

DME version 3.6 SP1

# Installing DME connector: Windows

Installing DME

Created on 24-04-2012

Windows



# Contents

<b>Installing connectors</b>	<b>2</b>
Collaboration system considerations.....	2
Domino .....	2
Exchange.....	3
Getting the connector installer .....	3
Installing the DME connector.....	4
Choose version.....	4
Choose connector location .....	5
Enter connector information .....	5
Installation complete.....	7
Adjusting RAM usage.....	7
Connectors outside the LAN .....	8
Securing the traffic between server and connector(s) .....	9
Configuration in the web interface .....	10
Configuring DIIOP/Corba connector.....	11
Configuring Notes session connector .....	11
Exchange NTLM setup .....	12
Upgrading the connector .....	12
Contact mapping files.....	13
If the connector upgrade fails.....	13

# Installing connectors

When the collaboration system has been prepared, and one or more DME servers have been installed, you are ready to install one or more DME connectors.

The DME connector requires Java 6 SDK. If this is not installed, the installer will download and install it. This requires an internet connection on port 80. Otherwise you must download and install the latest Java 6 SDK manually before installing the DME connector.

---

## Please note

Windows has a network limitation, which is described in Microsoft knowledge base article **KB196271** <http://support.microsoft.com/kb/196271>. According to this, MS Windows has a limit on the number of ports that can be used by a program (such as the DME connector) when used for longer period of times or under heavy load or unstable network conditions. This makes Windows prevent the connector from creating new TCP/IP connections. This, in turn, makes it appear that the connector does not operate correctly, but is actually periodically limited by the OS due to the port restrictions. To circumvent this, set the **MaxUserPort** setting as described in the knowledge base article.

---

Before installing the connector, you need to know the environment into which the connector is to be installed. This is outlined in the following.

---

## Collaboration system considerations

DME is a time-critical application, which relies on good I/O throughput to the filesystem, queues and timeout values to control the flow of data between the collaboration system and the clients. The use of anti-virus scanners (AV) can in some cases be disruptive to this process. For more information about the use of AV scanners, please see the chapter **Anti-virus programs** in the guide **Installing DME on Windows** on the **DME Install site** <http://install.excitor.dk/documentation/install/index.php>.

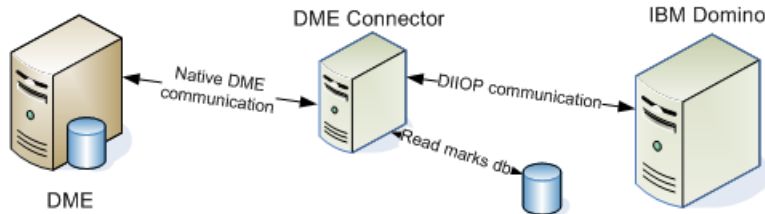
The connector installation depends on whether you run Lotus Domino or Microsoft Exchange as the backend collaboration system.

## Domino

The connector can be linked with a Domino system in one of three ways. For more information about how Domino should be set up, please see the **Domino Integration Guide**. The following is just an outline.

## DIIOP/Corba

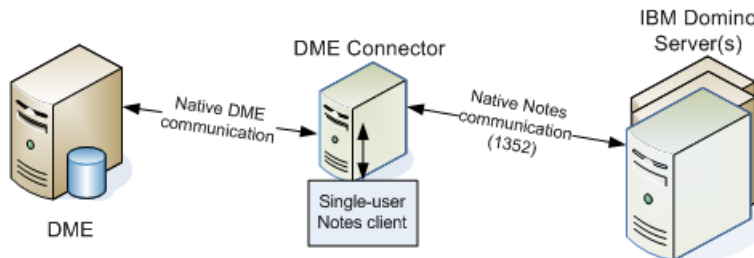
If the connector is to link with Domino using DIIOP/Corba, the connector can be installed on any machine in the network.



This is the traditional, pre-3.0 way of connecting to Domino.

