Sopheon Accolade®

Process Design - Importing Data Training Guide

Version: 15.3

About Sopheon Accolade®

Document Name:	Process Design - Importing Data Training Guide
Document Version:	1
Software Version:	Sopheon Accolade 15.3
Document Date:	May 2023

Ownership of Software and Documentation

The Sopheon® software described in this documentation is furnished under a license agreement and may be used only in accordance with the terms of that license agreement.

Sopheon Corporation and its associated Sopheon Group companies, including its subsidiaries, its immediate holding company and its ultimate holding company (together, "Sopheon") have created and own all rights to the software and documentation. Licensees of the software have purchased a limited right to use the software in accordance with their license agreement.

Copyright Notice

Trademarks

No Warranty

The technical documentation is being delivered to you AS-IS, and Sopheon makes no warranty as to its accuracy or use. Any use of the technical documentation or the information contained therein is at the risk of the user. Documentation may include technical or other inaccuracies or typographical errors. Sopheon reserves the right to make changes without prior notice. In no circumstances will Sopheon, its agents or employees be liable for any special, consequential or indirect loss or damage arising from any use of or reliance on any materials in this documentation or in the software.

Patents

Aspects of Sopheon software are protected by U.S. Patents 5634051, 6632251, and 6526404; European Patent EP0914637; and by U.K. Patent GB2341252A.

Contents

About the Accolade Education Program

This module is part of the Sopheon Accolade Education Program (AEP). The AEP modules are designed to help Accolade users perform the tasks in their company's business process using the Accolade application. The content in the modules is meant to be used side-by-side with the application, and is part of the overall documentation suite provided for Accolade.

The benefits of using Accolade as part of your company's innovation development process include the following:

- Reduced cycle time by displaying clear structure and visibility.
- Reduced rework through timely, properly sequenced completion of all key tasks and milestones.
- Assured positive user experience through properly developed product requirements.
- Improved communication by automating collaboration between multifunctional team members.
- Provided decision-making information. Poor projects are stopped or placed on hold so resources can be redirected to more promising and higher value projects and products.
- Provided clear project requirements. Expectations of a project team and project manager at each stage are clearly spelled out.
- Managed business risk. Break resource commitments into increments or stages.
- Established key baseline information and metrics.

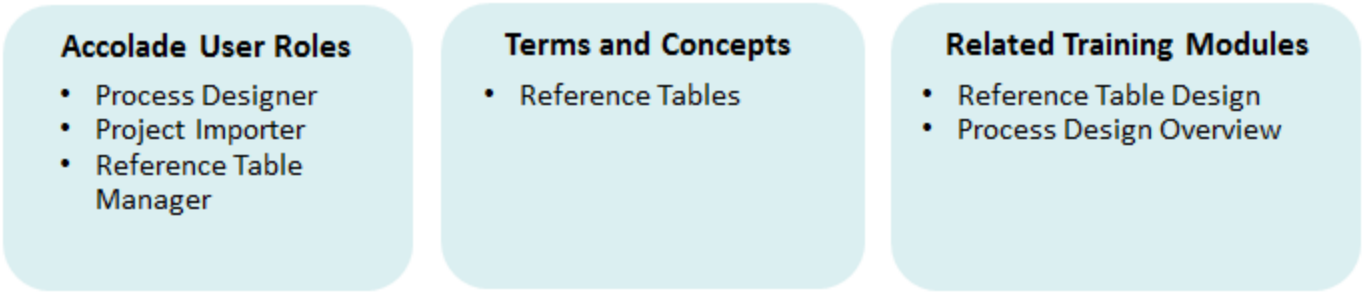
The Accolade documentation suite contains the following additional components:

Document	Contents
Sopheon Accolade What's New in This Release	For each release, review this document for an overview of the new features and changes within the release.
Accolade Online Help	Accessible directly through Accolade, the online Help provides comprehensive how-to and reference information about all aspects of using Accolade.
Sopheon Accolade Administrator's Guide	Provides information for administrative professionals regarding Accolade setup. This information is also provided in the online Help.
Sopheon Accolade Installation Guide	Provides information about the installation of the application and its required databases.
Dashboards for Accolade Installation Guide	Provides installation information for installing the Dashboards for Accolade component.
Quick Reference Cards	A PDF that can be printed double-sided that provides quick tips and navigation information for using Accolade.

Document	Contents
Online Help for Accolade Add-ins	Accolade add-ins, including Accolade Office Extensions, Accolade SmartDocuments for Google, Accolade SmartDocuments for Office, Accolade Portfolio Optimizer, and Accolade's integration with Microsoft Project, each include their own Sopheon created Help file accessible directly from the application after the add-in is installed. Each Help file describes how to use the features of that particular add-in.

Prerequisites for Using this Module

The contents of this training module assumes you are assigned the Accolade user roles and have a basic understanding of the terms and concepts listed below and how they are used in your installation. In addition, the content in the related training modules listed below may be helpful before reviewing the contents of this module.



Importing Project and Resource Data Overview

Create, modify, or delete multiple projects in a mass import of project data into Accolade using two specially named reference tables. The import process creates and updates project data, but does not create new metrics, access groups, or models.

Note: To import this type of data to Accolade, you must have the Process Designer, Project Importer, and Reference Table Manager user roles. To import a new version of an existing file, you must be assigned as the owner of the reference table file used for the import.

The following types of data can be imported:

	Import Options	
Data Type	**Spreadsheet without Reference Tables**	**Data/Config Reference Table Pairs (as spreadsheet, CSV, or XML)**
Matrices		X
Projects		X
Project Links		X
Resource Pools		X
Resources		X
Resource Capacities		X
Resource Demands		X
Resource Demand Curves		X
Resource Demand Values		X
Metrics	X	
Users	X	

Importing project data requires the following steps:

- Create a file that contains the data to import (Data table).
- Map the data to fields in the Accolade database (Config table).
- Validate and run the data import.
- Troubleshoot the import as necessary.

Importing and Exporting Data Best Practices

Keep the following set of best practice recommendations in mind when importing and exporting data:

- For reference table pairs that *import* data (Data/Config reference table pairs), every column referenced in the Config table must be present in the Data table, or the upload does not proceed. However, if there are extra columns in the Data table that are not referenced in the Config table, those columns are added to the new version of the reference table, but validation checks prevent the data from being imported.
- For reference table pairs that *update* data (Data/Config reference table pairs), if there is a change to the data, such as a metric column removed from the Data table, you must save the Config table containing that change to Accolade, before uploading the corresponding Data table. However, if the configuration remains unchanged, you can load multiple versions of the Data table in succession without the accompanying Config table.

Creating Matrix Data Import Files

Import updates to metric values in a matrix into Accolade using two specially named reference tables. The reference table pair begins with **ACC_PMI_**, which identifies the tables as reference tables that contain matrix update import data. Include only one matrix in a data/configuration file pair. Each matrix you want to update requires its own pair of reference tables.

Note: To import this type of data to Accolade, you must have the Process Designer, Project Importer, and Reference Table Manager user roles. To import a new version of an existing file, you must be assigned as the owner of the reference table file used for the import.

Create a File That Contains the Matrix Update Data to Upload (Data Table)

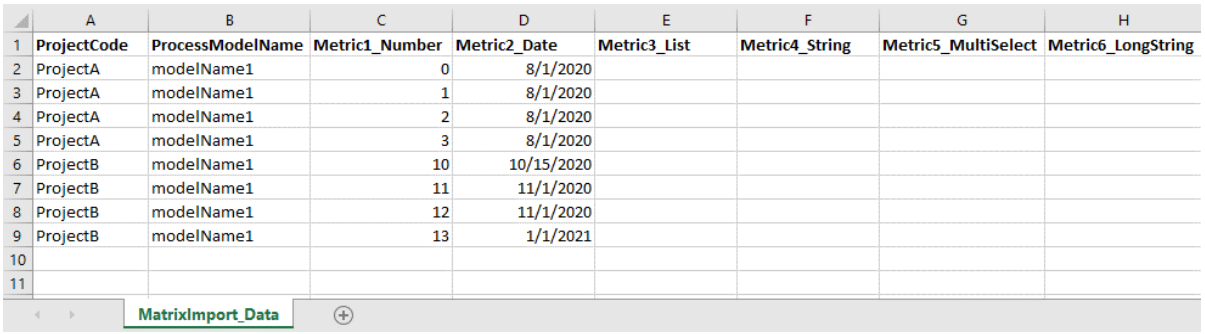
Important! The matrix import deletes an existing matrix referenced in the data table and *completely replaces* it with the values you provide in the file. Therefore, you must provide values for all the rows in the matrix. If a row is missing or a value is left blank, the existing values in the matrix are replaced with a blank value on import.

The data table contains the data to import to Accolade. Administrators and Process Designers add the initial versions of the **ACC_PMI** data table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the matrix data table as a spreadsheet, CSV, or XML file, ensuring that the contents of the file meets the following requirements:

Component	Requirements
File Name	Matrix updates must be in a file named **ACC_PMI_ <MatrixSystemName>_Data**, where <*MatrixSystemName* matches the <*MatrixSystemName*> in the matrix update configuration table and is unique within Accolade. For example, **ACC_PMI_ ConsumerElectronics_Data**. There must be only one pair of tables that contain a given matrix system name. Accolade can contain multiple different data/config reference table pairs to update multiple matrices.
Rows	Each row in the file represents a row in the matrix.
Column Names and Position	Column headings are in the first row of the worksheet. You can name columns as you see fit for your installation. Each metric within the matrix must be in a separate column, and the file must contain all the metrics within the matrix. Note that rich text metrics cannot be imported. Name the columns that contain metric data with their metric name and their data type. For example, ProjRisk_List, or Milestone_Date. This helps to identify the type of data required in the column. The configuration table described below maps the columns in the data table to the respective fields in the Accolade database.
Data	The following columns and data are ***required*** for each matrix to update: • **Project ID** or **Metric Name** - Either the Project ID that displays on the project header, or a string metric that identifies the project. • **Process Model Name** - Depending on your system's configuration, this may display on the **Project Home** page. • One column for each non-calculated metric in the matrix. It is not necessary to include a blank column for a calculated metric included in the matrix. The data type of each metric has to match the data type of its reference table column in the Data table *except* that list metrics are in string type columns. The data type of a reference table column is created automatically when the first version of the table is created, but it can be edited by the table owner.

EXAMPLE Example



	A	B	C	D	E	F	G	H
1	ProjectCode	ProcessModelName	Metric1_Number	Metric2_Date	Metric3_List	Metric4_String	Metric5_MultiSelect	Metric6_LongString
2	ProjectA	modelName1	0	8/1/2020				
3	ProjectA	modelName1	1	8/1/2020				
4	ProjectA	modelName1	2	8/1/2020				
5	ProjectA	modelName1	3	8/1/2020				
6	ProjectB	modelName1	10	10/15/2020				
7	ProjectB	modelName1	11	11/1/2020				
8	ProjectB	modelName1	12	11/1/2020				
9	ProjectB	modelName1	13	1/1/2021				
10								
11								

MatrixImport_Data

Note the following in the example above:

- Each metric contained in the matrix contains its own column.
- The import updates the metric values in a single matrix (based on the data/configuration file name) that is updated in both Project A and Project B, with the values for the respective projects.

Map the Matrix Data to the Accolade Database (Config Table)

The configuration table contains the data mapping information so Accolade knows what type of data to expect in the accompanying data file. Administrators and Process Designers add the initial versions of the **ACC_PMI** config table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table, as needed.

Create the matrix configuration table as a spreadsheet, CSV, or XML file, ensuring that the file meets the following requirements:

Component	Requirements
File Name	The matrix update configuration must be in a file named **ACC_PMI_ *<MatrixSystemName>*_Config**, where the *<MatrixSystemName>* matches the matrix updates data file and is unique within Accolade. For example, **ACC_PMI_ConsumerElectronics_Config**. There must be only one pair of tables that contain a given matrix system name. Accolade can contain multiple different data/config reference table pairs to update multiple matrices.
Rows	Each row maps a column in the data table to a metric or a metadata item.
Column Names and Position	The matrix update configuration must have the following column names, from left to right as listed below: • **Export Column Name** - Enter each column heading that is included in the data table. • **Metric System Name** - Enter the system name of the metric whose column heading is in the same row. • **Project Metadata Name** - Enter the metadata name below that

Component	Requirements
	identifies the data included is in each column in the data table. • **Unique Identifier** - Enter **TRUE** in the either the **ProjectCode** row or a row that contains a string metric that specifies which column in the data table contains the data that identifies the project in which to update the matrix values. Only one string metric can serve as the identifier for the import.
Accolade Metadata Names	Use the following metadata names in the **Project Metadata Name** column of the configuration table to map the data to the appropriate locations in the Accolade database: • ProjectCode • ProcessModelName - Required whether ProjectCode or a string metric is used.

