Aras Innovator 12

Batch Loader

Document #: 12.0.02019051801

Last Modified: *1/2/2020*

Copyright Information

Aras Corporation
100 Brickstone Square
Suite 100
Andover, MA 01810

Phone: 978-806-9400
Fax: 978-794-9826

E-mail: Support@aras.com

Website: https://www.aras.com/

Table of Contents

Send Us Your Comments

Aras Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for future revisions.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where and what level of detail?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, indicate the document title, and the chapter, section, and page number (if available).

You can send comments to us in the following ways:

Email:
Support@aras.com
Subject: Aras Innovator Documentation

Or,

Postal service:
Aras Corporation
100 Brickstone Square
Suite 100
Andover, MA 01810
Attention: Aras Innovator Documentation

Or,

FAX:
978-794-9826
Attn: Aras Innovator Documentation

If you would like a reply, provide your name, email address, address, and telephone number.

If you have usage issues with the software, visit https://www.aras.com/support/

Document Conventions

The following table highlights the document conventions used in the document:

Table 1: Document Conventions

Convention	Description
Bold	This shows the names of menu items, dialog boxes, dialog box elements, and commands. Example: Click **OK**.
`Code`	Code examples appear in `courier` text. It may represent text you type or data you read.
`Yellow highlight`	Code with yellow highlight is used to draw attention to the code that is being indicated in the content.
`Yellow highlight with red text`	Red color text with yellow highlight is used to indicate the code parameter that needs to be changed or replaced.
Italics	Reference to other documents.
Note:	Notes contain additional useful information.
Warning	Warning contains important information. Pay special attention to information highlighted this way.
Successive menu choices	Successive menu choices may appear with a greater than sign (-->) between the items that you will select consecutively. Example: Navigate to **File** --> **Save** --> **OK**.

1 Overview

The Aras Batch Loader tool is a utility for loading data from a flat file into Aras Innovator. This tool transforms the flat file data into AML for loading directly into Aras Innovator similar to an alternative client. The Batch Loader has two distinct modes for loading data. The first, more direct method is through execution of the Batch Loader utility at the command line. This allows for rapid data loading into an Aras Innovator instance based upon a configuration pre-defined in an xml file. The use of the command line executable also makes it suitable for use in a daily batch job through the use of Windows Scheduler. The other option for batch loading data is through the graphical user interface (GUI). A screenshot of the batch load GUI is shown below:

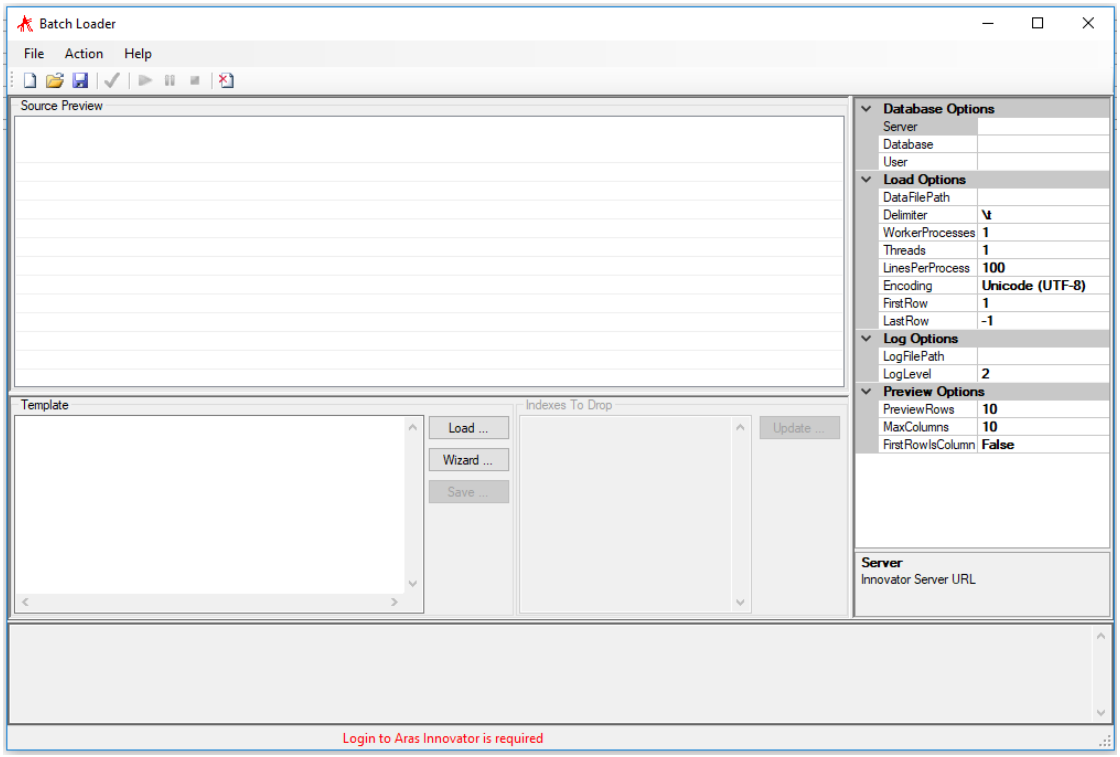


Figure 1.

The visual tool provides the same capabilities as the command line tool but allows for easier user manipulation of the data and format to be loaded. This document explains the use of both modes, starting with the GUI.

2 License Policy

The Aras Innovator Batch Loader requires a Feature License in order to run. To check whether a valid Feature License exists in Aras Innovator:

1. Log into Aras Innovator as an administrator.
2. Go to **Administration** > **Feature Licenses**.
3. If a feature license is found, check the expiration date to confirm that the license is still active.

2.1 Requesting a feature license

In order to request a feature license, customers with an active Aras subscription can send an email to licenses@aras.com or to your account representative in the following format:

- Subject – 'Batch Loader activation key required'
- Body - Version of Aras Innovator

You receive a reply containing the Feature License activation key.

2.2 Installing the license

Once you have the activation key, install the feature license as follows:

1. Log into Aras Innovator as an administrator.
2. Click the User Menu icon 

and select **Activate Feature** from the dropdown menu**.** The Activate Feature dialog box appears.

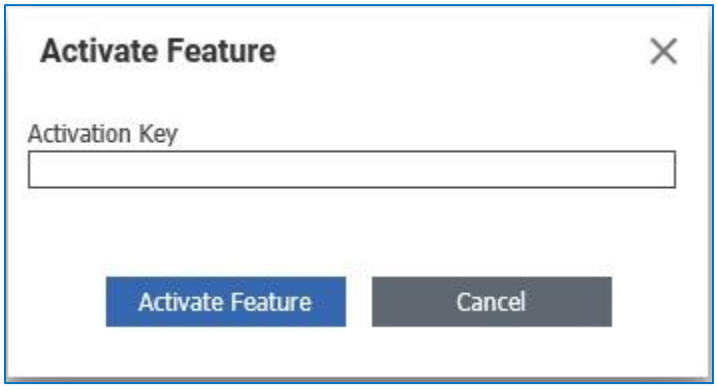


Figure 2.

