Overview

The Technical Documentation Framework is a content authoring tool, fully integrated into Aras Innovator PLM that collectively provides component content management capability for topics-based, modular documentation. The editor enables users of various disciplines to author, visualize, share, and publish information all within a central, controlled, and collaborative environment. Unlike other systems that provide Document-level Management functions for content produced by external applications, the Technical Documentation Framework leverages the Aras Innovator core PLM functionality (Access Control, Workflow/Lifecycle, Change Notification, etc.) and provides a web-based authoring tool for creating content directly, embedding shared document components, or referencing other elements managed by the PLM system. So as related Documents, Parts, Project Tasks, Requirements, or any other information managed by Aras Innovator evolve through subsequent revisions, authors of technical documentation can be alerted and navigated to pending changes, or the system can update the related documentation automatically.

Technical Documents can be composed of individual components (sections, paragraphs, illustrations, tables, etc.), where each component can change independently; and optionally created/managed by multiple users. The aggregated information forms the complete published document. This Structured Content is specified by a user-defined data model that identifies the elements of each component, their relationships and cardinality, and any restrictions on data type and value limitations. The built-in authoring User Interface directs the author as to the types of document elements that can be inserted, constantly validating the document as it is created. Candidate documents for release can be routed through workflow and published on final acceptance.

This document describes the functions and features of the Technical Documentation Framework. It is meant to be a reference for the Technical Document Author (Section 5.2.1) as they use the software to author technical content. Section 1 provides an overview of the features of the software. Section 2 describes the User Interface. Section 3 describes how you use the software to create technical content. Section 4 describes how to publish/export content in various formats.

Terms

The following Terms are used throughout this document:

Term	Definition
Business Object	Refers to any Item in Innovator that contains information which may be referred to in a Technical Document.
Cascading Style Sheets (CSS)	Industry standard text-based format for controlling the position, color, size, etc. of web-based content. It is used by the Technical Documentation Framework to control the format of HTML and PDF content.
Content Filters	Metadata (name/value pairs) that are defined by the Technical Document Administrator and chosen by the Technical Document Author to characterize Document Elements within a Technical Document.
Context Menus	Pop-up menus, used mostly by Document Elements, that contain menu items used to invoke various functions on the Document Element instance. The term ‘context’ is used because the specific menu items displayed in the menu may be different depending on the specific Document Element instance selected.
Document Element	Refers to the components that make up a Technical Document. Every Document Element is associated with a specific node in the Navigation Pane and XML Element in the resulting content. Document Elements are explicitly defined in the Technical Document Type Definition.
Document Style	Refers to the formatting and positional logic that defines the placement and look-and-feel of the rendered content of a Technical Document. Document Style is controlled using Cascading Style Sheets (CSS), associated with a specific Document Type, and defined by the Technical Document Administrator.
Graphic	Any 2D image that can be displayed in a Technical Document.
Mapped Property	Refers to a Document Element with content that is automatically created from a Property of a referenced Item
Structured Content	Refers to the general use of an underlying schema to define the content of a document. Structured Content is contrasted with unstructured (or freeform) content in which case there are no rules that help define and control the content.
Technical Document	Refers to instances of a Technical Document ItemType. In this case, the term is capitalized. When lower case, the context is more general – any technical document.
Technical Document Type	Refers to instances of a Document Type Item, which is used to configure the Structured Content and Document Style for a Technical Document.

Features

This section provides a brief overview of each of the main features of the Technical Documentation Framework with links to associated sections for more information.

Structured Documents and Modules

Technical Documents can exist either as a single entity - with all content contained within – or aggregate content from multiple document *components*. Breaking up documents into smaller components has multiple advantages including the ability to:

1. Reuse each component.
2. Enable concurrent editing.
3. Apportion of each component to areas of specific specialty or expertise

In the Technical Publication domain, technical content is sometimes referred to as ‘Topics-based’ or ‘Modular’. This refers to content that is about a specific topic or area of focus and is, to some degree, context insensitive. This helps to characterize the content and enable it to exist in multiple places within a document. Contrast this with a narrative in other forms of documents where every paragraph builds upon (and thus depends on) previous paragraphs and context.

Each Technical Document instance can evolve independently and be protected by specific access rights. To be reusable, each document component needs to be constructed in a way that allows it to fit within a larger context. To achieve this, the content of each Document / Document Component will be based on an overarching schema or document structure (Section 3.1). This document structure helps ensure consistency and provides a mechanism for determining how individual components can fit into a larger context.

Document Content Styling

The style and layout of Technical Documents is dictated by a separate style configuration that is maintained separately but associated directly with a Technical Document Type definition. Each Document Element defined in the Schema can be referenced and have a unique set of styling parameters that are used to position and render the associated Document Element instances in each document. Style settings can be assigned to HTML output, PDF Output, and the Document Editor.

Referencing centrally controlled style settings ensure consistency in look and feel for published content and provides a convenient mechanism to update style in one location and have it automatically apply to all associated documents.

Metadata

All Document Element content can have additional metadata information assigned to provide semantics for content classification and aid in content search and retrieval. This metadata is provided

by the Technical Document Administrator when the Document Structure is designed and configured.

Item Referencing

Technical Documents can have explicit relationships created with other Items maintained in Innovator – Business Objects (Section 3.2.7). This relationship establishes a direct binding between the document and associated Business Objects. In addition, property data from these Business Objects can be queried and used within the Technical Document (Section 3.7). When the Business Object property data is updated, the associated Technical Documents are updated accordingly. This binding helps maintain data integrity and identify a hard link across Objects and the documentation that was created specifically for them.

Conditional Content and Views

Technical documentation content should evolve as the information it references evolves. Typically, product information does not change substantially between versions. As a result, documentation about each version should conserve and reuse all that is common and identify the areas that are unique to one or more product versions. The Technical Documentation Framework provides a mechanism to tag elements within a document with one or more characteristics, a particular version or model number for example. These characteristics can then be used to filter (Section 3.3) document content to only display or include information that is relevant.

Publication and Data Export

The Technical Documentation Framework, together with Aras Innovator, provides an authoring and data management environment. To make use of the content created, it needs to be exposed outside the PLM environment. The publishing/export function (Section 4) provides an ability to convert the content of each Technical Document to a format more suitable for end-user consumption.