

Configuring Adeptia Suite Backend Database on SQL Server

Adeptia Inc.
443 North Clark Ave,
Suite 350
Chicago, IL 60654, USA

Objective:

This document explains the steps to configure the Adeptia Suite backend database from HSQLDB to SQL Server.

1. If objects are present in the Adeptia Suite whose Backend Database is to be moved then you need to create a Offline zip using Migration Utility. Refer to the MIGRATION Section(i.e. Section 4) in the Migration utility User guide. You can download the Migration Utility User Guide from:
<http://www.adeptia.com/products/download/MigrationUtilityUserGuide.pdf>
2. Create a Database in the SQL Server. For Example **Adeptia_Backend**.
3. Go to `..\AdeptiaSuite\AdeptiaServer-5.2\ServerKernel\etc` folder and open **create-backend-sqlserver.sql**
4. Run the SQL query given in **create-backend-sqlserver.sql** on the database that you have created. This will create the required tables in the database that you have created.
5. From the same folder open **initialize-backend-sqlserver.sql**.
6. Run the SQL Query given in **initialize-backend-sqlserver.sql** on the database that you have created. This will apply the required indexes on the Backend tables.
7. Also run the **initialize-quartz-sqlserver.sql** on this Database.
8. Stop the Adeptia Suite Services if started.
9. Goto `..\AdeptiaSuite\AdeptiaServer-5.2\ServerKernel\etc` and open **server-configure.properties**.
10. In this file go to **Embedded Database Settings** category (see figure below).

```
<Category hide="true" name="Embedded Database Settings">
  <Property comment="Database names comma separated" dynamic="no" name="abpm.embedded.db.names">
    indigo_logs</Property>
  <Property comment="Database port" dynamic="no" name="abpm.embedded.indigo.port">2476</Property>
  <Property comment="Database Server memory configuration" dynamic="no" name="abpm.embedded.indigo.memory">-Xms128M -Xmx256M -Xrs</Property>
  <Property comment="Database port" dynamic="no" name="abpm.embedded.logs.port">2477</Property>
  <Property comment="Database Server memory configuration" dynamic="no" name="abpm.embedded.logs.memory">-Xms128M -Xmx256M -Xrs</Property>
</Category>
```

Remove indigo and logs from the value as shown in the above figure.

Note: logs would only be present if the log database is the embedded HSQLDB or else it would not be present

11. Then go to Systems Category Group in the server-configure.properties and change the properties according to the desired Database.

```

</Category>
<Category name="Systems">
  <Group hide="true" name="Database Configuration">
    <Property comment="JDO Database Driver Url" dynamic="no" name="abpm.jdo.driver">
      org.hsqldb.jdbcDriver</Property>
    <Property comment="JDO Database Connection Url" dynamic="no" name="abpm.jdo.url">
      jdbc:hsqldb:hsql://127.0.0.1:2476</Property>
    <Property comment="JDO Database UserName" dynamic="no" name="abpm.jdo.username">sa</Property>
    <Property comment="JDO Database Password" dynamic="no" encrypt="y" name="abpm.jdo.password">I99KL/1B3Ss=
    </Property>
    <Property comment="JDO Mapping File" dynamic="no" hide="true" name="abpm.jdo.mapping.file">
      mapping.xml </Property>
    <Property comment="JDO Database Name" dynamic="no" name="abpm.jdo.database.name"> indigo </Property>
    <Property comment="JDO Database Engine Name" dynamic="no" name="abpm.jdo.engine.name">hsql</Property>
    <Property comment="Backend Database Type" dynamic="no" name="abpm.backend.dbType">HSQL</Property>
  </Group>

```

Change the following properties.

