

# Relational Population Service (RPS) White Paper

VERSION 2022

Jade Software Corporation Limited cannot accept any financial or other responsibilities that may be the result of your use of this information or software material, including direct, indirect, special or consequential damages, or loss of profits. There are no warranties extended or granted by this document or software material.

You should be very careful to ensure that the use of this software material and/or information complies with the laws, rules, and regulations of the jurisdictions with respect to which it is used. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Jade Software Corporation Limited.

The information contained herein is subject to change without notice. Revisions may be issued to advise of such changes and/or additions.

Copyright © 2023 Jade Software Corporation Limited.

All rights reserved.

JADE is a trademark of Jade Software Corporation Limited. All trade names referenced are the service mark, trademark, or registered trademark of the respective manufacturer.

For details about other licensing agreements for third-party products, you must read the Jade **Readme.txt** file.

# Contents

<b>Relational Population Service (RPS)</b> .....	<b>4</b>
Overview .....	4
Configuration .....	5
Components .....	5
Jade Primary Database Server .....	5
Relational Population Service (RPS) Node .....	5
Journal Transfer .....	6
SDS Administration Application .....	6
RPS Manager Application .....	6
Datapump Application .....	6
Relational Database Environment .....	6
Setup .....	6
Before You Start .....	6
Defining a Secondary RPS Database .....	7
Set Up RPS Folders .....	7
Define RPS Mapping .....	8
Create RPS Database .....	11
Defining Shortcuts and Setting Up the Initialization File .....	12
Set Specific Initialization File Settings .....	13
Create Shortcuts .....	15
Defining a Relational Database .....	16
Create a New ODBC Data Source .....	17
Create Relational Database .....	18
Operation .....	18
Starting RPS .....	19
Run the Primary Jade Database Components .....	19
Run the Jade RPS Database Components .....	19
SQL Server Initial Table Definition and Data Load .....	19
Enable the RPS Data Pump .....	21
Testing RPS .....	21
Look at Data in the Relational Database .....	21
Make a Change to the Primary Database .....	23
Check the Change is Replicated to the Relational Database .....	23
Schema Changes .....	24
Appendix .....	24
Useful References .....	24
Troubleshooting .....	24
jommsg.log .....	25
SDS and RPS Work Logs .....	25
Application Logs .....	25
Common Errors .....	25
Tips .....	25
Glossary .....	27

# Relational Population Service (RPS)

---

The Relational Population Service (RPS) provides automatic replication of objects from a primary Jade database to one or more relational databases.

RPS enables Jade systems to interoperate with relational databases for reporting, Business Intelligence (BI), Extract, Transform, Load (ETL), and data warehousing purposes.

For details about the Microsoft SQL Server ODBC drivers that must be used, see "[RPS Node Requirements](#)", in Chapter 1 of the *Installation and Configuration Guide*.

RPS enables Jade systems to automatically replicate data into external relational databases in near real-time. Developers can specify the classes, objects, and properties that are to be replicated to the required relational database or databases. This allows an entire Jade database to be replicated, or just the data that is required for external purposes.

RPS is built on Synchronized Database Service (SDS) technology and uses the same mechanisms as SDS for propagating updates to secondary databases. Updates recorded in database journals written by the primary database are mirrored to attached secondary databases and applied.

RPS can populate an entirely independent relational database, or tables within an existing relational database where that database can be updated from multiple sources (that is, via Jade's RPS and also from non-Jade systems). In either case, the tables updated by RPS are defined and controlled by Jade.

Additionally, an RPS node operating in **full replica mode** can also be used for disaster recovery, as it can acquire the primary database role in a negotiated or hostile takeover.

This white paper walks you through the steps required to set up an RPS environment. It uses Jade's Erewhon Investments demonstration application and Microsoft SQL Server Express (freely downloadable) for the relational database.

You need a license for the primary database node and a license for the RPS node. If you do not have these, send an email to [jadelicensing@jadeworld.com](mailto:jadelicensing@jadeworld.com) for consideration of licenses for evaluation purposes.

---

**Note** Jade Platform licenses are not transferred automatically between databases in an SDE. It is your responsibility to apply new licenses to any existing databases in an SDE. In addition, to ensure proper operation, you must apply the primary license to every secondary.

---

For details about future RPS enhancements, refer to the Jade Roadmap at:

<https://www.jadeworld.com/jade-platform/developer-centre/roadmap>

You can download other Jade Platform documentation referred to in this white paper from:

<https://www.jadeworld.com/jade-platform/developer-centre/learn/documentation>

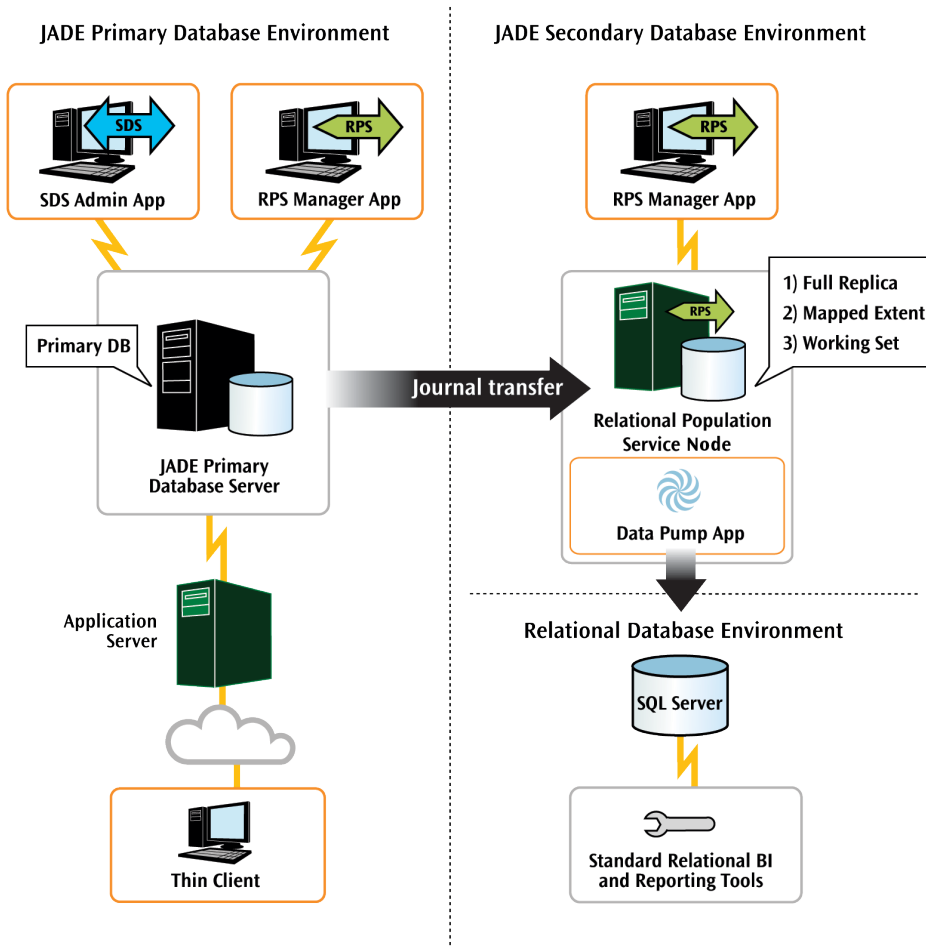
## Overview

This section provides an overview of the RPS:

- [Configuration](#)
- [Components](#)

## Configuration

The RPS configuration is shown in the following image.



## Components

The components shown in the previous image and described in the following subsections are required when using RPS. The setup of these components is described in "[Setup](#)", later in this document.

For more details about components, see the following subsections.

### Jade Primary Database Server

The Jade primary database server is the database server for the production Jade database.

### Relational Population Service (RPS) Node

An RPS node is an SDS secondary database configured with a **Relational** subrole. This specialized secondary database server can be referred to as an RPS secondary database node or just an RPS node, for short. It is convenient to refer to the persistent storage managed by an RPS node as an *RPS database*. An RPS database can be a full copy or a subset of the primary database, and it always includes a full set of user schema files. Configuring the extent of an RPS database is described in "[Create RPS Database](#)", later in this document.

The RPS database can be hosted on the same machine or on a different machine to the primary database. The examples in this document assume that the RPS database is hosted on the same machine as the primary.

An RPS node participates in a *Synchronized Database Environment (SDE)*. An SDE comprises a primary database server and one or more secondary database servers. A secondary database is either a native secondary or one configured with a relational subrole to make it RPS-capable. This document describes setting up a single RPS node.

## Journal Transfer

Objects in the RPS database are kept synchronized with objects in the primary by periodically applying audit records from journals written on the primary and mirrored to all attached secondary databases.

## SDS Administration Application

The SDS Administration application provides a facility for administering the primary and one or more secondary database servers in an SDE. For more details, see "[Using the SDS Administration Application](#)", in Chapter 1 of the *Synchronized Database Service (SDS) Administration Guide*.

## RPS Manager Application

Administration of the RPS node is done from the RPS Manager application. This application can be run from the primary or from RPS systems. For more details, see "[Managing an RPS Node](#)" in Chapter 2 of the *Synchronized Database Service (SDS) Administration Guide*.

## Datapump Application

The RPS **Datapump** application executes in the RPS node. It connects to the relational database (RDB) and propagates updates that were made on the primary as described by the RPS mapping.

## Relational Database Environment

The relational database environment is the target Relational Database Management System (RDBMS) database server.

For details about the supported Microsoft SQL Server versions that can be hosted on the same machine or a different machine to the primary and RPS database servers, see "[Requirements for RPS Nodes](#)", in Chapter 1 of the *Installation and Configuration Guide*.

The example in this document uses SQL Server Express and it assumes that the same machine hosts all database servers.

# Setup

This document uses the Erewhon system as an example to demonstrate the setup and running of an RPS environment. For more details, see the following subsections.

## Before You Start

The following must be set up before proceeding.

1. Jade Platform installed.

This is referred to as the *primary* database environment and these instructions assume that it is installed in the default location. For example, for Jade 7.1 the default installation folder is **C:\Jade71**.

---

**Note** Throughout this document, **<JadeVer>** is used to refer to the installation folder name. Whenever you see **<JadeVer>**, replace it with the name of your Jade Platform installation folder.

---

2. RPS licence key applied to the primary database.

This license is transferred to the RPS environment when it is created. If you do not have an RPS license key and you require one, send an email to [jadelicensing@jadeworld.com](mailto:jadelicensing@jadeworld.com).

3. Shortcuts defined for the primary database environment, including the following.

- Jade Database Server
- Jade Application Server
- Jade Presentation Client
- Jade RPS Manager
- Jade SDS Administration

All of these shortcuts should have been created as part of the Jade Platform installation.

4. Erewhon schemas loaded into the primary database. These schemas are available from the Jade Platform installation in **C:\<JadeVer>\examples\erewhon**.

Sample Jade schemas, including Erewhon (**JADE-Erewhon**), can be downloaded from the links at <https://github.com/jadesoftwarez>.

5. Erewhon data loaded.

To initialize Erewhon, follow the instructions in the **Erewhon.pdf** document.

6. Microsoft SQL Server Express installed. This is the relational database that is used. If you already have a full version of SQL Server installed, you can proceed using that version instead of SQL Server Express.
7. Microsoft SQL Server Management Studio Express installed. This is used to manage your SQL Server Express database.
8. Microsoft SQL Server Command Tool **sqlcmd** installed. This is used to run SQL scripts from the Jade RPS Manager.

Both Microsoft SQL Server Management Studio Express and Microsoft SQL Server Command Tool **sqlcmd** products can be downloaded and installed with default settings from Microsoft.

