SSP v2.5.2 Installation Instructions

Released August 21, 2014

For all existing installations of 2.0.X and 2.1.X, important upgrade instructions exist in the previous 2.1, 2.2, 2.3, 2.4 and 2.5.1 Release notes.

- To upgrade from 2.0.X follow the upgrade instructions for 2.1, 2.2, 2.3, 2.4 and 2.5.1 Release Notes before deploying the 2.5.2 code
- To upgrade from 2.1.X follow the upgrade instructions for the 2.2, 2.3, 2.4 and 2.5.1 Release Notes before deploying the 2.5.2 code
- To upgrade from 2.2.X follow the upgrade instructions for the 2.3, 2.4 and 2.5.1 Release Notes before deploying the 2.5.2 code
- To upgrade from 2.3.X follow the 2.4 and 2.5.1 Release Notes before deploying the 2.5.2 code
- To upgrade from 2.4.X following the 2.5.1 Release Notes before deploying the 2.5.2 code
- New installations of 2.5.2 are not required to make any additional change

If you are running a SSP version prior to 1.1.1, you are strongly encouraged to upgrade or otherwise apply the reporting subsystem security patches described by SSP-701.

If you are running SSP version 2.0.0 or 2.0.0-b3, you are strongly encouraged to upgrade to 2.0.1 or 2.1.0 or 2.2.0 or later or otherwise apply the Confidentiality Level-related patches for the Student Documents tool as described by SSP-1917.

Also please take a few minutes to review additional security-related announcements detailed at the top of the SSP space here in Confluence.

Step by step instructions for building and deploying the SSP 2.5.2 release.

1. Software Prerequisites (JDK, Tomcat, Maven, Ant, RDBMS)
2. SSP Platform build and deployment

Software Prerequisites

The following software prerequisites must be installed with the appropriate environment variables to build and run SSP:ssp-platform.PNG

- JDK 1.6 update 21 or later (JDK 1.7 is not supported as of 2014/08; the SSP development team has also observed somewhat better GC performance with the Sun/Oracle JDK vs OpenJDK)
  - Download Location: http://java.sun.com
  - Environment Variable: JAVA_HOME

Java Environment Variable

```
JAVA_HOME=/path/to/your/java (ie: /usr/local/java or C:\java\jdk)



(optional)
PATH= append the bin subdirectory to the path statement
```

- Tomcat 6.X (Tomcat 7 is not supported as of 2014/04)
  - Instructions for installing and configuring Tomcat for the SSP-Platform (uPortal 4.0)

Tomcat Configuration

It is important to complete sections: Environment Variables, Shared Libraries, Shared Sessions, Java Heap. Minimally, the catalina.properties file must contain:

```
shared.loader=${catalina.base}/shared/lib/*.jar
```

And your active connector/s in `<tomcat>/conf/server.xml` must have the `emptySessionPath` flag set:

```
<Connector port="8080" protocol="HTTP/1.1"
    connectionTimeout="20000" redirectPort="8443" emptySessionPath="true"/>
```

And increase the heap in `<tomcat>/bin/setenv.sh` (*nix) or `<tomcat>/bin/setenv.bat` (Windows). Smaller sizing is probably feasible, but the examples below match what our SSP CI envs run with. For production systems, start with a max heap of roughly half available physical memory and increase from there if necessary.
The uPortal instructions above recommend using `JAVA_OPTS` for heap sizing. This can lead to problems on memory constrained systems because `JAVA_OPTS` will be used when trying to stop Tomcat with its own scripts. You don't typically need a large heap *at all* for that operation. So `CATALINA_OPTS` is a better choice for sizing the heap in `setenv` scripts, because that var will only be used for Tomcat's http-serving runtime.

setenv.sh:

CATALINA_OPTS=-Xms2G -Xmx2G -XX:PermSize=256m -XX:MaxPermSize=256m

setenv.bat (uPortal instructions linked to above are missing the 'set'):

set CATALINA_OPTS=-Xms2G -Xmx2G -XX:PermSize=256m -XX:MaxPermSize=256m

- Additionally, a performance improvement has been experienced by enabling compression in Tomcat

Add compression="force" to the server.xml in the connector like the following:

<Connector port="8080" protocol="HTTP/1.1

connectionTimeout="20000"

redirectPort="8443"

emptySessionPath="true"

compression="force" />

- Maven 3.0.3 or later
  - Download Loation: http://maven.apache.org
  - Environment Variable: MAVEN_HOME

Maven Environment Variables

```
MAVEN_HOME= /path/to/your/maven (ie: /usr/local/maven or C:\tools\maven)

M2_HOME= /path/to/your/maven (ie: /usr/local/maven or C:\tools\maven)


(optional)
PATH= append the bin subdirectory to the path statement
```

- Ant 1.8.2 (use this exact version)
  - Download Location: http://ant.apache.org
  - Environment Variable: ANT_HOME

Ant Environment Variable

```
ANT_HOME= /path/to/your/ant (ie: /usr/local/ant or C:\tools\ant)

(optional)
PATH= append the bin subdirectory to the path statement
```

- Sencha SDK
    - Download Location: http://www.sencha.com/products/sdk-tools/download

    ```
    PATH= append the root to the path statement
    ```

    - See SSP Sencha Build Tool Usage for additional installation steps on 64-bit OSs

- RDBMS (support for PostgreSQL and Microsoft SQL Server)
    - PostgreSQL 9.1 or later
        - Download Location: http://www.postgresql.org
            - On Unix:
                - PostgreSQL is available in the package manager of most linux distributions.
                    - Install it according to the distribution's instructions
                - Ubuntu - https://help.ubuntu.com/11.10/serverguide/C/postgresql.html
            - On Mac:
                - PostgreSQL is available via the homebrew package manager or as a download on the postgresql.org site.
            - On Windows:
                - PostgreSQL is available as a download on the postgresql.org site.
        - Configure PostgreSQL
            - **Server Connection**
                - Launch the PG Admin application
                - In the Object Browser, navigate to and right click on Server Groups -> Servers -> PostgreSQL 9.1 (localhost:5432)
                - Click Connect and the enter the administrator password
            - **Login Roles**
                - In the Object Browser, right click on Login Roles and click New Login Role
                    - In the Properties tab, enter a Role name of "sspadmin" without the quotes
                    - In the Definition tab, enter a Password of "sspadmin" without the quotes
                - In the Object Browser, right click on Login Roles and click New Login Role
                    - In the Properties tab, enter a Role name of "ssp" without the quotes
                    - In the Definition tab, enter a Password of "ssp" without the quotes
                - Confirm the new Login Roles exist in the Object Browser
            - **Database**
                - In the Object Browser, right click on Databases and click New Database
                    - Enter "ssp" without the quotes as the database name
                    - Enter "sspadmin" without the quotes as the database owner
                - Confirm the new database exists in the Object Browser
    - Microsoft SQL Server 2008 R2
        - **Server Connection**
            - Launch the SQL Server Management Studio application
            - Enter your database connection info including administrator account credentials, and click Connect
        - **Login Roles**
            - Navigate to Security->Logins, and right click on New Login
                - Login name of "sspadmin" without the quotes
                - Select SQL Server authentication and enter a Password of "sspadmin" without the quote
                - Uncheck Enforce password policy
            - Right click on Logins again, and New Login Role
                - Login name of "ssp" without the quotes
                - Select SQL Server authentication and enter a Password of "ssp" without the quote
                - Uncheck Enforce password policy
            - Confirm the new users exist
        - **Database**
            - Navigate to and right click on Databases and click New Database
                - Enter "ssp" without the quotes as the database name
            - Confirm the new database exists
            - Run the following SQL to assign user permissions and configure the required database settings

SQL Server Configurations

```
USE [ssp]
GO
IF NOT EXISTS
(SELECT name FROM sys.filegroups WHERE is_default=1 AND name =
N'PRIMARY') ALTER DATABASE [ssp] MODIFY FILEGROUP [PRIMARY] DEFAULT
GO
IF NOT EXISTS (SELECT name  FROM sys.database_principals WHERE name =
'ssp')
BEGIN
CREATE USER [ssp] FOR LOGIN [ssp]
EXEC sp_addrolemember N'db_datawriter', N'ssp'
EXEC sp_addrolemember N'db_datareader', N'ssp'
END
GO
CREATE USER [sspadmin] FOR LOGIN [sspadmin]
GO
EXEC sp_addrolemember N'db_owner', N'sspadmin'
GO
```

```
For MSSQL 2008 or later (note that these statements must be executed while
*no other connections to the current database are open*):

ALTER DATABASE MyDatabase
 SET ALLOW_SNAPSHOT_ISOLATION ON
ALTER DATABASE MyDatabase
 SET READ_COMMITTED_SNAPSHOT ON
```