3. Paste the acquired activation key and click **Activate Feature**. A feature activation message appears.

3 Batch Loader Graphical User Interface

The Batch Loader GUI allows the user to set the required parameters interactively prior to loading the data. This GUI is designed to help user new to AML load data into Aras Innovator by providing an interface that provides feedback on how the AML will be written before applying it to the database. For advanced use of AML with this tool, it is recommended you take that Aras Innovator API class, to familiarize yourself with the Adaptive Markup Language (AML).

3.1 Description of Visual Batch Loader

To start the Batch Loader UI run the BatchLoader.exe file:

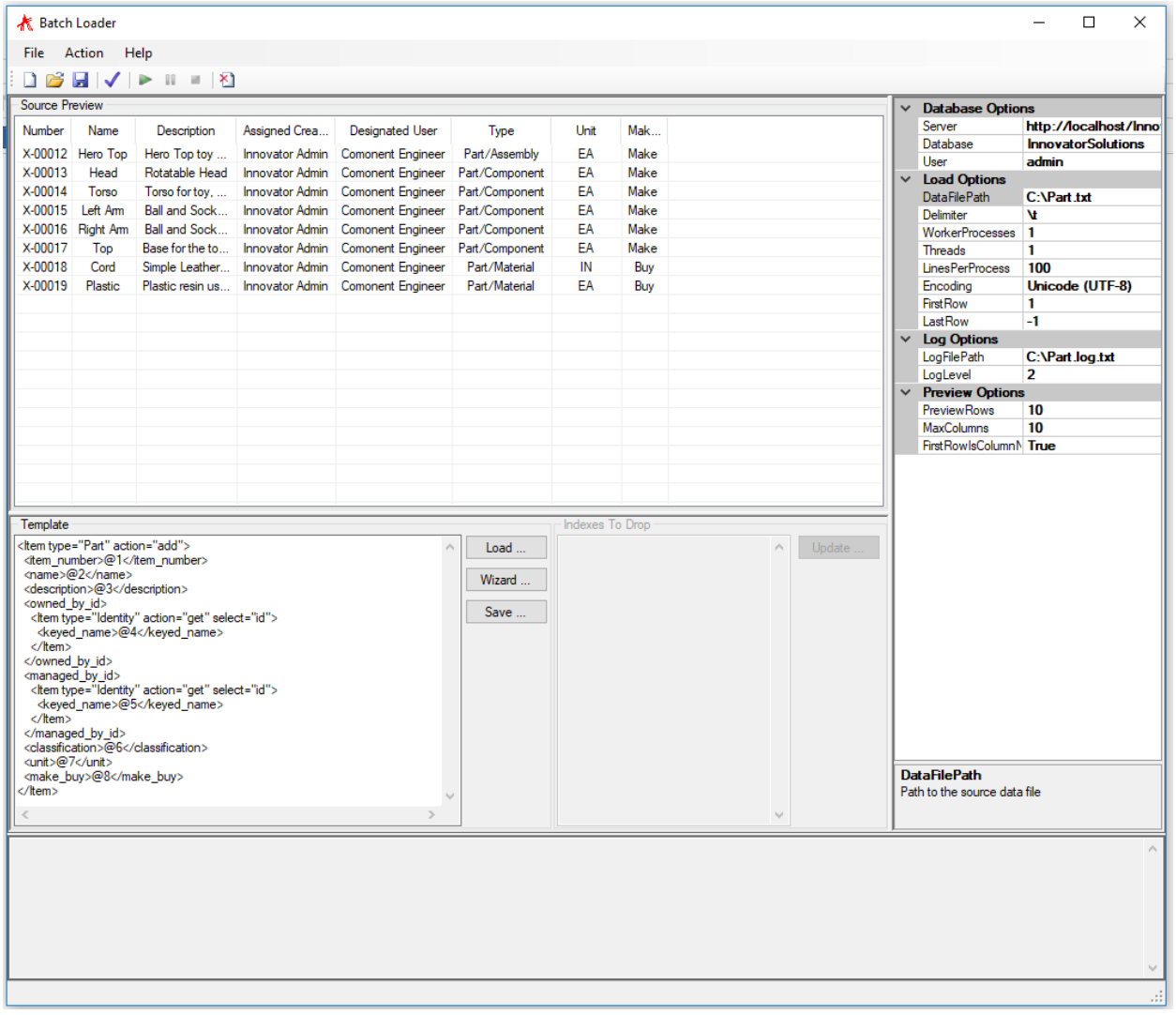


Figure 3.

The following is a list of toolbar icons, their corresponding menu items, and a brief explanation of what their function is.

Table 2: Toolbar Icons

Icon	Menu	Description
	**File --> New**	Starts a new configuration which can be later saved in an .ibl file. The configuration includes pointers to the Aras Innovator server and the data file to be parsed with specific parsing instructions.
	**File --> Open**	Opens an existing configuration file (.ibl). These files contain a specification of data mappings templates, the actual data file to be used, as well as auxiliary information, such as the server pointer, the log file specification, and other settings.
	**File --> Save**	Saves the current configuration file to a specified name and location. The configuration file contains a full specification of data mappings, the actual data file to be used, as well as auxiliary information, such as the server pointer, the log file specification, and other settings.
	**Actions --> Verify**	Verifies the AML template to be loaded.
	**Actions --> Load**	Loads the data into the specified data base, according to the data mappings and other instructions created by the user.
		Pauses the database load. Available only after the Load is activated and before it completes.
		Stops the database load. Available only after the Load is activated and before it completes.
		Clears the log window.

3.2 Loading Simple Item Types

Let's work through an example of a loading some Parts data into a database. As we go through this example, the Specifications Window parameters are explained.

Below is a sample data file that we use in this example. It is stored in Parts.txt. Although created in Excel, it was saved out as a tab delimited text file in order to be used with the Batch Loader.

