

SAVVAS EasyBridge

EasyBridge Plus Onboarding Guide

About this Guide

The information in this guide includes high level descriptions of topics related to Savvas EasyBridge Plus – including setup, onboarding, and implementation of this single sign-on, auto-rostering solution.

This document does not serve as a self-installation guide, but rather to help increase knowledge of the technical solution. If you are a new EasyBridge Plus customer and have not yet met with a Solution Engineer on our Integration & Rostering Services team for a Discovery/Planning call, contact your Account General Manager and get started today!

Released: May 11, 2020

Document Owner: Integration & Rostering Services

This edition applies to the most recent release of EasyBridge Plus.

The data and names used to illustrate the reports and screen images may include names of individuals, companies, brands, and products. All sample data and names are fictitious. Any similarities to actual names are entirely coincidental.

Contents

Released: May 11, 2020	2
EasyBridge Plus Integration Overview	5
Integration Project Requirements	6
EasyBridge Plus – The User Experience	7
EasyBridge User Login URL	7
SAML SSO Configuration	7
Example: ADFS	8
Example: Google	8
EasyBridge Plus – The Data Files	9
Data File Formats	9
Data Upload Process	9
Additional Information	10
SFTP Site Information	11
Data Import Processing Modes	11
Sample Data Ingestion Summary Email Message:	12
Creating Data Import Files	15
New Schools – Action Required	17
Pause Period - New Year Rollover (NYR) Process	17
Appendix A – OneRoster v1.1 (.csv) Specifications	18
manifest.csv	19
academicSessions.csv	20
classes.csv	21
courses.csv	23
enrollments.csv	24

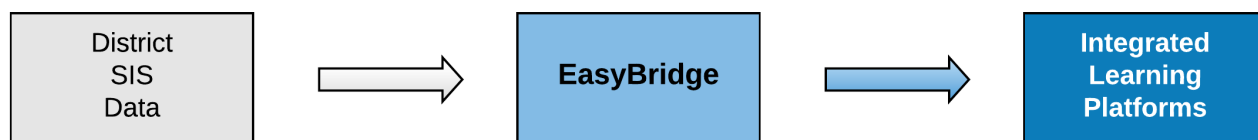
orgs.csv	25
classResources.csv	28
courseResources.csv	28
resources.csv	29
Appendix B – OneRoster 1.1 & Product Associations	30
Appendix C – OneRoster v1.1 Data Relationships	32
Appendix D – Legacy Text File Format (.txt) Specifications	33
SCHOOL.txt	34
STAFF.txt	35
STUDENT.txt	36
PIF_SECTION.txt	37
PIF_SECTION_STAFF.txt	39
PIF_SECTION_STUDENT.txt	40
ENROLLMENT.txt	41
ASSIGNMENT.txt	42
Appendix E – Legacy Text File Format (.txt) Data Relationships	43
Appendix F – Expected Data Behaviour	44
Appendix G – Types of User Accounts	51
Appendix H – EasyBridge Administrator Portal	52

EasyBridge Plus Integration Overview

One of the key components of an auto-rostering integration is EasyBridge. EasyBridge is a cloud-based data store that is used to securely integrate your SIS data with our online learning platforms, including Realize, SuccessNet, eText, MathXL for Schools, SuccessMaker, and Classview iLit.

Technical features of EasyBridge include the following:

- Secure cloud-based storage in a unified data model
- Enterprise administration portal for support and management
- Integration with your identity management solution



On a nightly basis, data is extracted from the student information system (SIS) and uploaded to a provided SFTP (Secure File Transfer Protocol) site. EasyBridge automatically detects the upload and processes the files. After the files are ingested, the updated roster data will be pushed to the integrated learning platforms based on your district's subscription: Realize, SuccessNet, Dash, ClassView (including iLit, myPerspectives ELD Companion, and Words Their Way), MathXL for School, SuccessMaker, and standalone eText.

Savvas Learning Company is committed to providing a seamless user experience for our online learning platform customers through **EasyBridge Plus** (SSO & auto-roster synchronization). Implementing this integration can improve your district's user experience by:

- Centralizing access to online curriculum products
- Reducing the number of separate user logins
- Easing the administrative burden of managing class rosters

Additionally, with the EasyBridge Plus single sign-on solution, teachers and students can authenticate into EasyBridge plus using their district identity provider (IdP) credentials. The district IdP must be SAML 2.0 enabled for use when configuring SSO EasyBridge logins for teachers and students.

Integration Project Requirements

The following list provides the **minimal** technical requirements that the customer must meet to effectively integrate roster synchronization and single sign-on with EasyBridge Plus:

1. Provide nightly data submissions to EasyBridge as discussed in detail within this document
2. Provide a central location to post the new EasyBridge Plus user login URL for easy user access both from school and when accessing content from home
3. Provide a single identity provider that contains accounts for both students and teachers. This identity provider must support single sign-on with SAML 2.0 and include the following:
 - a. An exposed secure federated login page on the internet
 - b. If using the eText App for iPad or Android, the federated login page must not use a browser pop-up window. If your login page must use a browser pop-up window, please discuss a possible alternative eText App login method with your EasyBridge Solutions Engineer
 - c. A Secure Socket Layer (SSL) certificate. (***This is required for secure access and will need to be purchased if not yet available***)
 - d. The ability to exchange metadata for setting up trust relationship
 - e. The ability to configure the relying party trust within your SAML provider
 - f. The district must have the ability to host the customized EasyBridge Plus portal single sign-on link

There are many aspects to an EasyBridge auto-rostering integration project with the technical set up of the files and SSO only being one facet. Below is a list of additional integration project requirements, designed to ensure a successful integration and roll out to district users:

1. Identify project stakeholders from both the curriculum and IT departments to participate in the integration project process
2. Identify general timelines for the project, keeping with potential training dates and key school year calendar dates
3. Clearly define stakeholder tasks in order to ensure project timelines are met
4. Work with an assigned solution engineer to generate clear login communications documentation for district users
5. Identify one or more district EasyBridge administrator(s) that will maintain the scheduled nightly upload of the data files as well as the integrated environment once the integration project has been completed. (SE will create EB Admin accounts for these users and will review Admin Account user interface in the project Orientation meeting.)

EasyBridge Plus – The User Experience

EasyBridge User Login URL

The EasyBridge Plus user experience will begin with the user login process. The district will be issued a unique EasyBridge Plus Login URL. This login portal link serves three functions:

- Provides a user federated login through single sign-on
- One login URL provides both teacher and student user account holders federated login access to one integrated environment
- Users can access several integrated platforms from one integrated environment. It consolidates the login process down to one login, many platforms

The EasyBridge Plus login URL has the following pattern:

<https://sso.rumba.pk12ls.com/sso/login?service=https://cat.easybridge.pk12ls.com/ca/dashboard.htm&idpmetadata=>

The second part of the URL is specific to each customer and usually matches the district's metadata URL:

<https://mydistrict.org/mymetadata/federationmetadata.xml>

Put together, a sample EasyBridge Plus login URL might look like the following:

<https://sso.rumba.pk12ls.com/sso/login?service=https://cat.easybridge.pk12ls.com/ca/dashboard.htm&idpmetadata=https://mydistrict.org/mymetadata/federationmetadata.xml>

SAML SSO Configuration

EasyBridge Plus has the following requirements for a successful SAML SSO configuration:

- The customer's identity provider **must** have an externally-accessible metadata URL, or the metadata file must be made available to EasyBridge for hosting. (If utilizing EB hosted metadata, the customer is responsible for notifying EasyBridge Support whenever the contents of the metadata file must be updated, i.e. SSL certificate updates, etc)
- The customer's identity provider may be configured to use EasyBridge's metadata URL, or the metadata file may be imported into the identity provider
- EasyBridge's metadata URL is **<https://sso.rumba.pk12ls.com/sso/SPMetadata>**
- SAML responses must be signed at the response level
- SAML responses must not use encryption
- SAML responses must include a custom attribute named *authenticationId*.