Also note that for SQLServer the "operational" SSP database user ('ssp' in the example above) must be allowed to execute stored procedures. In most deployments this does not require special configuration, but in the event your security policies are such that that user must be explicitly granted execute permissions on specific stored procs, here are the statements which you would likely need to run. (Use 'dbo' for <schema> unless you know the value should be something else (`db_schema` from `$SSP_CONFIGDIR/ssp-config.properties`); use 'ssp' for <ssp-operational-user> unless you've chosen a different name for that account (`db_username` from `$SSP_CONFIGDIR/ssp-config.properties`).
):

```
GRANT EXEC on <schema>.REFRESH_MV_DIRECTORY_PERSON to <ssp-operational-
user>;
GRANT EXEC on <schema>.REFRESH_MV_DIRECTORY_PERSON_BLUE to <ssp-
operational-user>;
GRANT EXEC on <schema>.update_directory_person_from_view_where_school_id
to <ssp-operational-user>;
GRANT EXEC on <schema>.update_directory_person_from_view_where_person_id
to <ssp-operational-user>;
```

RDBMS Platform Flexibility

Currently SSP supports use of PosgreSQL 9.x and Microsoft SQL Server 2008, or 2008 R2. Starting with 2.5.2, SSP has begun to include patches for SQLServer 2012 compatibility and at least one real-world 2.5.2 deployment is running against SQL Server 2012, although the SSP project team does not officially test against that SQL Server version.

Configure and Deploy SSP-Platform

The following configurations are required to build and deploy SSP-Platform.

1. Download the SSP-Platform Release

Zip Download

The source files can be downloaded in a zip file

Download Location: SSP-Platform-2-5-2.zip ("SSP Platform" is a portal application which acts container for SSP itself. The two applications are versioned independently. By default, version 2.5.2 of SSP Platform will include version 2.5.2 of SSP.)

- Unzip the file into a suitable path (e.g. on Windows C:\ssp\platform-src or on Unix/Linux/Mac /usr/local/ssp/platform-src)

2. SSP Configuration Files

- Create a directory for the local SSP configuration files
  - Example:
    - Unix/Linux/Mac example: /usr/local/ssp/ssp-local
    - Windows example: C:\ssp\ssp-local
  - Make the directory only readable by the user that is running Tomcat
  - Set an Environment Variable for the local configuration file location

```
SSP_CONFIGDIR=/path/to/your/local-configuration (ie: /usr/local/ssp/ssp-local or C:
\ssp\ssp-local)
```

- ssp-config.properties
  - The ssp-config.properties file must be modifed for database connectivity and email settings
  - **Baseline File Location**: ,ssp root>/src/main/config/external//ssp-config.properties
  - **Action**: Copy the baseline ssp-config.properties file into the local configuration directory created above and rename it to `ssp-config.properties`. Or start with an empty `ssp-config.properties` in that directory and add only the properties for which you need to override the default value.
  - **Configuration Values:**

Value	Description	Note
system_id	Unique identifier of the SSP instance	
db_username	Values for connecting to the SSP database	
db_password	Values for connecting to the SSP database	
db_admin_username	Values for connecting to the SSP database	
db_admin_password	Values for connecting to the SSP database	

db_username_liquibase	Value to allow for MS SQL Server domain accounts	${db_username_liquibase} and ${db_username} should be set the same value unless you're on SqlServer, using the JTDS driver, and SSP connects to the database as domain users. If that applies to you, keep ${db_username} set to the unqualified account name, but change ${db_username_liquibase} to the fully-qualified domain account name as shown here. Include the brackets and double back-slashes.
		db_username_liquibase=[DOMAIN\\username]
		default is ${db_username}