## Defining a Secondary RPS Database

This section describes the steps required to define a secondary RPS database.

### Set Up RPS Folders

Set up the folders and binary files to be used by the RPS environment, as follows.

1. Create a folder for the RPS environment.
  - Create **C:\<JadeVer>RPS**

This is the folder where the RPS binary and database files will physically reside.

---

**Note** This is creating a single folder with **RPS** appended to the **<JadeVer>** name. It is *not* creating an **RPS** subfolder within the **<JadeVer>** folder.

---

2. Copy the binary files to run the RPS environment from the primary to the RPS folder.

- Copy **C:\<JadeVer>\bin** to **C:\<JadeVer>RPS\bin**

3. Create a folder for the RPS database files.

- Create **C:\<JadeVer>RPS\system**

This is an empty folder that will contain the database files. The database files are created from the primary, as per the "[Create RPS Database](#)" section later in this document.

4. Create a folder for the SQL scripts.

- Create **C:\<JadeVer>RPS\RPSSQLScripts**

This is an empty folder that will contain the SQL Scripts used to define the tables in SQL Server. These scripts will be created later, in the "[Extract SQL Scripts](#)" section of this document.

## Define RPS Mapping

The RPS Mapping defines the classes and attributes to be used in the RDB and the tables and columns to which these will map. The table and column definitions will later be extracted and loaded into the RDB.

1. Bring up the Jade Platform development environment for the primary database.
  - a. Start the **Jade Database Server**.
  - b. Start the **Jade Application Server**.
  - c. Start the **Jade Presentation Client**.

The RPS Mapping is defined from the Jade Platform development environment.

2. Select the model schema.

- From the Schema Browser, click on **ErewhonInvestmentsModelSchema**.

The RPS Mapping is defined in the model schema because this schema contains all persistent class definitions.

3. To add or change the RPS mapping:

- a. Select **Browse | RPS Mappings**.
- b. Right-click on **ErewhonRPS** in the Relational Population Service Browser and then select **Change**.

**ErewhonRPS** was already defined in the **Erewhon** schema that you loaded. This definition can be customized as follows. Alternatively, you could define a new RPS mapping.

4. In the **Define RPS** sheet of the Relational Population Service wizard, the **Database Type** combo box value specifies the target RDB type (your RDB must be the specified type and version, or later) as well as the data type mappings that are used to convert Jade types to RDB types.

- Select the SQL Server version that you require.

5. Set the **Per Statement Exception Policy** create, update, and delete values.

- Leave the **Halt** default value setting for each of these.



If an exception occurs when creating, updating, or deleting a row in the RDB, the **Halt** value causes the transaction to be aborted and database tracking to cease.

6. Set the **Logging Options** value.

- Leave the **Default** value setting.

Exceptions are logged to the **jommsg.log** file by default. If selected, the **RPSTable** value also records exception information in a table in the RDB.

7. Set the **OID Mapping Options** (where OID stands for *Object Identifier*) value.

- Select the **Split Into Columns Int/BigInt** value setting, which splits OIDs into two integer values containing the class number and instance identifier whilst accommodating both large class numbers and instance identifiers.

If selected, the **Map to String (7.1 format)** value maps to a 28-character fixed-length string and the **Split Into Columns Int/Int** value splits OIDs into two integer values containing the class number and instance identifier.

---

**Note** The **Split Into Columns Int/Int** format may limit class number and instance identifier values.

---

8. Set the Default Database Name.

- Enter **Erewhon**.

This is the name of the target RDB, which will be used when the RDB is set up, described under "[Create Relational Database](#)", later in this document.

9. Set the **Default Connection String** value. Do *not* enclose the string in quotes.

- Enter **DSN=SQLServerODBC; Database=Erewhon**.

This is the connection string, including the ODBC DSN name, for the target RDB. The ODBC definition will be set up later, described under "[Create a New ODBC Data Source](#)", later in this document. This is required for the RPS database to connect to the SQL Server database.

10. Set the **Default Connection Usercode** and **Default Connection Password** values.

- Leave these values blank.

If specified, this user code and password are used on the RPS node to connect to the RDB. They can also be overridden from the RPS node. They are left blank here, as later steps specify that Windows Authentication will be used to connect to the RDB.

11. Set the **Select Top Most Schema** value.

- Leave the default **ErewhonInvestmentsModelSchema** value setting.

All persistent classes in the **Erewhon** system are defined in the model schema, so this is the highest-level schema required for the mapping.

12. Set the **Show Virtual Properties** and **Show Methods** values.

- Leave the default **Do Not Show Virtual Properties** and **Do Not Show Methods** values.

If you *do* want to display virtual properties and methods, select the **Show Virtual Properties** and **Show Methods** values, respectively.

13. Set the **Show Many To Many Properties** value.
  - Leave the default **Do Not Show Many To Many Properties** setting if you do not want to map many-to-many inverse properties.

If you *do* want to select the collections to be mapped to junction tables, select the **Show Many To Many Properties** value.

14. Go to the **Select Classes for RPS** sheet of the wizard, by clicking **Next >**.

15. Select the classes that are to be mapped to tables in the RDB.

- No action is required, as classes are already selected.

The selected classes are listed in the **Selected Classes** list box. Further classes can be added from the **Available Classes** list box or removed from the **Selected Classes** list box. Only user-defined classes that can have persistent instances are available for inclusion in the RPS mapping.

16. Go to the **Map Classes to Tables** sheet of the wizard, by clicking **Next >**.

17. Map the selected classes to the tables required in the RDB.

- No action is required, as classes are already mapped.

You can change the mapped table name for each class, by entering a new name in the relevant **Table** column cell. The name must be valid in the RDB, with a maximum length of 100 characters.

You can select the:

- **Excluded** value if you want to exclude the table from RPS replication but retain the table in the RPS mapping definition.
- **Output Callback** value if you want output for a table to be passed to your application before being replicated to the RDB. To avoid unnecessary callback processing, you should select the **Output Callback** value only if required.

