



Degree Works
Responsive Dashboard
Administration Guide

Release 5.0.3.1
March 2020

Notices

© 2017-2020 Ellucian.

Contains confidential and proprietary information of Ellucian and its subsidiaries. Use of these materials is limited to Ellucian licensees, and is subject to the terms and conditions of one or more written license agreements between Ellucian and the licensee in question.

In preparing and providing this publication, Ellucian is not rendering legal, accounting, or other similar professional services. Ellucian makes no claims that an institution's use of this publication or the software for which it is provided will guarantee compliance with applicable federal or state laws, rules, or regulations. Each organization should seek legal, accounting, and other similar professional services from competent providers of the organization's own choosing.

Ellucian's Privacy Statement is available at: www.ellucian.com/privacy.

Ellucian shall have the right to (a) use, store, process, modify, reproduce, distribute and display customer data, and to grant sublicenses to third parties, for the sole purposes of providing the software, performing Ellucian's obligations under its agreements with customers and complying with applicable law or legal requirements; (b) use, store, process, modify and reproduce customer data for Ellucian's internal business purposes, including development, diagnostic, forecasting, planning, analysis and corrective purposes in connection with the software, and for otherwise improving and enhancing the software; and (c) use, store, process, modify, reproduce, display, perform, distribute, disclose and otherwise exploit in any manner Aggregated Data for Ellucian's business purposes, including disclosure within its public statements and marketing materials describing or promoting Ellucian or the software. "Aggregated Data" means any data obtained or generated by Ellucian, including data pertaining to the software, Ellucian's systems and software, and the use of any of the foregoing, and includes data derived from customer data, which in all instances (i) does not identify any individual and (ii) is not attributed or attributable to a specific customer. Aggregated Data includes data that has been combined into databases which include third party data.

Ellucian
2003 Edmund Halley Drive
Reston, VA 20191
United States of America

Contents

Notices	2
Introduction	7
Initial Set-up and Configuration	8
Authentication	8
Granting Users Access to Responsive Dashboard	8
Shepherd Settings	8
Deploying the Responsive Dashboard	9
Integration with Portals	13
Accessing APIs from External Applications	14
External Course Information	14
Using Responsive Dashboard and Dashboard concurrently	15
Responsive Dashboard Navigation	16
Mega Menu	16
Logout	16
Student	17
Find Students	17
Student Data	18
Tools	21
Print	21

Contact	21
GPA Calculator	21
Class History	23
Petitions	23
Notes	24
Worksheets	26
Academic	26
What-If	30
Athletic Eligibility	35
Exceptions.....	37
Add Exception	37
Remove Exception	41
Exception Management.....	42
Manage Petitions.....	42
Fix Petition Status.....	44
Exceptions Report	44
Plans	46
Plan List	46
Creating and Editing Plans	46
Creating and Editing Terms	48
Creating and Editing Requirements.....	49

Creating and Editing Notes	55
Sidebar	56
Course Link	57
Course Information	57
Audit	58
What-If	59
Create Block	59
Plan Approval	61
Course Validation.....	61
Prerequisite Checking	62
Tracking	62
Batch Processes	66
Scribe Pointers	67
Template Management	71
Template List	71
Creating and Editing Templates	72
Creating and Editing Requirements.....	73
Creating and Editing Notes.....	78
Sidebar	79
Course Link	79
Transfer Finder.....	80

Admin	81
Debug	81
Theme	82
Links	83

Introduction

The Responsive Dashboard is the primary user interface for students, advisors and other users to process and view individual degree audits, run what-if scenarios, add and view notes and petitions, manage exceptions processing and create and track educational plans.

The Responsive Dashboard contains both the frontend (browser-based) application as well as the backend API that is used by the client code. It is deployed as a jar on any supported Unix server and is independent of the classic software. It includes an embedded container – a separate Java container like Tomcat or WebLogic is not required.

The Responsive Dashboard was introduced with Release 5.0.2, and is a new visual metaphor based on a design concept where information is clustered in “visual cards” to enhance the user experience when interfacing with Degree Works. Our motivation for adopting this new metaphor was that as an information container, cards may contain important data elements such as text, rich media, buttons, and so forth. The contents then drive the card to adapt its size to that of different devices and screens, enhancing the user experience through a responsive user interface. Compared to traditional UI design, an immediate advantage of the card UI is that it provides a more personalized user experience, making the entire interface more simple, clear, and balanced.

All components of Degree Works will eventually be deployed using this metaphor. The label used to describe the version of the Dashboard available prior to 5.0.2 is simply “Dashboard”. This next generation introduced in 5.0.2 has been labeled the “Responsive Dashboard” to differentiate which dashboard is being referenced.

With 5.0.2, Ellucian also introduced the fourth generation of the Student Educational Planner component, which is generally referred to as “SEP”. The first generation is no longer in use by any customer, and the second generation has been called the “Classic Version”. With 4.1.0 and 4.1.1 (2011 and 2012) a “New” version, aka “Next Generation” version was introduced. All three of these versions have been called “SEP”. For simplicity, this newest generation of the Planner will be referred to as “SEP4”, and for clarity, the Classic Version will be referred to as SEP2 and the Next Generation Version will be referred to as SEP3 when we have to explain some key enhancement or improvement or change from one to another.

This Administration Guide provides instructions to setup, configure and debug the Responsive Dashboard application.

Initial Set-up and Configuration

This section describes the steps to configure and deploy Responsive Dashboard.

Authentication

The Responsive Dashboard user interface supports SHP, CAS and SAML authentication methods, as well as external access managers such as Oracle Access Manager. For more information, please see the *Authentication* section of the *Degree Works Technical Guide*.

Granting Users Access to Responsive Dashboard

Access to the features of the Responsive Dashboard is managed by assigning Shepherd keys and groups to users. This can either be done globally based on a user's role with SHPCFG, or individually on a user's shp_user_mst. For example, all students may be assigned the SRNSTU and SEPSTUVW Shepherd groups, which will grant limited access to Worksheets and Planner functionality. An advisor, however, might be assigned SRNADV and SEPADV, granting broader access to Worksheets, editing capabilities in Planner, and the ability to process exceptions. For more information, please see the *Access Control (Authorization)* section of the *Degree Works Technical Guide*.

Clients that are migrating from Dashboard to Responsive Dashboard should have little additional user setup to do; when deployed against the same Degree Works database, Responsive Dashboard uses the same SHPCFG and shp_user_mst configurations as Dashboard.

Shepherd Settings

Most configurations to manage the behavior of Responsive Dashboard can be maintained using the Configuration tab of Controller. See the *Degree Works Configuration Technical Guide* for more information on each setting listed below.

Review the following Shepherd settings to ensure they are configured correctly for your Responsive Dashboard environment:

```
classicConnector.serverNameOrIp
core.audit.*
core.classicUrl.serverDomain
core.courseLink.classicUrl
core.credit.format
core.gpa.format
core.gpa.maxLength
core.requisite.validate.enable
core.security.cas.callbackUrl spec=newdash (if you use CAS authentication)
core.security.referrerAllowableUrls
core.whatIf.*
dash.*
studentPlanner.*
```


Deploying the Responsive Dashboard

The Responsive Dashboard is designed to deploy equally well in on-premise and cloud hosted scenarios on all supported platforms. It provides a mobile-ready, responsive user interface communicating with the servers using lightweight, stateless, restful API over HTTP SSL.

The deployment artifacts are two JAR files which are executable using java and include an embedded container: Dashboard and APIServices. Dashboard is required while APIServices is only needed if using the Transfer Finder functionality or integrating with external applications to provide degree audit data. This section provides information on deploying Dashboard and APIServices.

Prerequisites

Java

Java version 8 or above is the only third-party dependency required to run this application. Oracle Java and OpenJDK are supported.

SSL

SSL is in general a best practice and is recommended for Responsive Dashboard because the security implementation uses stateless tokens. Using signed certificates is recommended for all deployment scenarios, even test or development environments. Self-signed certificates, even for internal test deployments, can be problematic because of browser requirements and warnings. The SSL certificate needs to be configured in a keystore and that keystore will need to be accessible in the deployment environment. There are various options for how to do this, for example using Java's keytool command.

Environment Settings

Several environment variables are required to be configured before running the application. Depending on the desired deployment platform there are many options for how these environment variables can be configured. The only requirement is that they are available in the environment where the product's JAR file will be executed. For example, it may be desirable to configure them in a shell script, in a specific user's profile, a startup instruction like init.d, or in a continuous deployment tool like Jenkins.

The minimum required variables are documented here. But many optional variables are provided in the Spring Boot platform, for example tuning the embedded container. Please refer to Spring documentation for those: <http://docs.spring.io/spring-boot/docs/current/reference/html/common-application-properties.html>.

Environment Variables are the recommended method to configure these properties. But there are various other ways these can be configured. Please refer to this documentation for more information: <http://docs.spring.io/spring-boot/docs/current/reference/html/boot-features-external-config.html>.

Required Environment Variables for Dashboard

- **SPRING_DATASOURCE_URL:** A JDBC connection string for the Degree Works database. For example: `jdbc:oracle:thin:@{SERVER}:{PORT}/{SID}` where `{SERVER}` is the address, `{PORT}` is the port of the listener, and `{SID}` is the SID of the database. This is the same database and schema used by the classic Degree Works software.
- **SPRING_DATASOURCE_USERNAME:** The Degree Works database username.
- **SPRING_DATASOURCE_PASSWORD:** The Degree Works database password. This can be encoded. To get the encoded value, use the `showdbpasswords --password` command in the classic environment. It will display the encoded value to place here (e.g. `ENC(skdfjldjs)`).
- **SERVER_PORT:** The desired port for the Responsive Dashboard. For example, "443". If using 443, then your links or bookmarks to the application don't need to specify a port in the URL because it is the standard port browsers will assume for URLs that begin with `https://`.
- **SERVER_CONTEXT_PATH:** An optional context path. If this is blank, the application will be deployed at the root URL like `https://ResponsiveDashboard.myserver.tld`. If a value is specified like `/my/context/path/`, then the application would be accessible at a URL like `https://ResponsiveDashboard.myserver.tld/my/context/path/`.
- **SERVER_SSL_KEY_STORE:** The path to a keystore file you created to contain your SSL certificate.
- **SERVER_SSL_KEY_STORE_PASSWORD:** The password for your keystore file.
- **SERVER_SSL_KEYSTORETYPE:** The type of your keystore. JKS is the default.
- **SERVER_SSL_KEY_ALIAS:** The alias of the key you created.

Required Environment Variables for APIServices

- **DW_DATASOURCE_URL:** A JDBC connection string for the Degree Works database. For example `jdbc:oracle:thin:@{SERVER}:{PORT}:{SID}` where `{SERVER}` is the address, `{PORT}` is the port of the listener, and `{SID}` is the SID of the database.
- **DW_DATASOURCE_USERNAME:** The Degree Works database username.
- **DW_DATASOURCE_PASSWORD:** The Degree Works database password.
- **SERVER_PORT:** The desired port for APIServices. For example "443". This must be different than the ports used by TransferEquivalencyUI, ApplicantAPI and Gateway.
- **SERVER_CONTEXT_PATH:** An optional context path. If this is blank, the application will be deployed at the root URL like `https://services.myserver.tld`. If a value is specified like `/my/context/path`, then the application would be accessible at a URL like `https://services.myserver.tld/my/context/path`.
- **SERVER_SSL_KEY_STORE:** The path to a keystore file you created to contain your SSL certificate.
- **SERVER_SSL_KEY_STORE_PASSWORD:** The password for your keystore file.
- **SERVER_SSL_KEYSTORETYPE:** The type of your keystore. JKS is the default.
- **SERVER_SSL_KEY_ALIAS:** The alias of the key you created.

Optional Environment Variables

- **JAVA_OPTIONS:** The memory allocation for the application can be defined using this variable, and is recommended. The value defines the initial heap size and max heap size and is in the following format: `-Xms1536m -Xmx1536m`. The heap size values should be adjusted as appropriate for your server.
- **SERVER_TOMCAT_REDIRECT_CONTEXT_ROOT:** Embedded Tomcat for Spring Boot has a default behavior to redirect a request without a trailing slash to one with a trailing slash. However, that redirect happens without evaluating forward headers like X-Forwarded-Proto. So, when SSL offloading is in place, the redirect happens to http instead of https. This can be an issue especially for load balanced environments. To disable the default behavior, set this variable to “false”.
- **DEBUG:** A boolean “true” or “false” which will enable or disable debug logging. The log file is by default named dashboard.log. But, the location and name of that file can be customized by setting the LOGGING_PATH and LOGGING_FILE environment variables.
- **DEPLOY_LOCATION:** the location of the Dashboard.jar file. This location should be set then referenced in the “java -jar” command that starts the application.
- **LOGGING_FILE:** the name of the file including the path location. A date is appended to this name to produce the actual file name (see LOG_DATE_PATTERN below). If this is specified, LOG_PATH will not be used.
- **LOG_PATH:** the location of the log file. The default PATH location is the Java temporary java folder from the Java property `java.io.tmpdir`. This is usually `/tmp`. The file name will consist of this value plus “dashboard.log” plus a date (see LOG_DATE_PATTERN below).
- **LOG_DATE_PATTERN:** the date pattern to use as the suffix for the file. This is used to ensure that the file does not get too big. It will create a new file when the date pattern would normally produce a different value. The date patterns follow the Java SimpleDateFormat. See <https://docs.oracle.com/javase/8/docs/api/java/text/SimpleDateFormat.html> for details. The default is yyyy-MM-dd (e.g. 2020-01-31). This will create a new file each day. For a monthly cycle, you may use yyyy-MM (e.g. 2020-01), and for a weekly cycle use yyyy-ww (e.g 2018-01).

Database Pool Configuration for Microservice Applications

For production deployments of java applications, it is important to be able to configure the size of the database connection pool. In Tomcat this is done via the Resource definition in server.xml. For microservice applications like Responsive Dashboard and API Services, there are environment variables that can be used for this configuration.

These environment variables start with “dw_datasource_”. They end with the name of the datasource pooling attribute you want to control. These are the most common ones:

Attribute	Meaning
initialSize	The initial number of connections that are created when the pool is started.
maxActive	The maximum number of active connections that can be allocated from this pool at the same time.

maxIdle	The maximum number of connections that can remain idle in the pool, without extra ones being released.
minIdle	The minimum number of active connections that can remain idle in the pool, without extra ones being created.

To set the maximum number of active connections in the pool, for example, put the following line in your startup script:

```
export dw_datasource_maxActive=100
```

You can set any of the properties of a version 1.4 Apache Commons BasicDataSource object. These are documented at <http://commons.apache.org/proper/commons-dbcp/api-1.4/org/apache/commons/dbcp/BasicDataSource.html>.

The number of active connections, as well as the current configured values for the limits can be monitored using any JMX (Java Management Extensions) client, such as JConsole, attached to the microservice JVM. The values will appear under the mbean net.hedtech.degreeworks/JmxBasicDataSource/datasource. The getState operation will return a formatted report of the current pool state.

Deployment

Once the prerequisites and configuration are in place, start up the application using a "java -jar" command like this one:

```
java $JAVA_OPTIONS -jar ResponsiveDashboard.jar
java $JAVA_OPTIONS -jar APIServices.jar
```

Example Startup Script for Dashboard

If it is desirable to employ a shell script to start the application, here is an example. Note that the Dashboard.jar file must be staged at the DEPLOY_LOCATION:

```
#!/bin/ksh
export
SPRING_DATASOURCE_URL="jdbc:oracle:thin:@MyServer:1521/service_name"
export SPRING_DATASOURCE_USERNAME="my-dw-user"
export SPRING_DATASOURCE_PASSWORD="my-dw-password"
export SERVER_PORT="443"
export SERVER_CONTEXT_PATH=""
export SERVER_SSL_KEY_STORE="/usr/local/my-keystore.jks"
export SERVER_SSL_KEY_STORE_PASSWORD="my-key-password"
export SERVER_SSL_KEYSTORETYPE="jks"
export SERVER_SSL_KEY_ALIAS="my-keystore-alias"
export DEPLOY_LOCATION="/path/to/jarfile"
export JAVA_OPTIONS="-Xms1536m -Xmx1536m";
export DEBUG="false"
java -jar $DEPLOY_LOCATION/ResponsiveDashboard.jar > startup-log-
file.log &
```

Example Stop Script for Dashboard

If using a start script like the example above, here is a corresponding example stop script:

```
#!/bin/ksh
pkill -f ResponsiveDashboard
```

Example Startup Script for APIServices

If it is desirable to employ a shell script to start the application, here is an example. Note that the APIServices.jar file must be staged at the DEPLOY_LOCATION:

```
#!/bin/ksh
export
DW_DATASOURCE_URL="jdbc:oracle:thin:@MyServer:1521/service_name"
export DW_DATASOURCE_USERNAME="my-dw-user"
export DW_DATASOURCE_PASSWORD="my-dw-password"
export SERVER_PORT="8443"
export SERVER_CONTEXT_PATH=""
export SERVER_SSL_KEY_STORE="/usr/local/my-keystore.jks"
export SERVER_SSL_KEY_STORE_PASSWORD="my-key-password"
export SERVER_SSL_KEYSTORETYPE="jks"
export SERVER_SSL_KEY_ALIAS="my-keystore-alias"
export DEPLOY_LOCATION="/path/to/jarfile"
export JAVA_OPTIONS="-Xms1536m -Xmx1536m";
export DEBUG="false"
java -jar $DEPLOY_LOCATION/APIServices.jar > startup-log-file.log &
```

Example Stop Script for APIServices

If using a start script like the example above, here is a corresponding example stop script:

```
#!/bin/ksh
pkill -f APIServices
```

Integration with Portals

When connecting from your school's portal to the Responsive Dashboard you can pass in the ID of the student that is to appear in Degree Works. This may be the student user that has clicked the link to Degree Works from your portal or it may be an advisor user viewing this student in your portal. The student ID needs to be passed in the URL used to connect from your portal to Degree Works using this format:

```
https://<your server name>/<path to Degree Works>?studentId=<someid>
```

Only users who have been given permission to access this student's records will be successful in seeing this student loaded when they connect to Degree Works. If the user is an advisor with a fixed set of advisees, once the advisees have been loaded this specific student will appear as the selected student. If the user does not have the SDFIND or SDSTUANY keys they will not be able to switch to

a new student within the Responsive Dashboard; switching students would then have to occur within your portal.

