tasks that you must perform, such as defining entity types and configuring the clusters, before you can run the engine. You must also log into Detect and perform the tasks described in the *IBM Unica Detect Administrator's Guide* and the *IBM Unica Detect User's Guide*.

Chapter 4. Creating the IBM Unica Detect Database

IBM Unica Detect requires a database to hold its system tables. Oracle and Microsoft SQL Server are the two supported database types.

Have your database administrator create a database for Detect. Typically, all of the Detect database schemas are run in the same database. However, the outcome schema can be redirected to any user and table space on any database.

After you have run the Detect installer as many times as required on the machine or machines you have designated for your Detect configuration, you will run SQL scripts provided with your Detect installation to create the system tables. Instructions are provided in the chapter that applies to your database type.

Note: Make a note of the database server name and the database (schema) name, as you must enter these when you run the Detect installer.

Configure the database for double byte characters, if necessary

If your installation must support double byte characters (for example, Chinese or Korean), set up your database accordingly.

If your database is Oracle, set the `NLS_LANG` property for your Oracle client. Consult the Oracle documentation for details.

For additional information about configuring Detect for multi byte characters, see Appendix B, "Localizing Your Detect," on page 73.

About creating an Oracle database for Detect

Creating a new Oracle database (or instance) is beyond the scope of this document, so if you are unfamiliar with how to do this, contact your database administrator to perform this task. This section provides some guidelines specific to creating a database for Detect in Oracle.

You must also obtain an administrator user name and password for the database, so that you can run the SQL scripts that create the Detect users and schema in a later step.

- You can use the default tablespaces that Oracle provides, or you can create tablespaces especially for Detect. If you define your own tablespaces, you should make a note of their names, because you need them when you edit the SQL scripts in a later step.

  For instructions defining tablespaces specific to the Detect application, see "To create tablespaces in Oracle" on page 10.

  Note: During the installation of Oracle, if you create and use the default instance of the database you do not have to configure Net 8. If you create your own instance, then you do need to configure Net 8, as described in the chapter, Chapter 7, "Creating the IBM Unica Detect System Tables in Oracle," on page 33.
- If you have Apache running, be sure that you disable it when you install Oracle. If Apache is running during the Oracle installation it can result in a conflict that will cause Detect to fail.

- Install the Oracle client on all the other machines used in the Detect configuration. Install the 32-bit Oracle client, even if your Windows server is 64-bit. On the Oracle server, install it to a different Oracle home directory from the 64-bit client.
- If you are configuring Detect to support multi byte languages, change the Oracle client setting in the Windows registry from `NLS_LANG = AMERICAN_AMERICA.WE8MSWIN1252` to `NLS_LANG = AMERICAN_AMERICA.AL32UTF8`

To create tablespaces in Oracle

Important: Make a note of the names of the tablespaces that you create, because you need them when you edit the SQL scripts in a later step.

1. Open the Oracle Enterprise Manager Console Login window.
2. Select **Launch standalone** and click **OK** to start the Oracle Enterprise Manager Console, Standalone.
3. Click the **+** to display a list of the databases.
4. Double-click the name of the Detect database to open the Database Connect Information window.
5. Log in to the Detect database.
6. Click the **+** next to **Storage**.
7. Right-click **Tablespaces**, then select **Create** from the pop-up menu to display the Create Tablespace window.
8. Create a new tablespace as follows.
   a. In the **Name** text box, enter the name for the tablespace.
   b. After you enter the tablespace name, the name appears in the **Datafiles** table as the name given to the file created to store the contents of the tablespace. Double-click the name of the tablespace that you just created to display the Create Datafile window.
   c. In the **File Size** text box, enter 5, then click the drop-down box next to the **File Size** box and select ***MBytes***.
   d. Select the **Storage** tab to display the Storage window.
   e. Select the **Automatically extend datafile when full** checkbox, enter 5 in the **Increment** text box, and select **MBytes** in the drop-down box. Click **OK** to close the window and return to the Create Tablespace window.
   f. Click **Create** to create the new tablespace.
   g. Click **OK** to close the Tablespace Created Successfully message window.
9. Repeat steps 7 through 8 to create the index tablespace.

Chapter 5. Installing the IBM Unica Detect Web Server Unit

A web IBM Unica Detect server unit consists of IIS (Microsoft Internet Information Services) and the Enterprise Management Controller (EMC) and Trigger Set Manager (TSM) Detect components installed on one machine.

Follow the steps in this section in the order shown to install a Detect web server unit. A cluster unit is typically installed on the same machine with a web server unit. For those machines, in addition to the steps in this chapter, set sharing on the installation directory and configure DCOM settings as described in the chapter, Chapter 6, "Installing the IBM Unica Detect Cluster Units," on page 25.

Important: If you are upgrading, and Detect is already installed on the targeted machines, you should contact Unica Consulting Services before continuing. Because there are many environments and versions, each upgrade should be handled on a case-by-case basis. See Appendix A, "Upgrading IBM Unica Detect," on page 65 for information about upgrading.

Task: Verify installation requirements

Before beginning the installation process, verify that your environment meets the requirements described in Chapter 3, "Preparing to Install IBM Unica Detect," on page 7. Also, for a detailed set of operating system, web application server, and database requirements, see the *IBM Unica Detect Recommended Software Environments and Minimum System Requirements* document for this version of the software.

Note: Detect supports Secure Sockets Layer (SSL) certificate authentication. The steps for configuring it on your system are beyond the scope of this document. If you need help, contact Unica Technical Support.

Task: Configure role services and the application pool (64-bit machines only)

Perform these steps on all 64-bit machines where you plan to install Detect components. This step is necessary only on 64-bit machines, not on 32-bit machines.

1. Log into the machine as a user with administrative privileges on the machine.

2. Open the Windows Server Manager, right-click **Roles > WebServer**, and select **Add Role Services**
3. In the Select Role Services window, check the checkboxes for the following services.
   - ASP.NET
   - ASP
   - ISAPI Extensions
   - ISAPI Filters
   - IIS 6 Metabase Compatibility
4. Close the Windows Server Manager.
5. Open the Internet Information Services (IIS) Manager and select your application pool.

   If you have not created your own application pool, the default is **Application Pools**.
6. Click **Set Application Pool Defaults**.
7. In the **General** section, set **Enable 32-bit Applications** to **True**.

Task: Create a Detect system user

Before you install the Detect software, you must designate a network or local Windows user for Detect. This Detect system user must have administrative privileges on the machine to enable proper communications among the Detect components. Although you can use any existing user with administrative privileges, you should create a user for Detect to make system setup and maintenance a little clearer.

Use this section to create a Detect system user and add that user to the Administrators group.

1. Log in to the machine as a user with administrative privileges on the machine.
2. Right-click the **My Computer** system icon on the desktop and select **Manage** from the pop-up menu to display the Computer Management window.
3. Expand **Local Users and Groups**, then click the **Users** directory to display the list of users.
4. Click **Action > New User** to display the New User window.
5. Create the new user using these steps.
   a. In the **User name** text box, type the name for the user.

      You can use any name you want; the examples in this guide refer to this user as the Detect system user. Be sure to record the name of the Detect system user; you will need this name during the installation process and when you run the system.

      Note: If you are running Detect components on more than one machine, use the same name and permission for the Detect system user for each machine.
   b. In the **Password** text box enter a password and enter the same password in the **Confirm Password** text box.

      Make a note of the password, as you will log in as this user later in the installation process.
   c. Clear the checkbox **User must change password at next logon**.
   d. Select the checkbox **Password never expires**.

Note: If you require that passwords expire, you must periodically manually change the Detect system user password as well as change that password in each of the DCOM components described later in this chapter.

e. Click **Create** to create the user.

f. Click **Close** to close the New User window and return to the Computer Management window.

6. In the Computer Management window, expand **Groups** to display the list of groups, and then double-click the Administrators group to open the group's Properties window, which displays a list of members..
7. If your Detect system user is not listed as a member, add this user as follows.
   a. Click **Add** to open the Select Users, Computers, or Groups window.
   b. Click **Locations** and select the computer you are on (at the top of the list).
   c. Enter the name of your Detect system user and click **OK**.

      Your user is added to the Administrators group.

Copying installation files (DVD only)

If you received your IBM Unica installation files on a DVD, or if you created a DVD from a downloaded ISO image file, you must copy its contents to a writable directory available to the system on which you will be installing the IBM Unica products before running the installers.

You cannot run IBM Unica Marketing installers directly from read-only media, such as the installation DVD, an ISO image mounted read-only, or a write-restricted directory or volume.

Task: Run the Detect installer on the web server unit

Before you run the Detect installer, do the following.

- Stop the `IIS Admin` service.
- If the Services window is open, close the window before starting the installation.

Follow the guidelines when you install the Detect web server unit.

1. Log in to the machine as a user with administrative privileges on the machine.
2. Run the `Unica_Detect_`*`n.n.n.n_OS`*`.msi` file, where *`n.n.n.n`* is the version number and *`OS`* is the operating system.

   Note: Run the installer from a logical drive (not a network share). Mapped network drives are acceptable, but a drive letter is required.
3. Select the **IIS Server** checkbox.
4. Complete the fields as described in "Detect installer screen reference" on page 14.

   Note: When you install Detect on a Windows 64-bit machine, and accept the default location, Windows automatically installs it in the `C:\Program Files (x86)\Unica` folder. If this is the case in your environment, you must edit a file located in the `libararymanager\conf` directory. Edit either the `detect.oracle.config` or `detect.sqlserver.config` file, depending on your database type. Edit the file by changing the value of the `db.home.dir` property from its default value of `c:/Program Files/Unica/Affinium Detect/Application/Database` to c`C:/Program Files (x86)/Unica/Affinium Detect/Application/Database`.

Detect installer screen reference

The following table describes the fields in the Detect installation wizard

Field name	Description
IIS Server	Select this checkbox when you install Detect on the the machine where IIS is installed. De-select it when you install on a machine you will configure as a cluster unit. Each time you run the Detect installer, it places all of the Detect components under the directory you specify. If you select the **IIS Server** checkbox, the installer also provides the database scripts, creates the reports and recovery folders, and updates the web configuration files. Later, when you configure Detect, you specify which machines are web server units and which are cluster units.
Database Type	Select either **SQL Server** or **Oracle**, depending on the type of your Detect database.
Server name	Enter the name of the database server. This is required if you are using SQL Server. Although this field is optional if your Detect Database is in Oracle, as a best practice you should enter the server name so that the name will appear the registry. The installer replaces a parameter in the SQL scripts with the value you enter here.
Database name	Enter the name of the Detect database. The installer replaces a parameter in the SQL scripts with the value you enter here.
Vendor	This is used internally by Detect. The vendor code must be any three upper case letters from the English alphabet. The installer replaces a parameter in the SQL scripts with the value you enter here.
Rule Schema User Name	The name for the Rule schema user that will be created when you run the SQL scripts to create the Detect system tables. The installer replaces a parameter in the SQL scripts with the value you enter here. The default value is RuleUser. If you are performing an upgrade, enter the user name that is used in the existing installation. (In earlier versions of Detect, this user name was CEE4.)
History Schema User Name	The name for the History schema user that will be created when you run the SQL scripts to create the Detect system tables. The installer replaces a parameter in the SQL scripts with the value you enter here. The default value is HistoryUser. If you are performing an upgrade, enter the user name that is used in the existing installation. (In earlier versions of Detect, this user name was CEE1.)
Outcome Schema User Name	The name for the Outcome schema user that will be created when you run the SQL scripts to create the Detect system tables. The installer replaces a parameter in the SQL scripts with the value you enter here. The default value is OutcomeUser. If you are performing an upgrade, enter the user name that is used in the existing installation. (In earlier versions of Detect, this user name was CEE3.)

Field name	Description
Database password	The password that will be set for the three database schema users listed above when you run the SQL scripts to create the Detect system tables, if you are performing a new installation. The installer replaces a parameter in the SQL scripts with the value you enter here. If you are upgrading an existing Detect installation, and you have assigned different passwords to the three schema users listed above, then you should enter the password currently in use for the Rule Schema user.
Folder	Select a directory under which you want Detect to be installed. You can click the **Disk Cost** button to determine whether the drive on which you are installing has sufficient space. When you install Detect on a Windows 64-bit machine, and accept the default location, Windows automatically installs it in the `C:\Program Files (x86)\Unica` folder. If this is the case in your environment, you must edit the `libararymanager\conf` file under your Detect installation to change the value of the `db.home.dir` property from its default value of `c:/Program Files/Unica/Affinium Detect/Application/Database` to `c:/Program Files (x86)/Unica/Affinium Detect/Application/Database`.

Task: Unzip the Library Manager

Extract the LibraryManager zip archive to your Detect installation folder. While this location is not required, it is convenient to have all of your Detect files in the same location

Unzipping the files creates a `librarymanager` directory with several sub-directories.

Task: Set installation directory security on the web server unit

You must set security for the `Affinium Detect` directory on the web server unit as described in this section. Read the introductory topics to learn how to determine what security settings are required for your environment. Then follow the instructions in the procedure "To set installation directory security on the web server unit" on page 16.

Settings required on all machines

On **every** machine where you have installed Detect components, you must grant full access to the `Affinium Detect` directory to the following users.

- Detect system user—the Windows user you created in an earlier step.
- `NETWORK SERVICE` (IIS 6.0 only)–A built-in account that has reduced privileges.
- `ASPNET` (IIS 5.0 only)– A user account created by Microsoft .NET Framework to limit the access rights of .NET applications.
- Administrators group—The group of administrative users on the machine. The user who runs Detect's Outcome Manager Tool must be a member of this group, because the OMT needs the permissions you will grant to this group to be able to do bulk loads to the database.

Settings required only on the web server unit

Only on the web server unit where IIS is installed, grant full access to the `Affinium Detect` directory to `IUSR_`*`machine_name`* . When a user attempts to connect to your public Web site, your Web server assigns the user to the Windows user account called `IUSR_`*`machine_name`* , where *machine_name* is the name of the server on which IIS is running. Note that if you have IIS installed on more than one machine, you will have a different IIS user on each of these machines.

About security settings in non-default environments (IIS 6.0 only)

You must read this section if your version of IIS is 6.0 and either of the following is true on the web server unit.

- Windows is not using the default directory for temporary files, (`C:\Windows\Temp` on 32-bit machines).
- The IIS user identity is not the default user (`NETWORK SERVICE`).

If neither of the above is true for your environment, you can proceed to the procedure described in "To set installation directory security on the web server unit." If either is true, read the sections below to learn how modify the procedure for setting installation directory security.

Non-default temporary directory

If Windows is not using the default directory for temporary files, (`C:\Windows\Temp` on 32-bit machines), grant Read/Write permission to the `NETWORK SERVICE` user, or the designated IIS user, on the non-default Windows temporary directory.

