

A faint, light-grey BPM diagram is visible in the background. It consists of several circular nodes connected by arrows. Some nodes are double-circled, indicating start or end points. The flow starts from the top left, moves right, then down, then right again, and finally down to a bottom node. There are also some curved arrows and a feedback loop.

FUEGOBPM 5 MIGRATION GUIDE

MIGRATING FROM VERSION 5.1 TO VERSION 5.5

Pablo Victory
pvictory@fuego.com

April 20, 2005

Contents

1	Introduction	4
2	Before Starting	4
2.1	ANT Tool	6
2.1.1	ANT in Fuego 5.1	6
2.1.2	ANT in Fuego 5.5	7
3	Migration Overview	8
3.1	Migration Steps	8
4	Installing FuegoBPM 5.5	10
5	Exporting from FuegoBPM 5.1	11
5.1	ANT Script	11
5.2	Exporting Portal Admin Views	12
5.3	Exporting Projects	13
5.4	Exporting the Organization	14
5.5	Exporting Passwords	15
5.6	Exporting Instances	17
6	Importing into FuegoBPM 5.5	20
6.1	ANT Script	20
6.2	Building the Project	22
6.3	Importing the Organization	23
6.3.1	User Passwords	25
6.4	Publish and Deploy	26
6.5	Importing Views	27
6.6	Importing Instances	28

6.7 Starting the Engine	30
7 Conclusion	31
A Scripts	32
A.1 ANT Export Script	32
A.2 ANT Import Script	34

1 Introduction

The present document describes how to migrate from FuegoBPM version 5.1 SP7 to FuegoBPM version 5.5 SP5. If you are using an older version of Fuego and want to migrate to FuegoBPM 5.5 you will need to migrate first to version 5.1 SP7.

Migrating from one version of FuegoBPM to the other involves not only copying the existing organization, processes and instances information; but also, modifying this information in order to make it compatible with the new version.

2 Before Starting

Before the migration can be started certain elements should be gathered and analyzed. The following information should always be collected prior to any migration:

- Fuego version
- List of engines to migrate
- List of projects to migrate
- List of projects not to be migrated
- List of JSP to migrate for any Web Application (ie: Work Portal)
- Number of instances in each engine's database for each of the process versions to be migrated
- CAPI/PAPI programs and components
- Other external applications being used by FuegoBPM
- Test cases to be used to validate the migration

This information is needed to determine the scope of the migration effort. There can be several path for a succesfull migration:

- **Migrate only some processes.** Not every deployed process will be migrated into the new FuegoBPM 5.5 version. Moreover, for those processes that are migrated, no instances will be moved. This implies that any instance present in FuegoBPM version 5.1 will remain there and will ultimately be lost after the migration. This scenario usually implies that old instances will be pushed through the processes in the old version before or after the actual migration.
- **Migrate only some processes and some instances.** In this scenario the decision was made to migrate only some of the processes present in the old 5.1 version together with the instances for these processes. However, if you select to migrate the instances from a process, all the instances belonging to all the processes have to be migrated.
- **Migrate all processes and all the instances.** This option implies the migration of every deployed process and their corresponding instances.

Ultimately the appropriate path depends on each particular FuegoBPM installation and must be decided by the customer.

Note that FuegoBPM version 5.5 can be installed in the same machine as FuegoBPM version 5.1; however, a different installation path should be used.

Before starting the migration process make sure that no users are currently logged on to the old FuegoBPM version and that all engines are stopped. Also verify that all the FuegoBPM services (launcher, combridge) as well as all FuegoBPM applications (console, component manager, designer, etc) are stopped.

Fuego strongly recommends that *back up copies* of the following elements are made:

- **Deployed Projects:** Obtain a copy of all the deployed projects currently being used in FuegoBPM version 5.1. To get a copy of each deployed project go to Fuego Studio and from the 'File' menu choose the 'Export project' menu option.
- **Engine's Database** Perform a back up of each engine's database. These databases are defined by the database properties configured in the *WebConsole* tool for each engine. You should use your RDBMS specified backup or export tools to perform the actual backup. Consult your DBA on how to do this.

- **Directory Service** Perform a back up copy of the Directory Service being used. The directory is defined by the properties configured in the *Enterprise's Admin Center*. Check your Directory Service documentation to see how to perform the back up for your particular Directory Service.

2.1 ANT Tool

The migration procedure uses several ANT¹ tasks to export and import data. You should check that you have ANT available and your environment is properly configured.

2.1.1 ANT in Fuego 5.1

To set up the export environment you can create a .bat file in Windows containing the following code:

```
1 SET JAVA_HOME=C:\j2sdk1.4.2_03
2 SET CLASSPATH=%JAVA_HOME%\jre\lib\rt.jar;
3   %JAVA_HOME%\jre\lib\jce.jar;%JAVA_HOME%\jre\lib\jsse.jar
4 SET PATH=%JAVA_HOME%\jre\bin;%JAVA_HOME%\bin;%PATH%
5
6 SET PATH=%PATH%;C:\ant16\bin
7 SET ANT_HOME=C:\ant16
8 SET ANT_ARGS=-lib C:\fuego5.1\enterprise\antlib
```

This file assumes that you have installed ANT under the C:\ant16 directory. Also it assumes that you have installed FuegoBPM 5.1 also in the C:\dirve. FuegoBPM 5 requires ANT version 1.6 or greater.

