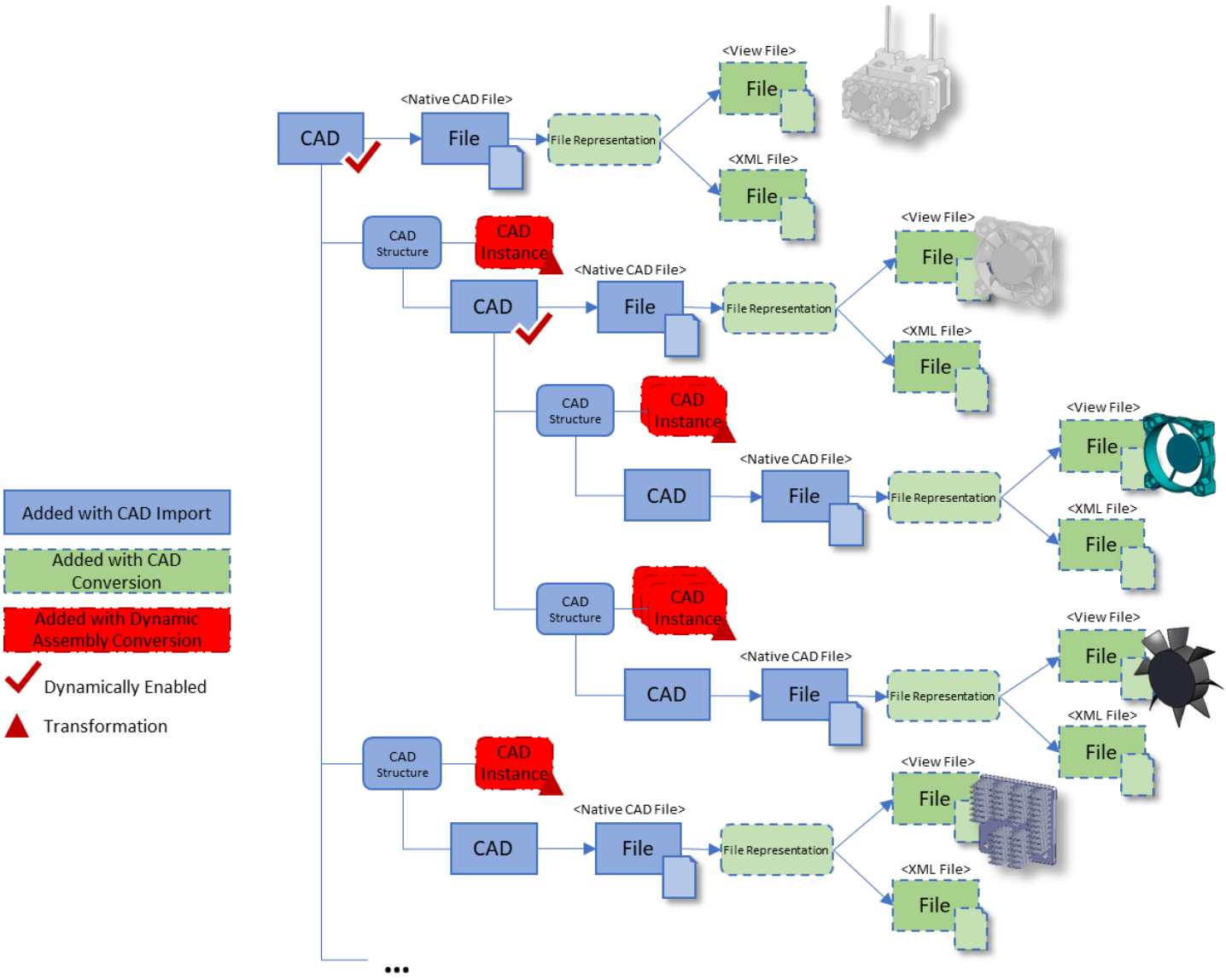
Aras 3D Viewers

Dynamic Viewer

This section describes the Dynamic Viewer and compares its features with the Monolithic Viewer. It also provides an overview of the assumed ItemType data model, from which the data required by Dynamic Viewer is queried.