In the **Update Mode** columns, when:

- In **Historical Mode**, object updates and deletes are mapped to a new row in the target table.
- Not in historical mode, an object update or delete will update or delete the corresponding row.
- The **No Delete** value is selected, object deletes are not propagated to the target RDB.

The existing definition has mapped subclasses to the same table as their superclass. This was done from the **Map SubClasses to this Class Map** right-click menu item.

18. Go to the **Select Columns for Tables** sheet of the wizard, by clicking **Next >**.

19. Select the features that are to be mapped to columns in the RDB.

- No action is required, as properties are already selected.

The properties for the selected table/class are listed in the **Features** list box. You can add or remove properties, by clicking the relevant check boxes.

20. Go to the **RPS Column Mapping** sheet of the wizard, by clicking **Next >**.

21. Map the selected attributes to the columns required in the RDB.

- No action is required, as attributes are already mapped.

You can change the mapped column name, mapping type, ID, and key for each feature, by entering a new value in the relevant row and column. To see the features for one table at a time, select the **View Selected Table** value. By default, the primary key for all tables is the OID.

22. Go to the **Summary and Build** sheet of the wizard, by clicking **Next >**.
23. Check the details and finalize the RPS mapping.
  - Check that the details listed are correct and then click **Finish**.

This completes the RPS mapping changes.

24. Reorganize the database, if required.
  - a. Select the **ErewhonInvestmentsModelSchema** schema.

If a reorganization is required, the Schema Browser window will display two **ErewhonInvestmentsModelSchema** items. One of the items will have a highlighted background – this is the item that you should select.

- b. When the reorganization has completed, click **OK**.

## Create RPS Database

The secondary RPS database files are produced from the primary database in the folder specified in the "Set Up RPS Folders" section, earlier in this document. (The **JadeDatabaseAdmin** class **commitCoherentBackup** method, which can be called only from a secondary, enables you to create a backup of a secondary that when recovered, stops tracking and is in a commit-coherent state. This provides an SDS secondary native backup that you can use to create (re-clone) an SDS secondary RPS node, and replicates the creation of an RPS database on the primary server.)

---

**Note** Make sure that you have applied your RPS license key to the primary database before proceeding. You can use **C:\<JadeVer>\bin\jadreg.exe** to do this.

---

1. Start the RPS Manager application for the primary database.
  - a. Start the **Jade Database Server**, if it is not already running.
  - b. Start the **Jade RPS Manager**.

The secondary RPS database is created from the RPS Manager application.

2. Select the RPS Mapping schema.
  - a. Select **ErewhonInvestmentsModelSchema**.
  - b. Select **ErewhonRPS**.
  - c. Click **OK**.

This specifies the mapping that was defined earlier. This mapping is used to create the RPS database.

---

**Note** This screen appears only if this is the first time the RPS Manager is run; otherwise the schema and mapping can be selected in the next step, if required.

---

3. Create a full RPS database.
  - Select **RPS | Create RPS Database | Full**.

The **Full** value replicates all persistent system and user schema database files from the primary database.

The **Mapped Extent** value replicates user schema files and all user database files that contain classes included in the RPS mapping defined in the primary database.

The **Working Set** mode is deprecated in Jade 2020 and higher.

The RPS Database Creation dialog is displayed.

4. Specify the RPS directory.

- Enter **C:\<JadeVer>RPS\system** or use the browse button to select this directory.

This specifies the folder where the secondary RPS database files are created. This is the same folder created in the "[Set Up RPS Folders](#)" section, earlier in this document.

5. Set the copy mode.

- Leave the default **Quiesced – Read Only** value selected, to place the primary database in quiescent read-only state while the data is copied. For more details, see "[Online Quiesced Backup](#)" in Chapter 3 of the *Database Administration Guide*.

The **Online – Updating Allowed** value indicates that updates to the primary database are allowed while the data is copied. For more details, see "[Online Full Backup](#)" in Chapter 3 of the *Database Administration Guide*.

6. Set the **Overwrite Existing Files** value.

- Leave this as the default unchecked value.

Checking this value would allow any existing files in the specified destination RPS folder to be overwritten when the data is copied from the primary database. There should currently be no files in this folder, so the option is not selected.

7. Start the database creation.

- Click **OK**.

Progress messages are displayed and when finished, an RPS Database Create Completed dialog is shown.

8. Finish the database creation.

- Click **OK** on the RPS Database Create Completed dialog.

You are returned to the RPS Manager application window, with the following progress messages displayed.

```
> RPS 'Full' Database create started.  
> RPS Database create completed.
```

---

**Note** You may see a message saying that files have not been copied. You can safely ignore this.

---

The secondary RPS database has now been created in **C:\<JadeVer>RPS\system**.

## Defining Shortcuts and Setting Up the Initialization File

Shortcuts and initialization (**ini**) files to be used by the primary and secondary databases must be set up to define where these are run. The secondary database is the database for the RPS environment.

For more details, see the following subsections.

## Set Specific Initialization File Settings

You can use the same initialization (**ini**) file for both the primary and secondary databases. Our examples assume that the same **ini** file is used.

When a common **ini** file is used, separate sections must be defined to specify different values required for the primary and secondary databases. These sections are distinguished by prefixing the section name with a *unique-id*. The format of this is [*unique-id.section-name*].

The *unique-id* value must match that specified in the shortcuts defined in ["Create Shortcuts"](#) section, later in this document. In this section, **RPSServer** has been used as the *unique-id* value to define this as applying to the secondary database only and **Primary** has been used as the *unique-id* value to define this as applying to the primary database only. The *unique-id* value must also be specified in the command line for the Jade program that is to use it. For examples of the required command line parameter, see ["Create Shortcuts"](#).

If you use separate **ini** files for the primary and secondary databases, the *unique-id* prefix is not required. See the [Initialization File Reference](#) for more details about individual **ini** file settings.