Open a DOS prompt window and run the .bat file you have just created. This .bat file will configure your environment in order to run ANT commands to export your existing FuegoBPM 5.1 information.

For a complete description on how to use ANT in Fuego 5 you can check the *ANT Tool Integration* guide available through Fuego's download site.

¹For documentation about ANT see <http://ant.apache.org/>

2.1.2 ANT in Fuego 5.5

To set up the import environment you can create another .bat file in Windows containing the following code:

```
1 SET JAVA_HOME=C:\j2sdk1.4.2_03
2 SET CLASSPATH=%JAVA_HOME%\jre\lib\rt.jar;
3   %JAVA_HOME%\jre\lib\jce.jar;%JAVA_HOME%\jre\lib\jsse.jar
4 SET PATH=%JAVA_HOME%\jre\bin;%JAVA_HOME%\bin;%PATH%
5
6 SET PATH=%PATH%;E:\ant16\bin
7 SET ANT_HOME=E:\ant16
8 SET ANT_ARGS=-lib e:\fuego5.5\enterprise\lib;e:\fuego5.5\
   enterprise\ext;e:\fuego5.5\enterprise\ext\db2java.zip
```

Again this file assumes that you have installed ANT under the C:\ant16 directory. Also it assumes that you have installed FuegoBPM 5.5 also in the C:\ drive. FuegoBPM 5 requires ANT version 1.6 or greater.

Open a DOS prompt window and run the .bat file you have just created. This .bat file will configure your environment in order to run ANT commands to import your existing information into FuegoBPM 5.5.



Database Drivers: You may need to add your particular driver jar file to the last line of the .bat file. In the previous example the DB2 driver was added. Replace it with your particular driver.

3 Migration Overview

Migrating from FuegoBPM version 5.1 to FuegoBPM version 5.5 implies moving and verifying that all the information used by the old version of FuegoBPM is readily available in the new version. The organization information stored in the old FuegoBPM Directory will be moved to the Directory used by the new version. The projects present in the old FuegoBPM Directory will also be moved to the new version's own FuegoBPM Directory. This may imply modifying the existing Fuego Object's FBL code together with the processes code to bring it up to version 5.5 standards. Each engine present in FuegoBPM version 5.1 will be recreated in the new version and all the processes present in each engine will be published and deployed for the new version.

The migration effort involves first exporting all necessary information from the existing FuegoBPM 5.1 installation and databases, and then importing this information into FuegoBPM 5.5 through a series of scripts and ANT tasks.

At the end of the migration process you should find that FuegoBPM 5.5 is already configured with all the elements of your organization (roles, calendar rules, participants, etc.) present in the previous version. Also the new engines are ready to be started, with the FuegoBPM version 5.1 processes already present in their corresponding engines. Moreover, all the instances created in FuegoBPM version 5.1 are now ready to be used in FuegoBPM 5.5, each one preserving its state from the old version to the new one. This means that the instances will remain in the activities they were left prior to the migration process, ready to be processed or routed accordingly.

Below, we will describe the steps needed to successfully migrate a FuegoBPM 5.1 environment into FuegoBPM 5.5.

3.1 Migration Steps

The following is a checklist of migration steps which needs to be performed in order to migrate into FuegoBPM 5.5. Note that the steps have to be performed in the order here listed. You may skip a step if it does not apply to your particular installation, for example, if you do not have customized your portal's views then you do not need to export them from your old 5.1 environment.

1. **Install FuegoBPM 5.5:** Install and configure your new FuegoBPM 5.5 environment. See section [4](#).
2. **Exporting Views:** Export any customized portal views through the *WebConsole*. See section [5.2](#).
3. **Exporting Projects:** Export your deployed projects using *Fuego Studio*. See section [5.3](#).
4. **Exporting the Organization:** Export the organization information (participant, roles, etc) through the *WebConsole*. See section [5.4](#).
5. **Exporting Passwords:** Export the participant's passwords through an ANT task. See section [5.5](#).
6. **Exporting Instances:** Export the instances' information through an ANT task. See section [5.6](#).
7. **Building the Project:** Import your projects in *Fuego Studio* version 5.5 and export it again for publication. See section [6.2](#).
8. **Importing the Organization:** Import the organization information through the *WebConsole*. See section [6.3](#).
9. **Importing Passwords:** Import the participant's passwords through an ANT task. See section [6.3.1](#).
10. **Publish and Deploy:** Publish and deploy your projects through the *WebConsole*. See section [6.4](#).
11. **Importing Views:** Import any customized views you have through the *WebConsole*. See section [6.5](#).
12. **Importing Instances:** Import the process instances through an ANT script. See section [6.6](#).
13. **Starting the Engine:** Start your new environment. See section [6.7](#).



Staging Area: When exporting files it is a good idea to build some *staging area* where all these exported files can then be easily accessed for importing them into the new FuegoBPM 5.5 environment.

4 Installing FuegoBPM 5.5

Before you begin you should install and configure your new Fuego Enterprise 5.5 environment. Follow the installation instructions ² and create your Directory having in mind that the *organization name must be the same as the one used in your previous FuegoBPM 5.1 environment*.

