



Version 0.1 August | 2019

Ed-Fi to Generate

ETL Plug-In Implementation Guide

aemcorp.com/education

Contents

[Introduction 3](#_Toc25758898)

[Documentation 3](#_Toc25758899)

[Installation 4](#_Toc25758900)

[Installation Instructions – Batch Files 5](#_Toc25758901)

[Installation Instructions – Incremental 5](#_Toc25758902)

[Configuration 6](#_Toc25758903)

[Configuring Generate 6](#_Toc25758904)

[Configuring Enumerations 6](#_Toc25758905)

[Configuring Stored Procedures 7](#_Toc25758906)

[Ed-Fi MetaEd Extension 8](#_Toc25758907)

[Implementation & Validation 9](#_Toc25758908)

[Supporting Documentation 10](#_Toc25758909)

[Appendix A – Migration Script Descriptions 11](#_Toc25758910)

|  |  |  |
| --- | --- | --- |
| **Version History** | | |
| **Version #** | **Author(s)** | **Version Date** |
| 1.0 | Duane Brown, Nathan Clinton | 8/30/2019 |
|  |  |  |
|  |  |  |

# Introduction

The purpose of this document is to provide instructions and documentation for configuring and implementing the Ed-Fi to Generate ETL Plug-In. The Plug-In is designed to work with the Ed-Fi 3.x version Operational Data Store (ODS) and was created for the 2018-2019 ED*Facts* reporting period. It encompasses every ED*Facts* file specification for the 2018-2019 school year. Subsequent changes to file specifications in future years that do not involve new data elements or option sets are handled by Generate and do not require updates to the Ed-Fi to Generate Plug-In. Changes to the file specifications that require new data elements and/or options sets will require updates to the Plug-In. The Ed-Fi to Generate Plug-In is available in the Ed-Fi Exchange located here: <https://exchange.ed-fi.org/>.

This document is divided into 5 sections:

* **Documentation** – all documents and scripts related to this installation
* **Installation** – setting up the code necessary to migrate data from the Ed-Fi ODS to the Generate ODS
* **Configuration** – customization of the enumerations specific to your state
* **Ed-Fi MetaEd Extension** – extending the Ed-Fi 3.x ODS to include additional elements required for ED*Facts* reporting
* **Implementation & Validation** – executing the ETLs and discovering any issues in the process
* **Supporting Documentation** – additional resources and information about the Ed-Fi to Generate ETL Plug-In

# Documentation

The installation materials reside in two locations:

1. The [Ed-Fi GitHub location](https://github.com/Ed-Fi-Alliance/Ed-Fi-X-ETL-To-Generate) set aside for this project specifically. Please check the repository for recent updates.

(!) Requires that you have a GitHub account to access the code. If you do not have access to the Ed-Fi GitHub repositories, please [submit an access request](https://license.ed-fi.org/access-request-form/).

1. Sign-in to [GitHub](https://github.com/) if you are not currently logged in. Navigate to <https://github.com/Ed-Fi-Alliance/Ed-Fi-X-ETL-To-Generate> repository to access the documentation and the migration scripts.

These sites contain the SQL scripts and documentation necessary to install and configure the Plug-In. It contains the following File Folders and Resources:

* **Documentation**
  + **Ed-Fi to Generate Plug-In Implementation Guide v0.1** – This document that details out the installation and configuration requirements.
  + **ETL Checklists** – A complete crosswalk of all applicable Ed-Fi 3.x Table/ Column/ Enumerations to applicable CEDS Table/ Column/ Enumerations required for populating the complete list of ED*Facts* reports.
* **Functions** – All of the functions that need to be added to Generate to execute the Stored Procedures
* **ODS**
  + **App.DataMigrationTasks\_CreateTasks\_EdFi** – Script that populates the App.DataMigrationTasks table with the list of stored procedures needed to run the migration from Ed-Fi to Generate prior to extending the model (Only Core Ed-Fi E.x).
  + **App.DataMigrationTasks\_CreateTasks\_EdFi\_MetaEdExtension** – Script that populates the App.DataMigrationTasks table with the list of stored procedures needed to run the migration from Ed-Fi to Generate that also includes executing the stored procedures after Ed-Fi has been extended to include elements missing in Ed-Fi 3.x that are required for ED*Facts* reporting (Core Ed-Fi 3.x Plus Extensions).
  + **ODS.SourceSystemReferenceData** – This script creates the table necessary to house the enumeration mapping from Ed-Fi enumerations to CEDS enumerations..
  + **ODS.SourceSystemReferenceData.Data** – This script populates the ODS.SourceSystemReferenceData table with the standard Ed-Fi enumerations to CEDS. This can be modified to reflect your own internal enumerations in use for Ed-Fi.
* **Package**
  + **1\_EdFi to Generate Plug-In** – This script can be used in lieu of the Functions, ODS, Staging, Stored Procedures Encapsulated, and Stored Procedures Staging scripts as it contains all of the scripts in a single executable statement. This will stand up everything necessary to migrate the data from the Ed-Fi 3.x Core model (without extensions and with Ed-Fi default enumerations) to Generate.
  + **2\_EdFi to Generate Plug-In\_MetaEd EdFi ODS Extension** – This script extends the Ed-Fi ODS to include additional tables and elements necessary to populate Generate.
  + **3\_EdFi to Generate Plug-In\_MetaEd Generate Update** – This script adds in the additional stored procedures in the Stored Procedures Staging folder that reference MetaEd and prepares Generate to migrate data that includes those tables.
* **Staging** – Contains scripts that build the Staging tables in Generate.
* **Stored Procedures Encapsulated** – Contains the ETL scripts that move the data from the Staging Tables to the CEDS ODS tables in Generate.
* **Stored Procedures Staging** – Contains the ETL scripts that move the data from the Ed-Fi ODS to the Staging Tables (contains both the Ed-Fi 3.x Core model scripts and scripts for the MetaEd Extended tables).

# Installation

Installation of the Ed-Fi to Generate Plug-In can be done with the execution of one SQL statement + the Ed-Fi extensions provided in the package or it can be done incrementally by following the steps below. In both instances, there must be an existing Ed-Fi ODS and an existing Generate database called “generate”. The installation instructions below only impact the Generate database configuration. There is an additional step that extends the Ed-Fi ODS to account for data elements that do not exist in version 3.x of Ed-Fi. These instructions are provided in their own section.

## Installation Instructions – Batch Files

There are two batch files that will run the SQL files in the proper order. One file runs all scripts, including those for all Ed-Fi extension tables, the other excludes extension tables. You may need to open the batch file and update the server name if you are running the code remotely. To do so, open the file in a text editor and find the text “(local)” and replace it with the fully qualified server name and SQL instance (e.g. “SERVER\_NAME[\INSTANCE\_NAME]”).

Batch File Including Extension Tables: “run-all-scripts.bat”

Batch File Excluding Extension Tables: “run-all-scripts-excluding-extensions.bat

## Installation Instructions – Incremental

There are several file folders provided in addition to the “Package” folder. These have been combined to produce the Single SQL Statement “1\_EdFi to Generate Plug-In”. Instead of executing that single statement, you can implement them individually. This provides a means for looking through the scripts based on their purpose and adjusting according to your need. This will be discussed further in the configuration section as well. To install these incrementally, run them in the following order on the server which houses the generate database.