Non-default IIS user

You can determine the IIS 6.0 user identity as follows.

- Open Internet Information Services and select the machine.
- Right-click the `Application Pools` directory and select **Properties**.
- In the Identity tab, the Application pool identity setting identifies the security account being used.

If the IIS user identity is not `NETWORK SERVICE`, grant Read/Write permissions to that user on the following two directories:

- Your Detect installation directory
- The Windows temporary directory

To set installation directory security on the web server unit

Perform the following steps on the web server unit.

1. Log in to the machine as a user with administrative privileges on the machine.
2. Right-click the `Affinium Detect` directory then select **Properties** to display the **Properties** window.

   Do this on the `Affinium Detect` directory, not the `Unica` directory which typically contains the `Affinium Detect` directory.
3. Select the Security tab and click **Add** to display the Select Users or Groups window.
4. Click **Locations**.

a. In the window that opens, click the name of the local machine for local accounts you want to add, or select the domain the Detect system user account is under if it is not on the local machine.
b. Click **OK**.
c. The **Select Users or Groups** window appears again.

5. Click **Advanced**.
6. Click **Find Now** to to display the Select Users or groups window.
7. Select the following users and group.
   - IUSR_*machine name* , where *machine name* is the name of the server on which IIS is running.

     If you have IIS installed on more than one machine, as you might have if you are setting up multiple environments as described elsewhere in this guide, you will have multiple IIS users, and you must grant access to each of these IIS users on every machine where you have Detect components installed.
   - NETWORK SERVICE (IIS 6.0 only)
   - ASPNET (IIS 5.0 only)
   - Detect system user you created in an earlier step
   - Administrators group

   The IUSR_*machine name* , NETWORK SERVICE (for IIS 6.0) , and ASPNET (for IIS 5.0) users and the Administrators group are all under the directory with the local machine name.

   If the Detect system user is a local account, it is also under this directory. However, if it is a network account, you can find it under the network path for that user in this Locations dialog box. You can add users from only one location at a time.
   a. To select, hold down the **Control** key and click each.
   b. Click **OK** to see the object names listed in the window.
8. Click **OK** to add the selected users and group to the list of names on the Security list and return to the Security screen.
9. Set the permission for each of the users and the group that you just added to the Name list:
   a. In the Group or user names section, click the name of one of the users or the group that you just added to the list.
   b. In the Permissions list, next to **Full Control**, select the checkbox under the column **Allow** to give the selected user full security permission. When you select **Full Control**, all the other checkboxes are automatically selected.
   c. Click **Advanced**. Confirm that the checkbox to **Allow inheritable permissions** is selected. If it is not selected, then select it.
   d. Repeat these steps for each user that you added to the Name list.
10. Click **OK** to complete the security settings and close the Properties window.

Task: Set sharing in the installation directory on the web server unit

On the web server unit, set the sharing on the Affinium Detect directory so that full access is granted to the same users or group that you gave security privileges in the previous section

- The Detect system user you created in a previous step
- NETWORK SERVICE (IIS 6.0 only)
- ASPNET (IIS 5.0 only)

- IUSR_*machine_name* , where *machine_name* is the name of the server on which IIS is running.

  Note that if you have IIS installed on more than one machine, you will have a different IIS user on each of these machines.
- Administrators (group)

1. Log in to the machine as a user with administrative privileges on the machine.
2. Locate the Affinium Detect directory, right-click it, and select **Properties** to display the directory's Properties window

   Do this on the Affinium Detect directory, not the Unica directory which typically contains the Affinium Detect directory.
3. Select the Sharing tab on the Properties window.
4. Select the option **Share this folder** to enable sharing, and then click **Permissions** to open the Share Permissions window.
5. Select **Everyone** in the list of groups or users, and click **Remove**.
6. Click **Add** to open the Select Users or Groups window.
7. Click **Locations**.
   a. In the window that opens do the following .
      - For local accounts you want to add, click the name of the local machine
      - If the Detect system user account is not on the local machine, navigate to the machine where it exists.
   b. Click **OK**.

      The **Select Users or Groups** window appears again.
8. Click **Advanced**.

   The IUSR_*machine_name* , NETWORK SERVICE, and ASPNET (for IIS 5.0) users are all under the directory with the local machine name.

   The Administrators group is also under this directory with the local machine name.

   The Detect system user is under this directory as well, if it is a local account. If it is a network account, you can find it under the proper network path for that user in this Locations dialog box. You can add users from only one location at a time.
9. Click **Find Now** to display a list of all the local users.
10. Scroll down the list of names and find the name of one of the users or group that you are adding to give sharing privileges.

    Note: If the Detect system user is not on the local machine, select **Locations** and choose the network domain the user is on.
11. Select a user or group and then click on **OK**. Repeat this step for each of the users or the group listed in the introduction to this procedure.
12. Click **OK** to add the selected users or group to the list on names on the Sharing list and return to the Sharing window.
13. Set the Share permission for each of the users or group that you just added to the Name list.
    a. In the Name pane, click on one of the users or group that you just added to the list.

       In the Permissions list, next to **Full Control**, select the checkbox under the column **Allow** to give the selected user full sharing permission. When you select **Full Control**, all the other checkboxes will be selected automatically.
    b. Repeat these steps for each user or group that you added to the Name list.

c. Click **OK** to complete the security settings and close the Sharing window.

14. Click **OK** to close the Properties window.

Task: Configure DCOM settings

DCOM enables and coordinates communication between software processes across servers.

About DCOM identity and security in Detect

DCOM security grants Windows users permission to access, launch, and configure components. For Detect, the easiest way to set DCOM security is to grant full control on all Detect components to the same users you specified when you set sharing on the installation directory.

These users are:

- `NETWORK SERVICE`

  This is done on both cluster units and web server units, but applies only when your web server is IIS 6.0.

- `ASPNET`

  This is done on both cluster units and web server units, but applies only when your web server is IIS 5.0.

- `IUSR_`*`machine_name`*, where *`machine_name`* is the name of the server on which IIS is running.

  This is done only on web server units. Note that if you have IIS installed on more than one machine, you will have a different IIS user on each of these machines.

- Administrators (group)

  This is done on both cluster units and web server units. This grants access to the Detect system user because you made that user a member of the Administrator's group in a previous step.

DCOM identity determines the Windows user account(s) that can run a component. Windows provides three options for determining which account runs a component. For Detect, you select **This User**, which allows you to specify the Detect system user as the account whose security context is used to run the component. Because you made the Detect system user a member of the Administrators group in a previous step, this account has the correct privileges to perform all the actions required for Detect.

What to look for in the DCOM Component Services window

When you set DCOM identity and security, you select the Detect components from a list in the Component Services window. The Detect components are listed as follows.

- EMC
- TSM
- EngineController
- FeederWrapper
- ListenerWrapper
- SystemWrapper

See “To set DCOM identity and security” for details on performing this procedure.

To set DCOM identity and security

You should set both security and identity on all Detect components as described in this procedure. While it is not strictly required to configure DCOM for all components as described here, this method helps prevent access issues that are difficult to diagnose.

See “Task: Configure DCOM settings” on page 19 for more information about the DCOM settings.

1. Log in to the machine as a user with administrative privileges on the machine.
2. Open a command window and run the `dcomcnfg` command to open the Component Services window.
3. Expand Component Services in the left panel until you get to the **DCOM Config** portion.
4. Right-click each Detect component, select **Properties** to open the Properties window.
5. Set Launch and Activation Permissions **for each Detect component** as follows.
   a. On the **Security** tab, under Launch and Activation Permissions, select the **Customize** option and click **Edit** to open the Launch Permissions window.
   b. Add the users described in “Task: Configure DCOM settings” on page 19.
   c. Click **OK**.
   d. For each of the users, set all of the permissions to **Allow** and then click **OK** to return to the Properties window.
6. Set Access Permissions **for each Detect component** as follows.
   a. On the Security tab, under **Access Permissions**, select **Customize** and then click **Edit**.
   b. For each of the users, repeat the process you used to give them launch permissions in order to give them each access permissions.
   c. Click **OK** to return to the Properties window.
7. Set identity **for each Detect component** as follows.
   a. Select the Identity tab, and choose **This user**.
   b. Click **Browse** to find the Detect system user.
   c. Enter the password for the Detect user in the **Password** and **Confirm Password** boxes, click **Apply** and then click **OK**.

Task: Configure IIS (32-bit machines only)

Perform the procedures in this section if your web server unit is on a 32-bit machine using IIS 5 or IIS 6.

If your web server unit is on a 64-bit machine, see “Task: Configure IIS (64-bit machines only)” on page 22.

To configure IIS on a 32-bit machine, you must create a virtual directory for Detect, set the directory properties, and, if you are using IIS 6.0, enable the Web Service Extensions, as follows.

- “To create the virtual directory” on page 21
- “To set properties on the virtual directory” on page 21

• "To set the Web Service Extension (IIS6.0 only)" on page 22

To create the virtual directory

1. Click the Windows **Start** button and select **Programs > Administrative Tools > Internet Information Services (IIS) Manager**.

   The Internet Information Services (IIS) Manager window opens.
2. Expand the the local computer in the tree on the left side, and expand **Web Sites**.
3. Right-click **Default Web Site** and select **New > Virtual Directory**.

   The Virtual Directory Creation Wizard opens.
4. Click **Next**.
5. Under **Alias**, set the name for the System (for example: Detect) and click **Next**.
6. Browse to the `Application` directory under your Detect installation directory and click **OK**.
7. Click **Next** and set the Access Permissions as follows.
   - Allow **Read, Run scripts (such as ASP)**, and **Execute (such as ISAPI applications or CGI)**.
   - Do not allow **Write** or **Browse**.
8. Click **Next**, and then click **Finish**.

The Virtual Directory is now created under the Default Web Site.

To set properties on the virtual directory

Follow these steps to set the properties of the newly created virtual directory.

1. Select the newly created virtual directory.
2. Right-click and select **Properties**.

   The *<Alias Name>* Properties property window opens.
3. On the **Virtual Directory** tab, verify or adjust the settings as follows.
   - Clear the **Log visits** checkbox.
   - Clear the **Index this Resource** checkbox.
   - Verify that the **Read** checkbox is selected.
   - (IIS 6.0 only) In the Application Pool list box, select the **DefaultAppPool** option.
4. Click **Configuration**.

   The Application Configuration property window opens.
5. Select the Options tab and adjust or verify the settings as follows.
   - Set the **Session timeout** edit box to 60 minutes.
   - Select the **Enable parent paths** checkbox.
   - Set the **ASP Script timeout** edit box to 18000 seconds.
6. Click **OK**.

   The Application Configuration property page closes.
7. Select the **Directory Security** tab.
   - Click **Edit** under the Authentication and Access control to open the Authentication Methods dialog.
   - Verify that the **Integrated Windows authentication** option is not checked.
   - Click **OK** to close the Authentication Methods dialog.
8. Select the ASP.NET tab and adjust or verify the setting as follows.

- In the ASP.NET version setting, ensure that 2.0 is selected.
- If it is not, then select it and click **Apply**.

9. Select the Documents tab and adjust or verify the setting as follows.
   - Ensure that the **Enable default content page** checkbox is selected.
   - Remove any existing content pages.
   - Add **login.aspx** as the enabled default content page and click **Apply**.
   - Click **OK** to close the window
10. Click **OK** again to close the <*Alias Name*> Properties property sheet.

To set the Web Service Extension (IIS6.0 only)

Perform this task to ensure that the Web Service Extensions are enabled.

Note: This task applies to IIS 6.0 only. If you are using IIS 5.0, skip this procedure.

1. In the Internet Information Services tree, click **Web Service Extensions**.
2. Ensure that **Active Server Pages** and **ASP.NET v2.0** are listed and that both are set to **Allowed**.

   To change an item from Prohibited to Allowed, select it and click **Allow**. If you make a change, click **Web Service Extensions** again before continuing to the next step.
3. If ASP.NET v2.0 is not listed, then use this step. Otherwise, skip to step 4.
   a. Click the **Add a new Web service extension** link.
   b. When the New Web Service Extension window opens, click **Add**.
   c. Browse to the `aspnet_isapi.dll`, which by default is in:

      `C:\WINDOWS\Microsoft.NET\Framework\v2.0.`*n*`\aspnet_isapi.dll`

      In this step, *n* represents the rest of the version number. Make a note of the version number so you can use it in step e.
   d. Click **OK**.
   e. Give it the Extension name of `ASP.NET v2.0.`*n*, replacing *n* with the rest of the version number.
   f. Select the **Set the extension status to Allowed** checkbox if it is not already selected.
   g. Click **OK** and close the window.
4. Close the Internet Information Services dialog.

Task: Configure IIS (64-bit machines only)

Perform the procedure in this section if your web server unit is on a 64-bit machine using IIS 7.

If your web server unit is on a 32-bit machine, see "Task: Configure IIS (32-bit machines only)" on page 20.

1. Open Internet Information Services (IIS) Manager, select **Sites > Default Web Site** in the Connections panel on the left, right-click and select **Add Application** from the drop-down menu.

   The Add Application window opens.
2. Complete the fields as follows and then click **OK**.
   - **Alias** — enter `Detect`.
   - **Physical path** — Select the Detect installation directory.

   The Detect site is displayed in the Connections panel.

3. Select the **Detect** site in the Connections panel to open the Detect home page in the center panel, and double-click the **ASP** icon.

   The ASP window opens.
4. Change the **Enable Parent Paths** property to **True** and then click **Apply** in the Actions panel on the right.
5. Select **Detect** site in the navigation bar above the center panel to return to the Detect home page, and double-click the **Default Document** icon.

   The Default Document window opens.
6. Remove any existing pages and add `Login.aspx`.
7. Double-click **Application Pools** in the Connections panel to open Application Pools in the center panel, select **DefaultAppPool**, and then click **Advanced Settings** in the Actions panel.

   The Advanced Settings window opens.
8. In the General section, do the following.
   - Ensure that **Enable 32-bit Applications** is set to **True** for the application pool, and that the pool is started.
   - Verify that the Identity property is set to **NetworkService**, and change it if it is not.

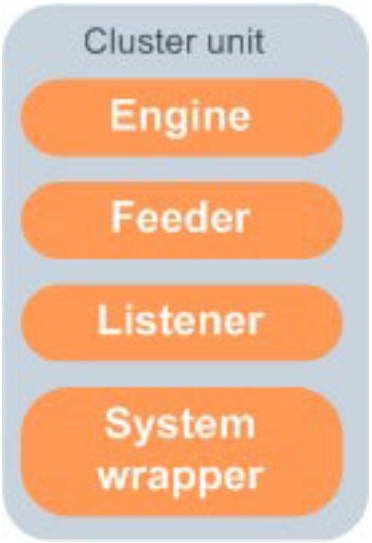
Task: Run the Crystal Reports installer

Perform the following steps, only on the machine or machines where IIS is installed, to install Crystal Reports.