Number	Name	Assigned Creator	Designated User	Unit	MakeBuy	Classification
M112-B000002	STEERING, HYDRAULIC	Innovator Admin	Innovator Admin	EA	MAKE	Assembly
M112-B000003	STEERING, HYDRAULIC, COUGAR	Innovator Admin	Innovator Admin	EA	MAKE	Assembly
M112-B000005FP	STEERING - HYDRAULIC	Innovator Admin	Innovator Admin	EA	MAKE	Assembly
M113-B000001	STEERING, COLUMN	Innovator Admin	Innovator Admin	EA	MAKE	Assembly
M113-B000003FP	STEERING COLUMN	Innovator Admin	Innovator Admin	EA	MAKE	Assembly
M115-B000005	WHEELS & TIRES 4X4	Innovator Admin	Innovator Admin	EA	MAKE	Assembly
M115-B000010	WHEELS & TIRES 6X6	Innovator Admin	Innovator Admin	EA	MAKE	Assembly
M115-B000014FP	WHEEL AND TIRES	Innovator Admin	Innovator Admin	EA	MAKE	Assembly
M120-B00LRIP	ENGINE - LRIP, MECHANICAL	Innovator Admin	Innovator Admin	EA	MAKE	Assembly
M120-B000002	ENGINE-MECHANICAL	Innovator Admin	Innovator Admin	EA	MAKE	Assembly
M120-B000007	ENGINE-MECHANICAL	Innovator Admin	Innovator Admin	EA	MAKE	Assembly
M120-B000010	ENGINE-MECHANICAL	Innovator Admin	Innovator Admin	EA	MAKE	Assembly
M120-B000011	ENGINE-MECHANICAL	Innovator Admin	Innovator Admin	EA	MAKE	Assembly
M120-B000016FP	ENGINE-MECH	Innovator Admin	Innovator Admin	EA	MAKE	Assembly
M120-CHEETAH	ENGINE-MECHANICAL	Innovator Admin	Innovator Admin	EA	MAKE	Assembly
M125-B00LRIP	TRANSMISSION, MECH	Innovator Admin	Innovator Admin	EA	MAKE	Assembly

Figure 4.

To load the Parts.txt into a database:

1. Start the `BatchLoader.exe`
2. Select **New** from the main Tool Bar. Notice that the Configuration Pane appears differently, as shown below. All these properties and their values are saved in the configuration file for future use.

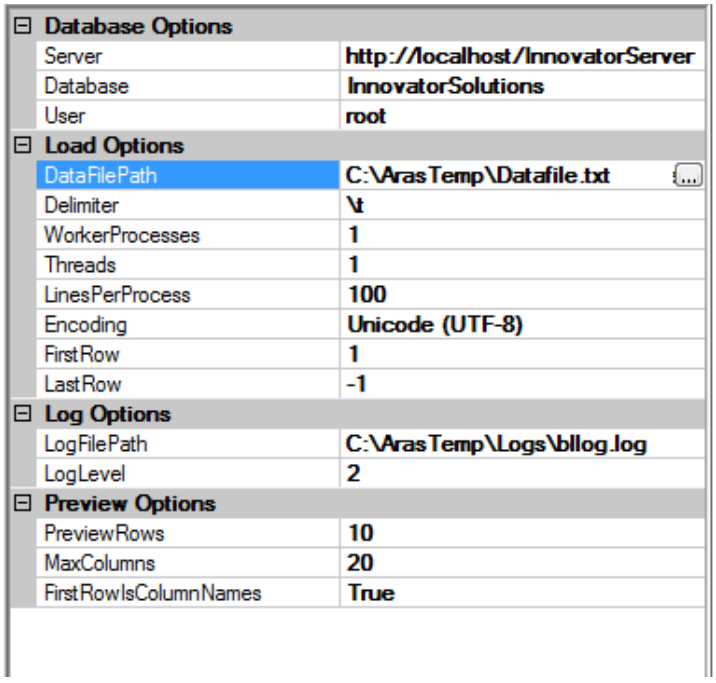


Figure 5.

3. The following are explanations of the properties of this window:
   - **Server** – The connection URL for Aras Innovator. If all the defaults were taken during the Aras Innovator installation, the path should be something such as:

http://localhost/InnovatorServer. If unsure, check in IIS to get the exact path. This URL should not include reference to the /Client folder

- **Database** – The database to which the data is loaded. Selecting this field, after defining the connection URL, makes this a pick list of available databases.
- **User** – The user login to be used for connecting to and loading data into the database.
- **DataFilePath** – The fully qualified path to data file that contains the data to be loaded into the database. In our case this is C:\Parts.txt
- **Delimiter** – the delimiter used to separate data, usually a tab (\t), or a space ( ), or a comma (,).
- **WorkerProcesses** – The number of worker processes to be used by the Batch Loader while loading data. Recommend using the default of 1.
- **Threads** – The number of threads per worker process. Recommend using the default of 1.
- **LinesPerProcess** – The number of lines in the data file that are loaded by a single worker process. If the worker process finishes processing all its lines and the data file has more lines to be processed, then a new worker process is started.
- **Encoding** – Encoding (or codepage number) of data file
- **FirstRow** – The number of the row where the actual data starts. Sometimes the first row is used for row headings. In that case, the data starts in the second row. (See FirstRowIsColumnNames property below.)
- **LastRow** – The number of the row where the actual data stops. Default of -1 indicates that the file should be read until the end of the file
- **LogFilePath** – The fully specify the name of the log file where all information and errors are to be written by the Batch Loader.
- **LogLevel** – The Level of detail included in the logging.
  - 1 – Low, recommended for automated jobs with low risk on failure. Details about start and stop, and how many items succeeded.
  - 2 – Medium, recommended for use while developing new Batch load job. Logs details about failure, as well as details logged in low mode.
  - 3 – High, recommended for debugging. Provides detail and AML about every line loaded.
- **PreviewRows** – The number of rows visible in the Preview Pane
- **MaxColumns** – The number of columns in the Preview Pane.
- **FirstRowIsColumnNames** – When set to True, the Batch Loader starts parsing the data file from the second row after the FirstRow value.

4. Let's fill out the information in the Configuration Pane as follows:
   - FileName – C:\Parts.txt
   - Server – http://localhost/InnovatorServer
   - Database – InnovatorSolutions
   - User - admin
   - LogFilePath – C:\Parts.log
   - FirstRowIsColumnNames – True

This takes care of all the auxiliary properties and settings. The next step is to set up the data mappings from the values in the data file to the items and properties in the database. Notice that the data file contains 8 columns – Number, Name, Description, Assigned Creator, Designated User, Type, Unit, and Make/Buy. So, we have to specify that we are importing instances of Part with 8 property values.

To set up the data mappings using the Batch Loader UI, there are two options. First, you can use the template wizard to build a basic template for loading. Or second, you can use a predefined template loaded from a file. For simplicity's sake, we walk through using the wizard. Creating templates is covered in the command line execution of the batch loader. Or finally, you may edit the AML template directly in the Template Pane.

Note: Changes made in the AML template during a session are saved only if you use the Save button next in the AML Template pane.