1. Close the Jade database.
  - a. Close the Jade RPS Manager.
  - b. Close the Jade presentation client.
  - c. Close the Jade application server.
  - d. Close the Jade database server.

The database is closed so that when it is next run, the following **ini** file changes will take effect.

2. Create a copy of the **ini** file.
  - Copy **C:\<JadeVer>\System\jade.ini** as **C:\<JadeVer>\jadeRPS.ini**.

This creates a copy of the default primary database **ini** file so that changes can be made for use by both the primary and secondary databases.

3. Edit the **ini** file.
  - Open the **C:\<JadeVer>\jadeRPS.ini** file in an editor.

The new **ini** file can now be updated with unique sections used by the secondary database.

4. Specify the server details for the secondary database.
  - Add the following section.

```
[RPSServer.JadeServer]
NodeName=RPSServer
NetworkSpecification1=TcpIp,Enabled,6007
```

This defines the parameters used to run the secondary Jade server node.

5. Specify the client details for the secondary database.
  - Add the following section.

```
[RPSServer.JadeClient]
ServerNodeSpecifications=TcpIp,LocalHost,6007
```

This defines the parameter used to run the secondary Jade client node.

6. Specify SDS (that is, secondary) details for the primary database.

- Add the following section.

```
[Primary.SyncDbService]
DatabaseRole=PrimaryRole
MyName=Primary_SiteA
```

This section defines the SDS parameters used by the primary database server. An RPS node is a special kind of SDS secondary server and therefore the same SDS **ini** file options are used. Note that the **MyName** parameter must match the extension used in the [\[ConnectionParams\]](#) section defined later in this document.

7. Specify the SDS parameters for the secondary RPS database.

- Add the following section.

```
[RPSServer.SyncDbService]
MyName=RPSServer
DatabaseRole=SecondaryRole
PrimaryServerName=Primary_SiteA
DatabaseSubrole=RelationalRole
```

This section defines the parameters used for the SDS functions in the secondary RPS database. Note that the **PrimaryServerName** parameter must match the extension used in the [\[ConnectionParams\]](#) section defined later in this document.

8. Specify the connection details for the primary database.

- Add the following section.

```
[ConnectionParams.Primary_SiteA]
NetworkSpecification1 = TcpIp,Enabled,60001
ServerNodeSpecifications = TcpIp,127.0.0.1,60001
RestrictedHostAccess=false
```

These parameters are used to establish connections between the primary and RPS nodes. **Primary\_SiteA** specified after **ConnectionParams** in the section heading must match the **MyName** parameter in the [\[SyncDbService\]](#) section of step 6 of this instruction.

9. Specify the location for the secondary to which log files are output, if required.

- Add the following section with your required parameter values.

```
[RPSServer.JadeLog]
LogDirectory = r:\jade\Rps\MessageLogs
```

If you want to output RPS messages to a directory or file other than that relative to the Jade Home directory (that is, **JadeLogs**), parameters in the [\[JadeLog\]](#) section of the Jade initialization file enable you to do this.

10. Update primary database shortcuts to specify the new **ini** file.

- Add **ini=C:<JadeVer>jadeRPS.ini** to the command line for all shortcuts (if **ini=** is present already, replace it).
  - Jade Database Server
  - Jade Application Server
  - Jade Presentation Client

- Jade RPS Manager
- Jade SDS Administration

When next run, these programs use the newly created **ini** file, therefore all programs run from the primary and secondary database use the same **ini** file.

11. Update the primary Jade database server to use the primary **ini** file sections.
  - Add **name=Primary** to the command line for the **Jade Database Server** shortcut

When next run, the primary database server uses the [Primary.SyncDbService] **ini** file section that was defined above, instead of the default [SyncDbService] section.

## Create Shortcuts

It is assumed that the shortcuts to run the Jade database server, Jade application server, Jade presentation client, Jade RPS Manager, and Jade SDS Administration are already defined as specified in the "Before You Start" section, earlier in this document.

Define new shortcuts for the following.

- Primary database Erewhon **Administration** application.

- a. Enter the **Target** value of:

```
C:\<JadeVer>\bin\jade.exe appServer=localhost appServerPort=6006
ini=C:\<JadeVer>\jadeRPS.ini schema=ErewhonInvestmentsViewSchema
app=Administration name=Primary
```

- b. Enter the **Start In** value of:

```
C:\<JadeVer>\bin
```

- c. Enter the **Name** value of:

```
Erewhon Admin
```

This shortcut runs the Erewhon Administration application against the primary database as a presentation client.

---

**Note** The **appServerPort** number must be the same as that specified on the Jade application server shortcut. If the Jade application server is on another machine, **appServer=** must specify the relevant DNS name.

Although the port number does not have to be that of the target specified in the example in this topic, it must be unique.

---

- Secondary RPS database server.

- a. Enter the **Target** value of:

```
C:\<JadeVer>RPS\bin\jadrap.exe path=C:\<JadeVer>RPS\System
ini=C:\<JadeVer>\jadeRPS.ini name=RPSServer
```

- b. Enter the **Start In** value of:

```
C:\<JadeVer>RPS\bin
```

- c. Enter the **Name** value of:

```
RPS - Jade Database Server
```

This shortcut runs the Jade database server for the secondary RPS database. Note the **name=** parameter specified in the target. This value is the same as the *unique-id* specified in the **ini** file and it indicates the **ini** file section to be used.

- Secondary database RPS Manager application.

- a. Enter the **Target** value of:

```
C:\<JadeVer>RPS\bin\jade.exe path=C:\<JadeVer>RPS\System server=multiUser
ini=C:\<JadeVer>\jadeRPS.ini schema=JadeMonitorSchema app=RPSManager
name=RPSServer
```

- b. Enter the **Start In** value of:

```
C:\<JadeVer>RPS\bin
```

- c. Enter the **Name** value of:

```
RPS - RPS Manager
```

This shortcut runs the RPS Manager application against the secondary RPS database as a standard client. Note that the **name=** parameter is also specified in the target for this application, which must be the same as the *unique-id* specified in the **ini** file.