After creating the directory you can start the *Web Application Server*. Log in to the *WebConsole* and create a new engine. You must create a new engine for each of the engines present in your old 5.1 environment. For each engine created you must create a new and separate database (do not use the old 5.1 engine's database). After configuring them, start each new engine to ensure that they are properly configured. Stop the new engines before moving to the next step.



Configuration Restrictions: Again, your only constraint when installing and configuring FuegoBPM 5.5 is to keep the same name used in the previous version for the Organization when creating the Directory.

²Check the *System Administrator Guide* for installation and configuration instructions

5 Exporting from FuegoBPM 5.1

5.1 ANT Script

To perform some of the export operations we will be using ANT scripts. These ANT scripts starts by defining a set of properties that will be used through out the scripts.³

```
1 <property name="migration.directory"  
2     value="${basedir}/migration" />  
3 <property name="migration.directoryid" value="fdi_new" />  
4 <property name="migration.participant" value="db2admin" />  
5 <property name="migration.password" value="password" />  
6 <property name="migration.directory.file"  
7     value="directory.properties" />  
8 <property name="migration.engineID" value="fuego_engine" />  
9 <property name="fuego.basedir" value="e:/fuego5.1/enterprise" />
```

- Line 1 sets the `migration.directory` property to be used through out the ANT script. This is the directory to be used as root for the migration. Every file exported will be placed under this directory (or a subdirectory of it). This will be our *staging area*.
- Line 3 sets the `migration.directoryid` property to be used through out the ANT script. This property identifies the directory which will be migrated. The value must correspond to an ID present in your `directory.properties` file.
- Line 4 sets the `migration.participant` property to be used through out the ANT script. The participant to be used to establish the connection to the directory. It should be the Fuego Administrator login used when connecting to the *WebConsole* in FuegoBPM 5.1.
- Line 5 sets the `migration.password` property to be used through out the ANT script. The password for the user declared in the previous property.
- Line 6 sets the `migration.directory.file` property to be used through out the ANT script. The path and file name of the *direc-*

³You can find the complete scripts at the end of this guide.

tory.properties file. This *directory.properties* file is the one located in \$ENTERPRISE_51/conf

- Line 8 sets the `migration.engineID` property to be used through out the ANT script. This is the name of the engine to be migrated.
- Line 9 sets the `fuego.basedir` property to be used through out the ANT script. This is the installation path for Fuego Enterprise 5.1.

The export script will then look like this:

```

1 <?xml version="1.0"?>
2 <project name="migration" default="exportAll" basedir="."
3     xmlns:fuego="antlib:fuego.tools.ant">
4
5     <property name="migration.directory"
6         value="${basedir}/migration"/>
7     <property name="migration.directoryid" value="db2"/>
8     <property name="migration.participant" value="db2admin"/>
9     <property name="migration.password" value="password"/>
10    <property name="migration.directory.file"
11        value="directory.properties"/>
12    <property name="migration.engineID" value="engine1"/>
13    <property name="fuego.basedir"
14        value="e:/fuego5.1/enterprise"/>
15
16
17    <!-- ===== -->
18    <!--           ANT Tasks           -->
19    <!-- ===== -->
20
21
22 </project>

```

We will be defining each task and adding them to this script in the following sections as needed.

5.2 Exporting Portal Admin Views

If you have created new views or modified the default portal views through the *Portal Admin* tool, you will need to export them in order to migrate them to the new FuegoBPM version. To export the views you will need to login to the *WebConsole* in version 5.1 and select the *Organizational Setting* node.



Figure 1: Select the **Organizational Settings** node from the WebConsole

From the *Export* tab select the radio button labeled *Views*.

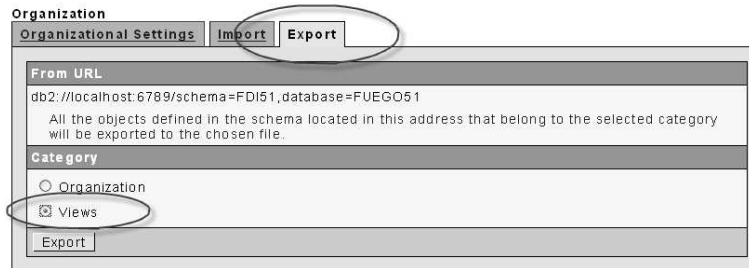


Figure 2: Select the **Export** tab to export the views

Press the **Export** button, this will generate a file with the views present in the *Portal Admin* for FuegoBPM version 5.1. Place this file in your staging area.

5.3 Exporting Projects

You will need to export the projects that are currently deployed in your production environment. From Studio 5.1 open your deployed projects and from the menu bar select '*File — Export Project*' menu option. You will be prompted for a filename for your exported project.

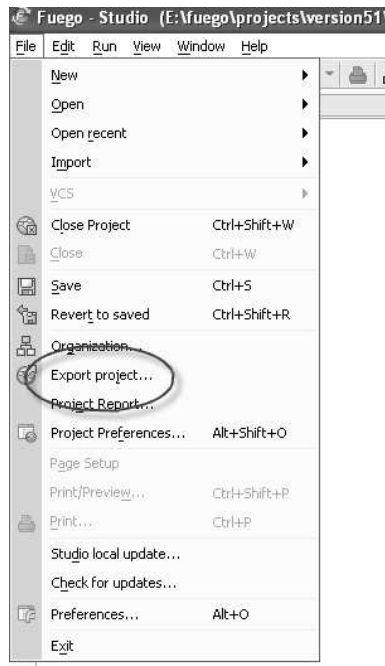


Figure 3: Export your production projects

Place your exported project files in your staging area. Clearly mark them as version 5.1 projects (you may wish to place them in a special folder).