1. Locate the `CRRedist2008_x86.msi` file, included in the Detect installation package
2. Run the file on the IIS machine.

Chapter 6. Installing the IBM Unica Detect Cluster Units

An IBM Unica Detect cluster unit consists of the Detect feeder, engine, listener, and system wrapper installed on one machine, as shown below.



Follow the steps in this chapter in the order shown to install a Detect cluster unit.

Note that many of the steps are exactly the same as those you performed to install a web server unit. However, the steps for setting security and sharing on the installation directory are different, and you do not configure role services, set the Internet Service, or install Crystal Reports on the cluster units.

Important: If you are upgrading, and Detect is already installed on the targeted machines, you should contact IBM Unica Consulting Services before continuing. Because there are many environments and versions, each upgrade should be handled on a case-by-case basis. See Appendix A, "Upgrading IBM Unica Detect," on page 65 for information about upgrading.

Task: Verify installation requirements

Before beginning the installation process, verify that your environment meets the requirements described in Chapter 3, "Preparing to Install IBM Unica Detect," on page 7. Also, for a detailed set of operating system, web application server, and database requirements, see the *IBM Unica Detect Recommended Software Environments and Minimum System Requirements* document for this version of the software.

Note: Detect supports Secure Sockets Layer (SSL) certificate authentication. The steps for configuring it on your system are beyond the scope of this document. If you need help, contact Unica Technical Support.

Task: Configure role services and the application pool (64-bit machines only)

Perform these steps on all 64-bit machines where you plan to install Detect components. This step is necessary only on 64-bit machines, not on 32-bit machines.

1. Log into the machine as a user with administrative privileges on the machine.
2. Open the Windows Server Manager, right-click **Roles > WebServer**, and select **Add Role Services**
3. In the Select Role Services window, check the checkboxes for the following services.
   - ASP.NET
   - ASP
   - ISAPI Extensions
   - ISAPI Filters
   - IIS 6 Metabase Compatibility
4. Close the Windows Server Manager.
5. Open the Internet Information Services (IIS) Manager and select your application pool.

   If you have not created your own application pool, the default is **Application Pools**.
6. Click **Set Application Pool Defaults**.
7. In the **General** section, set **Enable 32-bit Applications** to **True**.

Task: Create a Detect system user

Before you install the Detect software, you must designate a network or local Windows user for Detect. This Detect system user must have administrative privileges on the machine to enable proper communications among the Detect components. Although you can use any existing user with administrative privileges, you should create a user for Detect to make system setup and maintenance a little clearer.

Use this section to create a Detect system user and add that user to the Administrators group.

1. Log in to the machine as a user with administrative privileges on the machine.
2. Right-click the **My Computer** system icon on the desktop and select **Manage** from the pop-up menu to display the Computer Management window.
3. Expand **Local Users and Groups**, then click the **Users** directory to display the list of users.
4. Click **Action > New User** to display the New User window.
5. Create the new user using these steps.
   a. In the **User name** text box, type the name for the user.

      You can use any name you want; the examples in this guide refer to this user as the Detect system user. Be sure to record the name of the Detect system user; you will need this name during the installation process and when you run the system.

Note: If you are running Detect components on more than one machine, use the same name and permission for the Detect system user for each machine.

b. In the **Password** text box enter a password and enter the same password in the **Confirm Password** text box.

   Make a note of the password, as you will log in as this user later in the installation process.

c. Clear the checkbox **User must change password at next logon**.

d. Select the checkbox **Password never expires**.

   Note: If you require that passwords expire, you must periodically manually change the Detect system user password as well as change that password in each of the DCOM components described later in this chapter.

e. Click **Create** to create the user.

f. Click **Close** to close the New User window and return to the Computer Management window.

6. In the Computer Management window, expand **Groups** to display the list of groups, and then double-click the Administrators group to open the group's Properties window, which displays a list of members..
7. If your Detect system user is not listed as a member, add this user as follows.
   a. Click **Add** to open the Select Users, Computers, or Groups window.
   b. Click **Locations** and select the computer you are on (at the top of the list).
   c. Enter the name of your Detect system user and click **OK**.

      Your user is added to the Administrators group.

Copying installation files (DVD only)

If you received your IBM Unica installation files on a DVD, or if you created a DVD from a downloaded ISO image file, you must copy its contents to a writable directory available to the system on which you will be installing the IBM Unica products before running the installers.

You cannot run IBM Unica Marketing installers directly from read-only media, such as the installation DVD, an ISO image mounted read-only, or a write-restricted directory or volume.

Task: Run the Detect installer on the cluster units

Follow the steps in this section to run the Detect installer on the cluster units. The installation wizard allows you to specify the components you want to install.

1. Log in to the machine as a user with administrative privileges on the machine.
2. Run the `Unica_Detect_`*`n.n.n.n_OS`*`.msi` file, where *`n.n.n.n`* is the version number and *`OS`* is the operating system.

   Note: Be sure to run this file from a logical drive (not a network share). Mapped network drives are acceptable, but a drive letter is required.
3. De-select the **IIS Server** checkbox.
4. Complete the fields as described in “Detect installer screen reference” on page 14.

   Note: When you install Detect on a Windows 64-bit machine, and accept the default location, Windows automatically installs it in the `C:\Program Files`

(x86)\Unica folder. If this is the case in your environment, you must edit a file located in the libararymanager\conf directory. Edit either the detect.oracle.config or detect.sqlserver.config file, depending on your database type. Edit the file by changing the value of the db.home.dir property from its default value of c:/Program Files/Unica/Affinium Detect/Application/Database to c:/Program Files (x86)/Unica/Affinium Detect/Application/Database.

Task: Set installation directory security on the cluster units

On **every** cluster unit, you must grant full access to the Affinium Detect directory to the following users. (In fact, you must do this on *every* machine where you install Detect components, as you saw when you performed this procedure in the chapter about installing on the IIS machine.)

- Detect system user—the Windows user you created in an earlier step.
- NETWORK SERVICE (IIS 6.0 only)–A built-in account that has reduced privileges.
- ASPNET (IIS 5.0 only)–A user account created by Microsoft .NET Framework to limit the access rights of .NET applications.
- Administrators group—The group of administrative users on the machine. The user who runs Detect's Outcome Manager Tool must be a member of this group, because the OMT needs the permissions you will grant to this group to be able to do bulk loads to the database.

To set directory security on the cluster units

On cluster units, you must grant the Detect system user full access to the Affinium Detect directory, as described in the following steps.

1. Log in to the machine as a user with administrative privileges on the machine.
2. Right-click the Affinium Detect directory then select **Properties** to display the **Properties** window.

   Do this on the Affinium Detect directory, not the Unica directory which typically contains the Affinium Detect directory.
3. Select the Security tab and click **Add** to display the Select Users or Groups window.
4. Click **Locations**.
   a. In the window that opens, click the name of the local machine for local accounts you want to add, or select the domain the Detect system user account is under if it is not on the local machine.
   b. Click **OK**.
   c. The **Select Users or Groups** window appears again.
5. Click **Advanced**.
6. Click **Find Now** to to display the Select Users or groups window.
7. Select the following users and group.
   - NETWORK SERVICE (IIS 6.0 only)
   - ASPNET (IIS 5.0 only)
   - Detect system user you created in an earlier step
   - Administrators group

   The NETWORK SERVICE (for IIS 6.0) and ASPNET (for IIS 5.0) users and the Administrators group are all under the directory with the local machine name.

If the Detect system user is a local account, it is also under this directory. However, if it is a network account, you can find it under the network path for that user in this Locations dialog box. You can add users from only one location at a time.

a. To select, hold down the **Control** key and click each.

b. Click **OK** to see the object names listed in the window.

8. Click **OK** to add the selected users and group to the list of names on the Security list and return to the Security screen.
9. Set the permission for each of the users and the group that you just added to the Name list:
   a. In the Group or user names section, click the name of one of the users or the group that you just added to the list.
   b. In the Permissions list, next to **Full Control**, select the checkbox under the column **Allow** to give the selected user full security permission. When you select **Full Control**, all the other checkboxes are automatically selected.
   c. Click **Advanced**. Confirm that the checkbox to **Allow inheritable permissions** is selected. If it is not selected, then select it.
   d. Repeat these steps for each user that you added to the Name list.
10. Click **OK** to complete the security settings and close the Properties window.

Task: Set sharing in the installation directory on the cluster units

On every cluster unit, set the sharing on the `Affinium Detect` directory so that full access is granted to the same users or group that you gave security privileges in the previous section

- The Detect system user you created in a previous step
- `NETWORK SERVICE` (IIS 6.0 only)
- `ASPNET` (IIS 5.0 only)
- Administrators (group)

Note: If you are using IIS 5.0, also grant full access to `ASPNET` as shown in this document. It is a user account created by Microsoft .NET Framework to limit the access rights of .NET applications. If you are using IIS 6.0, that limitation is handled by the `NETWORK SERVICE` account.

1. Log in to the machine as a user with administrative privileges on the machine.
2. Locate the `Affinium Detect` directory, right-click it, and select **Properties** to display the directory's Properties window

   Do this on the `Affinium Detect` directory, not the `Unica` directory which typically contains the `Affinium Detect` directory.
3. Select the Sharing tab on the Properties window.
4. Select the option **Share this folder** to enable sharing, and then click **Permissions** to open the Share Permissions window.
5. Select **Everyone** in the list of groups or users, and click **Remove**.
6. Click **Add** to open the Select Users or Groups window.
7. Click **Locations**.
   a. In the window that opens do the following .
      - For local accounts you want to add, click the name of the local machine
      - If the Detect system user account is not on the local machine, navigate to the machine where it exists.
   b. Click **OK**.

The **Select Users or Groups** window appears again.

8. Click **Advanced**.

   The `NETWORK SERVICE (for IIS 6.0)` and `ASPNET` (for IIS 5.0) users are all under the directory with the local machine name.

   The Administrators group is also under this directory with the local machine name.

   The Detect system user is under this directory as well, if it is a local account. If it is a network account, you can find it under the proper network path for that user in this Locations dialog box. You can add users from only one location at a time.

9. Click **Find Now** to display a list of all the local users.
10. Scroll down the list of names and find the name of one of the users or group that you are adding to give sharing privileges.

    Note: If the Detect system user is not on the local machine, select **Locations** and choose the network domain the user is on.

11. Select a user or group and then click on **OK**. Repeat this step for each of the users or the group listed in the intorduction to this procedure.
12. Click **OK** to add the selected users or group to the list on names on the Sharing list and return to the Sharing window.
13. Set the Share permission for each of the users or group that you just added to the Name list.
    a. In the Name pane, click on one of the users or group that you just added to the list.

       In the Permissions list, next to **Full Control**, select the checkbox under the column **Allow** to give the selected user full sharing permission. When you select **Full Control**, all the other checkboxes will be selected automatically.

    b. Repeat these steps for each user or group that you added to the Name list.
    c. Click **OK** to complete the security settings and close the Sharing window.
14. Click **OK** to close the Properties window.

Task: Configure DCOM settings

DCOM enables and coordinates communication between software processes across servers.

About DCOM identity and security in Detect

DCOM security grants Windows users permission to access, launch, and configure components. For Detect, the easiest way to set DCOM security is to grant full control on all Detect components to the same users you specified when you set sharing on the installation directory.

These users are:

- `NETWORK SERVICE`

  This is done on both cluster units and web server units, but applies only when your web server is IIS 6.0.

- `ASPNET`

  This is done on both cluster units and web server units, but applies only when your web server is IIS 5.0.

- `IUSR_machine_name`, where *machine_name* is the name of the server on which IIS is running.

  This is done only on web server units. Note that if you have IIS installed on more than one machine, you will have a different IIS user on each of these machines.
- Administrators (group)

  This is done on both cluster units and web server units. This grants access to the Detect system user because you made that user a member of the Administrator's group in a previous step.

DCOM identity determines the Windows user account(s) that can run a component. Windows provides three options for determining which account runs a component. For Detect, you select **This User**, which allows you to specify the Detect system user as the account whose security context is used to run the component. Because you made the Detect system user a member of the Administrators group in a previous step, this account has the correct privileges to perform all the actions required for Detect.

What to look for in the DCOM Component Services window

When you set DCOM identity and security, you select the Detect components from a list in the Component Services window. The Detect components are listed as follows.

- EMC
- TSM
- EngineController
- FeederWrapper
- ListenerWrapper
- SystemWrapper

See "To set DCOM identity and security" on page 20 for details on performing this procedure.

To set DCOM identity and security

You should set both security and identity on all Detect components as described in this procedure. While it is not strictly required to configure DCOM for all components as described here, this method helps prevent access issues that are difficult to diagnose.

See "Task: Configure DCOM settings" on page 19 for more information about the DCOM settings.

1. Log in to the machine as a user with administrative privileges on the machine.
2. Open a command window and run the `dcomcnfg` command to open the Component Services window.
3. Expand Component Services in the left panel until you get to the **DCOM Config** portion.
4. Right-click each Detect component, select **Properties** to open the Properties window.
5. Set Launch and Activation Permissions **for each Detect component** as follows.
   a. On the **Security** tab, under Launch and Activation Permissions, select the **Customize** option and click **Edit** to open the Launch Permissions window.

b. Add the users described in "Task: Configure DCOM settings" on page 19.
c. Click **OK**.
d. For each of the users, set all of the permissions to **Allow** and then click **OK** to return to the Properties window.

6. Set Access Permissions **for each Detect component** as follows.
   a. On the Security tab, under **Access Permissions**, select **Customize** and then click **Edit**.
   b. For each of the users, repeat the process you used to give them launch permissions in order to give them each access permissions.
   c. Click **OK** to return to the Properties window.
7. Set identity **for each Detect component** as follows.
   a. Select the Identity tab, and choose **This user**.
   b. Click **Browse** to find the Detect system user.
   c. Enter the password for the Detect user in the **Password** and **Confirm Password** boxes, click **Apply** and then click **OK**.

When you have finished with this chapter

When you have finished with this chapter, do one of the following.

- If you are using Oracle, continue to Chapter 7, "Creating the IBM Unica Detect System Tables in Oracle," on page 33
- if you are using SQL Server, see Chapter 8, "Creating the IBM Unica Detect System Tables in SQL Server," on page 39.

After you have set up the database, see Chapter 9, "Creating an Initial IBM Unica Detect User," on page 41 for the final step you must perform before you begin to configure your Detect system.