NOTE: The value of *authenticationId* for a given user must match the *federated_id* value in the nightly data upload for the same user. You will work with your Solutions Engineer to determine the available options for *authenticationId* and *federated_id*.

Example: ADFS

If you are currently using Active Directory Federation Services (ADFS) to configure SAML SSO, the list below outlines a standard configuration. SAMAccountName is given in this example, but it is possible to use a different attribute value when configuring the custom claim rule.

- Attribute Value name = **authenticationId**
- Attribute Value = SAMAccountName
- **Encryption** of outbound assertions = **OFF**
- Message and **assertions** must be **signed**
- EasyBridge's metadata location: <https://sso.rumba.pk12ls.com/sso/SPMetadata>

Example: Google

If you are currently using Google to configure SAML SSO, the list below outlines a standard configuration. It is possible to use a different attribute when configuring the attribute mapping, however, primary email is highly recommended.

- ACS URL and Entity ID: <https://sso.rumba.pk12ls.com/sso/login>
- Start URL: <https://cat.easybridge.pk12ls.com/ca/dashboard.htm>
- Signed Responses = checked
- Attribute Mapping
 - Name = authenticationId
 - Value = Primary Email

EasyBridge Plus – The Data Files

Data File Formats

While EasyBridge is compatible with both the OneRoster and our legacy text file formats, ***the recommended data file format for EasyBridge Plus is OneRoster version 1.1.***

Using OneRoster provides industry-standard formats to exchange roster data between different educational systems. Detailed file specifications along with corresponding data examples, can be found in **Appendix A** of this guide. The ability to add product associations to your section data within your data file upload is an added benefit of producing OneRoster 1.1 files. Detailed file specifications along with corresponding data examples, can be found in **Appendix B**.

If your Student Information System does not support OneRoster version 1.1, EasyBridge continues to accept data files in our legacy text file format (seven required files and two optional but recommended files). Detailed file specifications along with corresponding data examples, can be found in **Appendix D** of this guide.

Data Upload Process

In a standard EasyBridge Plus integration, the district must upload their roster data from the Student Information System to the Savvas SFTP site for import into EasyBridge. There are several ways to do this:

- A district can script their own files in either OneRoster 1.1 or the legacy text file format and push the data to the Savvas SFTP site as a scheduled nightly task.
- Depending on the district's SIS, there might be an API available to configure a data pull from the district SIS to the Savvas SFTP site. APIs are currently available for:
 - Skyward
 - Infinite Campus
 - Aeries
 - ClassLink
 - NY RICOOne Customers
- Some SIS vendors have built OneRoster csv file extract tools into the SIS. The district can use one of these to export OneRoster files and push the upload to the Savvas SFTP site
- PowerSchool has a PS-Plugin tool that can be used to configure, export, and upload the legacy text files to the Savvas SFTP site

Regardless of the method by which a district exports their data files and uploads them to the SFTP site, every nightly upload must include all the required files. Within these files, the required fields must be in the header rows and data value formats must be followed. Please be sure to review **Appendix A** for OneRoster 1.1 file requirements and **Appendix D** for the legacy text file format requirements.

Additional information regarding the data file upload process:

- The Solution Engineer assigned to your integration project will generate your district specific SFTP site directory of folders and provide you with your unique username and password.
- Your SFTP directory is specific to your district and will only contain your data.
- A district can log into their SFTP site directory anytime using a secure client like FileZilla or WinSCP.
- For added security, once the files are ingested each night, they are removed from the SFTP server.
- Files must be uploaded into a district's SIS folder on the SFTP site.
- Files should be zipped together then uploaded to SFTP site. Zipped folders can be named with any naming convention and must be binary. ASCII zips cannot be processed.
- If a district cannot zip the files before upload, EasyBridge can accept unzipped files, however your assigned Solution Engineer will need to configure your district for this type of upload.
- Unzipped data files will not be picked up for import until 90 mins of inactivity have been measured by EasyBridge in the customer's SIS folder on the SFTP site.
- EasyBridge recommends the upload be scheduled to execute nightly at 8:00 PM, local time.
- To configure an API data pull, the district will need to provide their Solution Engineer with the SIS API or Host URL, Consumer Secret, and Consumer Key.
- The data ingestion process runs on a diff import and will only detect and write changes to the database related to new, modified, and/or deleted records. On occasion, tech support may run a full import to resolve issues in the integrated environment.
- Records removed from the upload will be deleted in EasyBridge. Threshold check will stop an ingestion if more than 50% of the records from that file will be deleted in EasyBridge.
- After each ingestion specific district contacts will receive an EdCloud Ingestion Summary Email which will notify the district whether the import ran or failed and will include a count of errors and/or warnings by file.
- After ingestion a detailed error report will be placed on the SFTP site in the district's REPORTS folder. A district can log into the SFTP site and download a detailed error report anytime.
- EasyBridge scans district SIS folders on the SFTP site 24 hours a day. Anytime a district uploads a dataset to their SIS folder, EasyBridge will pick it up for ingestion.

SFTP Site Information

The Host SFTP location is: **sftp.pifdata.net**

The transmission port is: **22**

The destination folder: **SIS**

The district username and password to this site will be unique to your district and is provided by your Solution Engineer. This information will be sent to the district via encrypted communication.

Data Import Processing Modes

There are three modes of operation for processing and monitoring data ingestion files:

- **OFF Mode** – is typically used with new customers going through integration for the first time. In this mode, files land on the SFTP site and stay there until the district is set to either Dry Run or On mode. This allows the Solutions Engineer to manually check the initial set of files before first Dry Run import. If a district that has completed integration and is live in their integrated environment begins to see threshold checks stop their data imports, its possible for support to set them to off mode until the problem in the data file is corrected. Otherwise, live customers should not be in OFF mode, especially during a currently running school year.
- **Dry Run Mode** – is typically used with initial integration projects and during the Pause Period between school years. In this mode, files are uploaded to the SFTP site, and EasyBridge picks them up and runs them through a pre-ingestion validation analysis. Errors and warnings within the individual files will be reported to the district. Data is not written to the database in Dry Run mode and validation will analyse the data WITHIN a given file, it will not validate relational fields across files. This means that data previously imported into EasyBridge will stay frozen in time until the ingestions are taken out of Dry Run mode and put back into On or Production Mode. Again, live customers should not be in Dry Run mode during a currently running school year.
- **“ON” or Production Mode** – is typically used by live districts during a school year. In this mode, the changes in the data files are written to the database. It will take up to 4 hours for changes in the data set to fully sync throughout EasyBridge and the integrated learning platforms. Validation in On mode will analyse not only errors and warnings within the individual files but also in the relational fields across files. The only time a district will come out of On mode for ingestions is between school years while on Pause Period.

If you experience any errors or warnings when uploading data files to EasyBridge, please reference the appropriate data ingestion error guide:

[OneRoster file format ingestion errors](#)

[Legacy Text file format data ingestion errors](#)

Sample Data Ingestion Summary Email Message:

The email summary each night will include totals of the records processed, including those that changed, did not change, produced errors and warnings, and those that were deleted. The same data import summary will also be saved to your SFTP REPORTS folder. Should any of your files contain errors or warnings, this will be reflected in the subject line of your summary email with "COMPLETED WITH ERRORS." Below is a sample of the summary email:

Subject Line - COMPLETED: EdCloud Ingestion for Apple Grove - Demonstration Instance (PRD)

Tue Mar 03 21:06:00 EST 2018 [COMPLETED] ProcessStart

Tue Mar 03 21:06:00 EST 2018 [COMPLETED] StartImport

Tue Mar 03 21:06:00 EST 2018 [COMPLETED] GetImport

Tue Mar 03 21:06:01 EST 2018 [COMPLETED] GetIngestionZip

Tue Mar 03 21:06:01 EST 2018 [COMPLETED] GetOriginalZip

Tue Mar 03 21:06:01 EST 2018 [COMPLETED] CheckIngestionForEmptyDistrict: District file is not empty.