- Specify the location of the journal directory for the secondary, if required.
  - Add the following Jade initialization file section with your required parameter values:

```
[RPSServer.PersistentDb]
JournalRootDirectory = drive:\jade\Rps\system\journals
```

## Defining a Relational Database

The RDB used in this document is SQL Server 2005 Express or later release. An ODBC data source must be defined for this and the initial tables, columns, and data must be extracted and loaded from the primary database.

The instructions in the following subsections demonstrate how to do this.



## Create a New ODBC Data Source

This defines a DSN for the SQL Server Express ODBC driver. The RPS secondary Jade database will use this to connect to the SQL Server database.

1. Start the ODBC Data Source Administrator.
  - a. Bring up the **Data Sources (ODBC)** program in the Windows Administrative Tools.
  - b. Select the **User DSN** sheet.

This opens the dialog to allow a new ODBC data source to be added to your machine.

2. Specify the data source.
  - a. Click **Add...**
  - b. Select the correct SQL Native Client driver version for your version of SQL Server.
  - c. Click **Finish**.

The data source will be SQL Server. This is the only RDB supported by RPS.

3. Specify SQL Server details.
  - a. Enter Name of **SQLServerODBC**.
  - b. Enter Description of **SQL Server**.
  - c. Enter Server of **machine-name\SQLExpress**.
  - d. Click **Next >**.

---

**Note** The name must match the name entered in the **Default Connection String** in the "[Define RPS Mapping](#)" section. If SQL Server is running locally, the server name must be specified *without* leading backslashes (\\). Include the backslashes if a DNS name is used.

---

4. Specify the login details.
  - a. Leave the default value of **With Integrated Windows authentication**.
  - b. Click **Next >**.

This defines the login details used when connecting to SQL Server. The Windows login id is used. When **Next >** is clicked, an attempt is made to connect to SQL Server. If this is unsuccessful, an error is displayed and the next dialog is not shown.

If an error occurs, check that the server name has been entered correctly in the first configuration dialog and that SQL Server is installed and running. If you are still encountering problems, run SQL Server Management Studio. Note the **Server Name** on the logon window, as this should be the same as the server name entered in step 3 of this instruction. Check that you can log on to Management Studio using Windows Authentication. If this logon fails, this problem must be resolved before you can connect using the ODBC data source.

5. Complete the DSN set up to meet your requirements. In general, default values should be sufficient; for example, the third sheet of the **System DSN** wizard has the **Use ANSI quoted identifiers** check box checked by default, which enables RPS to interrogate the RPS ODBC data source that it uses to determine if to quote SQL identifiers (that is, table, column, and procedure names).

6. Finish defining the data source.
  - a. Click **Next >**.
  - b. Click **Finish**.
7. Test the data source.
  - Click **Test Data Source**.

This attempts to connect to SQL Server using the specified configuration. If this fails, check that the server name has been entered correctly in the first configuration dialog and that SQL Server is installed and running. If the test is successful, an SQL Server ODBC Data Source Test dialog is displayed.

8. Exit from the ODBC Data Source Administrator.

**SQLServerODBC** is now defined as an ODBC data source.

## Create Relational Database

Create a new RDB from SQL Server, as follows.

1. Start the SQL Server Manager.
  - From the Start menu, run **SQL Server Management Studio**.

This application is used to manage the SQL Server database.

2. Select to add a new database.
  - a. Select **Databases** in the Object Explorer.
  - b. Right-click and select **New Database**.

The New Database dialog will be shown, to enable the definition of a new database.

3. Add new database.
  - a. Enter a database name of **Erewhon**.
  - b. Click **OK**.

The new database should be displayed in the Object Explorer under **Databases**. The database name *must* be the same as that specified in the **Connection String** of the "[Define RPS Mapping](#)" section. When adding this database, modify any details you require for your use or change its properties after the database is added.

## Operation

This section describes:

- [Starting RPS](#)
- [Testing RPS](#)
- [Schema Changes](#)

## Starting RPS

All components should now be installed and defined. Start the following (if they are not already running) to see RPS in action.

- [Run the Primary Jade Database Components](#)
- [Run the Jade RPS Database Components](#)
- [SQL Server Initial Table Definition and Data Load](#)
- [Enable the RPS Data Pump](#)

## Run the Primary Jade Database Components

1. Start the primary database server.
  - Start the **Jade Database Server** (if it is not already running).

The primary database server is started to enable the connection to the RPS database and so that changes can be made and propagated to the SQL Server database.

2. Start the SDS Administration application.
  - Start the **Jade SDS Administration** application.

The SDS Administration application is used to view the connection between the primary and RPS servers and to help diagnose any problems that might occur. Note that the primary database should currently be listed in this window.

## Run the Jade RPS Database Components

1. Start the RPS database server.
  - Start the **RPS – Jade Database Server** (if it is not already running).

The RPS database server is started so that changes made to the primary database can be processed and propagated to the SQL Server database.

The RPS database is now displayed in the Jade SDS Administration window and is highlighted with an orange background. This indicates that database tracking is halted. Tracking is halted because we have not yet started the data pump.

2. Start the RPS Manager application.
  - Start the **RPS - RPS Manager** (if it is not already running).

The RPS Manager is used to start the data pump, later in following section.

## SQL Server Initial Table Definition and Data Load

From the RPS - RPS Manager application, the following steps create SQL scripts to create the tables and columns required in SQL Server. These tables and columns are those that were defined earlier in the RPS Mapping. Files are also created containing the initial data to be loaded into these tables and the data is loaded.

1. Select to configure the RPS node.
  - Select the menu item **File | RPS Node Configuration**.

The Configure RPS Node dialog is then displayed. (The details in steps a through f of this step should be pre-filled and match those details entered during the "[Define RPS Mapping](#)" section.)