db_schema	Db schema for the SSP database	Examples: Postgres: public SQLServer: dbo
db_name	Value for the SSP database	
db_url	jdbc connection syntax	For Microsoft SQL Server, either specify a port (the default is 1433) or ensure that the SQL Server Browser service is running because the SQL Server JDBC driver defaults to port 1434 which is the SQL Server Server Browser service default port. Depending on the server configuration, either may work, or you may want to explicitly specify the port and instance name, if applicable. For best results with SQL Server, the JTDS driver included with the Platform installation is recommended. Examples of the url are provided in the sample ssp-config.properties file. SQL Server db_url w DOMAIN USER AUTHN may look like this; substitute machine name, instance and domain names w/o <>'s db_url=jdbc:jtds:sqlserver://<machine_name>:1433/${db_name};instance=<instance_name>;domain=<domain_name>
db_driver_class	jdbc database connectivity syntax	For best results with SQL Server, the JTDS driver included with the Platform installation is recommended. Examples of the class are provided in the sample ssp-config.properties file.
db_dialect	Hibernate dialect	Use of one of the `org.jasig.ssp.util.hibernate.ExtendedSQLServer*Dialects` is strongly encouraged if running against SQLSever. The default `ssp-config.properties` has an example.)
db_conns_max_active	Values for the database connection pool	The default value will need to be increased for test and production

db_conns_max_idle	Values for the database connection pool	The default value will need to be increased for test and production
db_conns_max_wait	Values for the database connection pool	
db_conns_validation_query	Values for the database connection pool	
db_liquibase_enabled	Enables the liquibase script for database table management	
db_liquibase_changelog	Location for the liquibase change log	
db_liquibase_set_mssql_snapshot_isolation	Parameter for configuring a MSSQL database	**IMPORTANT** The default value is 'true'. Set this value to 'false for MSSQL. The liquibase changeset 000014.xml will be ignored. The sql above configures the database correctly.
db_liquibase_strip_journal_comment_markup	Parameter to enable a script to convert HTML Journal Entries to plain text	
db_liquibase_strip_tuition_paid_is_y	True value will delete the existing values forced into the database in v1.2.0, False will leave the existing values alone	This only applies to implementers who installed v1.2.0 or earlier AND populated the external_registration_status_by_term.tuition_paid field with external data
db_liquibase_external_fa_not_null_drop_y	True value allows the table to be re-created with the correct column definitions for null values	
db_liquibase_external_apply_natural_keys	True value will apply the new primary keys to the external database	Version 2.0.0 added primary keys to the external database tables for performance and uniqueness enhancements. If there are non-unique values in the database, the liquibase will fail to make the table changes.
`db_liquibase_manage_external_database_by_default`	True value will allow SSP to manage the tables and views	If you want to take total control of SSP's external views and tables, change that property to false in your SSP_CONFIGDIR/ssp-config.properties *before first startup*. And once you've started up, there's really no point in ever changing that value afterwards. (If you turn it off, then decide you want SSP to manage external views and tables after all, you'll need to update config set value = 'true' where name = 'manage_integration_database' and then restart.)
db_liquibase_convert_external_term_timestamps	True value in external_term.start_date and external_term.end_date will be interpreted in ${db_time_zone_legacy} and re-written in ${db_time_zone_legacy}.	True usually makes sense for both upgrades and fresh installs. Would only set to false if for some reason these fields have already been converted to ${db_time_zone) via some external process.
`db_batchsize`	The number of records to process for database transactions.	The default value is 300. Use of the parameter can increase performance of queries writing large sums of data into the database. This is primarily used in the Caseload Re-assignment tool.

