1Integrate

Installation Guide

WildFly Application Server

Product version: v 1.5

Document version: v 1.2.9

Document date: 12/05/2017

1Spatial
Tennyson House
Cambridge Business Park
Cambridge
CB4 0WZ
United Kingdom

Phone: +44 (0)1223 420414

Fax: +44 (0)1223 420044

Web: www.1spatial.com

Contents

1 Introduction

This guide explains how to install 1Integrate.

The procedures apply to both Windows and Linux environments, unless specifically indicated.

For more information about new features and changes in this release, and hardware and software requirements, refer to the *1Integrate Release Notes*.

Audience

This guide is intended for personnel responsible for the installation, configuration, and administration of software.

The procedures detailed in the guide should be performed by a system administrator who is familiar with the application environment of the organisation.

Licenses

1Integrate licences will be issued via email.

1Spatial product support

If assistance is required while installing 1Integrate, please call 1Spatial support on +44 (0)1223 423069, or visit the support section of the 1Spatial website via the Services menu at www.1spatial.com.

2 Prerequisites

Before installing 1Integrate, please ensure you have met all system requirements and installed all necessary prerequisite components:

- ▶ Check "System Requirements" below (including the version of Java required)

- ▶ "Configuring the Database Server" below

- ▶ "ICU Libraries (Linux only)" on page 7

- ▶ "Open Motif Libraries (Linux only)" on page 8

- ▶ "Loading non-Oracle spatial data using FME" on page 18

System Requirements

Please see the *1Integrate Release Notes* for specific system requirements for this release.

Configuring the Database Server

1Integrate requires access to an *Oracle*, *PostgreSQL* or *SQL Server* database to store configuration data such as rule and session definitions, and conformance results. This is referred to as the "1Integrate repository".

> **Note:** Please refer to the relevant product documentation when installing your chosen database server.

> **Note:** If creating a SQL Server database, follow the process through Microsoft SQL Server Management Studio (MSSMS). The name should be "1Integrate" and all other defaults should be used.

Creating a Database User

A database user is required for the 1Integrate repository.

> **Note:** When deploying 1Integrate to multiple servers, one database user is required per environment.

Create an Oracle Database User:

To create a database user, run the following SQL*Plus commands and specify the location of the datafile (including the full file name and extension) as recommended by your Database Administrator:

```
Prompt:> sqlplus [your_system_user_name]/[your_
password]@[tnsname]
```

```
SQL> create tablespace [repository_tablespace_name]
datafile '[datafile_location]' size 10m reuse
autoextend on next 10m maxsize unlimited;
```

```
SQL> create user [repository_username] identified by
[repository_password] default tablespace [repository_
tablespace_name];
```

```
SQL> GRANT create procedure, create sequence, create
session, create table, create view, create trigger,
unlimited tablespace, create type TO [repository_
username];
```

```
SQL> ALTER user [your_system_username] quota unlimited
on [repository_tablespace_name];
```

Create a SQL Server Database User:

1. Create a new user following the **New Login** process in MSSMS.

2. Set the login name to `rsuser`.

3. Select **SQL Server Authentication**, set the password to `rsuser` and de-select **Enforce Password Policy**.

4. Set the default database to 1Integrate.

5. In **User Mapping**, select 1Integrate.

6. Specify the user as `rsuser` and select all database role memberships except `db_denydatareader` and `db_denydatawriter`.

7. In the MSSMS, select the root node of SQL Server and change the security authentication to **SQL Server and Windows Authentication mode**.

> **Note:** To allow the application server to log in to the database, the TCP/IP protocol must be enabled. You can enable this in the SQL Server Configuration Manager in the SQL Server Network Communication section.

Create a PostgreSQL User:

Create a user with the default properties.

When creating your repository, ensure your new user is the owner of this database. This will ensure the user has all required privileges.

Configuring a PostgreSQL repository

In order to use a PostgreSQL repository, the **settings.properties** file (located within C:\Program Files\1Spatial\1Integrate\1Integrate-[version]_ wildfly\SETTINGS) must be edited with the following properties:

```
repository.driver=postgresql
repository.username=[username]
repository.password=[password]
```

> **Note:** Ensure the selected user has the required privileges to access the database.

To use the *public schema* to store the repository:

```
repository.url=jdbc:postgresql://[URL_of_database]/
[database_name]
```

To use a *specified schema* to store the repository:

```
repository.url=jdbc:postgresql://[URL_of_database]/
[database_name]?currentSchema=[repository_schema]
```

ICU Libraries (Linux only)

ICU libraries are required for data and timestamp support in Linux.