## Notes session

When connecting to Domino using a Notes session, the connector can be installed on any machine in the network. A stand-alone Notes client for Windows must be installed on the server on which you are installing the connector, and configured to connect to the Domino server through a proxy user as described in the *Domino Integration Guide*.



The Notes client will then communicate with Domino using the native protocol on port 1352. Be aware of the following:

- 1 During connector installation, you must point out the Lotus Notes program directory.
- 2 During connector installation, you must point out the location of **Notes.jar**. The installer will download and install the Java SDK if it is not already installed on the server.

## Exchange

The connector can be installed on any machine in the network. Access to Active Directory and the Exchange OWA or CAS server are required. For more information, see the *Exchange 2003 integration Guide* and the *Exchange 2007/2010 integration Guide*.

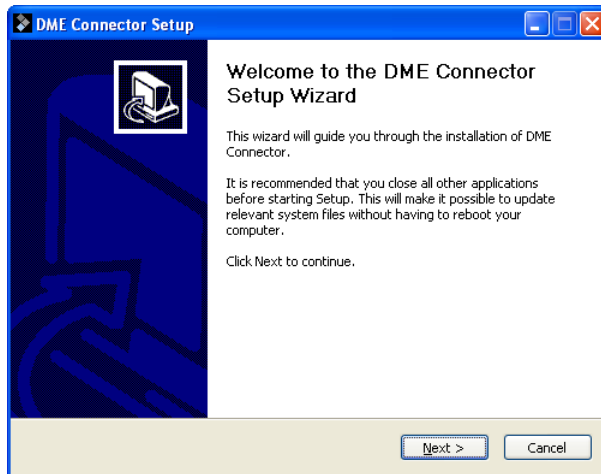
---

# Getting the connector installer

The connector installer can be downloaded from the *DME install site* <http://install.excitor.dk>. No password is required.

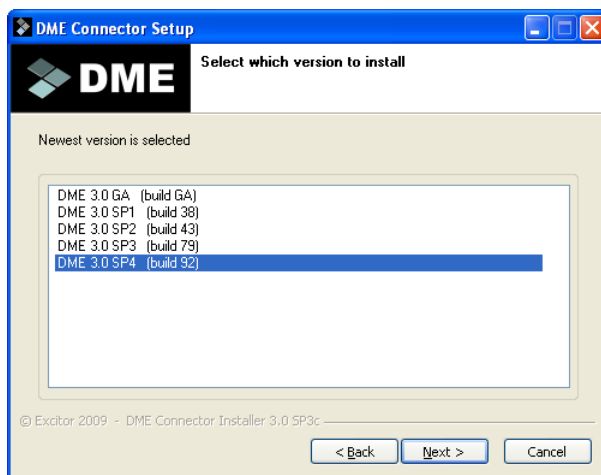
# Installing the DME connector

Run the installation program `dme-connector.exe`.



Click **Next** to continue.

## Choose version



Choose the version of DME that the connector will bind to.

---

It is important that you choose the same service pack level as the DME server.

---

Click **Next** to continue.

## Choose connector location

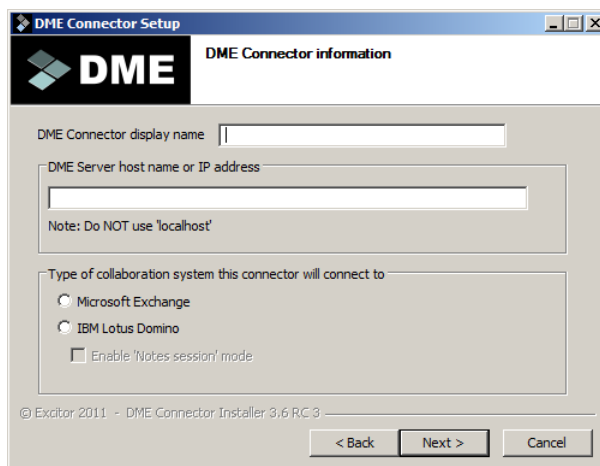


Enter the path to install the connector files to. The default is **C:\Program Files\dmeconnector**.