Tue Mar 03 21:06:01 EST 2018 [COMPLETED] CheckIngestionNewYearRollover: No New Year Rollover

Tue Mar 03 21:06:01 EST 2018 [COMPLETED] TypeOfZipFile

Tue Mar 03 21:06:01 EST 2018 [COMPLETED] District

Changed = 0

Unchanged = 1

Skipped = 0

Deleted = 0

Total = 1

Tue Mar 03 21:06:01 EST 2018 [COMPLETED] School

Changed = 0

Unchanged = 12

Skipped = 0 (Errors = 0, Warnings = 5)

Deleted = 0

Total = 12

Data File Formats – Quick Reference

OneRoster version 1.1 (.csv) Files – REQUIRED	
manifest.csv	A static file indicating which files are included in the upload
academicSessions.csv	A list of defined terms for the current school year
classes.csv	A list of sections offered by each school
courses.csv	A list of courses offered
enrollments.csv	Associates teachers, co-teachers, and students to class sections
orgs.csv	District/LEA information as well as a list of schools in the district
users.csv	Staff, teacher, and student information
OneRoster version 1.1 (.csv) Files – OPTIONAL (required only for product association to sections)	
classResources.csv	Establishes relationship of product to section
courseResources.csv	Establishes relationship of product to course
resources.csv	List of online products to be associated to sections

Legacy Text File Format (.txt) - REQUIRED	
CODE_DISTRICT.txt	District or Local Education Agency (LEA) information
SCHOOL.txt	A list of schools in the district
PIF_SECTION.txt	A list of sections/classes offered by each school
STAFF.txt	Staff/teacher information
STUDENT.txt	Student information
PIF_SECTION_STAFF.txt	Associates teachers to classes/sections
PIF_SECTION_STUDENT.txt	Associates students to classes/sections
Legacy Text File Format (.txt) - OPTIONAL	
ASSIGNMENT.txt	Used if district requires user accounts for non-teaching staff (administrators, SPED resource teachers, classroom aides, etc.) Access is given on a school-by-school basis.
ENROLLMENT.txt	Association between the student and their grade level and school. (REQUIRED file if SuccessMaker is being integrated)

Creating Data Import Files

	OneRoster v1.1 (.csv)	Legacy Text Files (.txt)
Files	Required character encoding is UTF-8. Incorrectly named files will not process properly.	
	<p>Must match the data specification and include the .csv extension</p> <p>File names must be named in the proper mixed-case capitalization with lower-case extension.</p> <p>CORRECT: academicSessions.csv INCORRECT: academicsessions.csv INCORRECT: AcademicSessions.csv</p>	<p>Must match the data specification and include the .txt extension</p> <p>File names must be in UPPER-CASE with lower-case extension.</p> <p>CORRECT: STUDENT.txt INCORRECT: STUDENT.TXT INCORRECT: student.txt</p>
SFTP Folders	<p>SIS: Stores the uploaded zip files</p> <p>REPORTS: Stores error and warning details on file ingestion issues</p>	
Header Rows	A header row is required for each file. Field names are case-sensitive. The order and number of fields in a given file, needs to be consistent between the header row and data rows. Fields are either required or optional. All <i>required</i> fields in the file specification MUST be included, otherwise the data import will be rejected.	
	<p>The header row must be a comma delimited list of <i>ALL</i> field names.</p> <p>Field names must be listed in the header row in the order given by the OneRoster specification</p>	<p>The header row must be a comma delimited list and contain, at a minimum, all required field names.</p> <p>Header fields can be enclosed in quotes, but it is not required.</p> <p>Optional fields may be left out of the header and data rows.</p>

	OneRoster v1.1 (.csv)	Legacy Text Files (.txt)
Data Rows	Data fields must be comma delimited. Data must be enclosed in double quotes. Fields without data should be represented as an empty string (""). Dates should be represented as a string in the following format: "yyyy-mm-dd". For date fields requiring only a year, the format is "yyyy". Boolean values should be represented as "true" or "false" and are case-sensitive. All trailing space characters at end of line should be removed. Otherwise, a single double quote or comma within a data field will cause processing to fail.	
	Fields containing special characters like commas, hyphens, or apostrophes must be enclosed in double-quotes. If double-quotes are used to enclose a field, then a double-quote appearing inside the field must be escaped by preceding it with another double-quote.	A single double-quote character (") within a data field must be encoded as a pair of double quotes ("). When loaded, the pair will be replaced with a single double quote. Note: <i>If not including data for optional fields, the field can be omitted from the header row.</i>
Source Ids	It is important to note that your source id field must never change. (This is also referred to as immutable.) We recommend using database keys such as a database primary key and not a value that can change such as a student's email address or student number. Source ids are case sensitive and unique. Source id = "ABC" will be treated as a separate record from source id= "abc". Note: <i>Changing a source id value for a teacher or a student will result in a new record being created and any pre-existing teacher/student data will be lost.</i>	
	Fields highlighted in yellow are source id values that uniquely identify records and are used to make connections between files. See Appendix C for a complete data relationship diagram.	Fields highlighted in yellow are source id values that uniquely identify records and are used to make connections between files. See Appendix E for a complete data relationship diagram.

New Schools – Action Required

After initial implementation, it is important to notify Technical Support whenever new schools are added to the data. Currently, a manual configuration step is required to map the new incoming school data to the Savvas internal licensing database. Users from a school that is not mapped will not be able to log into your new EasyBridge integrated environment.

Pause Period - New Year Rollover (NYR) Process

New Year Rollover (NYR) is a process that helps prepare districts for the new school year in EasyBridge and several of the integrated learning platforms, including Realize, SuccessNet, and Dash. NYR, or rollover, is synonymous with the first day that new school year classes can be imported into EasyBridge. New Year Rollover is determined by a date range called **Pause Period**. The Pause Period provides a customizable time frame during which the district can make data changes in their Student Information System in preparation for the new school year without impacting teacher and student access to the current school year's section and online curriculum.

It is imperative to populate appropriate dates in the following files when running NYR in EasyBridge on the Pause Period end date:

OneRoster file format (.csv)	Legacy Text file format (.txt)
academicSessions.csv For the school year 2020-2021, 'schoolYear' field would be populated with '2021'	PIF_SECTION.txt For the school year 2020-2021, 'school_year' would be populated with '2020'
enrollments.csv "beginDate" and "endDate" must be aligned with the current school year defined in academicSessions	PIF_SECTION_STAFF.txt For the school year 2020-2021, 'school_year' would be populated with '2020'
	PIF_SECTION_STUDENT.txt For the school year 2020-2021, 'school_year' would be populated with '2020'

Please see the [Pause Period for New Year Rollover \(NYR\) Guide](#) on our support site for detailed information and sample date ranges.

Appendix A – OneRoster v1.1 (.csv) Specifications

The following chart lists all 14 OneRoster files and identifies which are required by OneRoster and/or EasyBridge. See the [IMS Global document](#) for more information.

FILE NAME	REQUIRED BY ONEROSTER	REQUIRED BY EASYBRIDGE
manifest.csv	YES	YES
academicSessions.csv	No	YES
categories.csv	No	No
classResources.csv	No	No *
classes.csv	No	YES
courseResources.csv	No	No *
courses.csv	No	YES
demographics.csv	No	No
enrollments.csv	No	YES
lineItems.csv	No	No
orgs.csv	No	YES
results.csv	No	No
resources.csv	No	No *
users.csv	No	YES
* Required for product association to sections. See “Sync Modes” for more information.		

manifest.csv

The manifest file is a static file that indicates which files have been included in a data upload. Every file from the OneRoster 1.1 format must be in the list and will be marked with their processing mode or “absent”. EasyBridge requires bulk mode processing. Additional files can be included in the upload but will be ignored by EasyBridge.