5.4 Exporting the Organization

The next step in the migration process is exporting the information stored in the Directory Service (e.g. LDAP) including:

1. Participants
2. Roles
3. Calendar Rules
4. Holiday Rules
5. External Variables

To export the organization data you will need to login to the *WebConsole* in version 5.1 and select the *Organizational Setting* node.



Figure 4: Select the **Organizational Settings** node from the WebConsole

From the *Export* tab select the radio button labeled *Organization*.

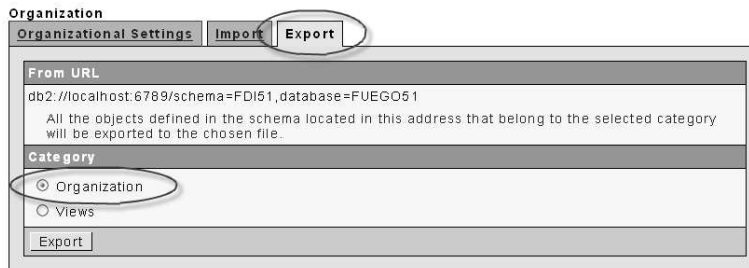


Figure 5: Select the **Export** tab to export the organization

Press the **Export** button, this will generate a file with the organization data present in FuegoBPM version 5.1. Place this file in your staging area.

5.5 Exporting Passwords

If you are using a relational database as your Directory provider, you can export the passwords that each user uses to login to FuegoBPM 5.1 and

migrate them to your new environment ⁴. You will, again, use an ANT task to export them into a file which can later be imported into FuegoBPM 5.5.

```

1 <target name="exportPasswords">
2   <fuego:exportpasswords
3     driverName="COM.ibm.db2.jdbc.net.DB2Driver"
4     user="${migration.participant}"
5     password="${migration.password}"
6     action="export"
7     file="${migration.directory}/organization/passwords.xml"
8     url="jdbc:db2://192.168.0.202:6789/FUEGO51"
9     schema="FDI51" />
10 </target>

```

You will need to provide the following parameters to the `<fuego:exportpasswords>` tag:

- **driverName:** The class name of the driver. The following table shows the driver names for the most common databases.

Database	Driver Name
Oracle	oracle.jdbc.driver.OracleDriver
MS SQL Server	com.inet.tds.TdsDriver
DB2	COM.ibm.db2.jdbc.net.DB2Driver

Table 1: JDBC driver names

- **user:** The username used to establish the connection to the database.
- **password:** The username's password.
- **action:** Action to be performed over the directory information. ⁵
- **file:** Location (path and filename) of the file to be either imported or exported.
- **url:** The directory's URL. This value must follow the JDBC URL format:

⁴If your directory provider is an LDAP and you are switching directory providers, then the existing user passwords can not be migrated.

⁵Available options are **import** or **export**.

Database	URL format
Oracle	jdbc:oracle:thin:@<host>:<port>:<sid>
MS SQL Server	jdbc:inetdae:<host>:<port>?database=<database>
DB2	jdbc.db2://<host>:<port>/<database>

Table 2: JDBC URL formats

- *schema*: The name of the database schema.

5.6 Exporting Instances

Finally the ANT script will also be used to export the in-flight instances belonging to any non-deprecated process present⁶.

In order to successfully execute the commands described below, you will *need to intall the migration tool*. This tool is a ajar file named *fuegoenginemigration.jar* that need to be placed in the \$ENTERPRISE/ext directory.

```

1 <target name="exportInstances">
2   <fuego:passport
3     id="root.passport"
4     directoryid="${migration.directoryid}"
5     participant="${migration.participant}"
6     password="${migration.password}"/>
7   <fuego:session
8     passportref="root.passport"
9     verbose="true"
10    properties="${migration.directory.file}" >
11     <fuego:instancesmanager
12       file="${migration.directory}/instances/export.xml"
13       action="export"
14       engineid="${migration.engineID}"/>
15   </fuego:session>
16 </target>

```

- Line 2 creates a passport to connect to the directory supplying some of the properties defined in section 6.1.

⁶This means that inflight instances in deprecated processes *are not going to be migrated*.

- Line 3 creates a session using the passport we created in the previous line.
- Line 4 exports the instances to the specified file.

Alternatively you could export each table in the engine database to its own file. This may be a better option if the number of instances is quite large.

```
1 <target name="exportInstances">
2   <fuego:passport
3     id="root.passport"
4     directoryid="${migration.directoryid}"
5     participant="${migration.participant}"
6     password="${migration.password}"/>
7   <fuego:session
8     passportref="root.passport"
9     verbose="true"
10    properties="${migration.directory.file}" >
11     <fuego:instancesmanager
12       file="${migration.directory}/instances"
13       onefile="false"
14       action="export"
15       engineid="${migration.engineID}"/>
16   </fuego:session>
17 </target>
```

- Lines 1-3 are the same as the above example.
- Line 4 exports the engine database tables each to a different file. The files being generated are:
 - PATTACHMENT
 - PATTACHPERMS
 - PEXCEPTION
 - PINSTREMARKS
 - PNOTIF
 - PPROCINSTANCE
 - PPROCINSTEVENT
 - PPROCRELATION

- PROCINSTITEM
- PTODOITEMS
- PWKTODO



Exported Tables Note that if any table is empty then a file will not be generated for it. This is normal.