Examples:

Link to the Worksheets tab in Responsive Dashboard:

```
https://{server}/{deployment-path}?studentId=<someid>
```

Link to the Exceptions tab in Responsive Dashboard:

```
https://{server}/{deployment-path}/exceptions?studentId=<someid>
```

Link to the Plans tab in Responsive Dashboard:

```
https://{server}/{deployment-path}/plans?studentId=<someid>
```

Accessing APIs from External Applications

The Degree Works APIs that provide access to articulations, audits, plans, and requirements can be integrated with external applications like CRM Advise and Banner's Registration and CPoS processes. See the Ellucian API Documentation on the Customer Center for more information.

Authentication

Stateless token authentication is recommended when accessing Degree Works APIs, but some applications will need to use HTTP Basic authentication over SSL. For these applications, you will need to create a user access record in Degree Works.

For example, if you use Degree Works plans with Banner registration, use the Users function of Controller to create a record for the username and password defined on SOAREST. We recommend assigning the API Rest Users user class, and at a minimum, assigning the RSPLAN, ANYSTUID and NOREFER keys.

External Course Information

Additional course information from either Banner or an external source like Leepfrog's CourseLeaf can be configured to display when a user clicks on a course in the worksheet or in the planner. For information on displaying course information from Banner with CourseLink, please see the Worksheets > Academic > Course Link and Worksheets > Plans > Course Link sections.

When configured to display external course information a new tab or window will be opened, based on the user's browser configuration. The URL for the external source should be appended with tags defining the course. These optional tags will be replaced with the actual course details when the request is made.

- {courseDiscipline} is replaced with the course discipline
- {courseNumber} is replaced with the course number
- {attributeValue} is replaced with the attribute code that is part of the WITH qualifier. For example, 1 Class in MATH 2@ (With Attribute=ABCD)
- {attributeOperator} is replaced with the attribute operator that is part of the WITH qualifier. Usually this is "=" (equals) or "<>" (not equals).

For example, if this URL is configured as:

```
https://externalcourseinformation.myserver.tld/?sDisc={courseDiscipline}&sNumber={courseNumber}&attribute={attributeValue}&attributeOperator={attributeOperator}
```

The request URL for CHEM 106 in the worksheet or the planner would be:

```
https://externalcourseinformation.myserver.tld/?sDisc=CHEM&sNumber=106&attribute=&attributeOperator=
```

For CHEM 2@ (With Attribute=WRIT) the URL would be:

```
https://externalcourseinformation.myserver.tld/?sDisc=CHEM&sNumber=2@&attribute=WRIT&attributeOperator==
```

Configuration

- Review the *Show CourseLink* flag in RPT036 to ensure it is configured correctly.
- Review the following Shepherd settings to ensure they are configured correctly:
dash.courseInformation.enabled
dash.courseInformation.url

Using Responsive Dashboard and Dashboard concurrently

You can deploy the Responsive Dashboard in the same environment as the Dashboard and use both applications concurrently. This is useful if you want to grant access to a set of users to test the Responsive Dashboard interface. Note that when pointing to the same database and classic environment, both applications use the same underlying infrastructure – Shepherd keys and groups, most Shepherd settings, UCX tables, etc. – so any changes made for one application will likely impact the other.

Responsive Dashboard Navigation

The header bar at the top of Responsive Dashboard helps users navigate through the application. The user can access the functionality to which they have been given access by clicking the link for that item.

Configuration

- Access to Worksheets is granted if the user has the `SDAUDIT` Shepherd key.
- Access to Exceptions is granted if the user has the `SDEXCEPT` Shepherd key.
- Access to Plans is granted if the user has the `SEPPLAN` Shepherd key.
- Access to Exception Management (under Admin) is granted if the user has the `SDEXPMGT` Shepherd key.
- Access to Debug (under Admin) is granted if the user has the `DEBUG` Shepherd key.
- Access to Theme (under Admin) is granted if the user has the `SDADMIN` Shepherd key.

Mega Menu

Users can also navigate throughout the application using the Mega Menu on the right of the header bar. Note that for users on a mobile device, only the mega menu will be available for navigation.

Logout

To log out of Responsive Dashboard, select Sign Out from the persona dropdown button on the right of the header bar.

Student

Persistent on all pages that focus on a student (i.e. Worksheets, Exceptions and Plans) is a student data card. Depending on the user's access, they can enter a known student ID, see a pre-loaded list of advisees or assigned students, and search for students.

Find Students

If a user has access to "Advanced search", a dialog will open that allows the user to select criteria specific to the students for which they would like to search for. The user will always have the option to search by a student's ID, first name and last name. Curriculum data can also be configured as search options. Degree, Level, Classification, Catalog year and Degree source will only allow for one value to be selected while the other curriculum data will allow for multiple values to be selected. Multiple selections in the same data item will be an AND. This means that if you choose majors of both ART and CHEM the search results will only return students who have both majors – either on the same degree or different degrees. Selections from different data items are also an AND. For example, if you select Degree=BS and Major=ART and Major=CHEM, you are saying to select any student with the BS degree and both the MATH major and the CHEM major, so BS degree students with dual majors of ART and CHEM (and possibly additional majors). You can reset your select options to the default by clicking "Clear".

Click "Search" to execute the search. All students who meet the search criteria will be displayed in the bottom window of the Find Students search dialog. The list of students is sorted by name in descending order as a default but can be sorted by clicking on any of the column headers in the results grid. By default, all students in the results list are selected; unchecking the check box to the left of the student will deselect them.

Click "Select" to close the search window and load the list of students into Degree Works. You may then select any student from the list to process a degree audit, apply an exception or manage a plan.

Click "Cancel" to discard your search and go back to the originating page.

Some institutions may limit advisors to having access only to their list of advisees or department heads to having access only to students in their department. In this case, a preselected list of students will appear in the "Select student" dropdown and the "Advanced search" functionality will not be available. Please refer to the "Additional Advisee Filtering" section of the *Degree Works Technical Guide* for additional information.

Configuration

- Review the following UCX tables that populate the Advanced search dropdowns: STU307 (Degree), STU350 (Level), STU305 (Classification), STU035 (Catalog year), STU024 (Major), STU023 (Minor), STU560 (College), STU323 (Specialization), STU563 (Concentration), STU324 (Liberal learning), STU316 (Program), STU306 (Student type).
- Review the following Shepherd settings to ensure they are configured correctly:
`dash.search.catalogYear.enabled`

```
dash.search.college.enabled
dash.search.concentration.enabled
dash.search.degree.enabled
dash.search.degreeSource.enabled
dash.search.level.enabled
dash.search.liberalLearning.enabled
dash.search.major.enabled
dash.search.maximum
dash.search.minor.enabled
dash.search.program.enabled
dash.search.school.enabled
dash.search.specialization.enabled
dash.search.studentId.mask
dash.search.studentStatus.enabled
```

- Access to search by ID is granted if the user has the `SDFINDID` and `ANYSTUID` Shepherd key.
- Access to Advanced search is granted if the user has the `SDFIND` and `ANYSTUID` Shepherd keys.

Student Data

Once a student has been selected, the student's information is loaded into the student data area. This includes the Student ID, Degree and other goal data. Only goal data items for which the student has a record will display; for example, if the student doesn't have a concentration, the Concentration label will not display. Additionally, custom data can be configured to display in the student data area.

Localizing Student Data

By default any goal data that a student has will display. If there is a goal type that you do not want to display for all students, you can localize this with Composer. In `DashboardMessages.properties`, change the `dash.home.context.*Format` to a blank value. For example, if you do not want Specialization to display, change this

```
dash.home.context.SPECFormat={label} {value}
```

to

```
dash.home.context.SPECFormat=
```

Displaying Custom Student Data

Data from the `rad_report_dtl` and `rad_custom_dtl` can be configured to display in the student data card. Any item defined in SCR002 and stored on the `rad_custom_dtl` for the student, or any item defined in RPT046 and stored on the `rad_report_dtl` for the student can be displayed. Like goal data, only custom and report items for which the student has a record will display.

Configuration has several steps. First, set the `dash.studentHeader.custom.enabled` setting to true to enable the display of custom data. Then define the items to display and the order in which to display them in the `dash.studentHeader.custom.items` setting. And finally, use Composer to add localized properties for the custom data labels. These should be on `DashboardMessages.properties` and each custom data item should have a property in the format of `dash.studentHeader.custom.xxxx`, where “xxxx” is the lowercase code from SCR002 or RPT046.

Configuration

- Review SCR002 and RPT046 to ensure they are configured correctly.

- Review the following Shepherd settings to ensure they are configured correctly:

```
dash.studentHeader.custom.enabled
```

```
dash.studentHeader.custom.items
```

Refreshing Student Data

For Banner and Colleague clients, a user may be given access to refresh the student's data by performing a dynamic extract. If enabled to display, the "Data refreshed" date and time indicate the last time the student's academic data was copied from the SIS into Degree Works while the refresh button allows a user to once again pull in this student's data. Once the extract process completes the user will get a confirmation message and the "Data refreshed" date/time will be updated and a new audit is automatically generated.

For additional information on this process, please refer to the "Dynamic Data Extract Process" section of the *Banner Considerations Technical Guide* or the *Colleague Considerations Technical Guide*.

Configuration

- Access to see the refresh data and time is granted if the user has the `SDREFRES` Shepherd key.
- Access to the refresh button is granted if the user has the `SDREFBTN` Shepherd key.

Tools

Functions that are specific to the student context (i.e. Worksheets, Exceptions and Plans) have a common set of tools that are accessible on each page. Using a set of icons in the upper right, the user can print audits, get student or advisor contact information, use GPA calculators, view class history data, and manage notes and petitions.

Print

The Print icon will display on any page that displays an audit – i.e. Academic or What-If on Worksheets, or Audit on Plans. Clicking the button will open the browser's print dialog. The user can choose to send the output to a printer, save the output as a PDF, or any other option available in the browser.

Contact

The Contact icon opens a dialog that displays either the student's or advisor's email address. Clicking on the email address will open the user's default email program with the contact's email address loaded in the To: field. Typically, a student will see their advisor's email address on the Contact dialog, while advisors and other administrative users will see the email address for the selected student.

GPA Calculator

The GPA Calculators are located under the More menu. There are three different GPA calculators: Graduation, Term, and Advice. The calculators can help students set realistic goals at the beginning of the term or academic career and calculate their end-of-term GPA using actual academic information.

Configuration

- Review the *Graded Attempted* and *Use in GPA Calculators* flags in STU385 to ensure they are configured correctly.
- Review the following Shepherd settings to ensure they are configured correctly:
`dash.gpaCalculator.decimals`
`dash.gpaCalculator.round`
- Access to GPA Calculators is granted if the user has the `SDGPACLC` Shepherd key.
- Access to the Graduation Calculator is granted if the user has the `SDGPAGR` Shepherd key.
- Access to the Term Calculator is granted if the user has the `SDGPATRM` Shepherd key.
- Access to the Advice Calculator is granted if the user has the `SDGPAADV` Shepherd key.

Graduation Calculator

This calculator gives the student a general view of the average GPA they will need to earn over their final "X" credits in order to achieve their desired GPA. In some cases, the student may be informed that their desired GPA is not possible to achieve, considering their number of credits remaining. This calculator helps students to set long-term general goals.

Graduation Calculator inputs:

- Current GPA (defaults from the rad_cum_gpa on the rad_term_dtl)
- Credits Remaining (defaults from the most recent academic audit)
- Credits Required (defaults from the most recent academic audit)
- Desired GPA

Term Calculator

This calculator allows a student to predict what their GPA will be after the current term is completed. If the student has coursework in the current term on their most recent academic audit, these classes and their credits will automatically load into the calculator. Classes can be deleted, and additional classes can be added.

Term Calculator inputs:

- Current GPA (defaults from the rad_cum_gpa on the rad_term_dtl)
- Credits Earned (defaults from the rad_cum_gr_att on the rad_term_dtl)
- Class information
- In-progress classes

Advice Calculator

This calculator is used to figure out how a student can raise or lower their GPA using actual grades as advice. Note that the Graded Attempted flag in STU385 is used to filter out grades – only those grades with Graded Attempted = Y are used in the Advice Calculator – allowing you to exclude P grades, for example.

Advice Calculator inputs

- Current GPA (defaults from the rad_cum_gpa on the rad_term_dtl)
- Credits Earned (defaults from the rad_cum_gr_att on the rad_term_dtl)
- Desired GPA

Class History

Class History is located under the More menu. It is a listing of all the student's coursework, organized by term.

Configuration

- Access to Class History is granted if the user has the `SDXML31` Shepherd key.

Petitions

Located under the More menu, Petitions allows users to enter requests for exceptions. A petition can be a request to have a particular requirement modified or waived for a particular student.

A preview of all petitions that have been entered for the student displays initially. The list can be filtered by the petition status – either waiting for approval, approved, applied as exceptions or rejected. The date the petition was created, the author of the petition and the petition status will display for each petition.

Click “Add a new petition” to add a new petition. Once you have entered the text for the petition, click “Save petition” to save the petition to the database. A message will appear telling you your petition was added successfully. All new petitions have a default status of waiting for approval when first created. Once a petition is created, it can be acted upon by registrar-class users or users having access to Exceptions Management.

To see all the text of a petition, select “View petition” from the action menu in the right corner of a petition. Click “Back” to go back to the petition list. From this dialog, you can also edit or delete the petition.

To modify a petition, select “Edit petition” from the action menu in the right corner of a petition. Click “Save petition” to save your changes. Click “Cancel” to discard your changes and go back to the petition list. From this dialog, you can also delete the petition.

To delete a petition, select “Delete petition” from the action menu in the right corner of a petition. Click “Delete petition” to delete the petition. Click “Cancel” to keep the petition and go back to the petition list.

Petitions can be configured to display on an audit. If enabled, they will be in the Notes section.

Configuration

- Review the *Show Petitions* flag in RPT036 to ensure it is configured correctly.
- Access to Petitions is granted if the user has the `SDPETVIEW` Shepherd key.
- Access to add petitions is granted if the user has the `SDPETADD` Shepherd key.
- Access to modify petitions is granted if the user has the `SDPETMOD` Shepherd key.
- Access to delete petitions is granted if the user has the `SDPETDEL` Shepherd key.

Petition Email Notification (petsend)

Degree Works can be configured to send an email that notifies the recipient that there are new petitions either waiting approval or approved petitions waiting to be applied as exceptions.

However, if you are a Banner school using Workflow to manage petitions you will not want to use this email notification as Workflow takes care of this for you.

The petsend Perl script is used to configure where the notification email is to be sent. Below is an example on how to run petsend to send notification emails:

Example:

```
petsend someaddress@yourschool.edu WAITING
petsend someaddress@yourschool.edu APPROVED
```

The Reply To email address is set to the same email address as the Send To address by default as shown below:

```
$ReplyToEmailAddress = $ToEmailAddress;
```

To modify the Reply To e-mail address, change the \$ToEmailAddress in the petsend script to the e-mail address you wish replies to be sent to as shown below:

```
$ReplyToEmailAddress = "computer_center\@yourschool.edu";
```

You can also configure petsend to carbon copy (CC) one or more individuals at your institution.

To activate the CC function, modify the following line with the e-mail address of the person you wish to CC:

```
$CC = "someone_else\@yourschool.edu";
```

The petsend job can be configured not to send a notification e-mail if there are no petitions either waiting approval or waiting to be applied. To disable sending e-mails under these circumstances, change the value for SEND_ZERO_PETITIONS_MSG to "N" as shown below:

```
$SEND_ZERO_PETITIONS_MSG = "N";
```

You can edit the content of the notification e-mail by modifying the DefineSubjectAndBody function of the petsend script.