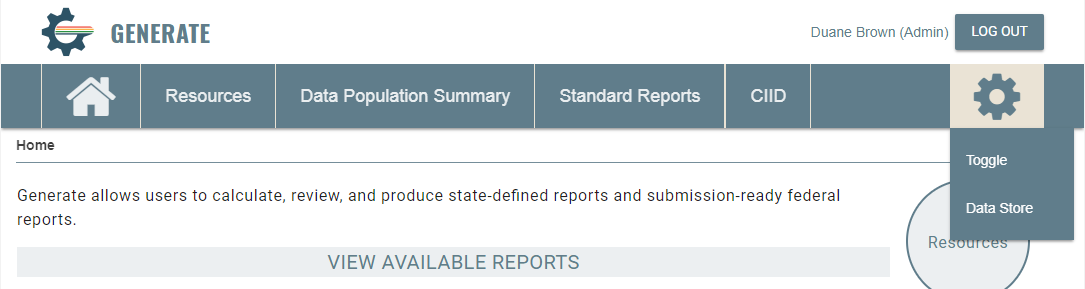
1. Ensure the generate database is named “generate”.
2. Functions Folder – First execute each of the functions provided in the file folder (Note: the order of execution does not matter).
3. Staging Folder – First execute the statement entitled “Staging Schema”. After this execute the remaining statements in any order.
4. ODS Folder – First execute the statement entitled “ODS.SourceSystemReferenceData”. After that, execute the statement entitled “ODS.SourceSystemReferenceData.Data”.  **DO NOT** execute the App.DataMigrationTasks\_CreateTasks\_Ed-Fi or the App.DataMigrationTasks\_CreateTasks\_EdFi\_MetaEdExtension. These will be run at the end to set up the sort order for the data migration.
5. Stored Procedures Encapsulated Folder – Execute all the stored procedures in the folder being certain to run the App.Migrate\_Data\_Validation first, then the rest in any other order.
6. Stored Procedures Staging Folder – Do a Find and Replace. The SQL statements will have several instances of the path “[EdFi\_Sample\_ODS\_Data\_For\_Generate].[edfi]” pointing to a test instance of the Ed-Fi ODS. Replace this with your path to the Ed-Fi ODS. Note that the path assumes the Ed-Fi ODS to be on the same server as Generate as it does not include the server name. If they are on a different server, you will need to add the server name at the beginning of the path. Execute all of the stored procedures in any order (Note: If the Ed-Fi ODS is not extended, do not execute the stored procedures with the “\_MetaEd” extension; if the Ed-Fi ODS is extended with the exact table structure provided in this package, execute the additional scripts as well).
7. ODS Folder – Now run either the App.DataMigrationTasks\_CreateTasks\_Ed-Fi (if the Ed-Fi ODS is not extended) or the App.DataMigrationTasks\_CreateTasks\_EdFi\_MetaEdExtension (if the Ed-Fi ODS is extended)
8. You can now move on to the Configuration section.

# Configuration

The Ed-Fi to Generate Plug-In is based on the core Ed-Fi ODS 3.x model with the standard Ed-Fi enumerations provided. The Plug-In is extremely flexible as it relates to the adjustment of the enumerations and the location of data. The instructions below provide information for configuring Generate, your states custom enumerations, and any modification to the table structure for where data are provided in your instance of the Ed-Fi ODS.

## Configuring Generate

Generate contains a section of questions that are very similar to the EMAPS questions states fill out prior to their yearly ED*Facts* reporting. These questions indicate metadata about their state that is unique to how they will report the data. Generate uses this information to produce the ED*Facts* reports based on how the state is configured. The Ed-Fi to Generate Plug-In also takes advantage of some of this pre-filled data when determining which data to use in the Ed-Fi ODS for certain reports. The “Toggle” section of Generate will need to be filled out prior to executing the migration of data from Ed-Fi to Generate. The screen shot below shows the location of the “Toggle” section after logging in as an administrator.



## Configuring Enumerations

**Source System Reference Data**

The Ed-Fi to Generate Plug-In comes with the standard Ed-Fi enumerations already mapped to the CEDS enumerations used in Generate. This information can be found in the script entitled ODS.SourceSystemReferenceData.Data and can also be found when querying the ODS.SourceSystemReferenceData table in the generate database. For each row, 5 columns are defined:

1. **SchoolYear** – The year for which this enumeration is valid. This allows for tracking variations in enumerations from year to year. Each year, a new set of enumerations needs to be copied applicable for the school year. The Stored Procedures each have a parameter for school year (@SchoolYear) which will refer to the applicable school year for that enumeration.
2. **TableName** – This is the name of the CEDS enumeration reference table. This column should NOT be adjusted.
3. **TableFilter** – At times, CEDS enumerations are based on a second set of enumerations. For instance, an Organization Identification System may vary depending on whether or not the Organization is an LEA or an SEA. The table filter indicates whether this is LEA or SEA or any other filter that is in place for the enumeration in CEDS. Do NOT adjust the TableFilter.
4. **InputCode** – This is your enumeration stored in the ed-fi.Descriptors or “Type” tables.
5. **OutputCode** – This is the CEDS code the InputCode will be transformed into. Do NOT adjust the OutputCode.

|  |
| --- |
| Mapping for default Ed-Fi Descriptors has been provided in the script **ODS.SourceSystemReferenceData.Data** described above. If you have custom Descriptors, you will need to enter the proper mapping in the SourceSystemReferenceData table manually. |

Note: Many to one relationships can exist. There can be two input codes pointed to one output code by creating a new row. If, however, one input code could be transformed into two CEDS output codes, an additional breakdown of your enumerations is required so the result is either a one to one relationship or a many to one relationship. The additional breakdown of enumerations in this instance will have to be done by creating a separate stored procedure.

## Configuring Stored Procedures

By default, no adjustment to the Stored Procedures are necessary if the core Ed-Fi ODS is used with the provided MetaEd extensions. Also, no adjustments are needed if only enumerations are custom as those are handled in the ODS.SourceSystemReferenceData table in generate. If, however, an adjustment needs to be made for where data are located (tables & column names are different), then a change needs to occur at the stored procedure level. The package comes with two sets of stored procedures. One contains the words “EncapsulatedCode.” These stored procedures move data from a staging area into the CEDS ODS. They should NEVER be adjusted. The other set of stored procedures without the text “EncapsulatedCode” move data out of the Ed-Fi ODS and into the staging area. When a change needs to be made to where the source data is located, there are two possible ways to achieve this:

1. **Recommended** – Create an additional stored procedure that will run after the first one that needs to be adjusted. This ensures the versioning of the original stored procedure remains the same. The original stored procedure will pull data out of the core location in the Ed-Fi ODS. The additional stored procedure you add will either entirely truncate and replace the staging table(s) associated with the stored procedure or will simply update the staging table for the relevant elements that need to be adjusted.
2. **Not Recommended** – Modify the existing stored procedure to point to the new source for data.

There are 17 Stored Procedures (these do not include the MetaEd Stored Procedures that pull data from the Ed-Fi extended tables) that can be adjusted which DO NOT contain the phrase “EncapsulatedCode”. They are listed below. A table is provided in Appendix A that shows the Staging tables they populate and the ED*Facts* reports categories that require their execution.

* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP01\_Organization
* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP02\_CharterSchoolManagementOrganization
* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP03\_Person
* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP04\_PersonRace
* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP05\_PersonStatus
* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP06\_Enrollment
* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP07\_ProgramParticipationSpecialEducation
* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP08\_Migrant
* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP09\_ProgramParticipationTitleI
* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP10\_ProgramParticipationCte
* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP11\_ProgramParticipationNorD
* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP12\_ProgramParticipationTitleIII
* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP13\_Discipline
* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP14\_StudentCourse
* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP15\_Assessment
* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP16\_StaffAssignment
* App.Migrate\_Data\_ETL\_IMPLEMENTATION\_STEP17\_StateDefinedCustomIndicator

The purpose of these stored procedures is to move the data out of the Ed-Fi ODS and into the staging tables that were created earlier in the installation section. There are also 17 corresponding “EncapsulatedCode” stored procedures. They move data out of the Staging tables and into Generate using the ODS.SourceSystemReferenceData to account for custom enumerations.

# Ed-Fi MetaEd Extension

Ed-Fi version 3.x does not contain all of the elements needed to produce the ED*Facts* reports contained in Generate. The MetaEd code provided will build the required extensions and enumerations to complete the model. If you have an existing Ed-Fi ODS implementation, you may have already extended the model to account for these additional elements or enumerations. Review the MetaEd documentation provided and determine what, if any, extensions you will want to implement.