A root user (or a user with root access privileges) is required to copy the files from the installation package and run the `ld_config` command.

Install the ICU libraries:

1. Run the `su` command to switch to the root user.

2. Copy the **.so** files from the installation folder (within the ICU folder) to the **/usr/local/lib64/** folder.

3. Create a new configuration file: **/etc/ld.so.conf.d/intergrate.conf**

    > **Note:** Both the **.so** files and the **.conf** file must have read permissions for all users.

4. Inside the configuration file, reference the location of the ICU libraries, for example: `/usr/local/lib64/*`

5. Enter the following `ld_config` command:

    ```
    /sbin/ldconfig -v /usr/local/lib64/
    ```

    Alternatively, edit **~/.bashrc** or similar for the user used to run 1Integrate to include `/usr/local/lib64` on the `LD_LIBRARY_PATH`, as in the following example:

```
# User specific
if [ -z "$LD_LIBRARY_PATH" ]; then
export LD_LIBRARY_PATH="/usr/local/lib64"
else
export LD_LIBRARY_PATH="/usr/local/lib64:$LD_
LIBRARY_PATH"
fi
```

Open Motif Libraries (Linux only)

The 64-bit Open Motif libraries are a prerequisite for Linux installations of 1Integrate.

The required Open Motif libraries are as follows:

- ▶ libX11.so.6
- ▶ libm.so.6
- ▶ libpthread.so.0
- ▶ libnsl.so.1
- ▶ libdl.so.2
- ▶ libc.so.6
- ▶ libcrypt.so.1
- ▶ libXau.so.6
- ▶ libXdmcp.so.6

3 Installing 1Integrate on Wildfly

> **Note:** Before proceeding, ensure you have completed all pre-requisite steps (see "Prerequisites" on page 5).

1Integrate runs as several separate applications: an interface application and one or more session queue applications.

The following instructions describe how to configure an interface and a session queue. This is suitable for running both parts of 1Integrate on one host.

Deployment Preparation

The following information is required prior to deploying 1Integrate:

▶ JDBC URL of the repository created

▶ Username and password for the database user

▶ Path of the **tnsnames.ora** file (if used)

▶ Location of a directory that can be used to store 1Integrate temporary files

> **Note:** On both Windows and Linux, paths must be entered with forward slashes (/) as the path separator instead of backslashes (\). Ensure that no spaces are entered after each parameter as this will result in an unsuccessful installation.

Copying the WildFly directory

Copy the WildFly directory (typically called `1Integrate-[version]-wildfly`) from the installation package to a local folder, then extract the files to a location such as **C:\Program Files\1Spatial\1Integrate**. These files will be used to configure settings for 1Integrate in the following procedures.

In the WildFly installation, the **standalone** directory contains the configuration for the 1Integrate interface and the **standalone-sessionqueue** directory contains the configuration for the session queue.

Configuring Users and Roles

To configure users and roles, navigate to the **\wildfly-[version]\SETTINGS** folder. This folder contains the following files:

▶ **users.properties** contains a list of usernames and passwords, in the form `username=password`.

> 📑 **Note:** All users listed in the following table are included as default.

▶ **roles.properties** contains a mapping from user names to 1Integrate roles in the form `username=role1,role2,role3`

The following table contains a list of pre-defined users, provided upon installation to help get you started with 1Integrate:

Username	Password	Assigned roles
1Spatial	admin1	rs_admins rs_users rswsuser rs_rule_definers rs_data_loaders rs_data_engineers rs_data_quality_stewards
INTAdmin	int101	rs_admins
INTFull	int102	rs_users rswsuser rs_rule_definers rs_data_loaders rs_data_engineers rs_data_quality_stewards
INTUser	int103	rs_users
INTWService	int104	rswsuser
INTRule	int105	rs_rule_definers

Username	Password	Assigned roles
INTLoader	int106	rs_rule_definers rs_data_loaders
INTEng	int107	rs_data_engineers
INTQuality	int108	rs_data_quality_stewards

The following is a list of 1Integrate roles:

Role	Description
int_admins	The administrator can set up system parameters and has all the privileges of the other roles. These include creating and modifying: ▶ rules ▶ data stores ▶ sessions ▶ actions ▶ action maps The administrator can also define sessions and upload files.
int_data_engineers	A data engineer can: ▶ create and modify actions ▶ create and modify action maps ▶ define and run sessions
int_data_loaders	A data loader can upload files into 1Integrate.
int_data_quality_stewards	A data quality steward can define and run sessions only.

Role	Description
int_rule_definers	A rule definer can create and modify: ▶ rules ▶ data stores ▶ sessions The rule definer can also run sessions.
int_users	A user can only view data presented on the interface.
intwsuser	A web service user can use the web services.