Example manifest.csv:

propertyName	value
manifest.version	1.0
oneroster.version	1.1
file.academicSessions	bulk
file.categories	absent
file.classes	bulk
file.classResources	absent
file.courses	bulk
file.courseResources	absent
file.demographics	absent
file.enrollments	bulk
file.lineItems	absent
file.orgs	bulk
file.resources	absent
file.results	absent
file.users	bulk

*If uploading product associations, change file.classResources, file.courseResources, and file.resources to value = bulk.

academicSessions.csv

FIELD NAME	REQUIRED BY ONEROSTER 1.1	REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
sourcedId	YES	YES		Unique Id for the term records. This sourcedId referenced by other files.
status	No	No		Must not contain data for bulk mode
dateLastModified	No	No		Must not contain data for bulk mode
title	YES	YES		Name/title of the academic session
type	YES	YES		Limited to values term, gradingPeriod, schoolyear, semester
startDate	YES	YES	YYYY-MM-DD	Inclusive start date. The academic session DOES include the startDate.
endDate	YES	YES	YYYY-MM-DD	Exclusive end date. The academic session does NOT include the endDate.
parentSourcedId	No	No		
schoolYear	YES	YES	YYYY	The current school year and is defined as the year the school year ends. Example: For 2020-2021 school year, schoolYear = 2021.

Example header row:

sourcedId,status,dateLastModified,title,type,startDate,endDate,parentSourcedId,schoolYear

classes.csv

FIELD NAME	REQUIRED BY ONEROSTER 1.1	REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
sourcedId	YES	YES		Unique Id for the class records. This sourcedId referenced by other files.
status	No	No		Must be blank for bulk mode processing
dateLastModified	No	No		Must be blank for bulk mode processing
title	YES	YES		Title will be displayed in the EasyBridge portal as the section name. Titles should be unique for a given teacher.
Grades	No	Recommended	Use CEDS grade codes	
courseSourcedId	YES	YES		sourcedId from courses.csv of which this class is an instance.
classCode	No	Recommended		Used to help identify the class.
classType	YES	YES	homeroom scheduled	
location	No	No		Description of where the class is physically located.
schoolSourcedId	YES	YES		sourcedId from orgs.csv that teaches this class of OrgType "school"

termSourcedIds	YES	YES		sourcedId from academicSessions.csv in which the class is taught.
subjects	No	Recommended		Subject name
subjectCodes	No	No		
periods	No	No		

Example header row:

sourcedId,status,dateLastModified,title,grades,courseSourcedId,classCode,classType,location,
schoolSourcedId,termSourcedIds,subjects,subjectCodes,periods

courses.csv

FIELD NAME	REQUIRED BY ONEROSTER 1.1	REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
sourcedId	YES	YES		Unique Id for the course record
status	No	No		Must be blank for bulk mode processing
dateLastModified	No	No		Must be blank for bulk mode processing
schoolYearSourcedId	No	No		EasyBridge uses the academic session that is referenced in the classes file.
title	YES	YES		Name of the course.
courseCode	No	Recommended		
grades	No	No		EasyBridge uses the class grade.
orgSourcedId	YES	YES		sourcedId from orgs.csv to which this course belongs.
subjects	No	No		EasyBridge uses the class subject.
subjectCodes	No	No		

Example header row:

sourcedId,status,dateLastModified,schoolYearSourcedId,title,courseCode,grades,OrgSourcedId,subjects,subjectCodes

enrollments.csv

FIELD NAME	REQUIRED BY ONEROSTER 1.1	REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
sourcedId	YES	YES		Unique Id for the enrollment record.
status	No	No		Must be blank for bulk mode processing
dateLastModified	No	No		Must be blank for bulk mode processing
classSourcedId	YES	YES		sourcedId from classes.csv
schoolSourcedId	YES	YES		sourcedId from orgs.csv
userSourcedId	YES	YES		sourcedId from users.csv
role	YES	YES	teacher student	
primary	No	Recommended	Boolean lowercase	true = lead teacher false = co-teacher
beginDate	No	Recommended	YYYY-MM-DD	<i>Inclusive</i> date. Teachers and students are enrolled on the beginDate.
endDate	No	Recommended	YYYY-MM-DD	<i>Exclusive</i> date. Teachers and students are NOT enrolled on the endDate.

Example header row:

sourcedId,status,dateLastModified,classSourcedId,schoolSourcedId,userSourcedId,role,primary,beginDate,endDate

orgs.csv

FIELD NAME	REQUIRED BY ONEROSTER 1.1	REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
sourcedId	YES	YES		Unique Id for the organization. sourcedId is referenced by other files.
status	No	No		Must be blank for bulk mode processing
dateLastModified	No	No		Must be blank for bulk mode processing
name	YES	YES		Name of the organization
type	YES	YES	school district	School records required. District record recommended
identifier	No	No		
parentSourcedId	No	No		
metadata.address1	No	Recommended		
metadata.address2	No	Recommended		
metadata.city	No	Recommended		
metadata.state	No	Recommended		
metadata.postCode	No	Recommended		

Example header row:

sourcedId,status,dateLastModified,name,type,identifier,parentSourcedId,metadata.address1,metadata.address2,metadata.city,metadata.state,metadata.postCode

users.csv

FIELD NAME	REQUIRED BY ONEROSTER 1.1	REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
sourcedId	YES	YES		Unique Id for the user. sourcedId is referenced by other files
status	No	No		Must be blank for bulk mode processing
dateLastModified	No	No		Must be blank for bulk mode processing
enabledUser	YES	YES	true false	Boolean value
orgSourcedIds	YES	YES		Defined by the sourcedIds value in orgs.csv. Non-teaching staff must be assigned to at least one school(*)
role	YES	YES	teacher student administrator	EasyBridge will ignore records with any role other than teacher, student, or administrator
username	YES	YES		Required. Used to create district managed or initial secondary credential usernames
userIds	No	YES	{Fed:Id}	User's federated id value. Must match authenticationId value in IdP.
givenName	YES	YES		First name
familyName	YES	YES		Last name

FIELD NAME	REQUIRED BY ONEROSTER 1.1	REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
middleName	No	Recommended		
identifier	No	Recommended		
email	No	Recommended		Highly recommended for all users. Can be used as Fed id with Google or other IdP.
sms	No	No		
phone	No	No		
agentSourcedIds	No	No		
grades	No	Recommended		Required for SuccessMaker integration with EasyBridge
password	No	No		

(*) Non-teaching staff assigned to more than one school – comma delimit multiple orgSourcedId values and wrap in double quotes.

Example header row:

sourcedId,status,dateLastModified,enabledUser,orgSourcedIds,role,username,userIds,givenName,
familyName,middleName,identifier,email,sms,phone,agentSourcedIds,grades,password

classResources.csv (OPTIONAL: used for product association)

FIELD HEADER	REQUIRED BY ONEROSTER	REQUIRED BY EASYBRIDGE	NOTES
sourcedId	YES	YES	The unique Id for the record
status	No	No	Must be blank for bulk mode processing
dateLastModified	No	No	Must be blank for bulk mode processing
title	No	No	
classSourcedId	YES	YES	The sourcedId defined by classes.csv
resourceSourcedId	YES	YES	The sourcedId defined by resources.csv

Example header row: sourcedId,status,dateLastModified,title,classSourcedId,resourceSourcedId

courseResources.csv (OPTIONAL: used for product association)

FIELD HEADER	REQUIRED BY ONEROSTER	REQUIRED BY EASYBRIDGE	NOTES
sourcedId	YES	YES	The unique id for the record
status	No	No	Must be blank for bulk mode processing
dateLastModified	No	No	Must be blank for bulk mode processing
title	No	No	
courseSourcedId	YES	YES	The unique sourcedId used in courses.csv
resourceSourcedId	YES	YES	The sourcedId from resources.csv