Notes

Located under the More menu, Notes allows users to document academic advising on student records. Notes can be marked as internal so that students will not see. Notes made available to the student appear in audit reports in a Notes section at the bottom of the report.

A preview of all notes that have been entered for the student displays initially. The date the note was created, the author of the note and whether the note is internal will display for each note.

Click “Add a new note” to add a new note. Enter your note text or select a predefined note if this functionality has been enabled. Users with the `SDNTECHG` key can also enter additional text to the predefined note in the note field. To make a note internal, check the “Not available to student” checkbox if this functionality has been enabled. Click “Save note” to save the note to the database. A message will appear telling you your note was added successfully. If the user has access, a new academic audit will be automatically generated when adding a note.

To see all the text of a note, select “View note” from the action menu in the right corner of a note. Click “Back” to go back to the note list. From this dialog, you can also edit or delete the note.

To modify a note, select “Edit note” from the action menu in the right corner of a note. Click “Save note” to save your changes. Click “Cancel” to discard your changes and go back to the note list. From this dialog, you can also delete the note.

To delete a note, select “Delete note” from the action menu in the right corner of a note. Click “Delete note” to delete the note. Click “Cancel” to keep the note and go back to the note list.

Configuration

- Review CFG071 to ensure it is configured correctly.
- Review the following Shepherd settings to ensure they are configured correctly:
 - `dash.notes.internal.enabled`
 - `dash.notes.predefined.enabled`
 - `dash.audit.process.inprogress.default`
 - `dash.audit.process.preregistered.default`
- Access to Notes is granted if the user has the `SDNOTES` Shepherd key.
- Access to view notes is granted if the user has the `SDNTEVUE` Shepherd key.
- Access to add notes is granted if the user has the `SDNTEADD` Shepherd key.
- Access to modify notes is granted if the user has the `SDNTEMOD` Shepherd key.
- Access to delete notes is granted if the user has the `SDNTEDEL` Shepherd key.
- Access to add text to predefined notes is granted if the user has the `SDNTECHG` Shepherd key.
- Access to automatically run a new audit after adding a note is granted if the user has the `SDNTERUN` Shepherd key.

Worksheets

A user can view and run a variety of audit reports in Worksheets. Each audit report displays specific information about students and their progress towards degree completion. After a student is selected, the most recent audit for the student will load automatically. For more information about creating audits, please refer to the *Scribe Language User Guide* and the *Degree Works Technical Guide*.

Configuration

- Access to Worksheets is granted if the user has the SDAUDIT Shepherd key.

Academic

The Academic audit is the primary audit used to track and evaluate a student's progress towards a degree.

Academic Audit

The academic audit is a series of blocks that define degree and program requirements. All blocks initially load as expanded in a desktop view; for mobile users all blocks initially load as collapsed. You can manually expand or collapse individual blocks by clicking the arrow in the upper right of each block or expand/collapse all block by clicking the link at the top of the audit.

While the content of the academic audit is driven by the requirements defined in Scribe, the appearance is controlled by the settings in RPT036 for each report format. For example, Show Label controls whether the completeness icons and labels should display on the audit, while Show Legend and Show Disclaimer control whether to display the legend and disclaimer on the audit. You should review all of these flags to ensure they are set correctly for the audit formats you use.

The default behavior for block header data in each block type can be configured in SCR004. Additionally, HeaderTags can be scribed to override the flags as needed. Please see the SCR004 section of the *Configuration Technical Guide* and the HeaderTag section of the *Scribe Language User Guide* for more information.

Configuration

- Review RPT036 to ensure it is configured correctly for WEB30, WEB31, WEB33, WEB44, WEB50, WEB51, WEB55, WEB56 and SEP31 as well as any custom formats you may have.
- Review the *Show.** flags in SCR004 for each block type.

Format

The Format dropdown allows the user to select in which format the audit report should be displayed. There are several different report formats available, although not all formats are

appropriate for all user classes. For example, many schools choose to not make the Registrar Report available to students. These report formats can be customized by the client to meet their respective needs. Additional audit reports can also be created and added to the drop-down list box.

Configuration

- Access to the Registrar Report is granted if the user has the `SDWEB30` Shepherd key.
- Access to the Student View is granted if the user has the `SDWEB31` Shepherd key.
- Access to the Graduation Checklist is granted if the user has the `SDWEB33` Shepherd key.
- Access to the Registration Checklist is granted if the user has the `SDWEB36` Shepherd key.
- Access to custom report formats is granted if the user has the `SDWEB34`, `SDWEB35` or `SDWEB37` Shepherd keys.

Degree Progress

Degree progress can be measured in percentage of degree requirements met, the percentage of credits completed and the overall GPA. The values are calculated by the auditor based on the requirements and qualifiers defined in the audit's blocks. For each report format, displaying the requirements and credits progress is configurable.

Configuration

- Review the *Show Progress Bar Requirements* and *Show Progress Bar Credits* flags in RPT036 to ensure they are configured correctly for each report.

Process Audit

If the user has access, they may generate a new audit by clicking on the "Process" button. Options to select whether to include in-progress and preregistered as well as the default selection for these options can also be configured. Note that if the in-progress and preregistered checkboxes are configured to not display but the user can still process a new audit, the auditor will follow the default value settings to determine whether in-progress and preregistered classes should be included.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly:
`dash.audit.process.inprogress.defaultValue`
`dash.audit.process.inprogress.enabled`
`dash.audit.process.preregistered.defaultValue`
`dash.audit.process.preregistered.enabled`
- Access to process an audit is granted if the user has the `SDAUDRUN` Shepherd key.

Historic Audits

"View historic audits" allows a user to view historical audits for a student. The maximum number of historical audits to keep per student per School/Degree combination can be configured. If the user has access to view historic audits, they can select the audit they wish to view from the dropdown and it will load using the currently selected report format.

Configuration

- Review the *Audit History Count* flag in CFG020 DAP14 to ensure it is configured correctly.
- Access to view historic audits is granted if the user has the `SDHIST` Shepherd key.

Diagnostics

Diagnostics allows a user to view additional information about an audit and can be used to troubleshoot audit issues. If the user has access to Diagnostics, when they click on the link a new tab or window with the diagnostics audit will open.

Configuration

- Access to Diagnostics is granted if the user has the `SDXML30` Shepherd key.

Student Data

Student data is a report of the what is in the Degree Works database for the selected student. This can be useful in troubleshooting issues with the audit. If the user has access to Student data, when they click on the link a new tab or window with the student data report will open.

Configuration

- Access to Student Data is granted if the user has the `SDXML32` Shepherd key.

Save Audit

Audits may be saved or “frozen” so they do not get deleted when the maximum CFG020 DAP14 Audit History Count has been reached. Frozen audits do not count against the history count value. Depending on the user’s access, when they click on “Save audit” a dialog will open that prompts for the Freeze type and a Description. When the user clicks “Save”, the audit will be saved in the database with the specified freeze type and description, and the screen will be updated with the user who saved the audit, the date, the freeze type and description. For more information, please refer to the “Freezing Audits” special topic in the *Degree Works Technical Guide*.

Configuration

- Access to freeze audits is granted if the user has the `AUDFREEZ` Shepherd key.
- Access to add a description is granted if the user has the `AUDESCR` Shepherd key.

Delete Audit

Audits that have been saved or “frozen” can be deleted if the user has access to do so. Clicking on the “Delete audit” link opens a confirmation dialog. If the user clicks “Delete”, the audit will be deleted from the database. If the user clicks “Cancel”, they will be returned to Worksheets without deleting the audit.

Configuration

- Access to delete audits is granted if the user has the `SDAUDEL` Shepherd key.

Course Link

Course Link displays additional information about a course or courses, such as description, prerequisites and sections. When enabled, still needed courses on an audit will appear as a link and clicking on this link will open a dialog with the additional course information. For more information, please refer to the “Course Link” section in the *Degree Works Technical Guide*.

Configuration

- Review the *Show CourseLink* and *CourseLink*Order* flags in RPT036 to ensure they are configured correctly.

- Review the following Shepherd settings to ensure they are configured correctly:

```
dash.courseInformation.enabled
```

```
dash.courseLink.meetingTime.24hour
```

What-If

What-If audits allow you to process speculative degree audits for a student using their current class history. You can audit a student against the requirements for a different major, minor, degree, catalog year or other goal data as well as add classes that may be taken in the future. Responsive Dashboard can be configured to use curriculum rules to filter the goal data dropdowns. If Responsive Dashboard is not configured to use curriculum rules for filtering, the goal data dropdowns will contain all values from the associated UCX table.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly:

```
core.whatIf.enableCurriculumRules
```

- Access to What-If is granted if the user has the `SDWHATIF` Shepherd key.

What-If Analysis with Standard UCX Filtering

When Responsive Dashboard is configured to not use curriculum rules, standard UCX filtering will be used in the What-If Analysis dropdowns. Level, Degree and Catalog year are required goal data items for a What-If and will default to the student’s data. The options that display in “Areas of study” are configurable.

The Catalog year will always be displayed and is a required field. The catalog years that are in the dropdown come from the codes in STU035 with Web What-If flag = Y. If a default catalog year is configured, this will initially be displayed instead of the student’s catalog year. Additionally, the site may be configured to not allow the default catalog year to be changed.

If configured to display, the Level will be a primary item and is a required field. The values in the level dropdown will be all STU350 values with Show in What-If flag = Y or blank.

If configured to display, the College will be a primary item and is a required field. If showProgramAdditional = N or mustUsePrimaryRule = D or N, the college will also be available as a secondary item. The values in the college dropdown will be all STU560 values with Show in What-If flag = Y or blank.

The Degree will always display as a primary item and is a required field. The values in the degree dropdown will be all STU307 values with Show in What-If flag = Y or blank.

If configured to display, the Major will be both a primary and secondary item, and can be configured to be a required primary item. The values in the major dropdown will be all AUD027 values.

If configured to display, the Concentration will be both a primary and secondary item. The concentration can be required if the selected primary major has AUD027 Concentration Required = Y. The values in the concentration dropdown list will be all STU563 values with Show in What-If flag = Y or blank.

If configured to display, the Minor will be both a primary and secondary item. The values in the minor dropdown will be all AUD029 values.

Configuration

- Review AUD027 and AUD029 to ensure they are configured correctly.
- Review the *Show in What-If* flag in the following tables to ensure it is configured correctly: STU307 (Degree), STU350 (Level), STU035 (Catalog year), STU560 (College), STU323 (Specialization), STU563 (Concentration), STU324 (Liberal learning), STU316 (Program).

- Review the following Shepherd settings to ensure they are configured correctly:

```
core.whatIf.changeCatalogYear
core.whatIf.defaultCatalogYear
core.whatIf.majorRequired
core.whatIf.showCollege
core.whatIf.showConcentration
core.whatIf.showLiberalLearning
core.whatIf.showMajor
core.whatIf.showMinor
core.whatIf.showProgramAdditional
core.whatIf.showSchool
core.whatIf.showSpecialization
```

What-If Analysis with Curriculum Rule Filtering

When Responsive Dashboard is configured to use curriculum rules, the values that appear in the dropdown lists will come from the curriculum rules loaded in rad_currule_dtl. The Catalog year and Degree dropdowns will always display, while Campus, Program, Level, College, Major, Concentration and Minor can be configured to display.

The Catalog year will always be displayed and is a required field. The catalog years that are in the dropdown come from the codes in STU035 with Web What-If flag = Y. If a default catalog year is configured, this will initially be displayed instead of the student's catalog year. Additionally, the site may be configured to not allow the default catalog year to be changed.

If configured to display, the Campus will be a primary item and is a required field. The values in the campus dropdown will only be those from STU576 that are on the selected curriculum rule.

If configured to display, the Level will be a primary item and is a required field. The values in the level dropdown will only be those from STU350 with Show in What-If flag = Y or blank and are on the selected curriculum rule.

If configured to display, the College will be a primary item and is a required field. If showProgramAdditional = N or mustUsePrimaryRule = D or N, the college will also be available as a secondary item. The values in the college dropdown will only be those from STU560 with Show in What-If flag = Y or blank and are on the selected curriculum rule. If the college is configured to display before the degree, college will drive the degree in the curriculum rule filtering.

The Degree will always display as a primary item and is a required field. If showProgramAdditional or mustUsePrimaryRule are false, the Degree will also be available as a secondary item. The values in the degree dropdown will only be those from STU307 with Show in What-If flag = Y or blank and are on the selected curriculum rule. If the degree is configured to display before the college, degree will drive the college in the curriculum rule filtering.

If the Program is configured to display as a primary item, it will be a required field and the Level, Degree, and College (if displayed) will be inactive. The program may also be configured to display in the secondary area. If configured to display in the secondary area, the program can be configured to be a required field. The values in the program dropdown will only be those from STU316 with Show in What-If flag = Y or blank and are on the selected curriculum rule.

If configured to display, the Major will be both a primary and secondary item, and can be configured to be a required primary item. The values in the major dropdown will only be those from AUD027 that are on the selected curriculum rule.

If configured to display, the Concentration will be both a primary and secondary item. The concentration can be required if the selected primary major has AUD027 Concentration Required = Y. The values in the concentration dropdown will only be those from STU563 with Show in What-If flag = Y or blank and are on the selected curriculum rule. Additional configuration will require that the concentration be tied to a specific major. If showing all concentrations on a rule or major is enabled, all STU563 values with Show in What-If flag = Y or blank will be displayed.

If configured to display, the Minor will be both a primary and secondary item. The values in the minor dropdown list will only be those from AUD029 that are on the selected curriculum rule. If showing all minors on a rule is enabled, all AUD029 values will be displayed.

Configuration

- Review AUD027 and AUD029 to ensure they are configured correctly.
- Review the *Show in What-If* flag in the following tables to ensure it is configured correctly: STU035 (Catalog year), STU307 (Degree), STU316 (Program), STU350 (Level), STU560 (College), STU563 (Concentration), STU576 (Campus).
- Review the following Shepherd settings to ensure they are configured correctly:


```
core.whatIf.changeCatalogYear
core.whatIf.concTiedToMajor
core.whatIf.defaultCatalogYear
core.whatIf.degreeBeforeCollege
core.whatIf.enableCurriculumRules
core.whatIf.majorRequired
core.whatIf.mustUsePrimaryRule
core.whatIf.programAdditionalRequired
core.whatIf.showAllConcsInPrimary
core.whatIf.showAllConcsInSecondary
core.whatIf.showAllMinors
core.whatIf.showCampus
core.whatIf.showCollege
core.whatIf.showConcentration
core.whatIf.showMajor
core.whatIf.showMinor
core.whatIf.showProgramAdditional
core.whatIf.showProgramPrimary
core.whatIf.showSchool
```

Future Classes

The user can also add courses that may be taken in the future to see how they might apply to the student's current program or a program they may be considering. A course Subject and course Number must be entered. Note that there is no validation on these fields so care should be taken to enter them correctly. Click "Add" to add the course to the What-If Analysis criteria. An unlimited number of courses can be added.

Process Audit

See the Worksheets > Academic > Process Audit section for information on this functionality.

Configuration

- Access to process an audit is granted if the user has the `SDAUDRUN` Shepherd key.

Historic Audits

"View historic what-if audits" allows a user to view historical What-If audits for a student. The maximum number of historical What-If audits to keep per student per School/Degree combination can be configured. If the user has access to view historic What-If audits, they can select the audit they wish to view from the dropdown and it will load using the currently selected report format.

Configuration

- Review the *What-If History Count* flag in CFG020 DAP14 to ensure it is configured correctly.

- Access to view historic audits is granted if the user has the `SDWIFHIS` Shepherd key.

Diagnostics

See the Worksheets > Academic > Diagnostics section for information on this functionality.

Student Data

See the Worksheets > Academic > Student Data section for information on this functionality.

Save Audit

Audits may be saved or “frozen” so they do not get deleted when the maximum CFG020 DAP14 What-If History Count has been reached. Frozen What-If audits do not count against the What-If history count value. Depending on the user’s access, when they click on “Save audit” a dialog will open that prompts for the Freeze type and a Description. When the user clicks “Save”, the What-If audit will be saved in the database with the specified freeze type and description, and the screen will be updated with the user who saved the audit, the date, the freeze type and description. For more information, please refer to the “Freezing Audits” special topic in the *Degree Works Technical Guide*.

Configuration

- Access to freeze What-If audits is granted if the user has the `WIFFREEZ` Shepherd key.
- Access to add a description is granted if the user has the `WIFDESCR` Shepherd key.

Delete Audit

What-If audits that have been saved or “frozen” can be deleted if the user has access to do so. Clicking on the “Delete audit” link opens a confirmation dialog. If the user clicks “Delete”, the What-If audit will be deleted from the database. If the user clicks “Cancel”, they will be returned to Worksheets without deleting the audit.

Configuration

- Access to delete audits is granted if the user has the `SDWIFDEL` Shepherd key.

Course Link

See the Worksheets > Academic > Course Link section for information on this functionality.