If you want to download the connector installation files to disk instead of installing them directly from the Excitor website, you should download the offline installer from the **DME install site** <http://install.excitor.dk>. With the offline installer, you can install both the DME server and connectors on other servers, even if they do not have Internet access.

Click **Next** to continue.

## Enter connector information



Enter the display name of the connector, which will be shown in the **Connector** tab of the DME server administration web interface.

Then enter the host name (resolvable by DNS) or IP address of the DME server to which the connector is to link. Note that `localhost` is not a valid entry here.

Please note that these fields cannot be changed during an upgrade.

Finally choose whether this connector should be used in connection with a Microsoft Exchange or an IBM Lotus Domino collaboration server.

If you are running Domino, you are given the option to enable the connector in **Notes session** mode. If not, it is assumed that the connector will be used as a **Remote/Corba** connector. See *Lotus Domino only* on page 6.

## Cluster setup

If the connector is to be part of a DME cluster, enter the host name of *all DME servers in the cluster* in the field **DME Server host name(s) or IP address(es)** - separated by comma and no spaces. For example:

**dme1.company.com:1100,dme2.company.com:1100,dme3.company.com:1100**

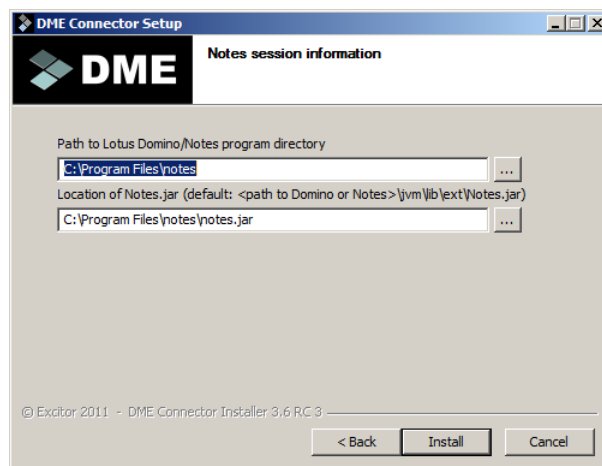
Port **1100** is the default port used by DME for server nodes in a cluster. For information about enabling and configuring the cluster, please request separate documentation.

## Lotus Domino only



In the previous **DME Connector information** window, you could choose to enable **Notes session** mode (that is, not using Remote/Corba/DIIOP mode).

In this mode, the connector must be installed on a computer running a *Notes 8.5.x Basic* client. Supply the full path to the Domino or Notes program directory, and the full path to **Notes.jar**.



The default location of **Notes.jar** is:

**<path to Domino or Notes>\jvm\lib\ext**

In both cases, the installer will check if the Java 6 SDK is installed on the server. If it is not, the installer will download and install it. There is a difference though:

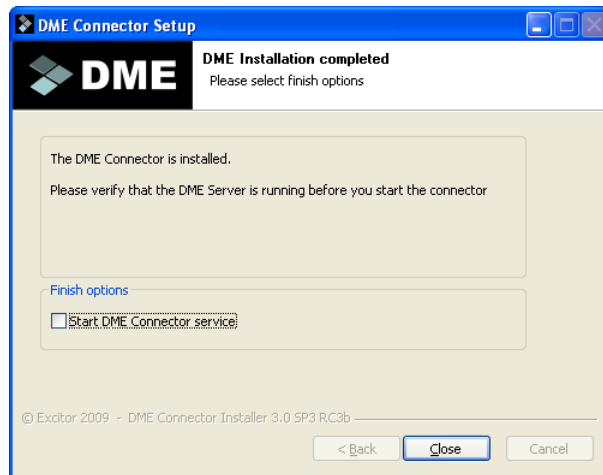
- The installer will install Java 32 bit if **Notes session** is selected.
- The installer will install Java 64 bit if **Notes session** is not selected (Remote/Corba/DIIOP is selected).