EXAMPLE Example

	A	B	C	D	E	F	G	H
1	**ProjectCode**	**ProcessModelName**	**Metric1_Number**	**Metric2_Date**	**Metric3_List**	**Metric4_String**	**Metric5_MultiSelect**	**Metric6_LongString**
2	ProjectA	modelName1	0	8/1/2020				
3	ProjectA	modelName1	1	8/1/2020				
4	ProjectA	modelName1	2	8/1/2020				
5	ProjectA	modelName1	3	8/1/2020				
6	ProjectB	modelName1	10	10/15/2020				
7	ProjectB	modelName1	11	11/1/2020				
8	ProjectB	modelName1	12	11/1/2020				
9	ProjectB	modelName1	13	1/1/2021				
10								
11								

MatrixImport_Data

Each column in the data table is listed in the **Export Column Name** column in the configuration table.

	A	B	C	D
1	**Export Column Name**	**Metric System Name**	**Project Metadata Name**	**Unique Identifier**
2	ProjectCode		ProjectCode	TRUE
3	ProcessModelName		ProcessModelName	
4	Metric1_Number	Metric1		
5	Metric2_Date	Metric2		
6	Metric3_List	Metric3		
7	Metric4_String	Metric4		
8	Metric5_MultiSelect	Metric5		
9	Metric6_LongString	Metric6		
10				
11				
12				
13				

MatrixImport_Config

Note the following about the example above:

- Each item in the **Export Column Name** column is a column heading in the data table.
- The **Metric System Name** must match the metric system name exactly for the value to update.
- The names listed in the **Project Metadata Name** column must match the metadata names listed above exactly for the import to process successfully. Enter the metadata name for each column that matches, or maps, the data in the data file to the Accolade database to ensure the data is imported to the correct location.

- The **TRUE** setting in the **Unique Identifier** column indicates that project code is used to uniquely identify projects.

Creating Project Data Import Files

To create or conduct mass edits on project data, import project data into Accolade. Before running the import, create two specially named reference tables that will create a file that contains the data to import and map the data to fields in the Accolade database. The reference table pair begins with **ACC_PI_**, which identifies the tables as reference tables that contain project import data.

Note: To import this type of data to Accolade, you must have the Process Designer, Project Importer, and Reference Table Manager user roles. To import a new version of an existing file, you must be assigned as the owner of the reference table file used for the import.

Create a File That Contains the Project Data to Import (Data Table)

The data table contains the data to import to Accolade. Administrators and Process Designers add the initial versions of the **ACC_PI** data table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the project data table as a spreadsheet, CSV, or XML file, ensuring that the contents of the file meets the following requirements:

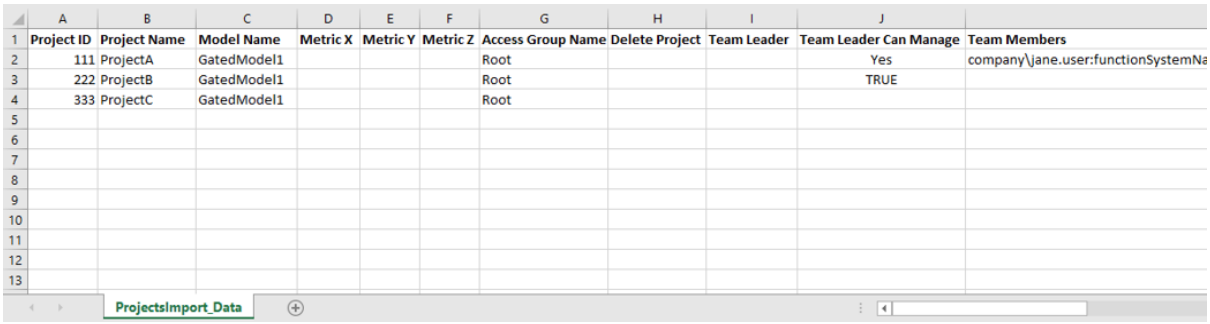
Important! Enter all dates in mm/dd/yyyy format, and enter the decimal separator in numbers as a period (.).

Component	Requirements
File Name	Project data must be in a file named **ACC_PI_<*pair identifier*>_Data**, where <*pair identifier*> matches the <*pair identifier*> in the project importer configuration table and is unique within Accolade. For example, **ACC_PI_ConsumerElectronics_Data**.
Rows	Each row contains a different project, and each project exists in only one row.
Column Names and Position	Column headings are in the first row of the worksheet. You can name columns as you see fit for your installation. The configuration table described below maps the columns in the data table to the respective fields in the Accolade database.
Data	The following column and data is ***required*** to create an imported project or to update an existing project: • **Project ID** - Either the **Project ID** displayed on the project header, not the system ID, or a string metric that is used to identify the project. No special characters or spaces are allowed. If the project code does not match any project on the importing server, the import creates a new project. If the project code does match, the data in the

Component	Requirements
	import is used to modify the matching project. **Important!** If project IDs for existing projects are calculated by a metric or created automatically by the **Auto-Generate Project IDs** system parameter, use a string metric as the unique identifier in the imported data. If you set the unique identifier to **Project Code** the projects import successfully, but create duplicate projects. Use a string metric as the unique identifier to avoid duplicating projects. You can ensure that projects are added by giving them project codes that you are certain do not match those of any existing projects in Accolade, such as a, b, c, etc. Project Codes entered as numbers strip all leading and trailing zeros, unless you first import with a reference table of the same name without data that sets the project code to a **String** data type. The following columns and data are ***required*** to create an imported project, but ***optional*** when updating an existing project: • **Project Name** - The name, up to 64 characters long, that identifies the project's purpose. If the project name is generated using a calculated metric, this field can be blank and is not required. • **Model Name** - The name of the model the project follows for completion. You can update projects that use an in-active process model. • **Access Group Name** - The name of the access group that determines which users have access to this project. The access group must already exist in Accolade, the import process does not create access groups. The following columns and data are ***optional*** when creating or updating projects: • **Metrics** - Each metric field is in a separate column. When importing values for multi-select list metrics, enter each selection for the metric in a single cell in a pipe (|) delimited list. For example, Value-A|Value-B. The data type for each imported metric must match the data type of its reference table column in the Data table. List metrics are in string type columns. Rich text metrics cannot be imported. • **Delete Project** - Enter **Yes, Y, True,** or **1** to delete the project. All other values are treated as **No**. • **Team Leader** - The login of the Accolade user that is assigned as the project's project manager. Only users with the Project Manager

Component	Requirements
	user role can be assigned as team leaders. If you create a project without designating a project manager, you can select one later. • **Team Leader Can Manage** - Enter **Yes, Y, True,** or **1** if the user in the **Team Leader** field also is given the rights to assign and replace team members on the project. All other values are treated as **No**. • **Team Members** - The user login and system name of the function the user is assigned, using the following format: *<domain>\<UserLogin1>:<systemNameFunction1>*. Separate users with a pipe (|) character to create a delimited list. If the colon is not specified, all text between the pipes (|) is considered a user login. • **Auto Migrate Rules** - The system name of the workflow containing rules to migrate a project to a different model. The workflow does not need to be in a certain state or associated with a model. If the import contains projects with migrations, a slight delay in migrating the projects could occur after the import completes. • **Enforce Project Security** - Enter **Yes, Y, True,** or **1** if the team members and the project manager must have access to the project through access groups, security lists, or security profiles. All other values are treated as **No**. Any users that do not have the same security as the project are not added to the team. Existing users on projects that are being updated are removed accordingly. • **Project Description** - Enter the project description. • **Project Start Date** - Enter the project start date in mm/dd/yyyy format. • **Metadata Fields** - Each metadata field is in a separate column. Enter **Yes, Y, True,** or **1** for metadata fields that are either on or off, such as **VisibleToReports**. • **Security Lists** - For the project's security list access, create a column for each security list in the system. For each project, enter the ID numbers of the project's selected security list items in a pipe delimited list that begins and ends with a pipe (|) character. The pipes ensure that each security list column is created as a String type in the reference table. If the lists were created manually, the IDs can be found in the database. If the lists were created in reference tables, the IDs are found in the ID column.

Example



	A	B	C	D	E	F	G	H	I	J	
1	Project ID	Project Name	Model Name	Metric X	Metric Y	Metric Z	Access Group Name	Delete Project	Team Leader	Team Leader Can Manage	Team Members
2	111	ProjectA	GatedModel1				Root			Yes	company\jane.user:functionSystemNa
3	222	ProjectB	GatedModel1				Root			TRUE	
4	333	ProjectC	GatedModel1				Root				
5											
6											
7											
8											
9											
10											
11											
12											
13											

ProjectsImport_Data

Note the following in the example above:

- The import includes the required information - project code, project name, model name, and access group name.
- The import does not delete any projects, because the **Delete** column is blank.
- The import does not add any **Team Leaders** (Project Managers) because the column is blank.
- The import adds user jane.user to ProjectA as a team member.

Map the Project Data to the Accolade Database (Config Table)

The configuration table contains the data mapping information so Accolade knows what type of data to expect in the accompanying data file. Administrators and Process Designers add the initial versions of the **ACC_PI** config table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the projects configuration table as a spreadsheet, CSV, or XML file, ensuring that the file meets the following requirements:

Component	Requirements
File Name	The project configuration must be in a file named **ACC_PI_*<pair identifier>*_Config**, where the *<pair identifier>* matches the project data file and is unique within Accolade. For example, **ACC_PI_ ConsumerElectronics_Config**.
Rows	Each row maps a column in the data table to a specific type of project data.
Column Names and Position	The project configuration must have the following column names, from left to right as listed below: • **Export Column Name** - Enter each column heading that is included in the data table. • **Metric System Name** - Contains the system name of the metric whose column in the data table is in the same row. • **Project Metadata Name** - Identifies the data in the data table whose name is in the same row. • **Unique Identifier** - Specifies the column containing the data that

Component	Requirements
	uniquely identifies the projects. The column can be either the metadata **ProjectCode** or a string metric. Only one string metric can serve as the identifier for the import. Enter **TRUE** in the row that contains the identifier column heading name. **Important!** If project IDs for existing projects are calculated by a metric or created automatically by the **Auto-Generate Project IDs** system parameter, use a string metric as the unique identifier in the imported data. If you set the unique identifier to **Project Code** the projects import successfully, but create duplicate projects. Use a string metric as the unique identifier to avoid duplicating projects. A unique identifier is required to run a successful import. If Accolade is set to auto-generate project IDs, use a string metric as the unique identifier in the imported data. New projects are created with an auto-generated ID.
Accolade Metadata Names	Use the metadata names as listed in **Allowed Project Metadata** in the **Metadata Name** column of the configuration table to map the data to the appropriate locations in the Accolade database.

EXAMPLE Example

	A	B	C	D	E	F	G	H	I	J	
1	Project ID	Project Name	Model Name	Metric X	Metric Y	Metric Z	Access Group Name	Delete Project	Team Leader	Team Leader Can Manage	Team Members
2	111	ProjectA	GatedModel1				Root			Yes	company\jane.user:functionSystemNa
3	222	ProjectB	GatedModel1				Root			TRUE	
4	333	ProjectC	GatedModel1				Root				

ProjectsImport_Data

Each column in the data table is listed in the **Export Column Name** column in the configuration table.

	A	B	C	D
1	Export Column Name	Metric System Name	Project Metadata Name	Unique Identifier
2	Project ID		ProjectCode	TRUE
3	Project Name		ProjectName	
4	Model Name		ProcessModelName	
5	Metric X			
6	Metric Y			
7	Metric Z			
8	Access Group Name		AccessGroup	
9	Delete Project		DeleteProject	
10	Team Leader		TeamLeaderLogin	
11	Team Leader Can Manage		TeamLeaderCanManageTeam	
12	Team Members		TeamMemberLogins	
13	Auto Migrate Rules		AutoMigrateRules	
14	Enforce Project Security		EnforceProjectSecurity	
15	Project Description		ProjectDescription	
16	Project Start Date		ProjectStartDate	
17	Metadata X			
18	Metadata Y			
19	Metadata Z			
20				

ProjectsImport_Config

Note the following about the example above:

- Each item in the **Export Column Name** column is a column heading in the data table.
- The **Metric System Name** must match the metric system name exactly for the value to update.
- The names listed in the **Project Metadata Name** column must match the metadata names listed above exactly for the import to process successfully. Enter the metadata name for each column that matches, or maps, the data in the data file to the Accolade database to ensure the data is imported to the correct location.
- The **TRUE** setting in the **Unique Identifier** column indicates that project code is used to uniquely identify projects.

Allowed Project Metadata

Use the metadata names as listed in below the **Project Metadata Name** column of the configuration table to map the data to the appropriate locations in the Accolade database. Unless otherwise noted, if an item is missing or invalid, the import ignores the data and makes no updates in Accolade.