Chapter 7. Creating the IBM Unica Detect System Tables in Oracle

The IBM Unica Detect installer provides several SQL scripts that you use to create the Detect database schema. They are located on the web server unit, in the `Application\Database\Install\` directory under your Detect installation directory.

Note: The `Application\Database\Install\ Parameterized Scripts` directory under your Detect installation directory on the web server unit contains copies of the original parameterized scripts which are not modified by the installer. You will not typically need to use them, but they are included for reference purposes, or in case you make a mistake. If you make a mistake, you can drop the users, change the scripts, and run them again.

Parameters in the SQL scripts for Oracle

The SQL scripts contain some parameters. All but two of these parameters are set by the the Detect installer.

You must set the `%TBLSP%` and `%IDXSP%` parameters manually, by editing the script files to replace the parameters with the values specific to your environment. See "Task: Configure the Oracle database scripts" on page 34 for instructions on performing the replacements.

The following table provides additional detail about the parameters in the scripts that **you do not have to edit**, as you entered these values when you ran the installer and the installer updated the parameters in these scripts.

Parameter	In Script(s)	Description

`%Vendor%`	`H_Vendor.sql`
	`H_VRGrants.sql`
	`H_VROutcomeSchema.sql`
		This is the three-character value that you specified in when you ran the Detect installer. This value is the same as the name of the three letter subdirectory under your Detect installation directory. The vendor code is used internally by Detect.

`%DatabasePWD%`	`H_VROutcomeSchema.sql` `H_Global.sql`	The un-encrypted password for the Detect database schema users.
`%DatabasePWDEncrypted%`	`H_Vendor.sql`	The encrypted password for the Detect database schema users, used in all connection strings.
`%DetectDatabase%`	`H_Vendor.sql`	The name of the Detect database.
`%DatabaseServer%`	`H_Vendor.sql`	The name of the database server.

`%OutcomeUser%`	`H_VROutcomeSchema.sql`
	`H_VRGrants.sql`
	`H_Vendor.sql`
		The name of the outcome schema user. (Formerly CEE3.)

Parameter	In Script(s)	Description
%HistoryUser%	H_VRGrants.sql H_Vendor.sql H_Global.sql	The name of the history schema user. (Formerly CEE1.)
%RuleUser%	H_VRGrants.sql H_Vendor.sql H_Global.sql	The name of the rule schema user. (Formerly CEE4.)

Task: Configure the Oracle database scripts

You must edit the Oracle SQL scripts to replace the following two parameters with values specific to your environment. You may use a text editor with a global search and replace function to make the changes.

- %TBLSP%

  If you used the default Oracle tablespace, replace this parameter with USERS

  If you created your own tablespace, replace this parameter with the name you gave the main tablespace.
- %IDXSP%

  If you used the default Oracle tablespace, replace this parameter with INDX

  If you created your own tablespace, replace this parameter with the name you gave the index tablespace.

Replace the whole parameter, including percentage signs (%), with your value. That is, your replacement text should not include percentage signs.

1. Locate the following files in the Application\Database\Install directory under your Detect installation directory.
   - H_Global.sql
   - H_Vendor.sql
   - H_VRGrants.sql
   - H_VROutcomeSchema.sql
2. Change the value of %TBLSP% in the following files.
   - H_Global.sql
   - H_Vendor.sql
   - H_VROutcomeSchema.sql
3. Change the value %IDXSP% of in the following files.
   - H_Global.sql
   - H_Vendor.sql
4. If your configuration writes the outcome schema to a separate database or to a different machine, perform the following steps.
   a. In the file H_VRGrants.sql, search for the following text.
      ```
      GRANT SELECT ON %RuleUser%.%Vendor%__DSMDATASOURCES TO %OutcomeUser%
      ```
   b. Insert two dash characters (--) in front of this line and the line after it as shown in the following code:
      ```
      --GRANT SELECT ON %RuleUser%.%Vendor%__DSMDATASOURCES TO %OutcomeUser%
      --GRANT SELECT ON %RuleUser%.%Vendor%__DSMENTITYTYPE TO %OutcomeUser%
      ```

c. Save the file.

Task: Run the database scripts

To complete the set up of the Detect database, run the SQL scripts that you customized in "Task: Configure the Oracle database scripts" on page 34. Be sure to run them in the order listed here.

1. Log in as the admin user to Oracle Enterprise Manager Console on the machine that hosts the Detect system table database. You must be logged in as the admin user to run steps 2. through step 5.
2. Run the edited version of the file `H_Global.sql`. Verify that there are no errors before you continue to the next step. If you receive an error when you run any of the SQL scripts, see the instructions in "To correct a problem with the database scripts."
3. On the same machine, run the edited version of the file `H_Vendor.sql`. Verify that there are no errors before you continue to the next step.
4. Next, run the script `H_VROutcomeSchema.sql` to configure the Outcome Schema. Before you run this script, determine where the Outcome Schema is going to reside in your configuration. There are three possibilities:
   - If the Outcome Schema resides in the same database that you configured in 2. and 3.—run the edited script `H_VROutcomeSchema.sql`.
   - If the Outcome Schema resides on the same machine that you configured in 2. and 3. but is in a different database—go to SQL *Plus and select **File>Change Database Connection**, then sign onto the database that contains the Outcome Schema as an administrator. Run the edited script `H_VROutcomeSchema.sql`.
   - If the Outcome Schema is on a different machine than the main database—go to the machine that hosts the Outcome Schema, log on as the administrator, and run the edited file `H_VROutcomeSchema.sql`.

   Verify that there are no errors before you continue to the next step.
5. On the Detect machine with the rule schema (the same machine you used in steps 2. and 3., run the edited file `H_VRGrants.sql`. Verify that there are no errors before you continue to the next step.

   Note: If you configured the outcome schema to be on a different database, after you complete the installation you must open the Configuration Utility and configure the outcome connection to point to the outcome database.
6. Once you have successfully run all four database scripts, continue to the following section.

To correct a problem with the database scripts

Refer to this section only if you encounter an error when you run the database scripts or if after installation you need to recreate the schema without installing Detect again. This section tells you how to clean out the database to correct errors in the scripts and then rerun them.

If you run a script that has an error, perform the following steps.

1. Delete all the users that were created when you ran the installation. If you accepted the default names during installation, the names of the users are: HistoryUser, OutcomeUser, and RuleUser.

   You can use the SQL Scratchpad to delete a user:

a. From the Oracle Enterprise Manager Console click the name of the Detect database.
b. From the main menu select **Object>SQL Scratchpad** to display the SQL Scratchpad.
c. Type `drop user HistoryUser cascade` into the scratchpad, then click **Execute** to execute the command. Repeat for each user, *OutcomeUser* and *RuleUser*.

   In each case, use the schema name you used during the installation. This example shows the default names.
d. Click **Close** to close the SQL Scratchpad.

2. Delete the two tablespaces that you created for Detect as described in "About creating an Oracle database for Detect" on page 9, and then recreate the tablespaces using the instructions in the same section.
3. You can either locate the backup copy (which was partially edited by the installer) or the parameterized script. In either case, edit the file. Refer to the instructions in "Task: Configure the Oracle database scripts" on page 34.
4. Rerun all the scripts again, using the instructions in "Task: Run the database scripts" on page 35.

Task: Modify the Oracle file sqlnet.ora

Use the following steps to modify the Oracle file `sqlnet.ora`.

1. Open the `qlnet.ora` file in a text editor.

   The default location on the `C` drive is `C:\oracle\ora92\network\admin\sqlnet.ora`
2. Place the character # in front of the following line in the file to comment it out:

   `SQLNET.AUTHENTICATION_SERVICES= (NTS)`

   As an alternative, you could just replace *NTS* with *None*.
3. Save the file, then close the text editor.

Task: Give authenticated users privileges to Oracle home

The following steps provide a work-around for an incompatibility between Oracle and .NET that enables you to give Authenticated Users privileges to Oracle Home on the Oracle 9.2 client software. In these steps you will de-select then re-select a permission.

1. Log onto Windows as a user with Administrator privileges.
2. Launch Windows Explorer from the Start menu, then navigate to the `ORACLE_HOME` directory.
3. Right-click the `ORACLE_HOME` directory and choose the **Properties** option from the drop-down list. A Properties window should appear.
4. In the Properties window, select the **Security** tab.
5. Click **Authenticated Users**.
6. Clear the **Allow** checkbox for the **Read & Execute** permission.
7. Select the **Allow** checkbox for the **Read & Execute** permission that you just cleared.
8. Click **Apply**.
9. Click **OK**.
10. You may have to re-start your computer after making these changes.

Task: Enable MTS for Oracle 10 and 11g

Only if your Detect system table database is Oracle 10 or Oracle 11g, ensure that Oracle Service for Microsoft Transaction Service is enabled.

Chapter 8. Creating the IBM Unica Detect System Tables in SQL Server

The IBM Unica Detect installer provides several SQL scripts that you use to create the Detect database schema. They are located on the web server unit, in the `Application\Database\Install` directory under your Detect installation directory.

Note: The `Application\Database\Install\ Parameterized Scripts` directory under your Detect installation directory on the web server unit contains copies of the original parameterized scripts which are not modified by the installer. You will not typically need to use them, but they are included for reference purposes, or in case you make a mistake. If you make a mistake, you can drop the users, change the scripts, and run them again.

Parameters in the SQL scripts for SQL Server

The SQL scripts contain some parameters. All but one of these parameters are set by the Detect installer.

You must set the `%OutcomeDatabase%` parameter manually, by editing the `H_VROutcomeSchema.sql` and `H_Vendor.sql` script files to replace the parameter with the value specific to your environment. See "Task: Configure the SQL Server database script" on page 40 for instructions on performing the replacement.

The following table provides additional detail about the parameters in the scripts that **you do not have to edit**, as you entered these values when you ran the installer and the installer updated the parameters in these scripts..

Parameters	In Script(s)	Description

`%DetectDatabase%`	`H_Global.sql`
	`H_Vendor.sql`
	`H_VROutcomeSchema.sql`
		Name of the Detect database.

`%Vendor%`	`H_Vendor.sql` `H_VROutcomeSchema.sql`	This is the three-character value that you specified in when you ran the Detect installer. This value is the same as the name of the three letter subdirectory under your Detect installation directory.

`%HistoryUser%`	`H_Login.sql`
	`H_Vendor.sql`
	`H_Global.sql`
		Name of the history schema user. (Formerly known as CEE1.)

`%RuleUser%`	`H_Login.sql`
	`H_Vendor.sql`
	`H_Global.sql`
		Name of the rule schema user. (Formerly known as CEE4.)

Parameters	In Script(s)	Description
%OutcomeUser%	H_VROutcomeSchema.sql H_Login.sql H_Vendor.sql	Name of the outcome schema user. (Formerly known as CEE3.)
%OutcomeDatabase%	H_Vendor.sql H_VROutcomeSchema.sql	Name of the database that will hold the outcome schema. This could be the same database as the one that holds all the other Detect system tables, or it could be a separate database.
%DatabaseServer%	H_Vendor.sql	Name of the Detect database server.
%DatabasePWD%	H_Login	The un-encrypted password for the Detect schema users.
%DatabasePWDEncrypted%	H_Vendor.sql	The encrypted password for the Detect schema users.

Task: Configure the SQL Server database script

You must edit the SQL Server SQL scripts H_VROutcomeSchema.sql and H_Vendor.sql to replace the %OutcomeDatabase% parameter with a value specific to your environment.

The file is located in the \Application\Database\Install directory under your Detect installation directory.

You may use a text editor with a global search and replace function to make the changes.

Replace the whole parameter, including percentage signs (%), with your value. That is, your replacement text should not include percentage signs.

Task: Execute the scripts

1. Locate the following scripts:
   - H_Login.sql
   - H_Global.sql
   - H_Vendor.sql
   - H_VROutcomeSchema.sql
2. Execute the scripts in the order in which they are listed in step 1.
3. Only if you plan to use the Outcome Management Tool (OMT), you must give bulkadmin permission to the outcome table user. Do this using your database management tool, on the Login Properties screen for this user.

Chapter 9. Creating an Initial IBM Unica Detect User

When you have finished the IBM Unica Detectinstallation process, you must perform additional setup using the Detect user interface and Configuration Utility, as described in the *IBM Unica Detect Administrator's Guide*. Before moving on to perform those steps, you must create a Detect user with appropriate permissions, and verify your installation.

Detect provides a pre-defined login account that has the **host** role, which enables you to create users and assign security roles in the Detect web application, and set password policies in the Detect Configuration Utility. It does not allow access to any other features, and you cannot modify or delete this role, so you must create a user with wider permissions in order to complete your Detect setup. See "To add an initial Detect user" for details.

To add an initial Detect user

1. Log in to the Detect user interface, using the following information.
   - URL—`http://`*`machine_name`*`/Detect`, where *`machine_name`* is the name of the machine where the web server unit is deployed.
   - User name—system
   - Password—host

   The Detect Users page appears.
2. Click the New user icon.

   The New User page appears.
3. Complete the fields and click **Save changes**.

   Passwords are case-sensitive, but user names are not.

   The new user is created. By default, all new users have the **Default** role, which allows view-only access to all functions in the user interface, except for user administration.
4. Select the user you just created to display that user's detail page and click **Assign Roles**.

   A page appears showing the currently assigned and available roles.
5. Select the **Power User** role in the Available Roles box and click **Add** to assign that role to your user.

   The Power User role is listed in the Assigned Roles box.
6. Click **Save Changes**.
7. Log out of Detect and log in again as your new user.

   You should see all of the Detect menu items except **Users** in the menu bar at the top of the page. Note that, if you have upgraded, all existing users should have the **Power User** role by default.
8. Verify your installation by navigating to each of the functional areas.
9. To ensure maximum security, you should do the following.
   - Create a new role with `User Administration` and `Password Policy` permissions. This grants the same access as the `system` account's `host` role.
   - Create a new user and assign this new role to the user.
   - Change the password of the `system` account.

If all user administration accounts become disabled (for example, due to multiple failed login attempts), you must contact Unica Technical Support to restore this access to Detect.