Example header row: sourcedId,status,dateLastModified,title,courseSourcedId,resourceSourcedId

resources.csv (OPTIONAL: used for product association)

FIELD HEADER	REQUIRED BY ONEROSTER	E REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
sourcedId	YES	YES		The unique Id for the record
status	No	No		Must be blank for bulk mode processing
dateLastModified	No	No		Must be blank for bulk mode processing
vendorResourceId	YES	YES		The product id for the resource provided by Solution Engineer.
title	No	YES		The name of the resource provided by Solution Engineer. Include the name for troubleshooting purposes.
roles	No	No		EasyBridge does not differentiate product ids between teachers & students
importance	No	No		
vendorId	No	YES	savvas	
applicationId	No	YES	easybridge	

Example header row:

sourcedId,status,dateLastModified,vendorResourceId,title,Roles,Importance,vendorId,applicationId

Appendix B – OneRoster 1.1 & Product Associations

Administrators can complete product-to-class section associations for their teachers using one of two methods:

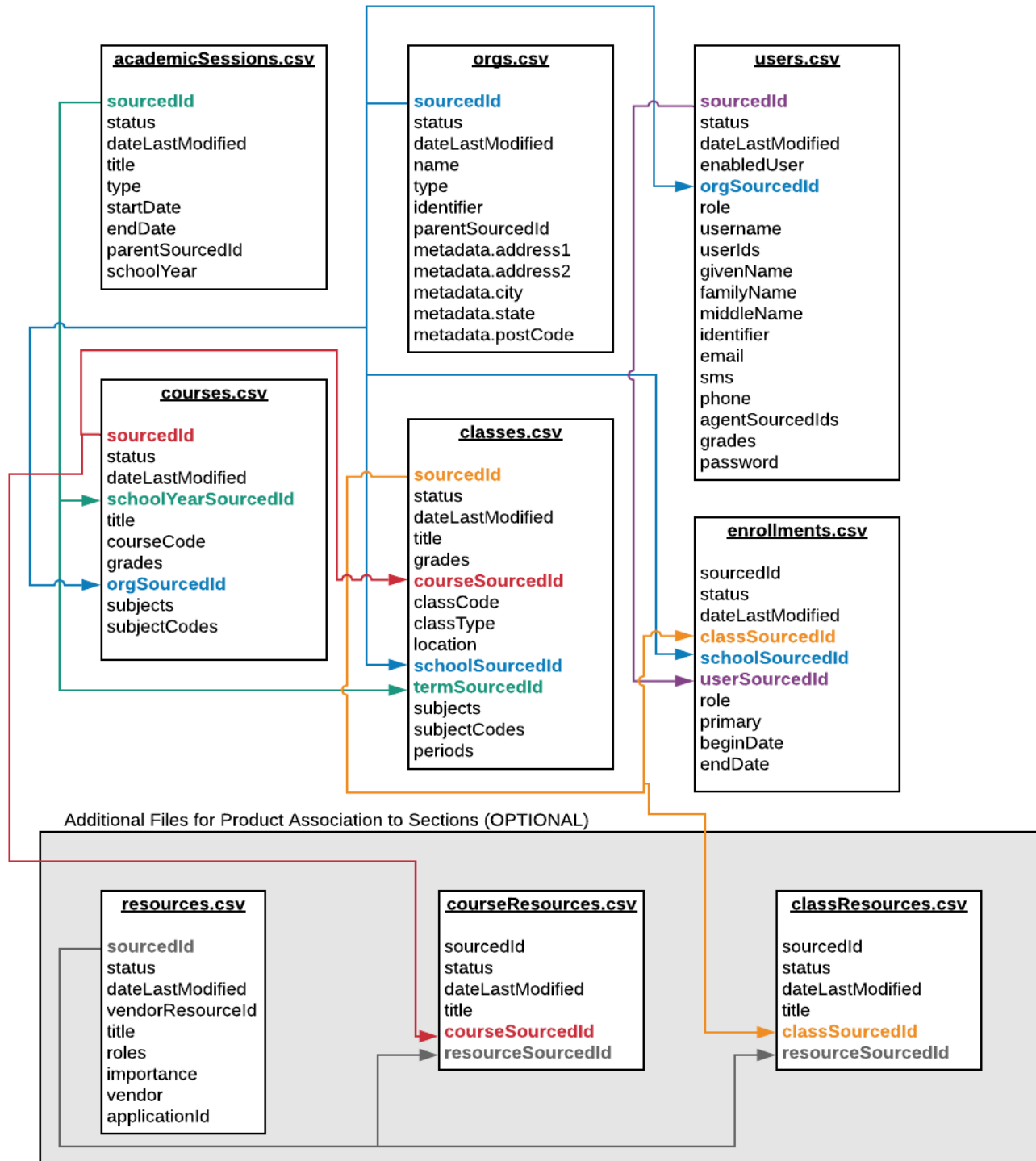
- The EasyBridge Admin portal to add classes to products
- Automated product-to-class associations using OneRoster v 1.1 resource mapping files (see “Sync Modes for Use with OneRoster version 1.1” below)

Sync Modes for Use with OneRoster version 1.1

Districts using OneRoster version 1.1 have many options when associating products through the ingestion files. The table on the following page describes the details for each sync mode and the options available. Configuration of sync modes requires the help of an EasyBridge Solution Engineer. Be sure to speak with your Solutions Engineer before Go Live to select the mode and option combination that works best for your district.

MODE	OPTION 1	OPTION 2
STRICT	<ul style="list-style-type: none"> Products added via the ingestion files can only be removed via the import files. Products cannot be removed manually by administrators or teachers through the EasyBridge teacher or admin portals. 	
	<p>Teachers CANNOT add products</p> <ul style="list-style-type: none"> Administrators and teachers cannot remove products that were added by the import files. Products can only be removed via the import files. Administrators can add and remove additional products. Teachers only see classes that have products associated to them. 	<p>Teachers CAN add products</p> <ul style="list-style-type: none"> Administrators and teachers cannot remove products that were added by the import files. Products can only be removed via the import files. Administrators and teachers can add additional products (and remove those same products) through the EasyBridge teacher or admin portals.
NON-STRICT	<ul style="list-style-type: none"> Products associated through the import files are added to classes. Products can be removed from the import files but will not be removed from classes. Administrators and teachers can remove products through the EasyBridge teacher or admin portals. 	
	<p>Teachers CANNOT add/remove products</p> <ul style="list-style-type: none"> Products can only be added by the import process. Products cannot be removed using the import files. Administrators must manually remove products through the EasyBridge admin portal. Administrators can add and remove any product through the EasyBridge admin portal. Teachers cannot add or remove products through the EasyBridge teacher portal. Teachers only see classes that have products associated to them. 	<p>Teachers CAN add/remove products</p> <ul style="list-style-type: none"> Products are added by the import process but cannot be removed by the import process. Administrators must manually remove products through the EasyBridge admin portal. Administrators and teachers can add and remove products through the EasyBridge teacher or admin portals. If a product is removed through EasyBridge that is also in the data import, the product will be added back the next time the roster is updated.

