[MS-SQLXQUERY]:
SQL XQuery Standards Support Document

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft Open Specification Promise or the Community Promise. If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights.
- **Fictitious Names.** The example companies, organizations, products, domain names, email addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

Reservation of Rights. All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

Revision Summary

Date	Revision History	Revision Class	Comments
11/03/2011	0.1	New	Released new document.
01/19/2012	0.1	No change	No changes to the meaning, language, or formatting of the technical content.
02/23/2012	0.1	No change	No changes to the meaning, language, or formatting of the technical content.
03/27/2012	0.1	No change	No changes to the meaning, language, or formatting of the technical content.
05/24/2012	0.1	No change	No changes to the meaning, language, or formatting of the technical content.
06/29/2012	0.1	No change	No changes to the meaning, language, or formatting of the technical content.
07/16/2012	0.1	No change	No changes to the meaning, language, or formatting of the technical content.
10/08/2012	0.1	No change	No changes to the meaning, language, or formatting of the technical content.

Contents

1 Introduction 11
1.1 Glossary 11
1.2 Microsoft Implementations 11
1.3 References 11
1.3.1 Normative References 11
1.3.2 Informative References 12
1.4 Standards Support Requirements 12
1.5 Notation 12

2 Standards Support Statements 13
2.1 Normative Variations 13
2.1.1 Data Model Items 13
2.1.1.1 [XDM/2] Section 1 Introduction 13
2.1.1.2 Concepts 13
2.1.1.2.1 [XDM/2] Section 2.4 Document Order 13
2.1.1.3 Data Model Construction 13
2.1.1.3.1 [XDM/2] Section 3 Data Model Construction 13
2.1.1.3.2 [XDM/2] Section 3.1 Direct Construction 13
2.1.1.3.3 [XDM/2] Section 3.2 Construction from an Infoset 13
2.1.1.3.4 [XDM/2] Section 3.3 Construction from a PSVI 13
2.1.1.3.5 [XDM/2] Section 3.3.1 Mapping PSVI Additions to Node Properties 14
2.1.1.3.6 [XDM/2] Section 3.3.1.1 Element and Attribute Node Type Names 14
2.1.1.3.7 [XDM/2] Section 3.3.1.2 Typed Value Determination 14
2.1.1.3.8 [XDM/2] Section 3.3.1.3 Relationship Between Typed Value and String Value 14
2.1.1.3.9 [XDM/2] Section 3.3.1.4 Pattern Facets 14
2.1.1.3.10 [XDM/2] Section 3.3.3 QNames and NOTATIONS 14
2.1.1.4 Accessors 14
2.1.1.4.1 [XDM/2] Section5 Accessors 14
2.1.1.5 Nodes 14
2.1.1.5.1 [XDM/2] Section 6.2.4 Construction from a PSVI 14
2.1.1.5.2 [XDM/2] Section 6.4 Namespace Nodes 15
2.1.1.6 Conformance 15
2.1.1.6.1 [XDM/2] Section 7 Conformance 15
2.1.2 Basics 15
2.1.2.1 Expression Context 15
2.1.2.1.1 [XQuery1.0/2] Section 2.1.1 Static Context 15
2.1.2.1.1.1 [XQuery1.0/2] Appendix C.1 Static Context Components 15
2.1.2.1.1.1.1 XPath 1.0 Compatibility Mode 15
2.1.2.1.1.1.2 Statically Known Namespaces 15
2.1.2.1.1.1.3 Default Function Namespace 15
2.1.2.1.1.1.4 In-Scope Schema Types 15
2.1.2.1.1.1.5 In-Scope Element Declarations 15
2.1.2.1.1.1.6 In-Scope Attribute Declarations 16
2.1.2.1.1.1.7 In-Scope Variables 16
2.1.2.1.1.1.8 Function Signatures 16
2.1.2.1.1.1.9 Statically Known Collations 16
2.1.2.1.1.1.10 Default Collation 16
2.1.2.1.1.1.11 Construction Mode 16
2.1.2.1.1.1.12 Ordering Mode 16

2.1.2.1.1.1.13 Default Order for Empty Sequences 16
2.1.2.1.1.1.14 Boundary-Space Policy 16
2.1.2.1.1.1.15 Copy-Namespaces Mode 16
2.1.2.1.1.1.16 Base URI 17
2.1.2.1.1.1.17 Statically Known Documents 17
2.1.2.1.1.1.18 Statically Known Collections 17
2.1.2.1.1.1.19 Statically Known Default Collection Type 17
2.1.2.1.2 [XQuery1.0/2] Section 2.1.2 Dynamic Context 17
2.1.2.1.2.1 [XQuery1.0/2] Appendix C.2 Dynamic Context Components 17
2.1.2.1.2.1.1 Variable Values 17
2.1.2.1.2.1.2 Function Implementations 17
2.1.2.1.2.1.3 Current dateTime 17
2.1.2.1.2.1.4 Implicit Timezone 17
2.1.2.1.2.1.5 Available Documents 17
2.1.2.1.2.1.6 Available Collections 17
2.1.2.1.2.1.7 Default Collection 17
2.1.2.2 Processing Model 18
2.1.2.2.1 [XQuery1.0/2] Section 2.2.1 Data Model Generation 18
2.1.2.2.2 [XQuery1.0/2] Section 2.2.2 Schema Import Processing 18
2.1.2.2.3 [XQuery1.0/2] Section 2.2.3.1 Expression Processing - Static Analysis Phase 18
2.1.2.2.4 [XQuery1.0/2] Section 2.2.3.2 Expression Processing - Dynamic Evaluation Phase 18
2.1.2.2.5 [XQuery1.0/2] Section 2.2.4 Serialization 18
2.1.2.2.6 [XQuery1.0/2] Section 2.2.5 Consistency Constraints 18
2.1.2.3 Error Handling 18
2.1.2.3.1 [XQuery1.0/2] Section 2.3.1 Kinds of Errors 18
2.1.2.3.2 [XQuery1.0/2] Section 2.3.2 Identifying and Reporting Errors 18
2.1.2.3.3 [XQuery1.0/2] Section 2.3.3 Handling Dynamic Errors 19
2.1.2.3.4 [XQuery1.0/2] Section 2.3.4 Errors and Optimization 19
2.1.2.4 Concepts 19
2.1.2.4.1 [XQuery1.0/2] Section 2.4.1 Document Order 19
2.1.2.4.2 [XQuery1.0/2] Section 2.4.2 Atomization 19
2.1.2.4.3 [XQuery1.0/2] Section 2.4.3 Effective Boolean Value 19
2.1.2.4.4 [XQuery1.0/2] Section 2.4.4 Input Sources 19
2.1.2.4.5 [XQuery1.0/2] Section 2.4.5 URI Literals 19
2.1.2.5 Types 19
2.1.2.5.1 [XQuery1.0/2] Section 2.5.1 Predefined Schema Types 19
2.1.2.5.2 [XQuery1.0/2] Section 2.5.3 SequenceType Syntax 19
2.1.2.5.3 [XQuery1.0/2] Section 2.5.4 SequenceType Matching 22
2.1.2.5.4 [XQuery1.0/2] Section 2.5.4.1 Matching a Sequence Type and a Value 22
2.1.2.5.5 [XQuery1.0/2] Section 2.5.4.2 Matching an Item Type and an Item 22
2.1.2.5.6 [XQuery1.0/2] Section 2.5.4.3 Element Test 22
2.1.2.5.7 [XQuery1.0/2] Section 2.5.4.4 Schema Element Test 22
2.1.2.5.8 [XQuery1.0/2] Section 2.5.4.5 Attribute Test 22
2.1.2.5.9 [XQuery1.0/2] Section 2.5.4.6 Schema Attribute Test 23
2.1.3 Data Types 23
2.1.3.1 XSD Types 23
2.1.3.1.1 xs:ID 23
2.1.3.1.2 xs:IDREF 23
2.1.3.1.3 xs:untypedAtomic 23
2.1.3.1.4 xs:untyped 23
2.1.3.1.5 xs:NOTATION 23

2.1.3.1.6 xs:yearMonthDuration 23
2.1.3.1.7 xs:dayTimeDuration 23
2.1.3.1.8 xs:IDREFS 23
2.1.3.2 Casting Primitive Type to Primitive Type 23
2.1.4 Expressions 24
2.1.4.1 Expression Syntax 24
2.1.4.1.1 [XQuery1.0/2] Section 3 Expressions 24
2.1.4.1.2 [XQuery1.0/2] Section 3.1 Primary Expressions 24
2.1.4.1.3 [XQuery1.0/2] Section 3.1.1 Literals 24
2.1.4.1.4 [XQuery1.0/2] Section 3.1.2 Variable References 25
2.1.4.1.5 [XQuery1.0/2] Section 3.1.3 Parenthesized Expressions 26
2.1.4.1.6 [XQuery1.0/2] Section 3.1.4 Context Item Expression 26
2.1.4.1.7 [XQuery1.0/2] Section 3.1.5 Function Calls 26
2.1.4.1.8 [XQuery1.0/2] Section 3.2 Path Expressions 26
2.1.4.1.9 [XQuery1.0/2] Section 3.2.1 Steps 26
2.1.4.1.10 [XQuery1.0/2] Section 3.2.1.1 Axes 27
2.1.4.1.11 [XQuery1.0/2] Section 3.2.1.2 Node Tests 27
2.1.4.1.12 [XQuery1.0/2] Section 3.2.2 Predicates 28
2.1.4.1.13 [XQuery1.0/2] Section 3.2.3 Unabbreviated Syntax 28
2.1.4.1.14 [XQuery1.0/2] Section 3.2.4 Abbreviated Syntax 28
2.1.4.1.15 [XQuery1.0/2] Section 3.3.1 Constructing Sequences 28
2.1.4.1.16 [XQuery1.0/2] Section 3.3.2 Filter Expressions 28
2.1.4.1.17 [XQuery1.0/2] Section 3.3.3 Combining Node Sequences 29
2.1.4.1.18 [XQuery1.0/2] Section 3.4 Arithmetic Expressions 29
2.1.4.1.19 [XQuery1.0/2] Section 3.5 Comparison Expressions 30
2.1.4.1.20 [XQuery1.0/2] Section 3.5.1 Value Comparisons 30
2.1.4.1.21 [XQuery1.0/2] Section 3.5.2 General Comparisons 30
2.1.4.1.22 [XQuery1.0/2] Section 3.5.3 Node Comparisons 30
2.1.4.1.23 [XQuery1.0/2] Section 3.6 Logical Expressions 30
2.1.4.1.24 [XQuery1.0/2] Section 3.7 Constructors 31
2.1.4.1.25 [XQuery1.0/2] Section 3.7.1 Direct Element Constructors 33
2.1.4.1.26 [XQuery1.0/2] Section 3.7.1.1 Attributes 33
2.1.4.1.27 [XQuery1.0/2] Section 3.7.1.2 Namespace Declaration Attributes 33
2.1.4.1.28 [XQuery1.0/2] Section 3.7.1.3 Content 33
2.1.4.1.29 [XQuery1.0/2] Section 3.7.1.4 Boundary Whitespace 33
2.1.4.1.30 [XQuery1.0/2] Section 3.7.2 Other Direct Constructors 33
2.1.4.1.31 [XQuery1.0/2] Section 3.7.3 Computed Constructors 34
2.1.4.1.32 [XQuery1.0/2] Section 3.7.3.1 Computed Element Constructors 34
2.1.4.1.33 [XQuery1.0/2] Section 3.7.3.2 Computed Attribute Constructors 34
2.1.4.1.34 [XQuery1.0/2] Section 3.7.3.3 Document Node Constructors 34
2.1.4.1.35 [XQuery1.0/2] Section 3.7.3.4 Text Node Constructors 35
2.1.4.1.36 [XQuery1.0/2] Section 3.7.3.5 Computed Processing Instruction Constructors 35
2.1.4.1.37 [XQuery1.0/2] Section 3.7.3.6 Computed Comment Constructors 35
2.1.4.1.38 [XQuery1.0/2] Section 3.7.4 In-scope Namespaces of a Constructed Element 35
2.1.4.1.39 [XQuery1.0/2] Section 3.8 FLWOR Expressions 35
2.1.4.1.40 [XQuery1.0/2] Section 3.8.1 For and Let Clauses 36
2.1.4.1.41 [XQuery1.0/2] Section 3.8.2 Where Clause 37
2.1.4.1.42 [XQuery1.0/2] Section 3.8.3 Order By and Return Clauses 37
2.1.4.1.43 [XQuery1.0/2] Section 3.9 Ordered and Unordered Expressions 37
2.1.4.1.44 [XQuery1.0/2] Section 3.10 Conditional Expressions 37
2.1.4.1.45 [XQuery1.0/2] Section 3.11 Quantified Expressions 37

2.1.4.1.46 [XQuery1.0/2] Section 3.12.1 Instance Of 37
2.1.4.1.47 [XQuery1.0/2] Section 3.12.2 Typeswitch 38
2.1.4.1.48 [XQuery1.0/2] Section 3.12.3 Cast 38
2.1.4.1.49 [XQuery1.0/2] Section 3.12.4 Castable 38
2.1.4.1.50 [XQuery1.0/2] Section 3.12.5 Constructor Functions 39
2.1.4.1.51 [XQuery1.0/2] Section 3.12.6 Treat 39
2.1.4.1.52 [XQuery1.0/2] Section 3.13 Validate Expressions 39
2.1.4.1.53 [XQuery1.0/2] Section 3.14 Extension Expressions 39
2.1.4.1.54 [XQuery1.0/2] Appendix A.1 EBNF 39
2.1.4.2 Modules and Prologs 40
2.1.4.2.1 [XQuery1.0/2] Section 4 Modules and Prologs 40
2.1.4.2.2 [XQuery1.0/2] Section 4.1 Version Declaration 41
2.1.4.2.3 [XQuery1.0/2] Section 4.2 Module Declaration 41
2.1.4.2.4 [XQuery1.0/2] Section 4.3 Boundary-space Declaration 41
2.1.4.2.5 [XQuery1.0/2] Section 4.4 Default Collation Declaration 41
2.1.4.2.6 [XQuery1.0/2] Section 4.5 Base URI Declaration 41
2.1.4.2.7 [XQuery1.0/2] Section 4.6 Construction Declaration 41
2.1.4.2.8 [XQuery1.0/2] Section 4.7 Ordering Mode Declaration 42
2.1.4.2.9 [XQuery1.0/2] Section 4.8 Empty Order Declaration 42
2.1.4.2.10 [XQuery1.0/2] Section 4.9 Copy-Namespaces Declaration 42
2.1.4.2.11 [XQuery1.0/2] Section 4.10 Schema Import 42
2.1.4.2.12 [XQuery1.0/2] Section 4.11 Module Import 43
2.1.4.2.13 [XQuery1.0/2] Section 4.12 Namespace Declaration 43
2.1.4.2.14 [XQuery1.0/2] Section 4.13 Default Namespace Declaration 43
2.1.4.2.15 [XQuery1.0/2] Section 4.14 Variable Declaration 43
2.1.4.2.16 [XQuery1.0/2] Section 4.15 Function Declaration 43
2.1.4.2.17 [XQuery1.0/2] Section 4.16 Option Declaration 44
2.1.5 Operators 44
2.1.5.1 Numeric Operators 44
2.1.5.1.1 [XQuery1.0XPath2.0/2] Section 6.2.4 op:numeric-divide 44
2.1.5.1.2 [XQuery1.0XPath2.0/2] Section 6.2.5 op:numeric-integer-divide 44
2.1.5.2 String Operators 44
2.1.5.3 anyURI Operators 44
2.1.5.4 Boolean Operators 45
2.1.5.4.1 [XQuery1.0XPath2.0/2] Section 9.2.2 op:boolean-less-than 45
2.1.5.4.2 [XQuery1.0XPath2.0/2] Section 9.2.3 op:boolean-greater-than 45
2.1.5.5 Durations, Dates, and Times Operators 45
2.1.5.5.1 [XQuery1.0XPath2.0/2] Section 10.4.1 op:yearMonthDuration-less-than ... 45
2.1.5.5.2 [XQuery1.0XPath2.0/2] Section 10.4.2 op:yearMonthDuration-greater-than 45
2.1.5.5.3 [XQuery1.0XPath2.0/2] Section 10.4.3 op:dayTimeDuration-less-than 45
2.1.5.5.4 [XQuery1.0XPath2.0/2] Section 10.4.4 op:dayTimeDuration-greater-than.. 45
2.1.5.5.5 [XQuery1.0XPath2.0/2] Section 10.4.6 op:dateTime-equal 45
2.1.5.5.6 [XQuery1.0XPath2.0/2] Section 10.4.8 op:dateTime-greater-than 45
2.1.5.5.7 [XQuery1.0XPath2.0/2] Section 10.4.9 op:date-equal 45
2.1.5.5.8 [XQuery1.0XPath2.0/2] Section 10.4.10 op:date-less-than 45
2.1.5.5.9 [XQuery1.0XPath2.0/2] Section 10.4.12 op:time-equal 46
2.1.5.5.10 [XQuery1.0XPath2.0/2] Section 10.6.1 op:add-yearMonthDurations 46
2.1.5.5.11 [XQuery1.0XPath2.0/2] Section 10.6.2 op:subtract-yearMonthDurations.. 46
2.1.5.5.12 [XQuery1.0XPath2.0/2] Section 10.6.3 op:multiply-yearMonthDuration ... 46
2.1.5.5.13 [XQuery1.0XPath2.0/2] Section 10.6.4 op:divide-yearMonthDuration 46
2.1.5.5.14 [XQuery1.0XPath2.0/2] Section 10.6.5 op:divide-yearMonthDuration-by-yearMonthDuration 46