Configuring System Properties

The following parameters can be configured in the **settings.properties** file, located within the **SETTINGS** folder within the installation location (e.g. C:\Program Files\1Spatial\1Integrate\1Integrate-[version]-wildfly\SETTINGS).

> **Note:** On Windows, folder paths within the **settings.properties** file must be specified using a forward slash. Spaces in folder names are allowed (e.g. C:/Program Files/1Spatial/1Integrate).

Parameter	Description
Repository Database Connection Details	
repository.driver	The database type (oracle, sqlserver or postgresql).

repository.url	The URL for the database in which to store 1Integrate metadata and entities (rules, actions, sessions and so on), and files that are uploaded to the server used by the data store (such as MapInfo Tab files).
	
	This is in the format:
	
	▶ for Oracle:
	`repository.url=jdbc:oracle:thin:@server1:1521:orcl`
	
	▶ for MS SQL:
	`jdbc:sqlserver://server1;databaseName=1Integrate`

Parameter	Description
repository. username	Username to connect to the repository schema, in which to store 1Integrate metadata and entities (e.g. rules, actions, sessions) as well as files that are uploaded to the server used by the data store (e.g. MapInfo Tab files).
repository. password	Password for the user described by repository.username.
Licence file Location	
ls_license	The location and name of the provided product licence file. **Note:** The licence file is provided by 1Spatial Support and can be placed in any folder.
Location for Temporary Files	
temp.directory	This is a temporary folder location for files that are eventually stored in the database. For Linux, it is recommended that this is set to `/tmp` or `/var/tmp`. For Windows, it is recommended that this is set to `C:/WINDOWS/TEMP`.
Location for the Cache Directory	
cache.directory	The location of the cache directory. When a session is run, a folder is created called "1Integratecache", within which the cache is stored. **Note:** This stores the data cache from data read by 1Integrate. This may require large amounts of disk space depending on the size of data being read into sessions and the number of concurrent sessions. For Linux, if left blank this will default to `/tmp`. For Windows, if left blank this will default to `C:\Users\[user]\AppData\Local\Temp`.

Parameter	Description
Wildfly Server Port Settings	
interface.http.port	The port to use for the interface. **Note:** If possible, use the default port number specified in the file. If there is a clash with another application that already uses this port number, increment the value by 1 until an unused port number is found.
sessionqueue.http.port	The port to use for the session queue. **Note:** If possible, use the default port number specified in the file. If there is a clash with another application that already uses this port number, increment the value by 1 until an unused port number is found.
Additional Settings	
studiotask. countTotalObjects	If set to true, a progress bar is displayed in the user interface during open data tasks to allow the user to see what proportion of the total objects have been loaded so far. This reduces performance for an open data task. If set to false, only the number of objects currently loaded is shown and the user is not shown the progress. This improves performance for an open data task, particularly when loading data from non-database sources. **Note:** Counting the total number of target objects in order to calculate the progress may take excessive time to complete. It is recommended that this parameter is set to **false** for production deployment or when running sessions via web services.
grid.local.address grid.communication. tcp.port	(optional) See "NIC/Network Adaptor Configuration" on the facing page.

NIC/Network Adaptor Configuration

The Grid discovery used to find session queues by default uses the first found non-loopback address, for example a machine with Ethernet adaptors "eth0" and "eth1" and Local Loopback "lo" will likely use "eth0".

> 📋 **Note:** If you do not need to override the adaptor default behaviour, then the following properties do not need to be included.

The following properties must be included in the **settings.properties** file:

```
grid.local.address=[NIC Address]
grid.discovery.tcp.port=[default: 51300]
grid.communication.tcp.port=[default: 51401]
```

Where:

- ▶ **grid.local.address** - specifies the IP address of the network adaptor used for grid communication.

- ▶ **grid.communication.tcp.port** and **grid.discovery.tcp.port** - allows environments to specify known ports (for example, when using a firewall).

> 📋 **Note:** The communication port must be a minimum of 100 greater than the discovery port, in order to avoid conflict.

Starting the Deployment

Before opening the component files, they must first be configured to refer to the correct version of Java.

The procedure is different for Windows and Linux.

