Product Description for Saxon-HE (Home Edition)

This document lists the features provided by Saxon 9.8 open source Home Edition (Saxon-HE).

This document does not form part of any contract unless expressly incorporated.

Language Support

1. XSLT (Transformation Processing)

1.1 XSLT 3.0 (Basic)

Provides all features from the XSLT 3.0 specification (including try/catch, iterate, accumulators, maps, named modes, content value templates, and extended patterns, as well as serialization and XPath 3.1 features, and from 9.8.0.7 also XSLT 1.0 compatibility); with the notable exceptions of schema-awareness, streaming, packages, dynamic evaluation, and higher-order functions.

For more details see: XSLT 3.0 conformance.

Relevant W3C Specification: XSLT 3.0 Recommendation (08 June 2017).

2. XPath

2.1 XPath 3.1 (Basic)

Provides all XPath 3.1 features other than schema-awareness and higher-order functions. This includes an implementation of maps and arrays, and support for JSON.

For more details see: XPath 3.1 conformance.

Relevant W3C Specification: XPath 3.1 Recommendation (21 March 2017).

3. XQuery

3.1 XQuery 3.1 (Basic)

Provides the following features, as defined in section 5 of the XQuery 3.1 specification: Minimal Conformance (including try/catch and "group-by"); Module Feature; and Serialization Feature.

For more details see: XQuery 3.1 conformance.

Relevant W3C Specification: XQuery 3.1 Recommendation (21 March 2017).

Performance Features

4. Optimizer (Basic)

The Basic optimizer provided with all Saxon editions provides a wide range of static and dynamic optimizations including full pipelining of list operations, lazy evaluation of variables, elimination of redundant sorting operations, etc.

5. Reading W3C schemas and DTDs

The W3C web server now routinely rejects requests for commonly-referenced files such as the DTD for XHTML, causing parsing failures. In response to this, Saxon now includes copies of these documents within the issued JAR file, and recognizes requests for these documents, satisfying the request using the local copy.

Extensibility

6. Extensibility using custom classes

Ability to write extension functions (for use in XSLT, XQuery, or XPath) by implementing a Saxon-defined interface and registering the implementation with the Saxon Configuration.

For more details see: Extensibility.

Localization

7. Localization (Basic)

Run-time localization support for formatting of dates and numbers, and sorting and comparison of strings, building on the capabilities of the Java Virtual Machine.

For more details see: Unicode collation, Localizing numbers and dates.

Interfaces and APIs

8. JAXP API

Implementations of the standard JAXP interfaces for XSLT transformation, XPath evaluation, and XML Schema validation. Applies to the Java platform only.

For more details see: JAXP API conformance.

9. S9API API

Saxon's native interface for processing XSLT, XQuery, XPath, and XML Schema. Available in slightly different forms on the Java and .NET platforms.

10. Support for DOM

Ability to use a DOM (Document Object Model) for the input and output of transformations and queries. On the .NET platform this includes the System.XML DOM classes.

For more details see: Object models.

11. Support for JDOM, JDOM2, AXIOM, DOM4J, and XOM (not included, but open source)

Although the code for these interfaces does not come packaged with the Saxon-HE download, it is open source and can be compiled to work with Saxon-HE. This provides the ability to use a JDOM, JDOM2, AXIOM, DOM4J, and XOM for the input or output of transformations and queries. Applies to the Java platform only.

For more details see: Object models.

12. XQJ API (not included, but open source)

Implementations of the standard XQJ interfaces for XQuery processing have been removed from the Saxon-HE download because the Oracle specification license is not open source, but they are available on request. Applies to the Java platform only.

For more details see: XQJ API conformance.