2.1.5.5.15 [XQuery1.0XPath2.0/2] Section 10.6.6 op:add-dayTimeDurations 46
2.1.5.5.16 [XQuery1.0XPath2.0/2] Section 10.6.7 op:subtract-dayTimeDurations..... 46
2.1.5.5.17 [XQuery1.0XPath2.0/2] Section 10.6.8 op:multiply-dayTimeDuration 46
2.1.5.5.18 [XQuery1.0XPath2.0/2] Section 10.6.9 op:divide-dayTimeDuration 46
2.1.5.5.19 [XQuery1.0XPath2.0/2] Section 10.6.10 op:divide-dayTimeDuration-by-dayTimeDuration..... 47
2.1.5.5.20 [XQuery1.0XPath2.0/2] Section 10.8.4 op:add-yearMonthDuration-to-dateTime..... 47
2.1.5.5.21 [XQuery1.0XPath2.0/2] Section 10.8.5 op:add-dayTimeDuration-to-dateTime..... 47
2.1.5.5.22 [XQuery1.0XPath2.0/2] Section 10.8.6 op:subtract-yearMonthDuration-from-dateTime..... 47
2.1.5.5.23 [XQuery1.0XPath2.0/2] Section 10.8.7 op:subtract-dayTimeDuration-from-dateTime..... 47
2.1.5.5.24 [XQuery1.0XPath2.0/2] Section 10.8.8 op:add-yearMonthDuration-to-date 47
2.1.5.5.25 [XQuery1.0XPath2.0/2] Section 10.8.9 op:add-dayTimeDuration-to-date 47
2.1.5.5.26 [XQuery1.0XPath2.0/2] Section 10.8.10 op:subtract-yearMonthDuration-from-date..... 47
2.1.5.5.27 [XQuery1.0XPath2.0/2] Section 10.8.11 op:subtract-dayTimeDuration-from-date..... 47
2.1.5.5.28 [XQuery1.0XPath2.0/2] Section 10.8.12 op:add-dayTimeDuration-to-time 48
2.1.5.5.29 [XQuery1.0XPath2.0/2] Section 10.8.13 op:subtract-dayTimeDuration-from-time..... 48
2.1.5.6 Notation Operators 48
2.1.5.6.1 [XQuery1.0XPath2.0/2] Section 13.1.1 op:NOTATION-equal..... 48
2.1.5.7 Sequences Operators 48
2.1.5.7.1 [XQuery1.0XPath2.0/2] Section 15.3.2 op:union..... 48
2.1.5.7.2 [XQuery1.0XPath2.0/2] Section 15.3.3 op:intersect 48
2.1.5.7.3 [XQuery1.0XPath2.0/2] Section 15.3.4 op:except 48
2.1.5.7.4 [XQuery1.0XPath2.0/2] Section 15.5.1 op:to..... 48
2.1.5.8 Context 48
2.1.6 Functions 48
2.1.6.1 Accessors 49
2.1.6.1.1 [XQuery1.0XPath2.0/2] Section 2.1 fn:node-name..... 49
2.1.6.1.2 [XQuery1.0XPath2.0/2] Section 2.2 fn:nilled 49
2.1.6.1.3 [XQuery1.0XPath2.0/2] Section 2.3 fn:string..... 49
2.1.6.1.4 [XQuery1.0XPath2.0/2] Section 2.5 fn:base-uri 49
2.1.6.1.5 [XQuery1.0XPath2.0/2] Section 2.6 fn:document-uri..... 49
2.1.6.2 Numeric Functions 49
2.1.6.2.1 [XQuery1.0XPath2.0/2] Section 6.4.1 fn:abs 49
2.1.6.2.2 [XQuery1.0XPath2.0/2] Section 6.4.2 fn:ceiling 49
2.1.6.2.3 [XQuery1.0XPath2.0/2] Section 6.4.3 fn:floor..... 49
2.1.6.2.4 [XQuery1.0XPath2.0/2] Section 6.4.4 fn:round 49
2.1.6.2.5 [XQuery1.0XPath2.0/2] Section 6.4.5 fn:round-half-to-even 50
2.1.6.3 String Functions 50
2.1.6.3.1 [XQuery1.0XPath2.0/2] Section 7.2.1 fn:codepoints-to-string..... 50
2.1.6.3.2 [XQuery1.0XPath2.0/2] Section 7.2.2 fn:string-to-codepoints..... 50
2.1.6.3.3 [XQuery1.0XPath2.0/2] Section 7.3.2 fn:compare..... 50
2.1.6.3.4 [XQuery1.0XPath2.0/2] Section 7.3.3 fn:codepoint-equal..... 50
2.1.6.3.5 [XQuery1.0XPath2.0/2] Section 7.4.1 fn:concat..... 50

2.1.6.3.6 [XQuery1.0XPath2.0/2] Section 7.4.2 fn:string-join 50
2.1.6.3.7 [XQuery1.0XPath2.0/2] Section 7.4.3 fn:substring 50
2.1.6.3.8 [XQuery1.0XPath2.0/2] Section 7.4.4 fn:string-length 50
2.1.6.3.9 [XQuery1.0XPath2.0/2] Section 7.4.5 fn:normalize-space 51
2.1.6.3.10 [XQuery1.0XPath2.0/2] Section 7.4.6 fn:normalize-unicode 51
2.1.6.3.11 [XQuery1.0XPath2.0/2] Section 7.4.9 fn:translate 51
2.1.6.3.12 [XQuery1.0XPath2.0/2] Section 7.4.10 fn:encode-for-uri 51
2.1.6.3.13 [XQuery1.0XPath2.0/2] Section 7.4.11 fn:iri-to-uri 51
2.1.6.3.14 [XQuery1.0XPath2.0/2] Section 7.4.12 fn:escape-html-uri 51
2.1.6.3.15 [XQuery1.0XPath2.0/2] Section 7.5.1 fn:contains 51
2.1.6.3.16 [XQuery1.0XPath2.0/2] Section 7.5.2 fn:starts-with 51
2.1.6.3.17 [XQuery1.0XPath2.0/2] Section 7.5.3 fn:ends-with 51
2.1.6.3.18 [XQuery1.0XPath2.0/2] Section 7.5.4 fn:substring-before 51
2.1.6.3.19 [XQuery1.0XPath2.0/2] Section 7.5.5 fn:substring-after 51
2.1.6.3.20 [XQuery1.0XPath2.0/2] Section 7.6.2 fn:matches 51
2.1.6.3.21 [XQuery1.0XPath2.0/2] Section 7.6.3 fn:replace 52
2.1.6.3.22 [XQuery1.0XPath2.0/2] Section 7.6.4 fn:tokenize 52
2.1.6.4 anyURI Functions 52
2.1.6.4.1 [XQuery1.0XPath2.0/2] Section 8.1 fn:resolve-uri 52
2.1.6.5 Durations, Dates, and Times Functions 52
2.1.6.5.1 [XQuery1.0XPath2.0/2] Section 10.5.1 fn:years-from-duration 52
2.1.6.5.2 [XQuery1.0XPath2.0/2] Section 10.5.2 fn:months-from-duration 52
2.1.6.5.3 [XQuery1.0XPath2.0/2] Section 10.5.3 fn:days-from-duration 52
2.1.6.5.4 [XQuery1.0XPath2.0/2] Section 10.5.4 fn:hours-from-duration 52
2.1.6.5.5 [XQuery1.0XPath2.0/2] Section 10.5.5 fn:minutes-from-duration 52
2.1.6.5.6 [XQuery1.0XPath2.0/2] Section 10.5.6 fn:seconds-from-duration 52
2.1.6.5.7 [XQuery1.0XPath2.0/2] Section 10.5.7 fn:year-from-dateTime 52
2.1.6.5.8 [XQuery1.0XPath2.0/2] Section 10.5.8 fn:month-from-dateTime 52
2.1.6.5.9 [XQuery1.0XPath2.0/2] Section 10.5.9 fn:day-from-dateTime 52
2.1.6.5.10 [XQuery1.0XPath2.0/2] Section 10.5.10 fn:hours-from-dateTime 53
2.1.6.5.11 [XQuery1.0XPath2.0/2] Section 10.5.11 fn:minutes-from-dateTime 53
2.1.6.5.12 [XQuery1.0XPath2.0/2] Section 10.5.12 fn:seconds-from-dateTime 53
2.1.6.5.13 [XQuery1.0XPath2.0/2] Section 10.5.13 fn:timezone-from-dateTime 53
2.1.6.5.14 [XQuery1.0XPath2.0/2] Section 10.5.14 fn:year-from-date 53
2.1.6.5.15 [XQuery1.0XPath2.0/2] Section 10.5.15 fn:month-from-date 53
2.1.6.5.16 [XQuery1.0XPath2.0/2] Section 10.5.16 fn:day-from-date 53
2.1.6.5.17 [XQuery1.0XPath2.0/2] Section 10.5.17 fn:timezone-from-date 53
2.1.6.5.18 [XQuery1.0XPath2.0/2] Section 10.5.18 fn:hours-from-time 53
2.1.6.5.19 [XQuery1.0XPath2.0/2] Section 10.5.19 fn:minutes-from-time 53
2.1.6.5.20 [XQuery1.0XPath2.0/2] Section 10.5.20 fn:seconds-from-time 53
2.1.6.5.21 [XQuery1.0XPath2.0/2] Section 10.5.21 fn:timezone-from-time 53
2.1.6.5.22 [XQuery1.0XPath2.0/2] Section 10.7.1 fn:adjust-dateTime-to-timezone .. 53
2.1.6.5.23 [XQuery1.0XPath2.0/2] Section 10.7.2 fn:adjust-date-to-timezone 54
2.1.6.5.24 [XQuery1.0XPath2.0/2] Section 10.7.3 fn:adjust-time-to-timezone 54
2.1.6.6 QNames Functions 54
2.1.6.6.1 [XQuery1.0XPath2.0/2] Section 11.1.1 fn:resolve-QName 54
2.1.6.6.2 [XQuery1.0XPath2.0/2] Section 11.1.2 fn:QName 54
2.1.6.6.3 [XQuery1.0XPath2.0/2] Section 11.2.2 fn:prefix-from-QName 54
2.1.6.6.4 [XQuery1.0XPath2.0/2] Section 11.2.3 fn:local-name-from-QName 54
2.1.6.6.5 [XQuery1.0XPath2.0/2] Section 11.2.5 fn:namespace-uri-for-prefix 54
2.1.6.6.6 [XQuery1.0XPath2.0/2] Section 11.2.6 fn:in-scope-prefixes 54
2.1.6.7 base64Binary and hexBinary Functions 54
2.1.6.8 Notation Functions 54

2.1.6.9 Nodes Functions 54
2.1.6.9.1 [XQuery1.0XPath2.0/2] Section 14.1 fn:name 54
2.1.6.9.2 [XQuery1.0XPath2.0/2] Section 14.3 fn:namespace-uri 54
2.1.6.9.3 [XQuery1.0XPath2.0/2] Section 14.4 fn:number 55
2.1.6.9.4 [XQuery1.0XPath2.0/2] Section 14.5 fn:lang 55
2.1.6.9.5 [XQuery1.0XPath2.0/2] Section 14.9 fn:root 55
2.1.6.10 Sequences Functions 55
2.1.6.10.1 [XQuery1.0XPath2.0/2] Section 15.1.1 fn:boolean 55
2.1.6.10.2 [XQuery1.0XPath2.0/2] Section 15.1.3 fn:index-of 55
2.1.6.10.3 [XQuery1.0XPath2.0/2] Section 15.1.5 fn:exists 55
2.1.6.10.4 [XQuery1.0XPath2.0/2] Section 15.1.6 fn:distinct-values 55
2.1.6.10.5 [XQuery1.0XPath2.0/2] Section 15.1.7 fn:insert-before 56
2.1.6.10.6 [XQuery1.0XPath2.0/2] Section 15.1.8 fn:remove 56
2.1.6.10.7 [XQuery1.0XPath2.0/2] Section 15.1.9 fn:reverse 56
2.1.6.10.8 [XQuery1.0XPath2.0/2] Section 15.1.10 fn:subsequence 56
2.1.6.10.9 [XQuery1.0XPath2.0/2] Section 15.1.11 fn:unordered 56
2.1.6.10.10 [XQuery1.0XPath2.0/2] Section 15.2.1 fn:zero-or-one 56
2.1.6.10.11 [XQuery1.0XPath2.0/2] Section 15.2.2 fn:one-or-more 56
2.1.6.10.12 [XQuery1.0XPath2.0/2] Section 15.2.3 fn:exactly-one 56
2.1.6.10.13 [XQuery1.0XPath2.0/2] Section 15.3.1 fn:deep-equal 56
2.1.6.10.14 [XQuery1.0XPath2.0/2] Section 15.4.2 fn:avg 56
2.1.6.10.15 [XQuery1.0XPath2.0/2] Section 15.4.3 fn:max 56
2.1.6.10.16 [XQuery1.0XPath2.0/2] Section 15.4.4 fn:min 57
2.1.6.10.17 [XQuery1.0XPath2.0/2] Section 15.4.5 fn:sum 57
2.1.6.10.18 [XQuery1.0XPath2.0/2] Section 15.5.2 fn:id 57
2.1.6.10.19 [XQuery1.0XPath2.0/2] Section 15.5.3 fn:idref 57
2.1.6.10.20 [XQuery1.0XPath2.0/2] Section 15.5.4 fn:doc 57
2.1.6.10.21 [XQuery1.0XPath2.0/2] Section 15.5.5 fn:doc-available 57
2.1.6.10.22 [XQuery1.0XPath2.0/2] Section 15.5.6 fn:collection 57
2.1.6.10.23 [XQuery1.0XPath2.0/2] Section 15.5.7 fn:element-with-id 57
2.1.6.11 Context 58
2.1.6.11.1 [XQuery1.0XPath2.0/2] Section 16.1 fn:position 58
2.1.6.11.2 [XQuery1.0XPath2.0/2] Section 16.2 fn:last 58
2.1.6.11.3 [XQuery1.0XPath2.0/2] Section 16.3 fn:current-dateTime 58
2.1.6.11.4 [XQuery1.0XPath2.0/2] Section 16.4 fn:current-date 58
2.1.6.11.5 [XQuery1.0XPath2.0/2] Section 16.5 fn:current-time 58
2.1.6.11.6 [XQuery1.0XPath2.0/2] Section 16.6 fn:implicit-timezone 58
2.1.6.11.7 [XQuery1.0XPath2.0/2] Section 16.7 fn:default-collation 58
2.1.6.11.8 [XQuery1.0XPath2.0/2] Section 16.8 fn:static-base-uri 58
2.2 Clarifications 58
2.2.1 [XDM/2] Section 3.3.1.3, Relationship Between Typed-Value and String-Value 58
2.2.2 [XDM/2] Section 3.3.1.4, Pattern Facets 58
2.2.3 [XDM/2] Section 6.2.4, Element Nodes, Construction from a PSVI 59
2.2.4 [XQuery1.0/2] Section 5.2, Optional Features 59
2.2.4.1 [XQuery1.0/2] Section 5.2.1, Schema Import Feature 59
2.2.4.2 [XQuery1.0/2] Section 5.2.2, Schema Validation Feature 59
2.2.4.3 [XQuery1.0/2] Section 5.2.3, Static Typing Feature 59
2.2.4.4 [XQuery1.0/2] Section 5.2.4, Full Axis Feature 59
2.2.4.5 [XQuery1.0/2] Section 5.2.5, Module Feature 59
2.2.4.6 [XQuery1.0/2] Section 5.2.6, Serialization Feature 59
2.2.5 sql:column Function 60
2.2.6 sql:variable Function 60
2.3 Error Handling 60

2.4 Security 60
3 Change Tracking 61
4 Index 62

1 Introduction

This document specifies the level of support provided by Microsoft SQL Server 2008 R2 for the XQuery query language.