Athletic Eligibility

Athletic Eligibility audits allow you to process degree audits specifically for your student athletes. Rules for Athletic Eligibility are built using Scribe, and processing an athlete's athletic data against these requirement blocks will verify if a student is eligible to participate in the sport, in accordance with rules laid out by the NCAA, NJCAA, and other organizations. For more information, please refer to the "Athletic Eligibility Audit" special topic in the *Degree Works Technical Guide*.

Configuration

- Review RPT036 to ensure it is configured correctly for WEB55 and WEB56 as well as any custom formats you may have.
- Review the following Shepherd settings to ensure they are configured correctly:
`dash.audit.athletic.football.enabled`
- Access to Athletic Eligibility is granted if the user has the `SDATHAUD` Shepherd key.

Configuration

- Review RPT036 to ensure it is configured correctly for WEB30, WEB31, WEB33, WEB44, WEB50, WEB51, WEB55, WEB56 and SEP31 as well as any custom formats you may have.

Format

See the Worksheets > Academic > Format section for information on this functionality.

Configuration

- Access to the Athletic Eligibility Report is granted if the user has the `SDWEB55` Shepherd key.
- Access to the Athletic and Academic Report is granted if the user has the `SDWEB56` Shepherd key.

Degree Progress

See the Worksheets > Academic > Degree Progress section for information on this functionality.

Process Audit

See the Worksheets > Academic > Process Audit section for information on this functionality.

Configuration

- Access to process an athletic eligibility audit is granted if the user has the `SDATHRUN` Shepherd key.

Historic Audits

See the Worksheets > Academic > Historic Audits section for information on this functionality.

Configuration

- Access to view historic athletic eligibility audits is granted if the user has the `SDATHHIS` Shepherd key.

Diagnostics

See the Worksheets > Academic > Diagnostics section for information on this functionality.

Student Data

See the Worksheets > Academic > Student Data section for information on this functionality.

Save Audit

See the Worksheets > Academic > Save Audit section for information on this functionality.

Delete Audit

See the Worksheets > Academic > Delete Audit section for information on this functionality.

Configuration

- Access to delete audits is granted if the user has the `SDATHDEL` Shepherd key.

Course Link

See the Worksheets > Academic > Course Link section for information on this functionality.

Exceptions

The Exceptions function allows users to apply or remove exceptions to the requirements for degree completion for a specific student. To begin, select a student and then choose either “Add exception” or “Remove exception”.

Configuration

- Access to Exceptions is granted if the user has the `SDEXCEPT` Shepherd key.
- Access to add exceptions is granted if the user has the `SDEXPADD` Shepherd key.
- Access to delete exceptions is granted if the user has the `SDEXPDEL` Shepherd key.

Add Exception

There are two steps to applying an exception: entering the details for the selected exception type and selecting the requirement to which the exception should be applied. After selecting a student, this step process will load and if configured, a new audit will be run in the background. Select the exception type you wish to add from the dropdown. Enter a description of the exception being added – this is a required field and will display on the audit.

A Details field may be configured to display, allowing the user to provide additional details about the exception. This text doesn't display directly on the audit but can be viewed as a tool tip when hovering over the exception description on the audit but only if the user has the `EXPVWDTL` Shepherd key. Click the “Complete step” button to move to the next step.

Select the course rule or qualifier where you wish to add the exception by clicking the correct radio button. Exceptions are both student-specific and block-specific. In other words, an exception applies to a specific block used in a student degree audit. If an exception is processed in a major block for a student and that student then changes his/her major, the previous exception will no longer apply to the student's new major. However, if the `CFG020` Apply Exception To The Same Block Type flag is set to Y then the exception can be used on the new major but only if the label-tag or qualifier tags between the two blocks match. Unused exceptions will appear at the bottom of the audit report. Exceptions are also used with What-If audits. Exceptions will only appear in the exception screen and on audit reports configured to show Exceptions. By default, the Registrar's Audit report is configured to show exceptions.

Exceptions on rules are tied to the rule's label-tag, or label if no label-tag exists. When a block is modified using Scribe the exception is placed on the correct rule as long as the original rule label can be found in the new block. Qualifiers do not have labels so instead a Tag should be scribed for each header qualifier to prevent the exceptions from becoming unhooked. See the “Label Tags” section of the *Scribe Language User Guide* for more information on using tags.

After selecting where to apply the exception, click the “Add exception” button. A new audit will be automatically run, applying the exception. This audit can be configured to be saved to the database when it is run; otherwise, it will only be visible to the user while on this page.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly:
`dash.exceptions.details.option`
`dash.exceptions.runAuditOnEnter`
`dash.exceptions.saveNewAudit`
- Access to Also Allow exceptions is granted if the user has the `EXPALLOW` Shepherd key.
- Access to Apply Here exceptions is granted if the user has the `EXPAPPLY` Shepherd key.
- Access to Change the Limit / Remove Course exceptions is granted if the user has the `EXPCHANG` Shepherd key.
- Access to Force Complete exceptions is granted if the user has the `EXPFORCE` Shepherd key.
- Access to Substitute exceptions is granted if the user has the `EXPSUBST` Shepherd key.

Exception Types

Force Complete

Force Complete completes a course rule, subset rule, block qualifier or rule qualifier without applying additional classes. It is the most powerful exception type available.

The force complete exception can be used on any course rule and most qualifiers. This exception type is completely independent of all student data. It will simply complete a rule on a student degree audit regardless of any qualifiers that might apply. Forcing a qualifier complete tells the auditor to ignore the qualifier the next time and audit is performed for this student.

Substitute

Substitute is used to substitute one course for another. This is distinct from the Also Allow exception type in that one course is exchanged for another. If the rule contains only a single course then the substituted course is required for completion of the block. If a substitute exception is processed on a rule with more than one course option that can be used to complete the rule, then the substituted course is not required and is an option available to the student. Only qualifiers that list courses support this type of exception.

To create a substitute exception, enter the target course from the course rule in the “Change” fields. The target course must be found on the rule where the exception is to be placed in order to be saved to the database. Enter the substituted course in the “To” fields.

You can further define the exception using “With” qualifiers. The values listed in the dropdown are Ellucian baseline “With” qualifiers (DW Credits, DW Grade, etc) as well as user-defined “With” qualifiers from SCR044. When “With” qualifiers are included as a condition for this exception type, only those courses meeting the WITH qualifier criteria will be evaluated for the exception.

You can also create a global substitution exception by choosing the option at the top of the page.

These substitutions will be applied to all requirements found in all blocks.

Also Allow

Also Allow modifies a course rule by appending a course to the course rule. This exception can be used when you wish to expand the course options available on a specific rule. Courses applied using the Also Allow exception are still subject to header qualifiers in the blocks in which they are used and are still subject to the best-fit algorithm. For example, if an Also Allow exception is processed allowing ENGL 215 to be used to satisfy a course rule and there is a block header or rule qualifier preventing the use of ENGL 215 within that block, the exception will be added to the database, but ENGL 215 will not be allowed to be used within that block. You will need to first modify the header qualifier preventing this course from being used before processing the Also Allow exception.

When processing an Also Allow exception to a group rule, the exception can only be processed on the individual rule labels inside the group rule, not the GROUP RULE HEADER LABEL.

To create an also allow exception, enter the course discipline and number in the “Allow” fields.

You can further define the exception using “With” qualifiers. The values listed in the dropdown are Ellucian baseline “With” qualifiers (DW Credits, DW Grade, etc) as well as user-defined “With” qualifiers from SCR044. When “With” qualifiers are included as a condition for this exception type, only those courses meeting the WITH qualifier criteria will be evaluated for the exception.

The Also Allow exception does not require that the selected course be used on the modified rule.

The best-fit algorithm of the Audit Processor Engine will still function with this type of exception. Consequently, the allowed course may not be applied to rule bearing the exception if there is a better fit for this course elsewhere in the degree audit.

Block header exceptions do not support the Also Allow exception.

Apply Here

Apply Here allows the user to apply a course to a rule even if the course is not listed as an option. This exception is very useful in correcting audits in cases where the user wishes to dictate specifically where courses are to be used within the degree audit. The Apply Here exception will apply a course to a rule regardless of any scribing, rule or block header qualifiers. For example, suppose a block contains a 'MAXPASSFAIL 0 CLASSES' header qualifier. You can use the Apply Here exception to apply a PASSFAIL class to a rule in this block without having to modify the block header qualifier. The Apply Here exception will also override an EXCEPT list on a rule. Courses applied to rules using this exception type will not be moved around within the audit. When processing an Apply Here exception on a group rule, the exception can only be processed on the individual rule labels inside the group rule. Apply Here exceptions cannot be processed on a Group Rule Header.

You can also use an Apply Here exception to place a failed or incomplete class on a requirement. This could be a problem wherein you place the exception when the class is still in-progress and it applies to the requirement just fine. However, if the student subsequently fails the class the exception is still enforced. If you do not want this to occur then you should add a WITH parameter on the exception, such as DW Grade Number, to ensure only passed or in-progress classes apply.

To create an apply here exception, enter the course you want applied to a rule. The course must be a course already taken by the student and found on the degree audit or one the student is planning to take. The Apply Here exception cannot be used for courses that have not yet been taken.

You can further define the exception using “With” qualifiers. The values listed in the dropdown are Ellucian baseline “With” qualifiers (DW Credits, DW Grade, etc) as well as user-defined “With” qualifiers from SCR044. When “With” qualifiers are included as a condition for this exception type, only those courses meeting the WITH qualifier criteria will be evaluated for the exception.

When placing an Apply Here exception any existing WITH data is removed so that a course that normally did not meet the WITH qualifiers will be able to fit on the rule. If the Apply Here exception contains a new WITH qualifier that will be applied – replacing the previous WITH data.

Block header exceptions do not support the Apply Here exception.

Remove Course

Remove Course allows the user to remove a course from a course rule or qualifier or to change the limit on a course rule or qualifier. This exception type is very useful in modifying audit reports when students successfully petition to have a specific course or part of a specific requirement waived. This exception type is also useful in correcting the advice given to students when a specific course is disallowed on a specific rule by a student. For example, if a student receives advice that they can take CSC 113 to fulfill the Technology Requirement but the department chair has disallowed this course on this rule for a specific student, you can correct the advice given to the student by using the Remove Course exception.

The Remove Course exception type can also be used to change the limit on a rule without removing a course from the course list.

To create a remove course exception, enter the course discipline and number you wish to remove. Enter a brief description for the exception. If the exception also involves changing the limits on the rule or qualifier for the student, enter the new limit in the “Change” fields.

In situations where a course is applied to a rule or qualifier as a result of a wildcard statement, using the Remove Course exception will remove ALL courses that have been applied as a result of the wildcard statement. For example, if HIST 330 and HIST 427 are both applied to the scribe rule 3 Classes in HIST @, removing HIST @ using the Remove Course exception will remove both HIST 330 and HIST 427.

However, if you remove just HIST 330 the rule gets changed to 3 Classes in HIST @ Except HIST 330 – making sure that HIST 330 does not get applied to this requirement.

Only qualifiers that list courses or specify a number of classes or credits support the Remove Course exception.

Qualifier Exceptions

The Force Complete, Substitute, Remove Course and Change Limit exceptions can be used on block header qualifiers and rule qualifiers. The header qualifiers appear above the rules in the block and have a radio button next to them.

Not all qualifiers support the same types of exceptions. Below are the exceptions and the qualifiers on which they can be placed:

Exception Type	Qualifiers Allowed
Force Complete	All except NonExclusive (aka ShareWith), SameDisc, and Optional
Substitute	MaxClasses, MinClasses, MaxCredits, MinCredits, MaxTerm, MinTerm, SpMaxCredit, SpMaxTerm
Remove Course	MaxClasses, MinClasses, MaxCredits, MinCredits, MaxTerm, MinTerm, SpMaxCredit, SpMaxTerm
Change Limit	All except SameDisc, Option, MinGPA, MinGrade

Remove Exception

To remove an exception, click on the “Remove” icon that appears to the left of the applied exception. The exception will be removed and a new audit will be automatically generated.

Exception Management

Exception Management allows users to manage petitions and apply exceptions, fix petition statuses and generate exceptions reports. The Exceptions Management function is located under the Admin menu.

Configuration

- Access to Exception Management is granted if the user has the `SDEXPMGT` Shepherd key.

Manage Petitions

Manage Petitions allows the user to view, approve, reject and apply petitions requests for all students.

Configuration

- Access to Awaiting Approval is granted if the user has the `SDEMPEWA` Shepherd key.
- Access to Approved Petitions is granted if the user has the `SDEMPEAV` Shepherd key.
- Access to Rejected Petitions is granted if the user has the `SDEMPERJ` Shepherd key.
- Access to Applied Petitions is granted if the user has the `SDEMPEAL` Shepherd key.

Awaiting Approval

Awaiting Approval is used to approve, reject or add additional documentation to pending petitions. Any petition with a pending status will appear on the “Petitions Awaiting Approval” grid.

To approve an individual petition, expand the petition by clicking on the arrow on the right and click the “Approve” button. The status of the selected petition will change to approved and the exception will be removed from the list of petitions waiting approval. To approve multiple petitions, select the petitions that should be approved by checking the checkbox on the left and then click the “Approve” button at the top of the grid.

To reject an individual petition, expand the petition by clicking on the arrow on the right and click the “Reject” button. The status of the selected petition will change to rejected and the exception will be removed from the list of petitions waiting approval. To reject multiple petitions, select the petitions that should be rejected by checking the checkbox on the left and then click the “Reject” button at the top of the grid.

It is also possible to add additional documentation to a pending petition while approving or rejecting an individual petition. With the petition row expanded, enter your text in “My comments” before clicking “Approve” or “Reject”.

It is important to remember that an approved petition is not the same as an applied exception. Once a petition has been approved, it is still necessary to actually apply the approved petition as an exception on the student's record.

Approved Petitions

Approved Petitions allows a user to process exceptions for petitions that have been approved. Any petition with an approved status will appear on the "Apply Approved Petitions" grid.

To process an exception for an approved petition, expand the petition by clicking on the arrow on the right. The exceptions step process will load, and an exception can either be added or removed. For information on how to apply an exception, please refer to the *Exceptions* section of this document. After adding the exception, click the "Update petition and close" button. The status of the selected petition will change to applied and the exception will be removed from the list of approved petitions.

Rejected Petitions

Rejected Petitions allows a user to view all petitions that have been rejected. Select the petition date range to filter the list of rejected petitions that appear on the "Rejected Petitions" grid.

Rejected petitions can be maintained in the database or deleted as necessary. To delete rejected petitions, select one or more petitions by checking the checkbox on the left and then click the "Delete" button at the top of the grid. The exception will be removed from the list of rejected petitions.

Configuration

- Access to Delete rejected petitions is granted if the user has the `SDEMPERD` Shepherd key.

Applied Petitions

Applied Petitions allows a user to view all petitions that have been applied as exceptions. Select the petition date range to filter the list of applied petitions that appear on the "Petitions Applied as Exceptions" grid.

Applied petitions can be maintained in the database or deleted as necessary. To delete applied petitions, select one or more petitions by checking the checkbox on the left and then click the "Delete" button at the top of the grid. The exception will be removed from the list of applied petitions.

Configuration

- Access to delete rejected petitions is granted if the user has the `SDEMPHAD` Shepherd key.

Fix Petition Status

Fix Petition Status allows a user to change the status of a petition. Petitions can be filtered by date range and/or student ID.

All petitions meeting the search criteria will be displayed in the “Fix Petition Status” grid. To change the status of a petition, select the correct status from the dropdown in the Status column. Click the “Save Changes” button to save your changes to the database.

If a petition has a status of Approved and additional comments were entered when the petition was approved, changing the petition status back to waiting will not affect the additional comments that were originally entered. These comments will display with the petition until the petition is either applied as an exception or deleted from the database.

If a petition has a status of Applied, changing the status of the petition will have no effect on the exception that has already been applied. One should therefore be careful when changing the status of petitions that have already been applied as exceptions as it is possible to apply more than one exception for a given petition.

Configuration

- Access to fix petition status is granted if the user has the `SDEMPEFX` Shepherd key.

Exceptions Report

Exceptions Report allows users to query the database and create reports on the number and type of exceptions processed on your audits. You can search for exceptions created by an individual, exceptions applied for a particular student or exceptions applied within a specific block. You can also use this function to identify unhooked exceptions as well.

You can filter the exceptions report by any combination of exception date, type, status, create ID, student ID and/or requirement ID. Click the “Run exceptions report” button to process your query. Your search will be displayed in the grid below.

The search results can be sorted by any of the columns displayed. Simply click the header label for the column you wish to use as your sort and the list will sort based upon that criterion. Sorts can be done in ascending or descending order. To change from descending to ascending, click the column header again and the indicator will change from a down arrow to an up arrow.

By default, the search results are returned to “Show exception details”. Expanding the row will display additional information about the exception. Select “Show requirement block counts” to show which scribe blocks have exceptions processed against them. This is a useful tool for determining if a scribe block is requiring an excessive number of exceptions.

Configuration

- Access to search for exceptions is granted if the user has the `SDEMEXSR` Shepherd key.