6 Importing into FuegoBPM 5.5

Once we have all the necessary elements exported from our FuegoBPM 5.1 environment we can move into FuegoBPM 5.5.

6.1 ANT Script

To perform some of the import operations we will be again using ANT scripts. These ANT scripts starts by defining a set of properties that will be used through out the scripts.⁷

```
1 <property name="migration.directory"  
2     value="${basedir}/migration" />  
3 <property name="migration.target.directoryid" value="default" />  
4 <property name="migration.target.participant" value="db2admin" />  
5 <property name="migration.target.password" value="password" />  
6 <property name="migration.target.directory.file"  
7     value="target.directory.properties" />  
8 <property name="migration.target.engineID" value="engineOne" />  
9 <property name="fuego.basedir" value="E:\fuego5.5\enterprise" />
```

- Line 1 sets the `migration.directory` property to be used through out the ANT script. This is the directory to be used as root for the migration. Every file exported will be placed under this directory (or a subdirectory of it). This will be our *staging area*.
- Line 3 sets the `migration.target.directoryid` property to be used through out the ANT script. This property identifies the new directory. The value must correspond to an ID present in your FuegoBPM 5.5 `directory.properties` file. In our example we have renamed the FuegoBPM 5.5 `directory.properties` file to `target.directory.properties` (see line 5 below).
- Line 4 sets the `migration.target.participant` property to be used through out the ANT script. The participant to be used to establish the conneciton to the directory. It should be

⁷You can find the complete scripts at the end of this guide.

the Fuego Administrator login used when connecting to the *WebConsole* in FuegoBPM 5.5.

- Line 5 sets the `migration.target.password` property to be used through out the ANT script. The password for the user declared in the previous property.
- Line 6 sets the `migration.target.directory.file` property to be used through out the ANT script. The path and file name of the *directory.properties* file. This *directory.properties* file is the one located in `$ENTERPRISE_51/conf`
- Line 8 sets the `migration.target.engineID` property to be used through out the ANT script. This is the name of the new engine in FuegoBPM 5.5.
- Line 9 sets the `fuego.basedir` property to be used through out the ANT script. This is the installation path for Fuego Enterprise 5.5.

The import script will then look like this:

```

1  <?xml version="1.0"?>
2  <project name="migration" default="importInstances" basedir="."
3      xmlns:fuego="antlib:fuego.tools.ant.enterprise">
4
5      <property name="migration.directory"
6          value="${basedir}/migration"/>
7      <property name="migration.target.directoryid" value="default"/>
8      <property name="migration.target.participant" value="db2admin"/>
9      <property name="migration.target.password" value="password"/>
10     <property name="migration.target.directory.file"
11         value="target.directory.properties"/>
12     <property name="migration.target.engineID" value="engineOne"/>
13     <property name="fuego.basedir"
14         value="E:\fuego5.5\enterprise"/>
15
16
17     <!-- ===== -->
18     <!--           ANT Tasks           -->
19     <!-- ===== -->
20
21
22 </project>

```

We will be defining each task and adding them to this script in the following sections as needed.

6.2 Building the Project

The first step is to import the projects exported in section 5.3 into Studio version 5.5. You should check that the projects check properly and contain no migration errors. Once you verify that the project has no design or coding errors you should export them into the staging area.



Project Versions: The projects imported in this step will then be used to import the instances being migrated they MUST be the same version that the projects currently in use in your 5.1 production environment. If they are not then the correct migration of instance information can not be guaranteed.

From Studio 5.5 open your deployed projects and from the menu bar select 'File — Export Project' menu option. You will be prompted for a filename for your exported project.

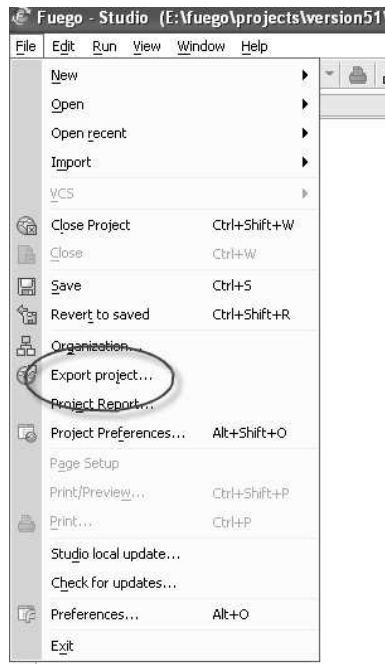


Figure 6: Export your production projects

Place your exported project files in your staging area. Clearly mark them as version 5.5 projects (you may wish to place them in a special folder).

6.3 Importing the Organization

The next step is to start the *WebConsole* from Fuego Enterprise and import the organization information. After login into the *WebConsole*, select the *Organizational Setting* option.