Column Name	Accepted Values	Additional Notes
ProjectCode	String	
ProjectName	String	

Column Name	Accepted Values	Additional Notes
ProcessModelName	Valid Process Model display name	
AccessGroup	Valid Access Group display name	
DeleteProject	Yes, Y, True, 1*	
TeamLeaderLogin	user login	As it displays for the user on the User Admin page.
TeamLeaderCanManageTeam	Yes, Y, True, 1*	
AllowTeamMemberDeletion	Yes, Y, True, 1*	
TeamMemberLogins	user login	As it displays for the user on the User Admin page. Login and function combinations are in a pipe-delimited string: *<domain>* \ *<UserLogin1>* : *<systemNameFunction1>* | *<domain>* \ *<UserLogin2>* :*<systemNameFunction2>*. Invalid team member logins are skipped but the rest of the project is created.
AutoMigrateRules	Configured migration rule name	Values can be found on the workflow details page.

Column Name	Accepted Values	Additional Notes
		The workflow must be checked as **Is Smart** with a migration rule.
EnforceProjectSecurity	Yes, Y, True, 1*	
ProjectDescription	Any	
ProjectStartDate	Date	
ProjectEndDate	Date	
ProjectClosed	Yes, Y, True, 1*	
ProjectClosedNotes	Any	Only updated if the project is closed.
VisibleToReports	Yes, Y, True, 1*	
ExcludeFromLineUp	Yes, Y, True, 1*	
LastGateNumber	Numeric	Requires LastGateDecisionName in the table. **Important!** LastGateNumber and LastGateDecisionName changes can only be applied to current or future gates. If you are updating a project and the LastGateNumber is set to a gate that has already past, the import does not update the gate information. For example, you have an existing project that had gate decisions of Go for gates 1 & 2, and is currently approaching gate 3. If your project

Column Name	Accepted Values	Additional Notes
		import has a **LastGateNumber** of 2 and a **LastGateDecisionName** value of Conditional Go, the import does not update the changes for the gate 2.
LastGateDecisionName	String	Requires LastGateNumber in the table. Values are available on the **Configuration > Entity Names** page.
IdeaSubmitterLogin	user login	As it displays for the user on the User Admin page.
IdeaSubmitterEmail	String	Must be in a valid email format (with the @ and . in the address, for example, name@sopheon.com)
IdeaSubmitterName	String	
NotifySubmitter	Yes, Y, True, 1*	
InTrouble	Yes, Y, True, 1*	
InTroubleReason	String	
InTroubleNotes	String	
TeamLeaderChangeReason	Event Reason Code	Event reason code is configured on the Event Reason List tab of the Class definition page.

Column Name	Accepted Values	Additional Notes
GateDate01, GateDate02,...GateDate20	Date	Updates to gate dates through an import are considered manually entered dates for the project. If gate dates for a project are set using a metric defined in the model, any updates to the gate date through an import override the date set by the metric. Each gate date must be represented in individual columns.
ProjectCurrencyCode	three character currency code value	Currency codes are configured using the currency reference table.
ProjectSecurityList1...ProjectSecurityList 5	Security List value ID	If security lists are disabled in your system, or you provide a security list that does not exist, the project is created without a security list designation. To include more than one security list use a (|) pipe character.

* For any column that accepts **Yes**, **Y**, **True**, or **1**, you can also enter **No**, **N**, **False**, or **0** if it helps you when entering data in the spreadsheet. All values other than **Yes**, **Y**, **True**, or **1** are treated as **No** when you upload the spreadsheet.

Notes:

- If you want to use a saved team for projects created through the import process, these teams must be manually assigned on the project's Team page.

Creating Project Link Data Import Files

Import project link data into Accolade using two specially named reference tables. The reference table pair begins with **ACC_PLI_**, which identifies the tables as reference tables that contain project link import data.

Note: To import this type of data to Accolade, you must have the Process Designer, Project Importer, and Reference Table Manager user roles. To import a new version of an existing file, you must be assigned as the owner of the reference table file used for the import.

Create a File That Contains the Project Link Data to Import (Data Table)

The data table contains the data to import to Accolade. Administrators and Process Designers add the initial versions of the **ACC_PLI** data table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the project links data table as a spreadsheet, CSV, or XML file, ensuring that the contents of the file meets the following requirements:

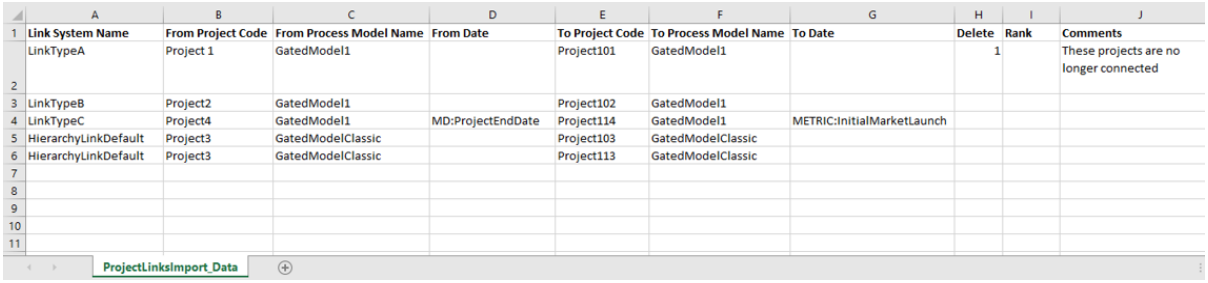
Component	Requirements
File Name	Project link data must be in a file named **ACC_PLI_*<pair identifier>*_Data**, where *<pair identifier>* matches the *<pair identifier>* in the project links configuration table and is unique within Accolade. For example, **ACC_PLI_BabyCare_Data**.
Rows	Each row contains a different project link, and each project link exists in only one row.
Column Names and Position	Column headings are in the first row of the worksheet. You can name columns as you see fit for your installation. The configuration table described below maps the columns in the data table to the respective fields in the Accolade database.

Data	The following columns and data are ***required*** for each project link:
	• **Link System Name** - The system name of the link type that this project link is based on. The **HierarchyLinkDefault** link type creates a project portfolio (adds child projects to a portfolio project). Importing this link type adds the Link To project (child) to the portfolio of the Link From project (parent). Additional link system names at your company can be found on the Link Type page (**Process > Configuration > Link Types**).
	• **From Project Code** - The **Project Code** displayed on the project header to identify the project the link points *from* (for hierarchy links, this is the code of the parent project). No special characters or spaces are allowed.
	• **From Process Model Name** - The name of the model of the project the link points from.
	• **From Date** - Required only for project links that use the **Date - Date** link type. The **From Date** can be any milestone on the project. Enter one of the following:
	  • **Milestone date metric** - METRIC:*<metric system name>*
	  • **Project start date** - MD:ProjectStartDate
	  • **Project end date** - MD:ProjectEndDate
	  • **Gate** - MD:ProjectNextGateDate, MD:ProjectPreviousGateDate, MD:ProjectFollowingGateDate, MD:ProjectGateDate-*<gate number>*, or METRIC:*<metric system name>*

Component	Requirements

	• **To Project Code** - The **Project Code** displayed on the project header to identify the project the link points *to* (for hierarchy links, this is the child project). No special characters or spaces are allowed.
	• **To Process Model Name** - The name of the model of the project the link points to.
	• **To Date** - Required only for project links that use the **Date - Date** link type. The **To Date** can be any milestone within a project. Enter one of the following:
	  • **Milestone date metric** - METRIC:<*metric system name*>
	  • **Project start date** - MD:ProjectStartDate
	  • **Project end date** - MD:ProjectEndDate
	  • **Gate** - MD:ProjectNextGateDate, MD:ProjectPreviousGateDate, MD:ProjectFollowingGateDate, MD:ProjectGateDate-<*gate number*>, or METRIC:<*metric system name*>
	The projects identified in the **From Project Code** and **To Project Code** data columns must use process models that are selected for the link type identified in the **Link System Name** column. If the process models are not included in the link type definition, the import does not create the links.
	
	The following columns and data are ***optional*** for each project link:
	• **Delete Link** - Enter **Yes, Y, True,** or **1** to delete the link. Enter **0** or leave blank to add or modify the link.
	• **Comments** - Enter a description of the link.
	• **Rank** - Sets the rank of the Link To project in the portfolio of the Link From project. Used when the **Link System Name** is **HierarchyLinkDefault**.

EXAMPLE Example



	A	B	C	D	E	F	G	H	I	J
1	**Link System Name**	**From Project Code**	**From Process Model Name**	**From Date**	**To Project Code**	**To Process Model Name**	**To Date**	**Delete**	**Rank**	**Comments**
2	LinkTypeA	Project 1	GatedModel1		Project101	GatedModel1		1		These projects are no longer connected
3	LinkTypeB	Project2	GatedModel1		Project102	GatedModel1				
4	LinkTypeC	Project4	GatedModel1	MD:ProjectEndDate	Project114	GatedModel1	METRIC:InitialMarketLaunch			
5	HierarchyLinkDefault	Project3	GatedModelClassic		Project103	GatedModelClassic				
6	HierarchyLinkDefault	Project3	GatedModelClassic		Project113	GatedModelClassic				
7										
8										
9										
10										
11										

ProjectLinksImport_Data

Note the following in the example above:

- The import creates a link between the project with code Project2 and Project 102 using the LinkTypeB link type.
- The import creates a parent-child hierarchy link between Project3 (the parent) and Project103 and Project113 (the children) using the **HierarchyLinkDefault** link type.
- The import creates a link with a dependency on Project4 project end date and the Initial market Launch date metric.
- The import removes the link between the projects in row two.

Map the Project Link Data to the Accolade Database (Config Table)

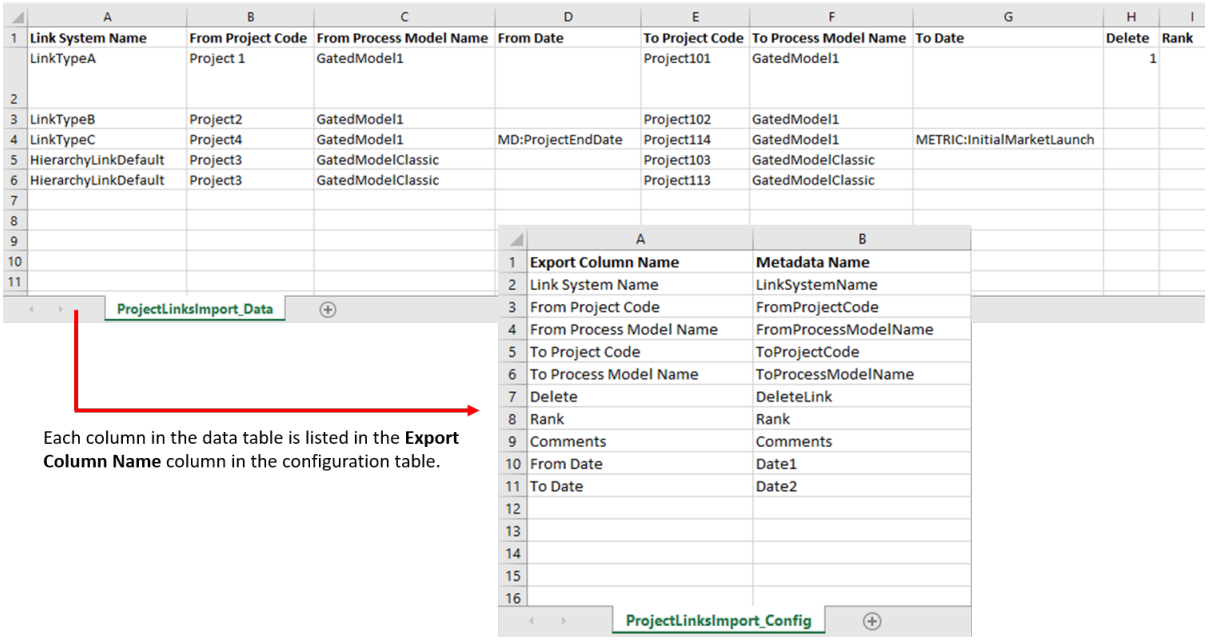
The configuration table contains the data mapping information so Accolade knows what type of data to expect in the accompanying data file. Administrators and Process Designers add the initial versions of the **ACC_PLI** config table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the project links configuration table as a spreadsheet, CSV, or XML file, ensuring that the file meets the following requirements:

Component	Requirements
File Name	The project link configuration must be in a file named **ACC_PLI_<*pair identifier*>_Config**, where the <*pair identifier*> matches the project link data file and is unique within Accolade. For example, **ACC_PLI_ BabyCare_Config**.
Rows	Each row maps a column in the data table to a specific type of project data.
Column Names and Position	The project links configuration must have the following column names, from left to right as listed below: • **Export Column Name** - Enter each column heading that is included in the data table. • **Metadata Name** - Enter the metadata name below that identifies the data included is in each column in the data table.
Accolade Metadata Names	Use the following metadata names in the **Metadata Name** column of the configuration table to map the data to the appropriate locations in the Accolade database: • LinkSystemName • FromProjectCode • FromProcessModelName • ToProjectCode • ToProcessModelName • Date1 • Date2 • DeleteLink

Component	Requirements
	• Comments • Rank

EXAMPLE Example



Note the following about the example above:

- Each item in the **Export Column Name** column is a column heading in the data table.
- The names listed in the **Metadata Name** column must match the metadata names listed above exactly for the import to process successfully. Enter the metadata name for each column that matches, or maps, the data in the data file to the Accolade database to ensure the data is imported to the correct location.