Exception Status Codes

The OK status code indicates that the exception applied correctly. The UN status code means that the exception became unhooked from its intended requirement during the reparse of the block; this usually is because the rule on which the exception was placed changed so much that new rule could not be recognized. To prevent this from happening you should ensure your rules have label tags and your qualifiers have qualifier tags. The other codes listed below indicate that the exception was unenforced meaning that when the audit was run the exception could not be applied as intended.

If you see no value in the Label Tag field for any exception you should strongly consider adding a label tag to that requirement. Please review the “Label Tags” section in the *Scribe Language User Guide* for help preventing unhooked exceptions.

Status code	Meaning
OK	Exception applied successfully
UN	Unhooked – typically caused by a Scribe block change (be sure to use label tags!)
BK	Block was not in audit – typically caused by a major or catalog year change
CR	Rule is not a course rule – node-ids are incorrect
CO	Course to be substituted was not found on the rule
IF	IF-statement was resolved – this rule was not used
ND	Node-id not found in block
PA	Rule is a plus-list and Also Allow is not allowed on a plus-list
PN	Rule is a plus-list and the course is not on the rule; the course can't be added to the end of the rule
R1	Not enforced but we don't know why exactly
R2	Not enforced and an unknown error code was found

Plans

Using Plans, a student or advisor can create an academic plan either from scratch or using a pre-defined template for the course of study. Using their plan, a student can generate an audit to evaluate their intended degree progress or have a custom Scribe block for their degree program created.

Configuration

- Access to Plans is granted if the user has the `SEPPLAN` Shepherd key.

Plan List

The Plan List displays all of a student's plans, regardless of degree, school, or status. Plans in the view can be sorted by any column by clicking on the column header. To open a plan for viewing or editing, click on the plan Description.

From the Plan List, if a user has access they can also create a new plan or delete an existing plan.

The user can return to the Plan List from a plan by clicking on the "Plan list" button.

Configuration

- Access to create a plan from scratch is granted if the user has the `SEPPADD` Shepherd key.
- Access to create a plan from a template is granted if the user has the `SEPPTEMP` Shepherd key.
- Access to delete any plan is granted if the user has the `SEPPDEL` Shepherd key.
- Access to delete any plan that is not active, locked, or approved is granted if the user has the `SEPPDELL` Shepherd key.

Creating and Editing Plans

A new plan can be created while on the Plan List or in the plan view by clicking "New plan". If the user has the keys to create a plan from scratch and create a plan from a template, they will be prompted to choose the method they want to use when creating a new plan.

A plan Description is required. This field is 80 bytes long and does not need to be unique; that is, the student can have more than one plan with the same description.

Plan type can be configured to display. This value can be used to identify the type of plan this is and is typically be used for state funding reporting. If the user has access to modify the plan type, they can select a value from a dropdown of values from SEP011. The Plan Type can also be configured to be a required field. When the plan type is modified, the Scope Date will be updated with the date modified.

Active can be modified by the user if either Locking or no approval method is configured. If the approval method is Workflow, the active flag will be updated through the workflow process. A student can have only one active plan per degree/school, or multiple active plans per degree/school, depending upon configuration. If configured to only allow one active plan, the student may still have more than one active plan, if they have several degree/school goal combinations.

Locked will display only if the approval method is Locking. Status will display only if the approval method is Workflow.

The Degree and School associated with the plan will default from the selected values in the student header, and if the student has goal data for more than one degree/school combination, the user will be able to assign a different degree for the plan. If a different degree is selected, the corresponding school will automatically load. Only the degrees for which the student has active goal data will be available in the dropdown.

Tracking status will initially load as “Not evaluated” if tracking is enabled. This will be updated automatically when the plan is edited.

Configuration

- Review the *Show in SEP Picklist* flag and *Term Type* in SEP011 to ensure they are configured correctly.
- Review the following Shepherd settings to ensure they are configured correctly:
`studentPlanner.planner.allowMultipleActivePlans`
`studentPlanner.planner.planApprovalMethod`
`studentPlanner.planner.showScope`
`studentPlanner.planner.tracking.enable`
- Access to modify a plan is granted if the user has the `SEPPMOD` Shepherd key.
- Access to modify the Locked flag on a plan is granted if the user has the `SEPPLOCK` Shepherd key.
- Access to modify the Plan Scope on a plan is granted if the user has the `SEPSCOPE` Shepherd key.

Creating a Plan from Scratch

When the user chooses to create a plan from scratch, they will be prompted to select a starting term for the plan. Additional terms can be added when editing plan. After providing a description and other details, a blank plan with the starting term but no requirements will load.

Configuration

- Review the *Show in SEP Picklist* flag in STU016 to ensure it is configured correctly.

Creating a Plan from a Template

The user first selects the template they want to base the plan on, from a list of all active templates, and then provides a starting term for the plan. The term values in the dropdown will be those that have the STU016 *Show in SEP Picklist* flag = Y and STU016 *Term Type* equal to the SEP002 *Term Type* for first term in the Template Term Scheme. The generic template term on the template will be replaced with an actual term on the plan, based on the template's Term Scheme and the Starting Term.

The template Description will be loaded as the default plan Description, but it can be modified by the user.

Configuration

- Review the *Show in SEP Picklist* flag and *Term Type* in STU016 and *Term Type* in SEP002 to ensure they are configured correctly.

Creating and Editing Terms

An unlimited number of terms can be added to a plan by clicking on the “Add term” button. A dropdown of available terms will be displayed. The values in this dropdown are all terms from STU016 with *Show in SEP Picklist* flag = Y which have not already been added to the plan. The student's active term and past terms will be filtered out of this list if changes are not allowed to the current or past terms.

The tracking status and total number of planned course credits will be displayed on the term header. These values are updated when a plan is edited.

Up to three terms will display in the desktop view. Other terms on the plan can be scrolled into view by clicking on the left or right arrows on the left side of the plan.

To delete a term, select “Delete this term” from the action menu in the right corner of a term.

All the requirements on a term can be moved to another term by selecting “Reassign this term” from the action menu in the right corner of a term and selecting the new term from the dropdown. After clicking “Reassign”, the term and its requirements will be moved to the correct chronological position on the plan.

Configuration

- Review the *Show in SEP Picklist* flag in STU016 to ensure it is configured correctly.
- Access to add and edit terms on a plan is granted if the user has the SEPPTADD Shepherd key.
- Access to delete terms from a plan is granted if the user has the SEPPTDEL Shepherd key.
- Access to only modify existing terms on a plan is granted if the user has the SEPPTMOD Shepherd key.

Creating and Editing Requirements

An unlimited number of requirements can be added to a planner term by either clicking on the “+” on the requirement card in the Requirements menu on the sidebar or by dragging and dropping the requirement card to the desired term. The options in this menu come from the requirement types in SEP004 with *Show in SEP Picklist* flag = Y. The description for these requirement types can be modified by the site, but the keys should not be changed or deleted. There are 6 requirement types: Course, Choice, GPA, Non-Course, Test and Placeholder.

When adding a requirement, a dialog will open where the details of the requirement can be added. If the requirement card was dragged and dropped into a term, that term will load automatically. If added from the sidebar, the user can select the term to which the requirement should be added.

If the Critical indicator is configured to display, it will appear on all requirement types except Placeholder. It is used to identify requirements that are critical for tracking. This indicator can only be changed by a user who has been given access to do so.

To edit a requirement, select “Edit this requirement” from the action menu in the right corner of a requirement.

To delete a requirement, select “Delete this requirement” from the action menu in the right corner of a requirement.

To move a requirement to another term, select “Reassign this requirement” from the action menu in the right corner of a requirement or drag and drop it to the desired term.

Control over whether requirements on past or current terms can be modified or deleted can be configured. These settings apply to all users regardless of the access granted by Shepherd keys. Note that when Tracking is enabled, these settings only apply to On-Track requirements. Off-Track and not tracked requirements can always be modified or deleted.

Configuration

- Review the *Show in SEP Picklist* flag in SEP004 to ensure it is configured correctly.
- Review the following Shepherd settings to ensure they are configured correctly:
`studentPlanner.courseReq.showCritical`
`studentPlanner.planner.allowChangesToCurrentTerms`
`studentPlanner.planner.allowChangesToPastTerms`
`studentPlanner.planner.allowDeletesInCurrentTerms`
`studentPlanner.planner.allowDeletesInPastTerms`
- Access to add and edit requirements on a plan is granted if the user has the `SEPPRQAD` Shepherd key.
- Access to delete requirements on a plan is granted if the user has the `SEPPRQDL` Shepherd key.
- Access to only modify existing requirements on a plan is granted if the user has the `SEPPRQED` Shepherd key.

- Access to set the Critical indicator on a requirement is granted if the user has the `SEPCRIT` Shepherd key.

Course

A single class can be planned for with the Course requirement. In addition to adding individual Course requirements from the Requirements menu on the sidebar, multiple Course requirements can be added to a term using the requirement drawer. This is accessed by clicking in the “+” button in a term.

When adding a Course requirement from the Requirements menu on the sidebar, enter the course number in the “Course requirement” field. The user’s input will be validated before the requirement can be saved.

When adding Course requirements from the requirement drawer, the user can filter the list of all courses by subject, or by course title. Additionally, a list of all requirements that are still needed can be viewed. One or more requirements can be selected while in the requirement drawer, and then added at once by clicking “Add to plan”.

Note that the course requirement search results in the requirement drawer can be filtered by the school associated with the plan and/or the course end date. When filtering by school is enabled, only courses with a `DW-SCHOOL rad_crs_attr_dtl` equal to the school on the plan will appear in the list. When filtering out courses no longer needed is enabled, if the student’s active term is greater than `rad_course_mst.rad_cat_yr_stop`, the course will not appear in the list. Sites with custom student data extracts can add functionality to bridge the required data into the `rad_crs_attr_dtl` and `rad_course_mst`. Depending on your student information system, see the *Banner Data Mapping for BIF Technical Guide* or the *Colleague Data Mapping for BIF Technical Guide* for more information.

The Credits will automatically load from the course record in the database. If the course is a variable credit course, then the value in `rad_credits` (typically this is equal to `rad_credits_high`, per the extract) for that course is displayed, and the user can change the number of credits to a value within the range for that course. The format and length of the credits can be configured by the site.

If the Honors flag is configured to display, the user can check this field to indicate if an honors section is needed. This field is informational only.

If the Minimum Grade is configured to display, the user can select the minimum grade needed for the course. This can be used as criteria in tracking. The values in this dropdown come from the grades in `STU385` with *Show in SEP Picklist* flag = Y.

If the Campus is configured to display, the user can select the campus where the course should be taken. This field is informational only. The values in this dropdown come from the location codes in `STU576` with *Show in SEP Picklist* flag = Y.

If the Delivery is configured to display, the user can select a content delivery method for the course. This field is informational only. The values in this dropdown come from the delivery codes in `SEP003` with *Show in SEP Picklist* flag = Y.

Configuration

- Review STU050 and SEP009, as well as the *Show in SEP Picklist* flag in SEP003, STU576 and STU385 to ensure they are configured correctly.
- Review the following Shepherd settings to ensure they are configured correctly for your SEP environment:

```
core.credit.format
studentPlanner.choiceReq.enableCourseAttribute
studentPlanner.choiceReq.enableScribePointer
studentPlanner.courseReq.showHonors
studentPlanner.courseReq.showCampus
studentPlanner.courseReq.showDelivery
studentPlanner.courseReq.showGrade
```

Choice

When there are several course options that can fulfill a planned requirement, a Choice requirement can be used. An unlimited number of options can be defined in the Choice requirement, and an option can be any combination of actual courses, wildcards or ranges, for example, STAT 101 OR MATH @.

Each option can have up to two items. An item can be an actual course such as MATH 101, a wildcard such as @ @, MATH @, or MATH 3@, or a range such as MATH 100:299. If configured, a user can specify a course attribute on any item. The values in this dropdown come from the course attribute codes in STU050. STU050 can be loaded through the extract, or manually by the user.

To add an option, click “Add” link and enter the course number in the “Course requirement” field. The user’s input will be validated before the requirement can be saved.

To add an additional item to an option, click “Add a paired course or lab”.

One of the options may be selected by clicking on the radio button next to the preferred option. Clicking the “Clear selection” button will remove the selection from the option.

The Minimum Credits for this requirement can be manually entered but are not required. The format and length of the minimum credits can be configured by the site.

If the Pointer is configured to display, the user can define a value that links this requirement to a rule in a requirement block. The values in this dropdown come from the pointer codes in SEP009 with *Show in SEP Picklist* flag = Y. See the *Scribe Pointers* section in this document for additional information.

If the Honors flag is configured to display, the user can check this field to indicate if an honors section is needed. This field is informational only.

If the Minimum Grade is configured to display, the user can select the minimum grade needed for the course. This can be used as criteria in tracking. The values in this dropdown come from the grades in STU385 with *Show in SEP Picklist* flag = Y.

If the Campus is configured to display, the user can select the campus where the course should be taken. This field is informational only. The values in this dropdown come from the location codes in STU576 with *Show in SEP Picklist* flag = Y.

If the Delivery is configured to display, the user can select a content delivery method for the course. This field is informational only. The values in this dropdown come from the delivery codes in SEP003 with *Show in SEP Picklist* flag = Y.

Configuration

- Review STU050 and SEP009, as well as the *Show in SEP Picklist* flag in SEP003, STU576 and STU385 to ensure they are configured correctly.
- Review the following Shepherd settings to ensure they are configured correctly for your SEP environment:

```
core.credit.format
studentPlanner.choiceReq.enableCourseAttribute
studentPlanner.choiceReq.enableScribePointer
studentPlanner.courseReq.showHonors
studentPlanner.courseReq.showCampus
studentPlanner.courseReq.showDelivery
studentPlanner.courseReq.showGrade
```

- Access to select a choice requirement option is granted if the user has the SEPPSEL Shepherd key.
- Access to change a choice requirement pointer is granted if the user has the SEPPPTR Shepherd key.

GPA

Four types of minimum GPAs for a term can be planned for with the GPA requirement – Overall Degree Works GPA, Overall Student System GPA, Major GPA, and a Class List GPA. These can be used as criteria in tracking. The GPA types that are in the dropdown come from the codes in SEP008 with *Show in SEP Picklist* flag = Y.

The format and length of the minimum GPA can be configured by the site.

When adding a Major GPA, a major must be specified. The majors that are in the dropdown come from the major codes in STU023 with *Show in SEP Picklist* flag = Y.

When adding a Class List GPA, a group of classes in which the student needs to have earned a minimum GPA can be defined. Each class should be separated by a plus sign, and ranges and wildcards can be included in the list. Classes are validated, but ranges and wildcards are not.

Configuration

- Review the *Show in SEP Picklist* flags in SEP008 and STU023 to ensure they are configured correctly.
- Review the following Shepherd settings to ensure they are configured correctly:

```
core.gpa.format
core.gpa.maxLength
```

Non-Course

Non-Course items that need to be completed as part of a degree can be planned for with the Non-Course Requirement. These can also be used as criteria in tracking. The Non-Course Requirement types that are in the dropdown come from SCR003.

The Status can be left blank to indicate that the student just needs to have completed the non-course item. It can also be the alphanumeric score or status needed to consider the requirement complete.

Configuration

- Review SCR003 to ensure it is configured correctly.

Test

Tests that need to be completed as part of a degree can be planned for with the Test Requirement. These can also be used as criteria in tracking. The Test Codes that are in the dropdown come from the codes in SEP006 with *Show in SEP Picklist* flag = Y. A Minimum Score is required, and the length, type, and valid score range of the Minimum Score can be defined in SEP006. Note that alpha scores require additional setup in SEP007 to define the sort order for these values.

If you will be using the Create Block functionality, your SEP006 Test Codes must also be in SCR002 Custom Data. If it does not, the creation of the block will fail with a parsing error.

Configuration

- Review the *Show in SEP Picklist* flag, *Score Length*, *Score Type*, *Minimum Valid Score*, and *Maximum Valid Score* in SEP006 to ensure that they are configured correctly.
- Review the *Sort Order* in SEP007 to ensure that it is configured correctly.

Placeholder

User-defined requirements that need to be completed as part of a degree can be planned for with the Placeholder Requirement. These requirements are informational only and are not included in tracking or do not apply to planner audit rules. The Placeholder Requirement types that are in the dropdown come from the codes in SEP005 with *Show in SEP Picklist* flag = Y.

The Value is a free-form text field and is required.

Configuration

- Review the *Show in SEP Picklist* flag in SEP005 to ensure it is configured correctly.

Creating and Editing Notes

An unlimited number of notes can be added to requirements, terms or to the overall plan. All notes are viewable to all users, and users can be given access to add, modify, and/or delete notes they have created on each of these plan components. Special access can be given to a user to modify or delete notes created by another user.

Click on the note icon to open the note list window, where you can choose to view, add, edit, or delete notes, depending on your level of access.

To add a note, click on the “Add a new note” button. Notes can have an unlimited amount of text. To make a note internal, select the “Not available to student” checkbox. Click “Save note” to keep your note, or “Cancel” to go back to the note list window without keeping your note.