Appendix C – OneRoster v1.1 Data Relationships



Appendix D – Legacy Text File Format (.txt) Specifications

CODE_DISTRICT.txt

FIELD NAME	REQUIRED BY EASYBRIDGE	FORMAT	NOTES
district_code	YES	case sensitive	An immutable code that uniquely identifies the district/LEA
district_name	YES		Does not have to match licensed org name
address_1	Recommended		Not required but highly recommended
address_2	No		Can be used for PO Boxes or other address info
city	Recommended		Not required but highly recommended
state	Recommended		Not required but highly recommended
zip	Recommended		Not required but highly recommended
phone	No		

Example header row:

district_code,district_name,address_1,address_2,city,state,zip,phone

SCHOOL.txt

FIELD NAME	REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
school_code	YES	Case sensitive	An immutable code that uniquely identifies a school
school_name	YES		Does not have to match licensed org names
district_code	YES		The immutable code that uniquely identifies the school district/LEA. (Defined in CODE_DISTRICT.txt file).
grade_start	No		
grade_end	No		
address_1	Recommended		Not required but highly recommended
address_2	No		
city	Recommended		Not required but highly recommended
state	Recommended		Not required but highly recommended
zip	Recommended		Not required but highly recommended
phone	No		

Example header row:

school_code,school_name,district_code,grade_start,grade_end,address_1,address_2,city,state,zip,
phone

STAFF.txt

FIELD NAME	REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
staff_code	YES	Case sensitive	An immutable code that uniquely identifies a staff user
last_name	YES		
first_name	YES		
middle_name	No		
email	Recommended		Highly recommended for users
title	No		Used to add descriptive values like Dr., Mrs., or Mr.
staff_number	YES		Typically, staff_code is mapped to an internal id. This field can be populated with staff_code value or a more recognizable user value from the SIS.
federated_id	YES		Federated id = authenticationId in IdP. Must be unique across staff and students.

Example header row:

staff_code,last_name,first_name,middle_name,email,title,staff_number,federated_id

STUDENT.txt

FIELD NAME	REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
student_code	YES	Case sensitive	An immutable code that uniquely identifies a student
last_name	YES		
first_name	YES		
middle_name	No		
gender_code	No		
dob	No	YYYY-MM-DD	
email	Recommended		Highly recommended for users
student_number	YES		Typically, student_code is mapped to an internal id. This field can be populated with student_code or a more recognizable user value from the SIS.
federated_id	YES		Federated id = authenticationId in IdP. Must be unique across staff and students.

Example header row:

student_code,last_name,first_name,middle_name,gender_code,dob,email,student_number,federated_id

PIF_SECTION.txt

FIELD NAME	REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
native_section_code	YES	Case sensitive	An immutable code that uniquely identifies a section
school_code	YES		The immutable code that uniquely identifies a school (defined in SCHOOL.txt file)
section_type	Optional		A code indicating the duration of a section (e.g. Quarter, Trimester, Semester, Year)
section_type_description	Optional		A brief description for the section duration (e.g. Semester 1, Quarter 3)
date_start	YES	YYYY-MM-DD	The starting date for the section
date_end	YES	YYYY-MM-DD	The ending date, inclusive, for the section
school_year	YES	YYYY	The school year for the section identified by the starting calendar year (i.e. 2019 for the 2019-2020 school year)
course_number	Optional		A number (or code) that identifies the course material being taught during the section
course_name	Optional		A name for the course
section_name	YES		A meaningful name to identify the instance of the course; for example, Algebra 10 - Period 1 (S1)
section_number	YES		A meaningful unique identifier for the section. Example: ALG10-01

FIELD NAME	REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
subjects	Optional		A list of subjects the section belongs to. The value should be a single entry, or a comma-separated list enclosed in double quotes.
grades	Optional	Use CEDS grade codes	A comma separated list of grades the section serves. The value should be a single entry, or a comma-separated list enclosed in double quotes.
class_type	Optional	homeroom scheduled	

Example header row:

native_section_code,school_code,section_type,section_type_description,date_start,date_end,
school_year,course_number,course_name,section_name,section_number

PIF_SECTION_STAFF.txt

FIELD NAME	REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
section_teacher_code	YES	Case sensitive	An immutable code that uniquely identifies a section/teaching assignment
staff_code	YES		The immutable code that uniquely identifies a staff member (defined in STAFF.txt file)
native_section_code	YES		The immutable code that uniquely identifies a section (defined in PIF_SECTION.txt file)
date_start	YES	YYYY-MM-DD	The starting date for the teacher's assignment to the section
date_end	YES	YYYY-MM-DD	The ending date, inclusive, for the teacher's assignment to the section
school_year	YES	YYYY	The school year for the section identified by the starting calendar year (i.e. 2019 for the 2019-2020 school year)
teacher_of_record	YES		Indicates lead(true) versus co-teacher(false) of section. Should be only one lead teacher of record for a section.
teaching_assignment	Optional		An indication of the teacher's role for the section (e.g. Teacher, Lead Teacher, Co-teacher, Teacher's Aide)

Example header row:

section_teacher_code,staff_code,native_section_code,date_start,date_end,school_year,
teacher_of_record,teaching_assignment

PIF_SECTION_STUDENT.txt

FIELD NAME	REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
section_student_code	YES	Case sensitive	An immutable code that uniquely identifies a section/student assignment
student_code	YES		The immutable code that uniquely identifies a student (defined in STUDENT.txt file)
native_section_code	YES		The immutable code that uniquely identifies a section record (defined in PIF_SECTION.txt file)
date_start	YES	YYYY-MM-DD	The starting date for the student's assignment to the section
date_end	YES	YYYY-MM-DD	The ending date, inclusive, for the student's assignment to the section. This is typically the same value as PIF_SECTION.date_end.
school_year	YES	YYYY	The school year for the section identified by the starting calendar year (i.e. 2019 for the 2019-2020 school year)

Example header row:

section_student_code,student_code,native_section_code,date_start,date_end,school_year

ENROLLMENT.txt

FIELD NAME	REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
native_enrollment_code	YES		An immutable code that uniquely identifies a student enrollment in a defined school
student_code	YES		The immutable code that uniquely identifies a student (defined in STUDENT.txt file)
school_year	Recommended	YYYY	The school year for the section identified by the starting calendar year (i.e. 2019 for the 2019-2020 school year)
school_code	YES		The immutable code that uniquely identifies a school record (defined as school_code in SCHOOL.txt) (*)
grade_code	YES	Use CEDS grade codes	The value should be a single entry, or a comma separated list enclosed in double quotes. Required to add SuccessMaker into an EasyBridge integration
date_enrolled	YES	YYYY-MM-DD	The starting date for the student's school enrollment for the current school year
date_withdrawn	Optional	YYYY-MM-DD	The end date for the student's school enrollment for the current school year

(*)students with more than one school assignment - upload multiple records in this file. One record per location per student person.

Example header row:

native_enrollment_code,student_code,school_year,school_code,grade_code,date_enrolled,
date_withdrawn