Creating Resource Pool Import Files

Import resource pool data to add new pools and to modify existing pools using two specially named reference tables. The reference table pair begins with **ACC_RPI_**, which identifies the tables as reference tables that contain resource pool import data.

Note: To import this type of data to Accolade, you must have the Process Designer, Project Importer, and Reference Table Manager user roles. To import a new version of an existing file, you must be assigned as the owner of the reference table file used for the import.

Create a File That Contains the Resource Pool Data to Import (Data Table)

The data table contains the data to import to Accolade. Administrators and Process Designers add the initial versions of the **ACC_RPI** data table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the resource pools data table as a spreadsheet, CSV, or XML file, ensuring that the contents of the file meets the following requirements:

Component	Requirements
File Name	Resource pool data must be in a file named **ACC_RPI_*<pair identifier>*_Data**, where *<pair identifier>* matches the *<pair identifier>* in the resource pools configuration table and is unique within Accolade. For example, **ACC_RPI_GlobalPools_Data**.
Rows	Each row contains a different resource pool, and each resource pool exists in only one row.
Column Names and Position	Column headings are in the first row of the worksheet. You can name columns as you see fit for your installation. The configuration table described below maps the columns in the data table to the respective fields in the Accolade database.
Data	The following columns and data are ***required*** for each resource pool: • **Pool Name** - Identifies the resource pool in the system. The pool name must be unique. • **Pool Unit of Measure** - The chosen unit of measure in which requests are made for resources that belong in this pool. For example, FTE for a full time employee. The following columns and data are ***optional*** for each resource pool: • **Pool Owner Login** - The user login of the Resource Demand Planner who manages the pool. If left blank for a pool any Resource Demand Planner is able to manage the pool. To add additional owners to a pool using the import, run the import once with the additional owner user login listed in the **Pool Owner** column. Run the import a second time, updating the **Pool Owner** column to the user login of the primary pool owner. The import moves the original owner of the pool to an additional owner, and imports the new owner as the primary pool owner. **Note**: If the **Pool Owner** column is not included in the import files, the **Pool Owner** is set to **NONE**, and any existing assigned owner is set as an **Additional Owner**. • **Pool Requested** - Enter **Yes, Y, True,** or **1** to set the pool's demand

Component	Requirements
	type to **Requested** (all other values indicate **False**). If a value is present in this column, a value is also required in the **Assigned** column. A pool can be **Requested**, **Assigned**, or both. Defaults to **TRUE** if the column is not provided on initial pool creation. • **Pool Assigned** - Enter **Yes, Y, True,** or **1** to set the pool's demand type to **Assigned** (all other values indicate **False**). If a value is present in this column, a value is also required in the **Requested** column A pool can be **Requested**, **Assigned**, or both. Defaults to **TRUE** if the column is not provided on initial pool creation. • **Pool Parent** - For assigned-only pools, enter the pool name of the requested-only pool to which the assigned pool is linked. You can associate assigned-only pools with a parent requested-only pool to establish an associated between the pools, which allows Resource Demand Planners to know which group of resources they can assign to certain requests. Assigned-only pools can be linked to multiple parent pools if the **Allow Multiple Links For an Assigned Pool** system parameter is enabled. If disabled, an assigned-only pool can be linked to only one requested-only pool. List multiple parents using a pipe (|) delimited list. --- **Important!** When linking assigned-only and requested-only pools, the pool named in the **Pool Parent** column must exist and must be a requested-only pool. Each pool in the link must contain the same **Pool Unit of Measure**. If you are linking pools, complete the import in two stages: one to import the requested-only pools, and one to import the assigned-only pools. Completing the import in two stages helps to ensure that the pool names listed in the **Pool Parent** column are established prior to trying to establish a link. To make demand type or unit of measure changes to a pool that is linked to another pool, you must first remove the link before making any changes to either pool. --- • **Pool Security List (1-5)** - The security list system name for each list applied to a pool. Security Lists IDs are unique, listed in a pipe (|) delimited list. • **Pool Extended Fields** - One or more columns containing the values in extended fields set to apply to pools. • **Active** - Enter **Yes, Y, True,** or **1** to activate the pool. Other values or an empty cell sets the status of the pool to **Inactive**. **Note**: If the **Active** column is not included in the import files, the status of the pool is set to **Inactive** as well.

Component	Requirements
	• **Delete** - Enter **Yes, Y, True,** or **1** to delete the resource pool. The importer will delete all pools even if it has resources within it with existing, active demands against it. As a best practice, delete or move the demands to another pool before running the importer. • **Pool ID** - To rename an existing pool, provide the existing pool's system ID in a Pool ID column, and the new pool name in the **Pool Name** column. You can find existing pool IDs with the **System Resource Pool ID** column using reports.

EXAMPLE Example

	A	B	C	D	E	F	G	H	I	J
1	Pool Name	Unit of Measure	PoolOwner	PoolRequested	PoolAssigned	Pool Parent	Location	Active	Delete	Pool Id
2	Project Managers	FTE	companyDomain\jane_manager	Y	N		USA	TRUE		
3	R&D Engineers	FTE	companyDomain\jane_manager	Y	N		Asia	TRUE		45
4	Manufacturing	FTE	companyDomain\joe_manufacturer	Y	Y	Project Managers	USA	TRUE		
5	Operations	FTE	companyDomain\fred_operations	Y	N		Europe	TRUE		
6										
7										
8										
9										
10										
11										

ResourcePoolsImport_Data

Note the following in the example above:

- The import renames the existing resource pool in row three to R&D Engineers.
- The import creates a link between the Manufacturing and Project Managers pools.
- The import assigns a value to the Location extended field that is available to pools for each of the pools listed.

Map the Resource Pools Data to the Accolade Database (Config Table)

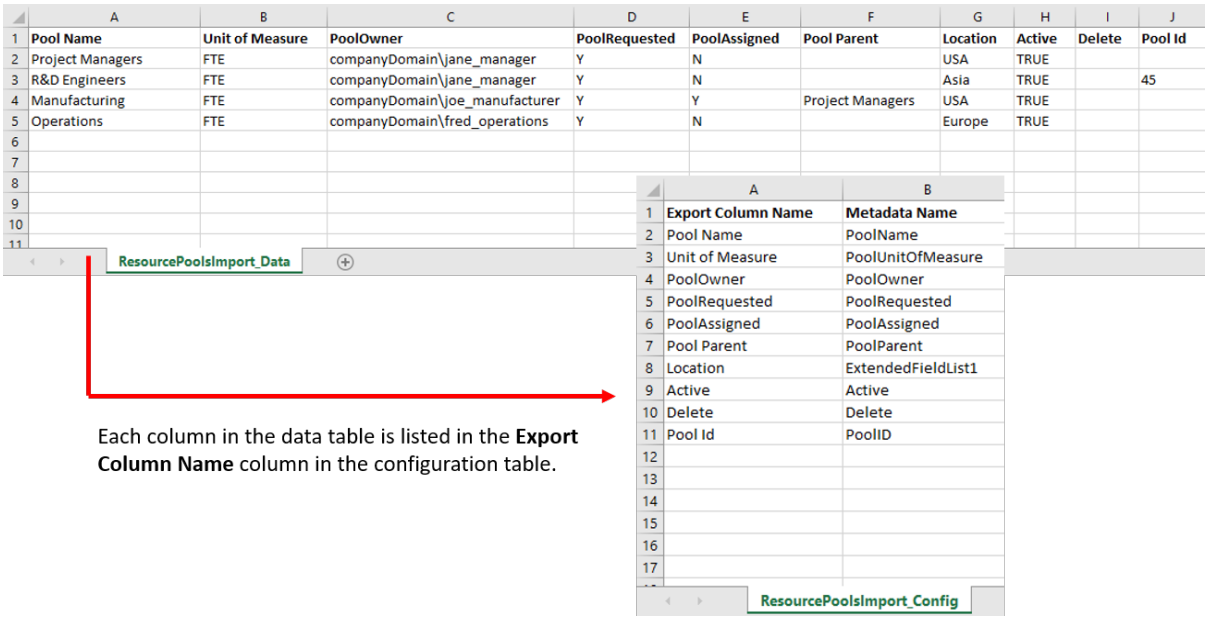
The configuration table contains the data mapping information so Accolade knows what type of data to expect in the accompanying data file. Administrators and Process Designers add the initial versions of the **ACC_RPI** config table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the resource pools configuration table as a spreadsheet, CSV, or XML file, ensuring that the file meets the following requirements:

Component	Requirements
File Name	The resource pools configuration must be in a file named **ACC_RPI_*<pair identifier>*_Config**, where the *<pair identifier>* matches the resource pools data file and is unique within Accolade. For example, **ACC_RPI_GlobalPools_Config**.
Rows	Each row maps a column in the data table to a specific type of resource pool data in Accolade.

Component	Requirements
Column Names and Position	The resource configuration must have the following column names, from left to right as listed below: • **Export Column Name** - Enter each column heading that is included in the data table. • **Metadata Name** - Enter the metadata name below that identifies the data included is in each column in the data table.
Accolade Metadata Names	Use the following metadata names in the **Metadata Name** column of the configuration table to map the data to the appropriate locations in the Accolade database: • PoolName • PoolOwner • PoolUnitOfMeasure • PoolRequested • PoolAssigned • PoolParent • SecurityList(1-5) - For example, SecurityList2 • ExtendedField(DefaultName) - Name is "Extended Field" plus the default display name of the field. For example, ExtendedFieldMultiSelect6, ExtendedFieldNumber2, ExtendedFieldString3, and so on. • Delete • Active • PoolID

Example



	A	B	C	D	E	F	G	H	I	J
1	**Pool Name**	**Unit of Measure**	**PoolOwner**	**PoolRequested**	**PoolAssigned**	**Pool Parent**	**Location**	**Active**	**Delete**	**Pool Id**
2	Project Managers	FTE	companyDomain\jane_manager	Y	N		USA	TRUE		
3	R&D Engineers	FTE	companyDomain\jane_manager	Y	N		Asia	TRUE		45
4	Manufacturing	FTE	companyDomain\joe_manufacturer	Y	Y	Project Managers	USA	TRUE		
5	Operations	FTE	companyDomain\fred_operations	Y	N		Europe	TRUE		
6										
7										
8										
9										
10										
11										

ResourcePoolsImport_Data

Each column in the data table is listed in the **Export Column Name** column in the configuration table.

	A	B
1	**Export Column Name**	**Metadata Name**
2	Pool Name	PoolName
3	Unit of Measure	PoolUnitOfMeasure
4	PoolOwner	PoolOwner
5	PoolRequested	PoolRequested
6	PoolAssigned	PoolAssigned
7	Pool Parent	PoolParent
8	Location	ExtendedFieldList1
9	Active	Active
10	Delete	Delete
11	Pool Id	PoolID
12		
13		
14		
15		
16		
17		

ResourcePoolsImport_Config

Note the following about the example above:

- Each item in the **Export Column Name** column is a column heading in the data table.
- The names listed in the **Metadata Name** column must match the metadata names listed above exactly for the import to process successfully. Enter the metadata name for each column that matches, or maps, the data in the data file to the Accolade database to ensure the data is imported to the correct location.

Notes:

- If an assigned pool that is linked to a requested pool is deleted and auto-calculated capacities are set for the requested pool, the system auto-calculates capacities accordingly.

Creating Resource Import Files

Import resources data into Accolade using two specially named reference tables. The reference table pair begins with **ACC_RI_**, which identifies the tables as reference tables that contain resource import data.

Note: To import this type of data to Accolade, you must have the Process Designer, Project Importer, and Reference Table Manager user roles. To import a new version of an existing file, you must be assigned as the owner of the reference table file used for the import.