Additional Stored Procedures are provided separately that pertain only to the extended data for the Ed-Fi ODS. These additional stored procedures point to the MetaEd extended version of the Ed-Fi ODS. They contain the text “MetaEd” on the end of their name.

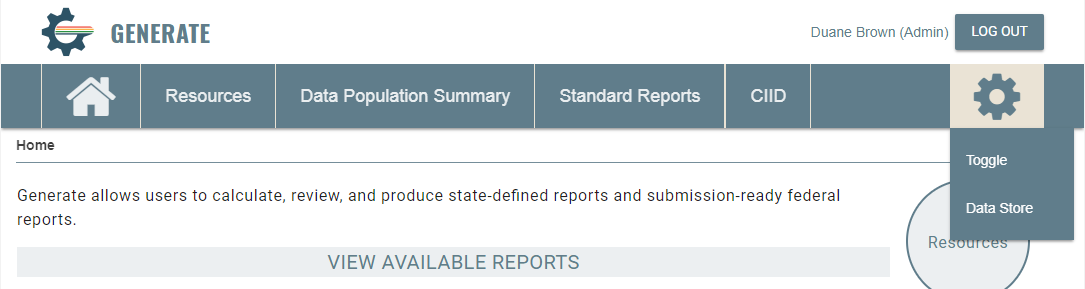
Only running the core ETL Stored Procedures without extending Ed-Fi or adjusting the “MetaEd” stored procedures to point to your own custom locations will still result in several of the ED*Facts* reports populating successfully, but for all 84 to populate, additional data are required which do not exist in the core Ed-Fi model and will require extension and population of data in the Ed-Fi ODS.

# Implementation & Validation

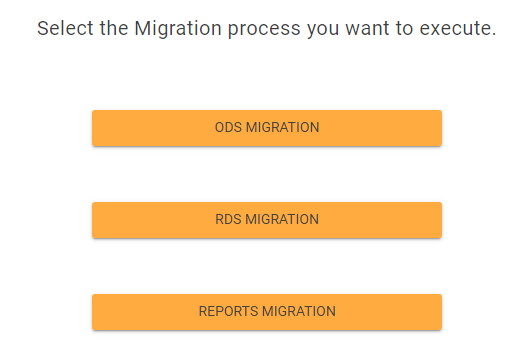
Installation and implementation guides related to the Generate application are provided by the Center for the Integration of IDEA Data (CIID). This guide assumes Generate is installed and provides only a quick summary of how to navigate the Generate application. Refer to resources provided by CIID (<https://ciidta.grads360.org/#program/generate>) for installing and configuring the Generate application and for greater detail in execution of the tools in the user interface.

Once the Ed-Fi to Generate Plug-In has been successfully installed in an existing Generate environment, you can run the migrations to move the data out of your Ed-Fi 3.x ODS and into Generate. To migrate the data:

1. Navigate to your local instance of Generate
2. Log in as an administrator and under the settings wheel, select “Data Store”



1. Select “ODS” Migration”



1. Select the School Year you want to retrieve data for and then select the migration steps you want to execute. Remember to select the corresponding “Encapsulated Code” stored procedure along with the staging stored procedure. This step will migrate data into the CEDS ODS.
2. When the migration has completed, return to the Data Store page and select “RDS Migration”. You will refer to the CIID Generate Implementation Guide which indicates the appropriate stored procedures to run during this step per grouping of ED*Facts* reports.
3. When the RDS Migration completes, return to the Data Store and run the Reports Migration selecting the applicable school year and ED*Facts* reports.