To view a note, select “View note” from the action menu in the right corner of a note.

To edit a note, select “Edit note” from the action menu in the right corner of a note.

To delete a note, select “Delete note” from the action menu in the right corner of a note.

If a note has been marked as Internal, the note will only be viewable by users who have been granted the `SEPINOTE` key. Additionally, the Internal checkbox will only display in the note list window for users with access to view internal notes.

Notes that came from the template used to create a plan have a special indicator that can be configured to display in the note edit window. These notes are viewable by all users but can only be edited or deleted by users that have been given access to do so.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly for your SEP environment:

```
studentPlanner.planner.notes.showComesFromTemplateIndicator
studentPlanner.notes.enableSpellCheck
```

- Access to add and edit plan notes is granted if the user has the `SEPPNTAD` Shepherd key.
- Access to delete plan notes is granted if the user has the `SEPPNTDL` Shepherd key.
- Access to edit only plan notes is granted if the user has the `SEPPNTED` Shepherd key.
- Access to add and edit term notes is granted if the user has the `SEPPNMAD` Shepherd key.
- Access to delete term notes is granted if the user has the `SEPPNMDL` Shepherd key.
- Access to edit only term notes is granted if the user has the `SEPPNMED` Shepherd key.
- Access to add and edit requirement notes is granted if the user has the `SEPPNRAD` Shepherd key.
- Access to delete requirement notes is granted if the user has the `SEPPNRDL` Shepherd key.
- Access to edit only requirement notes is granted if the user has the `SEPPNRED` Shepherd key.
- Access to edit notes copied from a template is granted if the user has the `SEPPNTET` Shepherd key.

- Access to delete notes copied from a template is granted if the user has the `SEPPNTDT` Shepherd key.
- Access to edit notes created by other users is granted if the user has the `SEPPNTOW` Shepherd key.
- Access to delete notes created by other users is granted if the user has the `SEPPNTGD` Shepherd key.
- Access to internal notes is granted if the user has the `SEPINOTE` Shepherd key.

Sidebar

A sidebar panel of courses the student still needs to plan for, all courses available and requirement types can be found to the left of the plan. Courses from either the Still Needed list or the Courses list as well as Requirement types can be dragged and dropped into a term on the plan.

Courses

The Courses list will show all courses in the `rad_course_mst` ordered by discipline. Clicking on the arrow next to the discipline will expand the list to show all available courses in that discipline.

The Courses list can be filtered by the school associated with the plan and/or the course end date. When filtering by school is enabled, only courses with a DW-SCHOOL `rad_crs_attr_dtl` equal to the school on the plan will appear in the list. When filtering out courses no longer needed is enabled, if the student's active term is greater than `rad_course_mst.rad_cat_yr_stop`, the course will not appear in the list. These fields are populated by the Course extract, so before enabling either filtering option, you must run the Course extract or else the Course List will not populate. Depending on your student information system, see the *Banner Data Mapping for BIF Technical Guide* or the *Colleague Data Mapping for BIF Technical Guide* for more information.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly:

```
studentPlanner.planner.filterCoursesByPlanSchool
```

```
studentPlanner.planner.filterCoursesNoLongerOffered
```

Still Needed List

The Still Needed list is a mini planner audit that shows only the requirements that are still needed to complete the plan's associated program. When all requirements on a rule have been planned for, the rule will no longer appear in the Still Needed list.

For requirements with wildcards, the maximum number of matching courses that are displayed in the Still Needed List can be configured by the site.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly:

```
studentPlanner.planner.sidebar.stillNeeded.courses.maxResults
```

Requirements

The Requirements menu is a list of all the requirement types that can be used on a plan. The options in this menu come from the requirement types in SEP004 with *Show in SEP Picklist* flag = Y. The description for these requirement types can be modified by the site, but the keys should not be changed or deleted. There are 6 requirement types: Course, Choice, GPA, Non-Course, Test and Placeholder.

Configuration

- Review the *Show in SEP Picklist* flag in SEP004 to ensure it is configured.

Course Link

If Course Link is enabled, users will be able to click to get additional information such as the course description, meeting times, instructors, and prerequisites on courses in the Course and Choice requirements, terms, Planner Audit, Planner What-If Audit, Still Needed list, and on the Courses list.

To display Course Link information in Course and Choice requirements or for all Course and Choice requirements on a term, select “More information” from the action menu in the right corner of the item. In the Planner Audit, Planner What-If Audit, Still Needed List, and Course List, click on the actual course to display Course Link information.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly:

```
studentPlanner.courseLink.enable
```

Course Information

Instead of retrieving course information from Banner you may retrieve information from a page you have configured on campus or from a third-party tool such as CourseLeaf. However, this configuration is not specific to the planner – it is used in all worksheets through the Responsive Dashboard. Please see the *Course Information* section under *Integration with Portals* for more information.

When the Course Information setting is enabled the “More Information” menu will only appear for Course requirements; it will not show for Choice requirements and will not show for the term. In the Planner Audit, Planner What-If Audit, Still Needed List, and Course List, click on the actual course to display this alternate form of course information.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly. Both of these flags must be enabled to get this special course information URL to be used.

```
studentPlanner.courseLink.enable
dash.courseInformation.enabled
dash.courseInformation.url
```

Audit

Generating an audit from a student's plan will show their progress towards their degree if all of their planned requirements are completed successfully. Class requirements will be applied to rules, as will Choice requirements in which a selection has been made or which have a Pointer associated with a rule. Placeholder requirements will display in the Planner Placeholders section of the planner audit. Test and GPA requirements will not be applied to the audit. Non-Course requirements will also be displayed in the Planner Audit, in rules or in fall-through.

For historical terms, only actual classes will appear on the planner audit; planned requirements in these terms will not be applied. Only planned requirements for future terms will apply on the planner audit. However, when using planned courses on the active term has been enabled, the student's active term will include both classes for which they have registered as well as any planned courses in that term for which the student hasn't registered. In the special situation where a student has not registered for classes but does have planned requirements in their active term, the planned requirements will be applied to the planner audit regardless of the use planned courses on active term setting.

If using pre-registered classes is enabled, the planner audit will show these classes for future terms rather than the planned course.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly for your SEP environment:

```
studentPlanner.planner.audit.usePlannedCoursesOnActiveTerm
studentPlanner.planner.audit.usePreregistered
```

- Access to generate a planner audit is granted if the user has the `SEPPAUD` Shepherd key.

Diagnostics

See the Worksheets > Academic > Diagnostics section for information on this functionality.

Student Data

See the Worksheets > Academic > Student Data section for information on this functionality.

What-If

Planner What-If audits allow the student to forecast how their planned and completed coursework would apply to a different goal. The Planner What-If audit can be configured to use curriculum rules to filter the goal data drop down lists. Additionally, which goal data can be used for filtering, required goal data and other behavior can be configured. If the Planner What-If audit is not configured to use curriculum rules for filtering, the goal data drop down lists will contain all values from the associated UCX table. Planner What-If audits are not saved to the database.

Click the “What-If” link to open the Planner What-If Analysis interface.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly:
`core.whatIf.enableCurriculumRules`
- Access to the Planner What-If functionality is granted if the user has the `SEPPWIF` key.

Curriculum Rule Filtering

See the Worksheets > What-If > Curriculum Rule Filtering section for information on this functionality.

Standard UCX Filtering

See the Worksheets > What-If > Standard UCX Filtering section for information on this functionality.

Create Block

Create Block can generate a requirement block from a student’s plan. This is useful when the degree requirements are decided upon on a student-by-student basis as is the case in many graduate programs and eliminates the need to build individualized blocks in Scribe.

After creating a plan for a student, click on Create Block to save the plan and generate an OTHER requirement block with one rule for each requirement listed in the plan. The value and title for this OTHER block can be configured by the site. If these settings are left blank, the default values of ‘PLAN’ and ‘Planner Block’ are used. The catalog year on the plan is used as the start and stop catalog years on the block, and the student’s ID is added as a secondary tag to block. A degree block that calls this OTHER block must exist in order to generate an audit for this program.

Example:

For an MBA degree, you would create a `DEGREE=MBA` block in Scribe and then call in the `OTHER=PLAN` block:

```
1 Block (OTHER=PLAN)
  Label "Educational Plan requirements";
```

When a new audit is run the correct OTHER=PLAN block with the specific student ID as the secondary tag will be pulled into the audit.

If the plan is modified, click Create Block again to regenerate the block. This will overwrite the block that had been previously saved.

Once a block is created, Scribe may be used to modify it. If Create Block is used to regenerate the block after Scribe changes have been made, the user will be asked to confirm that they want to overwrite the changes made in Scribe with the Create Block modifications.

Blocks created using this option have a block ID that begins with "RB" instead of "RA". When using dapblockunload/dapblockload, these "RB" blocks are ignored. So if you are moving requirement blocks from one environment to another, these "RB" blocks are not considered.

The requirements on a plan can be configured to be grouped together by term in a subset in the block, or they can be single rules in the block. If subsets are used, the text preceding the term literal in the block can be defined by the site.

Course requirements can be configured to use either classes or credits in the rules, and whether to include the number of credits as part of the course title. The text preceding a Placeholder requirement can be defined by the site.

Test requirements will be created as If statements using RuleComplete/RuleIncomplete to evaluate if the minimum score was achieved.

Non-Course requirements will be created as a NonCourse rule in the block. If the requirement value is blank, then the rule will be written as follows:

```
1 Noncourse (THESIS)
  Label 14 "THESIS required";
```

If the requirement value is alpha then the rule will be written as follows:

```
1 Noncourse (THESIS = XYZ)
  Label 14 "THESIS with a value of XYZ required";
```

If the requirement value is numeric then the rule will be written as follows:

```
1 Noncourse (THESIS >= 123)
  Label 14 "THESIS with a score of at least 123 required";
```

Overall and Major GPA requirements will be created as Remarks in the block, while Class GPA requirements will be written as header qualifiers.

Choice requirements will be created as Group rules in the block. If an attribute is specified on the choice option, these will be written to the block as well.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly:
 - `studentPlanner.planner.createBlock.placeholderLabelPrefix`
 - `studentPlanner.planner.createBlock.planBlockTitle`
 - `studentPlanner.planner.createBlock.planBlockValue`
 - `studentPlanner.planner.createBlock.showCreditsInRuleLabel`
 - `studentPlanner.planner.createBlock.subsetLabelPrefix`
 - `studentPlanner.planner.createBlock.subsetsForTerms`
 - `studentPlanner.planner.createBlock.useClassesInRules`
- Access to Create Block is granted if the user has the `SEPPBLCK` Shepherd key.

Plan Approval

Two methods of plan approval are supported in SEP – Locking and Workflow.

When Locking is enabled, a locked plan cannot be modified by a user who does not have access to change the Locked flag. The Save button will be disabled and the user will only be able to save their changes as a new plan. The Locked flag is manually maintained and can be changed only by users that have been given access to change it. Access to modify the Locked flag on a plan is granted if the user has the `SEPPLOCK` key.

When Workflow is enabled, Banner Workflow 8 can be used to manage plan approvals. When a user saves a new plan, they will be given an option to submit it for approval or just save it. Once a plan has been submitted for approval, it cannot be edited while it is pending approval. If the user needs to make changes, they will have to save their modified plan as a new plan. Approved plans also cannot be modified unless allowing changes to approved plans has been enabled. Rejected plans can be modified and resubmitted for approval.

If the user has been given access, they will have an option to save and approve the plan from within SEP, bypassing Workflow. See the *Degree Works Banner Considerations Technical Guide* for additional information. Access to approve plans within the planner when Workflow is the Approval Method is granted if the user has the `SEPPAUTO` key.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly:
 - `studentPlanner.planner.allowChangesToApprovedPlans`
 - `studentPlanner.planner.planApprovalMethod`
 - `studentPlanner.planner.workflow.plannerEventName`
 - `studentPlanner.planner.workflow.plannerModelName`

Course Validation

In which terms courses will – or will not – be offered can be configured for Plans. When a course is added to a Course or Choice requirement, it will be validated against CFG072. If the course is not

valid for that term, an error message is displayed. The user will be given the option to add the course even though it is not being offered in that term, if the `studentPlanner.planner.addOnlyCoursesOfferedInTerm` setting is false. Otherwise, the course will need to be deleted or moved to a valid term before the plan can be saved.

For additional information on how to define when a course will be offered, see the “CFG072 Planned Courses” section of the *Degree Works Configuration Technical Guide*.

Configuration

- Review CFG072 to ensure it is configured correctly.
- Review the following Shepherd setting to ensure it is configured correctly:

```
studentPlanner.planner.addOnlyCoursesOfferedInTerm
```

Prerequisite Checking

For information on configuring and using Degree Works for prerequisite checking in SEP, see the *Prerequisite Checking Technical Guide*.

Tracking

When tracking is enabled, a student’s completed coursework will be evaluated against their active plan to determine if they have completed their planned requirements by the planned term and are on-track for completing their degree program on-time. A Tracking Status is calculated for requirements, terms, and the overall plan. This information can be used for advising and reporting.

Tracking is assessed on requirements in the current term and in past terms on active and locked/approved plans. The Tracking Status will be either On-Track or Off-Track. For plans that are not active and locked/approved, or requirements and for terms that are beyond the current term, the tracking status will be Not Evaluated.

The tracking process can be run in a batch through DAP58, or on an individual basis through the Web interface when a plan is saved. Each method has an option to update the Official Tracking Status or the Unofficial Tracking Status. The Unofficial Tracking Status does not display on the plan in the Web interface, so it is recommended that processing in the Official Mode on the Web is always enabled. When using the Unofficial Mode in DAP58, you can forecast tracking for future terms and create advising reports to help students who may become Off-Track because they have not planned for the requirements they need.

Note: When Workflow is being used as the approval method for a plan and when the plan is transitioning from a Pending Approval to an Approved state, the tracking status is not evaluated. The tracking status on the requirement, term, and overall plan will not be automatically updated in such a scenario, and DAP58 should be run to update tracking on these plans in batch.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly:

```
studentPlanner.planner.currentTerm.isDeterminedFromUcx
studentPlanner.planner.tracking.enable
studentPlanner.planner.tracking.showTrackingGraphic
studentPlanner.planner.tracking.showTrackingText
studentPlanner.planner.tracking.web.processInOfficialMode
studentPlanner.planner.tracking.web.processTrackingStatusCurrent
studentPlanner.planner.tracking.web.updatePlanTrackingStatus
studentPlanner.planner.tracking.web.updateTermTrackingStatus
studentPlanner.planner.tracking.allowTakenTermDifferentThanPlanned
```

Current Term

As a default, the student's active term is used as the current term in Tracking. However, a site can be configured to calculate the current term for all students based on a term start and stop date. If this setting is set to "true", the current date is compared to the Term Start Date and Term Stop Date values in STU016. The term in which the current date falls between that defined start and stop dates is then used as the current term in SEP.

Configuration

- Review *Term Start Date* and *Term Stop Date* for all terms in STU016 to ensure they are configured correctly.
- Review the following Shepherd setting to ensure it is configured correctly:

```
studentPlanner.planner.currentTerm.isDeterminedFromUcx
```

Requirement Tracking Status

A tracking status is calculated for Course, Choice, GPA, Non-Course, and Test requirements. Tracking can be evaluated for all requirements or just for those requirements that are identified as critical. Additionally, the interface can be configured to only display the tracking status for critical requirements, even if all requirements are included in the actual tracking evaluation. Tracking is not evaluated on Placeholder requirements. If the requirement was completed in or prior to the planned term, and the minimum grade, GPA, or score (if applicable) was met, the requirement will be On-Track. If the requirement was completed in a later term, did not meet the minimum grade, GPA, or score (if applicable), or was not completed at all, the requirement will be Off-Track.

Course and Choice Requirement Tracking

If a class was completed in or prior to the planned term, and at least the minimum grade was earned (if the requirement has a minimum grade), the requirement will be On-Track. If the requirement was completed in a later term, did not meet the minimum grade or was not completed at all, the requirement will be Off-Track.

Note that if the class is in-progress and not graded, it will be considered passed with the minimum grade met. When tracking is triggered after the class is graded, the status will be re-evaluated and if the minimum grade on the requirement was not met, the requirement will then be Off-Track.

The planner can also be configured to consider classes that were taken in any tracked term, by configuring the `studentPlanner.planner.tracking.allowTakenTermDifferentThanPlanned` setting to "true". In that case, if a planned class was taken prior to, during, or even after the planned term, the term and the plan would still be considered On-Track.

Incomplete classes can be configured to meet the requirement as a completed class would, and transfer classes with blank grades or Pass/Fail classes can be configured to meet any minimum grade requirement.