XQuery is a query language that allows queries against the content of an XML document or against XML that is stored within a relational database system. The schema for the XML may be known or unknown. The XQuery language also allows the construction of XML output.

1.1 Glossary

The following terms are defined in [MS-GLOS]:

XML

The following terms are defined in [XDM/2]:

document
fragment

The following terms are specific to this document:

schema collection: A collection of XML schema [XMLSCHEMA1/2] components, such as attribute, element, and type declarations, that belong to one or several target namespaces. A schema collection can be used to constrain an XML data type in SQL Server and provides type information for the nodes and values in an XML data type instance.

MAY, SHOULD, MUST, SHOULD NOT, MUST NOT: These terms (in all caps) are used as described in [RFC2119]. All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

1.2 Microsoft Implementations

The standard described by this document is implemented in the product Microsoft SQL Server 2008 R2.

1.3 References

1.3.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information. Please check the archive site, http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624, as an additional source.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, http://www.ietf.org/rfc/rfc2119.txt

[XQuery1.0/2] Boag, S., Chamberlin, D., Fernández, M., Eds., et al., "XQuery 1.0: An XML Query Language (Second Edition)", W3C Recommendation, December 2010, http://www.w3.org/TR/2010/REC-xquery-20101214/

[XQuery1.0XPath2.0/2] Malhotra, A., Melton, J., Walsh, N. Eds., et al., "XQuery 1.0 and XPath 2.0 Functions and Operators (Second Edition)", W3C Recommendation, December 2010, http://www.w3.org/TR/2010/REC-xpath-functions-20101214/

[XDM/2] Berglund, A. Fernández, M. Malhotra, A., Eds., et al., "XQuery 1.0 and XPath 2.0 Data Model (XDM) (Second Edition)", W3C Recommendation, December 2010, http://www.w3.org/TR/2010/REC-xpath-datamodel-20101214/

1.3.2 Informative References

[MSDN-PrimExpXQuery] Microsoft Corporation, "Primary Expressions (XQuery)", http://msdn.microsoft.com/en-us/library/ms190754.aspx

[MSDN-SeqTypExpXQuery] Microsoft Corporation, "SequenceType Expressions (XQuery)", http://msdn.microsoft.com/en-us/library/ms190179.aspx

[MSDN-XQueryLangRef] Microsoft Corporation, "XQuery Language Reference (Database Engine)", http://msdn.microsoft.com/en-us/library/ms189075.aspx

[XMLINFOSET] World Wide Web Consortium, "XML Information Set (Second Edition)", February 2004, http://www.w3.org/TR/2004/REC-xml-infoset-20040204

[XMLSCHEMA1/2] Thompson, H.S., Ed., Beech, D., Ed., Maloney, M., Ed., and Mendelsohn, N., Ed., "XML Schema Part 1: Structures Second Edition", W3C Recommendation, October 2004, http://www.w3.org/TR/xmlschema-1/

1.4 Standards Support Requirements

[XQuery1.0/2] section 5.1 defines minimal conformance for an XQuery processor. This conformance is defined as support for all features in the document except the following:

- Features that are called out as optional.
- Support for all functions that are listed in [XQuery1.0XPath2.0/2].
- Support for the data model [XDM/2] as defined in [XQuery1.0/2] section 5.3.

Variations from optional features are described in section 2.1 of this document. The implemention of optional features is described in section 2.2.

1.5 Notation

The following notations are used to identify clarifications in section 2.2:

Notation	Explanation
C####	This notation identifies a clarification of ambiguity in the target specification. This includes imprecise statements, omitted information, discrepancies, and errata. This does not include data formatting clarifications.
V####	This notation identifies an intended point of variability in the target specification, such as the use of MAY, SHOULD, or RECOMMENDED. This does not include extensibility points.
E####	Because the use of extensibility points (such as optional implementation-specific data) could impair interoperability, this notation identifies such points in the target specification.

2 Standards Support Statements

2.1 Normative Variations

The following subsections detail the normative variations from [XQuery1.0/2], [XQuery1.0XPath2.0/2], and [XDM/2] in XQuery in SQL Server. For complete information on the XQuery in SQL Server language, see [MSDN-XQueryLangRef].

2.1.1 Data Model Items

Section numbers that are included in the titles within the Data Model Items section refer to sections in [XDM/2].

2.1.1.1 [XDM/2] Section 1 Introduction

XQuery in SQL Server supports XML schema types with a variation. The variations are described for each type in section 2.13 in this document. XQuery in SQL Server supports representations of documents and of complex values. XQuery in SQL Server supports typed atomic values. XQuery in SQL Server supports ordered, heterogeneous sequences.

2.1.1.2 Concepts

2.1.1.2.1 [XDM/2] Section 2.4 Document Order

XQuery in SQL Server supports document order with a variation. XQuery in SQL Server does not support namespace nodes.

2.1.1.3 Data Model Construction

2.1.1.3.1 [XDM/2] Section 3 Data Model Construction

XQuery in SQL Server supports model instances that are based on document fragments and sequences of document nodes.

2.1.1.3.2 [XDM/2] Section 3.1 Direct Construction

2.1.1.3.3 [XDM/2] Section 3.2 Construction from an Infoset

XQuery in SQL Server supports the construction of a data model instance from an Infoset [XMLINFOSET].

2.1.1.3.4 [XDM/2] Section 3.3 Construction from a PSVI

XQuery in SQL Server supports the construction of a data model instance from a Post Schema Validation Infoset (PSVI) with a variation. XQuery in SQL Server supports the construction of a data model instance from a PSVI only with element or attribute information that is valid, or with information for which the validity is unknown. XQuery in SQL Server does not support construction of a data model from an instance with element or attribute information that is known to be invalid.

2.1.1.3.5 [XDM/2] Section 3.3.1 Mapping PSVI Additions to Node Properties

XQuery in SQL Server supports construction from a PSVI with a variation. XQuery in SQL Server supports valid nodes, and nodes whose validity is unknown. XQuery in SQL Server does not support data model construction from an instance with nodes that are known to be invalid.

2.1.1.3.6 [XDM/2] Section 3.3.1.1 Element and Attribute Node Type Names

XQuery in SQL Server supports element and attribute node type names with a variation. XQuery in SQL Server supports valid element and attribute node type names as well as element and attribute node type names whose validity is unknown. XQuery in SQL Server does not support data model construction from an instance with element and attribute node type names that are known to be invalid.

2.1.1.3.7 [XDM/2] Section 3.3.1.2 Typed Value Determination

XQuery in SQL Server supports typed value determination with a variation. XQuery in SQL Server does not support the **nilled** property.

2.1.1.3.8 [XDM/2] Section 3.3.1.3 Relationship Between Typed Value and String Value

See [XDM/2] section 3.3.1.3 for a description of items that a processor may optionally support.

2.1.1.3.9 [XDM/2] Section 3.3.1.4 Pattern Facets

See [XDM/2] section 3.3.1.4 for a description of items that a processor may optionally support.

2.1.1.3.10 [XDM/2] Section 3.3.3 QNames and NOTATIONS

XQuery in SQL Server supports QNames and NOTATIONS with a variation. XQuery in SQL Server does not support NOTATIONS.

2.1.1.4 Accessors

2.1.1.4.1 [XDM/2] Section5 Accessors

This section of [XDM/2] describes theoretical accessors for items that exist in the data model. XQuery in SQL Server supports these accessors with a variation. XQuery in SQL Server supports the accessors for the items that are supported by XQuery in SQL Server. It does not support accessors for items that XQuery in SQL Server does not support.

Item support is described in the appropriate section for the item or type. See section 2.1.2 ["Basics"], section 2.1.3 ["Data Types"], section 2.1.4 ["Expressions'], section 2.1.5 ["Operators"], and section 2.1.6. ["Functions"]. Accessors described in this section that do not appear in the [XDM/2] are not supported by XQuery in SQL Server.

2.1.1.5 Nodes

2.1.1.5.1 [XDM/2] Section 6.2.4 Construction from a PSVI

See section 6.2.4 of [XDM/2] for a description of items that a processor may optionally support.

2.1.1.5.2 [XDM/2] Section 6.4 Namespace Nodes

XQuery in SQL Server does not support namespace nodes.

2.1.1.6 Conformance

2.1.1.6.1 [XDM/2] Section 7 Conformance

Section 7 Conformance of [XDM/2] states that conformance is defined by specifications that depend on the data model specification and not by the data model specification itself. The extent to which XQuery in SQL Server conforms to the requirements that are stated in [XQuery1.0/2] and [XQuery1.0XPath2.0/2] is defined in the remainder of this document.

2.1.2 Basics

All section numbers that are included in the section names in this section correspond to the section numbers in [XQuery1.0/2]. All numbered expressions in this section correspond to numbered expressions in [XQuery1.0/2].

2.1.2.1 Expression Context

2.1.2.1.1 [XQuery1.0/2] Section 2.1.1 Static Context

Section 2.1.1 of [XQuery1.0/2] contains definitions of the individual components of the static context, but this section does not specify the normative requirements. The normative requirements are defined in appendix C.1 of [XQuery1.0/2].

2.1.2.1.1.1 [XQuery1.0/2] Appendix C.1 Static Context Components

2.1.2.1.1.1.1 XPath 1.0 Compatibility Mode

XQuery in SQL Server does not support XPath 1.0 compatibility mode.

2.1.2.1.1.1.2 Statically Known Namespaces

XQuery in SQL Server supports the statically known namespaces with a variation. By default, the local namespace is not known by XQuery in SQL Server.

2.1.2.1.1.1.3 Default Function Namespace

XQuery in SQL Server supports the default function namespace fn. XQuery in SQL Server does not support overwrite of the default function namespace in the prologue.

2.1.2.1.1.1.4 In-Scope Schema Types

The types in the xs namespace that are supported by XQuery in SQL Server are described in section 2.1.3 ["Data Types"]. XQuery in SQL Server does not support schema import in the prologue. XQuery in SQL Server does implicitly import the schema components that are contained in the XML schema collection that constrains the context-providing XML data type, if such a collection is specified.

2.1.2.1.1.1.5 In-Scope Element Declarations

XQuery in SQL Server does not support in-scope element declarations.

2.1.2.1.1.1.6 In-Scope Attribute Declarations

XQuery in SQL Server does not support in-scope attribute declarations.

2.1.2.1.1.1.7 In-Scope Variables

XQuery in SQL Server supports in-scope variables with a variation. XQuery in SQL Server does not allow specification of in-scope variables in the prologue.

2.1.2.1.1.1.8 Function Signatures

XQuery in SQL Server supports the default fn prefix for functions. For specific function support, see section 2.1.6 ["Functions"]. XQuery in SQL Server does not support augmentation of functions through schema import, module import, or through function declaration in the prologue. Constructor functions for user-defined atomic types and the **sqltypes** namespace are implicitly imported for imported XML schema collections.

2.1.2.1.1.1.9 Statically Known Collations

XQuery in SQL Server has not augmented the statically known collations with any additional collations.

2.1.2.1.1.1.10 Default Collation

XQuery in SQL Server uses the Unicode codepoint collation with a variation, as the default collation. Surrogate pairs are sorted as two characters, not one, during a collation operation.

2.1.2.1.1.1.11 Construction Mode

XQuery in SQL Server does not support the specified default construction mode. XQuery in SQL Server only supports construction mode value "strip". XQuery in SQL Server does not support setting the construction mode component in the prologue.

2.1.2.1.1.1.12 Ordering Mode

XQuery in SQL Server does not support the ordering mode value "unordered". XQuery in SQL Server supports only the ordering mode value "ordered". XQuery in SQL Server does not support specifying the ordering mode in the prologue.

2.1.2.1.1.1.13 Default Order for Empty Sequences

The implementation in XQuery in SQL Server defines the default value for the order for empty sequences as "least". XQuery in SQL Server does not support the value "greatest" for the default order for empty sequences.

2.1.2.1.1.1.14 Boundary-Space Policy

XQuery in SQL Server supports "strip" as the default value for boundary-space policy. XQuery in SQL Server does not support any other values.

2.1.2.1.1.1.15 Copy-Namespaces Mode

Copy-namespaces mode is not supported by XQuery in SQL Server.

2.1.2.1.1.1.16 Base URI

XQuery in SQL Server does not support the base URI.

2.1.2.1.1.1.17 Statically Known Documents

XQuery in SQL Server does not support statically known documents.

2.1.2.1.1.1.18 Statically Known Collections

XQuery in SQL Server does not support statically known collections.

2.1.2.1.1.1.19 Statically Known Default Collection Type

XQuery in SQL Server does not support the statically known default collection type.

2.1.2.1.2 [XQuery1.0/2] Section 2.1.2 Dynamic Context

2.1.2.1.2.1 [XQuery1.0/2] Appendix C.2 Dynamic Context Components

2.1.2.1.2.1.1 Variable Values

XQuery in SQL Server supports dynamic variable values. XQuery in SQL Server does not support variable declaration in the prologue.

2.1.2.1.2.1.2 Function Implementations

The functions supported by XQuery in SQL Server are documented in section 2.1.6 ["Functions"]. XQuery in SQL Server does not support function implementations through schema import, module import, or through function declaration in the prologue.

2.1.2.1.2.1.3 Current dateTime

Current dateTime is not supported by XQuery in SQL Server.

2.1.2.1.2.1.4 Implicit Timezone

The implicit timezone is not supported by XQuery in SQL Server.

2.1.2.1.2.1.5 Available Documents

Available documents is not supported by XQuery in SQL Server.

2.1.2.1.2.1.6 Available Collections

Available collections is not supported by XQuery in SQL Server.

2.1.2.1.2.1.7 Default Collection

Default collection is not supported by XQuery in SQL Server.

2.1.2.2 Processing Model

2.1.2.2.1 [XQuery1.0/2] Section 2.2.1 Data Model Generation

[XQuery1.0/2] section 2.2.1 specifies no normative requirements.

2.1.2.2.2 [XQuery1.0/2] Section 2.2.2 Schema Import Processing

XQuery in SQL Server supports schema import with a variation. XQuery in SQL Server does not support explicit schema import. XQuery in SQL Server implicitly imports the schemas that are included in the XML schema collection that is associated with the context-setting XML data type.

2.1.2.2.3 [XQuery1.0/2] Section 2.2.3.1 Expression Processing - Static Analysis Phase

XQuery in SQL Server supports the static analysis phase of expression processing. XQuery in SQL Server supports the static typing feature and raises the type error XPTY0004 if static analysis encounters a type error.

2.1.2.2.4 [XQuery1.0/2] Section 2.2.3.2 Expression Processing - Dynamic Evaluation Phase

XQuery in SQL Server supports a dynamic evaluation phase with a variation. XQuery in SQL Server does not raise dynamic errors. In cases where a dynamic error is encountered, XQuery in SQL Server returns an empty sequence.

2.1.2.2.5 [XQuery1.0/2] Section 2.2.4 Serialization

Data model serialization is an optional feature in [XQuery1.0/2]. XQuery in SQL Server does not support a data model serialization interface. However, XQuery in SQL Server supports serialization of an instance of an XML data type through conversion or casting to SQL string or binary types.

2.1.2.2.6 [XQuery1.0/2] Section 2.2.5 Consistency Constraints

XQuery in SQL Server supports applicable consistency constraints that relate to features supported by XQuery in SQL Server. Many of the constraints that are enumerated in section 2.2.5 of [XQuery1.0/2] are for features that are documented elsewhere in this document as not supported by XQuery in SQL Server and hence such consistency constraints are not supported.

2.1.2.3 Error Handling

2.1.2.3.1 [XQuery1.0/2] Section 2.3.1 Kinds of Errors

Raising static errors and type errors during the static analysis phase is supported by XQuery in SQL Server. Dynamic errors are not supported by XQuery in SQL Server. Dynamic errors are indicated by XQuery in SQL Server by the return of an empty sequence.