<u>Windows:</u>

1. Edit **start-Interface.cmd** and **start-SessionQueue.cmd**.

2. Locate the following lines:

   ```
   @REM Uncomment and set this path to be correct for
   your environment
   REM SET "JAVA_HOME=C:\PROGRA~1\Java\jdk1.7.0_80"
   ```

3. Uncomment the second line and remove "REM", so that it begins with "SET". Ensure that it is pointing to the location of Java home for the correct version of Java installed as part of the pre-requisites. The

resulting line should appear as follows:

```
SET "JAVA_HOME=C:\PROGRA~1\Java\jdk1.7.0_80"
```

4. Once configured, open the files to start the 1Integrate components.

Linux:

1. Edit **start-Interface.sh** and **start-SessionQueue.sh**.

2. Uncomment the line just below "`#Uncomment and set this path to be correct for your environment`":

   ```
   export JAVA_HOME="/usr/java/jdk1.7.0_80"
   ```

3. Once configured, open the files to start the 1Integrate components.

Deploying additional Session queues

1Integrate can be deployed with multiple session queues each of which can run either on the same server as the interface or on separate machines (as long as they are on the same network as the interface).

Session queues can be run in parallel, deployed using different port numbers. This requires a license from 1Spatial that enables the required number of parallel session queues.

> **Note:** On a machine running Windows, this may present a path name length error. If this occurs, copy the **standalone-sessionqueue** folder to a higher level directory, rename it, and paste it back into the **wildfly-[version]** folder.

> **Note:** For more advanced environments, contact either your distributor or 1Spatial prior to continuing to discuss an appropriate deployment plan.

You will need to copy and edit a different file, depending on if you are using Windows or Linux.

Windows:

1. Create a copy of the **standalone-sessionqueue** folder and rename it **standalone-sessionqueue2**.

2. Create a copy of the **start-SessionQueue.cmd** file and rename it **start-SessionQueue2.cmd**.

3. Edit this file to refer to **standalone-sessionqueue2** instead of **standalone-sessionqueue** (or standalone-sessionqueue3 for the third session queue etc.) and change the port offset at the end of the last

`CALL` line from 100 to 200 (or 300 for the third session queue etc.).

> **Note:** For example, the line
> ```
>  CALL bin\standalone.bat -
> P=SETTINGS\settings.properties -
> Djboss.server.base.dir=standalone-
> sessionqueue -Djboss.socket.binding.port-
> offset=100
> ```
> would be changed to
> ```
> CALL bin\standalone.bat -
> P=SETTINGS\settings.properties -
> Djboss.server.base.dir=standalone-
> sessionqueue2 -Djboss.socket.binding.port-
> offset=200.
> ```

Linux:

1. Create a copy of the **standalone-sessionqueue** folder and rename it **standalone-sessionqueue2**.

2. Create a copy of the **start-SessionQueue.sh** file and rename it **start-SessionQueue2.sh**.

3. Edit this file to refer to **standalone-sessionqueue2** instead of **standalone-sessionqueue** (or standalone-sessionqueue3 for the third session queue etc.) and change the port offset at the end of the last `CALL` line from 100 to 200 (or 300 for the third session queue etc.).

> **Note:** For example, the line
> ```
> CALL bin\standalone.bat -
> P=SETTINGS\settings.properties -
> Djboss.server.base.dir=standalone-
> sessionqueue -Djboss.socket.binding.port-
> offset=100
> ```
> would be changed to
> ```
> CALL bin\standalone.bat -
> P=SETTINGS\settings.properties -
> Djboss.server.base.dir=standalone-
> sessionqueue2 -Djboss.socket.binding.port-
> offset=200.
> ```

Loading non-Oracle spatial data using FME

1Integrate, by default, loads spatial data from an Oracle database. To load spatial data from other sources (e.g. Shapefile, Tab files, MS SQL Server, PostGIS) 1Integrate uses functionality from Safe Software's FME.

To use this capability, you must have FME Desktop installed on the server or on a local file system location available to the server, with an FME license available for use.

> **Note:** In order to be editable, tables must have a primary key defined. You can read data from tables that do not have primary keys, but you will not be able to write data back to those tables.

> **Note:** For PostGIS, only tables with at least one geometry column are supported.

To allow 1Integrate to access non-Oracle data sources, use the following procedure, depending on your operating system:

Configure FME on Windows

> **Note:** The following steps are only necessary if FME has not already been added to the system or user's PATH environment variable in Windows.

On Windows, set the FME_HOME directory to the path in **bin\standalone.conf.bat** by adding the following lines to the top of the file:

```
@rem add FME to PATH
set PATH=[FME path];%PATH%
```

Where `[FME path]` is the location where FME is installed (e.g. C:\PROGRA~1\FME).

Configure FME on Linux

Set the FME_CORE directory to the LD_LIBRARY_PATH in **bin/standalone.conf** by adding the following lines to the top of the file:

```
# add FME to LD_LIBRARY_PATH
export LD_LIBRARY_PATH="[FME_CORE_PATH]:$LD_LIBRARY_
PATH"
```

Where `[FME_CORE_PATH]` is the location of the fme/fmecore/ directory.