5. From the template pane, select **Wizard**. A new dialog is brought up for data mapping:

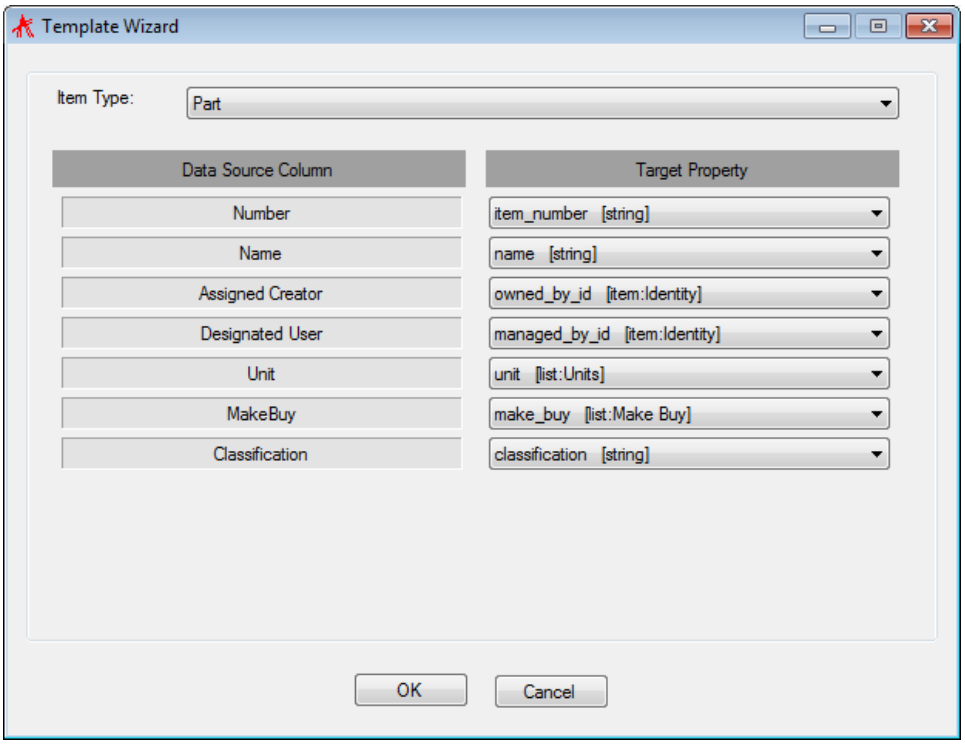


Figure 6.

6. Select the name of the ItemType from the first dropdown, Part in our case.

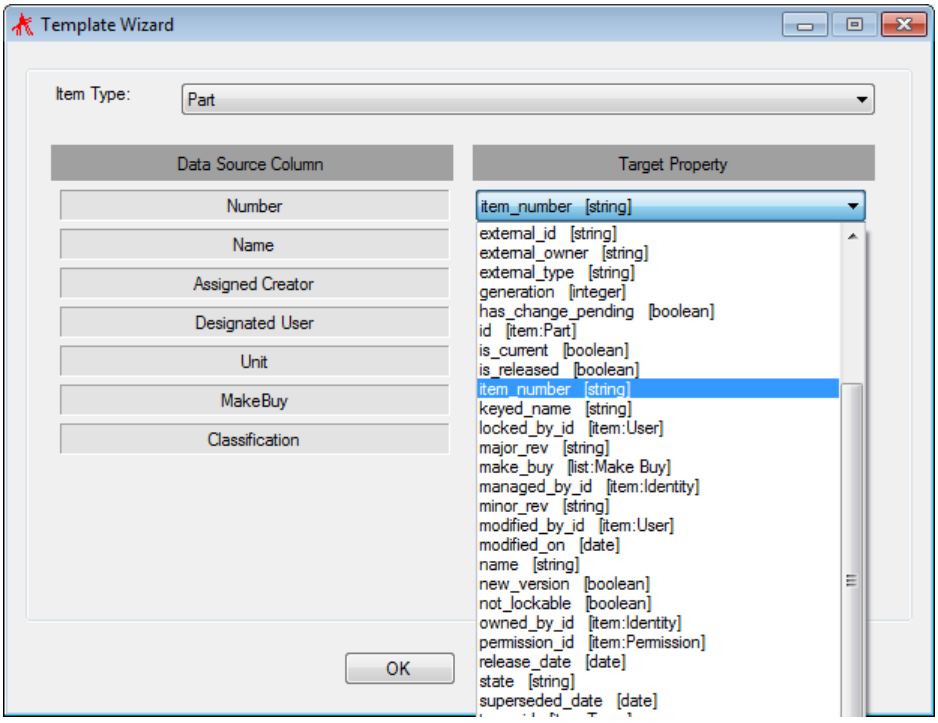


Figure 7.

7. Select **Part** from the Item Type scroll list.
8. Next, you need to map each column in the data file to each corresponding property of the ItemType. Click on the down arrow of the Target Property to select the property that receives the values in this column. Any properties of type Item are mapped by keyed name of the target Item.
9. Click **OK** once all mappings are finished. You should see the mappings displayed in the Batch Loader template pane.

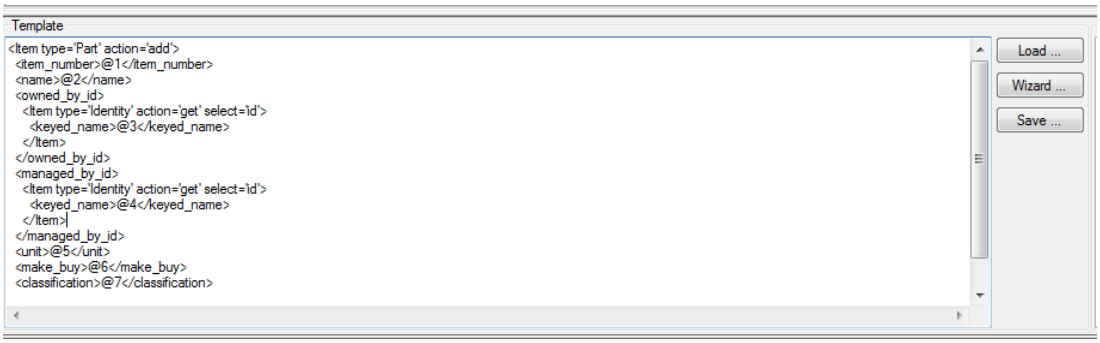
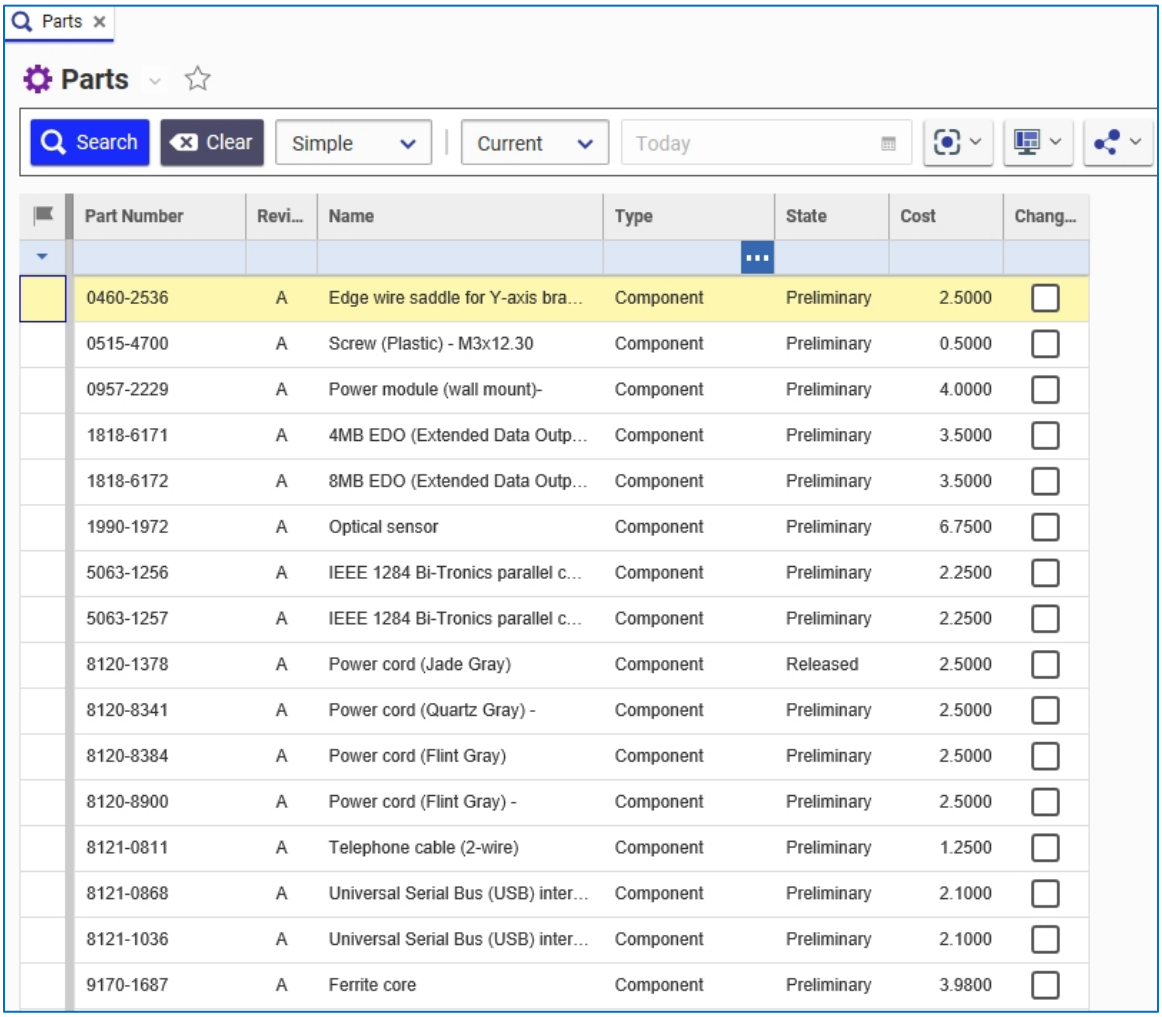