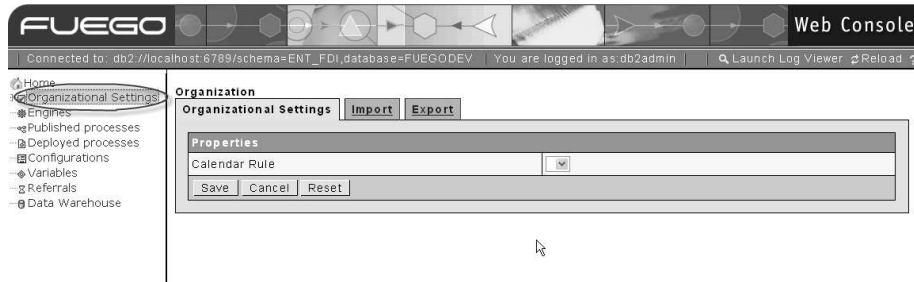


Figure 7: Select the Organizational Settings option

Two tabs will be displayed on the right hand-side panel. Select the one labeled *Import*. Make sure that the *Organization* radio button is selected. Then you will need to enter or select, through the *Browse* button, the file exported in section 5.4 containing the organizational information (should be located in your staging area).

After you select the file then import it by pressing the *Import* button.

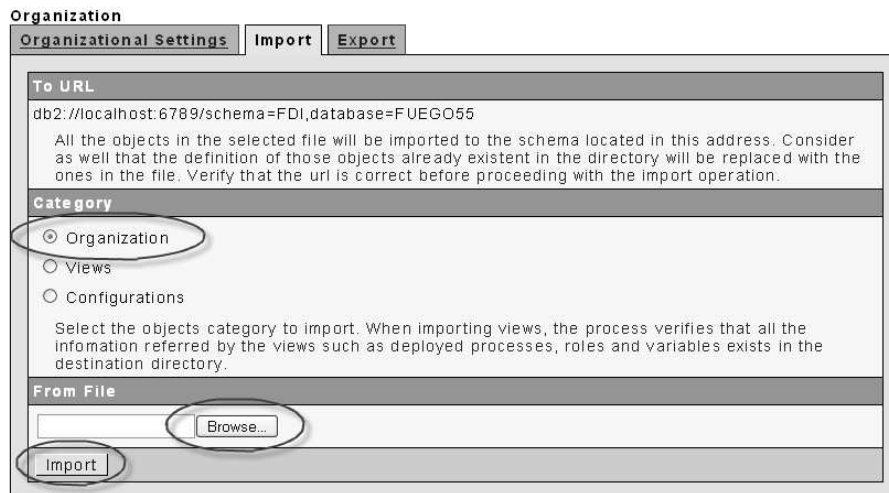


Figure 8: Select the appropriate file and click on the *Import* button

6.3.1 User Passwords

Note that passwords are not migrated with each participant. If your directory is an LDAP or if you are using a relational database and FuegoBPM is not handling the authentication then user passwords can not be migrated and need to be regenerated.

When importing the participants, FuegoBPM will reset each password and will generate a text file with these new passwords. This text file named `output.txt` will be placed in the `\$ENTERPRISE/bin` directory. The file has the following format:

```
pvictory:31882181:pvictory@migration.com
pbarletta:405577947:pbarletta@migration.com
```

Each participant id is followed by an 8 digit non-encrypted password separated by colons, followed by the e-mail of that participant again separated by colons. You can then email each participant the new passwords through a script.

If your directory information resides within a relational database and you let FuegoBPM handle user authentication, then you can import the existing password into your new installation⁸. FuegoBPM provides an ANT task that will allow you to import 5.1 passwords to a relational database.

You can import the password through the following ANT task⁹:

```
1 <target name="importPasswords">
2   <fuego:exportpasswords
3     driverName="COM.ibm.db2.jdbc.net.DB2Driver"
4     user="${migration.target.participant}"
5     password="${migration.target.password}"
6     action="import"
7     file="${migration.directory}/organization/passwords.xml"
8     url="jdbc:db2://192.168.0.202:6789/FUEGODEV"
9     schema="ent_fdi" />
10 </target>
```

You will need to provide the following parameters to the `<fuego:exportpasswords>`

⁸To export the existing passwords see section 5.5

⁹Note that even though the task is called *exportPasswords* it actually covers import and export operations

tag:

- **driverName:** The class name of the driver. The following table shows the driver names for the most common databases.

Database	Driver Name
Oracle	oracle.jdbc.driver.OracleDriver
MS SQL Server	com.inet.tds.TdsDriver
DB2	COM.ibm.db2.jdbc.net.DB2Driver

Table 3: JDBC driver names

- **user:** The username used to establish the connection to the database.
- **password:** The username's password.
- **action:** Action to be performed over the directory information.
- **file:** Location (path and filename) of the file to be either imported or exported.
- **url:** The directory's URL. This value must follow the JDBC URL format:

Database	URL format
Oracle	<code>jdbc:oracle:thin:@<host>:<port>:<sid></code>
MS SQL Server	<code>jdbc:inetdae:<host>:<port>?database=<database></code>
DB2	<code>jdbc:db2://<host>:<port>/<database></code>

Table 4: JDBC URL formats

- **schema:** The name of the database schema.

6.4 Publish and Deploy

You are ready to publish and deploy the project you built in section 6.2. When deploying the project, *do not select Smart Publish* since

this will generate any missing roles. We want to be certain that all the roles have been properly imported from the existing 5.1 environment.