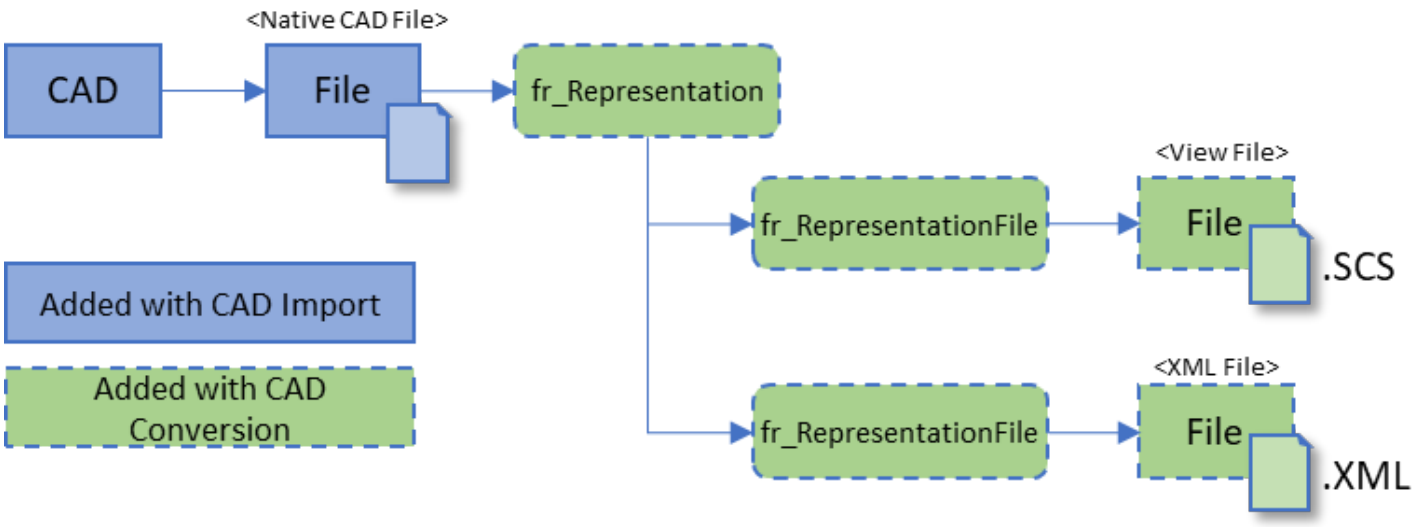
CAD Data Model

Before reviewing the Dynamic Viewer features, it is important to understand the **CAD** ItemType data model, so it is clear what data is required to support Dynamic Visualization and where this information is stored and retrieved by default. This section describes the process of creating **CAD** Items, their related ItemTypes, and **CADFile** conversions using the following diagram.



The **CAD** Items and **CAD Structure** Relationship Items are typically created by a CAD Connector when CAD native files are checked into the Aras Innovator. A **CAD** Item represents a single 3D CAD assembly, part, or 2D drawing. For this document, **CAD** Items representing 3D CAD assemblies and parts are assumed. In this case, the native CAD file is attached (via the **File** ItemType property) to each **CAD** Item. If this **CAD** Item represents an assembly, all related subassemblies and parts will be included in separate **CAD** Items and referenced via **CAD Structure** Items. The **CAD** and **CAD Structure** Items represent a mechanical BOM.

