W3C: XSL Transformations (XSLT) Version 3.0 (acronym)

return to the W3C Standards

Table 1: Data sheet for XSL Transformations (XSLT) Version 3.0

Title	XSL Transformations (XSLT) Version 3.0
Acronym	XSLT
Version	3.0
Series	TR
Document Number	
Release Date	8 June 2017
Download	https://www.w3.org/TR/2017/REC-xslt-30-20170608/

Note: The following is an excerpt from the official NIST catalog. It is provided here as a connivence and is not authoritative. Refer to the original document as the authoritative reference.

Introduction

What is XSLT?

This specification defines the syntax and semantics of the XSLT 3.0 language.

A transformation in the XSLT language is expressed in the form of a stylesheet. A stylesheet is made up of one or more well-formed XML documents conforming to the Namespaces in XML Recommendation.

A stylesheet generally includes elements that are defined by XSLT as well as elements that are not defined by XSLT. XSLT-defined elements are distinguished by use of the namespace `http://www.w3.org/1999/XSL/Transform` (see 3.1 XSLT Namespace), which is referred to in this specification as the XSLT namespace. Thus this specification is a definition of the syntax and semantics of the XSLT namespace.

The term stylesheet reflects the fact that one of the important roles of XSLT is to add styling information to an XML source document, by transforming it into a document consisting of XSL formatting objects (see XSL-FO), or into another presentation-oriented format such as HTML, XHTML, or SVG. However, XSLT is used for a wide range of transformation tasks, not exclusively for formatting and presentation applications.

A transformation expressed in XSLT describes rules for transforming input data into output data. The inputs and outputs will all be instances of the XDM data model, described in XDM 3.0. In the simplest and most common case, the input is an XML document referred to as the source tree, and the output is an XML document referred to as the result tree. It is also possible to process multiple source documents, to generate multiple result documents, and to handle formats other than XML. The transformation is achieved by a set of template rules. A template rule associates a pattern, which typically matches nodes in the source document, with a sequence constructor. In many cases, evaluating the sequence constructor will cause new nodes to be constructed, which can be used to produce part of a result tree. The structure of the result trees can be completely different from the structure of the source trees. In constructing a result tree, nodes from the source trees can be filtered and reordered, and arbitrary structure can be added. This mechanism allows a stylesheet to be applicable to a wide class of documents that have similar source tree structures.

Stylesheets have a modular structure; they may contain several packages developed independently of each other, and each package may consist of several stylesheet modules.