This is important to know if you later change modes, for instance from **DIIOP** to **Notes session**: Uninstalling the connector in **Notes session** mode will *not* remove 32 bit Java, and uninstalling the connector in **DIIOP** mode will *not* remove 64 bit Java. As a consequence, you need to manually uninstall Java before reinstalling the connector in a new mode.

For more information about preparing Domino for the DME connector, see the Domino Integration Guide.

Click **Install** to complete the installation.

## Installation complete



When the installation is finished, you can select the field **Start DME Connector service** to start the DME connector service when the installer is closed. The DME connector service is called `dmeconnector`.

All configuration of the DME connector is done from the DME server web interface.

## Adjusting RAM usage

If you are running DME on a 64bit OS with plenty of RAM and many DME users per connector (1000+), you can optimize the RAM allocation.

You do this by changing the initial and maximum JVM (Java Virtual Machine) memory settings in the connector configuration files. Normally the connector will be the service on the machine using the most RAM, up to 2 or 4 times more than the DME server. However, you will have to test what works best in your environment. Be sure not to allocate too much RAM, as you may risk that the entire server becomes unstable if you allocate more than the OS can easily spare.

The *location* of the configuration files and *default values* of the two parameters are as follows:

Location of the DME server configuration file (*Windows*):

```
..\dme\jboss\conf\wrapper.conf
```

```
set.MEM_INIT=512
```

```
set.MEM_MAX=1280
```

### Windows

Location of the DME connector configuration file (*Windows*):

```
..\dmeconnector\conf\wrapper.conf
```

```
# Initial Java Heap Size (in MB)
```

```
wrapper.java.initmemory=512
```

```
# Maximum Java Heap Size (in MB)
```

```
wrapper.java.maxmemory=1024
```

### Linux

Location of the DME server configuration file (*Linux*):

```
/etc/init.d/dme_<instancename>
```

```
# MINMEM          This is the minimum amount of RAM in Mb that DME
#                is to allocate
MINMEM=512

# MAXMEM          This is the maximum amount of RAM in Mb the DME
#                is to allocate. This cannot be more than 2048
#                on a 32bit operating system.
MAXMEM=1280
```

Location of the DME connector configuration file (*Linux*):  
`/var/dme/instances/<instancename>/connector/conf/wrapper.conf`

```
# Initial Java Heap Size (in MB)
wrapper.java.initmemory=512

# Maximum Java Heap Size (in MB)
wrapper.java.maxmemory=1024
```

When you make a change, restart the service, and test the results.

---

## Connectors outside the LAN

Connectors can be installed in remote places, even in places that require connection via a remote LAN via a VPN connection (WAN) or via the Internet.

This can cause some problems, since the initial connection between the connector and the DME server uses an initial IP address or hostname to connect to the DME server, with the DME server returning its IP address or hostname (the bind address) via the JNDI protocol. If there is a difference between the connector's connection point (IP or hostname) and the IP or hostname of the DME server, then the connector cannot establish a connection to the DME server.

The connection is established by the connector via the information in the `dme-config.xml` file. If the server IP is for instance `172.16.15.15` and DME is on `172.16.15.15` then DME server will respond via JNDI with the `172.16.15.15` IP address, and the connection is established.

If the connector's connection point is for instance `dme.example.com` (an external IP address), and the DME server bind address is `172.16.15.15`, then the connector cannot connect to the DME server since it will try to establish the connection to the DME server's bind address (`172.16.15.15`) and not `dme.example.com`.

**As a work-around** for this problem, you can configure the connector machine in the following way.

- 1 On the machine where the connector is installed, enter the IP and hostname of the DME server in the `hosts` file:

Linux: `/etc/hosts`

Windows: `c:\windows\system32\drivers\etc\hosts`

Enter `x.x.x.x` `dme.example.com`

- 2 On the DME server, set the `IP_ADDRESS` in the `dme_base` init script (Linux) or the bind address in the `wrapper.conf` (Windows) to `dme.example.com` or whatever the DME server's external connection point is, and set up a `hosts` file entry with the local IP address on the machine where the DME server is installed:

`172.16.15.15` `dme.example.com`

This will enable the DME server to bind to the local IP mentioned in the `hosts` file, but it will bind as `dme.example.com` and will return that hostname to the connector, which uses the different IP for `dme.example.com` in its `hosts` file during the connection process. Thus the connection will succeed.