2.1.2.3.2 [XQuery1.0/2] Section 2.3.2 Identifying and Reporting Errors

[XQuery1.0/2] states that implementations can represent errors by a URI reference that is derived from the QName for the error. XQuery in SQL Server does not represent errors by a URI reference that is derived from the QName for the error.

2.1.2.3.3 [XQuery1.0/2] Section 2.3.3 Handling Dynamic Errors

XQuery in SQL Server does not report dynamic errors. When the result of the dynamic evaluation phase is an error, XQuery in SQL Server returns an empty sequence.

2.1.2.3.4 [XQuery1.0/2] Section 2.3.4 Errors and Optimization

XQuery in SQL Server performs the required cardinality check and returns an error if the cardinality check fails.

2.1.2.4 Concepts

2.1.2.4.1 [XQuery1.0/2] Section 2.4.1 Document Order

This section contains no normative requirements.

2.1.2.4.2 [XQuery1.0/2] Section 2.4.2 Atomization

This section contains no normative requirements.

2.1.2.4.3 [XQuery1.0/2] Section 2.4.3 Effective Boolean Value

XQuery in SQL Server does not support the **fn:boolean** function. XQuery in SQL Server supports the semantics of effective Boolean value in predicates and Boolean expressions with a variation. XQuery in SQL Server returns an error if the operand is a singleton value of type **xs:string**, **xs:anyURI**, or **xs:untypedAtomic**, or a type derived that is from these types.

2.1.2.4.4 [XQuery1.0/2] Section 2.4.4 Input Sources

The **fn:doc** function is not supported by XQuery in SQL Server. The **fn:collection** function is not supported by XQuery in SQL Server. The implicit context for the query is the XML data type instance of the query.

2.1.2.4.5 [XQuery1.0/2] Section 2.4.5 URI Literals

```
[140]    URILiteral    ::=    StringLiteral
```

[140] is supported by XQuery in SQL Server. The whitespace normalization rules for URILiteral are supported by XQuery in SQL Server.

2.1.2.5 Types

2.1.2.5.1 [XQuery1.0/2] Section 2.5.1 Predefined Schema Types

This document covers the types that are supported by XQuery in SQL Server in section 2.1.3 ["Data Types"].

2.1.2.5.2 [XQuery1.0/2] Section 2.5.3 SequenceType Syntax

```
[119]    SequenceType    ::=    ("empty-sequence" "(" ")")
| (ItemType OccurrenceIndicator?)
```

[119] is supported by XQuery in SQL Server.

```
[120]    OccurrenceIndicator    ::=    "?" | "*" | "+"
```

[120] is supported by XQuery in SQL Server with variations. The "?" syntax is supported by XQuery in SQL Server. The "+" and "*" syntaxes are not supported by XQuery in SQL Server.

```
[121]    ItemType    ::=    KindTest | ("item" "(" ")") | AtomicType
```

[121] is supported by XQuery in SQL Server with a variation. See the information on expression [123] later in this section for limitations in KindTest.

```
[122]    AtomicType    ::=    QName
```

[122] is supported by XQuery in SQL Server.

```
[123]    KindTest    ::=    DocumentTest
| ElementTest
| AttributeTest
| SchemaElementTest
| SchemaAttributeTest
| PITest
| CommentTest
| TextTest
| AnyKindTest
```

[123] is supported by XQuery in SQL Server with a variation. The following tests are not supported by XQuery in SQL Server: DocumentTest, SchemaElementTest, SchemaAttributeTest. The following tests are supported only for type checks, but not in path expressions: ElementTest, AttributeTest.

```
[124]    AnyKindTest    ::=    "node" "(" ")"
```

[124] is supported by XQuery in SQL Server with a variation. AnyKindTest cannot be used to access the text node of a simple typed type or of content of type **xs:anyType**.

```
[125]    DocumentTest    ::=    "document-node" "(" (ElementTest | SchemaElementTest)? ")"
```

[125] is not supported by XQuery in SQL Server.

```
[126]    TextTest    ::=    "text" "(" ")"
```

[126] is supported by XQuery in SQL Server with a variation. TextTest cannot be used to access the text node of a simple typed type or of content of type **xs:anyType**.

```
[127]    CommentTest    ::=    "comment" "(" ")"
```

[127] is supported by XQuery in SQL Server.

```
[128]    PITest    ::=    "processing-instruction" "(" (NCName | StringLiteral)? ")"
```

[128] is supported by XQuery in SQL Server.

```
[129]    AttributeTest    ::=    "attribute" "(" (AttribNameOrWildcard ("," TypeName)?)? ")"
```

[129] is supported by XQuery in SQL Server with a variation. [129] is supported for type tests, but is not supported for path expressions.

```
[130]    AttribNameOrWildcard    ::=    AttributeName | "*"
```

[130] is supported by XQuery in SQL Server.

```
[131]    SchemaAttributeTest    ::=    "schema-attribute" "(" AttributeDeclaration ")"
```

[131] is not supported by XQuery in SQL Server.

```
[132]    AttributeDeclaration    ::=    AttributeName
```

[132] is supported by XQuery in SQL Server.

```
[133]    ElementTest    ::=    "element" "(" (ElementNameOrWildcard ("," TypeName "?"?)?)?
")"
```

[133] is supported by XQuery in SQL Server with a variation. [133] is supported in type expressions but is not supported in path expressions.

```
[134]    ElementNameOrWildcard    ::=    ElementName | "*"
```

[134] is supported by XQuery in SQL Server.

```
[135]    SchemaElementTest    ::=    "schema-element" "(" ElementDeclaration ")"
```

[135] is not supported by XQuery in SQL Server.

```
[136]    ElementDeclaration    ::=    ElementName
```

[136] is supported by XQuery in SQL Server with a variation. [136] is supported in type expressions but is not supported in path expressions.

```
[137]    AttributeName    ::=    QName
```

[137] is supported by XQuery in SQL Server.

```
[138]    ElementName    ::=    QName
```

[138] is supported by XQuery in SQL Server.

```
[139]    TypeName    ::=    QName
```

[139] is supported by XQuery in SQL Server. However, only the types documented in section 2.1.3 ["Data Types"] are supported.

2.1.2.5.3 [XQuery1.0/2] Section 2.5.4 SequenceType Matching

2.1.2.5.4 [XQuery1.0/2] Section 2.5.4.1 Matching a Sequence Type and a Value

Matching a sequence type and a value is supported by XQuery in SQL Server with a variation. XQuery in SQL Server supports only the "?" operator, not the "*" or the "+" operator.

2.1.2.5.5 [XQuery1.0/2] Section 2.5.4.2 Matching an Item Type and an Item

XQuery in SQL Server supports Matching an Item Type and an item with a variation. XQuery in SQL Server does not support the document(E) syntax.

2.1.2.5.6 [XQuery1.0/2] Section 2.5.4.3 Element Test

XQuery in SQL Server supports the element test with a variation. XQuery in SQL Server requires that the TypeName token, when specified, is followed by the "?" syntax. XQuery in SQL Server does not support the "*" syntax for specifying TypeName.

2.1.2.5.7 [XQuery1.0/2] Section 2.5.4.4 Schema Element Test

The schema element test is not supported by XQuery in SQL Server.

2.1.2.5.8 [XQuery1.0/2] Section 2.5.4.5 Attribute Test

XQuery in SQL Server supports the attribute test with a variation. XQuery in SQL Server requires that the **TypeName** attribute, when specified, is followed by "?". XQuery in SQL Server does not support "*" for the TypeName specification.

2.1.2.5.9 [XQuery1.0/2] Section 2.5.4.6 Schema Attribute Test

The schema attribute test is not supported by XQuery in SQL Server.

2.1.3 Data Types

The data types that are assessed in this document are those that appear in the figure titled "XPath 2.0 and XQUERY 1.0 Type Hierarchy" in section 2.6.3 of [XDM/2].

2.1.3.1 XSD Types

2.1.3.1.1 xs:ID

The **xs:ID** type is supported by XQuery in SQL Server with a variation. This type is supported only for attributes, not for elements.

2.1.3.1.2 xs:IDREF

The **xs:IDREF** type is supported by XQuery in SQL Server with a variation. This type is supported only for attributes, not for elements.

2.1.3.1.3 xs:untypedAtomic

The **xs:untypedAtomic** type is supported by XQuery in SQL Server with a variation. The type must be referenced as **xdt:untypedAtomic**.

2.1.3.1.4 xs:untyped

The **xs:untyped** type is supported by XQuery in SQL Server with a variation. The type must be referenced as **xdt:untyped**.

2.1.3.1.5 xs:NOTATION

The **xs:NOTATION** type is not supported by XQuery in SQL Server.

2.1.3.1.6 xs:yearMonthDuration

The **xs:yearMonthDuration** type is not supported by XQuery in SQL Server.

2.1.3.1.7 xs:dayTimeDuration

The **xs:dayTimeDuration** type is not supported by XQuery in SQL Server.

2.1.3.1.8 xs:IDREFS

The **xs:IDREFS** type is supported by XQuery in SQL Server with a variation. This type is supported only for attributes, not for elements.

2.1.3.2 Casting Primitive Type to Primitive Type

[XQuery1.0XPath2.0/2] section 17.1 has a table that describes which primitive types can be explicitly cast to other primitive types. XQuery in SQL Server supports the documented valid casts, with variations, as described in this section.

- The yearMonthDuration type and the dayTimeDuration type are not supported by XQuery in SQL Server, so no casts are valid.
- XQuery in SQL Server always supports casting between untypedAtomic and anyURI and does not restrict casting between untypedAtomic and anyURI to certain values. [XQuery1.0XPath2.0/2] specifies that casting between untypedAtomic and anyURI is restricted to certain values.
- XQuery in SQL Server always supports casting between string and anyURI, not restricted to certain values, as specified in [XQuery1.0XPath2.0/2].
- XQuery in SQL Server does not support casts from dateTime to time, date, gYearMonth, gyear, gMonthDay, gDay, and gMonth.
- XQuery in SQL Server does not support casts from date to dateTime, gYearMonth, gyear, gMonthDay, gDay, gMonth.
- XQuery in SQL Server does not support casts from QName to untypedAtomic or string.
- XQuery in SQL Server does not support casts from NOTATION to untypeAtomic, string, or NOTATION.

2.1.4 Expressions

Section numbers in this section correspond to section numbers in [XQuery1.0/2]. All numbered expressions in this section correspond to numbered expressions in [XQuery1.0/2].

2.1.4.1 Expression Syntax

2.1.4.1.1 [XQuery1.0/2] Section 3 Expressions

For [31] see section 3.3.1 ["Constructing Sequences"].

```
[32]    ExprSingle    ::=    FLWORExpr
| QuantifiedExpr
| TypeswitchExpr
| IfExpr
| OrExpr
```

[32] is supported by XQuery in SQL Server.

2.1.4.1.2 [XQuery1.0/2] Section 3.1 Primary Expressions

2.1.4.1.3 [XQuery1.0/2] Section 3.1.1 Literals

```
[85]    Literal    ::=    NumericLiteral | StringLiteral
```

[85] is supported by XQuery in SQL Server.

```
[86]    NumericLiteral    ::=    IntegerLiteral | DecimalLiteral | DoubleLiteral
```

[86] is supported by XQuery in SQL Server.

```
[141]    IntegerLiteral    ::=    Digits
```

[141] is supported by XQuery in SQL Server.

```
[142]    DecimalLiteral    ::=    ("." Digits) | (Digits "." [0-9]*)
```

[142] is supported by XQuery in SQL Server.

```
[143]    DoubleLiteral    ::=    (("." Digits) | (Digits ("." [0-9]*)?)) [eE] [+-]? Digits
```

[143] is supported by XQuery in SQL Server.

```
[144]    StringLiteral    ::=    ('"' (PredefinedEntityRef | CharRef | EscapeQuot | [^"&])*
'"') | ("'" (PredefinedEntityRef | CharRef | EscapeApos | [^'&])* "'")
```

[144] is supported by XQuery in SQL Server.

```
[145]    PredefinedEntityRef    ::=    "&" ("lt" | "gt" | "amp" | "quot" | "apos") ";"
```

[145] is supported by XQuery in SQL Server.

```
[158]    Digits    ::=    [0-9]+
```

[158] is supported by XQuery in SQL Server.

2.1.4.1.4 [XQuery1.0/2] Section 3.1.2 Variable References

```
[87]    VarRef    ::=    "$" VarName
```

[87] is supported by XQuery in SQL Server.

```
[88]    VarName    ::=    QName
```

[88] is supported by XQuery in SQL Server with variance. Only local names are supported; fully qualified QNames are not supported. XQuery in SQL Server provides an extension function, the **sql:variable** function, which can be used in to refer to a QName [MSDN-PrimExpXQuery].

[XQuery1.0/2] section 3.1.2 specifies that variables may be declared in the Prolog or an imported module. XQuery in SQL Server does not support module import.

2.1.4.1.5 [XQuery1.0/2] Section 3.1.3 Parenthesized Expressions

```
[89]    ParenthesizedExpr    ::=    "(" Expr? ")"
```

[89] is supported by XQuery in SQL Server.

2.1.4.1.6 [XQuery1.0/2] Section 3.1.4 Context Item Expression

```
[90]    ContextItemExpr    ::=    "."
```

[90] is supported by XQuery in SQL Server.

2.1.4.1.7 [XQuery1.0/2] Section 3.1.5 Function Calls

```
[93]    FunctionCall    ::=    QName "(" (ExprSingle ("," ExprSingle)*)? ")"
```

[93] is supported by XQuery in SQL Server with a variation. XQuery in SQL Server returns an empty sequence for a dynamic error result instead of returning a dynamic error.

2.1.4.1.8 [XQuery1.0/2] Section 3.2 Path Expressions

```
 [68]    PathExpr    ::=    ("/" RelativePathExpr?)
| ("//" RelativePathExpr)
| RelativePathExpr
```

[68] is supported by XQuery in SQL Server with a variation. In the PathExpr token, a PrimaryExpression can only be used as the root of the PathExpr.

```
[69]    RelativePathExpr    ::=    StepExpr (("/" | "//") StepExpr)*
```

[69] is supported by XQuery in SQL Server with a variation.

2.1.4.1.9 [XQuery1.0/2] Section 3.2.1 Steps

```
 [70]    StepExpr    ::=    FilterExpr | AxisStep
```

[70] is supported by XQuery in SQL Server.

```
[71]    AxisStep    ::=    (ReverseStep | ForwardStep) PredicateList
```

[71] is supported by XQuery in SQL Server.

```
[72]    ForwardStep    ::=    (ForwardAxis NodeTest) | AbbrevForwardStep
```

[72] is supported by XQuery in SQL Server.

```
[75]    ReverseStep    ::=    (ReverseAxis NodeTest) | AbbrevReverseStep
```

[75] is supported by XQuery in SQL Server.

```
[82]    PredicateList    ::=    Predicate*
```

[82] is supported by XQuery in SQL Server.

2.1.4.1.10 [XQuery1.0/2] Section 3.2.1.1 Axes

```
[73]    ForwardAxis    ::=    ("child" "::")
| ("descendant" "::")
| ("attribute" "::")
| ("self" "::")
| ("descendant-or-self" "::")
| ("following-sibling" "::")
| ("following" "::")
```

[73] is supported by XQuery in SQL Server with variations. XQuery in SQL Server does not support the following-sibling axis, or the following axis.

```
[76]    ReverseAxis    ::=    ("parent" "::")
| ("ancestor" "::")
| ("preceding-sibling" "::")
| ("preceding" "::")
| ("ancestor-or-self" "::")
```

[76] is supported by XQuery in SQL Server with variations. XQuery in SQL Server supports only the parent axis; it does not support the ancestor, preceding-sibling, preceding, or ancestor-or-self axes.

2.1.4.1.11 [XQuery1.0/2] Section 3.2.1.2 Node Tests

```
[78]    NodeTest    ::=    KindTest | NameTest
```

[78] is supported by XQuery in SQL Server with a variation. Not all of the KindTests are supported. Some KindTests are not allowed in some expressions. The limitations are described in section 2.1.2.5.3 ["SequenceType Syntax"], expression [123].

```
[79]    NameTest    ::=    QName | Wildcard
```

[79] is supported by XQuery in SQL Server.

```
[80]    Wildcard    ::=    "*"
| (NCName ":" "*")
| ("*" ":" NCName)
```

[80] is supported by XQuery in SQL Server.