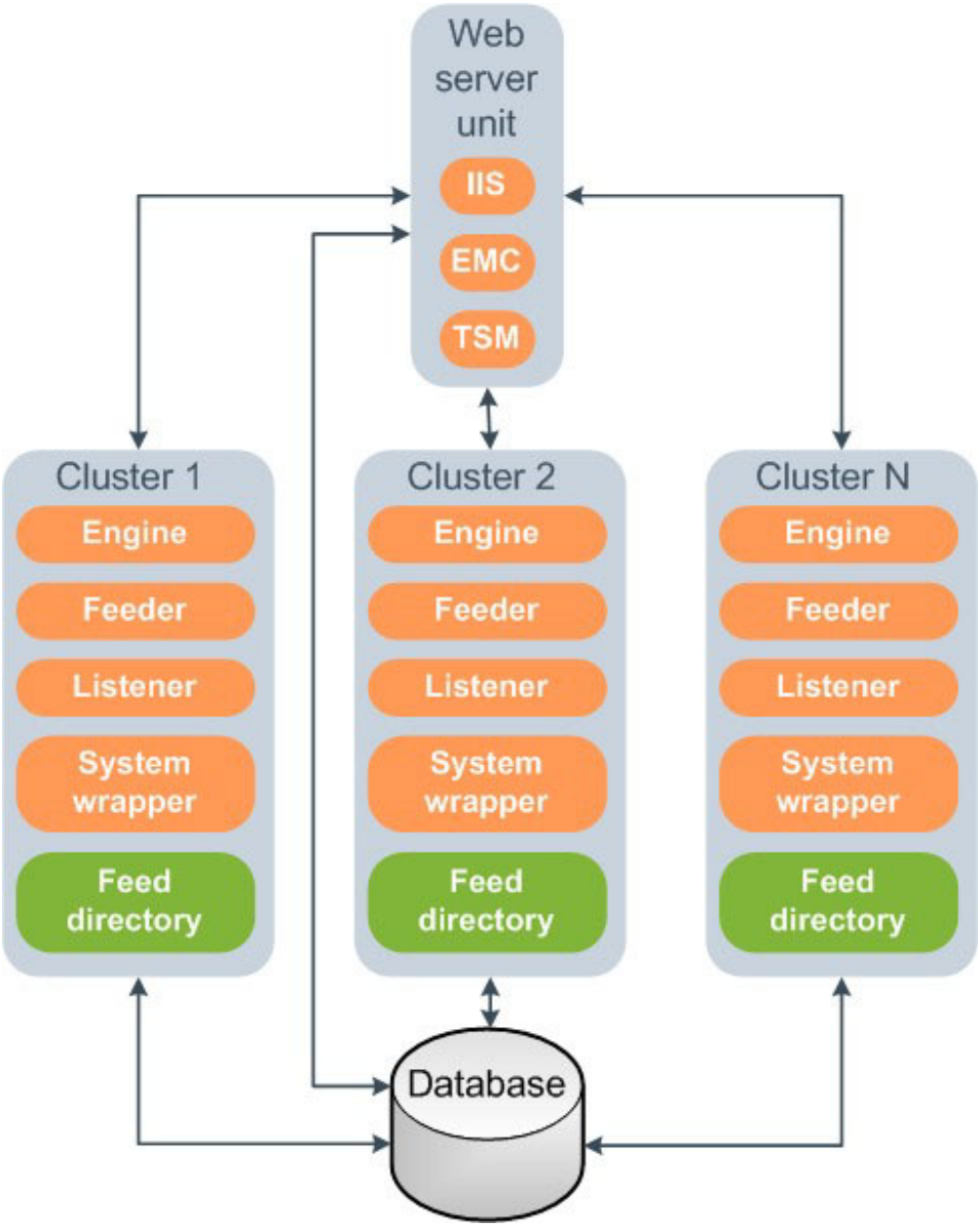
To set up multiple environments or clusters, see the releavant chapters in this guide. For information about further required configuration, see the *IBM Unica Detect Administrator's Guide* .

Chapter 10. Configuring Multiple Cluster Units

IBM Unica Detect can be installed on multiple machines, called cluster units, to improve performance. Detect cluster units have the following characteristics.

- Each cluster unit must have its own feeder and outcome listener.
- Each cluster unit must at least one engine component, and may have multiple engines.
- Each cluster server machine can have one or more clusters defined within it.
- Clusters share a central monitoring process (TSM), a single IIS instance, and a single database.

The Configuration Utility enables you to choose a hash algorithm to enable the feeders in each cluster unit to access the same feed files from a central location, or to partition the feed files by range, so that the feed files are partitioned and distributed on each of the clusters. The following diagram shows the clustered configuration.



Note: For simplicity, this document describes one cluster per cluster server machine.

About configuring the clustered environment

Configure each machine in your Detect setup using the tasks in this section. Perform these tasks using the Configuration Utility, which is a separate utility from Detect.

- "Task: Log into the Configuration Utility" on page 45
- "Task: Identify each server in the system" on page 45
- "Task: Set general configuration settings" on page 45
- "Task: Set internal system connections" on page 46
- "Task: Optionally, set external user connections" on page 47

- "Task: Adjust system preferences using Advanced Configuration settings" on page 48
- "Task: Define entity types" on page 56
- "Task: Optionally, add outcome destination tables" on page 56
- "Task: Define each cluster" on page 56
- ""Task: Enable the clusters" on page 57

Task: Log into the Configuration Utility

Perform this task using the Configuration Utility, using an administrator user login. The Configuration Utility is a separate utility that is apart from the Detect web interface.

Note: The options available in the Configuration Utility are different when an admin login is used than they are for the host_admin user. Admin users can manage system connections, advanced configurations, and clustering.

1. Locate `ConfigurationUtility.exe` in the `Application\bin` directory under your Detect installation.
2. Run the executable.
3. Log into the Configuration Utility as a system administrator user.
4. Expand **Detect** to see **Configuration**, **Entity Types**, **Clusters**, and **Servers**.

Task: Identify each server in the system

In this task, identify each server by giving it a name. Provide the names of all the machines that are used in the Detect installation. Of these machines, exactly one server will hold the TSM and the IIS. Other servers will each hold a feeder, outcome listener, and one or more engines.

Note: Each cluster unit can have one or more clusters defined within it. Each of the clusters in the cluster unit must have its own feeder and listener and can have multiple engines. For simplicity, this chapter describes one cluster per cluster unit.

1. In the Configuration Utility, right click **Servers** and select **New** to open the Server dialog, where you define a new server.
2. Enter the name of an actual server that will be used in the Detect configuration.

   The ID field is automatically populated.

   Specify only the machine name; do not include the domain.
3. Click **OK**.

   If you come back to this list of servers after defining the IIS, TSM, and cluster definitions for the servers, you can click the name of the server on the left, the display area on the right will identify which components are mapped to that server.
4. Repeat these steps to add each machine running Detect.

Task: Set general configuration settings

In this task you identify which of the servers should be the IIS and TSM server. IIS and TSM can be on separate machines, but typically they both reside on the same machine. In this task you also choose the partitioning method (if any).

1. Start the Configuration Utility, located in the `Application\bin` directory under your Detect installation.

2. Log in using an account with full access to the Configuration utility.

   By default, the built-in **system** account does not have this access. You may need to use or create an account with the Power User role. See the "Managing Users and Security" chapter in the *IBM Unica Detect Administrator's Guide* for details.
3. Double click **Configuration** to access the Configuration popup.
4. On the General tab, choose a partitioning method for the entities.
   - You can choose **None** if there is only one cluster or if there is just one set of feeds.
   - If you select **Range**, the system will use the range set for the cluster.
   - If you choose **Hash**, then also select the Hash Algorithm. In tests, PG, RS, and SDBM tended to work the best, but another method may be most efficient on your system. Hashing is based on entity ID. (You may want to use the System Run Time Distribution Report to help analyze the system distribution.)
5. You may also want to set the **Feed File Collation** property on this tab.

   For details, see "About the Feed File Collation property."
6. Click **OK**.

About the Feed File Collation property

The **Feed File Collation** property controls the sort order in your feed files. It is located on the General tab in the Configuration Utility's pop-up window.

If your feed files have a different sort order from the collation set in your Detect database, you should set this property to reflect the expected sort order in your feed files.

Note that, for Event Enabled Time Queue (EETQ) profile feed files, this setting is applied only when you set the **Inactivity Feed Path** property in the Configuration Utility. This property is located in the **Processing Options** section on the Advanced Configuration tab in the Configuration pop-up window.

Also note that you must now save EETQ files in UTF-8 format (previously they could be in ASNI format). Ensure that the program you use to edit and save these files is capable of saving in UTF-8 format.

Task: Set internal system connections

Use this task to adjust the internal system database connections.

1. In the Configuration Utility, double click **Configuration** to access the Configuration popup.
2. On the System Connections tab, view the list of Detect system connections.

   The system connections are the standard internal database connections to the database schemas that were created during the Detect installation. There is one connection for each of the schemas. Each connection is named for the schema to which it corresponds. The list identifies the name of each schema connection, the database provider, and the name of the database on which the schema resides.

   Note: The schema names and connection string names are created during installation. You cannot change them.

   The connections are:

- **outcome** — This is the connection to the outcome schema. The outcome schema contains the outcomes from the rules that are triggered during a run.
- **rule** — This is the connection to the rule schema. The rule schema contains the metadata for the rule engine, including rule definitions, machine states, and logging information. It contains all the global information used by the system.
- **visitor** — This is the connection to the history schema. The history schema contains the contains visitor history information about customers that is written to the database when rules fire.

3. To view or adjust the details of a connection, double click the name or highlight it and click **Properties**.

   If you want to put your own schema into the Detect database, you can use these connection strings to access it. You may want to change the password, but typically you do not need to change anything.

   The details on this page define the connection string.

   - **ID**—The connection ID is an automatically generated ID for each connection.
   - **Name**—The name of the connection. For the three Detect system connections, the name is automatically created; it matches the name of the schema created during installation. When the name is automatically created, it cannot be changed.
   - **Description**—The description of the description.
   - **Database**—The name of the database on which the schema resides.
   - **Provider**—The database provider.
   - **Server**—The name of the server on which the database resides.
   - **User Name**—The database user name for the schema. The user name is set during installation. The default names used by the installation wizard are: **OutcomeUser** for the outcome schema (formerly CEE3), **HistoryUser** for the visitor history schema (formerly CEE1), and **RuleUser** for the rule schema (formerly CEE4).
   - **Password**—During installation the schemas are all given the same password, but you may choose to use a different password for each schema. If you change a password, the tool automatically encrypts it.

     Note: If you change the password for the rule user database user, you must also update the registry with the same encrypted password. You can use the Password Encryption Tool to encrypt a password.

     The visitor history and rule schemas are located together in a single database. The outcome schema may or may not be in the same database.

4. Click **OK**.

Task: Optionally, set external user connections

Use this task to adjust the external, or user database connections. User connections, if any, are connections that allow Detect to access a lookup table or database profile in your own database (rather than within the Detect schema). You can add a connection, or view and edit the details of an existing one.

1. In the Configuration Utility, double click **Configuration** to access the Configuration popup.
2. On the User Connections tab, view the connections.
3. To view or adjust the details of a connection, double click the name or highlight it and click **Properties**.

The fields are the same fields as on the General tab for System Connections, except that you can change the name of a user connection.

4. To add a connection, click **Add** then enter information in the fields.
   - **ID**—The connection ID is an automatically generated ID for each connection.
   - **Name**—Enter a name for the connection. The name will be used internally by Detect.
   - **Description**—Enter a descriptive name for the connection.
   - **Provider**—Select the provider of the database from the drop-down list.
   - **Server**—The name of the physical database server. This information is required only when the provider is set to SQLServer.
   - **Database Name**—The name that was given to the database service.
   - **User Name**—The user name that is used to log onto the database.
   - **Password** and **Re-enter Password** —The password associated with the supplied User Name. The password must be re-entered to detect mistakes.
5. Click **OK**.

Task: Adjust system preferences using Advanced Configuration settings

1. In the Configuration Utility, double click **Configuration** to access the Configuration popup.
2. In the Configuration Utility, select the **Advanced Configuration** tab.

   The default preferences page is broken into sections that contain the various types of preferences.

   Note: You can enlarge the description pane below the list of settings by dragging its top border up to make it taller.
3. View or set the Advanced Configuration settings.
4. Click **Apply** to save the settings.

Advanced configuration settings in the Configuration Utility

The following tables describe the configuration properties on the Advanced Configuration tab of the Configuration Utility in Detect.

Note: Items marked with an asterisk (*) do not revert to default settings if you click the **Restore Defaults** button.

Archive Service

These settings control the level of archiving.

Setting	Description	Default
Archive Feed Files	The archive process copies the feed files to the archive folder if the value is set to **YES**. The system copies the feed files from the feed folder (per cluster) to the folder: *Archive Server**Archive Share**runID**timestamp*\feeds*Cluster Name*\	NO
Archive Run Logs	If this value is set to **YES**, the archive service copies all run logs to the archive folder, organized by server name. The system copies all logs to the folder:*Archive Server**Archive Share**runID**timestamp*\logs	NO

Setting	Description	Default
Archive Server	The name of the server on which to store the archive logs, feeds, and outcome.	server name
Archive Shared Folder	The shared root folder on the archive server in which to archive the files. The folder must be a shared folder, with read and write permissions.	C$ *
Days for Archived Logs	The amount of time, in days, to keep run history, logs, and archive files. Data older than the number of days set is automatically purged from the following tables: • G_Run • G_Runsystemstats tables • G_Enginebookmarks • G_Errors • G_Subsystemstatus • G_Subsystemperf • G_Logs • G_Runstatus • G_Rulefiringcounts **Note:** The system purges archived files based on this setting. Archiving and purging date calculations are done based on the current machine's date, not on feed dates.	365
Outcome Extraction	Whether or not to extract outcome from the outcome table. The drop-down list has these choices: • **All**—Archive the entire outcome table, including every record and all the fields • **Counts**—Archive the firing count, grouped by Action ruleID • **None**—Do not archive any data in the outcome table.	None

Database

View or change the database configurations.

Setting	Description	Default
Database Type	The ID of the database type from the G_Databases table, which indicates 1 for Oracle and 2 for SQL Server.	(Controlled by installation scripts) *
Outcome Database User Name	Used by SQL server to connect to the outcome table. This is the one schema that can reside outside the Detect database.	(Controlled by installation scripts) *
Outcome Database Vendor	The ID of the outcome database type from the G_Databases table, which indicates 1 for Oracle and 2 for SQL Server.	(Controlled by installation scripts) *

Error Tolerance

These settings define the maximum number of errors that can occur for a run to be considered successful.

Setting	Description	Default
Engine Error Tolerance	The maximum number of errors that the engine can encounter during processing before the system stops and shuts down.	1
Feeder Error Tolerance	The maximum number of errors that the feeder can encounter during processing before the system stops and shuts down.	1
Listener Error Tolerance	The maximum number of errors that the listener can receive before the process stops and shuts down.	1

Locale

View or set the locale.