---

# Securing the traffic between server and connector(s)

In order to ensure end-to-end SSL encryption from device to DME server, the communication between the DME server and the DME connectors must be encrypted.

The following guide shows how to set up secure SSL data channels between the DME server and the DME connectors.

### Step 1: Create the SSL keystore on the server

For security, remember to change the default password (in the examples below set `mypassword` to something else in all commands and wrapper/init scripts).

Please note that `<JAVA_HOME>` below means the path to the root folder of the Java installation. This is

`C:\Program Files\Java\jdk1.6.x.yy`, where `x.yy` is the minor version and build number of Java.

- 1 Open a command prompt.
- 2 Go to the `etc` directory on the DME server:  

```
cd "C:\Program Files\dme\jboss\server\default\etc"
```
- 3 Run `<JAVA_HOME>\bin\keytool -genkey -alias dmessl.keypair -validity 3650 -keyalg RSA -keystore dmessl.keystore -storepass mypassword -keypass mypassword -dname "CN=dmessl"`
- 4 Run `<JAVA_HOME>\bin\keytool -export -alias dmessl.keypair -keystore dmessl.keystore -rfc -file dmessl.cer -storepass "mypassword" -keypass "mypassword"`
- 5 Run `<JAVA_HOME>\bin\keytool -import -noprompt -alias dmessl.keypair -file dmessl.cer -keystore dmessl.truststore -storepass mypassword -keypass mypassword`
- 6 Leave the file `dmessl.keystore` in the `C:\Program Files\dme\jboss\server\default\etc\` directory.
- 7 Copy the file `dmessl.truststore` to the connectors' `C:\Program Files\dmeconnector\conf\` directory.

The `truststore` file is no longer required on the server, and you may delete it or keep it as a backup.

You might find it easier to use a GUI keytool to perform these steps rather than using the command line. One such free tool is *KeyTool IUI*  
<http://www.softpedia.com/progDownload/KeyTool-IUI-Download-77555.html>.

### Step 2: Server settings

Insert the following in `C:\Program Files\dme\jboss\conf\wrapper.conf` (two lines), and remember to *change the default password*. Change the number `x` in `wrapper.java.additional.x` to a sequential number, if the number is already used.

```
wrapper.java.additional.18=-Djavax.net.ssl.keyStore="%DME_HOME%\etc\dmessl.keystore"  
wrapper.java.additional.19=-Djavax.net.ssl.keyStorePassword=mypassword
```

### Step 3: Configuration changes on the server

Please note that the files referenced here are available for download from the **DME install** <http://install.excitor.dk> site. Look for the link to **Files for SSL connection** in the **DME support files** tab.

- 1 Download the **SSL configuration files** from the **DME install** <http://install.excitor.dk> site, and unzip the package.
- 2 Delete (do not just rename) the file `C:\Program Files\dme\jboss\server\default\deploy\jboss-messaging.sar\remoting-bisocket-service.xml` with the equivalent file from the downloaded package.
- 3 Copy the file `remoting-sslbisocket-service.xml` from the downloaded package to `C:\Program Files\dme\jboss\server\default\deploy\jboss-messaging.sar\`.
- 4 Replace the `C:\Program Files\dme\jboss\server\default\deploy\jboss-messaging.sar\connection-factories-service.xml` with the equivalent file from the downloaded package.
- 5 Replace the `C:\Program Files\dme\jboss\server\default\deploy\jboss-messaging.sar\messaging-service.xml` with the equivalent file from the downloaded package.
- 6 Replace the `C:\Program Files\dme\jboss\server\default\deploy\ejb3.deployer\META-INF\jboss-service.xml` with the equivalent file from the downloaded package.