2.1.4.1.12 [XQuery1.0/2] Section 3.2.2 Predicates

XQuery in SQL Server supports predicates with a variation. XQuery in SQL Server supports only static dispatch for predicate expression evaluation.

2.1.4.1.13 [XQuery1.0/2] Section 3.2.3 Unabbreviated Syntax

Section 3.2.3 of [XQuery1.0/2] specification is not normative, and provides syntax examples. XQuery in SQL Server supports the example syntax except where it is noted in the corresponding section that XQuery in SQL Server does not support the step expression or the axis expression

2.1.4.1.14 [XQuery1.0/2] Section 3.2.4 Abbreviated Syntax

```
[74]    AbbrevForwardStep    ::=    "@"? NodeTest
```

[74] is supported by XQuery in SQL Server.

```
[77]    AbbrevReverseStep    ::=    ".."
```

[77] is supported by XQuery in SQL Server.

2.1.4.1.15 [XQuery1.0/2] Section 3.3.1 Constructing Sequences

```
[31]    Expr    ::=    ExprSingle ("," ExprSingle)*
```

[31] is supported by XQuery in SQL Server with variations. XQuery in SQL Server imposes the restriction that all items in the sequence must be either nodes or atomic values: Nodes and atomic values cannot be mixed.

```
[49]    RangeExpr    ::=    AdditiveExpr ( "to" AdditiveExpr )?
```

[49] is not supported by XQuery in SQL Server.

2.1.4.1.16 [XQuery1.0/2] Section 3.3.2 Filter Expressions

```
[81]    FilterExpr    ::=    PrimaryExpr PredicateList
```

[81] is supported by XQuery in SQL Server.

```
[82]    PredicateList    ::=    Predicate*
```

[82] is supported by XQuery in SQL Server.

2.1.4.1.17 [XQuery1.0/2] Section 3.3.3 Combining Node Sequences

```
[52]    UnionExpr    ::=    IntersectExceptExpr ( ("union" | "|") IntersectExceptExpr )*
```

[52] is not supported by XQuery in SQL Server.

```
[53]    IntersectExceptExpr    ::=    InstanceofExpr ( ("intersect" | "except")
InstanceofExpr )*
```

[53] is not supported by XQuery in SQL Server.

2.1.4.1.18 [XQuery1.0/2] Section 3.4 Arithmetic Expressions

```
[50]    AdditiveExpr    ::=    MultiplicativeExpr ( ("+" | "-") MultiplicativeExpr )*
```

[50] is supported by XQuery in SQL Server with a variation. Operands are constrained to be of numeric type, or of type **xs:untypeAtomic**. Operations on operators of type **xs:integer** are returned as type **xs:decimal**.

```
[51]    MultiplicativeExpr    ::=    UnionExpr ( ("*" | "div" | "idiv" | "mod") UnionExpr )*
```

[51] is supported by XQuery in SQL Server with a variation. Arguments MUST be of numeric type, or of type **xs:untypeAtomic**. Operations on operators of type **xs:integer** are returned as type **xs:decimal**. The idiv operator is not supported. A workaround is to cast a result of the div operator to **xs:integer**.

```
[58]    UnaryExpr    ::=    ("-" | "+")* ValueExpr
```

[58] is supported by XQuery in SQL Server with a variation. Arguments are constrained to be of numeric type, or of type **xs:untypeAtomic**. Operations on operators of type **xs:integer** are returned as type **xs:decimal**.

```
[59]    ValueExpr    ::=    ValidateExpr | PathExpr | ExtensionExpr
```

[59] is supported by XQuery in SQL Server with a variation. ValidateExpr is not supported by XQuery in SQL Server. ExtensionExpr is not supported by XQuery in SQL Server.

2.1.4.1.19 [XQuery1.0/2] Section 3.5 Comparison Expressions

```
[48]    ComparisonExpr    ::=    RangeExpr ( (ValueComp
| GeneralComp
| NodeComp) RangeExpr )?
```

[48] is supported by XQuery in SQL Server.

2.1.4.1.20 [XQuery1.0/2] Section 3.5.1 Value Comparisons

```
[61]    ValueComp    ::=    "eq" | "ne" | "lt" | "le" | "gt" | "ge"
```

[61] is supported by XQuery in SQL Server with variations. XQuery in SQL Server casts operands of type **xs:untypedAtomic** to the type of the other operand, not to type **xs:string**, as specified in [XQuery1.0/2].

2.1.4.1.21 [XQuery1.0/2] Section 3.5.2 General Comparisons

```
[60]    GeneralComp    ::=    "=" | "!=" | "<" | "<=" | ">" | ">="
```

[60] is supported by XQuery in SQL Server with variations. XQuery in SQL Server casts operands of type **xs:untypedAtomic** to the type of the other operand, not to type **xs:string**, as specified in [XQuery1.0/2]. Note that expression [60] is defined in [XQuery1.0/2] section 3.5, but the description of this expression is provided in [XQuery1.0/2] section 3.5.2.

2.1.4.1.22 [XQuery1.0/2] Section 3.5.3 Node Comparisons

```
[62]    NodeComp    ::=    "is" | "<<" | ">>"
```

[62] is supported by XQuery in SQL Server. Note that expression [63] is defined in [XQuery1.0/2] section 3.5, but the description of this expression is provided in [XQuery1.0/2] section 3.5.3.

2.1.4.1.23 [XQuery1.0/2] Section 3.6 Logical Expressions

```
[46]    OrExpr    ::=    AndExpr ( "or" AndExpr )*
```

[46] is supported by XQuery in SQL Server with a variation in error handling.

```
[47]    AndExpr    ::=    ComparisonExpr ( "and" ComparisonExpr )*
```

[47] is supported by XQuery in SQL Server with a variation in error handling.

Applicable to [46] and [47] is that XQuery in SQL Server maps dynamic errors to an empty sequence instead of returning a dynamic error.

2.1.4.1.24 [XQuery1.0/2] Section 3.7 Constructors

```
[94]    Constructor    ::=    DirectConstructor
| ComputedConstructor
```

[94] is supported by XQuery in SQL Server.

```
[95]    DirectConstructor    ::=    DirElemConstructor
| DirCommentConstructor
| DirPIConstructor
```

[95] is supported by XQuery in SQL Server.

```
[96]    DirElemConstructor    ::=    "<" QName DirAttributeList ("/>" | (">" DirElemContent*
"</" QName S? ">"))
```

[96] is supported by XQuery in SQL Server.

```
[101]    DirElemContent    ::=    DirectConstructor
| CDataSection
| CommonContent
| ElementContentChar
```

[101] is supported by XQuery in SQL Server.

```
[148]    ElementContentChar    ::=    Char - [{}<&]
```

[148] is supported by XQuery in SQL Server.

```
[102]    CommonContent    ::=    PredefinedEntityRef | CharRef | "{{" | "}}" | EnclosedExpr
```

[102] is supported by XQuery in SQL Server.

```
[107]    CDataSection    ::=    "<![CDATA[" CDataSectionContents "]]>"
```

[107] is supported by XQuery in SQL Server.

```
[108]    CDataSectionContents    ::=    (Char* - (Char* ']]>' Char*))
```

[108] is supported by XQuery in SQL Server.

```
[97]    DirAttributeList    ::=    (S (QName S? "=" S? DirAttributeValue)?)*
```

[97] is supported by XQuery in SQL Server.

```
[98]    DirAttributeValue    ::=    ('"' (EscapeQuot | QuotAttrValueContent)* '"')
| ("'" (EscapeApos | AposAttrValueContent)* "'")
```

[98] is supported by XQuery in SQL Server.

```
[99]    QuotAttrValueContent    ::=    QuotAttrContentChar
| CommonContent
```

[99] is supported by XQuery in SQL Server.

```
[100]    AposAttrValueContent    ::=    AposAttrContentChar
| CommonContent
```

[100] is supported by XQuery in SQL Server.

```
[149]    QuotAttrContentChar    ::=    Char - ["{}<&]
```

[149] is supported by XQuery in SQL Server.

```
[150]    AposAttrContentChar    ::=    Char - ['{}<&]
```

[150] is supported by XQuery in SQL Server.

```
[146]    EscapeQuot    ::=    '""'
```

[146] is supported by XQuery in SQL Server.

```
[147]    EscapeApos    ::=    "''"
```

[147] is supported by XQuery in SQL Server.

```
[29]    EnclosedExpr    ::=    "{" Expr "}"
```

[29] is supported by XQuery in SQL Server.

2.1.4.1.25 [XQuery1.0/2] Section 3.7.1 Direct Element Constructors

XQuery in SQL Server supports direct element constructors with a variation. XQuery in SQL Server does not support the base-URI mechanism for specifying a reference as a relative URI.

2.1.4.1.26 [XQuery1.0/2] Section 3.7.1.1 Attributes

XQuery in SQL Server supports direct attribute constructors with a variation. XQuery in SQL Server does not support multiple or mixed embedded attribute expressions. The expression MUST be either a single string or a single expression. XQuery in SQL Server requires homogeneous sequences. Castability is checked statically and atomization is not applied before the check.

2.1.4.1.27 [XQuery1.0/2] Section 3.7.1.2 Namespace Declaration Attributes

Namespace declaration attributes are supported by XQuery in SQL Server with a variation. XQuery in SQL Server does not check the validity of any provided string against the type **xs:anyURI**.

2.1.4.1.28 [XQuery1.0/2] Section 3.7.1.3 Content

XQuery in SQL Server supports constructors for content with a variation. The specification [XQuery1.0/2] specifies that if nodes within expressions cannot be cast to type **xs:string**, an error is generated. XQuery in SQL Server evaluates types statically instead of attempting to cast to string and generates a static error if the static types do not enable string output.

XQuery in SQL Server does not support the **base-uri** property for nodes created by expressions. XQuery in SQL Server supports only construction mode value "strip", and so statements in [XQuery1.0/2] that describe any other construction modes are not applicable.

XQuery in SQL Server does not support the concept of in-scope namespaces. A recommendation to work around this is to always make all namespaces explicit. [XQuery1.0/2] specifies that if an attribute node follows other nodes, a type error is raised. XQuery in SQL Server raises a dynamic error if an attribute node follows other nodes.

2.1.4.1.29 [XQuery1.0/2] Section 3.7.1.4 Boundary Whitespace

XQuery in SQL Server supports only one boundary whitespace policy, "strip".

2.1.4.1.30 [XQuery1.0/2] Section 3.7.2 Other Direct Constructors

```
[105]    DirPIConstructor    ::=    "<?" PITarget (S DirPIContents)? "?>"
```

[105] is supported by XQuery in SQL Server.

```
[106]    DirPIContents    ::=    (Char* - (Char* '?>' Char*))
```

[106] is supported by XQuery in SQL Server.

```
[103]    DirCommentConstructor    ::=    "<!--" DirCommentContents "-->"
```

[103] is supported by XQuery in SQL Server.

```
[104]    DirCommentContents    ::=    ((Char - '-') | ('-' (Char - '-')))*
```

[104] is supported by XQuery in SQL Server.

2.1.4.1.31 [XQuery1.0/2] Section 3.7.3 Computed Constructors

```
[109]    ComputedConstructor    ::=    CompDocConstructor
| CompElemConstructor
| CompAttrConstructor
| CompTextConstructor
| CompCommentConstructor
| CompPIConstructor
```

[109] is supported by XQuery in SQL Server with a variation. XQuery in SQL Server does not support CompDocConstructor, CompCommentConstructor, or CompPIConstructor. XQuery in SQL Server does not support expressions in the definition of the name for computed element or computed attribute constructors.

2.1.4.1.32 [XQuery1.0/2] Section 3.7.3.1 Computed Element Constructors

```
[111]    CompElemConstructor    ::=    "element" (QName | ("{" Expr "}")) "{" ContentExpr?
"}"
```

[111] is supported by XQuery in SQL Server with a variation. XQuery in SQL Server supports only QName and does not support Expr for the name of the computed element.

```
[112]    ContentExpr    ::=    Expr
```

[112] is supported by XQuery in SQL Server.

2.1.4.1.33 [XQuery1.0/2] Section 3.7.3.2 Computed Attribute Constructors

```
[113]    CompAttrConstructor    ::=    "attribute" (QName | ("{" Expr "}")) "{" Expr? "}"
```

[113] is supported by XQuery in SQL Server with a variation. XQuery in SQL Server does not support expressions for the name of the computed attribute.

2.1.4.1.34 [XQuery1.0/2] Section 3.7.3.3 Document Node Constructors

```
[110]   CompDocConstructor   ::=   "document" "{" Expr "}"
```

[110] is not supported by XQuery in SQL Server. However, all query method results are implicitly wrapped in a document node.

2.1.4.1.35 [XQuery1.0/2] Section 3.7.3.4 Text Node Constructors

```
[114]    CompTextConstructor    ::=    "text" "{" Expr "}"
```

[114] is supported by XQuery in SQL Server with a variation. XQuery in SQL Server supports only ExprSingle and not Expr.

2.1.4.1.36 [XQuery1.0/2] Section 3.7.3.5 Computed Processing Instruction Constructors

```
[116]    CompPIConstructor    ::=    "processing-instruction" (NCName | ("{" Expr "}")) "{"
Expr? "}"
```

[116] is not supported by XQuery in SQL Server. The direct processing instruction (PI) constructor can be used as an alternative.

2.1.4.1.37 [XQuery1.0/2] Section 3.7.3.6 Computed Comment Constructors

```
[115]    CompCommentConstructor    ::=    "comment" "{" Expr "}"
```

[115] is not supported by XQuery in SQL Server. The direct comment constructor can be used as an alternative.

2.1.4.1.38 [XQuery1.0/2] Section 3.7.4 In-scope Namespaces of a Constructed Element

XQuery in SQL Server supports the In-scope Namespaces of a Constructed Element rules with a variation. XQuery in SQL Server does not carry over namespaces that have been declared but which are not actually used in the constructor.

2.1.4.1.39 [XQuery1.0/2] Section 3.8 FLWOR Expressions

```
[33]    FLWORExpr    ::=    (ForClause | LetClause)+ WhereClause? OrderByClause? "return"
ExprSingle
```

[33] is supported by XQuery in SQL Server.

```
[34]    ForClause    ::=    "for" "$" VarName TypeDeclaration? PositionalVar? "in" ExprSingle
("," "$" VarName TypeDeclaration? PositionalVar? "in" ExprSingle)*
```

[34] is supported by XQuery in SQL Server with a variation. TypeDeclaration is not supported by XQuery in SQL Server. PositionalVar is not supported by XQuery in SQL Server.

```
[36]    LetClause    ::=    "let" "$" VarName TypeDeclaration? ":=" ExprSingle ("," "$"
VarName TypeDeclaration? ":=" ExprSingle)*
```

[36] is supported by XQuery in SQL Server with a variation. XQuery in SQL Server does not support TypeDeclaration in the Let clause.

```
[118]    TypeDeclaration    ::=    "as" SequenceType
```

[118] is not supported by XQuery in SQL Server.

```
[35]    PositionalVar    ::=    "at" "$" VarName
```

[35] is not supported by XQuery in SQL Server.

```
[37]    WhereClause    ::=    "where" ExprSingle
```

[37] is supported by XQuery in SQL Server.

```
[38]    OrderByClause    ::=    (("order" "by") | ("stable" "order" "by")) OrderSpecList
```

[38] is supported by XQuery in SQL Server with a variation. The syntax "stable order by" is not supported. However, the order obtained with the syntax "order by" results in a stable order by default.

```
[39]    OrderSpecList    ::=    OrderSpec ("," OrderSpec)*
```

[39] is supported by XQuery in SQL Server.

```
[40]    OrderSpec    ::=    ExprSingle OrderModifier
```

[40] is supported by XQuery in SQL Server.

```
[41]    OrderModifier    ::=    ("ascending" | "descending")? ("empty" ("greatest" |
"least"))? ("collation" URILiteral)?
```

[41] is supported by XQuery in SQL Server with a variation. XQuery in SQL Server does not offer an option to declare empty as greatest or least, and empty is treated as least by default. XQuery in SQL Server does not support collation.

2.1.4.1.40 [XQuery1.0/2] Section 3.8.1 For and Let Clauses

For and Let clauses are supported by XQuery in SQL Server. See section 2.1.4.1.39 ["3.8 FLWOR Expressions"], items [34] and [36].

2.1.4.1.41 [XQuery1.0/2] Section 3.8.2 Where Clause

The Where clause is supported by XQuery in SQL Server. See section 2.1.4.1.39 ["3.8 FLWOR Expressions"], item [37].