Setting	Description	Default
Locale Identifier	Sets the date format for the following. • Reports • Inactivity target date of Engine run • Constants in Simple or Qualifier editors • Outcome messages • Outcome Management Tool parsing of dates from outcome tables • Timestamps in Rampup feeds Use the drop-down list to see and select the options.	ISO 8601 Date Format *

Logging or Reporting

Setting	Description	Default
AlwaysOn Log Listener Port Number	This port number is used by the log server and log clients to communicate log messages when the engine runs in always-on mode. It must be an available port on the server that has the TSM.	4935
AlwaysOn Performance Listener Port Number	This port number is used by the log server and log clients to communicate log messages when the engine runs in always-on mode. It must be an available port on the server that has the TSM.	4945
Error Logs Persistent Method	This setting controls where to write the error logs. May be set to Database or File. By default, the error logs are written to the database to be viewable within the engine manager.	Database
Log Listener Port Number	This port number is used by the log server and log clients to communicate log messages when the engine runs in batch mode. It must be an available port on the server that has the TSM.	4915
Performance Listener Port Number	This port number is used by the log server and log clients to communicate log messages when the engine runs in batch mode. It must be an available port on the server that has the TSM.	4925
Save Firing Counts to the Database	This is an optional setting that turns on or off the gathering of component firing information to be used in reports	On

Setting	Description	Default
System Log Persistent Method	This setting controls where to write the system run time log information. May be set to Database or File.	File
TSM Log Rotation Interval	The number of messages the TSM will log before rotating the file. A value of zero results in no rotation. If you run in Always-on mode, you should set this value to 1000 or greater..	File

Notifications

These settings define the names of files Detect attempts to run after various processes complete. You can change the names of the specified file here, but if you do, you must supply a file of that name. These settings are typically used when a separate system is set up to automatically run other batch jobs or reports.

Setting	Description	Default
Alert File Name	This file is executed by the EMC N seconds before the end of any always-on run, if the **Alert Before Processing Completion** option in the **System Options** section is set to a value greater than 0.	alert.bat
AlwaysOn Initialization Failed	This file is executed by the EMC if an attempt to initialize an always-on run mode fails.	initFailedAO.bat
AlwaysOn Initialization Success	This file is executed by the EMC if an attempt to initialize an always-on run mode succeeds.	initSuccessAO.bat
AlwaysOn Shutdown Failed	This file is executed by the EMC if an attempt to shut down an always-on run mode fails.	shutdownFailedAO.bat
AlwaysOn Shutdown Success	This file is executed by the EMC if an attempt to shut down an always-on run mode succeeds.	shutdownSuccessAO.bat
AlwaysOn Transaction Failed	This file is executed by the EMC if a run in always-on mode fails.	runFailedAO.bat
AlwaysOn Transaction Success	This file is executed by the EMC if a run in always-on mode succeeds.	runSuccessAO.bat
AlwaysOn Transaction Success with Errors	This file is executed by the EMC if a run in always-on mode succeeds with errors. The errors in this case would not be enough to shut down the system. (See related settings in this table, listed under Error Tolerances.)	runSuccessWith ErrorsAO.bat
OMT Failed	This file is executed by the EMC if the OMT Populate Outcome operation fails.	omtError.bat
OMT Success	This file is executed by the EMC when the OMT Populate Outcome operation finishes successfully. You can use this file to automate down stream processing of the application outcome.	omtDone.bat
PreProcessor Failed	This file is executed by the Preprocessor if it fails.	PreProcessorError.bat
PreProcessor Success	This file is executed by the Preprocessor when it runs successfully. You can use this file to automate transaction processing.	PreProcessorDone.bat
Ramp Up Failed	This file is executed by the EMC if the Ramp Up process fails.	rampupError.bat
Ramp Up Success	This file is executed by the EMC when the Ramp Up process finishes successfully.	rampupDone.bat

Setting	Description	Default
Transaction Failed	This file is executed by the EMC if a run in batch mode fails.	ftpError.bat
Transaction Success	This file is executed by the EMC when the system finishes a run in batch mode successfully. You can use this file to automate down stream processing of the application outcome.	ftpPut.bat
Transaction Success with Errors	This file is executed by the EMC if the system finishes a run in batch mode successfully with errors. The errors in this case would not be enough to shut down the system. (See related settings in this table, listed under Error Tolerances.)	ftpPutWithErrors

OMT Options

Setting	Description	Default
Float Field Precision	Sets the numeric precision and scale for floating numbers used by the Outcome Management Tool. Applies to systems using the Oracle database only.	(15,5)
Integer Field Precision	Sets the numeric precision and scale for integer numbers used by the Outcome Management Tool. Applies to systems using the Oracle database only.	(10,0)

Processing Options

Setting	Description	Default
Container Aging	How to calculate the cutoff time for container aging. O: Exclusive, 1: Inclusive	Exclusive *
Feeder Data Validation	If true, the feeder validates the data being processed against the associated data type defined within the data source. If the data type does not match the data, then the feeder generates an error and quit.	False
Feeder Throttle	This setting turns on and off the Feeder Throttle. The options are On and Off. **Note:** When Feeder Throttle is set to On, ensure that the Feeder Pause Interval parameter is set to 0 (zero, for off). The Feeder Pause Interval parameter is a less effective mechanism for controlling the number of messages in the queue. The Feeder Throttle is a monitoring capability for the Feeder. It prevents messages from backing up in the engine queues and exhausting the MSMQ service resources. The Feeder Throttle controls the number of messages on the MSMQ Service during a batch run. If there are more than a certain number of messages, the Feeder sleeps. If the engines are fast enough and keep up with the feeder, the Feeder Throttle ensures that the Feeder never sleeps (slowdown).	On
Float Operation Accuracy	This setting is used by the back engine when dealing with floating numbers and significant digits. It is used to determine the tolerance level by the engine when dealing with rounding issues.	12 *

Setting	Description	Default
Inactivity Feed Path	Inactivity Feed Path holds the location of a shared folder where Detect can write a file used during inactivity processing. With this option enabled, a single query gathers inactivity data, rather than queries from every Feeder in a cluster. If you set a value for this option, you can also choose to have them automatically deleted by setting the Delete Inactivity Files option in the Archive service section. Set the value to the path to a shared folder. For example: `\\servername\sharename` If you do not set the value of the the Inactivity Feed Path configuration option, multiple feeders can perform inactivity queries. However, synchronization across clusters still occurs; no engine processes a transaction until all Feeder queries for inactivity events are finished processing.	
Inactivity Mode	This setting controls how the engine processes inactivity events. **Disabled Firing**–All arming FLI events are processed and all firing events are suppressed by the engine. **CRM**–Process all transactions prior to processing inactivity events. **RealTime1**–Process inactivity events between transactions. It tells the system that before processing the transaction, check to see if any inactivity events are due to fire before the transaction date. If there are multiple transactions, check for inactivity events before each transaction. In this mode, the feeder checks for inactivity events prior to processing transactions. If processing files for multiple days at once, it checks for inactivity events between processing each day's transaction feeds **RealTime2**–The engine behavior is the same as for the RealTime1 option. However, with this option the feeder does one final query based on the inactivity end date. The inactivity end date is a parameter set in the command line or in the Engine Manager. In this mode the following occur: • The feeder checks for inactivity events prior to processing transactions. If processing files for multiple days at once, it checks for inactivity events between processing each days transaction feeds. • The system does one final inactivity processing pass, based on the target date (inactivity end date). • This mode (and the Process Between Transactions mode) are the most accurate and thorough. **Off**–No inactivity events are processed.	RealTime2 *

Setting	Description	Default
Max Messages To Send Without Monitoring	This setting determines the maximum number of messages to send before monitoring. This number correlates to the X in the algorithm. So, for example, when the value is set to 60,000, the total number of messages in all the engine queues should not be more than 90,000. **Note:** If this parameter is set to a very high value, the feeder never checks the queue. Setting the parameter to a high value is one way to bypass the mechanism on a daily run. If this parameter is set too low, the checks are too frequent and may slow the feeder to the point that the engine is waiting, thus starving the engine.	60,000
Sleep Time Maxed	This setting controls the Feeder's sleep time when the message count is equal to 1.5 times the setting for **Max Messages To Send Without Monitoring**. (This sleep time is rarely invoked.)	500 milliseconds
Sleep Time Not Maxed	This setting controls the Feeder's sleep time when the message count is below 1.5 times the setting for **Max Messages To Send Without Monitoring**. If this value is set too high there is a chance of starving the engine.	30 milliseconds

System Options

Setting	Description	Default
Alert Before Processing Completion	Amount of time, in seconds, before the engine completes an always-on run, when Detect attempts to execute the batch file specified by the **Alert File Name** option. If this value is set to `0`, Detect does not attempt to execute the batch file.	300
Bookmark Frequency	The engine writes a bookmark record to the G_Enginebookmark table after N number of records	1000
Command Timeout	Sets the command timeout when querying the database in critical areas.	300
Default input file encoding	Replaces the former Default Feed Encoding property used in previous releases. Sets the encoding for the following. • State snapshot reports • The input file for the Presentation Layer Manager • Ramp-up feed files • `DebugIds.txt` file for the engine • `SuppressedRules.txt` for the engine Use the drop-down list to see and select the options. All log and other files generated by Detect are saved in UTF 8 encoding, regardless of any configuration settings.	
Delete Inactivity Files	Specifies whether the feeder deletes old inactivity files. Deletion takes place if the value is set to YES.	NO
Estimate Processing Time	Specifies whether Detect calculates and monitors the time remaining until the run completes.	YES
Excessive Transaction Threshold	The Feeder logs users that have transaction counts that exceed this threshold.	2000

Setting	Description	Default
Feed Delimiter	The field delimiter used by the feeder to parse the data files	|
Feeder Pause Interval	This setting is used to slow down the feeder from processing too quickly. A value of 0 means the feeder does not sleep.	0 *
History Buffer Size	Size of the entity state buffer in bytes. Maximum value is 10000. This setting has a direct effect on performance.The larger the buffer, the more it affects performance.	11263 *
IIS Instance determined by	You need to change this setting only when you set up multiple environments and you want to use a different instance of IIS for each environment. If that is the case, select **Environment.** If you set up multiple environments and want to use one instance of IIS for all of the environments, leave the default value of **System**.	System
Inactivity Query Timeout	Sets the number of seconds that Detect waits before terminating an inactivity query.	300
Progress Recalculation Interval	The interval, in seconds, between calculations that Detect calculates the progress of a run and remaining time until completion for a run.	120
Simultaneous runs	Enables simultaneous batch runs. The following options are available. • **Off**–Only one run is allowed regardless of the number of environments. • **Contention Support**– Multiple runs are allowed, including multiple runs on the same workspace. The system protects itself from potential deadlocks in the state table. • **No Contention Support**–Multiple runs are allowed, but not on the same workspace.	Off
Time to Initialize	The maximum time allowed, in seconds, for the system to initialize.	300
Time to Shutdown	The maximum time, in seconds, allowed to shut down the system.	300
Time to Stop	The maximum time, in seconds, allowed to stop the system.	300
Transaction Packet Size	The feeder packages messages after N number of messages for a user. If the transactions for a user exceed the set value, then the transactions are split into separate packages to be sent to the engine for processing. You probably never need to adjust this setting.	2000
TSM Timeout	The time, in seconds, that the EMC waits for a response from the TSM. If this time period has expired then the EMC shuts down the system.	300

Task: Define entity types

Before you can define a cluster, entity types must be defined within Detect. You can define them within the Configuration Utility, or within the Data Source Editor. This section describes how to create them in the Configuration Utility.

1. In the Configuration Utility, right click **Entity Types** and select **New**.
2. Give the new entity type a descriptive Name.
3. Give the entity type a **Code**. The code must be a single, lower-case character that represents the entity type (for example, `c` for customer).

   The ID automatically fills.
4. Click **OK**.

   The list of entity types updates show your most recent addition.

Task: Optionally, add outcome destination tables

By default Detect sends outcome to the Outcome table. You can optionally add other outcome destinations, and then use the Action editor to select to which table the outcome of each trigger should be sent. Use this task to add outcome destinations.

1. In the Configuration Utility, expand **Configuration** so that you can see **Outcome Destinations**.
2. Right click on **Outcome Destinations** and click **New**.
3. Give the new outcome destination a descriptive name.
4. Click **OK**.

When you run the system, the system automatically generates the table.

Task: Define each cluster

In this task, you configure each cluster, including giving it a name and declaring its feed directory.

Important: Before you can define the cluster, entity types must be defined within Detect. You can create entity types within the Data Source Editor. You can also create them within the Configuration Utility.

Note: Each cluster server can have one or more clusters defined within it. Each of the clusters in the cluster unit must have its own feeder and listener and can have multiple engines. For simplicity, this chapter describes one cluster per cluster unit.

1. In the Configuration Utility, right click **Clusters** and select **New** to get the Processing Cluster form, where you define the cluster.
2. On the General tab, give the cluster a working name and description and check **Enabled** if you want the cluster unit to be active. (The Enabled checkbox allows you to enable or disable clusters as needed.)

   The name can be any unique alphanumeric string. The **Enabled** checkbox allows you to enable or disable clusters as needed.
3. Set the system settings on the System tab.
   - Select the server you are setting up as a cluster unit. The list shows the servers that have been identified for the system.
   - Enter the number of engines that the cluster will run.

Unica Services can help you determine the number of engines you need. The optimal number depends upon many factors including your environment, how fast each machine runs, and the network configuration. It is expected that the number of engines will be adjusted during testing of the system. (You do not need to set the number of feeders or listeners, because there is always one of each on a cluster unit.)

- Browse to enter the path to the feed folder directory.
- Select the **Entities** you want associated. If there is only one entity type name listed, there is only one choice. The entities are loaded from the database and can only be set via the user interface.

  If there is more than one entity, set the cluster so that each cluster runs one entity and distribute the entities across the clusters. Unica Services can advise you on this setting.

4. If the Entity Partitioning method is set as Range (on the General tab in Configuration), specify the lower and upper bound of the entity range. The cluster unit will use these bounds to restrict the entities that it processes. Each cluster unit must have its own range.
5. Click **Apply**.
6. Repeat 1 through 5 to define each cluster unit.

Task: Enable the clusters

Use this task if you are ready to enable the clusters. You must enable the clusters before you can run the engine.

1. Within the Configuration Utility, right click **Clusters** and select **New** to get the Processing Cluster form, where you defined the cluster.
2. On the General tab, check **Enabled** if you want the cluster unit to be active. (The Enabled checkbox allows you to enable or disable clusters as needed.)
3. Click **Apply**.
4. Repeat 1 through 3 to enable each cluster unit you want to enable.

Chapter 11. Configuring Multiple Environments

You can configure multiple IBM Unica Detect installations to run simultaneously while accessing the same entity state data. To do this, you define multiple Detect environments. Environments allow multiple runs to process transactions simultaneously on different workspaces or even on the same workspaces.

Multiple Detect environments have the following characteristics.