### Step 4: Settings on the connectors

In order to make the DME connector communicate with the DME server, make the following change in the `wrapper.conf` file on all connectors:

```
C:\Program Files\dmeconnector\conf\wrapper.conf
```

Insert the following in `wrapper.conf` (two lines). Change the `wrapper.java.additional.3=` to a sequential number if 3 is already used. Also remember to change the password.

```
wrapper.java.additional.3=-Djavax.net.ssl.trustStore=".conf\dmessl.truststore"  
wrapper.java.additional.4=-Djavax.net.ssl.keyStorePassword=mypassword
```

### Step 5: Restart the DME server and the connectors

Secure communication between the DME server and DME connector(s) is now established.

---

## Configuration in the web interface

After installing the connector, open the DME server web interface to complete the configuration of the DME connector.

- 1 Click the **Connector** tab.
- 2 Click the connector you just installed (identified by the name you gave it during the installation).
- 3 Click **Functions**.

## Configuring DIIOP/Corba connector



In the **Functions** section of the **Connector** setup panel, go to the **Domino integration** group of settings.

### ➤ **General settings**

- 1 Click **Using Remote / Corba connection**.
- 2 Enter the port used by Corba (default is 63148).  
If you change this, remember that the new port must be opened in the firewall.
- 3 Specify if the selected port is a secure (SSL) port by selecting the **Secure (SSL)** field.  
Note that this requires that the the Domino administrator has installed the required SSL certificates as described in the Domino documentation.

### ➤ **Read marks**

- 1 In the **Read marks** fields, specify the server where the DME UnreadMark database is installed. This is required for e-mails in the client to be marked as read or unread.  
If you leave the **Server** field blank and specify a database in the **Database** field, DME looks for the database on the current user's mail server, and the database must therefore be installed on all mail servers accessed by DME users. The database location is specified relative to Domino's data directory.

### ➤ **Encryption**

- 1 In the **Notes encryption** fields, choose if you want to provide encryption by getting the users' ID files from iNotes or from users uploading their ID files to the server. A description of each method is provided in the *Domino integration guide*.
  1. Click **Get user ID files from iNotes** if you want the users to upload their ID files using the iNotes webmail interface. The users use the **Import Notes ID** function in the **Preferences > Security** page of iNotes to upload their Notes ID to the server, and DME picks it up from there.
  2. Click **Use ID storage database** if you want the users to upload their ID files to a special database provided by DME. If you choose this, you must provide the path to the IDStorage database.
- 2 In the **ID temp directory** field you can specify a path to the location where temporary copies of the ID files are stored when used for decrypting or encrypting e-mails, and by clicking **Shred ID files after use** you can add extra security by wiping the directory after a temporary ID file has been deleted, making it impossible to restore it afterwards.

## Configuring Notes session connector



In the **Functions** section of the **Connector** setup panel, go to the **Domino integration** group of settings.

➤ **General settings**

- 1 Click **Using Notes session**.
- 2 The **Notes ID password** field is now available. In this field, enter the password of the proxy user (DME\_Proxy) created by the Domino administrator. This user provides access to all DME users in Domino.

➤ **Read marks**

- 1 If you are running a Domino version less than 8, you must install the DME UnreadMark database as described for the DIIOP/Corba connector. See **Configuring DIIOP/Corba connector** on page 11. If you are connecting to Domino version 8 (or above), the **Read marks** fields must be left blank, otherwise the DME server will produce an error.

➤ **Encryption**

- 1 Encryption works in the same way for all types of Domino connectors. See **Configuring DIIOP/Corba connector** on page 11.

---

## Exchange NTLM setup

As of DME 3.6 SP1, there are two ways of using NTLM authentication on MS Exchange: *Java native NTLM support* and *Oakland NTLM*. This is configured on the connector.