2.1.4.1.42 [XQuery1.0/2] Section 3.8.3 Order By and Return Clauses

Order By and Return clauses are supported by XQuery in SQL Server. See section 2.1.4.1.39 ["3.8 FLWOR Expressions"], item [38].

2.1.4.1.43 [XQuery1.0/2] Section 3.9 Ordered and Unordered Expressions

```
[91]    OrderedExpr    ::=    "ordered" "{" Expr "}"
```

[91] is not supported by XQuery in SQL Server. All expressions are evaluated as ordered.

```
[92]    UnorderedExpr    ::=    "unordered" "{" Expr "}"
```

[92] is not supported by XQuery in SQL Server. All expressions are evaluated as ordered.

2.1.4.1.44 [XQuery1.0/2] Section 3.10 Conditional Expressions

```
[45]    IfExpr    ::=    "if" "(" Expr ")" "then" ExprSingle "else" ExprSingle
```

[45] is supported by XQuery in SQL Server with a variation. XQuery in SQL Server permits only ExprSingle in the condition.

2.1.4.1.45 [XQuery1.0/2] Section 3.11 Quantified Expressions

```
[42]    QuantifiedExpr    ::=    ("some" | "every") "$" VarName TypeDeclaration? "in"
ExprSingle ("," "$" VarName TypeDeclaration? "in" ExprSingle)* "satisfies" ExprSingle
```

[42] is supported by XQuery in SQL Server with a variation.

```
[118]    TypeDeclaration    ::=    "as" SequenceType
```

[118] is not supported by XQuery in SQL Server.

2.1.4.1.46 [XQuery1.0/2] Section 3.12.1 Instance Of

```
[54]    InstanceofExpr    ::=    TreatExpr ( "instance" "of" SequenceType )?
```

[54] is supported by XQuery in SQL Server with a variation. The only occurrence indicator for the SequenceType that is supported is the "?" character. The "+" and the "*" occurrence indicators are not supported. The schema-element() and schema-attribute() sequence types are not supported.

XQuery in SQL Server does not support detection of **xs:nil** at run time. (This functionality can be obtained by using the "?" occurrence indicator, as for example xs:string?). For union types, the primitive type, not the derived type, is returned. For the **processing-instruction()** and **document-node()** types, only the form with 0 arguments are allowed by XQuery in SQL Server.

For more information see [MSDN-SeqTypExpXQuery]. For further information on the support for SequenceType, see section 2.1.2.5.3 ["Section 3.12.1 Sequence Type "].

2.1.4.1.47 [XQuery1.0/2] Section 3.12.2 Typeswitch

```
[43]    TypeswitchExpr    ::=    "typeswitch" "(" Expr ")" CaseClause+ "default" ("$"
VarName)? "return" ExprSingle
```

[43] is not supported by XQuery in SQL Server.

```
[44]    CaseClause    ::=    "case" ("$" VarName "as")? SequenceType "return" ExprSingle
```

[44] is not supported by XQuery in SQL Server.

2.1.4.1.48 [XQuery1.0/2] Section 3.12.3 Cast

```
[57]    CastExpr    ::=    UnaryExpr ( "cast" "as" SingleType )?
```

[57] is supported by XQuery in SQL Server with a variation. XQuery in SQL Server requires that the syntax contain the cardinality indicator, the "?" character, to be part of the SingleType expression.

```
[117]    SingleType    ::=    AtomicType "?"?
```

[117] is supported by XQuery in SQL Server with a variation. XQuery in SQL Server requires the "?" cardinality indicator to be present after the AtomicType.

2.1.4.1.49 [XQuery1.0/2] Section 3.12.4 Castable

```
[56]    CastableExpr    ::=    CastExpr ( "castable" "as" SingleType )?
```

[56] is not supported by XQuery in SQL Server.

```
[117]    SingleType    ::=    AtomicType "?"?
```

[117] is not supported by XQuery in SQL Server.

2.1.4.1.50 [XQuery1.0/2] Section 3.12.5 Constructor Functions

XQuery in SQL Server supports the constructor functions. This syntax is an alternate syntax for casting. Any limitations noted in section 2.1.4.1.48 ["Cast"] apply to the constructor functions as well.

2.1.4.1.51 [XQuery1.0/2] Section 3.12.6 Treat

```
[55]    TreatExpr    ::=    CastableExpr ( "treat" "as" SequenceType )?
```

[55] is not supported by XQuery in SQL Server.

2.1.4.1.52 [XQuery1.0/2] Section 3.13 Validate Expressions

```
[63]    ValidateExpr    ::=    "validate" ValidationMode? "{" Expr "}"
```

[63] is not supported by XQuery in SQL Server.

```
[64]    ValidationMode    ::=    "lax" | "strict"
```

[64] is not supported by XQuery in SQL Server.

2.1.4.1.53 [XQuery1.0/2] Section 3.14 Extension Expressions

Extension expressions are not supported by XQuery in SQL Server.

2.1.4.1.54 [XQuery1.0/2] Appendix A.1 EBNF

Appendix A.1 contains additional normative expressions that do not appear in other sections of [XQuery1.0/2]. Expressions not covered elsewhere are included in this section.

```
[65]    ExtensionExpr    ::=    Pragma+ "{" Expr? "}"
```

[65] is not supported by XQuery in SQL Server.

```
[66]    Pragma    ::=    "(#" S? QName (S PragmaContents)? "#)"
```

[66] is not supported by XQuery in SQL Server.

```
[67]    PragmaContents    ::=    (Char* - (Char* '#)' Char*))
```

[67] is not supported by XQuery in SQL Server.

```
[83]    Predicate    ::=    "[" Expr "]"
```

[83] is supported by XQuery in SQL Server.

```
[84]    PrimaryExpr    ::=    Literal | VarRef | ParenthesizedExpr | ContextItemExpr |
FunctionCall | OrderedExpr | UnorderedExpr | Constructor
```

[84] is supported by XQuery in SQL Server.

2.1.4.2 Modules and Prologs

2.1.4.2.1 [XQuery1.0/2] Section 4 Modules and Prologs

```
[1]    Module    ::=    VersionDecl? (LibraryModule | MainModule)
```

[1] is not supported by XQuery in SQL Server.

```
[3]    MainModule    ::=    Prolog QueryBody
```

[3] is supported by XQuery in SQL Server.

```
[4]    LibraryModule    ::=    ModuleDecl Prolog
```

[4] is not supported by XQuery in SQL Server.

```
[6]    Prolog    ::=    ((DefaultNamespaceDecl | Setter | NamespaceDecl | Import) Separator)*
((VarDecl | FunctionDecl | OptionDecl) Separator)*
```

[6] is supported by XQuery in SQL Server with a variation. Some of the declarations listed are not supported. See the expression for each token in this document for the description of its support.

```
[7]    Setter    ::=    BoundarySpaceDecl | DefaultCollationDecl | BaseURIDecl |
ConstructionDecl | OrderingModeDecl | EmptyOrderDecl | CopyNamespacesDecl
```

[7] is not supported by XQuery in SQL Server.

```
[8]    Import    ::=    SchemaImport | ModuleImport
```

[8] is not supported by XQuery in SQL Server.

```
[9]    Separator    ::=    ";"
```

[9] is supported by XQuery in SQL Server.

```
[30]    QueryBody    ::=    Expr
```

[30] is supported by XQuery in SQL Server.

2.1.4.2.2 [XQuery1.0/2] Section 4.1 Version Declaration

```
[2]    VersionDecl    ::=    "xquery" "version" StringLiteral ("encoding" StringLiteral)?
Separator
```

[2] is not supported by XQuery in SQL Server.

2.1.4.2.3 [XQuery1.0/2] Section 4.2 Module Declaration

```
[5]    ModuleDecl    ::=    "module" "namespace" NCName "=" URILiteral Separator
```

[5] is not supported by XQuery in SQL Server.

2.1.4.2.4 [XQuery1.0/2] Section 4.3 Boundary-space Declaration

```
[11]    BoundarySpaceDecl    ::=    "declare" "boundary-space" ("preserve" | "strip")
```

[11] is not supported by XQuery in SQL Server.

2.1.4.2.5 [XQuery1.0/2] Section 4.4 Default Collation Declaration

```
[19]    DefaultCollationDecl    ::=    "declare" "default" "collation" URILiteral
```

[19] is not supported by XQuery in SQL Server.

2.1.4.2.6 [XQuery1.0/2] Section 4.5 Base URI Declaration

```
[20]    BaseURIDecl    ::=    "declare" "base-uri" URILiteral
```

[20] is not supported by XQuery in SQL Server.

2.1.4.2.7 [XQuery1.0/2] Section 4.6 Construction Declaration

```
[25]    ConstructionDecl    ::=    "declare" "construction" ("strip" | "preserve")
```

[25] is not supported by XQuery in SQL Server.

2.1.4.2.8 [XQuery1.0/2] Section 4.7 Ordering Mode Declaration

```
[14]    OrderingModeDecl    ::=    "declare" "ordering" ("ordered" | "unordered")
```

[14] is not supported by XQuery in SQL Server.

2.1.4.2.9 [XQuery1.0/2] Section 4.8 Empty Order Declaration

```
[15]    EmptyOrderDecl    ::=    "declare" "default" "order" "empty" ("greatest" | "least")
```

[15] is not supported by XQuery in SQL Server.

2.1.4.2.10 [XQuery1.0/2] Section 4.9 Copy-Namespaces Declaration

```
[16]    CopyNamespacesDecl    ::=    "declare" "copy-namespaces" PreserveMode "," InheritMode
```

[16] is not supported by XQuery in SQL Server.

```
[17]    PreserveMode    ::=    "preserve" | "no-preserve"
```

[17] is not supported by XQuery in SQL Server.

```
[18]    InheritMode    ::=    "inherit" | "no-inherit"
```

[18] is not supported by XQuery in SQL Server.

2.1.4.2.11 [XQuery1.0/2] Section 4.10 Schema Import

```
[21]    SchemaImport    ::=    "import" "schema" SchemaPrefix? URILiteral ("at" URILiteral
("," URILiteral)*)?
```

[21] is not supported by XQuery in SQL Server. XQuery in SQL Server implicitly imports schema information from a [glossary] [schema collection] that can be associated with an XML data type.

```
[22]    SchemaPrefix    ::=    ("namespace" NCName "=") | ("default" "element" "namespace")
```

[22] is not supported by XQuery in SQL Server.

2.1.4.2.12 [XQuery1.0/2] Section 4.11 Module Import

```
[23]    ModuleImport    ::=    "import" "module" ("namespace" NCName "=")? URILiteral ("at"
URILiteral ("," URILiteral)*)?
```

[23] is not supported by XQuery in SQL Server.

2.1.4.2.13 [XQuery1.0/2] Section 4.12 Namespace Declaration

```
[10]    NamespaceDecl    ::=    "declare" "namespace" NCName "=" URILiteral
```

[10] is supported by XQuery in SQL Server with a variation. XQuery in SQL Server does not check for the validity of the provided URI.

2.1.4.2.14 [XQuery1.0/2] Section 4.13 Default Namespace Declaration

```
[12]    DefaultNamespaceDecl    ::=    "declare" "default" ("element" | "function")
"namespace" URILiteral
```

[12] is supported by XQuery in SQL Server with a variation. XQuery in SQL Server does not support declaration of a default namespace for functions. XQuery in SQL Server does not support the use of a zero-length string to undefine a namespace.

2.1.4.2.15 [XQuery1.0/2] Section 4.14 Variable Declaration

```
[24]    VarDecl    ::=    "declare" "variable" "$" QName TypeDeclaration? ((":=" ExprSingle)
| "external")
```

[24] is not supported by XQuery in SQL Server.

```
[88]    VarName    ::=    QName
```

[88] is not supported by XQuery in SQL Server.

```
[118]    TypeDeclaration    ::=    "as" SequenceType
```

[118] is not supported by XQuery in SQL Server.

2.1.4.2.16 [XQuery1.0/2] Section 4.15 Function Declaration

```
[26]    FunctionDecl    ::=    "declare" "function" QName "(" ParamList? ")" ("as"
SequenceType)? (EnclosedExpr | "external")
```

[26] is not supported by XQuery in SQL Server.

```
[27]    ParamList    ::=    Param ("," Param)*
```

[27] is not supported by XQuery in SQL Server.

```
[28]    Param    ::=    "$" QName TypeDeclaration?
```

[28] is not supported by XQuery in SQL Server.

```
[118]    TypeDeclaration    ::=    "as" SequenceType
```

[118] is not supported by XQuery in SQL Server.

2.1.4.2.17 [XQuery1.0/2] Section 4.16 Option Declaration

```
[13]    OptionDecl    ::=    "declare" "option" QName StringLiteral
```

[13] is not supported by XQuery in SQL Server.

2.1.5 Operators

The operators and sections referred to in this section are found in [XQuery1.0XPath2.0/2].

2.1.5.1 Numeric Operators

XQuery in SQL Server does not support the infinite value NaN. XQuery in SQL Server returns an empty sequence when NaN is specified.

2.1.5.1.1 [XQuery1.0XPath2.0/2] Section 6.2.4 op:numeric-divide

XQuery in SQL Server supports the division operation with the "/" operator.

2.1.5.1.2 [XQuery1.0XPath2.0/2] Section 6.2.5 op:numeric-integer-divide

XQuery in SQL Server does not support an integer-division operator. Casting can be used to obtain a result similar to an integer divide operation.

2.1.5.2 String Operators

The standard [XQuery1.0XPath2.0/2] does not specify any string operators.

2.1.5.3 anyURI Operators

The standard [XQuery1.0XPath2.0/2] does not specify any operators for anyURI.

2.1.5.4 Boolean Operators

2.1.5.4.1 [XQuery1.0XPath2.0/2] Section 9.2.2 op:boolean-less-than

XQuery in SQL Server supports Boolean less-than operations with the "lt" and "ge" operators.

2.1.5.4.2 [XQuery1.0XPath2.0/2] Section 9.2.3 op:boolean-greater-than

XQuery in SQL Server supports Boolean greater-than operations with the "gt" and "le" operators.

2.1.5.5 Durations, Dates, and Times Operators

2.1.5.5.1 [XQuery1.0XPath2.0/2] Section 10.4.1 op:yearMonthDuration-less-than

Because XQuery in SQL Server does not support type **xs:yearMonthDuration** (see 2.1.3.1.32 [xs:yearMonthDuration]), no operators are supported for this type.

2.1.5.5.2 [XQuery1.0XPath2.0/2] Section 10.4.2 op:yearMonthDuration-greater-than

Because XQuery in SQL Server does not support type **xs:yearMonthDuration** (see section 2.1.3.1.32 [xs:yearMonthDuration]), no operators are supported for this type.

2.1.5.5.3 [XQuery1.0XPath2.0/2] Section 10.4.3 op:dayTimeDuration-less-than

Because XQuery in SQL Server does not support type **xs:dayTimeDuration** (see section 2.1.3.1.33 [xs:dayTimeDuration]), no operators are supported for this type.

2.1.5.5.4 [XQuery1.0XPath2.0/2] Section 10.4.4 op:dayTimeDuration-greater-than

Because XQuery in SQL Server does not support type **xs:dayTimeDuration** (see section 2.1.3.1.33 [Hyperlink: section xs:dayTimeDuration]), no operators are supported for this type.

2.1.5.5.5 [XQuery1.0XPath2.0/2] Section 10.4.6 op:dateTime-equal

XQuery in SQL Server supports the dateTime-equal operation on values of type **xs:dateTime** with the "eq" and the "ne" operators.

2.1.5.5.6 [XQuery1.0XPath2.0/2] Section 10.4.8 op:dateTime-greater-than

XQuery in SQL Server supports the dateTime greater-than operation on values of type **xs:dateTime** with the "gt" and the "ge" operators.

2.1.5.5.7 [XQuery1.0XPath2.0/2] Section 10.4.9 op:date-equal

XQuery in SQL Server supports the date equal operation on values of type **xs:date** with the "eq" and the "ne" operators.

2.1.5.5.8 [XQuery1.0XPath2.0/2] Section 10.4.10 op:date-less-than

XQuery in SQL Server supports the date less-than operation on values of type **xs:date** with the "lt" and the "le" operators.

2.1.5.5.9 [XQuery1.0XPath2.0/2] Section 10.4.12 op:time-equal

XQuery in SQL Server supports the equality operation on values of type **xs:time** with the "eq" and the "ne" operators.