- Each environment runs against the same database, sharing users, data sources, workspaces, and so on.
- Each environment allows the full usage of every Detect component, with limitations.

Only if you are using SQL Server for the Detect system tables, multiple environments have the following additional characteristics.

- Each environment allows the same workspace (pub or other) to run in parallel with other environments.
- When running the same workspace, each environment accesses the same state history table.

You can use Oracle for the Detect system tables when you configure Detect to use multiple environments, but with Oracle, two environments cannot run the same workspace.

About defining environments

You define environments using the Configuration Utility. Each environment definition includes one or more IIS servers, a TSM server, and one or more clusters.

You can configure IIS in multiple environments in either of two ways.

- You can use a separate instance of IIS to run and manage each environment. See "To configure an environment to run with its own instance of IIS" on page 60.
- You can use a single instance of IIS to run and manage multiple environments. See "Configuring multiple environments to run with a single instance of IIS" on page 61.

If you are not using multiple environments, Detect automatically configures a single environment for you; no special configuration is required.

New environment parameter in EDMDriver, ESO, Outcome Management Tool, and RampUp

The EDMDriver, Entity State Optimizer (ESO), Outcome Management Tool, and RampUp processes are initiated on the command line. All three of them accept the optional `env` parameter. When multiple environments are defined, use this parameter with an environment ID to specify the environment. If the value is invalid or does not reference an existing environment, Detect reports an error and ends the process.

Here are some usage examples.

- `OutcomeManagerTool.exe /run /user:qa /password:qa /vendor:BNK /workspace:pub /FTP:On /env:1`
- `Entity State Optimizer.exe /start /user:qa /password:pqa /workspace:TA1 /delete`
- `edmdriver.exe /start /dir:C:\Feeds /r:pub /v:BNK /entity:a /User:qa /Pswd:qa /query:2010-03-15 /ArtTrans:1 /ftp:OFF /log:DEBUG /eetqmode:RealTime1 /env:1`
- `Rampup.exe /workspace:pub /ven:BNK /login:qa /password:qa /behavior:append /ftp:OFF /env:1`

Limitations

Note the following limitations when you run Detect with multiple environments.

- To run different triggers in each environment, you must reference copies of the same data source with different names in each environment. The data sources can have the same structure, but renaming them is essential so that triggers in one environment do not activate triggers in the other environment.
- When you run multiple environments simultaneously on the same workspace, there may be inaccuracies in the outcome because each run can inject transactions that occurred later but are processed before transactions in another run. For example, a run 1 could have a transaction for entity Z dated T1. But a shorter run 2 may include Z with a transaction dated T2. Therefore, the transactions may be processed in the order T2 then T1, instead of the proper order of T1 then T2.
- To avoid deadlock or data overwrites, Detect imposes some protocols for interaction when two environments go after the same record, or try to insert the first record for the same entity. Because environments are not likely to collide very often, this collision strategy should not slow performance noticeably.

To configure an environment to run with its own instance of IIS

Perform this procedure in each environment you set up.

1. Install Detect as described in this guide, following these guidelines.
   - Specify the same database information for the Detect system tables each time you run the Detect installer.
   - For each instance of IIS that you plan to use with Detect, perform the procedures described in this guide for the web server unit.
2. Configure Detect and set up the desired Detect clusters as described in this guide.

   Configure all of the instances of Detect to use the same data sources. Reference copies of the same datasource with different names in each environment.
3. Log in to the Detect Configuration Utility and double-click **Configuration** to access the Configuration pop-up.
4. Under **System Options**, locate the **IIS Instance determined by** property, set it to **Environment**, and click **OK**.
5. Right-click **Environments** and select **New** to access the Environment pop-up.
6. Complete the fields in the Environments pop-up as follows, and then click **OK**.

Field	Description
Name	Enter a name for the environment.
Description	Enter a description for the environment.

Field	Description
EMC Server	Select an EMC server for the environment.
TSM Server	Select a TSM server for the environment.
Clusters	Click **Edit** to open the Environment Clusters pop-up, select one or more enabled clusters for the environment, and then click **Save**.

7. Exit the Configuration Utility.

Configuring multiple environments to run with a single instance of IIS

The procedure for configuring multiple environments to run with a single instance of IIS differs depending on whether you are installing on a 32-bit or 64-bit machine.

Perform the appropriate procedure in each environment you set up.

- "To configure multiple environments to run with a single instance of IIS (32-bit machines only)"
- "To configure multiple environments to run with a single instance of IIS (64-bit machines only)" on page 62

To configure multiple environments to run with a single instance of IIS (32-bit machines only)

1. Install Detect as described in this guide, following these guidelines.
   - Specify the same database information for the Detect system tables each time you run the Detect installer.
   - For each instance of IIS that you plan to use with Detect, perform the procedures described in this guide for the web server unit.
2. Configure Detect and set up the desired Detect clusters as described in this guide.

   Configure all of the instances of Detect to use the same data sources. Reference copies of the same datasource with different names in each environment.
3. On each server where IIS is installed, verify that a Detect system user is created as described in this guide.
4. On each server where IIS is installed, do the following.
   a. Open IIS Manager, right-click **Application pools** and select **New**.

      The Add New Application Pool window opens.
   b. Enter `DetectAppPool` in the **Application pool ID** field and click **OK**.

      DetectAppPool is created under **Application Pools**.
   c. Right-click **DetectAppPool** and select **Properties**.

      The DetectAppPool Properties window opens.
   d. On the Identity tab of the dialog, do the following.
      - Select **Configurable**.
      - Enter the user name and password of the Detect system user you created.
      - Click **OK**.
5. In IIS Manager, under **WebSites > Default WebSite** right-click **Detect** and select **Properties**.

   The Detect Properties window opens.

6. On the Virtual Directory tab, select **DetectAppPool** in the **Application Pool** drop-down and click **OK**.
7. Exit IIS Manager.
8. Log in to the Detect Configuration Utility and double-click **Configuration** to access the Configuration pop-up.
9. Under **System Options**, locate the **IIS Instance determined by** property, set it to **Environment**, and click **OK**.
10. Right-click **Environments** and select **New** to access the Environment pop-up.
11. Complete the fields in the Environments pop-up as follows, and then click **OK**.

Field	Description
Name	Enter a name for the environment.
Description	Enter a description for the environment.
EMC Server	Select an EMC server for the environment.
TSM Server	Select a TSM server for the environment.
Clusters	Click **Edit** to open the Environment Clusters pop-up, select one or more enabled clusters for the environment, and then click **Save**.

12. Set DCOM permissions as described in this guide, as appropriate for your environment. Follow these guidelines.
    - Set DCOM permissions for the TSM and EMC that is defined for each environment.
    - Set DCOM permissions for each engine, feeder, and listener in each cluster.
    - Set DCOM permissions for each deployed Detect web application.
13. Exit the Configuration Utility.

To configure multiple environments to run with a single instance of IIS (64-bit machines only)

1. Install Detect as described in this guide, following these guidelines.
    - Specify the same database information for the Detect system tables each time you run the Detect installer.
    - For each instance of IIS that you plan to use with Detect, perform the procedures described in this guide for the web server unit.
2. Configure Detect and set up the desired Detect clusters as described in this guide.

   Configure all of the instances of Detect to use the same data sources. Reference copies of the same data source with different names in each environment.
3. On each server where IIS is installed, verify that a Detect system user is created as described in this guide.
4. On each server where IIS is installed, do the following.
    a. Open IIS Manager, select **Application Pools** in the Connections panel and in the Actions panel click **Add Application Pool**.

       The Add New Application Pool window opens.
    b. Enter `DetectAppPool` in the **Application pool ID** field and click **OK**.

       DetectAppPool is created under **Application Pools**.
    c. Select **DetectAppPool** in the Application Pools panel and in the Actions panel click **Advanced Settings**.

The DetectAppPool Properties window opens.

d. In the General section, do the following.
   - Ensure that **Enable 32-bit Applications** is set to **True** for the application pool, and that the pool is started.
   - Set **Managed Pipeline Mode** to **Classic**.

e. In the Process Model section, do the following.
   - Select the **Identity** field and click the button to open the Application Pool Identity window.
   - Select **Custom account** and click **Set** to open the Set Credentials window.
   - Enter the user name and password of the Detect system user you created.

f. Click **OK**.

5. Exit IIS Manager.
6. Log in to the Detect Configuration Utility and double-click **Configuration** to access the Configuration pop-up.
7. On the Advanced Configurations tab, under **System Options**, locate the **IIS Instance determined by** property, set it to **System**, and click **OK**.
8. Right-click **Environments** and select **New** to access the Environment pop-up.
9. Complete the fields in the Environments pop-up as follows, and then click **OK**.

Field	Description
Name	Enter a name for the environment.
Description	Enter a description for the environment.
EMC Server	Select an EMC server for the environment.
TSM Server	Select a TSM server for the environment.
Clusters	Click **Edit** to open the Environment Clusters pop-up, select one or more enabled clusters for the environment, and then click **Save**.

10. Exit the Configuration Utility.

Appendix A. Upgrading IBM Unica Detect

This section provides instructions for upgrading IBM Unica Detect. The tasks you perform depend on your existing version of Detect.

Perform these tasks on all machines on which Detect is installed.

Important: The information in this section applies to most customers upgrading from version 6.8.8, 7.0.x, 7.1.x, 7.2.x, 8.1.x, or 8.2.x. However, you should check with IBM Unica Consulting Services before upgrading. Due to the variety of environments and versions, each upgrade must be handled on a case-by-case basis.

Task: Check prerequisites

Before performing any installation and upgrade tasks, read "Task: Verify installation requirements" on page 11 and make sure your environment meets the installation requirements.

Task: Back up batch files

If any of your `.bat` files have been modified, save them to another directory so you can restore them after you run the installation. The existing batch files will be overwritten when you install Detect. The batch files are in the `Application\bin` directory under your Detect installation directory.

Task: Clean up the registry before you install Detect (6.8.8 upgrades only)

Perform this task only if you are upgrading from Detect version 6.8.8.

In this task, you unregister DLLs used by Microsoft.NET\Framework 1.

1. Find `UnReg.Net1.1Assemblies.bat` which is packaged with your downloaded Detect software.
2. Move it to the `Application\bin` directory under your Detect installation directory.
3. Open it and verify that the path to `regasm.exe` is set to:

   `C:\Windows\Microsoft.net\Framework\v1.1.4322.`
4. Run `UnReg.Net1.1Assemblies.bat` from the \bin directory.

   The script cleans the Detect registry entries.

Task: Uninstall Detect

1. Stop the EMC and the IIS services before uninstalling Detect.
2. Uninstall Detect using the WIndows **Add or Remove Programs** function.

Task: Install Detect

Read Chapters 4 and 5, Chapter 5, "Installing the IBM Unica Detect Web Server Unit," on page 11 and Chapter 6, "Installing the IBM Unica Detect Cluster Units," on page 25 to install Detect. Determine which of the steps you need to perform. You may not need to perform every step in Chapters 4 and 5, especially if your environment has not changed and you install to the same directory as your previous installation. However, you should always verify that the sharing and DCOM configurations described in every step are in effect.

Run the Detect installer on the web server unit and each of your cluster units, using the following guidelines.

- On the web server unit, select the **IIS Server** checkbox. On cluster units, deselect this checkbox. If a single machine is used for both purposes, select the checkbox.
- Enter the database user names that are used in the existing installation
- For the database password, enter the password for the existing Rule schema user. (In earlier installations, the name for the Rule schema user was CEE4.)

Task: Return the batch files to the bin directory

If you backed up the batch files in "Task: Back up batch files" on page 65, restore them to the `Application\bin` directory under your Detect installation directory.

Task: Edit Internet Information Services settings (6.8.8 upgrades only)

Perform this set of tasks only if you are upgrading from Detect version 6.8.8.

To set Detect as the default web site

1. In Internet Information Services (IIS) Manager, under the local computer, expand **Web Sites > Default Web Site**.
2. Under **Default Web Site**, right-click **Detect** and open **Properties**.

To adjust ASP.NET settings

1. In Internet Information Services (IIS) Manager, under the local computer, click the ASP.NET tab.
2. Use the dropdown list to change the ASP.NET version setting to 2.0.*n*.

   In this step, *n* represents the rest of the version number.
3. Click **Apply**.

To adjust Documents settings

1. In Internet Information Services (IIS) Manager, under the local computer, click the **Documents** tab.
2. Remove existing content pages.
3. Click **Add** and then add **login.aspx** as a document. Click **OK**.
4. Verify that **Enable default content page** is selected.
5. Click **OK** to close the Detect Properties window.

To set the web service extension (IIS 6.0 only)

Perform this task only if you have IIS version 6.0.

1. In the Internet Information Services tree, highlight **Web Service Extensions**.

2. Ensure that **Active Server Pages** and ASP.NET v2.0 are listed and that both are set to **Allowed**.

   To change an item from Prohibited to Allowed, highlight it in the list and click **Allow**.
3. If ASP.NET v2.0 is not listed, then use this step. Otherwise, skip to step 4.
   a. Click **Action > Add a new Web service extension**.
   b. When the New Web Service Extension window opens, click **Add**.
   c. Browse to the `aspnet_isapi.dll` , which by default is in:

      `C:\WINDOWS\Microsoft.NET\Framework\v2.0.n\aspnet_isapi.dll`

      In this step, n represents the rest of the version number. Make a note of the version number so you can use it in step e.
   d. Click **OK**.
   e. Give it the Extension name of **ASP.NET v2.0.*n*,** replacing n with the rest of the version number.
   f. Select the **Set the extension status to Allowed** checkbox if it is not already selected.
   g. Click **OK** and close the window.
4. Restart IIS service.

Task: Upgrade the Detect database

To upgrade the Detect database, you will perform the following steps, which are described in detail in this section.

- Locate and edit the SQL scripts provided with your upgrade installation.
- If necessary for your environment, perform some manual steps
- Run the modified SQL scripts against your Detect database.

Order for running the database scripts

The Detect database upgrade scripts are incremental by version. You must start with the script that applies to your current version of Detect and run each database upgrade script in order, stepping up to the next version.

The first script that you run is the one that is named with the next available version number that is higher than your current version of Detect. For example, if your current version of Detect is 7.0.1, you would start with the script named `Migration702.sql`.

Note that upgrade scripts are not provided for every step because the database was not changed in every version of Detect. See “List of database scripts” on page 68 for a complete list of scripts and the versions to which they apply.

Location of all database scripts

All of the database upgrade scripts are located in the `application\database\upgrade` directory under your Detect installation directory.

Instructions in the database scripts

Each script contains instructions for running it and a description of the database changes it makes. Read the instructions in the scripts carefully before running them.

Read this section to gain a general understanding of what you must do to run the scripts.