Figure 8.

10. Click **Save** in the main tool bar, and save this configuration file. The configuration file (with an .ibl extension) saves all the Configuration Pane settings, such as the server path, and the data file name, as well as all the data mappings.
11. Click **Verify** in the main tool bar to make sure that there are no errors in the AML template.
12. Click **Load**.
13. Log into Aras Innovator, using the same database that you specified for the data load.

14. Select **Design>Parts** and click the Search icon . The parts we loaded should appear.



Part Number	Revi...	Name	Type	State	Cost	Chang...
0460-2536	A	Edge wire saddle for Y-axis bra...	Component	Preliminary	2.5000	
0515-4700	A	Screw (Plastic) - M3x12.30	Component	Preliminary	0.5000	
0957-2229	A	Power module (wall mount)-	Component	Preliminary	4.0000	
1818-6171	A	4MB EDO (Extended Data Outp...	Component	Preliminary	3.5000	
1818-6172	A	8MB EDO (Extended Data Outp...	Component	Preliminary	3.5000	
1990-1972	A	Optical sensor	Component	Preliminary	6.7500	
5063-1256	A	IEEE 1284 Bi-Tronics parallel c...	Component	Preliminary	2.2500	
5063-1257	A	IEEE 1284 Bi-Tronics parallel c...	Component	Preliminary	2.2500	
8120-1378	A	Power cord (Jade Gray)	Component	Released	2.5000	
8120-8341	A	Power cord (Quartz Gray) -	Component	Preliminary	2.5000	
8120-8384	A	Power cord (Flint Gray)	Component	Preliminary	2.5000	
8120-8900	A	Power cord (Flint Gray) -	Component	Preliminary	2.5000	
8121-0811	A	Telephone cable (2-wire)	Component	Preliminary	1.2500	
8121-0868	A	Universal Serial Bus (USB) inter...	Component	Preliminary	2.1000	
8121-1036	A	Universal Serial Bus (USB) inter...	Component	Preliminary	2.1000	
9170-1687	A	Ferrite core	Component	Preliminary	3.9800	

Figure 9.

3.3 Loading Relationships

Loading relationships into the database using the Batch Loader is identical to loading the Item instances. The Configurations Pane parameters, such as the data file name and the Server fully specified path, must be completed. Alternatively, you can use an .ibl configuration file that may have these parameters previously set. The difference is of course in the data mappings. Let's look at the following data file:

Parent Part	Child Part	Quantity
0105-XX1-001	0003-MM6-001	3
0105-XX1-001	0005-MM6-001	5
0105-XX1-001	0006-MM6-001	8
0105-XX1-001	0001-MM6-001	4
0105-XX1-001	0007-MM6-001	2
0105-XX1-001	0014-GG6-SP006	1
0105-XX1-001	0014-GG6-SP1307	3
0105-XX1-001	0014-GG6-SP1308	6
0108-XX1-001	0005-MM6-001	3
0108-XX1-001	0001-MM6-001	4

Figure 10.

To set the data mappings:

1. Start the `BatchLoader.exe`
2. Select **New** from the main Tool Bar.
3. Fill out the information in the Configuration Pane as follows:
   a. FileName – C:\BOM.txt
   b. Server – http://localhost/InnovatorServer
   c. Database – InnovatorSolutions
   d. User - admin
   e. LogFilePath – C:\BOM.log
   f. FirstRowIsColumnNames – True
4. From the template pane, select **Wizard**. A new dialog is brought up for data mapping:

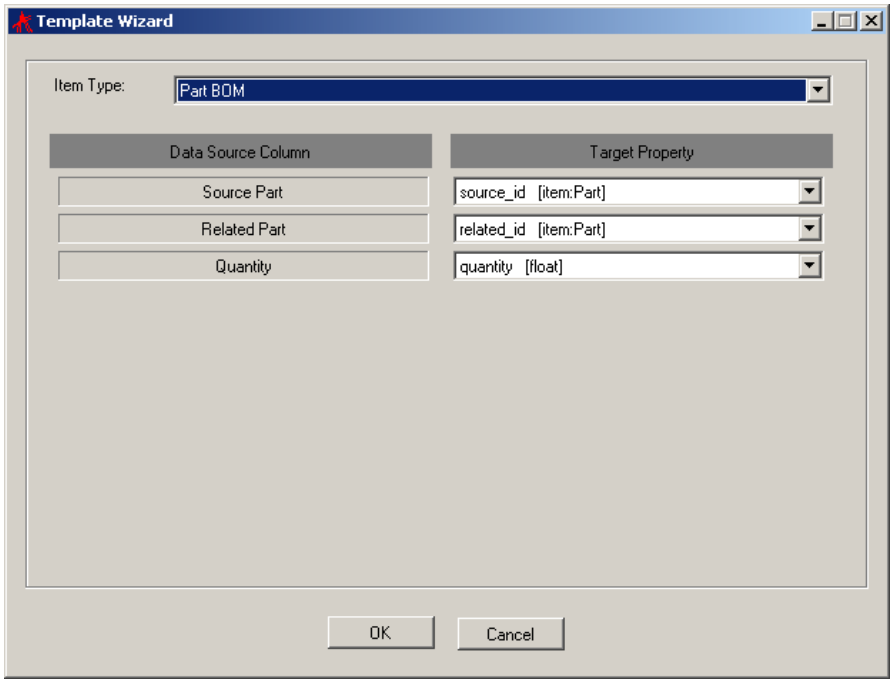


Figure 11.

5. Select Part BOM from the Item Type dropdown.

a. The Source Part maps to the keyed_name of the source_id Item
b. The Related Part maps to the keyed_name of the related_id Item

6. Click **OK** once all mappings are finished. You should see the mappings displayed in the Batch Loader template.

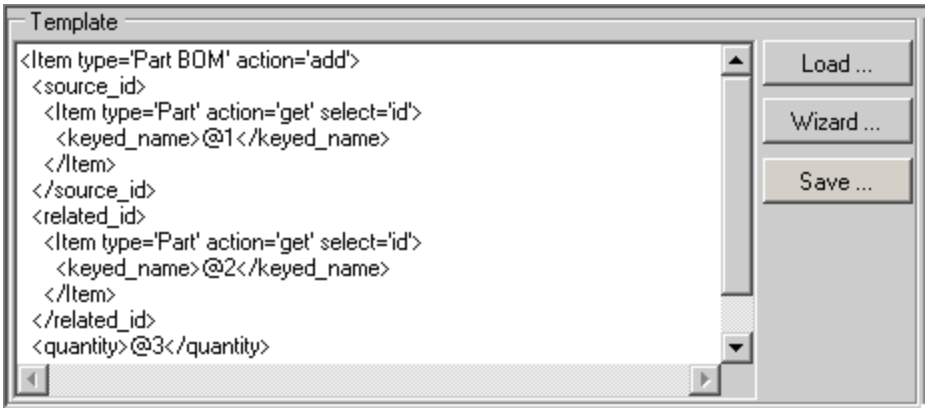
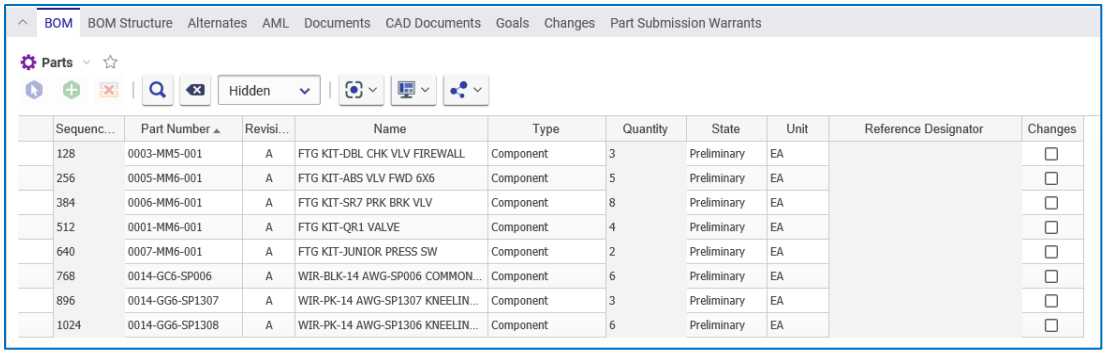


Figure 12.

7. Click **Save** in the main tool bar, and save this configuration file. The configuration file (with an .ibl extension) saves all the Configuration Pane settings, such as the server path, and the data file name, as well as all the data mappings.
8. Click **Verify** in the main tool bar to make sure that there are no errors in the AML template.
9. Click **Load**.
10. Log into Aras Innovator, using the same database that you specified for the data load.
11. Go to Part X-00012, and look at the added BOM relationships:



Sequenc...	Part Number	Revisi...	Name	Type	Quantity	State	Unit	Reference Designator	Changes
128	0003-MM5-001	A	FTG KIT-DBL CHK VLV FIREWALL	Component	3	Preliminary	EA		
256	0005-MM6-001	A	FTG KIT-ABS VLV FWD 6X6	Component	5	Preliminary	EA		
384	0006-MM6-001	A	FTG KIT-SR7 PRK BRK VLV	Component	8	Preliminary	EA		
512	0001-MM6-001	A	FTG KIT-QR1 VALVE	Component	4	Preliminary	EA		
640	0007-MM6-001	A	FTG KIT-JUNIOR PRESS SW	Component	2	Preliminary	EA		
768	0014-GC6-SP006	A	WIR-BLK-14 AWG-SP006 COMMON...	Component	6	Preliminary	EA		
896	0014-GG6-SP1307	A	WIR-PK-14 AWG-SP1307 KNEELIN...	Component	3	Preliminary	EA		
1024	0014-GG6-SP1308	A	WIR-PK-14 AWG-SP1306 KNEELIN...	Component	6	Preliminary	EA		

Figure 13.

3.4 Advanced AML Templates

The Batch Loader templates are simply AML templates. Because of this, the Batch Loader has all the features of AML when loading data. The previous sections have only walked through some basic Items created using the Template Wizard. The template wizard limits the load to a top level action of 'add', loads only one level of an Item structure at a time, and can only reference item properties through the keyed name. While these limitations are often enough for simple data loads, but often are not enough for regular batch jobs.

To take advantage of the advanced capabilities of the Batch Loader, you should do the following:

1. Attend the Aras Innovator API class. While this is not required, it is often easier to understand the AML structure when you have finished the lessons of AML in this class.
2. Review the command line section of this document. The command line section of this document contains further examples of templates, as well as an explanation of the parameterized query in AML.
3. Review the samples listed in the appendix of this document. To aid users with loading some common ItemTypes into Aras Innovator, an appendix of sample AML templates has been provided.

4 Batch Loader Command Line Tool

The command line version of the Batch Loader tool greatest advantage is that it allows the use of a known configuration in a regularly scheduled process, such as loading User info into a database. It could even be used to provide a set of sample data to be applied to any database with the same configuration. It is as flexible as the UI, but has the ability to save on setup time by reusing the previous configuration.