JDO Database Driver Url	com.microsoft.sqlserver.jdbc.SQLServerDriver
JDO Database Connection Url	jdbc:sqlserver:// Database Server:Port ;DatabaseName= Database Name
JDO Database UserName	dynamic="no" name="abpm.jdo.username"> Username </Property> >
JDO Database Password	dynamic="no" encrypt=" no " name="abpm.jdo.password"> Password </Property>
JDO Database Engine Name	dynamic="no" name="abpm.jdo.engine.name"> sql-server </Property>
Backend Database Type	dynamic="no" name="abpm.backend.dbType"> SQL-Server </Property>

12. Goto Pagination group in the server-configure.properties and change the jdbc driver

```

</Group>
<Group name="Pagination">
  <Property comment="Enable or Disable Pagination" dynamic="yes" name="abpm.pagination.enable">yes</Property>
  <Property comment="Maximum number of Records that will displayed in a page" dynamic="yes"
name="abpm.pagination.page.size">10</Property>
  <Property comment="Pagination Database Handler Class" hide="true" name="abpm.pagination.db.handler">
org.hdpagination.dataaccess.jdbc.hsqldb.HSQLDBJdbcQueryTemplate</Property>
  <Property comment="Pagination Backend Database Handler Class" hide="true"
name="abpm.pagination.backenddb.handler">
org.hdpagination.dataaccess.jdbc.hsqldb.HSQLDBJdbcQueryTemplate</Property>
</Group>

```

Change the following properties

Pagination Backend Database Handler Class	org.hdpagination.dataaccess.jdbc.sqlserver.SQLServer2005JdbcQueryTemplate
---	---

13. Save the **server-configure.properties**.

14. Goto **quartz.properties** file in the same folder.

Under Configure JobStore make the following changes:

```

#=====
# Configure JobStore
#=====

org.quartz.jobStore.misfireThreshold = 60000
org.quartz.jobStore.maxMisfiresToHandleAtATime = 5

org.quartz.jobStore.class = org.quartz.impl.jdbcjobstore.JobStoreTX
org.quartz.jobStore.driverDelegateClass = org.quartz.impl.jdbcjobstore.StdJDBCDelegate
org.quartz.jobStore.useProperties = false
org.quartz.jobStore.dataSource = aBPM
org.quartz.jobStore.tablePrefix = ABPM_
org.quartz.jobStore.isClustered = false
org.quartz.jobStore.useDBLocks = false

```

org.quartz.jobStore.driverDelegateClass	org.quartz.impl.jdbcjobstore.MSSQLDelegate
---	--

Then under Configure Dataasource make the following changes.

```

#=====
# Configure Datasources
#=====
org.quartz.dataSource.aBPM.driver = org.hsqldb.jdbcDriver
org.quartz.dataSource.aBPM.URL = jdbc:hsqldb:file:embeddedDb/db/abpmquartz
org.quartz.dataSource.aBPM.user = sa
org.quartz.dataSource.aBPM.encryptPassword = yes
org.quartz.dataSource.aBPM.password = I99KL/1B3Ss=
org.quartz.dataSource.aBPM.maxConnections = 5
org.quartz.dataSource.aBPM.validationQuery=select lock_name from ABPM_LOCKS where lock_name = 'TRIGGER_ACCESS';

```

org.quartz.dataSource.aBPM.driver	com.microsoft.sqlserver.jdbc.SQLServerDriver
org.quartz.dataSource.aBPM.URL	jdbc:sqlserver:// Database Server:Port ;DatabaseName= Database Name
org.quartz.dataSource.aBPM.user	Username
org.quartz.dataSource.aBPM.encryptPassword	no
org.quartz.dataSource.aBPM.password	Password

15. Place the sqljdbc4.jar file in the following locations:
 - **..\AdeptiaSuite\AdeptiaServer-5.2\ServerKernel\ext**
 - **..\AdeptiaSuite\AdeptiaServer-5.2\ServerKernel\MigrationUtility**
16. Save the **quartz.properties** and Start the services.
17. Deploy the zip i.e. created in Step 1 by using Migration Utility(Refer to Section 4 in Migration utility User guide).
18. Login Adeptia Suite and populate few logs in the backend database by running few process flows.