Requirement to replace parameters in the database scripts

Many of the scripts instruct you to replace parameters with values that apply to your database. In various scripts, the instructions may ask you to replace some or all of the following parameters.

- `%RuleUser%`—On SQLServer, the name of your current rule schema. On Oracle, the user name for your current rule schema, typically CEE4.
- `%HistoryUser%`—On SQLServer, the name of your current history schema. On Oracle, the user name for your current history schema, typically CEE1.
- `%Vendor%`—Your three-digit vendor code.
- `%TBLSP%` (Oracle only)—Name of the tablespace for tables.
- `%IDXSP%` (Oracle only)—Name of the tablespace for indexes.

You might want to gather this information before you start to run the database upgrade scripts.

Requirement to extract and run parts of some database scripts

Some of the scripts have portions that are intended to be run multiple times, with new values replacing parameters in the script each time. Comments in the scripts label the sections and provide instructions for changing the parameters. The following instructions occur in various scripts.

- `RUN ONCE ONLY`—Run this section only once during the migration process.
- `RUN ONCE PER VENDOR`—Run this section once for each vendor that exists in the database, appropriately replacing the `%Vendor%` parameter each time you run it. Typically, you will have only one vendor.
- `RUN ONCE PER WORKSPACE`—Run this section once for each workspace that exists in the database, appropriately replacing the `%Workspace%` parameter each time you run it.

In addition, there may be optional and mandatory sections in the scripts. In that case, run only the parts that apply to your database

List of database scripts

The following table lists the database upgrade scripts and the versions to which they apply.

Script	Version upgraded by the script
`Migration70.sql`	Upgrades the database from 6.8.8 to 7.0.0
`Migration701.sql`	Upgrades the database from 7.0.0 to 7.0.1
`Migration702.sql`	Upgrades the database from 7.0.1 to 7.0.2 **Important**: If you are upgrading from Detect version 7.0.x or later, and if you have used the Outcome Management Tool (OMT), you must perform the procedure described in "To remove database tables before running the Migration702.sql script" on page 69 before running this script.
`Migration71.sql`	Upgrades the database from 7.0.2, 7.0.3, or 7.0.4 to 7.1.0
`Migration72.sql`	Upgrades the database from 7.1.0 or 7.1.1 to 7.2.0

Script	Version upgraded by the script
Migration721.sql	Upgrades the database from 7.2.0 to 7.2.1
Migration722.sql	Upgrades the database from 7.2.1 to 7.2.2
Migration723.sql	Upgrades the database from 7.2.2 to 7.2.3
Migration724.sql	Upgrades the database from 7.2.3 to 7.2.4
Migration725.sql	Upgrades the database from 7.2.4 to 7.2.5 and 7.2.6
Migration810.sql	Upgrades the database from 7.2.5 or 7.2.6 to 8.1.0 **Important**: On Oracle, you must use a system level login to execute this script. Also, the upgrade script does not upgrade the PUB or XPB workspaces automatically. You must upgrade those tables manually. Contact IBM Unica technical support for assistance.
Migration811.sql	Upgrades the database from 8.1.0 to 8.1.1 **Important**: The migration script for Oracle contains instructions for changing the data type in the state tables. Also, for Oracle only, the upgrade script does not upgrade the PUB or XPB workspaces automatically. You must upgrade those tables manually, using the SQL described in "To upgrade an Oracle database from version 810 to 811." If you are not confident you can perform this procedure correctly, contact Unica technical support for assistance.
Migration820.sql	Upgrades the database from 8.1.1 to 8.2.0
Migration850.sql	Upgrades the database from 8.2.0 to 8.5.0

To remove database tables before running the Migration702.sql script

Perform this task only if both of the following are true.

- You are upgrading from Detect version 7.0.x or later.
- You have used the Outcome Management Tool (OMT).

1. Start the Outcome Management Tool (OMT).
2. Use the OMT to drop tables for all worskspaces.

   This step clears the *vendor*_outcome_triggers table. All OMT table definitions already created for the tool, such as table names and column aliases, are still available.
3. Quit the OMT.
4. Query the *vendor*_outcome_triggers table to verify that it is empty.

   If the *vendor*_outcome_triggers table is not empty, drop the tables it lists.

To upgrade an Oracle database from version 810 to 811

Perform this procedure after you have run the Migration811.sql script and before you run the Migration820.sql script. If you are not confident you can perform this procedure correctly, contact Unica technical support for assistance.

Important: Back up your Oracle database and your PUB and XPB workspaces before you perform this procedure, so that you can restore them if a problem occurs.

After you run the script, prepare a file with the following SQL. This SQL is similar to all the upgrade scripts supplied with Detect, in that you must substitute your values for the variables enclosed in percent signs. After you have inserted your values, run the SQL against your Oracle database.

```
ALTER TABLE %HistoryUser%.%Vendor%_pub_visitor MODIFY ( DATA BLOB )
/
ALTER INDEX %HistoryUser%.IX_%Vendor%_pub_visitor rebuild
/
ALTER INDEX %HistoryUser%.IX_%Vendor%_pub_visitor_FT rebuild
/
ALTER INDEX %HistoryUser%.PK_%Vendor%_pub_visitor rebuild
/
ALTER TABLE %HistoryUser%.%Vendor%_xpb_visitor MODIFY ( DATA BLOB )
/
ALTER INDEX %HistoryUser%.IX_%Vendor%_xpb_visitor rebuild
/
ALTER INDEX %HistoryUser%.IX_%Vendor%_xpb_visitor rebuild
/
ALTER INDEX %HistoryUser%.PK_%Vendor%_xpb_visitor rebuild
/
```

Task: Adjust the feeder throttle parameters

The `Migration701.sql` upgrade script sets the default values for the Feeder Throttle parameters. If you ran this script, adjust these values as necessary, in the Configuration Utility's Advanced Configuration tab.

If you leave the Feeder Throttle parameter set to **On**, then set the **Feeder Pause Interval** parameter to **0** (zero, for off).

Task: Clear the .NET server cache on the IIS machine

As a best practice, you should clear the .NET server cache on the IIS machine.

1. On the IIS machine, stop IIS.
2. Delete everything under the following directory:
   `C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727\Temporary ASP.NET Files\detect`
3. Restart IIS.

Task: Delete temporary internet files from client browsers

As a best practice, you should clear the temporary internet files on each client browser used to access the Detect user interface.

You can now run Detect.

Task: Adjust user roles (upgrades from pre-8.2.0 versions only)

Because the security model was completely revised in Detect version 8.2.0, when you upgrade Detect to version 8.2.0 or later from a pre-8.2.0 version, all users are assigned the **Power User** role, except for the **host_admin** user. After upgrade, the **host_admin** administrative account is replaced by a different administrative account with the user name **system**.

This account enables you to create users and assign security roles. It does not allow access to any other features, and you cannot modify or delete this role, so you must assign the appropriate roles to other existing users or to new users.

1. Log in to Detect with the following credentials.

- User name—system
- Password—host

2. Adjust user roles on the **Settings > Users** page.

   See the *IBM Unica Detect Administrator's Guide* for information on user roles and permissions.
3. If you want to set and enforce password policies, the system user can log in to the Configuration Utility and set them.

Task: Update passwords

The encryption algorithm changed in Detect version 8.5.0 to support multi-byte characters. If you are upgrading from a version earlier than 8.5.0, you must re-enter and save all the passwords for the following.

- User accounts—Set these through the Detect web application.
- Database access accounts—Set these on the System Connections and User Connections tab in the Configuration pop-up of the Configuration Utility.

Task: Update data sources, if necessary

If you are upgrading from a version earlier than 8.5.0, the upgrade scripts automatically assigned values to two new data source attributes, as follows.

- File Encoding—By default, this is set to **Western European (Windows)** after upgrade. This value must match the encoding of the data source file.
- File Date Format—By default, this is set to **English (United States)** after upgrade. This value must match the date format used in the data source file.

If the default values set by the upgrade process do not match the file encoding or date format used in the data source file, you must edit your existing profile and transaction data sources in the Detect application, and select date and currency formats. These options were added to the data source editor for these file types in version 8.5.0.

Appendix B. Localizing Your Detect

Starting with version 8.5.0, IBM Unica Detect supports double byte characters for user input and data.

A prerequisite is a Detect database that also supports the desired characters.

Broadly, this support consists of the following.

- Detect can accept input files containing double byte characters, manipulate those double byte characters in containers and comparisons, and generate double byte characters in the outcome table.
- You can use double byte characters when you log in to Detect, when you name most Detect objects, and when you configure components.
- Utilities support double byte characters for user input such as user name, password, and paths to feed files, and in the feed files themselves.

There are some exceptions to this support, as follows.

- Entity codes use only lower case letters of the English alphabet. If you use an invalid character, this rule is displayed in the validation message in the user interface.
- Vendor and workspace codes must use only upper case letters of the English alphabet. If you use an invalid character, this rule is displayed in the validation message in the user interface.
- In component editors, all numbers that you enter as a constant must be in U.S. format. You must use a period (.), not a comma (,) to delimit decimals.

Adding this support has affected configuration in several areas of the Detect application, as described in the remainder of this section.

Configuration for most feed files

The following table describes the configuration properties that affect the format of all feed files except the Ramp-up Engine feed files.

Property	Where to set	Required?
File Encoding—Sets the encoding of profile and transaction feed files, independently of the **Default Input File Encoding** property in the Configuration Utility. The format you specify in this field must match the format in which the feed file is saved.	Data source editor	Yes
File Date Format—Sets the date format of profile and transaction feed files, independently of the **Locale Identifier** property in the Configuration Utility. The date format you specify in this field must match the format used for dates in the feed file.	Data source editor	Yes
File Currency Format—Sets the currency format of profile and transaction feed files. The currency format you specify in this field must match the format of currency used in the feed file.	Data source editor	Required only when currency strings are present in profile or transaction feed files.

Property	Where to set	Required?
Feed File Collation—Sets the collation (sort order) of all feed files. When the Detect engine processes multiple feed files, it uses the collation specified in this property to sort recirds in all feeds and it processes the records in the sorted sequence.	Configuration utility	Yes

File encoding for Ramp-up Engine feed files and State Snapshot reports

One configuration property determines the file encoding for Ramp-up Engine feed files and State Snapshot reports.

Default Input File Encoding—Replaces the former **Default Feed Encoding** property used in previous releases. Determines the file encoding for Ramp-up Engine feed files, Presentation Layer List files, and State Snapshot reports. All log and other files generated by Detect are saved in UTF 8 encoding, regardless of any configuration settings.

Set this property in the Configuration Utility.

This property is required only when currency strings are present in the files listed above.

Date format for the user interface and engine

One configuration property affects the date format used by the Detect engine and for text boxes in the user interface.

Locale Identifier—Determines the date format that must be used when users enter dates in text boxes (rather than selecting them from a drop-down list), including start and end dates for reports, inactivity target dates in the engine manager, and date constants in Simple or Qualifier components. Also determines the dates added to outcome messages, the date format the OMT uses to parse dates from outcomes, and the date format of timestamps in Ramp-up Engine feed files. Note that the date format in the Type Descriptor and State Snapshot reports is ISO.

Set this property in the Configuration Utility. For double byte languages, set to ISO 8601.

This property is required.

About selecting date and currency formats for data sources

Use the following table as a guide for selecting the date and currency formats in the Detect data source editor.

Language	Date format	Currency format
English	mm/dd/yyyy	$123,456.00
Spanish	mm/dd/yyyy	$123,456.00
French	dd/mm/yyyy	123 456,00 €
German	dd.mm.yyyy	123.456,00 €

Language	Date format	Currency format
Italian	mm/dd/yyyy	€ 123.456,00
Brazilian Portuguese	dd/mm/yyyy	R$123.456,00
Simplified Chinese	yyyy/mm/dd	¥ 123,456.00
Japanese	yyyy/mm/dd	¥ 123,456.00
Korean	yyyy-mm-dd	₩ 123,456.00

About selecting the file encoding for data sources

Use the following table as a guide for selecting the file encoding in the Detect data source editor. The table provides guidelines for which options in the data source editor are appropriate for each supported language. It also maps the options in the data source editor to their corresponding standard and code page.

Note: Where multiple languages and/or options are shown in a row, any of the listed options is appropriate for any of the listed languages.

Language	Options in data source editor	Standard	Code page
All languages	• Unicode (UTF-8) • Unicode (UTF-7) • Unicode (Little endian) • Unicode (Big endian)	• UTF-8 • UTF-7 • UTF-16 • UTF-16	• 65001 • 65000 • 1200 • 1201
• English • Spanish • French • German • Italian	• Western European (Windows) • Western European (ISO) • Latin 3 (ISO) • Latin 9 (ISO)	• Windows-1252 • ISO-8859-1 • ISO-8859-3 • ISO-8859-15	• 1252 • 28591 • 28593 • 28605
Brazilian Portuguese	• Western European (Windows) • Western European (ISO) • Latin 9 (ISO)	• Windows-1252 • ISO-8859-1 • ISO-8859-15	• 1252 • 28591 • 28605
Simplified Chinese	Chinese (GB18030)	GB18030	54936
Japanese	• Japanese (Shift JIS) • Japanese (JIS 0208-1990 and 0212-1990)	• Shift_JIS • EUC-JP	• 932 • 20932
Korean	Korean (EUC)	EUC-KR	51949

Configuring the locale for documentation

When Detect documentation exists in languages other than English, you can set the locale for documentation as follows.

Note: Before performing this procedure, you may want to check with IBM Unica Technical Support to see if the translated documentation has been delivered for the language in which you are interested. However, if you do perform the procedure,

and the documentation is not yet available in the language you selected, the documentation links will be redirected to the English version. When the translated documentation is available, the documentation links will allow you to view the translated documents (Help and PDF).

1. Locate the `Application` directory under your Detect installation directory.
2. Open the `web.config` file in a text editor and find this line: `<add key="docLocale" value="en_us"/>`
3. Change `en_us` to your desired locale, as shown in the following table, and then save and close the file.

Language	**docLocale value**
English	en_us
Spanish	es_es
French	fr_fr
German	de_de
Italian	it_it
Brazilian Portuguese	pt_br
Simplified Chinese	zh_cn
Japanese	ja_jp
Korean	ko_kr

Appendix C. Installing MSMQ

1. Insert the Windows CD.
2. Open the Windows Control Panel.
3. Double-click **Add/Remove Program**.
4. Click **Add/Remove Windows Components** to start the Windows Components wizard.
5. Select **Application Server**.
6. Click **Details** to open the Application Server window.
7. Select the box associated with **Message Queuing**.

   Use the default settings for Message Queuing.
8. Click **OK**.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information about the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
170 Tracer Lane
Waltham, MA 02451
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not

been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml.

Printed in USA