Create a File That Contains the Resource Data to Import (Data Table)

The data table contains the data to import to Accolade. Administrators and Process Designers add the initial versions of the **ACC_RI** data table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the resources data table as a spreadsheet, CSV, or XML file, ensuring that the contents of the file meets the following requirements:

Component	Requirements
File Name	Resource data must be in a file named **ACC_RI_*<pair identifier>*_Data**, where *<pair identifier>* matches the *<pair identifier>* in the resources configuration table and is unique within Accolade. For example, **ACC_RI_ResourcesImport_Data**.
Rows	Each row contains a different resource, and each resource exists in only one row.
Column Names and Position	Column headings are in the first row of the worksheet. You can name columns as you see fit for your installation. For example, **AutoCalc** might be better column name for you than **Calculated Capacity** to indicate what data is required in that column. The configuration table described below maps the columns in the data table to the respective fields in the Accolade database.
Data	The following columns and data are ***required*** for each resource: • **Pool Name** - The name of the pool to which the resource belongs. This important does not create new resource pools; if the pool name entered does not exist, the row is skipped during import. • **Resource Type** - Set to either **User** or **General** to identify the resource as a user in the system or a general resource, such as Any Engineer. • **Resource System Name** - A unique, shorter name that identifies the resource in queries, reporting views, Accolade Office Extensions, and field codes. You can define the system names for both User and General resources. • **Resource Name** - Required if the resource type is **General**. For example, Any Engineer or Any Chemist. • **Resource User Login** - Required if resource type is **User**. If the user login does not exist, or exists as a resource with existing demands, the row is skipped during import. If the user login exists, and is currently not part of the pool, the user is added as a resource to the specified pool. The following columns and data are ***optional*** for each resource: • **Delete** - Enter **Yes, Y, True,** or **1** to delete the resource from the pool. You cannot delete a resource from a pool if it has existing

Component	Requirements
	demands against it. The demands must either be deleted or moved to another resource first. • **Calculated Capacity** - Typically set for a General resource. If you use requested-only pools and relate them to assigned-only pools, enter **Yes, Y, True,** or **1** for one resource in a pool to indicate that resource as the one that auto-calculates the resource capacity for the pool. As resources are added or removed from a pool that has auto-capacities set, the system recalculates the pool capacity accordingly. Only one resource in a pool can be set to automatically calculate capacity for the pool. If multiple resources are set for auto-calculation, the import fails.

An existing resource is updated if the **Pool Name** and either the **Resource System Name** or **Resource User Login** match.

EXAMPLE Example

	A	B	C	D	E	F	G
1	**Pool Name**	**Resource Type**	**Resource System Name**	**Resource Name**	**Resource User Login**	**Delete**	**AutoCalc**
2	Senior Engineers	General	AnySeniorEngineer	Any Senior Engineer			Y
3	Senior Engineers	User	JaneDoe		companyDomain\jane_doe		
4	Senior Engineers	User	JohnDoe		companyDomain\john_doe	TRUE	
5							
6							
7							
8							
9							
10							
11							
12							

ResourcesImport_Data

Note the following in the example above:

- The import adds resources to the Senior Engineers pool. A single file can include updating and adding resources to more than one pool.
- The import adds Jane Doe (row 3) as a member of the Senior Engineers pool.
- The import removes John Doe (row 4) as a member of the Senior Engineers pool.

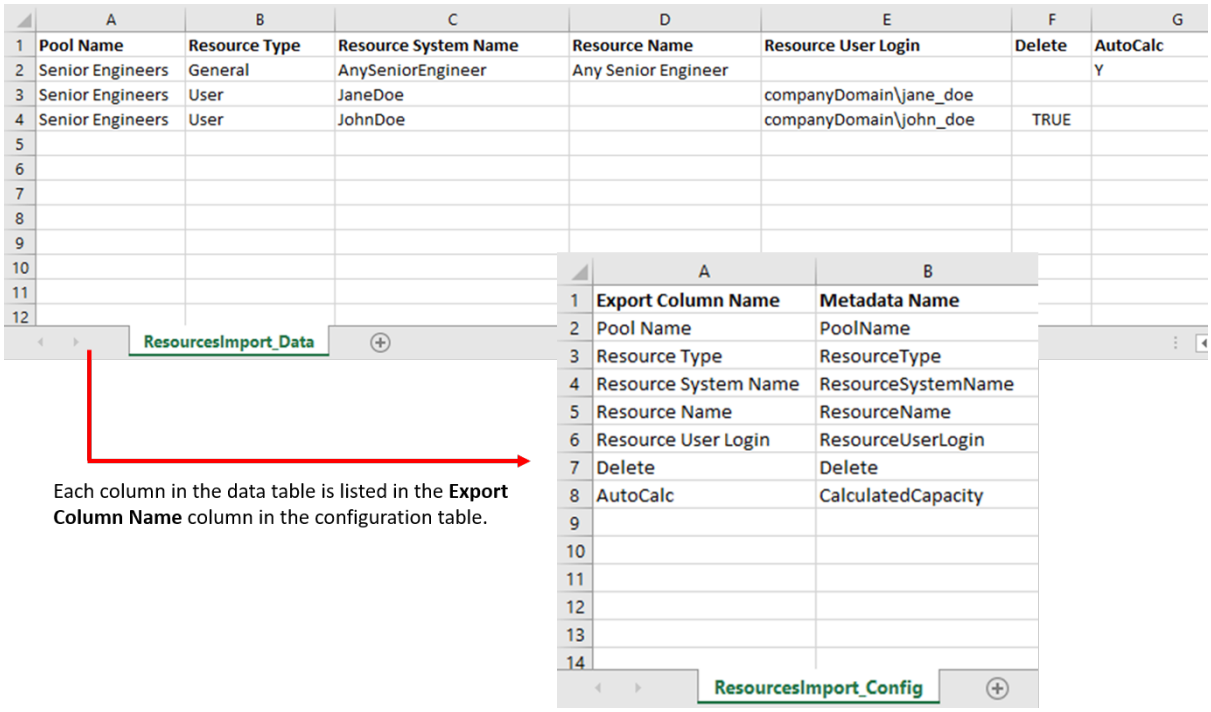
Map the Resources Data to the Accolade Database (Config Table)

The configuration table contains the data mapping information so Accolade knows what type of data to expect in the accompanying data file. Administrators and Process Designers add the initial versions of the **ACC_RI** config table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the resources configuration table as a spreadsheet, CSV, or XML file, ensuring that the file meets the following requirements:

Component	Requirements
File Name	The resource configuration must be in a file named **ACC_RI_*<pair identifier>*_Config**, where the *<pair identifier>* matches the resources data file and is unique within Accolade. For example, **ACC_RI_ResourcesImport_Config**.
Rows	Each row maps a column in the data table to a specific type of resource data.
Column Names and Position	The resource configuration must have the following column names, from left to right as listed below: • **Export Column Name** - Enter each column heading that is included in the data table. • **Metadata Name** - Enter the metadata name below that identifies the data included is in each column in the data table.
Accolade Metadata Names	Use the following metadata names in the **Metadata Name** column of the configuration table to map the data to the appropriate locations in the Accolade database: • ResourceType • ResourceSystemName • ResourceName • ResourceUserLogin • PoolName • Delete • CalculatedCapacity

Example



	A	B	C	D	E	F	G
1	**Pool Name**	**Resource Type**	**Resource System Name**	**Resource Name**	**Resource User Login**	**Delete**	**AutoCalc**
2	Senior Engineers	General	AnySeniorEngineer	Any Senior Engineer			Y
3	Senior Engineers	User	JaneDoe		companyDomain\jane_doe		
4	Senior Engineers	User	JohnDoe		companyDomain\john_doe	TRUE	

ResourcesImport_Data

Each column in the data table is listed in the **Export Column Name** column in the configuration table.

	A	B
1	**Export Column Name**	**Metadata Name**
2	Pool Name	PoolName
3	Resource Type	ResourceType
4	Resource System Name	ResourceSystemName
5	Resource Name	ResourceName
6	Resource User Login	ResourceUserLogin
7	Delete	Delete
8	AutoCalc	CalculatedCapacity

ResourcesImport_Config

Note the following about the example above:

- Each item in the **Export Column Name** column is a column heading in the data table.
- The names listed in the **Metadata Name** column must match the metadata names listed above exactly for the import to process successfully. Enter the metadata name for each column that matches, or maps, the data in the data file to the Accolade database to ensure the data is imported to the correct location.

Creating Resource Demand Curve Import Files

Import resource demand curves data into Accolade using two specially named reference tables. The reference table pair begins with **ACC_RDCI_**, which identifies the tables as reference tables that contain demand curve import data.

Note: To import this type of data to Accolade, you must have the Process Designer, Project Importer, and Reference Table Manager user roles. To import a new version of an existing file, you must be assigned as the owner of the reference table file used for the import.

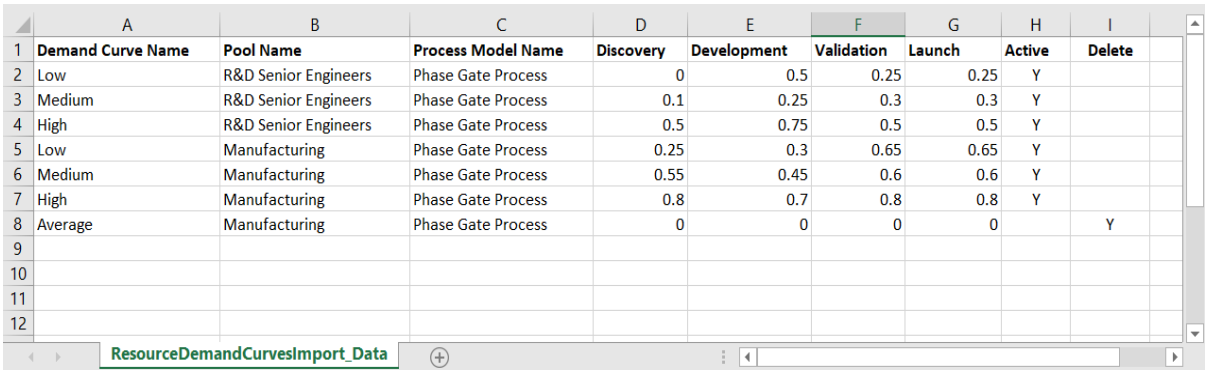
Create a File That Contains the Demand Curve Data to Import (Data Table)

The data table contains the data to import to Accolade. Administrators and Process Designers add the initial versions of the **ACC_RDCI** data table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the resource demand curve data table as a spreadsheet, CSV, or XML file, ensuring that the contents of the file meets the following requirements:

Component	Requirements
File Name	Demand curve data must be in a file named **ACC_RDCI_<*pair identifier*>_Data**, where <*pair identifier*> matches the <*pair identifier*> in the demand curve configuration table and is unique within Accolade. For example, **ACC_RDCI_GlobalDemandCurves_Data**.
Rows	Each row contains a different demand curve, and each demand curve exists in only one row.
Column Names and Position	Column headings are in the first row of the worksheet. You can name columns as you see fit for your installation. The configuration table described below maps the columns in the data table to the respective fields in the Accolade database.
Data	The following columns and data are ***required*** for each demand curve: • **Curve Name** - A name that identifies the demand curve. Demand curve names are not required to be unique; however, for import they must be unique for a pool-model combination. • **Pool Name** - The resource pool to which the demand curve applies. Works in conjunction with the model identified in the Model Name column to identify the projects to which the curve applies. • **Model Name** - The model to which the demand curve applies. Works in conjunction with the pool identified in the Pool Name column to identify the projects to which the curve applies. Only one pool-model name combination can exist in the import per demand curve. • **Stage** - The demand value for each stage in the model. It might be most helpful to name these columns using the stage name within the model. Use the configuration file to map the stages to their appropriate stage number. • **Last Stage Duration** - If the model ends in a stage, the number of time periods in that stage. This data is not required for models ending in a gate. The following columns and data are ***optional*** for each demand curve: • **Active** - Enter **Yes, Y, True,** or **1** to activate the demand curve. Other values or an empty cell sets the status of the curve to **Inactive**. • **Delete Curve** - Enter **Yes, Y, True,** or **1** in a row to delete the demand curve. Any other values do not delete the curve.

Example



	A	B	C	D	E	F	G	H	I
1	**Demand Curve Name**	**Pool Name**	**Process Model Name**	**Discovery**	**Development**	**Validation**	**Launch**	**Active**	**Delete**
2	Low	R&D Senior Engineers	Phase Gate Process	0	0.5	0.25	0.25	Y	
3	Medium	R&D Senior Engineers	Phase Gate Process	0.1	0.25	0.3	0.3	Y	
4	High	R&D Senior Engineers	Phase Gate Process	0.5	0.75	0.5	0.5	Y	
5	Low	Manufacturing	Phase Gate Process	0.25	0.3	0.65	0.65	Y	
6	Medium	Manufacturing	Phase Gate Process	0.55	0.45	0.6	0.6	Y	
7	High	Manufacturing	Phase Gate Process	0.8	0.7	0.8	0.8	Y	
8	Average	Manufacturing	Phase Gate Process	0	0	0	0		Y
9									
10									
11									
12									

ResourceDemandCurvesImport_Data

Note the following in the example above:

- The file uses the stage names as column headings (Discovery, Development, Validation, etc), to more clearly identify the stages within the model.
- The import adds demand curves to the R&D Senior Engineers pool and the Manufacturing pool for projects based on the Phase Gate Process model.
- The Phase Gate Process model ends in a gate, therefore data to define the duration of the final stage (Last Stage Duration) is not included in the file.
- The import deletes the Average demand curve for the Manufacturing pool.