2.1.5.5.10 [XQuery1.0XPath2.0/2] Section 10.6.1 op:add-yearMonthDurations

Because XQuery in SQL Server does not support type **xs:yearMonthDuration** (see section 2.1.3.1.32 [Hyperlink: section xs:yearMonthDuration]), no operators are supported for this type.

2.1.5.5.11 [XQuery1.0XPath2.0/2] Section 10.6.2 op:subtract-yearMonthDurations

Because XQuery in SQL Server does not support type **xs:yearMonthDuration** (see section 2.1.3.1.32 [Hyperlink: section xs:yearMonthDuration]), no operators are supported for this type.

2.1.5.5.12 [XQuery1.0XPath2.0/2] Section 10.6.3 op:multiply-yearMonthDuration

Because XQuery in SQL Server does not support type **xs:yearMonthDuration** (see section 2.1.3.1.32 [Hyperlink: section xs:yearMonthDuration]), no operators are supported for this type.

2.1.5.5.13 [XQuery1.0XPath2.0/2] Section 10.6.4 op:divide-yearMonthDuration

Because XQuery in SQL Server does not support type **xs:yearMonthDuration** (see section 2.1.3.1.32 [Hyperlink: section xs:yearMonthDuration]), no operators are supported for this type.

2.1.5.5.14 [XQuery1.0XPath2.0/2] Section 10.6.5 op:divide-yearMonthDuration-by-yearMonthDuration

Because XQuery in SQL Server does not support type **xs:yearMonthDuration** (see section 2.1.3.1.32 [Hyperlink: section xs:yearMonthDuration]), no operators are supported for this type.

2.1.5.5.15 [XQuery1.0XPath2.0/2] Section 10.6.6 op:add-dayTimeDurations

Because XQuery in SQL Server does not support type **xs:dayTimeDuration** (see section 2.1.3.1.33 [Hyperlink: section xs:dayTimeDuration]), no operators are supported for this type.

2.1.5.5.16 [XQuery1.0XPath2.0/2] Section 10.6.7 op:subtract-dayTimeDurations

Because XQuery in SQL Server does not support type **xs:dayTimeDuration** (see section 2.1.3.1.33 [Hyperlink: section xs:dayTimeDuration]), no operators are supported for this type.

2.1.5.5.17 [XQuery1.0XPath2.0/2] Section 10.6.8 op:multiply-dayTimeDuration

Because XQuery in SQL Server does not support type **xs:dayTimeDuration** (see section 2.1.3.1.33 [Hyperlink: section xs:dayTimeDuration]), no operators are supported for this type.

2.1.5.5.18 [XQuery1.0XPath2.0/2] Section 10.6.9 op:divide-dayTimeDuration

As XQuery in SQL Server does not support type **xs:dayTimeDuration** (see section 2.1.3.1.33 [Hyperlink: section xs:dayTimeDuration]), no operators are supported for this type.

2.1.5.5.19 [XQuery1.0XPath2.0/2] Section 10.6.10 op:divide-dayTimeDuration-by-dayTimeDuration

Because XQuery in SQL Server does not support type **xs:dayTimeDuration** (see section 2.1.3.1.33 [Hyperlink: section xs:dayTimeDuration]), no operators are supported for this type.

2.1.5.5.20 [XQuery1.0XPath2.0/2] Section 10.8.4 op:add-yearMonthDuration-to-dateTime

Because XQuery in SQL Server does not support type **xs:yearMonthDuration** (see section 2.1.3.1.32 xs:yearMonthDuration]), no operators are supported for this type.

2.1.5.5.21 [XQuery1.0XPath2.0/2] Section 10.8.5 op:add-dayTimeDuration-to-dateTime

Because XQuery in SQL Server does not support type **xs:dayTimeDuration** (see section 2.1.3.1.33 [Hyperlink: section xs:dayTimeDuration]), no operators are supported for this type.

2.1.5.5.22 [XQuery1.0XPath2.0/2] Section 10.8.6 op:subtract-yearMonthDuration-from-dateTime

Because XQuery in SQL Server does not support type **xs:yearMonthDuration** (see 2.1.3.1.32 xs:yearMonthDuration]), no operators are supported for this type.

2.1.5.5.23 [XQuery1.0XPath2.0/2] Section 10.8.7 op:subtract-dayTimeDuration-from-dateTime

Because XQuery in SQL Server does not support type **xs:dayTimeDuration** (see 2.1.3.1.33 [xs:dayTimeDuration]), no operators are supported for this type.

2.1.5.5.24 [XQuery1.0XPath2.0/2] Section 10.8.8 op:add-yearMonthDuration-to-date

As XQuery in SQL Server does not support type **xs:yearMonthDuration** (see 2.1.3.1.32 xs:yearMonthDuration]), no operators are supported for this type.

2.1.5.5.25 [XQuery1.0XPath2.0/2] Section 10.8.9 op:add-dayTimeDuration-to-date

Because XQuery in SQL Server does not support type **xs:dayTimeDuration** (see 2.1.3.1.33 [xs:dayTimeDuration]), no operators are supported for this type.

2.1.5.5.26 [XQuery1.0XPath2.0/2] Section 10.8.10 op:subtract-yearMonthDuration-from-date

Because XQuery in SQL Server does not support type **xs:yearMonthDuration** (see 2.1.3.1.33 [xs:dayTimeDuration]), no operators are supported for this type.

2.1.5.5.27 [XQuery1.0XPath2.0/2] Section 10.8.11 op:subtract-dayTimeDuration-from-date

Because XQuery in SQL Server does not support type **xs:dayTimeDuration** (see 2.1.3.1.33 [xs:dayTimeDuration]), no operators are supported for this type.

2.1.5.5.28 [XQuery1.0XPath2.0/2] Section 10.8.12 op:add-dayTimeDuration-to-time

Because XQuery in SQL Server does not support type **xs:dayTimeDuration** (see 2.1.3.1.33 [xs:dayTimeDuration]), no operators are supported for this type.

2.1.5.5.29 [XQuery1.0XPath2.0/2] Section 10.8.13 op:subtract-dayTimeDuration-from-time

As XQuery in SQL Server does not support type **xs:dayTimeDuration** (see 2.1.3.1.33 [xs:dayTimeDuration]), no operators are supported for this type.

2.1.5.6 Notation Operators

2.1.5.6.1 [XQuery1.0XPath2.0/2] Section 13.1.1 op:NOTATION-equal

Because XQuery in SQL Server does not support type **xs:NOTATION** (see 2.1.3.1.31 [xs:NOTATION]), no operators are supported for this type.

2.1.5.7 Sequences Operators

2.1.5.7.1 [XQuery1.0XPath2.0/2] Section 15.3.2 op:union

Because XQuery in SQL Server does not support a union operation (see [Hyperlink: section "Section 3.3.3 combining node sequences") expression [52], no operators are supported for this operation.

2.1.5.7.2 [XQuery1.0XPath2.0/2] Section 15.3.3 op:intersect

Because XQuery in SQL Server does not support an intersect operation (see 2.1.4.1.17 ["Section 3.3.3 combining node sequences") expression [53], no operators are supported for this operation.

2.1.5.7.3 [XQuery1.0XPath2.0/2] Section 15.3.4 op:except

Because XQuery in SQL Server does not support an except operation (see 2.1.4.1.17 ["Section 3.3.3 combining node sequences"), no operators are supported for this operation.

2.1.5.7.4 [XQuery1.0XPath2.0/2] Section 15.5.1 op:to

Because XQuery in SQL Server does not support a sequence generation operation (see 2.1.4.1.17 ["Section 3.3.3 combining node sequences") expression [49], no operators are supported for this operation.

2.1.5.8 Context

The standard [XQuery1.0XPath2.0/2] does not specify any operators for Context.

2.1.6 Functions

The functions and sections referred to in this section are found in [XQuery1.0XPath2.0/2].

2.1.6.1 Accessors

2.1.6.1.1 [XQuery1.0XPath2.0/2] Section 2.1 fn:node-name

The **fn:node-name** function is not supported by XQuery in SQL Server.

2.1.6.1.2 [XQuery1.0XPath2.0/2] Section 2.2 fn:nilled

The **fn:nilled** function is not supported by XQuery in SQL Server.

2.1.6.1.3 [XQuery1.0XPath2.0/2] Section 2.3 fn:string

The **fn:string** function is supported with a variation by XQuery in SQL Server. **fn:string**() (with no arguments) can be used only within a context-dependent predicate.

2.1.6.1.4 [XQuery1.0XPath2.0/2] Section 2.5 fn:base-uri

The **fn:base-uri** function is not supported by XQuery in SQL Server.

2.1.6.1.5 [XQuery1.0XPath2.0/2] Section 2.6 fn:document-uri

The **fn:document-uri** function is not supported by XQuery in SQL Server.

2.1.6.2 Numeric Functions

2.1.6.2.1 [XQuery1.0XPath2.0/2] Section 6.4.1 fn:abs

The **fn:abs** function is not supported by XQuery in SQL Server.

2.1.6.2.2 [XQuery1.0XPath2.0/2] Section 6.4.2 fn:ceiling

The **fn:ceiling** function is supported by XQuery in SQL Server with variations. The standard [XQuery1.0XPath2.0/2] requires that four types of arguments be returned as the same type as the argument: these are **xs:float**, **xs:double**, **xs:decimal**, **xs:integer**. XQuery in SQL Server returns three of those types as the same type as the argument: **xs:float**, **xs:double**, **xs:decimal**.

In XQuery in SQL Server, if the input is **xdt:untypedAtomic**, it is cast to **xs:double**. All other numeric types are returned as the base type from which they are derived.

2.1.6.2.3 [XQuery1.0XPath2.0/2] Section 6.4.3 fn:floor

The **fn:floor** function is supported by XQuery in SQL Server with variations. The standard [XQuery1.0XPath2.0/2] requires that four types of arguments be returned as the same type as the argument: these are **xs:float**, **xs:double**, **xs:decimal**, **xs:integer**. XQuery in SQL Server returns three of those types as the same type as the argument: **xs:float**, **xs:double**, **xs:decimal**.

In XQuery in SQL Server, if the input is **xdt:untypedAtomic**, it is cast to **xs:double**. All other numeric types are returned as the base type from which they are derived.

2.1.6.2.4 [XQuery1.0XPath2.0/2] Section 6.4.4 fn:round

The **fn:round** function is supported by XQuery in SQL Server with variations. The standard [XQuery1.0XPath2.0/2] requires that four types of arguments be returned as the same type as the

argument: these are **xs:float**, **xs:double**, **xs:decimal**, **xs:integer**. XQuery in SQL Server returns three of those types as the same type as the argument: **xs:float**, **xs:double**, **xs:decimal**.

In XQuery in SQL Server, if the input is **xdt:untypedAtomic**, it is cast to **xs:double**. All other numeric types are returned as the base type from which they are derived. If the argument is between -0.5 and 0, XQuery in SQL Server returns the value 0, instead of the value -0, which is specified in [XQuery1.0XPath2.0/2].

2.1.6.2.5 [XQuery1.0XPath2.0/2] Section 6.4.5 fn:round-half-to-even

The **fn:round-half-to-even** function is not supported by XQuery in SQL Server.

2.1.6.3 String Functions

2.1.6.3.1 [XQuery1.0XPath2.0/2] Section 7.2.1 fn:codepoints-to-string

The **fn:codepoints-to-string** function is not supported by XQuery in SQL Server.

2.1.6.3.2 [XQuery1.0XPath2.0/2] Section 7.2.2 fn:string-to-codepoints

The **fn:string-to-codepoints** function is not supported by XQuery in SQL Server.

2.1.6.3.3 [XQuery1.0XPath2.0/2] Section 7.3.2 fn:compare

The **fn:compare** function is not supported by XQuery in SQL Server.

2.1.6.3.4 [XQuery1.0XPath2.0/2] Section 7.3.3 fn:codepoint-equal

The **fn:codepoints-equal** function is not supported by XQuery in SQL Server.

2.1.6.3.5 [XQuery1.0XPath2.0/2] Section 7.4.1 fn:concat

The **fn:concat** function is supported by XQuery in SQL Server with a variation. [XQuery1.0XPath2.0/2] specifies that the arguments are of type **anyAtomicType**, but XQuery in SQL Server specifies that arguments be of type **xs:string** or a type that is derived from **xs:string**.

2.1.6.3.6 [XQuery1.0XPath2.0/2] Section 7.4.2 fn:string-join

The **fn:string-join** function is not supported by XQuery in SQL Server.

2.1.6.3.7 [XQuery1.0XPath2.0/2] Section 7.4.3 fn:substring

The **fn:substring** function is supported by XQuery in SQL Server with a variation. The W3C specification [XQuery1.0XPath2.0/2] specifies that the second argument, *$startingLoc*, and the third argument, *$length*, are of type **xs:double**, but XQuery in SQL Server specifies arguments of type **xs:decimal**. The **fn:substring** function in XQuery in SQL Server version SQL Server 2008 R2 counts surrogate pairs as two characters, not one. Starting with XQuery in Microsoft SQL Server 2012, surrogate pairs are counted as only one character].

2.1.6.3.8 [XQuery1.0XPath2.0/2] Section 7.4.4 fn:string-length

The **fn:string-length** function is supported by XQuery in SQL Server with a variation. XQuery in SQL Server supports only the form string-length() (with no specified arguments) within a context dependent predicate. The **fn:string-length** function in XQuery in SQL Server version SQL Server

2008 R2 counts surrogate pairs as two characters, not one. Starting with XQuery in Microsoft SQL Server 2012, this has been corrected and surrogate pairs are only counted as one character.

2.1.6.3.9 [XQuery1.0XPath2.0/2] Section 7.4.5 fn:normalize-space

The **fn:normalize-space** function is not supported by XQuery in SQL Server.

2.1.6.3.10 [XQuery1.0XPath2.0/2] Section 7.4.6 fn:normalize-unicode

The **fn:normalize-unicode** function is not supported by XQuery in SQL Server.

2.1.6.3.11 [XQuery1.0XPath2.0/2] Section 7.4.9 fn:translate

The **fn:translate** function is not supported by XQuery in SQL Server.

2.1.6.3.12 [XQuery1.0XPath2.0/2] Section 7.4.10 fn:encode-for-uri

The **fn:encode-for-uri** function is not supported by XQuery in SQL Server.

2.1.6.3.13 [XQuery1.0XPath2.0/2] Section 7.4.11 fn:iri-to-uri

The **fn:encode-for-url** function is not supported by XQuery in SQL Server.

2.1.6.3.14 [XQuery1.0XPath2.0/2] Section 7.4.12 fn:escape-html-uri

The **fn:escape-html-uri** function is not supported by XQuery in SQL Server.

2.1.6.3.15 [XQuery1.0XPath2.0/2] Section 7.5.1 fn:contains

The **fn:contains** function is supported by XQuery in SQL Server with a variation. [XQuery1.0XPath2.0/2] specifies three arguments for the **fn:contains** function. XQuery in SQL Server supports only the first two arguments, and does not allow inclusion of the third argument, *$collation*. XQuery in SQL Server requires that the string specified as *$arg2* must be less than or equal to 4000 characters in length.

2.1.6.3.16 [XQuery1.0XPath2.0/2] Section 7.5.2 fn:starts-with

The **fn:starts-with** function is not supported by XQuery in SQL Server.

2.1.6.3.17 [XQuery1.0XPath2.0/2] Section 7.5.3 fn:ends-with

The **fn:ends-with** function is not supported by XQuery in SQL Server.

2.1.6.3.18 [XQuery1.0XPath2.0/2] Section 7.5.4 fn:substring-before

The **fn:substring-before** function is not supported by XQuery in SQL Server.

2.1.6.3.19 [XQuery1.0XPath2.0/2] Section 7.5.5 fn:substring-after

The **fn:substring-after** function is not supported by XQuery in SQL Server.

2.1.6.3.20 [XQuery1.0XPath2.0/2] Section 7.6.2 fn:matches

The **fn:matches** function is not supported by XQuery in SQL Server.

2.1.6.3.21 [XQuery1.0XPath2.0/2] Section 7.6.3 fn:replace

The **fn:replace** function is not supported by XQuery in SQL Server.

2.1.6.3.22 [XQuery1.0XPath2.0/2] Section 7.6.4 fn:tokenize

The **fn:tokenize** function is not supported by XQuery in SQL Server.