There is no limit to the number of requirements that a course can apply to during tracking. The intent of tracking is to determine if a requirement was completed by a defined point-in-time, as defined in the plan. For example, if the plan contains two requirements, like this:

ENGL 101 (A course requirement)

ENGL @ (A choice requirement)

GPA Requirement Tracking

To evaluate the Overall Degree Works GPA and Major GPA, the planned minimum GPA on the requirement is compared to the actual GPA values calculated by Degree Works for that term. These values are stored in `dap_gpa_history` when CFG020 DAP14 Save GPA History = Y. If a `dap_gpa_history` record cannot be found for the planned term and GPA type, the requirement will be evaluated as Off-Track. Note that because in-progress classes do not have a grade, they will not be included in the Class List GPA calculation. Transfer classes will be included in the GPA calculation.

For a Class List GPA, a combined GPA for all the classes in the requirement list that have been completed in or by that term is calculated by tracking. This is stored on the GPA requirement, and if the calculated combined GPA is equal to or greater than the minimum GPA on the requirement, the requirement will be evaluated as On-Track. If the Class List is a wildcard or a range, the student must have at least one class that falls within the wildcard or range in order for the GPA to be calculated. Note that because in-progress classes do not have a grade, they will not be included in the Class List GPA calculation. Transfer classes will be included in the GPA calculation.

Note that current term GPA requirements will often be evaluated as Off-Track because classes in that term are still in progress and have not been graded. You can force all GPA requirements on the current term to be evaluated as On-Track by setting `studentPlanner.planner.tracking.currentGpaMeetsRequirement` to "true". When this setting is true, validation of Overall Degree Works GPA, Major GPA and Class List GPA requirements will be skipped and these requirements will be On-Track. Once this term becomes a historical term and grading is complete, tracking on these requirements will be evaluated based on the criteria specified above.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly:
`studentPlanner.planner.tracking.currentGpaMeetsRequirement`

Non-Course Requirement Tracking

A Non-Course requirement will be considered On-Track if it was completed in or prior to the planned term. If a Status is specified, this must be equal to the non-course score in order for the requirement to be considered On-Track.

Test Requirement Tracking

A Test requirement will be considered On-Track if it was completed in or prior to the planned term and the minimum score was earned. An alpha score will be evaluated using the Sort Order defined in SEP007.

Configuration

- Review Sort Order in SEP007 to ensure it is configured correctly.

- Review the following Shepherd settings to ensure they are configured correctly:
`studentPlanner.planner.tracking.trackCriticalRequirementsOnly`
`studentPlanner.planner.tracking.blankGradeTransferClassMeetsMinimumGrade`
`studentPlanner.planner.tracking.incompleteClassMeetsRequirement`
`studentPlanner.planner.tracking.passFailClassMeetsMinimumGrade`
`studentPlanner.planner.tracking.showTrackingForCriticalRequirementsOnly`

Term Tracking Status

If one or more trackable requirements is Off-Track, the term will be considered Off-Track. If enabled, a term will initially load as collapsed if all requirements are On-Track.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly:
`studentPlanner.planner.tracking.showOnTrackTermsCollapsed`

Overall Plan Tracking Status

A plan will be considered Off-Track if the number of Off-Track terms on the plan is equal to or greater than the number of Off-Track terms allowed.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly for your SEP environment:
`studentPlanner.planner.tracking.offTrackTermsAllowed`

Batch Processes

The Student Educational Planner relies on three batch processors for data creation and management.

DAP54 – Create Plan from Template Processor

Using DAP54, you can create plans from templates for a group of students that do not already have a plan. A specific template can be specified that will be applied to all students, or the processor can find the best template for the student based on matching Template Tags and student goal data. Temporary plans for students without an active and approved plan can also be created and used in reporting.

DAP58 – Batch Tracking Processor

DAP58 will update the tracking status on active plans for a group of students. Running the processor in the official mode calculates the requirement, term, and tracking statuses that are then

visible on a plan through the Web interface. By using the unofficial mode of DAP58 with a cutoff term that is in the future, tracking statuses can be forecasted and used in advising reports.

DAP59 – Batch Timetabling Processor

DAP59 generates reporting data by running an audit against a student's plan and storing the results in the CPA tables. This data can then be used for reporting and schedule forecasting.

See the *Degree Works Transit User Guide* for more information on batch processes.

Scribe Pointers

Using Scribe Pointers, a Choice requirement on a plan can be linked to a rule in the student's requirement blocks. When a Choice requirement on a plan has a Pointer assigned to it, if a selection hasn't been made then the auditor will apply a course to the associated rule. If a selection has been made on a Choice requirement then that option will be sent to the auditor instead. If the Pointer is blank or a matching rule can't be found then the auditor will not apply the Choice requirement in the planner audit.

SEPPONTER is a reserved word that can be used in IF and HIDE statements to associate the Pointer with a rule. Using a wildcard on a Pointer is also allowed.

There are several ways to Scribe a rule with a Pointer, depending on what results should appear on the audit. Some examples and the expected output are listed below.

Single Class Rules

Example 1

This option allows for a different label on the rule if an unselected CHOICE requirement is applied versus a selected CHOICE requirement or actual class. When the rule is written this way:

```
If (MATH 101 wasnt passed and MATH 102 wasnt passed and MATH 103 wasnt passed
and
    SEPPONTER COREMATH was FOUND) then
    1 class in SEPPONTER COREMATH
    Label "Core Mathematics Choice - Example 1"
Else
    3 credits in MATH 101, MATH 102, MATH 103
    Label "Core Mathematics - Example 1";
```

Then an unselected CHOICE requirement with a COREMATH Pointer will show on the audit like this:

Core Mathematics Choice - Example 1	COREMATH	Requirement planned but choice not selected	PLAN	(3)	Fall 2014
-------------------------------------	----------	---	------	-----	-----------

Note that the number of credits shown on the audit will come either from the Minimum Credits field on the CHOICE requirement or be defaulted to 3.

Using the scribing above the appropriate MATH class will apply to the 3 Credits requirement even with a COREMATH SEPPONTER in place once the student takes the MATH class. When no SEPPONTER is in place and when no MATH class was taken the default of "3 Credits in MATH ..." will appear.

Example 2

This option will show the rule as completed when an unselected CHOICE requirement is applied. When the rule is written this way:

```
If (MATH 101 wasnt passed and MATH 102 wasnt passed and MATH 103 wasnt passed
and
    SEPPONTER COREMATH was FOUND) then
    RuleComplete
    Label "Core Mathematics Choice - Example 1"
Else
    3 credits in MATH 101, MATH 102, MATH 103
    Label "Core Mathematics - Example 1";
```

Then an unselected CHOICE requirement with a COREMATH Pointer will show on the audit like this:

Core Mathematics Choice planned for but not selected

Note that the CHOICE requirement will apply to the Fallthrough section and the credits would apply to the degree.

Multiple Class Rules

Example 3

When more than one class is needed to satisfy a rule, multiple CHOICE requirements can be created to apply to this rule using sequential Pointer values on the plan and a wildcard in the rule. Using the HIDE keyword will apply the unselected CHOICE requirements to the rule:

```
3 Classes in MATH 101, 102, 103, 104, 105, 106, {HIDE SEPPONTER COREMATH@}
    Label "Core Mathematics - Example 3";
```

Then an unselected CHOICE requirement with a COREMATH Pointer will show on the audit like this:

Core Mathematics - Example 3 **COREMATH1** Requirement planned but choice not selected PLAN (3) Fall 2014
Still Needed: 2 Classes in MATH 101 or 102 or 103 or 104 or 105 or 106

Example 4

In this option, using an IF statement allows for different labels if the rule is satisfied by an unselected CHOICE requirement or an actual class:

```
If (SEPPONTER COREMATH@ was FOUND) then
```

```

3 Classes in MATH 101, 102, 103, 104, 105, 106, {HIDE SEPPONTER COREMATH@}
Label "Core Mathematics Choice - Example 4"
Else
3 Classes in MATH 101, 102, 103, 104, 105, 106
Label "Core Mathematics - Example 4";

```

Then an unselected CHOICE requirement with a COREMATH Pointer will show on the audit like this:

<input type="checkbox"/> Core Mathematics Choice - Example 4	COREMATH3 Requirement planned but choice not selected PLAN (3) Fall 2014
Still Needed: 2 Classes in MATH 101 or 102 or 103 or 104 or 105 or 106	

Example 5

When the student's plan has multiple unselected CHOICE requirements that match the SEPPONTER value, they will be applied to the associated rule:

```

If (SEPPONTER COREMATH@ was FOUND) then
3 Classes in MATH 101, 102, 103, 104, 105, 106, {HIDE SEPPONTER COREMATH@}
Label "Core Mathematics Choice - Example 5"
Else
3 Classes in MATH 101, 102, 103, 104, 105, 106
Label "Core Mathematics - Example 5";

```

If the student's plan has CHOICE requirements with COREMATH1 and COREMATH2 Pointers, the audit will look like this:

<input type="checkbox"/> Core Mathematics Choice - Example 5	COREMATH1 Requirement planned but choice not selected PLAN (3) Fall 2014
	COREMATH2 Requirement planned but choice not selected PLAN (3) Spring 2015
Still Needed: 1 Class in MATH 101 or 102 or 103 or 104 or 105 or 106	

Example 6

If the student's plan has CHOICE requirements with COREMATH1, COREMATH2 and COREMATH3 Pointers, the audit will look like this:

<input checked="" type="checkbox"/> Core Mathematics Choice - Example 6	COREMATH1 Requirement planned but choice not selected PLAN (3) Spring 2014
	COREMATH2 Requirement planned but choice not selected PLAN (3) Fall 2014
	COREMATH3 Requirement planned but choice not selected PLAN (3) Spring 2015

Group Rules

Example 7

Pointers can be used in Group rules as well to allow an unselected CHOICE requirement to apply to the best fit option by creating CHOICE requirements on the student's plan for the Group rule options and associating sequential Pointers to the requirements:

```

2 Groups in
(3 Classes in MATH 101, 102, 103, 104, 105, 106, {HIDE SEPPONTER COREMATH@}
Label "Core Mathematics - Example 7") OR
(3 Classes in ART 101, 102, 103, 104, 105, 106, {HIDE SEPPONTER COREART@}

```

Label "Core Humanities and Fine Arts - Example 7") OR
 (3 Classes in ENGL 101, 102, 103, 104, 105, 106, {HIDE SEPPONTER COREENGL@}
 Label "Core English - Example 7") OR
 (3 Classes in CHEM 101, 102, 103, 104, 105, 106, {HIDE SEPPONTER CORESCI@}
 Label "Core Science - Example 7")
 Label "CORE REQUIREMENTS - Example 7";

If the student has a CHOICE requirement for each of the corresponding Pointers, the audit might look like this:

<input type="checkbox"/> CORE REQUIREMENTS - Example 7	Still Needed: Choose from 2 of the following:
<input type="checkbox"/> Core Mathematics - Example 7	COREMATH1 Requirement planned but choice not selected PLAN (3) Fall 2014 (2 Classes in MATH 101 or 102 or 103 or 104 or 105 or 106) or
<input type="checkbox"/> Core Humanities and Fine Arts - Example 7	COREART1 Requirement planned but choice not selected PLAN (3) Fall 2014 (2 Classes in ART 101 or 102 or 103 or 104 or 105 or 106) or
<input type="checkbox"/> Core English - Example 7	(3 Classes in ENGL 101 or 102 or 103 or 104 or 105 or 106) or
<input type="checkbox"/> Core Science - Example 7	(3 Classes in CHEM 101 or 102 or 103 or 104 or 105 or 106)

Example 8

If the student has more than one CHOICE requirement for one of the Pointers, the audit might look like this:

<input type="checkbox"/> CORE REQUIREMENTS - Example 8	Still Needed: Choose from 2 of the following:
<input type="checkbox"/> Core Mathematics - Example 8	COREMATH1 Requirement planned but choice not selected PLAN (3) Fall 2014 (2 Classes in MATH 101 or 102 or 103 or 104 or 105 or 106) or
<input type="checkbox"/> Core Humanities and Fine Arts - Example 8	(3 Classes in ART 101 or 102 or 103 or 104 or 105 or 106) or
<input type="checkbox"/> Core English - Example 8	COREENGL1 Requirement planned but choice not selected PLAN (3) Fall 2014 COREENGL2 Requirement planned but choice not selected PLAN (3) Spring 2015 (1 Class in ENGL 101 or 102 or 103 or 104 or 105 or 106) or
<input type="checkbox"/> Core Science - Example 8	(3 Classes in CHEM 101 or 102 or 103 or 104 or 105 or 106)

Example 9

If the student has three CHOICE requirement for one of the Pointers, the audit might look like this:

<input type="checkbox"/> CORE REQUIREMENTS - Example 9	Still Needed: Choose from 1 of the following:
<input type="checkbox"/> Core Mathematics - Example 9	COREMATH1 Requirement planned but choice not selected PLAN (3) Fall 2014 (2 Classes in MATH 101 or 102 or 103 or 104 or 105 or 106) or
<input type="checkbox"/> Core Humanities and Fine Arts - Example 9	(3 Classes in ART 101 or 102 or 103 or 104 or 105 or 106) or
<input checked="" type="checkbox"/> Core English - Example 9	COREENGL1 Requirement planned but choice not selected PLAN (3) Spring 2014 COREENGL2 Requirement planned but choice not selected PLAN (3) Fall 2014 COREENGL3 Requirement planned but choice not selected PLAN (3) Spring 2015
<input type="checkbox"/> Core Science - Example 9	(3 Classes in CHEM 101 or 102 or 103 or 104 or 105 or 106)

Example 10

If the student has more than one CHOICE requirement for one of the Pointers and either taken or planned for some of the courses, the audit might look like this:

<input checked="" type="checkbox"/> CORE REQUIREMENTS - Example 10	
<input checked="" type="checkbox"/> Core Mathematics - Example 10	MATH 101 College Algebra PLAN (3) Spring 2014 MATH 103 Trigonometry PLAN (3) Spring 2015 COREMATH1 Requirement planned but choice not selected PLAN (3) Fall 2014
<input checked="" type="checkbox"/> Core English - Example 10	COREENGL1 Requirement planned but choice not selected PLAN (3) Spring 2014 COREENGL2 Requirement planned but choice not selected PLAN (3) Fall 2014 COREENGL3 Requirement planned but choice not selected PLAN (3) Spring 2015

Template Management

Template Management is an administrative tool to create and manage plan templates. Users can define typical degree paths that will be used as the basis for creating a student's actual plan.

Access to the Template Management tab is granted if the user has the `SEPTMGMT` key. This key is included in the standard `SEPADV` and `SEPREG` groups.

Template List

In the template list, a user can search for, select, and delete templates. There are two different view options – Tree View and Flat View. If both Tree View and Flat View are enabled, the user can switch to a different view by clicking on the radio button. Additionally, the default view can be configured.

The list of templates that displays in each view can be filtered either by Description or by Template Tags. To filter by description, enter a word or partial string that appears in the template description and hit return.

To filter by Template Tags, click on Advanced Search. In the search window, select the tags you want to use in your search and the required values from the drop down lists. Multiple values can be selected for filtering by checking more than one box. As a default, required Template Tags initially display in the Advanced Search window. Additional Template Tags to be used for filtering can be added by clicking the “+” in the upper right, and tags that are not required can be removed by clicking on the delete icon. All selected values can be cleared by clicking Clear. To filter based on selected Template Tags, click “Apply. To go back to the template list, click Cancel.

To view a template, click on the description link. To create a new template, click New Template. To delete a template, select the template and then click the delete icon.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly:
`studentPlanner.template.showFlatView`
`studentPlanner.template.showFlatViewByDefault`
`studentPlanner.template.showTreeView`
- Access to create a template is granted if the user has the `SEPTADD` Shepherd key.
- Access to delete a template is granted if the user has the `SEPTDEL` Shepherd key.

Tree View

The Tree View presents the selected templates in a hierarchical tree, ordered by Template Tags. The hierarchy is defined by `SEP001 Hierarchy Order`, and the number of open branches initially displayed in the tree is configured by site. Clicking on the arrow next to a label will open the next branch in the hierarchy, and the template is at the final branch of the tree. All branches of the tree can be opened by clicking Expand All, while clicking Collapse All will close the tree down to the first branch.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly:
`studentPlanner.template.initialDepth`

Flat View

The Flat View displays all selected templates in a grid. Templates in the Flat View can be sorted on any column, by clicking on the column header. The template Description and required Template tags will always be displayed, while the Modified Date, What, Who, Template ID, and Term Scheme can be configured to display.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly:
`studentPlanner.template.columnsToDisplay.modifyDate`
`studentPlanner.template.columnsToDisplay.modifyWhat`
`studentPlanner.template.columnsToDisplay.modifyWho`
`studentPlanner.template.columnsToDisplay.templateId`
`studentPlanner.template.columnsToDisplay.termScheme`

Please note that when you are filtering templates by Template Tags chosen in the Advanced Search you are not allowed to sort by the template tag columns. That is, both searching based on tags and sorting by tags is not allowed.

Creating and Editing Templates

A new template can be created while in the template list window or in the template edit view by clicking “New template”. The user will be prompted to choose the Term Scheme to be used when creating the template. A template Description is required. This field is 80 bytes long and must be unique.