Each step in the ODS Migration can be run manually and individually to check for issues by running the steps in correct order specified in the App.DataMigrationTasks section. Use the following query to pull the steps in order of execution:

SELECT 'exec ' + StoredProcedureName

FROM app.DataMigrationTasks

WHERE IsActive = 1

ORDER BY RunBeforeGenerateMigration DESC, DataMigrationTypeId, TaskSequence

The Ed-Fi to Generate Plug-In also comes with an additional Validation process. At the end of each Stored Procedure that moves the data into the staging tables, a validation stored procedure is executed (App.Migrate\_Data\_Validation) which checks for missing elements and issues in the data. It places its results in the Staging.Validation table which can be queried to determine problems with the migration of data. Additional validation rules can be added to the App.Migrate\_Data\_Validation stored procedure as you find ways to improve error and validity checking. Locate each new error within the proper IF/THEN statement based on the Stored Procedure Name and the Staging Table Name it is validating.

# Supporting Documentation

Other resources available to you include:

**The Center for the Integration of IDEA Data (CIID)** – CIID is the creator of Generate and focuses specifically on areas related to Special Education Data. Their website is <https://ciidta.grads360.org/#program>. When working with Special Education Data and this ETL process, they may be able to provide you with technical assistance. Documentation for Generate is located here: <https://ciidta.grads360.org/#program/generate>

**The Common Education Data Standards (CEDS)** – Generate is built entirely on the CEDS Normalized Data Schema. Additional resources about the NDS can be found here: <https://ceds.ed.gov/dataModelNDS.aspx>. CEDS may also be able to provide additional technical assistance about the CEDS standard itself or transforming data into the CEDS standard. You can contact CEDS through this link: <https://ceds.ed.gov/ContactUs.aspx>.

**AEM Corporation** – AEM has created this Ed-Fi to Generate Plug In through a grant from the Michael and Susan Dell Foundation and in collaboration with the Ed-Fi Alliance. You can reach out directly to AEM by emailing Duane Brown ([duane.brown@aemcorp.com](mailto:duane.brown@aemcorp.com)).

# Appendix A – Migration Script Descriptions

Listed below are the ETL Scripts, the staging tables they populate and the ED*Facts* report categories they are responsible for. Note that only the last portion of the ETL name is included to make the table readable. Some ETLs are essential for all or most ED*Facts* reports and those will be noted in the ED*Facts* Reports Column using the following key:

* ALL – Every ED*Facts* report will require this ETL
* ALL PERSON – Any ED*Facts* report that counts either students, teachers, or staff will require this ETL
* ALL STUDENT – Any ED*Facts* report that counts students will require this ETL
* ALL STAFF – Any ED*Facts* report that counts teachers or staffs will require this ETL
* ALL RACE – Any ED*Facts* report that uses Race as a category set or subpopulation will require this ETL

|  |  |  |
| --- | --- | --- |
| **ETL** | **Staging Tables** | **ED*Facts* Reports** |
| Organization | Organization OrganizationAddress OrganizationCustomSchoolIndicatorStatusType OrganizationFederalFunding OrganizationGradesOffered OrganizationPhone OrganizationSchoolComprehensiveAndTargetedSupport OrganizationSchoolIndicatorStatus StateDetail | ALL |
| Person | Person | ALL PERSON |
| Enrollment | Enrollment | ALL STUDENT |
| ProgramParticipationSpecialEducation | ProgramParticipationSpecialEducation | All Special Education reports (except for those ED*Facts* reports that only require IDEA = Yes/No – that is pulled from PersonStatus) |
| Migrant | Migrant | All Migrant Reports |
| ProgramParticipationTitleI | ProgramParticipationTitleI | All Title I Reports |
| Discipline | Discipline | All Discipline Reports |
| StaffAssignment | StaffAssignment | ALL STAFF |
| Assessment | Assessment AssessmentResult | All Assessment Reports including English Learner |
| PersonStatus | PersonStatus PrimaryDisability | ALL STUDENT |
| PersonRace | PersonRace | ALL RACE |
| CharterSchoolManagementOrganization | CharterSchoolApprovalAgency  CharterSchoolManagementOrganization | All Charter School Reports |
| ProgramParticipationCTE | ProgramParticipationCTE | All CTE Reports |
| ProgramParticipationTitleIII | ProgramParticipationTitleIII | All Title III Reports |
| ProgramParticipationNoD | ProgramParticipationNoD | All Neglected and Delinquent Reports |