Map the Demand Curve Data to the Accolade Database (Config Table)

The configuration table contains the data mapping information so Accolade knows what type of data to expect in the accompanying data file. Administrators and Process Designers add the initial versions of the **ACC_RDCI** config table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the resource demand curve configuration table as a spreadsheet, CSV, or XML file, ensuring that the file meets the following requirements:

Component	Requirements
File Name	The demand curve configuration must be in a file named **ACC_RDCI_*<pair identifier>*_Config**, where the *<pair identifier>* matches the demand curve data file and is unique within Accolade. For example, **ACC_RDCI_GLobalDemandCurves_Config**.
Rows	Each row maps a column in the data table to a specific type of demand curve data.
Column Names and Position	The demand curve configuration must have the following column names, from left to right as listed below: • **Export Column Name** - Enter each column heading that is included in the data table. • **Metadata Name** - Enter the metadata name below that identifies the data included is in each column in the data table.

Component	Requirements
Accolade Metadata Names	Use the following metadata names in the **Metadata Name** column of the configuration table to map the data to the appropriate locations in the Accolade database: • DemandCurveName • PoolName • ProcessModelName • Stage01, Stage02...Stage20 If the model starts with a gate instead of a stage, the first stage in the model is Stage02, not Stage01, even though it appears as the first stage in the model. • LastStageDuration • Active • Delete

EXAMPLE Example

	A	B	C	D	E	F	G	H	I
1	**Demand Curve Name**	**Pool Name**	**Process Model Name**	**Discovery**	**Development**	**Validation**	**Launch**	**Active**	**Delete**
2	Low	R&D Senior Engineers	Phase Gate Process	0	0.5	0.25	0.25	Y	
3	Medium	R&D Senior Engineers	Phase Gate Process	0.1	0.25	0.3	0.3	Y	
4	High	R&D Senior Engineers	Phase Gate Process	0.5	0.75	0.5	0.5	Y	
5	Low	Manufacturing	Phase Gate Process	0.25	0.3	0.65	0.65	Y	
6	Medium	Manufacturing	Phase Gate Process	0.55	0.45	0.6	0.6	Y	
7	High	Manufacturing	Phase Gate Process	0.8	0.7	0.8	0.8	Y	
8	Average	Manufacturing	Phase Gate Process	0	0	0	0		Y
9									
10									
11									
12									

ResourceDemandCurvesImport_Data

Each column in the data table is listed in the **Export Column Name** column in the configuration table.

	A	B	C
1	**Export Column Name**	**Metadata Name**	
2	Demand Curve Name	DemandCurveName	
3	Pool Name	PoolName	
4	Process Model Name	ProcessModelName	
5	Discovery	Stage01	
6	Development	Stage02	
7	Validation	Stage03	
8	Launch	Stage04	
9	Active	Active	
10	Delete	Delete	
11			
12			
13			
14			

ResourceDemandCurvesImport_Conf

Note the following about the example above:

- Each item in the **Export Column Name** column is a column heading in the data table.
- The names listed in the **Metadata Name** column must match the metadata names listed above exactly for the import to process successfully. Enter the metadata name for each column that

matches, or maps, the data in the data file to the Accolade database to ensure the data is imported to the correct location.

- Stage names, such as Discovery, are mapped to their stage number in the **Metadata Name** column.

Notes:

- Modifying or deleting a curve does not change project demand. The curve must be reapplied to projects through the Resource Editor to update project demand.

Creating Resource Capacity Import Files

Import resource capacity data into Accolade using two specially named reference tables. The reference table pair begins with **ACC_RCI_**, which identifies the tables as reference tables that contain resource capacity import data.

Note: To import this type of data to Accolade, you must have the Process Designer, Project Importer, and Reference Table Manager user roles. To import a new version of an existing file, you must be assigned as the owner of the reference table file used for the import.

Create a File That Contains the Capacity Data to Import (Data Table)

The data table contains the data to import to Accolade. Administrators and Process Designers add the initial versions of the **ACC_RCI** data table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the resource capacity data table as a spreadsheet, CSV, or XML file, ensuring that the contents of the file meets the following requirements:

Important! Enter all dates in mm/dd/yyyy format, and enter the decimal separator in numbers as a period (.).

Component	Requirements
File Name	Resource capacity data must be in a file named **ACC_RCI_*<pair identifier>*_Data**, where *<pair identifier>* matches the *<pair identifier>* in the resource capacity configuration table and is unique within Accolade. For example, **ACC_RCI_GlobalCapacities_Data**.
Rows	Each row contains a different capacity, and each capacity exists in only one row.
Column Names and Position	Column headings are in the first row of the worksheet. You can name columns as you see fit for your installation. The configuration table described below maps the columns in the data table to the respective fields in the Accolade database.

Component	Requirements
Data	The following columns and data are ***required*** for each resource: • **Resource System Name** - Required if the resource is a **General** resource, such as Any Engineer. Use reporting to determine the system name of a **General** resource. • **Resource User Login** - Required if the resource type is **User**. • **First Time Period Start Date** - The first day of the time period to which the capacity applies. Enter the date in mm/dd/yyyy format. • **Last Time Period Start Date** - The last day of the time period to which to which the capacity applies. Enter the date in mm/dd/yyyy format. If there are data rows that contain capacities that overlap time periods, the last row in the import determines the capacity value for the resource for that time period. The following columns and data are ***optional*** for each resource: • **Resource Capacity Value** - The capacity value applied for the resource for all time periods including the first and last. Enter the decimal separate as a period (.), for example, 1.5. You cannot set capacities for resources that set to auto-calculate capacities for a request-only pool. If a capacity value is set to auto-calculate capacity for a request-only pool, the import fails. If the capacity value is **0**, the capacity displays empty in Accolade. • **Unavailable** - Values to subtract from capacity to account for temporary capacity reductions, for example, vacations.

EXAMPLE Example

	A	B	C	D	E	F
1	**Resource System Name**	**Resource User Login**	**First Period**	**Last Period**	**Capacity**	**Unavailable**
2		companyDomain\jane_doe	1/1/2020	12/31/2020	1	
3		companyDomain\jane_doe	7/1/2020	7/31/2020		1
4		companyDomain\john_doe	1/1/2020	12/31/2020	1	
5	AnySeniorEngineer		1/1/2020	12/31/2020	5	
6	AnySeniorEngineer		1/1/2020	12/31/2021	5	
7	AnyQAAnalyst		1/1/2020	6/30/2020	2	
8	AnyQAAnalyst		7/1/2020	12/31/2020	3	
9						
10						
11						
12						
13						
14						

ResourceCapacityImport_Data

Note the following in the example above:

- The import enters capacity in terms of full time employees (FTE). For example, Jane Doe is full time employee, therefore her capacity is 1 FTE.
- The import sets Jane Doe's capacity at 1 FTE from January 1, 2020 to December 31, 2020. However, Jane is unavailable in July of 2020, therefor her capacity is reduced from July 1 to July 31 by 1 FTE.
- The import sets the general resource AnySeniorEngineer to a capacity of 5 FTEs for 2020 and 2021.
- The import sets the general resource AnyQAAnalyst to 2 FTEs for the first half of 2020; however, the company is gaining a QA Analyst for the second half of 2020, so the capacity is set to 3 FTEs.

Map the Resource Capacity Data to the Accolade Database (Config Table)

The configuration table contains the data mapping information so Accolade knows what type of data to expect in the accompanying data file. Administrators and Process Designers add the initial versions of the **ACC_RCI** config table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the resource capacity configuration table as a spreadsheet, CSV, or XML file, ensuring that the file meets the following requirements:

Component	Requirements
File Name	The resource capacity configuration must be in a file named **ACC_RCI_*<pair identifier>*_Config**, where the *<pair identifier>* matches the resource capacity data file and is unique within Accolade. For example, **ACC_RCI_GlobalCapacities_Config**.
Rows	Each row maps a column in the data table to a specific type of capacity data.
Column Names and Position	The resource capacity configuration must have the following column names, from left to right as listed below: • **Export Column Name** - Enter each column heading that is included in the data table. • **Metadata Name** - Enter the metadata name below that identifies the data included is in each column in the data table.
Accolade Metadata Names	Use the following metadata names in the **Metadata Name** column of the configuration table to map the data to the appropriate locations in the Accolade database: • ResourceSystemName • ResourceUserLogin • FirstTimePeriodStartDate • LastTimePeriodStartDate • Capacity • Unavailable

EXAMPLE Example

	A	B	C	D	E	F
1	**Resource System Name**	**Resource User Login**	**First Period**	**Last Period**	**Capacity**	**Unavailable**
2		companyDomain\jane_doe	1/1/2020	12/31/2020	1	
3		companyDomain\jane_doe	7/1/2020	7/31/2020		1
4		companyDomain\john_doe	1/1/2020	12/31/2020	1	
5	AnySeniorEngineer		1/1/2020	12/31/2020	5	
6	AnySeniorEngineer		1/1/2020	12/31/2021	5	
7	AnyQAAnalyst		1/1/2020	6/30/2020	2	
8	AnyQAAnalyst		7/1/2020	12/31/2020	3	
9						
10						
11						
12						
13						
14						

ResourceCapacityImport_Data

Each column in the data table is listed in the **Export Column Name** column in the configuration table.

	A	B
1	**Export Column Name**	**Metadata Name**
2	Resource System Name	ResourceSystemName
3	Resource User Login	ResourceUserLogin
4	First Period	FirstTimePeriodStartDate
5	Last Period	LastTimePeriodStartDate
6	Capacity	Capacity
7	Unavailable	Unavailable
8		
9		
10		
11		
12		
13		
14		

ResourceCapacityImport_Config

Note the following about the example above:

- Each item in the **Export Column Name** column is a column heading in the data table.
- The names listed in the **Metadata Name** column must match the metadata names listed above exactly for the import to process successfully. Enter the metadata name for each column that matches, or maps, the data in the data file to the Accolade database to ensure the data is imported to the correct location.

Notes:

- As capacity values change for resources within pools that are set to auto-calculate capacity, the pool's total capacity is recalculated accordingly.

Creating Resource Demand Import Files

Import resource demand data into Accolade and apply the demands to projects using two specially named reference tables. The reference table pair begins with **ACC_RDI_**, which identifies the tables as reference tables that contain resource demand import data. This import creates demands, but does not import the demand values.

Note: To import this type of data to Accolade, you must have the Process Designer, Project Importer, and Reference Table Manager user roles. To import a new version of an existing file, you must be assigned as the owner of the reference table file used for the import.

Create a File That Contains the Demand Data to Import (Data Table)

The data table contains the data to import to Accolade. Administrators and Process Designers add the initial versions of the **ACC_RDI** data table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the resource demands data table as a spreadsheet, CSV, or XML file, ensuring that the contents of the file meets the following requirements:

Component	Requirements
File Name	Resource demand data must be in a file named **ACC_RDI_<*pair identifier*>_Data**, where <*pair identifier*> matches the <*pair identifier*> in the resource demand configuration table and is unique within Accolade. For example, **ACC_RDI_NorthAmericaDemands_Data**.
Rows	Each row contains a different demand, and each demand exists in only one row.
Column Names and Position	Column headings are in the first row of the worksheet. You can name columns as you see fit for your installation. The configuration table described below maps the columns in the data table to the respective fields in the Accolade database.
Data	The following columns and data are ***required*** for each resource demand: • **Project ID** or **Identifier String Metric Name** - Either the Project ID that displays on the project header, or a string metric that identifies the project to which the demand applies. Only one is required to identify the project. **Important!** If the **Auto-Generate Project IDs** system parameter is enabled, use a string metric as the identifier for the project. • **Process Model Name** - Depending on your system's configuration, this may display on the **Project Home** page. Used to further identify

Component	Requirements
	the projects to which the demand applies. • **Resource System Name** - The unique, shorter name that identifies the resource in queries, reporting views, Accolade Office Extensions, and field codes. Required if the demand is for a General resource. • **Resource User Login** - The domain and Accolade login of the specific user for the demand. Required if resource type is **User** and not a **General** resource. • **Demand System Name** - For the initial import, enter a system name to identify a demand row in Resource Editor. A demand row contains all the demands for a single resource in a specific project. The following columns and data are ***optional*** for each resource demand: • **Demand Curve Name** - The demand curve name, if the demand is part of a demand curve. Only required if the resource is part of a pool that contains a demand curve assigned to the model. • **Demand Multiplier** - The amount by which the demand curve is multiplied for the demand. Defaults to **1**. • **Demand Curve Effective Date** - The first time period that the demands in a demand curve are applied. Defaults to the current time period. • **Demand Phase ID** - The sequence number of the stage in the model to which the demand applies. If the model starts with a gate instead of a stage, the first stage in the model is Stage02, not Stage01, even though it appears as the first stage in the model. • **Demand Type** - **Requested** or **Assigned**. Defaults to **Assigned**. If an assigned demand is associated with a request demand, you cannot change the demand type through an import. • **Delete** - Enter **Yes, Y, True,** or **1** to delete the demand. Any other values do not delete the demand.