student_documents_base_dir	Base Directory for student documents	The default is ${catalina.base}/ssp-uploads/student-docs It is important to not end in path separator like / or \
student_documents_volumes	Comma seperated list of subdirectories under student documents	It is important to not end in path separator like / or \
student_documents_file_types	Comma separated list of allowable file types that will be used to validate student document files	The initial types are pdf,gif,jpg,jpeg, doc,docx,xls,png. It is important to not include the period /dot in the definition. Only the type abbreviation is required.
student_documents_max_size	Maximum size of an individual file, in bytes	The default value is 5000000
`cacheLifeSpanInMillis`	This property will dictate how long lived a cache will be only external courses uses a cache	default is 86400000 = 1 day
db_time_zone_legacy	Parameter to set the timezone for data migration	Used for migrating persistent timestamps. Prior to work on SSP-1002, SSP-1035, and SSP-1076, timestamps were stored in the JVM default timezone. After that the application assumes they are stored in ${db_time_zone}. In order to correctly migrate existing data, though, the app needs to know the original timezone. This is almost always going to be the current JVM default timezone, hence the default value here, which is a special value instructing the app to lookup and inject that timezone into this config property. In the rare event you need to change that value, you can do so here. This would likely only be necessary if, for whatever reason you change the JVM default *after* the migrations run, which would result in a Liquibase checksum error. To avoid that, just set the relevant timezone here when and if you make that change. Default is CURRENT_JVM_DEFAULT
db_time_zone	Timezone ID for the JVM	JVM-recognized TimeZone ID for the zone in which persistent date/time values should be interpreted. Should rarely if ever need to be overridden. If overridden, should always be set to a TimeZone that does not observe Daylight Savings Time unless trying to cope with legacy data that was stored in a DST-aware TimeZone. Once set, should never be changed else date /time values in the database will be interpreted incorrectly. (SSP does not store timezone data on persistent date /time values and implements no logic for detecting and/or handling changes to this configuration option.) Default is UTC

highly_trusted_ips	The list of IP addresses that are allowed to access the APIs	This is used in conjunction with high_trusted_ips_enabled in the System Configuration
smtp_username	Value for email relay	
smtp_password	Value for email relay	
smtp_host	Value for email relay	
smtp_port	Value for email relay	
smtp_protocol	Protocol for email	Default is smtp
ssp_admins_email_addresses	Recipient of system generated messages	
scheduled_coach_sync_enabled	Parameter to enable coach sync process	
per_coach_sync_transactions	Parameter to enable the sync process to run per coach instead of one large transaction for all coaches	
`scheduled_task_cleanup_wait_millis`	Max amount of time, in milliseconds, the app will wait during shutdown for any background tasks to abandon their work.	Default is 10000
uportal_session_keep_alive_timeout	Length of time for uPortal sessions KeepAliveFilter	
oauth2_client_password_encoding_secret	Config for setting the key with which OAuth2 Client secrets are hashed before being placed into the database	See SSP OAuth2 API Authentication
spring.profiles.active	Deployment options	• dev-standalone: completely free of uPortal • standalone: as the only portlet in a uPortal instance • uPortal: as one of many portlets in a uPortal instance
ssp_main_use_minifed_js	Parameter to determine the javascript file used in the deployment	When set to true, ssp-main.jsp will include a minified js called app-all.js When set to false, ssp-main.jsp will include the non-minified app.js
`ssp_trusted_code_run_as_key`	When the scheduled jobs run they have to "run as" a particular user. SSP uses SpringSecurity for this, and the application code is allowed to sudo to a different user as long as it knows the special shared secret defined in the configuration.	Default is SZP. If you plan on running deployment-specific third-party code, or really even other webapps in the same Tomcat contains, you should probably select a more complex, deployment-specific value.
`ssp_platform_sso_ticket_service_shared_secret`	Enables LTI and legacy inbound SSO mechanisms	If unset, LTI and legacy inbound SSO will be disabled. To enable those features, set to a non-empty value and ensure the same value is set as `environment.build.sso.local.sharedSecret` in `$SSP_CONFIGDIR/ssp-platform-config.properties`.

logback.xml

- The logback.xml controls the log location and level
- **Baseline File Location**: <ssp root>/src/main/config/external/logback.xml
- **Action**: Copy the baseline logback.xml file into the local configuration directory created above
- **Configuration Values**: Typically you only need to modify the path at which this file will find the `ssp-config.properties` file you configured above. E.g.:

logback.xml

```
<property file="C:/ssp/ssp-local/ssp-config.properties" />
```

- Additional configuration options
    - Adjust the log levels for each log appender as necessary
    - Enable the smtpAppender (disabled by default)
    - Further details regarding managing the logback.xml are included in XML comments within the file

3. Modify SSP-Platform Configuration Files

- build.properties
    - Copy The build.properties.sample file is copied or renamed in the current directory. The parameter defines the location of Tomcat.
    - **File Location**: `<platform-src-dir>/build.properties.sample`
    - **Action:** Create a copy of that file in the same directory, renaming it it `build.properties`.
    - **Configuration Values**: Set `server.home` to the path below which your Tomcat `webapps` directory is located.