ASSIGNMENT.txt

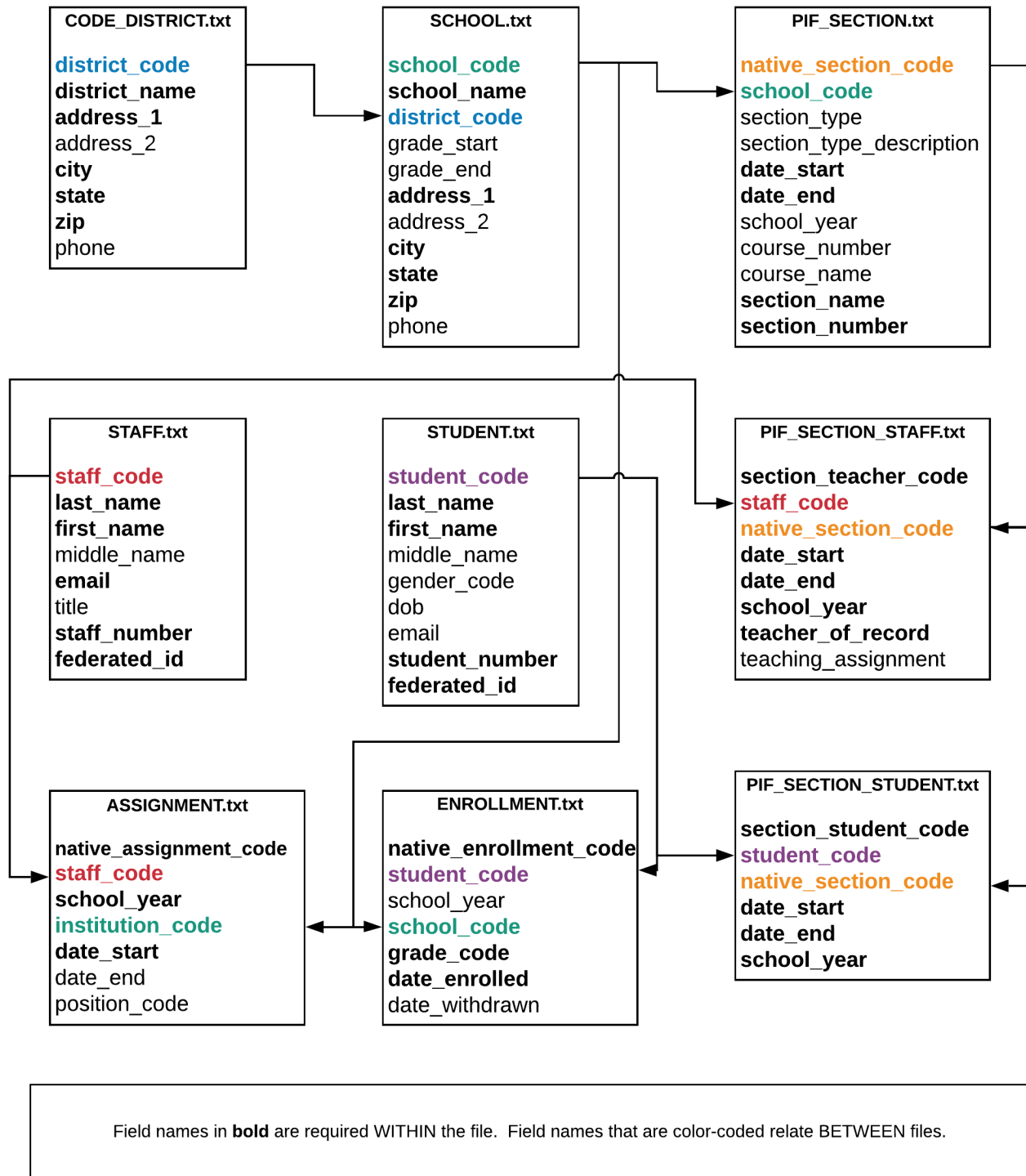
FIELD NAME	REQUIRED BY EASYBRIDGE	REQUIRED FORMAT	NOTES
native_assignment_code	YES	Case sensitive	An immutable code that uniquely identifies a staff member assignment to a school (staff_code + school_code)
staff_code	YES		The immutable code that uniquely identifies a staff member (defined in STAFF.txt file)
school_year	YES	YYYY	The school year for the assignment identified by the starting calendar year (i.e. 2019 for the 2019-2020 school year)
institution_code	YES		The immutable code that uniquely identifies a school record (defined as school_code in SCHOOL.txt) (*)
date_start	YES	YYYY-MM-DD	The starting date for the staff/school assignment
date_end	Optional	YYYY-MM-DD	The ending date, inclusive, for the staff/school assignment (empty indicates an assignment that has not ended)
grades	Optional	Use CEDS grade codes	
position_code	Optional	Teacher Staff	

(*)non-teaching staff with more than one school assignment - upload multiple records in this file. One record per location per student person.

Example header row:

native_assignment_code,staff_code,school_year,institution_code,date_start,date_end,position_code

Appendix E – Legacy Text File Format (.txt) Data Relationships



Appendix F – Expected Data Behaviour

The table below outlines some of the more common data changes in an SIS and how they should be properly represented in the data extract files.

DATA CHANGE	CHANGE IN DATASET		EXPECTED BEHAVIOR	
	ONEROSTER	LEGACY TEXT	ONEROSTER	LEGACY TEXT
SCHOOL				
New school added	New record created in orgs.csv	New record created in SCHOOL.txt	User logins for the new school will not work until manual configuration completed in EdCloud NOTE: Must contact technical support	
School information updated	Updated record in orgs.csv	Updated record in SCHOOL.txt	If no change is made to the sourcedId field value, the existing school record is updated District does not need to notify Technical Support	If no change is made to the school_code field value, the existing school record is updated District does not need to notify Technical Support
School closed and removed from SIS	Record removed from orgs.csv Related records removed from courses.csv, classes.csv, enrollments.csv, and users.csv	Record removed from SCHOOL.txt Related records removed from PIF_SECTION.txt and ASSIGNMENT.txt	User accounts associated with deleted school will no longer work Licensing associated with a deleted school will no longer be available Contact your AGM to move licensing to other active school(s)	

DATA CHANGE	CHANGE IN DATASET		EXPECTED BEHAVIOR	
	ONEROSTER	LEGACY TEXT	ONEROSTER	LEGACY TEXT
COURSE SECTION (Class)				
New course section created	New record created in classes.csv and courses.csv	New record created in PIF_SECTION.txt	Once the ingestion of the upload finishes, new section records will take up to 4 hours to appear in the EasyBridge teacher and student portals	
Course section updated		Updated record in PIF_SECTION.txt file	If sourcedId value is not changed, the original section record is updated If a teacher leaves and a new teacher is going to cover the class section, no student progress data will be lost so long as the sourcedId for the section record remains the same	If native_section_code value is not changed, the original section record is updated If a teacher leaves and a new teacher is going to cover the class section, no student progress data will be lost so long as the native_section_code for the section record remains the same
Course section deleted		Record removed from PIF_SECTION.txt Related records removed from PIF_SECTION_STAFF.txt and PIF_SECTION_STUDENT.txt	Section records removed from files are deleted from EasyBridge All related student progress data will be deleted	
STAFF (Teacher)				
New teacher created	New record created in users.csv	New record created in STAFF.txt	New teacher accounts start off blank in the new integrated EasyBridge environment New staff account holder will be able to log into EasyBridge and view sections after 4-hour sync completes	

DATA CHANGE	CHANGE IN DATASET		EXPECTED BEHAVIOR	
	ONEROSTER	LEGACY TEXT	ONEROSTER	LEGACY TEXT
Teacher updated	Updated record in users.csv file	Updated record in STAFF.txt file	<p>If no changes are made to sourcedId value, the existing teacher record is updated</p> <p>When changing the federated_id, userIds, username or password fields, the user can re-try login as soon as ingestion completes in EdCloud</p>	<p>If no changes are made to staff_code value, the existing teacher record is updated</p> <p>When changing the federated_id, userIds, username or password fields, the user can re-try login as soon as ingestion completes in EdCloud</p>
Teacher deleted	<p>Record removed from users.csv</p> <p>Related records removed from enrollments.csv</p>	<p>Record removed from STAFF.txt</p> <p>Related records removed from PIF_SECTION_STAFF.txt and ASSIGNMENT.txt</p>	<p>Custom content associated with deleted teacher accounts will be removed</p> <p>If a teacher leaves and a new teacher is going to cover the class section, no student progress data will be lost so long as the sourcedId for the section record remains the same (see course section changes above)</p>	<p>Custom content associated with deleted teacher accounts will be removed</p> <p>If a teacher leaves and a new teacher is going to cover the class section, no student progress data will be lost so long as the native_section_code for the section record remains the same (see course section changes above)</p>
Teacher assigned to a class	New record created in enrollments.csv	New record created in PIF_SECTION_STAFF.txt	Each class is associated with a school. By assigning a teacher to a class you are associating that teacher with the school that class belongs to.	
Teacher unassigned from a class	Update the endDate record in enrollments.csv	Updated record in PIF_SECTION_STAFF.txt with	When uploading files, the teacher is NOT considered to be	When uploading files, the teacher is considered to be