2. Check the configuration details.
  - a. Schema = **ErewhonInvestmentsModelSchema**.
  - b. Mapping = **ErewhonRPS**.
  - c. Jade Database Mode = **Full**.
  - d. Relational Database Name = **Erewhon**.
  - e. Connection String = **DSN=SQLServerODBC; Database=Erewhon**.
  - f. Set the **Auto Start Datapump** option.

With this option selected, the data pump is started automatically when the RPS node is started. It also allows operations such as database reorganizations to be performed without user intervention. (The data pump needs to be started after the load as it failed to run at start up because the SQL server database did not exist.)

Note that when **Auto Start Datapump** is enabled, the **Ask for RDB Login** option must be unchecked.

- g. Check **Auto Execute Drop/Reload** in the Data Pump Restart Options.
- h. Enter **C:\<JadeVer>RPS\RPSSQLScripts** as the **Alter Script Path**.
- i. Check **Use Sqlcmd** in the Data Pump Restart Options.
- j. The **Server Name** must be set to the server name of the SQL Server.
- k. Enter **C:\<JadeVer>RPS\RPSSQLScripts** as the **RDB Path**.
- l. Click **OK**.

The configuration options are applied and the following message is displayed.

```
Configuration successful
```

3. When the configuration has been checked, run **RPS | Setup RDBMS**. This extracts and runs the table creation scripts and extracts and loads the data.
4. Use the SQL Server Manager to check that the data has been loaded.
  - a. From the Start menu, run **SQL Server Management Studio Express**.
  - b. Select the **Erewhon** database.
  - c. Select menu item **File | New | Query with Current Connection**.
  - d. Enter the following query.

```
Select * from AddressableEntity
```

This query will output all of the data in the **AddressableEntity** table.

- e. Press F5, to run the new query.

All data from the **AddressableEntity** table is then displayed.

## Enable the RPS Data Pump

The **Datapump** application must be running to update the SQL Server database with changes that have been applied to the RPS database.

Perform the following from the RPS - RPS Manager application.

1. Start the data pump.
  - Select the menu item **RPS | Start Datapump...**

The data pump is started. Progress messages are displayed in the RPS Manager's main window.

2. Check that the data pump is running.
  - a. Go to the **SDS Administration** application
  - b. Check that the **RPS** database is highlighted in green

This indicates that the data pump is running and that the secondary RPS database is synchronized. To determine what all the background colors mean, select **Options | Show Legend**.

If you have any problems starting the data pump, see "[Troubleshooting](#)", later in this document.

## Testing RPS

RPS should now be up and running, and any changes made to the primary database should be sent (via journals) to the RPS database and then (via the data pump) to the relational database. Test that this is functioning as expected.

- [Look at Data in the Relational Database](#)
- [Make a Change to the Primary Database](#)
- [Check the Change is Replicated to the Relational Database](#)

## Look at Data in the Relational Database

1. Go to the RDB.
  - a. Go to **SQL Server Management Studio Express**
  - b. Select the **Erewhon** database

We will now look at data in this database.
2. Add a query.
  - a. Select menu item **File | New | Query with Current Connection**
  - b. Enter the following query:

```
SELECT shortDescription, fullDescription
FROM SaleItem
WHERE (shortDescription = 'Lamp')
```

This query will display details of a single object that we will later update from the primary database.

3. Run the query.
  - Press F5.

This returns the following details.

```
Lamp, Purchase this rare stained glass piece from the early 14th century.
```

As we will re-run this query later to show changes, keep this window open.

## Make a Change to the Primary Database

1. Start the user application in the primary database.

- a. Start the **Jade Application Server**.
- b. Start the **Erewhon Administration** application.
- c. Log on as **Erewhon Investments Inc**.

Data changes are made from the **Erewhon Administration** application. The application server is required, as Erewhon Administration is run as a presentation client.

2. Select an object to change.

- a. Select the menu item **View | Sale Items (by Category)**.
- b. Expand the **Antiques | Items For Sale** branch.
- c. Select the **ANQU2001 Lamp: \$250.00 – Unsold** item.
- d. Click **Edit**.

Details of the object are then displayed.

3. Change some details on the object.

- a. Add some extra text to the **Full Description**
- b. Click **OK**

The object is then updated with the new description. This change made on the primary database should have been replicated automatically to the RDB.

## Check the Change is Replicated to the Relational Database

1. Go to the RDB.
  - a. Go to **SQL Server Management Studio Express**.

We now look at the changed data in this database.

2. Query the data that has changed.
  - a. Re-run the previous query, as follows.

```
SELECT shortDescription, fullDescription
FROM SaleItem
WHERE (shortDescription = 'Lamp')
```

The query result should include the changes that you made to the **Lamp** description in the primary database (above). This shows that the change made in the primary database has been replicated to the SQL Server database.

## Schema Changes

Structural changes made to the primary database may require changes to the table definitions in the RDB. All changes are automatically sent and applied to the RDB.

## Appendix

This appendix contains:

- [Useful References](#)
- [Troubleshooting](#)
- [Tips](#)

## Useful References

For more details, refer to the following documents.

- "[Relational Population Service \(RPS\)](#)" in Chapter 2 of the *Synchronized Database Service (SDS) Administration Guide*
- "[Using the Relational Population Service \(RPS\) Wizard](#)" in Chapter 15 of the *Development Environment User's Guide*

Jade Platform documentation can be downloaded from <https://www.jadeworld.com/jade-platform/developer-centre/learn/documentation>.

White papers can be downloaded from <https://www.jadeworld.com/jade-platform/developer-centre/documentation/white-papers>.

## Troubleshooting

This section is intended to address common problems that might be encountered when setting up and running with RPS. Follow these steps when problems are encountered.

- Error 5505 (*This is an unregistered copy*) occurs if you attempt to run an SDS or RPS node without first entering the specific SDS or RPS license key.

If you want to purchase a Jade Platform license or you believe there is a problem with your existing license, contact Jade Sales and Marketing.