EXAMPLE Example

	B	C	D	E	F	G	H	I	J	K
1	Process Model Name	Demand System Name	Resource System Name	Resource User Login	Demand Type	Demand Curve Name	Curve Multiplier	Effective Date	Stage	Delete
2	Phase Gate Process	AnySeniorEngineer	AnySeniorEngineer		Requested	Low	1	11/1/2020	2	
3	Phase Gate Process			companyDomain\john_smith		Medium	0.9	1/2/2021	3	
4	Phase Gate Process			companyDomain\jane_doe		High	1	6/3/2020	4	
5	Phase Gate Process	AnyQAnalyst	AnyQAAnalyst		Requested					
6										
7										
8										
9										
10										
11										
12										

ResourceDemandsImport_Data

Note the following in the example above:

- Projects in this import are identified using the combination of the project code and process model name, not by an identifier string metric.
- Demands that are part of a demand curve apply to stages in the process model as defined in the Phase ID column.
- Demands are added to Project A for Any Senior Engineer and for John Smith.
- Demands are added to Project B for Any QA Analyst and for Jane Doe.
- No demands are deleted in this import.

Map the Resource Demand Data to the Accolade Database (Config Table)

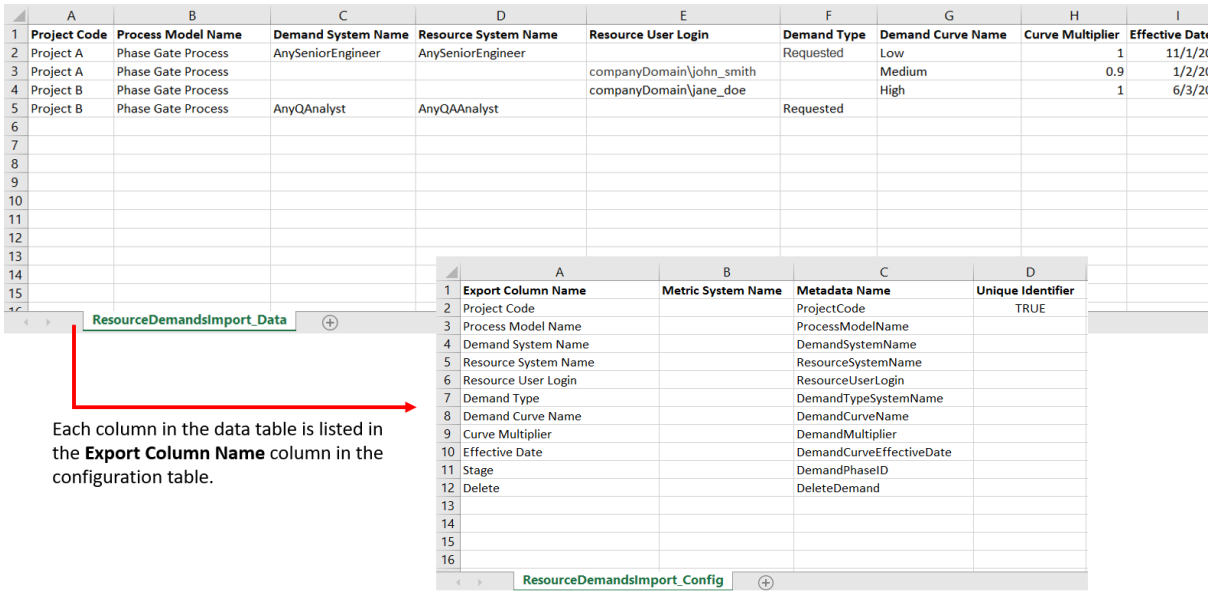
The configuration table contains the data mapping information so Accolade knows what type of data to expect in the accompanying data file. Administrators and Process Designers add the initial versions of the **ACC_RDI** config table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the resource demands configuration table as a spreadsheet, CSV, or XML file, ensuring that the file meets the following requirements:

Component	Requirements
File Name	The resource demand configuration must be in a file named **ACC_RDI_*<pair identifier>*_Config**, where the *<pair identifier>* matches the resource demand data file and is unique within Accolade. For example, **ACC_RDI_NorthAmericaDemands_Config**.
Rows	Each row maps a column in the data table to a specific type of resource demand data.
Column Names and Position	The resource demand configuration must have the following column names, from left to right as listed below: • **Export Column Name** - Enter each column heading that is included in the data table. • **Metric System Name** - If a string metric is used to identify the projects in the data file, enter the system name of the metric used to identify the projects. If the Project ID is used to identify projects, include this column but leave it blank. • **Metadata Name** - Enter the metadata name below that identifies the data included is in each column in the data table. • **Unique Identifier** - Enter **TRUE** in the row with the **Export Column Name** that uniquely identifies the projects within the import. Can be set for either the **ProjectCode** or a string metric. The import uses what you identify to match the demands to their respective projects.
Accolade Metadata Names	Use the following metadata names in the **Metadata Name** column of the configuration table to map the data to the appropriate locations in the Accolade database:

Component	Requirements
	• ProjectCode • ProcessModelName • DemandSystemName • ResourceSystemName • ResourceUserLogin • DemandTypeSystemName • DemandCurveName • DemandMultiplier • DemandCurveEffectiveDate • DemandPhaseID • DeleteDemand

EXAMPLE Example



	A	B	C	D	E	F	G	H	I
1	**Project Code**	**Process Model Name**	**Demand System Name**	**Resource System Name**	**Resource User Login**	**Demand Type**	**Demand Curve Name**	**Curve Multiplier**	**Effective Dat**
2	Project A	Phase Gate Process	AnySeniorEngineer	AnySeniorEngineer		Requested	Low	1	11/1/2(
3	Project A	Phase Gate Process			companyDomain\john_smith		Medium	0.9	1/2/2(
4	Project B	Phase Gate Process			companyDomain\jane_doe		High	1	6/3/2(
5	Project B	Phase Gate Process	AnyQAAnalyst	AnyQAAnalyst		Requested			

ResourceDemandsImport_Data

Each column in the data table is listed in the **Export Column Name** column in the configuration table.

	A	B	C	D
1	**Export Column Name**	**Metric System Name**	**Metadata Name**	**Unique Identifier**
2	Project Code		ProjectCode	TRUE
3	Process Model Name		ProcessModelName	
4	Demand System Name		DemandSystemName	
5	Resource System Name		ResourceSystemName	
6	Resource User Login		ResourceUserLogin	
7	Demand Type		DemandTypeSystemName	
8	Demand Curve Name		DemandCurveName	
9	Curve Multiplier		DemandMultiplier	
10	Effective Date		DemandCurveEffectiveDate	
11	Stage		DemandPhaseID	
12	Delete		DeleteDemand	

ResourceDemandsImport_Config

Note the following about the example above:

- Each item in the **Export Column Name** column is a column heading in the data table.
- The **Metric System Name** must match the metric system name exactly for the value to update.
- The names listed in the **Metadata Name** column must match the metadata names listed above exactly for the import to process successfully. Enter the metadata name for each column that matches, or maps, the data in the data file to the Accolade database to ensure the data is imported to the correct location.

- The **TRUE** setting in the **Unique Identifier** column indicates that project code is used to uniquely identify projects to which the demands apply.

Creating Resource Demand Value Import Files

Import resource demand values data into Accolade using two specially named reference tables. The reference table pair begins with **ACC_RDVI_**, which identifies the tables as reference tables that contain demand value import data.

Note: To import this type of data to Accolade, you must have the Process Designer, Project Importer, and Reference Table Manager user roles. To import a new version of an existing file, you must be assigned as the owner of the reference table file used for the import.

Create a File That Contains the Demand Value Data to Import (Data Table)

The data table contains the data to import to Accolade. Administrators and Process Designers add the initial versions of the **ACC_RDVI** data table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the resource demand values data table as a spreadsheet, CSV, or XML file, ensuring that the contents of the file meets the following requirements:

Important! Enter all dates in mm/dd/yyyy format, and enter the decimal separator in numbers as a period (.).

Component	Requirements
File Name	Resource demand value data must be in a file named **ACC_RDVI_ *<pair identifier>*_Data**, where *<pair identifier>* matches the *<pair identifier>* in the demand value configuration table and is unique within Accolade. For example, **ACC_RDVI_ResourceDemandValues_Data**.
Rows	Each row contains a different demand values, and each demand value exists in only one row.
Column Names and Position	Column headings are in the first row of the worksheet. You can name columns as you see fit for your installation. The configuration table described below maps the columns in the data table to the respective fields in the Accolade database.
Data	The following columns and data are ***required*** for each demand value: • **Project ID** or **Identifier String Metric Name** - Either the **Project ID** that displays on the project header, or a string metric that identifies the project to which the demand value applies. Only one is required to identify the project. **Important!** If the **Auto-Generate Project IDs** system parameter is enabled, use a string metric as the identifier for the project. • **Process Model Name** - Depending on your system's configuration,

Component	Requirements
	this may display on the **Project Home** page. Used to further identify the projects to which the demand value applies. • **Resource System Name** - The unique, shorter name that identifies the resource in queries, reporting views, and field codes. Required if the demand is for a General resource. • **Demand System Name** - Required if a **Resource System Name** is provided. If the field is blank or missing, the import generates a demand system name automatically. To edit existing values for a demand whose system name you do not know, query the database using the Accolade Office Extensions add-in. • **First Time Period Start Date** - The first time period to which the demand value applies. Enter the date in mm/dd/yyyy format. • **Last Time Period Start Date** - The last time period to which the demand value applies. Enter the date in mm/dd/yyyy format. The import loads to Accolade in the order the rows appear in the Data table. If the Data table contains resource demand values for overlapping time periods, the last row in the table for the time period is the row that displays in Accolade after the import. • **Demand Value** - The value to apply to all time periods between and including the dates specified in the **First Time Period Start Date** and the **Last Time Period Start Date** columns. The value can be between 0 and 100,000. If the field is blank, no value is added for the demand. If the value is 0, the demand is updated to a value of zero. Enter the decimal separator in numbers as a period (.). The following columns and data are ***optional*** for each demand value: • **Demand Type** - **Requested** or **Assigned**. If the field is blank or missing, the demand type is set to **Assigned**. If an assigned demand is associated with a requested demand, you cannot change the demand type for either request through an import.

EXAMPLE Example

	A	B	C	D	E	F	G	H
1	**Project Code**	**Process Model Name**	**Demand Type**	**Demand System Name**	**Resource System Name**	**First Time Period**	**Last Time Period**	**Demand Value**
2	Project A	Phase Gate Process	Assigned	75AAA-B777-4D40-A914-4D118D11AA	AA4DF334-E578-850-70780D6F8ABE	10/1/2020	4/30/2021	5
3	Project B	Phase Gate Process	Assigned	75BBB-B777-4D40-A914-4D118D11AA	BB4DF334-E578-850-70780D6F8ABE	10/5/2020	4/30/2021	4
4	Project C	Phase Gate Process	Assigned	75CCC-B777-4D40-A914-4D118D11AA	CC4DF334-E578-850-70780D6F8ABE	10/15/2020	4/30/2021	2
5								
6								
7								
8								
9								

ResourceDemandValuesImport_Data

Note the following in the example above:

- Projects in this import are identified using the combination of the project code and process model name, not by an identifier string metric.
- The import assigns demands to Project A, B, and C, for varying dates.

Map the Resource Demand Value Data to the Accolade Database (Config Table)

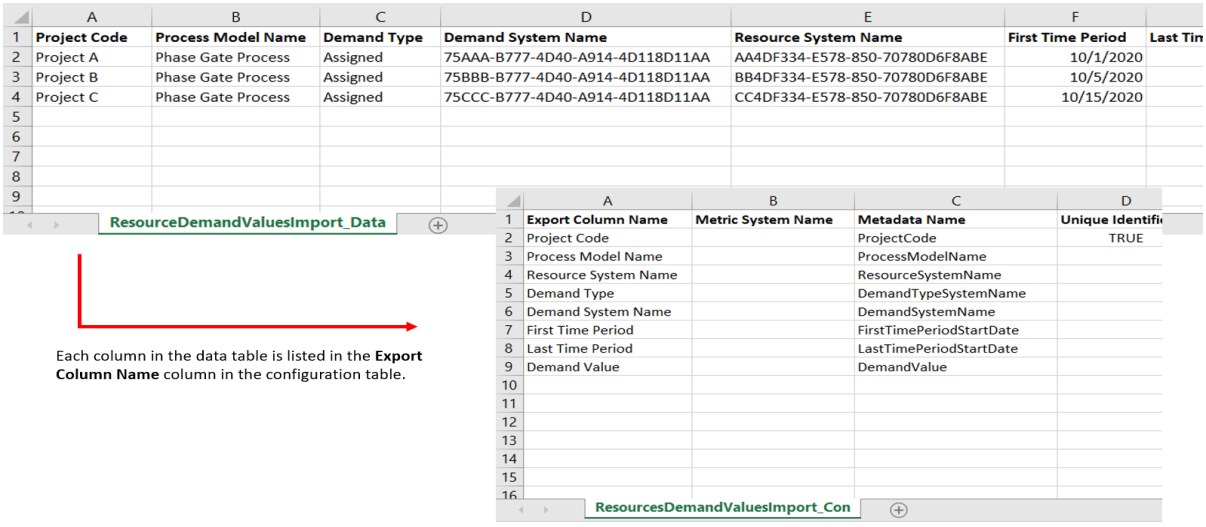
The configuration table contains the data mapping information so Accolade knows what type of data to expect in the accompanying data file. Administrators and Process Designers add the initial versions of the **ACC_RDVI** config table. The assigned table owner, which requires the Reference Table Manager user role, can add new versions of the table as needed.