The Active flag should be checked to indicate that the template is current and can be used to create plans from within Plans or via DAP54. Template Tags are assigned to a template to assist with organizing and searching for templates. Required Template Tags must be assigned, while Optional Template Tags can be assigned to further enhance organization and searching, but the template can be saved without them.

Configuration

- Access to create a template is granted if the user has the `SEPTADD` Shepherd key.
- Access to delete a template is granted if the user has the `SEPTDEL` Shepherd key.
- Access to modify a template is granted if the user has the `SEPTEDIT` Shepherd key.

Template Tags

Template Tags are used to organize templates in the Tree View as well as to match a student to the best template when creating plans from templates through DAP54. Template Tags are defined in SEP001, and in addition to the reserved tags used by Degree Works, user-defined tags can be defined to be used when organizing or searching for templates. Reserved tags are: CATYEAR, COLLEGE, CONC, DEGREE, LIBL, MAJOR, MINOR, PROGRAM, SCHOOL, and SPEC.

All Template Tags defined in SEP001 will be available in the tag drop downs lists. SEP001 *Required* indicates that the Template Tag is required when creating or modifying a template, while SEP001 *Hierarchy Order* defines the order of the tags in the Tree View. Do not change the settings for SEP001 *Validation Table* or *Match Student Goal Data?* for any of the reserved tag types used by Degree Works.

Configuration

- Review SEP001 to ensure it is configured correctly.

Term Schemes

The template Term Scheme can be changed by clicking “Change Term Scheme” and selecting the new scheme for the template.

Term Schemes define a typical term sequence that should be used as the basis for a template. These are defined in SEP002, and a school may configure several different schemes to accommodate different scenarios such as programs that start in a spring term rather than a fall term, or programs that require a summer term.

Each SEP002 Term Scheme should have a sequenced entry for each term that should appear on the template. The SEP002 *Literal* is what displays in the term header. The SEP002 *Term Type* is used when creating a plan from a template to match the generic template term to an actual term from STU016, and must be either “SUMMER”, “SPRING”, “FALL”, or “WINTER”.

Configuration

- Review SEP002 to ensure it is configured correctly.
- Access to change a template Term Scheme is granted if the user has the `SEPTTRMS` Shepherd key.

Creating and Editing Requirements

An unlimited number of requirements can be added to a template term by either clicking on the “+” on the requirement card in the Requirements menu on the sidebar or by dragging and dropping the requirement card to the desired term. The options in this menu come from the requirement types in SEP004 with Show in SEP Picklist flag = Y. The description for these requirement types can be

modified by the site, but the keys should not be changed or deleted. There are 6 requirement types: Course, Choice, GPA, Non-Course, Test and Placeholder.

When adding a requirement, a dialog will open where the details of the requirement can be added. If the requirement card was dragged and dropped into a term, that term will load automatically. If added from the sidebar, the user can select the term to which the requirement should be added.

If the Critical indicator is configured to display, it will appear on all requirement types except Placeholder. It is used to identify requirements that are critical for tracking.

To edit a requirement, select “Edit this requirement” from the action menu in the right corner of a requirement.

To delete a requirement, select “Delete this requirement” from the action menu in the right corner of a requirement.

To move a requirement to another term, select “Reassign this requirement” from the action menu in the right corner of a requirement or drag and drop it to the desired term.

Configuration

- Review the Show in SEP Picklist flag in SEP004 to ensure it is configured correctly.
- Review the following Shepherd settings to ensure they are configured correctly:
`studentPlanner.courseReq.showCritical`
- Access to add and edit requirements on a template is granted if the user has the `SEPTRQAD` key.
- Access to delete requirements on a template is granted if the user has the `SEPTRQDL` key.
- Access to only modify existing requirements on a template is granted if the user has the `SEPTRQED` key.

Course

A single class can be planned for with the Course requirement. In addition to adding individual Course requirements from the Requirements menu on the sidebar, multiple Course requirements can be added to a term using the requirement drawer. This is accessed by clicking in the “+” button in a term.

When adding a Course requirement from the Requirements menu on the sidebar, enter the course number in the “Course requirement” field. The user’s input will be validated before the requirement can be saved.

When adding Course requirements from the requirement drawer, the user can filter the list of all courses by subject, or by course title. One or more requirements can be selected while in the requirement drawer, and then added at once by clicking “Add to plan”.

The Credits will automatically load from the course record in the database. If the course is a variable credit course, then the value in `rad_credits` (typically this is equal to `rad_credits_high`, per the extract) for that course is displayed, and the user can change the number of credits to a value

within the range for that course. The format and length of the credits can be configured by the site.

If the Minimum Grade is configured to display, the user can select the minimum grade needed for the course. This can be used as criteria in tracking. The values in this dropdown come from the grades in STU385 with *Show in SEP Picklist* flag = Y.

If the Campus is configured to display, the user can select the campus where the course should be taken. This field is informational only. The values in this dropdown come from the location codes in STU576 with *Show in SEP Picklist* flag = Y.

If the Delivery is configured to display, the user can select a content delivery method for the course. This field is informational only. The values in this dropdown come from the delivery codes in SEP003 with *Show in SEP Picklist* flag = Y.

Configuration

- Review STU050 and SEP009, as well as the *Show in SEP Picklist* flag in SEP003, STU576 and STU385 to ensure they are configured correctly.
- Review the following Shepherd settings to ensure they are configured correctly for your SEP environment:

```
core.credit.format
studentPlanner.courseReq.showCampus
studentPlanner.courseReq.showDelivery
studentPlanner.courseReq.showGrade
```

Choice

When there are several course options that can fulfill a planned requirement, a Choice requirement can be used. An unlimited number of options can be defined in the Choice requirement, and an option can be any combination of actual courses, wildcards or ranges, for example, STAT 101 OR MATH @.

Each option can have up to two items. An item can be an actual course such as MATH 101, a wildcard such as @ @, MATH @, or MATH 3@, or a range such as MATH 100:299. If configured, a user can specify a course attribute on any item. The values in this dropdown come from the course attribute codes in STU050. STU050 can be loaded through the extract, or manually by the user.

To add an option, click “Add” link and enter the course number in the “Course requirement” field. The user’s input will be validated before the requirement can be saved.

To add an additional item to an option, click “Add a paired course or lab”.

One of the options may be selected by clicking on the radio button next to the preferred option. Clicking the “Clear selection” button will remove the selection from the option.

The Minimum Credits for this requirement can be manually entered but are not required. The format and length of the minimum credits can be configured by the site.

If the Pointer is configured to display, the user can define a value that links this requirement to a

rule in a requirement block. The values in this dropdown come from the pointer codes in SEP009 with *Show in SEP Picklist* flag = Y. See the *Scribe Pointers* section in this document for additional information.

If the Minimum Grade is configured to display, the user can select the minimum grade needed for the course. This can be used as criteria in tracking. The values in this dropdown come from the grades in STU385 with *Show in SEP Picklist* flag = Y.

If the Campus is configured to display, the user can select the campus where the course should be taken. This field is informational only. The values in this dropdown come from the location codes in STU576 with *Show in SEP Picklist* flag = Y.

If the Delivery is configured to display, the user can select a content delivery method for the course. This field is informational only. The values in this dropdown come from the delivery codes in SEP003 with *Show in SEP Picklist* flag = Y.

Configuration

- Review STU050 and SEP009, as well as the *Show in SEP Picklist* flag in SEP003, STU576 and STU385 to ensure they are configured correctly.
- Review the following Shepherd settings to ensure they are configured correctly for your SEP environment:

```
core.credit.format
studentPlanner.choiceReq.enableCourseAttribute
studentPlanner.choiceReq.enableScribePointer
studentPlanner.courseReq.showCampus
studentPlanner.courseReq.showDelivery
studentPlanner.courseReq.showGrade
```

- Access to select a choice requirement option is granted if the user has the SEPPSEL Shepherd key.
- Access to change a choice requirement pointer is granted if the user has the SEPPPTR Shepherd key.

GPA

Four types of minimum GPAs for a term can be planned for with the GPA requirement – Overall Degree Works GPA, Overall Student System GPA, Major GPA, and a Class List GPA. These can be used as criteria in tracking. The GPA types that are in the dropdown come from the codes in SEP008 with *Show in SEP Picklist* flag = Y.

The format and length of the minimum GPA can be configured by the site.

When adding a Major GPA, a major must be specified. The majors that are in the dropdown come from the major codes in STU023 with *Show in SEP Picklist* flag = Y.

When adding a Class List GPA, a group of classes in which the student needs to have earned a minimum GPA can be defined. Each class should be separated by a plus sign, and ranges and wildcards can be included in the list. Classes are validated, but ranges and wildcards are not.

Configuration

- Review the *Show in SEP Picklist* flags in SEP008 and STU023 to ensure they are configured correctly.
- Review the following Shepherd settings to ensure they are configured correctly:

```
core.gpa.format  
core.gpa.maxLength
```

Non-Course

Non-Course items that need to be completed as part of a degree can be planned for with the Non-Course Requirement. These can also be used as criteria in tracking. The Non-Course Requirement types that are in the dropdown come from SCR003.

The Status can be left blank to indicate that the student just needs to have completed the non-course item. It can also be the alphanumeric score or status needed to consider the requirement complete.

Configuration

- Review SCR003 to ensure it is configured correctly.

Test

Tests that need to be completed as part of a degree can be planned for with the Test Requirement. These can also be used as criteria in tracking. The Test Codes that are in the dropdown come from the codes in SEP006 with *Show in SEP Picklist* flag = Y. A Minimum Score is required, and the length, type, and valid score range of the Minimum Score can be defined in SEP006. Note that alpha scores require additional setup in SEP007 to define the sort order for these values.

If you will be using the Create Block functionality, your SEP006 Test Codes must also be in SCR002 Custom Data. If it does not, the creation of the block will fail with a parsing error.

Configuration

- Review the *Show in SEP Picklist* flag, *Score Length*, *Score Type*, *Minimum Valid Score*, and *Maximum Valid Score* in SEP006 to ensure that they are configured correctly.
- Review the *Sort Order* in SEP007 to ensure that it is configured correctly.

Placeholder

User-defined requirements that need to be completed as part of a degree can be planned for with the Placeholder Requirement. These requirements are informational only and are not included in tracking or do not apply to planner audit rules. The Placeholder Requirement types that are in the dropdown come from the codes in SEP005 with *Show in SEP Picklist flag* = Y.

The Value is a free-form text field and is required.

Configuration

- Review the *Show in SEP Picklist flag* in SEP005 to ensure it is configured correctly.

Creating and Editing Notes

An unlimited number of notes can be added to requirements, terms, or to the overall template. All notes are viewable to all users, and users can be given access to add, modify, and/or delete notes on each of these template components.

Click on the note icon to open the note list window, where you can choose to view, add, edit, or delete notes, depending on the access they have been granted.

To add a note, click on the “Add a new note” button. Notes can have an unlimited amount of text. If the note should be included when a plan is created from a template, select the “Copy to plan” checkbox. If the note should be marked as internal when it is included on a plan created from a template, select the “Internal on plan” checkbox. Click “Save note” to keep your note, or “Cancel” to go back to the note list window without keeping your note.

To view a note, select “View note” from the action menu in the right corner of a note.

To edit a note, select “Edit note” from the action menu in the right corner of a note.

To delete a note, select “Delete note” from the action menu in the right corner of a note.

Configuration

- Access to add and edit template notes is granted if the user has the `SEPTNTAD` Shepherd key.
- Access to delete template notes is granted if the user has the `SEPTNTDL` Shepherd key.
- Access to edit only template notes is granted if the user has the `SEPTNTED` Shepherd key.
- Access to add and edit term notes is granted if the user has the `SEPTNMAD` Shepherd key.
- Access to delete term notes is granted if the user has the `SEPTNMDL` Shepherd key.
- Access to edit only term notes is granted if the user has the `SEPTNMED` Shepherd key.
- Access to add and edit requirement notes is granted if the user has the `SEPTNRAD` Shepherd key.
- Access to delete requirement notes is granted if the user has the `SEPTNRDL` Shepherd key.

- Access to edit only requirement notes is granted if the user has the `SEPTNRED` Shepherd key.

Sidebar

A sidebar panel of all courses available and requirement types can be found to the left of the template. Courses from the Courses list as well as Requirement types can be dragged and dropped into a term on the template.

Courses

The Courses list will show all courses in the `rad_course_mst` ordered by discipline. Clicking on the arrow next to the discipline will expand the list to show all available courses in that discipline.

Requirements

The Requirements menu is a list of all the requirement types that can be used on a plan. The options in this menu come from the requirement types in SEP004 with *Show in SEP Picklist* flag = Y. The description for these requirement types can be modified by the site, but the keys should not be changed or deleted. There are 6 requirement types: Course, Choice, GPA, Non-Course, Test and Placeholder.

Configuration

- Review the *Show in SEP Picklist* flag in SEP004 to ensure it is configured.

Course Link

If Course Link is enabled, users will be able to click to get additional information such as the course description, meeting times, instructors, and prerequisites on courses in the Courses list.

To display Course Link information, click on the actual course to display Course Link information.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly:
`studentPlanner.courseLink.enable`

Transfer Finder

Transfer Finder allows a student at one Degree Works school to compare their completed classwork to the requirements at a partner Degree Works school. The goal is to let a student see their transfer potential at the partner school by sending their local classwork to the partner school and generating transfer audit. For additional information on using Transfer Finder, please see the *Transfer Finder Administrative Guide*.

Admin

The Admin menu contains functions that are not in the student context and are typically available to a limited set of users. This section covers the Debug and Theme functions, while Exception Management is covered in a separate section.

Debug

The Responsive Dashboard allows users to enable debugging to troubleshoot issues in the application for their session. Users with access to this functionality will be able to enable and disable debug from the Debug page, which is located under the Admin menu. All debugging output for the application is aggregated into one file, even if multiple applications or APIs are involved. Access to the application server is not needed to enable debugging. However, access to the server is required to access the generated log files. Debug files for the user interface can be found in the logs directory of the application server, while debug files for functionality like generating audits can be found in the logdebug directory of the classic server.

Enabling Debug

On the Debug page, toggle the Enable Debug switch to turn debugging on for the application. A random Debug Tag will be generated, and the user can copy this to their clipboard by clicking on the page icon. This Debug Tag will be appended to each line of debug in the log file so that the user can easily identify which lines are from their session, as opposed to other users using the application at the same time.

After enabling debug, the user can then resume using Responsive Dashboard and execute the steps to duplicate the issue they are experiencing. Debug should be turned off by going back to the Debug page and toggling the Enable Debug switch once the issue has been duplicated. Debug for this user's session will also be turned off when they log out of the application.

Accessing Debug Files

Debug will appear in the Responsive Dashboard log file on the application server. This is typically defined by the `$LOGGING_FILE` environment variable.

Searching for the user's Debug Tag will isolate the lines of debug specific to their actions. For example:

```
2017-10-24 16:22:23,144 68b4e19d-3fb4-4975-9815-83eea01266ff DEBUG [41-exec-17]
.h.d.s.s.StatelessPreAuthenticatedFilter : Checking secure context token: null
```

The packdebug script can be used to collect the log files generated for a user session's Debug Tag. For more information on using this script, please refer to the *Degree Works Technical Guide*.

Configuration

- Access to Debug is granted if the user has the `DEBUG` Shepherd key.

Theme

The color scheme and logo in Responsive Dashboard can be easily localized for your institution. Users with access to this functionality can configure the primary, secondary and tertiary colors as well as the logo using the Theme Editor, which is located under the Admin menu. Note that changes made in Theme will impact all users, so access to this functionality should only be given to users who are authorized to make these changes.

To change the color theme of your Responsive Dashboard deployment, specify the Hex color code for the Primary color and Secondary color. The Tertiary color also has limited configurability by selecting an option from the dropdown of available colors.

To change the logo, specify the URL of the graphic to be used. Note that this URL must be accessible to users of the Responsive Dashboard, so may need to reside outside your network and/or firewall.

Click “Save” to enable your changes.

The default Ellucian theme can be restored by clicking “Restore standard theme”.

Configuration

- Access to Theme is granted if the user has the `SDADMIN` Shepherd key.

Other Localizations

The text, labels and messages that appear in the application are defined in `DashboardMessages.properties` and can be localized for your institution. For information on how to modify application properties, please see the “Localizing an Application” section of the *Composer Administrative Guide*.

Links

The Links menu contains external links that you can add to connect from Degree Works to other locations on campus or anywhere else.

To configure this functionality, define the items to display and the order in which to display them in the `dash.navigation.links.items` setting. Then use Composer to add localized properties for the link label and URL. These should be on `DashboardMessages.properties` and each link should have a label property in the format of `dash.navigation.links.xxxx.label` and a URL property in the format of `dash.navigation.links.xxxx.url`, where “xxx” is the lowercase code for the link defined in `dash.navigation.links.items`.

Configuration

- Review the following Shepherd settings to ensure they are configured correctly :
`dash.navigation.links.items`
- Access to Links is granted if the user has the `EXTLINKS` Shepherd key.