- Check that all components are running.
  - Jade Primary Server
  - Jade RPS Server
  - RPS Data Pump application
  - SQL Server
- Run the **SDS Administration** application and verify that the primary and secondary servers (that is, databases) are connected.
- Check the logs for errors.



## jommsg.log

This log contains information about what is occurring on each database, including successful operations as well as errors. It is usually the first log that you should check when problems are encountered.

## SDS and RPS Work Logs

The primary SDS work log is named `sds_primary-server-name_yyyymmdd` and it contains detailed diagnostic and error information logged by SDS for the primary database.

The RPS work log is named `rps_rps-server-name_yyyymmdd` and it contains detailed diagnostics and error information logged by the SDS or RPS functions executing in the RPS node.

## Application Logs

An application log is created if errors occur for any of the applications. These contain details of any exceptions that occur in the relevant applications.

## Common Errors

### 3201 – SDS not initialized

This error normally means that your primary database is not configured as a primary or that the SDS service failed to initialize.

Check the `jommsg.log` for the primary. Also check that the `[SyncDbService]` and `[ConnectionParams.Primary]` sections are included in your Jade initialization file.

### 3225 – RPS instance ID mismatch with Target Relational DB

The instance ID held by the RPS database does not match the instance id of the target RDB. This id is generated when you create or recreate an RPS database.

If the RPS database is recreated, you should also recreate the RDB.

## Tips

This section provides the following tips about using RPS.

- Keep your first RPS configuration simple, by running all Jade and SQL Server databases on the same machine.
- Make your RPS node easily identifiable from your primary Jade database. Install the RPS database within a separately named directory structure. For example, locate your primary environment in `C:\Jade` and your RPS environment in `C:\JadeRPS`.
- Keep your first test mapping small and simple.
- When creating an initial RPS database, two flavors of RPS are available (mapped extent and full replica). While both modes will work, we recommend using the full replica mode, starting with a small primary database. (The **Working Set** mode is deprecated in Jade 2020 and higher.)
- To evaluate RPS with SQL Server, you can download SQL Server Express and SQL Server Management Studio Express free-of-charge from the Microsoft website. Even with certain size and connection limitations, SQL Server Express will provide a solid, full-featured platform for evaluation of RPS without cost or evaluation time limits.

- If you want to recreate your relational database completely, the simplest approach is to drop and recreate the database, and then reload a full table definition script before bulk-loading the data. Dropping and creating a database can be scripted and you can use the Management Studio (or equivalent) to generate the required scripts.

## Glossary

### Database Role

In a Synchronized Database Environment, the terms *primary* and *secondary* signify the intended function or role of each copy of the database. The valid roles are **Primary Database Role** and **Secondary Database Role**.

### Full Replica Mode

The mode of operation whereby an RPS node retains a full copy of the primary database, regardless of the mapped extent. In this mode, tables in the relational database can be repopulated without impacting the primary database.

An RPS node running in full replica mode can also be used for disaster recovery, as it can acquire the primary database role in a takeover situation.

### Jade System Model

A single Jade specification or object model, potentially spanning multiple Jade schemas.

### Mapped Extent

The set of objects in the production database that are included by all of the **ClassTable** maps in a database map, and are not excluded by filter conditions.

### Mapped Extent Mode

The mode of operation whereby an RPS node retains a copy of the mapped extent, even after user data objects have been replicated to the relational database. In this mode, tables in the relational database can be repopulated without impacting the primary database.

### Nodes and Processes

A primary system is comprised of primary nodes. A secondary system is comprised of secondary nodes. Secondary processes execute in a secondary system.

In addition, special processes called *replay processes* are created in secondary servers to reflect and represent the original updating processes on the primary database. These replay processes are associated with a single secondary node called a *replay node*.

### Primary Database

The database with the **Primary** role is the only database in a Synchronized Database Environment (SDE) where user applications can update persistent objects.

### Primary Database Server

The *primary database server* is the database server node hosting the Jade database with the primary role; that is, this server hosts the primary database.

### Primary System

The *primary system* is the Jade system comprising nodes attached to the primary database server, including the primary database server node itself. User applications executing in the primary system can create and update persistent objects, which reside in the primary database.

### Production Database

The source Jade database being replicated (also known as the *primary* database).

**[Relational] AttributeColumn Map**

Part of a **ClassTable** map. This defines how Jade attributes map to relational attributes (table columns).

**[Relational] ClassTable Map**

Defines how instances of a Jade class are mapped to rows in one or more relational tables.

**[Relational] Database Map**

Defines a mapping between a Jade system model and a single relational database.

**Relational Population Service (RPS)**

The service that provides automatic replication of objects from a primary Jade database to one or more relational databases.

**RDB**

Relational Database.

**RDBMS**

Relational Database Management System.

**RPS Data Store**

A native Jade database maintained by an RPS server node, either as temporary storage or to persist a mapped extent.

**RPS Node**

The Jade database server node that directly connects to a target RDBMS in order to replicate Jade objects to a relational database.

**Secondary Database Role**

Secondary databases are kept synchronized with a primary database by replaying journals automatically shipped from the primary system. User applications cannot update persistent objects (apart from environmental objects) in a secondary database.

A secondary database can assume the role of the primary if the primary database becomes unavailable.

**Secondary Database Server**

A secondary database server is a database server node that hosts a database with a secondary database role.

**Synchronized Database Environment (SDE)**

A *Synchronized Database Environment* (SDE) is the collective term that describes an environment comprising of one or more connected Jade systems that provide or subscribe to a Synchronized Database Service.

An SDE consists of a primary database (where updating occurs) and one or more secondary databases, which are transactionally consistent copies of the primary database. An SDE includes both SDS and RPS nodes.

**Synchronized Database Service (SDS)**

A *Synchronized Database Service* (SDS) is a service that enables you to keep one or more secondary databases synchronized with a primary database.

**Target Database**

A target database is an instance of a relational database connected to an RPS node.