DATA CHANGE	CHANGE IN DATASET		EXPECTED BEHAVIOR	
	ONEROSTER	LEGACY TEXT	ONEROSTER	LEGACY TEXT
	with the date for the day after the teacher's last day assigned to the class	"date_end" set to teacher's last effective day assigned to class	assigned to the class on the "endDate" date	assigned to the class on the "date_end" date
Teacher is a co-teacher of the class section		Record included in PIF_SECTION_STAFF.txt has the "teacher_of_record" value set to "false"	Only one lead teacher of record is allowed per section A district may have a maximum of 24 co-teachers associated to one section	
Non-Teaching STAFF ("Curriculum Specialist" User Account Holder)				
New non-teacher created	New record created in users.csv	New records created in STAFF.txt and ASSIGNMENT.txt	Non-teaching staff added to the data file upload will be able to log into EasyBridge, launch a learning platform and view curriculum content only	
Non-Teacher updated	Updated record in users.csv	Updated record in STAFF.txt And ASSIGNMENT.txt	If no changes are made to sourcedId value, the existing record is updated When changing the userIds, username or password fields, the user can re-try login as soon as ingestion completes in EdCloud	If no changes are made to staff_code value, the existing record is updated When changing the federated_id, username or password fields, the user can re-try login as soon as ingestion completes in EdCloud
Non-Teacher deleted	Record removed from users.csv	Record removed from STAFF.txt And ASSIGNMENT.txt	Custom content associated with the user account will be removed	

DATA CHANGE	CHANGE IN DATASET		EXPECTED BEHAVIOR	
	ONEROSTER	LEGACY TEXT	ONEROSTER	LEGACY TEXT
Non-Teacher assigned to a new or additional school	New record created in users.csv file or existing record modified	New record created in ASSIGNMENT.txt or existing record modified	The users.csv file is required to associate a user to a school A user can be associated with a maximum of 20 different schools	The ASSIGNMENT.txt file is required to associate a user to a school A user can be associated with a maximum of 20 different schools
Non-Teacher unassigned from a school	Updated "orgSourcedId" values on the record in users.csv file	Updated record in ASSIGNMENT.txt with "date_end" set to last effective day of user's school assignment	The non-teacher is no longer associated to the school on the date the orgSourcedId values are modified	The non-teacher is considered to be assigned to the school on the "date_end" date
STUDENT				
Student created	New record added to users.csv	New record added to STUDENT.txt		Student user will be added in EasyBridge but will not be able to log in until sections are uploaded for the user in the PIF_SECTION_STUDENT.txt file
Student updated	Updated record in users.csv	Updated record in STUDENT.txt	If the sourcedId value is not modified, then the existing record will be updated in EasyBridge Student user will be added in EasyBridge but will not be able to log in until sections are uploaded for the	If the student_code value is not modified, then the existing record will be updated in EasyBridge

DATA CHANGE	CHANGE IN DATASET		EXPECTED BEHAVIOR	
	ONEROSTER	LEGACY TEXT	ONEROSTER	LEGACY TEXT
			user in the enrollments.csv file When changing the userIds, username or password fields, the user can re-try login as soon as ingestion completes in EdCloud	When changing the federated_id, username or password fields, the user can re-try login as soon as ingestion completes in EdCloud
Student deleted	Record removed from users.csv and related records removed from enrollments.csv	Record removed from STUDENT.txt file and related records removed from PIF_SECTION_STUDENT.txt	Student progress data related to the student record will be deleted If a student withdraws but then returns to a district within 60 days, it is possible to restore the original user account by adding it back into the upload	
Student enrolled in a class	New record created in enrollments.csv	New record created in PIF_SECTION_STUDENT.txt	It will take up to 4 hours for the student to appear in the section roster in EasyBridge Students will only see sections in EasyBridge that a teacher has added a product	
Student unenrolled from a class	Record in enrollments.csv updated with "endDate" set to the day after the last effective date of student's enrollment in class section.	Record in PIF_SECTION_STUDENT.txt updated with "date_end" set to last effective date of student's enrollment in class section.	When uploading files, the student is NOT considered enrolled in the class on the "endDate" date	When uploading files, the student is considered to still be enrolled in the class on the "date_end" date
Student transferred from one	Record for class being dropped the "endDate" value set to the day after	Record for class being dropped the "date_end" value set to last effective date of student's enrollment in class in	When files, the student is considered enrolled in the class section from the	When uploading files, the student is considered to be enrolled in the class on the "date_start" date through

DATA CHANGE	CHANGE IN DATASET		EXPECTED BEHAVIOR	
	ONEROSTER	LEGACY TEXT	ONEROSTER	LEGACY TEXT
class to another	<p>the last effective date of student's enrollment in class in enrollments.csv</p> <p>Record for class being added, record created (or updated) in enrollments.csv with "startDate" set to the same date value as the dropped class section "endDate"</p>	<p>PIF_SECTION_STUDENT.txt</p> <p>Record for class being added, record created (or updated) in PIF_SECTION_STUDENT.txt with "date_start" set to the day after dropped class section "date_end"</p>	<p>"startDate" through but NOT including the "endDate" date</p> <p>If a student was dropped from a section erroneously, the original section record can be added back into the enrollments.csv file to restore the original section and progress data for the student (restore must be done within 60 days of dropping the original class section)</p>	<p>and including the "date_end" date</p> <p>If a student was dropped from a section erroneously, the original section record can be added back into the PIF_SECTION_STUDENT.txt file to restore the original section and progress data for the student (restore must be done within 60 days of dropping the original class section)</p>

Appendix G – Types of User Accounts

Student - Access to learning platform content that has been added to one or more classes in which the student is enrolled

Classroom Teacher (“Lead Teacher”) - Access to classes and rosters in EasyBridge and to learning platform content

Co-Teacher - Same access and functionality as the Classroom Teacher (lead teacher) user

Curriculum Specialist - No access to classes or rosters in EasyBridge. Access to learning platform content only for the content licensed at their school(s) or the district level. Cannot be assigned to more than twenty schools

USER TYPE	ONEROSTER (.csv)
STUDENT	[users.csv]role = "student" [enrollments.csv]role="student"
CLASSROOM TEACHER (lead teacher)	[users.csv]role = "teacher" [enrollments.csv]role="teacher" [enrollments.csv]primary="true"
CO-TEACHER	[users.csv]role = "teacher" [enrollments.csv]role="teacher" [enrollments.csv]primary="false"
CURRICULUM SPECIALIST	[users.csv]role = "teacher", "administrator" or "aide" [users.csv]orgSourcedIds sourcedId from orgs.csv (maximum of 20)
USER TYPE	LEGACY TEXT FILE FORMAT (.txt)
STUDENT	Must have a record in STUDENT.txt Must have a record in PIF_SECTION_STUDENT.txt
CLASSROOM TEACHER (lead teacher)	Must have a record in STAFF.txt Must have a record in PIF_SECTION_STAFF.txt [PIF_SECTION_STAFF.txt]teacher_of_record="true"
CO-TEACHER	Must have a record in STAFF.txt Must have a record in PIF_SECTION_STAFF.txt [PIF_SECTION_STAFF.txt]teacher_of_record="false"
CURRICULUM SPECIALIST	Must have a record in STAFF.txt Must have a record in ASSIGNMENT.txt [ASSIGNMENT.txt]institution_code must contain at least one school_code from SCHOOL.txt (maximum of 20)

Appendix H – EasyBridge Administrator Portal

Access to the EasyBridge Administrator account is only available via the EasyBridge Administrator login page: savvaseasybridge.com

EasyBridge Administrator accounts will not allow access via the district specific EasyBridge Plus login URL that staff, and students, use to log into EasyBridge Plus.

- Access to the EasyBridge Administrator portal requires administrator credentials
- Initial administrator credentials will be issued to the District's EasyBridge implementation team by the EasyBridge Solutions Engineer following the Orientation meeting
- Administrator credentials *cannot* be created or managed via a district's OneRoster or Legacy Text data file uploads
- Additional administrator credentials can be created in the EasyBridge Administrator portal by any district staff member with active EB Admin access
 - See this **Manage Administrators via EasyBridge** section of the [How to Manage Administrator Accounts](#) guide