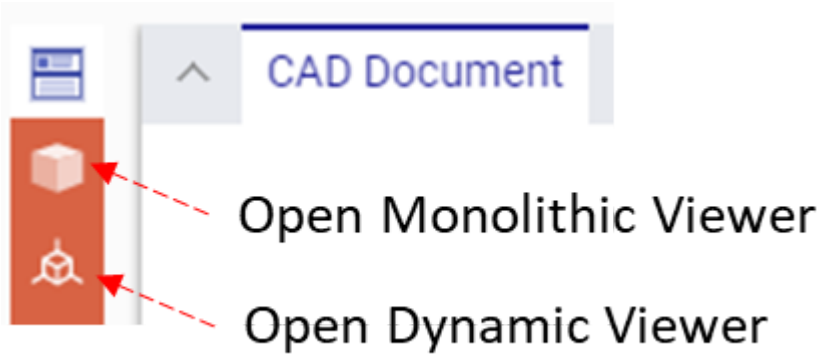
CAD Conversion is the process of generating alternate versions of native CAD data. It is triggered by the existence of a CAD file. ^[1] The Monolithic and Dynamic Viewers render 3D component geometry stored in a Stream Cache Single (SCS) view file. Note that thumbnail images, PRC, 3D PDF, and other converted formats can also be created as part of the CAD Conversion process and stored in a **CAD** Item or its related Items. An XML file is also created and stored. This XML file is generated from the conversion process and used to identify all the instances of parts and subassemblies included in a CAD assembly. This information is used to map 3D component geometry in the Monolithic Viewer and generate CAD Instances and 3D Transformations for the Dynamic Viewer. The SCS view and XML files are stored as related Items attached to a **File** Item for the native file. Note that the figure below uses a single Relationship graphic labeled **File Representation** to simplify the diagram. However, File Representations use two RelationshipTypes.



For the Dynamic Viewer, the conversion process appends the **CAD Instance** Items for all instances of an associated (related) **CAD** Item. **CAD Instances** contain a property that stores the 3D Transformation information as a 4X4 matrix. In addition, the added **Dynamic Enabled** Boolean property is set to **true** for a **CAD** Item with an assembly, when all data necessary to render that CAD assembly using the Dynamic Viewer has been created. Note that these Instances and their transformation information are created automatically by the 3D Conversion Process based on information extracted from a native CAD assembly file.