4.1 Command Line Options

There are several elements to the command line tool. First, is the executable file for the batch load tool itself, BatchLoaderCMD.exe. The details of the various options for running the executable at the command line are as follows:

```
Usage:  BatchLoaderCmd.exe -d {data file} [optional arguments]

where:
  -d {data file}       - path to a delimiter-separated data file
```

Optional arguments can either be specified in the command line or in the configuration file; if not defined in either place a default value is assumed (if a default value is mentioned below):

```
  -c {config file}     - path to configuration file; default 'config.xml'
  -t {template file}   - path to template file; default 'template.xml'
  -l {log file}        - path to log file
  -ll {log level}      - verbosity of logging (1 - 3); default 1
  -e {encoding}        - name of data file encoding; e.g. 'us-ascii'
  -s {url}             - Innovator server URL
  -db {db name}        - Database name
  -u {user name}       - Innovator user login name
  -p {user password}   - Innovator user password
  -th {threads}        - Number of threads in a worker process; default 1
  -pr {processes}      - Number of worker processes; default 1
  -lpp {lines}         - Lines processed by each worker process; default 1000
  -fr {row number}     - First row to process in the data file; default 1
  -lr {row number}     - Last row to process in the data file; default -1
```

Note: Most of these parameters are optional and are generally included in the xml configuration file. However, the command line arguments override any values in the configuration file.

4.2 Connection Mapping in the Configuration File

The configuration file that is referenced by the batch load application is a straightforward xml file that defines the parameters for the batch load activity. The configuration file is used to configure the Aras Innovator server and data source, as well as the process parameters. The following describes the different parameter tags that are included in the XML configuration file.

```
<BatchLoaderConfig>
  <server>http://localhost/InnovatorServer</server>
  <db>InnovatorSolutions</db>
  <user>admin</user>
```

```
  <password>innovator</password>
  <max_processes>1</max_processes>
  <threads>1</threads>
  <lines_per_process>250</lines_per_process>
  <delimiter>\t</delimiter>
  <encoding>utf-8</encoding>
  <first_row>2</first_row>
  <last_row>-1</last_row>
  <log_file>C:\User</log_file>
  <log_level>3</log_level>
</BatchLoaderConfig>
```

The only parameter defined in the configuration file that cannot be defined in the command line is the delimiter. Any other parameter tag may be left out of the configuration file.

- **server** – The connection URL for Aras Innovator. If all the defaults were taken during the Aras Innovator installation, the path should be something such as: http://localhost/InnovatorServer. If unsure, check in IIS to get the exact path. This URL should not include reference to the /Client folder
- **db** – The database to which the data is loaded. Selecting this field after defining the connection URL makes this a pick list of available databases.
- **user** – The user login to be used for connecting to and loading data into the database.
- **password** – The user password in plain text.
- **max_processes** – The number of worker processes to be used by the Batch Loader while loading data. Recommend using the default of 1.
- **threads** – The number of threads per worker process. Recommend using the default of 1.
- **lines_per_process** – The number of lines in the data file that are loaded by a single worker process. If the worker process reaches the end of its line and the data file has not completed processing, then a new worker process is started.
- **delimiter** – the delimiter used to separate data, usually a tab (\t), or a space ( ), or a comma (,).
- **encoding** – Encoding (or codepage number) of data file
- **first_row** – The number of the row where the actual data starts. Sometimes the first row is used for row headings. In that case, the data starts in the second row. (See FirstRowIsColumnNames property below.)
- **last_row** – The number of the row where the actual data stops. Default of -1 indicates that the file should be read until the end of the file
- **log_file** – The fully specify the name of the log file where all information and errors are to be written by the Batch Loader.
- **log_level** – The level of detail included in the logging.
  - 1 – Low, recommended for automated jobs with low risk on failure. Details about start and stop, and how many items succeeded.
  - 2 – Medium, recommended for use while developing new Batch load job. Logs details about failure, as well as details logged in low mode.
  - 3 – High, recommended for debugging. Provides detail and AML about every line loaded.

4.3 Item Mapping in the Template file

A Batch Loader template file contains a parameterized template of the AML request that is sent to the Aras Innovator server by the batch loader session for every processed row of the input data file. Parameters of the template have format @N, e.g. @1. The N number is the integer column number from left to right in the data file.

4.3.1 Basic Item Mapping

Below is an example of what a template file might look like for the User ItemType.

```
<BatchLoaderPrototype>
  <Item type="User" action="add">
    <login_name>@1</login_name>
    <password>607920b64fe136f9ab2389e371852af2</password>
    <logon_enabled>@2</logon_enabled>
    <first_name>@3</first_name>
    <last_name>@4</last_name>
    <email>@5</email>
  </Item>
</BatchLoaderPrototype>
```

This file could be used to add Users into Aras Innovator, similar to the GUI.

Note: The password has been hard coded to a specific value of 'innovator'.

By making a couple of small changes to the file

```
<BatchLoaderPrototype>
  <Item type="User" where="login_name='@1'" action="merge">
    <login_name>@1</login_name>
    <!-- password>607920b64fe136f9ab2389e371852af2</password -->
    <logon_enabled>@2</logon_enabled>
    <first_name>@3</first_name>
    <last_name>@4</last_name>
    <email>@5</email>
  </Item>
</BatchLoaderPrototype>
```

We make the file usable for a nightly load of user data. Notice that the password has been commented out, to prevent the batch job from overwriting the current values. Notice also, that the Item tag has a where clause included to specify how to query for existing Users. Lastly, notice that the top level action has been changed from 'add' to 'merge'. This is so that if the User in the AML sent to the server exists, then the action is to 'edit' the existing user. However, if the User does not exist, then the action is set to 'add', so that the new user is created. This is especially useful when trying to manage process such as User data uploaded from Active Directory.

4.3.2 Item and Relationship Simultaneous Load

Drawing from the example in the section Batch Loader Graphical User Interface, we can make an AML template that would load better represent how the Part BOM relationship should be loaded from the same sample data file.

```
<BatchLoaderPrototype>
  <Item type="Part" where="item_number='@1'" action="edit">
    <Relationships>
      <Item type="Part BOM" action="add">
        <related_id>
```

```
          <Item type="Part" action="get" select="id">
            <item_number>@2</item_number>
          </Item>
        </related_id>
        <quantity>@3</quantity>
      </Item>
    </Relationships>
  </Item>
</BatchLoaderPrototype>
```

This AML template make the additions in the context of exiting source Part Items only, and forces the part data to make updates to the Part Item history, while uploading the BOM data.

4.3.3 Uploading a File Item

Uploading files to Aras Innovator can be done using the Batch Loader, but Aras recommends loading the File Item in the context of the source item. The following sample template allows you to load the File Item as part of loading the source Document Item. This decreases the complexity of trying to create the correct Document File relationship, if the Document and File had been loaded separately.

```
<Item type="Document" action="add">
   <item_number>@1</item_number>
   <description>@2</description>
   <Relationships>
     <Item type="Document File" action="add">
       <related_id>
         <Item type="File" action="add">
           <actual_filename>@3</actual_filename>
           <filename>@4</filename>
           <Relationships>
             <Item type="Located" action="add">
               <related_id>67BBB9204FE84A8981ED8313049BA06C</related_id>
             </Item>
           </Relationships>
         </Item>
       </related_id>
     </Item>
   </Relationships>
 </Item>
```

From the AML, you can see that @1 and @2 are mapped to properties of the document item. To load the File Item, we are leveraging the standard Aras Innovator IOM. This requires that we define the following parameters, at a minimum.