Create the resource demand values configuration table as a spreadsheet, CSV, or XML file, ensuring that the file meets the following requirements:

Component	Requirements
File Name	The resource demand value configuration must be in a file named **ACC_RDVI_<*pair identifier*>_Config**, where the <*pair identifier*> matches the demand value data file and is unique within Accolade. For example, **ACC_RDVI_ResourceDemandValues_Config**.
Rows	Each row maps a column in the data table to a specific type of demand value data.
Column Names and Position	The demand value configuration must have the following column names, from left to right as listed below: • **Export Column Name** - Enter each column heading that is included in the data table. • **Metric System Name** - If a string metric is used to identify the projects in the data file, enter the system name of the metric used to identify the projects. If the Project ID is used to identify projects, include this column but leave it blank. • **Metadata Name** - Enter the metadata name below that identifies the data included is in each column in the data table. • **Unique Identifier** - Enter **TRUE** in the row with the **Export Column Name** that uniquely identifies the projects within the import. Can be set for either the **ProjectCode** or a string metric. The import uses what you identify to match the demand values to their respective projects.
Accolade Metadata Names	Use the following metadata names in the **Metadata Name** column of the configuration table to map the data to the appropriate locations in the Accolade database: • ProjectCode • ProcessModelName • ResourceSystemName • DemandTypeSystemName • DemandSystemName

Component	Requirements
	• FirstTimePeriodStartDate • LastTimePeriodStartDate • DemandValue

EXAMPLE Example



	A	B	C	D	E	F
1	**Project Code**	**Process Model Name**	**Demand Type**	**Demand System Name**	**Resource System Name**	**First Time Period**
2	Project A	Phase Gate Process	Assigned	75AAA-B777-4D40-A914-4D118D11AA	AA4DF334-E578-850-70780D6F8ABE	10/1/2020
3	Project B	Phase Gate Process	Assigned	75BBB-B777-4D40-A914-4D118D11AA	BB4DF334-E578-850-70780D6F8ABE	10/5/2020
4	Project C	Phase Gate Process	Assigned	75CCC-B777-4D40-A914-4D118D11AA	CC4DF334-E578-850-70780D6F8ABE	10/15/2020

ResourceDemandValuesImport_Data

Each column in the data table is listed in the **Export Column Name** column in the configuration table.

	A	B	C	D
1	**Export Column Name**	**Metric System Name**	**Metadata Name**	**Unique Identifi**
2	Project Code		ProjectCode	TRUE
3	Process Model Name		ProcessModelName	
4	Resource System Name		ResourceSystemName	
5	Demand Type		DemandTypeSystemName	
6	Demand System Name		DemandSystemName	
7	First Time Period		FirstTimePeriodStartDate	
8	Last Time Period		LastTimePeriodStartDate	
9	Demand Value		DemandValue	

ResourcesDemandValuesImport_Con

Note the following about the example above:

- Each item in the **Export Column Name** column is a column heading in the data table.
- The **Metric System Name** must match the metric system name exactly for the value to update.
- The names listed in the **Metadata Name** column must match the metadata names listed above exactly for the import to process successfully. Enter the metadata name for each column that matches, or maps, the data in the data file to the Accolade database to ensure the data is imported to the correct location.
- The **TRUE** setting in the **Unique Identifier** column indicates that project code is used to uniquely identify projects to which the demand values apply.

Validating and Running Data File Imports

Process Designers can validate files before importing the data to Accolade to view potential errors and a summary of project changes that would occur on import.

Note: Prior to importing project data to Accolade, ensure that all configuration components referenced in the data tables already exist in the database. The data import process does not create configuration components.

To validate a data import:

1. add both reference tables to Accolade, using the file names as the reference table names.
2. From the **Workspace** menu, select **Import > Data**.

   To narrow the list by importer name or type, add the criteria to filter by in the appropriate filter text box.
3. In the **Actions** column, click **Validate** next to the importer file containing the reference table pair.

   Click the import name in the **Name** column to view the data reference table.

   Validating an import returns a list of potential errors and outcomes that will occur on import. It does not import data or commit changes back to Accolade. The first five numbered rows detail the data that will be imported if run. These include the number of data types created, updated, and deleted, and the number of invalid rows and total rows if the data is imported. If the reference tables fail, the validate results will not display the project outcomes. Print validation results from the dialog by clicking **Print**.

To import data to Accolade:

1. add both reference tables to Accolade, using the file names as the reference table names.

   Administrators and Process Designers add the initial versions of the reference table pairs. From there, the table owner can update the existing files as needed. The file name prefix identifies the tables as reference tables that contain import data.
2. From the **Workspace** menu, select **Import > Data**.
3. In the **Actions** column, click **Run** next to the importer file containing the reference table pair.

   Click the import name in the **Name** column to view the data reference table.
4. *(Optional)* Click  in the **Status/Result** column to access last run results and history including errors, warnings, and abort and cancel information. Correct the errors or warnings to import the data successfully. Print import results and action history from the dialog by clicking **Print**.

Note: After the initial import, updated data and configuration mappings can be uploaded using the method above, replacing the existing files, or using automatic loading. If you have a large amount of data to import, Sopheon recommends using a reference table

import via automatic loading to perform the import after hours, as large imports can result in system slowness while the import is running.

Canceling and Aborting Imports

Once you've selected to run the import, you can cancel or abort at any time during the import process. This will terminate the import and stop project data from being imported into Accolade.

To cancel or abort an import:

1. From the **Workspace** menu, select **Import > Data**.
2. Click **Abort** to stop and roll back a currently running import, or to remove a pending import.

Note: Only one project import can run at a time. Each import runs in the order it was selected to run either manually or by an automatic reference table upload. The Autoloader Service that uploads reference tables automatically adds files to the import queue based on the time selected for them to upload. Additionally, the Autoloader service only adds one set of import files at a time and will not add additional files into the queue until the import has completed. However, you can add import files to the queue manually when the Autoloader service has automatically added a file. Use cancel and abort functions to remove imports from the lineup.

Troubleshooting Project Data Imports

Ensure the data you are entering in the data table and the config table is accurate. An import can fail as a whole, or only single lines within the data table can fail to import.

If the import fails to run for any lines in the data table, verify that your data and config tables meet the following requirements:

- The config table has one row identified as **TRUE** or **YES** in the **Unique Identifier** column.
- If you are using **Last Gate Number** or **Last Gate Decision Name**, confirm that either they both exist or both do not exist.
- All **Metric System Names** that are specified in the config table exist in the data table, even if they contain no data in their columns.

If the import runs, but you receive errors about some lines not being imported, verify that the data and config tables meet the following requirements:

- The **Unique Identifier** is truly unique. For example, if the config table sets the **Project Code** as the **Unique Identifier**, and the data table contains two projects named Project A, the second Project A does not import.
- All required metrics for a process model are included in each project creation row in the data table. If a required metric is missing, the project fails to import.

- All required metadata fields for a process model, such as **Start Date** or **End Date**, are included in each project creation row in the data table.
- The **Project Code** is included in each row.
- The date in the **GateDateXX** column for a project in the data table is a gate number within the process model. For example, if the process model has 5 gates, and the data table contains data in a **GateDate06** column, the import fails because that process model does not contain a 6th gate.
- Accolade expects project codes to be imported as strings. If the project codes are numbers, or the **Auto-Generate Project IDs** system parameter is enabled (which creates numeric project codes), you must first upload a data table following the format described above that contains a single row of data with a project code entered as a string. After uploading the data table with a single row, upload subsequent versions of the data table that contain the true project data information.
- Project links imported using the **Date - Date** link type are set to the default settings for the link type if the project does not contain the date information entered for the From and To dates.

Importing Related Documents

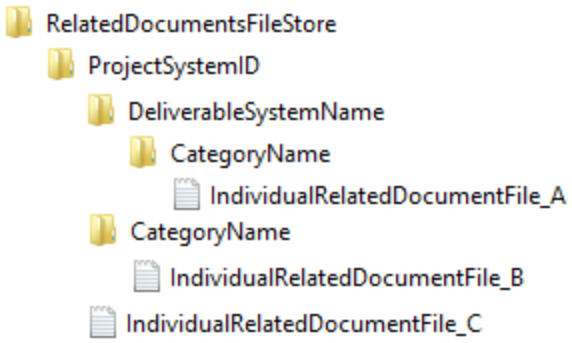
As you implement Accolade, you may have a large number of related document files from your previous innovation management system or process that you want to include with projects and initiatives when you start using Accolade. Use a filestore and the Accolade Autoloader Service to upload the related documents efficiently to Accolade.

Note: Users with the Project Importer user role can import related documents. However, an Administrator must enable automatic uploading of files within Accolade.

Related Project Filestore Folder Structure

Accolade uploads related documents using a specific directory structure defined within a drop box location. The drop box location used to store the related documents prior to uploading can be anywhere on your network that Accolade has access to. The Autoloader Service uses the folder structure within the directory to match the related documents to the correct projects in Accolade. Administrators tell Accolade where to find the files in the Autoloader Configuration section of the Accolade Administration Console.

Define the folder structure as follows, where **IndividualRelatedDocumentFile** is one or more files to upload as related documents:



Keep the following in mind as you are creating and copying files to the filestore location:

- The filestore cannot be an FTP site.
- The highest folder in the structure must be named **RelatedDocumentsFileStore** so the Autoloader Service recognizes the contents of the directory as a related documents upload.
- The filestore can have multiple **ProjectSystemID** folders representing different projects within Accolade.
- Each **ProjectSystemID** folder can have multiple **DeliverableSystemName** folders representing different deliverables within a project.
- Individual files added to the filestore are uploaded as follows:
  - Files in the **ProjectSystemID** are uploaded as project-level related documents with the **Default** category assignment.
  - Files in the **ProjectSystemID > DeliverableSystemName** are uploaded as deliverable-level related documents and are attached to the deliverable identified in the folder name.
  - Files in a category folder, either within the **DeliverableSystemName** folder or in the **ProjectSystemID** folder are uploaded and assigned that category.
  - If a file already exists, a new version is created on upload. If multiple files exist with the same name in the same location, the individual files do not upload.

Important! Ensure files you copy to the filestore directory are not set to Read Only and ensure that the file names, including extensions, do not exceed 64 characters. Files with a name longer than 64 characters do not upload.

Files uploaded using the filestore contain the following details after successful upload to Accolade:

- The related document name is the individual file name as it existed in the filestore directory.
- The uploaded by name is the Service Account used for the upload service.
- The author is the Microsoft Windows author of the file.

Uploading the Related Documents Filestore to Accolade

Consider the following prior to uploading related documents:

- If there is a large number of files to upload, consider running the upload after hours.
- If there is active training or other activity happening within Accolade, consider running the upload after the activity has ended.
- Files larger than the limit set in the **Maximum File upload Size** system parameter do not upload.

To upload the related document in the filestore:

1. Ensure files are not set as Read Only and are in the correct directory in the filestore as indicated above, and ensure that the filestore is in the drop box location defined in the Administration Console.
2. From the **Workspace** menu, select **Import > Related Documents**.
3. Do one of the following:
   - **To schedule the upload to run within the next 24 hours** - In the **Scheduled at** fields, enter the time within the next 24 hours to upload the files in the filestore location, and click **Apply**. For example, to upload all files at 10:00 pm, enter **22:00**. The time entered is the time in the application server's time zone, which may be different than the time zone you are in when entering the time in Accolade. After the upload runs, the time is cleared, indicating that an upload is not scheduled.
   - **To run the upload to run immediately** - Select **Now** and click **Apply**.

   After a file is processed and uploaded, it is removed from the import directory. A log file is generated and saved in the Outbox Path as defined in the Administration Console that contains the number of files successfully uploaded and information about any files that encountered an error during the upload process.

Sopheon Corporation

6870 West 52nd Avenue, Suite 215

Arvada, CO 80002