Published projects > Publication source

Publication source

☒ Remote Project

☐ Exported Project

☐ Vcs
Provider

Publication Properties

☐ Smart publish
When this option is selected, all data defined in the project will be imported to the directory (including organizational units, roles, calendar rules, holiday rules, configurations, variables and views). The needed mappings for roles, configurations and variables will also be resolved automatically.

☐ Increment project version
The version of the project will be incremented to a major version number.

☐ Republish a previous version
You will be able to select the version for which you want to create a new revision. The project must be compatible with the last published revision of the selected version.

☐ Keep generated files
If you select this option, the java files generated during publication won't be deleted from the build directory of the project. This option is useful when debugging.

Deploy Properties

☐ Deploy processes after having published them
You will be able to configure where to deploy each process of this project, and then deploy them.

☐ Import Project's custom Views and Presentations after Deploy
Custom views and custom presentations will be imported to the directory service using the role and variable mappings defined in the next step. This operation only takes place when previous Deploy option is selected.

Figure 9: Do NOT select Smart Publish

6.5 Importing Views

The next step is to import the views exported in section 5.2. Again from the WebConsole, select the *Organizational Settings* option.

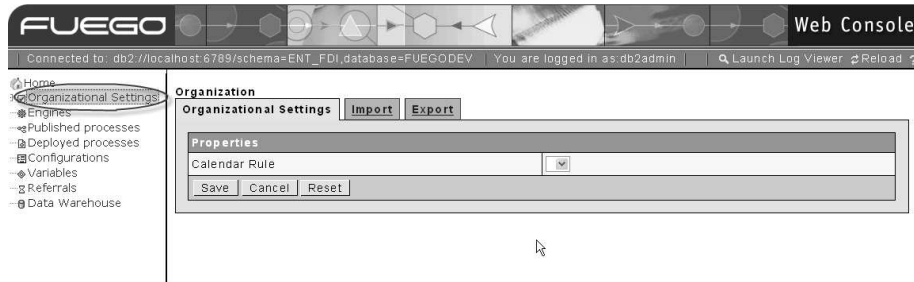


Figure 10: Select the Organizational Settings option

Two tabs will be displayed on the right hand-side panel. Select the one labeled *Import*. Make sure that the *Organization* radio button is selected. Then you will need to enter or select, through the *Browse* button, the file exported in section 5.4 containing the organizational information (should be located in your staging area).

After you select the file then import it by pressing the *Import* button.

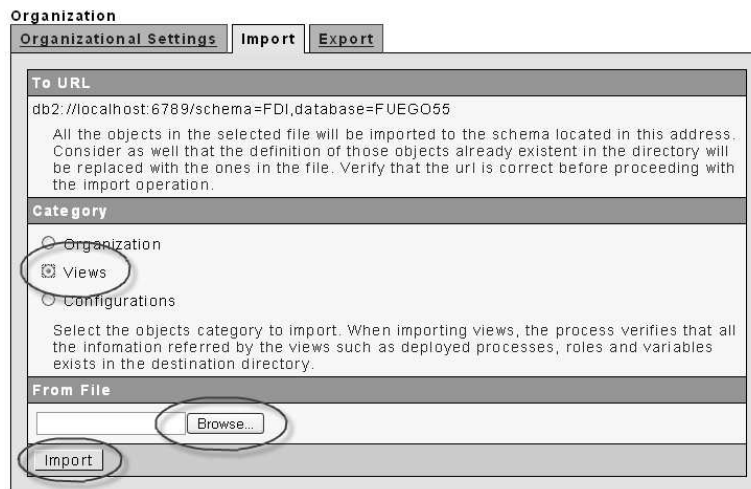


Figure 11: Select the appropriate file and click on the Import button

6.6 Importing Instances

The last step is to import the the in-flight instances belonging to any active processes present. We will again use ANT to import them but

this time we are going to use FuegoBPM 5.5 to do it. You will notice that some of the syntax has changed from the previous version and that we need to add some new properties.

```
1 <target name="importInstances">
2   <fuego:passport id="root.passport" directoryid="{migration.
      target.directoryid}" participant="{migration.target.
      participant}" password="{migration.target.password}"/>
3   <fuego:session passportref="root.passport" verbose="true"
      properties="{migration.target.directory.file}" >
4     <fuego:instancesmanager file="{migration.directory}/
      instances" onefile="false" action="import"
      engineid="{migration.target.engineID}"/>
5   </fuego:session>
6 </target>
```

- Line 2 creates a passport to connect to the new directory using the properties declared above.
- Line 3 creates a session using the passport we created in the previous line.
- Line 4 imports the instances from the specified directory into the new engine.

6.7 Starting the Engine

Once the instances have been imported you are ready to start the Fuego engine and the Work Portal. Log in to the portal and check that the instances are present in it.

7 Conclusion

Once the migration process is finished, FuegoBPM 5.5 should be configured with all the elements of your organization present in the previous version (roles, calendar rules, participants, etc.). The Fuego Web Portal is ready to be accessed by the same participants present in the old FuegoBPM version. Also the migrated engines are ready to be started, with all the previous processes already published, deployed and present in their corresponding engines. Moreover, all the instances are now ready to be processed, each one preserving its state and positioned in the activities they were left prior to the migration process.

A Scripts

The following appendix contains the complete scrips used through out this document. You can use this scripts as a starting point and adapt them for your particular environment.