build.properties

```
server.home=C:/path/to/your/tomcat/install
```

    - ssp-platform-config.properties
        - The ssp-platform-config.properties file must be modifed for database connectivity and email settings
            - Original File Location: ./uportal-war/src/main/resources/properties/ssp-platform-config.default.properties
            - Edit the file and save in the SSP_CONFIGDIR
        - **Run-Time File Location**: <SSP_CONFIGDIR>/ssp-platform-config.properties
        - **Configuration Values:**

Value	Description
environment.build.hibernate.connection.driver_class	jdbc driver file For best results with SQL Server, the JTDS driver included with the Platform installation is recommended.
environment.build.hibernate.connection.url	jdbc connection syntax For best results with SQL Server, the JTDS driver included with the Platform installation is recommended.
environment.build.hibernate.connection.username	jdbc connection database username
environment.build.hibernate.connection.password	jdbc connection database password
environment.build.hibernate.dialect	jdbc connection dialect For best results with SQL Server, the JTDS driver included with the Platform installation is recommended.
environment.build.uportal.server	Hostname and port for your SSP deployment. (Default: localhost:8080)
environment.build.uportal.protocol	HTTP/S protocol at which end users access your SSP deployment. (Default: http)
environment.build.uportal.email.fromAddress	Address from which Platform email will originate. Rarely used. (Default: ssp@university.edu)

environment.build.sso.*	Several properties which configure inbound LTI and legacy SSO. See documentation specific to those features: SSP LTI Provider and SSP Signed URL SSO. Note that to enable these features `environment.build.sso.local.sharedSecret` must be set to the same non-empty value as `ssp_platform_sso_ticket_service_shared_secret in $SSP_CONFIGDIR/ssp-config.properties`

4. Build SSP-Platform

- Use the following command to build, deploy, and initialize the SSP-Platform project:

```
*** When running a database initialization ant target (initportal,
initdb), you need to specify SSP_CONFIGDIR if it isn't already specified
as an env var.

E.g on *nix.... $> SSP_CONFIGDIR=/opt/ssp/sspconfig ant -Dmaven.test.
skip=true clean <target>

Most Common Commands

- Re/Initialize the SSP-Platform database, then run the equivalent of
deploy-ear. Destructive! Appropriate for first-time deployments.

  $> SSP_CONFIGDIR=/opt/ssp/sspconfig ant -Dmaven.test.skip=true clean
initportal

- Build and deploy entire SSP-Platform portal, including SSP:

  $> SSP_CONFIGDIR=/opt/ssp/sspconfig ant -Dmaven.test.skip=true clean
deploy-ear


Other commonly used ant targets:

testdb: Tests the database settings and connectivity

initdb: Drop SSP-Platform tables in the db & recreate them with configured
seed data (src/main/data, not including the "quickstart" folder).

deploy-war: Build & deploy _just the SSP-Platform war_ (i.e. not SSP or
other portlets, etc.).

deployPortletApp: Deploy one (already-built) portlet war file to Tomcat
(example ant deploPortletApp -DportletApp=../SSP-Open-Source-Project/target
/ssp.war)
```

- Additional step for Microsoft SQL Server to update column types

For Microsoft SQL Server ONLY

Follow steps 2 & 3 from the following page to update appropriate database tables for SSP-PLATFORM

- Restart Tomcat

5. Test Deployment

- Start Tomcat and point your browser to `http://localhost:8080/ssp-platform`
  Click Sign In, and use the credentials user: admin password: admin

6. Production Deployment Tips

Delete Demo Users

If you are *upgrading* an environment, you should delete or change the passwords for the uPortal users created for demonstration purposes. This can be done through the user interface: `Manage Users` -> `Find an Existing User` -> [Enter user ID from list below] -> [Click result] -> `Delete` or `Edit`, then change password. Demo users:

- advisor0
- ken
- student0
- student1

This is only necessary for upgrades. A fresh 2.5.2 install will not create these users.

A fresh install should also either change the `admin` user's password or add some other user to the Portal Administrators group and delete the `admin` user.

If anything in it is incorrect or unclear, please leave a comment below.