On the File Item:

actual_filename (@3 above)– This is the path to the file that is to be vaulted.

Example: C:\Drawing.bmp

filename (@4 above) – This is the name of the file that for the database.

Example: Drawing.bmp

Uploading a File Item also requires the ‘Located’ relationship be defined. In the example provided, the Located relationship has a related Item of the Default vault. If the File must be vaulted to a Vault other than the Default vault, the AML template must be altered to use the id of the alternate Vault.

5 Appendix I: AML Template File Samples

This contains some sample template AMLs files for commonly batch loaded ItemTypes. These templates do not have all properties defined, but enough of the common structure to get started.

Warning If you are using “where” clauses in your template AML , please review Aras Innovator 12 0 - AML Security Updates document before importing your data.

5.1 User

For adding new Users with a password of ‘innovator’:

```
<BatchLoaderPrototype>
  <Item type="User" where="login_name='@1'" action="add">
    <login_name>@1</login_name>
    <password>607920b64fe136f9ab2389e371852af2</password>
    <logon_enabled>@2</logon_enabled>
    <first_name>@3</first_name>
    <last_name>@4</last_name>
    <email>@5</email>
  </Item>
</BatchLoaderPrototype>
```

For a batch job maintaining user status in an external file Users:

```
<BatchLoaderPrototype>
  <Item type="User" where="login_name='@1' and is_current='1'"
action="merge">
    <login_name>@1</login_name>
    <logon_enabled>@2</logon_enabled>
    <first_name>@3</first_name>
    <last_name>@4</last_name>
    <email>@5</email>
  </Item>
</BatchLoaderPrototype>
```

5.2 Part

For adding new Parts:

```
<BatchLoaderPrototype>
  <Item type="Part" action="add">
    <item_number>@1</item_number>
    <name>@2</name>
    <description>@3</description>
    <owned_by_id>
      <Item type="Identity" action="get" select="id">
        <keyed_name>@4</keyed_name>
      </Item>
    </owned_by_id>
    <managed_by_id>
      <Item type="Identity" action="get" select="id">
        <keyed_name>@5</keyed_name>
      </Item>
```

```
    </managed_by_id>
    <classification>@6</classification>
    <unit>@7</unit>
    <make_buy>@8</make_buy>
  </Item>
</BatchLoaderPrototype>
```

5.3 Part BOM

For adding Part BOM relationships to an existing Part:

```
<BatchLoaderPrototype>
  <Item type="Part" where="item_number='@1' and is_current='1'"
action="edit">
    <Relationships>
      <Item type="Part BOM" action="add">
        <related_id>
          <Item type="Part" where="item_number='@2' and is_current='1'"
action="get" select="id">
        </related_id>
        <quantity>@3</quantity>
      </Item>
    </Relationships>
  </Item>
</BatchLoaderPrototype>
```

If you need to add many Part BOM relationships to the same Part, but do not want to version the item after every addition.

```
<BatchLoaderPrototype>
  <Item type="Part" where="item_number='@1' and is_current='1'" action="edit"
version="0">
    <Relationships>
      <Item type="Part BOM" action="add">
        <related_id>
          <Item type="Part" where="item_number='@2' and is_current='1'"
action="get" select="id">
        </related_id>
        <quantity>@3</quantity>
      </Item>
    </Relationships>
  </Item>
</BatchLoaderPrototype>
```

5.4 Part Goal

For adding Part Goal relationships to an existing Part:

```
<BatchLoaderPrototype>
  <Item type="Part" where="item_number='@1' and is_current='1'"
action="edit">
    <Relationships>
      <Item type="Part Goal" action="add">
        <goal>@2</goal>
        <target_value>@3</target_value>
        <estimated_value>@4</estimated_value>
```

```
        <actual_value>@5</actual_value>
      </Item>
    </Relationships>
  </Item>
</BatchLoaderPrototype>
```

For adding a new Part with a Part Goal:

```
<BatchLoaderPrototype>
  <Item type="Part" action="add">
    <item_number>@1</item_number>
    <name>@2</name>
    <description>@3</description>
    <owned_by_id>
      <Item type="Identity" action="get" select="id">
        <keyed_name>@4</keyed_name>
      </Item>
    </owned_by_id>
    <managed_by_id>
      <Item type="Identity" action="get" select="id">
        <keyed_name>@5</keyed_name>
      </Item>
    </managed_by_id>
    <classification>@6</classification>
    <unit>@7</unit>
    <make_buy>@8</make_buy>
    <Relationships>
      <Item type="Part Goal" action="add">
        <goal>@9</goal>
        <target_value>@10</target_value>
        <estimated_value>@11</estimated_value>
        <actual_value>@12</actual_value>
      </Item>
    </Relationships>
  </Item>
</BatchLoaderPrototype>
```

5.5 Document

For adding new Documents:

```
<BatchLoaderPrototype>
  <Item type="Document" action="add">
    <item_number>@1</item_number>
    <name>@2</name>
    <description>@3</description>
    <owned_by_id>
      <Item type="Identity" action="get" select="id">
        <keyed_name>@4</keyed_name>
      </Item>
    </owned_by_id>
    <managed_by_id>
      <Item type="Identity" action="get" select="id">
        <keyed_name>@5</keyed_name>
      </Item>
    </managed_by_id>
    <classification>@6</classification>
```

```
    <authoring_tool>@7</authoring_tool>
    <authoring_tool_version>@8</authoring_tool_version>
  </Item>
</BatchLoaderPrototype>
```

For adding a new document with an attached file:

```
<BatchLoaderPrototype>
  <Item type="Document" action="add">
    <item_number>@1</item_number>
    <name>@2</name>
    <description>@3</description>
    <owned_by_id>
      <Item type="Identity" action="get" select="id">
        <keyed_name>@4</keyed_name>
      </Item>
    </owned_by_id>
    <managed_by_id>
      <Item type="Identity" action="get" select="id">
        <keyed_name>@5</keyed_name>
      </Item>
    </managed_by_id>
    <classification>@6</classification>
    <authoring_tool>@7</authoring_tool>
    <authoring_tool_version>@8</authoring_tool_version>
    <Relationships>
      <Item type="Document File" action="add">
        <related_id>
          <Item type="File" action="add">
            <actual_filename>@9</actual_filename>
            <filename>@10</filename>
            <Relationships>
              <Item type="Located" action="add">
                <related_id>67BBB9204FE84A8981ED8313049BA06C</related_id>
              </Item>
            </Relationships>
          </Item>
        </related_id>
      </Item>
    </Relationships>
  </Item>
</BatchLoaderPrototype>
```