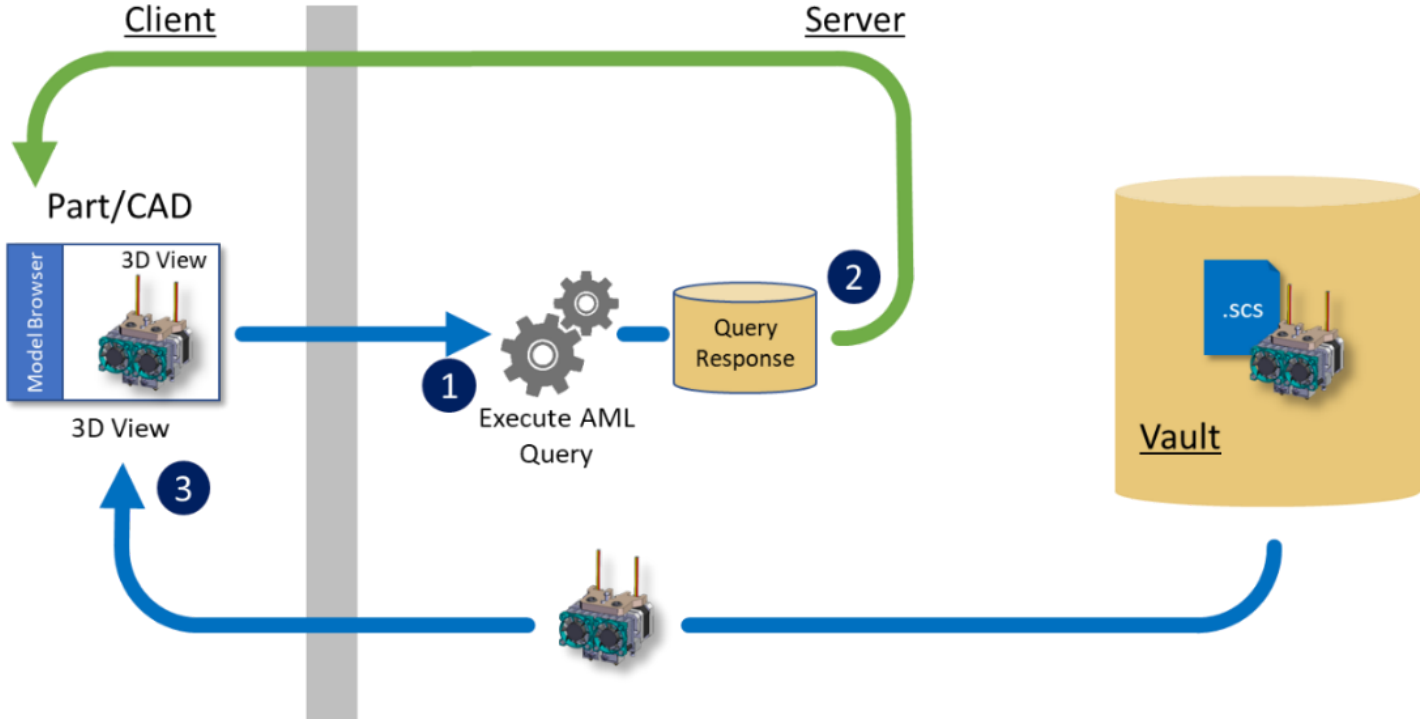
Monolithic vs Dynamic Viewer

This section compares the existing features of the Monolithic and Dynamic Viewers. The below figure shows the buttons on the sidebar of a **CAD** Item form that opens the Viewers.



Monolithic Viewer Process Overview

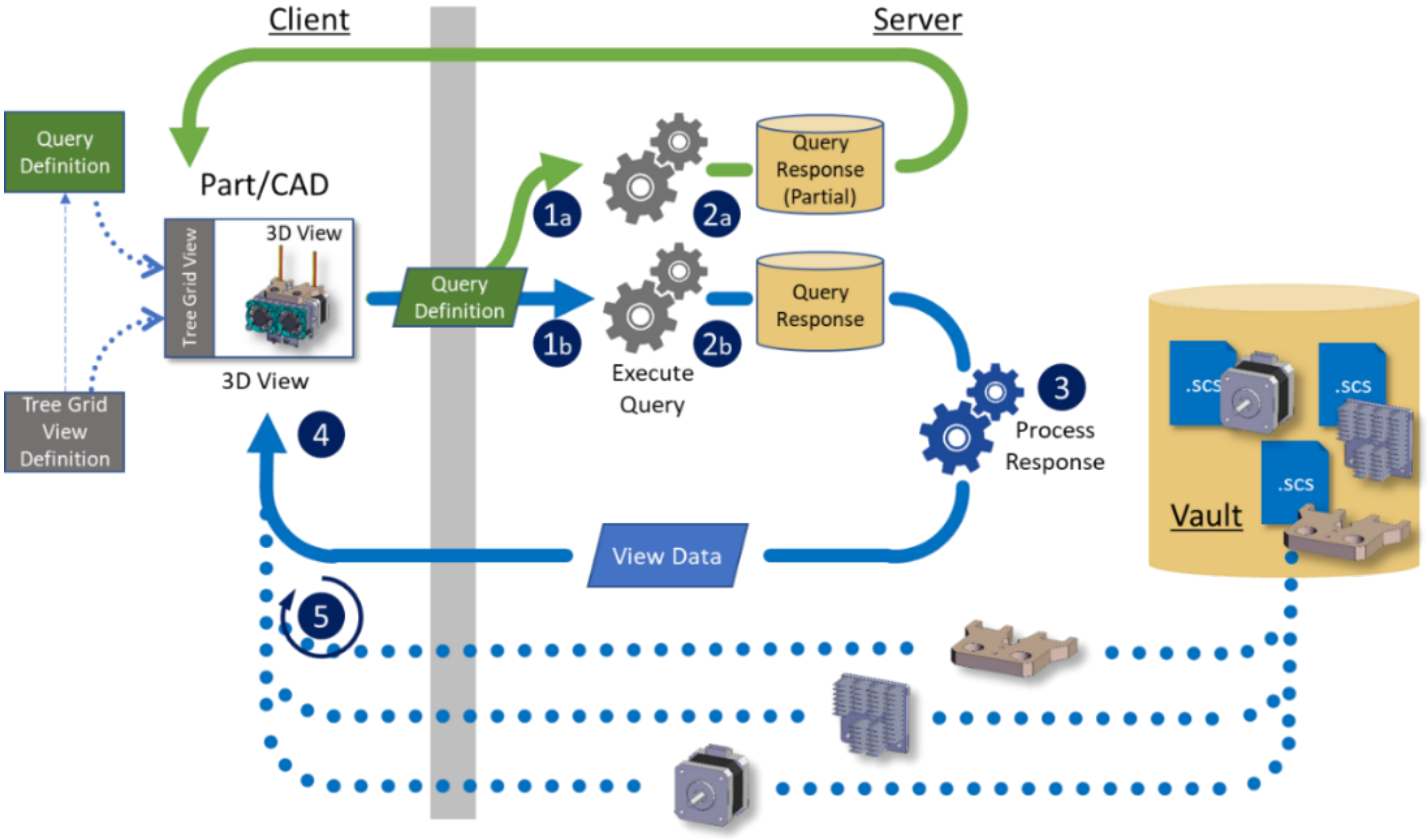
The Monolithic Viewer displays a single SCS view file attached to a corresponding **CAD** Item. If it is an assembly view file, the monolithic view file includes the 3D component geometry for all subassemblies and parts included in this assembly at the time the CAD file was checked in. In essence, it is a static view of the assembly because it will not include any geometry changes to subcomponents made after the assembly view file was created. The following figure shows the process of rendering the view file in the Monolithic Viewer.