Offline Help

By default, clicking the Help button within 1Integrate opens the online WebHelp. However, this button can be configured to open local offline documentation instead.

> **Note:** The configuration of offline help is only advised for environments without access to the internet, as local help files will not receive updates as often as the online help.

Before performing the configuration, ensure your local help files are places somewhere accessible on your local network. If the help files have not been provided with your release, please contact your vendor or 1Spatial Support.

Configure Offline Help :

1. Open the **settings.properties** file (located within C:\Program Files\1Spatial\1Integrate\1Integrate-[version]_wildfly\SETTINGS).

2. Edit the value for the **help_base_url** parameter to the location of your local help files.

    ```
    help_base_url=[location of help files]
    ```

    Where `[location of help files]` is replaced with the path to your local help files.

4 Testing the Installation

> **Note:** Empty your browser cache before testing your installation.

1Integrate can be accessed through the following site:
http://[machine]:[service_port]/1Integrate

> **Note:** Use the port number specified during installation, by default this is 8080.

Log in as a user with administrator permissions. The home page displays a traffic light system:

- ▶ If the status traffic light icon displayed on the page is green, 1Integrate has been successfully installed.

- ▶ If the traffic light icon is amber or red, click on it to display information for any configuration errors.

To verify the session queue installations, click the **Admin** tab and check that the Grid Topology matches the number of interfaces and session queues installed.

If you need to access the 1Integrate Web Services API, then a web page with documentation and a link to the WSDL document can be found at:
http://[machine]:[service_port]/soap

5 Upgrading an Installation

> **Note:** Before performing an upgrade to an existing installation, ensure **all sessions are stopped** and perform a **backup** of your repository.

> **Note:** These instructions apply when upgrading from one version to the immediately subsequent release only (e.g. from 1.1 to 1.2). If you are performing an upgrade from any older version, please consult your release notes or contact 1Spatial Support.

Upgrade an Installation:

Upgrading an installation on Wildfly consists of unpacking your new release folder, copying across your existing settings files, and editing the Java location in your interface and session queue component files.

> **Note:** The following instructions use an example where currently version 1.A is installed, and we want to upgrade to 1.B. We have used **C:\1Spatial** as a directory within which to store our installation package files, organised into **C:\1Spatial\Product-1.A** and **C:\1Spatial\Product-1.B**.

1. Unzip your new Release Package, and the Wildfly folder within it (e.g. **C:\Program Files\1Spatial\Product-1.B\Product-1.B_wildfly\wildfly-[version]**).

2. Copy the following properties files from the **SETTINGS** folder in your *existing* installation directory (e.g. **C:\Program Files\1Spatial\Product-1.A\Product-1.A_wildfly\wildfly-[version]\SETTINGS**, into the **SETTINGS** folder in your *new* installation directory:

    ▶ **settings.properties**

    ▶ **roles.properties**

    ▶ **users.properties**

3. Copy any built-ins and custom data stores (**.jar** files) from your *existing* installation directory (e.g. **C:\Program Files\1Spatial\Product-1.A\Product-1.A_wildfly\wildfly-**

[version]\standalone\deployments\Product-1.A.ear\lib and
**C:\Program Files\1Spatial\Product-1.A\Product-1.A_wildfly\wildfly-
[version]\standalone-sessionqueue\deployments\Product-
1.A.ear\lib**) into the corresponding **lib** folders within your new
installation directory.

4. Before opening the component files, they must first be configured to refer
to the correct version of Java.

The procedure is different for Windows and Linux.

<u>Windows:</u>

1. Edit **start-Interface.cmd** and **start-SessionQueue.cmd**.

2. Locate the following lines:

```
@REM Uncomment and set this path to be correct
for your environment
REM SET "JAVA_HOME=C:\PROGRA~1\Java\jdk1.7.0_
80"
```

3. Uncomment the second line and remove "REM", so that it begins
with "SET". Ensure that it is pointing to the location of Java home
for the correct version of Java installed as part of the pre-requisites.
The resulting line should appear as follows:

```
SET "JAVA_HOME=C:\PROGRA~1\Java\jdk1.7.0_80"
```

4. Once configured, open the files to start the 1Integrate components.

<u>Linux:</u>

1. Edit **start-Interface.sh** and **start-SessionQueue.sh**.

2. Uncomment the line just below "`#Uncomment and set this
path to be correct for your environment`":

```
export JAVA_HOME="/usr/java/jdk1.7.0_80"
```

3. Once configured, open the files to start the 1Integrate components.