A.1 ANT Export Script

```

1  <?xml version="1.0"?>
2  <project name="migration" default="prepare" basedir="." xmlns:
    fuego="antlib:fuego.tools.ant">
3
4      <property name="migration.directory" value="${basedir}/
        migration"/>
5      <property name="migration.directoryid" value="db2"/>
6      <property name="migration.participant" value="db2admin"/>
7      <property name="migration.password" value="password"/>
8      <property name="migration.directory.file" value="directory.
        properties"/>
9      <property name="migration.engineID" value="engine1"/>
10     <property name="fuego.basedir" value="e:/fuego5.1/enterprise"
        />
11
12
13
14     <!-- ===== -->
15     <!-- Prepares the build directory -->
16     <!-- ===== -->
17     <target name="prepare">
18         <mkdir dir="${migration.directory}"/>
19         <mkdir dir="${migration.directory}/views"/>
20         <mkdir dir="${migration.directory}/instances"/>
21         <mkdir dir="${migration.directory}/projects51"/>
22         <mkdir dir="${migration.directory}/projects55"/>
23         <mkdir dir="${migration.directory}/organization"/>
24     </target>
25
26
27     <!-- ===== -->
28     <!-- Cleans up generated stuff -->
29     <!-- ===== -->
30     <target name="clean" description="clean build files">
31         <delete includeemptydirs="true" failonerror="false" >
32             <fileset dir="${migration.directory}"/>
33         </delete>

```



```

34     </target>
35
36
37     <!-- ===== -->
38     <!--           Export tasks           -->
39     <!-- ===== -->
40
41
42     <!-- EXPORT INSTANCES -->
43     <target name="exportInstances">
44         <!-- we suppose that the schema is named after the
45             configuration -->
46         <fuego:passport id="root.passport" directoryid="{
47             migration.directoryid}" participant="{migration.
48                 participant}" password="{migration.password}"/>
49         <fuego:session passportref="root.passport" verbose="true"
50             properties="{migration.directory.file}" >
51             <!--instancesmanager file="{migration.directory}/
52                 instances/export.xml" action="export" engineid="{
53                     migration.engineID}"/-->
54             <fuego:instancesmanager file="{migration.directory}/
55                 instances" onefile="false" action="export"
56                 engineid="{migration.engineID}"/>
57         </fuego:session>
58     </target>
59
60     <!-- EXPORT PASSWORDS -->
61     <target name="exportPasswords">
62         <!-- DB2 -->
63         <fuego:exportpasswords driverName="COM.ibm.db2.jdbc.net.
64             DB2Driver" user="{migration.participant}" password="{
65                 migration.password}" action="export" file="{
66                     migration.directory}/organization/passwords.xml" url="
67                     jdbc:db2://localhost:6789/FUEGO51" schema="FDI51"/>
68         <!-- ORACLE
69         <fuego:exportpasswords driverName="oracle.jdbc.driver.
70             OracleDriver" user="{migration.participant}" password
71            ="{migration.password}" action="export" file="{
72                 migration.directory}/organization/passwords.xml" url="
73                 jdbc:oracle:thin:@localhost:1521:FUEGO" schema="FDI51"
74                 />
75         -->
76     </target>
77
78 </project>

```

A.2 ANT Import Script

```

1 <?xml version="1.0"?>
2 <project name="migration" default="importInstances" basedir="."
  xmlns:fuego="antlib:fuego.tools.ant.enterprise">
3
4   <property name="migration.directory" value="${basedir}/
     migration" />
5   <property name="migration.target.directoryid" value="default"
     />
6   <property name="migration.target.participant" value="db2admin"
     />
7   <property name="migration.target.password" value="password" />
8   <property name="migration.target.directory.file" value="target
     .directory.properties" />
9   <property name="migration.target.engineID" value="engineOne" />
10  <property name="fuego.basedir" value="E:\fuego5.5\enterprise"
     />
11
12  <!-- ===== -->
13  <!--          Import tasks          -->
14  <!-- ===== -->
15
16  <!-- IMPORT INSTANCES -->
17  <target name="importInstances">
18    <!-- we suppose that the schema is named after the
       configuration -->
19    <fuego:passport id="root.passport" directoryid="${
       migration.target.directoryid}" participant="${
       migration.target.participant}" password="${migration.
       target.password}" />
20    <fuego:session passportref="root.passport" verbose="true"
       properties="${migration.target.directory.file}" >
21      <fuego:instancesmanager file="${migration.directory}/
       instances" onefile="false" action="import"
       engineid="${migration.target.engineID}" />
22    </fuego:session>
23  </target>
24
25  <target name="importPasswords">
26    <!-- DB2 -->
27    <fuego:exportpasswords driverName="COM.ibm.db2.jdbc.net.
       DB2Driver" user="${migration.target.participant}"
       password="${migration.target.password}" action="import
       " file="${migration.directory}/organization/passwords.
       xml" url="jdbc:db2://localhost:6789/FUEGO55" schema="
       FDI" />
28    <!-- ORACLE

```

```
29      <fuego:exportpasswords driverName="oracle.jdbc.driver.  
        OracleDriver" user="${migration.target.participant}"  
        password="${migration.target.password}" action="import  
        " file="${migration.directory}/organization/passwords.  
        xml" url="jjdbc:oracle:thin:@localhost:1521:FUEGO"  
        schema="FDI55" />  
30      -->  
31    </target>  
32 </project>
```