When the Monolithic Viewer is opened, a static AML query is executed to retrieve the multi-level CAD structure of an opened **CAD** Item. The results are displayed in a tree user interface called the simple Model Browser of the Viewer. A request is made to load a single SCS view file associated with the opened **CAD** Item, the 3D component geometry of which is displayed in the Viewer. The XML data generated from the conversion process is also retrieved, and the information contained within it is used to map the Items displayed in the simple Model Browser with the associated 3D components in the 3D View scene. This provides selection synchronization between the Model Browser and the 3D View.

Dynamic Viewer Process Overview

The Dynamic Viewer is used only for assemblies and displays SCS view files as determined by the results returned from the execution of an associated Query Definition (QD). Unlike a monolithic view file for an assembly, the Query results for a dynamic Query target the view files for assembly parts and the instance and transformation data required to properly position each file given the returned assembly hierarchy. The following figure shows the process of rendering the view file in the Dynamic Viewer.



The Dynamic Viewer consists of two main components: Tree Grid View (TGV) and the 3D View. The TGV displays the results of the associated QD as defined by the chosen Tree Grid View Definition. The Tree Grid View is a composite of a Tree View and a Table (or Grid). The left-most column contains a hierarchical Tree View showing all the related contents starting from (rooted by) the **CAD** Item from which the Dynamic View was opened.

Upon refreshing the View, the system executes the associated QD in two simultaneous operations. The first executes the Query to populate the TGV. The second executes the Query and processes the complete response.

The TGV uses the partial response based on the configuration of the associated TGV Definition.[2]. By default, the TGV is lazy-loaded: only a portion is returned to a client and displayed. The full response is processed to generate the view data so that all component parts are identified.

While processing the Query results for the 3D View, XML data is constructed that identifies the assemblies, parts, part instances, and their transformations for each. Parts will have links to the respective view files in the Aras Innovator Vault.

The 3D Viewer processes the view data (XML) and sends subsequent requests to the Aras Innovator Vault to retrieve each of the corresponding view files to render. The 3D Viewer processes and renders the view files individually and in sequence.