By default, the DME connector is configured to use the native Java NTLM. However, in some cases you need to switch to using Oakland. For instance, you need to use Oakland for Exchange 2010 if NTLM SSP v2 is required (the local policy security setting **Network security: Minimum session security for NTLM SSP based (including RPC) servers** is set to Require NTLMv2 for the CAS server).

To switch to Oakland, open the file `wrapper.conf` on the DME connector. This file is located here:

**Windows:** `C:\Program Files\dmeconnector\conf\`

**Linux:** `/var/dme/instances/<connector-name>/connector/conf/`

Open the file, and change the additional parameter `oakland=false` to `oakland=true`.

Then restart the DME connector.

To determine whether your setup requires Oakland, check if the DME connector hangs when testing the connection to your Exchange server using NTLM as authentication.

This setting does not affect installations where **Basic** authentication is used.

---

## Upgrading the connector

To upgrade a connector, download the installer from the **DME install site** <http://install.excitor.dk>, and re-install the connector as outlined in this guide. Be sure to choose the Service Pack level that matches the DME server.

### Important


Make sure to upgrade the server before any connectors. If you upgrade the connectors first, remember to restart the connectors after upgrading the server.

---

When re-installing/upgrading, the original configuration and cryptokeystore files are not touched. However, see the next section for information about how to handle contact mapping files.

---

### Please note

You have to delete the old connector entry from the DME server by selecting it in the DME server web administration interface, and clicking . This is in order to free a connector license, making it available for the reinstalled connector.

---

If you have multiple connectors installed, repeat the upgrade process for each connector, until the Service Pack level of all connectors corresponds with that of the DME server.

---

### Important

If you run SSL between server and connectors, you will need to check the settings files after an upgrade. See **Securing the traffic between server and connector(s)** on page 9 for more information.

---

See also **Lotus Domino only** on page 6 for information about changing the mode on Domino connectors.

## Contact mapping files

With DME, you can create files for custom mapping of contact fields, for instance in order to always map a company short number to a specific field in the contacts application on the devices. You can even create mappings files that are specific to individual device types, brands, or operating systems. All this is described in separate documentation.

When you upgrade a connector, a backup is made of all the configuration files, including the custom mapping files, as described in the next section. In order to restore the custom mapping functionality, you must copy the custom mapping files back into the connector **conf** directory **before starting the connector**. This is very important, as the mapping of contact fields will be affected on the users' devices if this is not in place before you start the connector.

Sometimes, a new service pack will include changes in the mappings. This is for instance the case when upgrading to DME 3.5 Service Pack 1. In such cases, you need to manually merge the custom mappings into the new contact mapping file. The release notes for a given version or service pack will say if this is required.

## If the connector upgrade fails

If the upgrade of a DME connector should fail, for instance due to a power outage or similar, the installation is in an incomplete state. However, the installer always backs up essential files before upgrading.

Note that in order to prevent installation errors due to failed Internet connection during the installation, you have the option to use the *offline installer* instead. With the offline installer, you can include the connector installer in an installation package, which requires no Internet connection. See **Offline installation** in the "Installing DME on Windows" guide.

The following files are backed up before an upgrade:

- The database encryption key (`cryptoKeystore`)
- The connector configuration files (`dme-config.xml` and `wrapper.conf`)
- All `.xml` files in the `\conf` directory

The backup is placed in the folder of the Windows `TEMP` variable, and is called `DMEConnBackup_[timestamp]`.

To restore the backup, follow these steps:

- 1 Manually remove the DME Connector directory.
- 2 Perform a new, clean connector install as described in this guide.
- 3 Copy the following files from the backup created during the upgrade:  
`dmeconnector\conf\wrapper.conf`  
`dmeconnector\conf\dme-config.xml`  
`dmeconnector\conf\cryptoKeystore`
- 4 Start the connector.

---

### Please note

In rare cases, a file and/or directory can be locked by Windows, making it impossible to delete the connector directory. In such cases, you have to restart Windows to release the files in question, and continue with the deletion (or use a file unlock utility).

---