2.1.6.4 anyURI Functions

2.1.6.4.1 [XQuery1.0XPath2.0/2] Section 8.1 fn:resolve-uri

The **fn:resolve-uri** function is not supported by XQuery in SQL Server.

2.1.6.5 Durations, Dates, and Times Functions

2.1.6.5.1 [XQuery1.0XPath2.0/2] Section 10.5.1 fn:years-from-duration

The **fn:years-from-duration** function is not supported by XQuery in SQL Server.

2.1.6.5.2 [XQuery1.0XPath2.0/2] Section 10.5.2 fn:months-from-duration

The **fn:months-from-duration** function is not supported by XQuery in SQL Server.

2.1.6.5.3 [XQuery1.0XPath2.0/2] Section 10.5.3 fn:days-from-duration

The **fn:days-from-duration** function is not supported by XQuery in SQL Server.

2.1.6.5.4 [XQuery1.0XPath2.0/2] Section 10.5.4 fn:hours-from-duration

The **fn:hours-from-duration** function is not supported by XQuery in SQL Server.

2.1.6.5.5 [XQuery1.0XPath2.0/2] Section 10.5.5 fn:minutes-from-duration

The **fn:minutes-from-duration** function is not supported by XQuery in SQL Server.

2.1.6.5.6 [XQuery1.0XPath2.0/2] Section 10.5.6 fn:seconds-from-duration

The **fn:seconds-from-duration** function is not supported by XQuery in SQL Server.

2.1.6.5.7 [XQuery1.0XPath2.0/2] Section 10.5.7 fn:year-from-dateTime

The **fn:year-from-dateTime** function is not supported by XQuery in SQL Server.

2.1.6.5.8 [XQuery1.0XPath2.0/2] Section 10.5.8 fn:month-from-dateTime

The fn**:month-from-dateTime** function is not supported by XQuery in SQL Server.

2.1.6.5.9 [XQuery1.0XPath2.0/2] Section 10.5.9 fn:day-from-dateTime

The **fn:day-from-dateTime** function is not supported by XQuery in SQL Server.

2.1.6.5.10 [XQuery1.0XPath2.0/2] Section 10.5.10 fn:hours-from-dateTime

The **fn:hours-from-dateTime** function is not supported by XQuery in SQL Server.

2.1.6.5.11 [XQuery1.0XPath2.0/2] Section 10.5.11 fn:minutes-from-dateTime

The **fn: minutes-from-dateTime** function is not supported by XQuery in SQL Server.

2.1.6.5.12 [XQuery1.0XPath2.0/2] Section 10.5.12 fn:seconds-from-dateTime

The **fn:seconds-from-dateTime** function is not supported by XQuery in SQL Server.

2.1.6.5.13 [XQuery1.0XPath2.0/2] Section 10.5.13 fn:timezone-from-dateTime

The **fn:timezone-from-dateTime** function is not supported by XQuery in SQL Server.

2.1.6.5.14 [XQuery1.0XPath2.0/2] Section 10.5.14 fn:year-from-date

The fn:year-from-date function is not supported by XQuery in SQL Server.

2.1.6.5.15 [XQuery1.0XPath2.0/2] Section 10.5.15 fn:month-from-date

The **fn:month-from-date** function is not supported by XQuery in SQL Server.

2.1.6.5.16 [XQuery1.0XPath2.0/2] Section 10.5.16 fn:day-from-date

The **fn:day-from-date** function is not supported by XQuery in SQL Server.

2.1.6.5.17 [XQuery1.0XPath2.0/2] Section 10.5.17 fn:timezone-from-date

The **fn:timezone-from-date** function is not supported by XQuery in SQL Server.

2.1.6.5.18 [XQuery1.0XPath2.0/2] Section 10.5.18 fn:hours-from-time

The **fn:hours-from-time** function is not supported by XQuery in SQL Server.

2.1.6.5.19 [XQuery1.0XPath2.0/2] Section 10.5.19 fn:minutes-from-time

The **fn:minutes-from-time** function is not supported by XQuery in SQL Server.

2.1.6.5.20 [XQuery1.0XPath2.0/2] Section 10.5.20 fn:seconds-from-time

The **fn:seconds-from-time** function is not supported by XQuery in SQL Server.

2.1.6.5.21 [XQuery1.0XPath2.0/2] Section 10.5.21 fn:timezone-from-time

The **fn:timezone-from-time** function is not supported by XQuery in SQL Server.

2.1.6.5.22 [XQuery1.0XPath2.0/2] Section 10.7.1 fn:adjust-dateTime-to-timezone

The **fn:resolve-uri** function is not supported by XQuery in SQL Server.

2.1.6.5.23 [XQuery1.0XPath2.0/2] Section 10.7.2 fn:adjust-date-to-timezone

The **fn:adjust-date-to-timezone** function is not supported by XQuery in SQL Server.

2.1.6.5.24 [XQuery1.0XPath2.0/2] Section 10.7.3 fn:adjust-time-to-timezone

The **fn:adjust-time-to-timezone** function is not supported by XQuery in SQL Server.

2.1.6.6 QNames Functions

2.1.6.6.1 [XQuery1.0XPath2.0/2] Section 11.1.1 fn:resolve-QName

The **fn:resolve-QName** function is not supported by XQuery in SQL Server.

2.1.6.6.2 [XQuery1.0XPath2.0/2] Section 11.1.2 fn:QName

The **fn:QName** function is not supported by XQuery in SQL Server.

2.1.6.6.3 [XQuery1.0XPath2.0/2] Section 11.2.2 fn:prefix-from-QName

The **fn:prefix-from-QName** function is not supported by XQuery in SQL Server.

2.1.6.6.4 [XQuery1.0XPath2.0/2] Section 11.2.3 fn:local-name-from-QName

The **fn:local-name-from-QName** function is supported by XQuery in SQL Server with a variation. The form **fn:local-name()** (with no arguments) can be used only within a context-dependent predicate.

2.1.6.6.5 [XQuery1.0XPath2.0/2] Section 11.2.5 fn:namespace-uri-for-prefix

The **fn:namespace-uri-from-QName** function is not supported by XQuery in SQL Server.

2.1.6.6.6 [XQuery1.0XPath2.0/2] Section 11.2.6 fn:in-scope-prefixes

The **fn:in-scope-prefixes** function is not supported by XQuery in SQL Server.

2.1.6.7 base64Binary and hexBinary Functions

The standard does not specify any functions for **base64Binary** and **hexBinary**.

2.1.6.8 Notation Functions

The standard does not specify any functions for **Notation**.

2.1.6.9 Nodes Functions

2.1.6.9.1 [XQuery1.0XPath2.0/2] Section 14.1 fn:name

The **fn:name** function is not supported by XQuery in SQL Server.

2.1.6.9.2 [XQuery1.0XPath2.0/2] Section 14.3 fn:namespace-uri

The **fn:namespace-uri** function is supported by XQuery in SQL Server with a variation. XQuery in SQL Server returns a value of type **xs:string**, while [XQuery1.0XPath2.0/2] specifies that a value of

type **xs:anyURI** be returned. In XQuery in SQL Server, the form **fn:namespace-uri()** (with no arguments) can be used only within a context-dependent predicate.

2.1.6.9.3 [XQuery1.0XPath2.0/2] Section 14.4 fn:number

The **fn:number** function is supported by XQuery in SQL Server with a variation. XQuery in SQL Server supports the argument *$arg* as type node, while [XQuery1.0XPath2.0/2] specifies the argument *$arg* to be of type **xs:anyAtomicType**. The form **fn:number()** (with no arguments) can be used only within a context-dependent predicate.

2.1.6.9.4 [XQuery1.0XPath2.0/2] Section 14.5 fn:lang

The **fn:lang** function is not supported by XQuery in SQL Server.

2.1.6.9.5 [XQuery1.0XPath2.0/2] Section 14.9 fn:root

The **fn:root** function is not supported by XQuery in SQL Server.

2.1.6.10 Sequences Functions

2.1.6.10.1 [XQuery1.0XPath2.0/2] Section 15.1.1 fn:boolean

The **fn:boolean** function is not supported by XQuery in SQL Server.

2.1.6.10.2 [XQuery1.0XPath2.0/2] Section 15.1.3 fn:index-of

The **fn:index**-of function is not supported by XQuery in SQL Server.

2.1.6.10.3 [XQuery1.0XPath2.0/2] Section 15.1.5 fn:exists

The **fn:exists** function is not supported by XQuery in SQL Server.

2.1.6.10.4 [XQuery1.0XPath2.0/2] Section 15.1.6 fn:distinct-values

The **fn:distinct-values** function is supported by XQuery in SQL Server with a variation. The *$collation* optional argument is not supported.

XQuery in SQL Server requires that all of the arguments to the function be of the same type, and that the types either be one of the following types or be descended from one of them:

- One of the numeric base types
- One of the date/time base types
- The**xs:string** type
- The**xs:boolean** type
- The**xdt:untypedAtomic** type

The**xdt:untypedAtomic** type is cast to **xs:string**. The **fn:distinct-values** function is not supported for **xs:duration**.

2.1.6.10.5 [XQuery1.0XPath2.0/2] Section 15.1.7 fn:insert-before

The **fn:insert-before** function is not supported by XQuery in SQL Server.

2.1.6.10.6 [XQuery1.0XPath2.0/2] Section 15.1.8 fn:remove

The **fn:remove** function is not supported by XQuery in SQL Server.

2.1.6.10.7 [XQuery1.0XPath2.0/2] Section 15.1.9 fn:reverse

The **fn:reverse** function is not supported by XQuery in SQL Server.

2.1.6.10.8 [XQuery1.0XPath2.0/2] Section 15.1.10 fn:subsequence

The **fn:subsequence** function is not supported by XQuery in SQL Server.

2.1.6.10.9 [XQuery1.0XPath2.0/2] Section 15.1.11 fn:unordered

The **fn:unordered** function is not supported by XQuery in SQL Server.

2.1.6.10.10 [XQuery1.0XPath2.0/2] Section 15.2.1 fn:zero-or-one

The **fn:zero-or-one** function is not supported by XQuery in SQL Server.

2.1.6.10.11 [XQuery1.0XPath2.0/2] Section 15.2.2 fn:one-or-more

The **fn:one-or-more** function is not supported by XQuery in SQL Server.

2.1.6.10.12 [XQuery1.0XPath2.0/2] Section 15.2.3 fn:exactly-one

The **fn:exactly-one** function is not supported by XQuery in SQL Server.

2.1.6.10.13 [XQuery1.0XPath2.0/2] Section 15.3.1 fn:deep-equal

The **fn:deep-equal** function is not supported by XQuery in SQL Server.

2.1.6.10.14 [XQuery1.0XPath2.0/2] Section 15.4.2 fn:avg

The **fn:avg** function is supported by XQuery in SQL Server, with a variation. The W3C specification [XQuery1.0XPath2.0/2] specifies **xs:anyAtomicType** for the argument *$arg*. XQuery in SQL Server requires that the values passed as *$arg* must be a subtype of exactly one of the three built-in numeric base types or the **xdt:untypedAtomic** type. The values cannot be a mixture of numeric base types and **xdt:untypedAtomic**.

2.1.6.10.15 [XQuery1.0XPath2.0/2] Section 15.4.3 fn:max

The **fn:max** function is supported by XQuery in SQL Server, with a variation. [XQuery1.0XPath2.0/2] specifies **xs:anyAtomicType** for the argument *$arg*.

[XQuery1.0XPath2.0/2] specifies an optional second argument, *$collation*. XQuery in SQL Server does not support the optional argument *$collation*.

XQuery in SQL Server requires that all the values passed as *$arg* must be of exactly one type. Allowed types are a subtype of exactly one of the three built-in numeric base types, the date/time base types, **xs:string**, **xs:boolean**, or **xdt:untypedAtomic**.

2.1.6.10.16 [XQuery1.0XPath2.0/2] Section 15.4.4 fn:min

The **fn:min** function is supported by XQuery in SQL Server, with a variation. [XQuery1.0XPath2.0/2] specifies **xs:anyAtomicType** for the argument *$arg*. [XQuery1.0XPath2.0/2] specifies an optional second argument, *$collation*. XQuery in SQL Server does not support the optional argument *$collation*.

XQuery in SQL Server requires that all the values passed as *$arg* must be of exactly one type. Allowed types are a subtype of exactly one of the three built-in numeric base types, the date/time base types, **xs:string**, **xs:boolean**, or **xdt:untypedAtomic**.

2.1.6.10.17 [XQuery1.0XPath2.0/2] Section 15.4.5 fn:sum

The **fn:sum** function is supported by XQuery in SQL Server, with a variation. [XQuery1.0XPath2.0/2] specifies an optional second argument, *$zero*. XQuery in SQL Server does not support the optional argument *$zero*.

[XQuery1.0XPath2.0/2] specifies **xs:anyAtomicType** for the argument *$arg*. XQuery in SQL Server requires that the values passed as *$arg* must be a subtype of exactly one of the three built-in numeric base types or **xdt:untypedAtomic**. The values cannot be a mixture of types.

2.1.6.10.18 [XQuery1.0XPath2.0/2] Section 15.5.2 fn:id

The **fn:id** function is supported by XQuery in SQL Server, with a variation. The W3C specification [XQuery1.0XPath2.0/2] specifies an optional second argument, *$node*. XQuery in SQL Server does not support the optional argument *$node*.

[XQuery1.0XPath2.0/2] specifies **xs:string** for the argument *$arg*. XQuery in SQL Server specifies that the argument *$arg* is of type **xs:IDREF**.

2.1.6.10.19 [XQuery1.0XPath2.0/2] Section 15.5.3 fn:idref

The **fn:idref** function is not supported by XQuery in SQL Server.

2.1.6.10.20 [XQuery1.0XPath2.0/2] Section 15.5.4 fn:doc

The **fn:doc** function is not supported by XQuery in SQL Server.

2.1.6.10.21 [XQuery1.0XPath2.0/2] Section 15.5.5 fn:doc-available

The **fn:doc**-available function is not supported by XQuery in SQL Server.

2.1.6.10.22 [XQuery1.0XPath2.0/2] Section 15.5.6 fn:collection

The **fn:collection** function is not supported by XQuery in SQL Server.

2.1.6.10.23 [XQuery1.0XPath2.0/2] Section 15.5.7 fn:element-with-id

The **fn:element**-with-id function is not supported by XQuery in SQL Server.

2.1.6.11 Context

2.1.6.11.1 [XQuery1.0XPath2.0/2] Section 16.1 fn:position

The **fn:position** function is supported by XQuery in SQL Server with a variation. The form **fn:position()** (with no arguments) can be used only within a context-dependent predicate.

2.1.6.11.2 [XQuery1.0XPath2.0/2] Section 16.2 fn:last

The **fn:last** function is supported by XQuery in SQL Server with a variation. The form **fn:last()** (with no arguments) can be used only within a context-dependent predicate.

2.1.6.11.3 [XQuery1.0XPath2.0/2] Section 16.3 fn:current-dateTime

The **fn:current-dateTime** function is not supported by XQuery in SQL Server.

2.1.6.11.4 [XQuery1.0XPath2.0/2] Section 16.4 fn:current-date

The **fn:current-date** function is not supported by XQuery in SQL Server.

2.1.6.11.5 [XQuery1.0XPath2.0/2] Section 16.5 fn:current-time

The **fn:current-time** function is not supported by XQuery in SQL Server.

2.1.6.11.6 [XQuery1.0XPath2.0/2] Section 16.6 fn:implicit-timezone

The **fn:implicit-timezone** function is not supported by XQuery in SQL Server.

2.1.6.11.7 [XQuery1.0XPath2.0/2] Section 16.7 fn:default-collation

The **fn:default-collation** function is not supported by XQuery in SQL Server.

2.1.6.11.8 [XQuery1.0XPath2.0/2] Section 16.8 fn:static-base-uri

The **fn:static-base-uri** function is not supported by XQuery in SQL Server.

2.2 Clarifications

The following subsections identify clarifications relative to XQuery in SQL Server.

Unless otherwise stated, the specified products conform to all SHOULD and RECOMMENDED behavior in XQuery in SQL Server.

2.2.1 [XDM/2] Section 3.3.1.3, Relationship Between Typed-Value and String-Value

C00001:

The specification allows values to be stored as a typed value or as a string representation of a typed value. XQuery in SQL Server stores values as a typed value.

2.2.2 [XDM/2] Section 3.3.1.4, Pattern Facets

C00002:

The specification states that pattern facets, by their nature, may not be round-trippable. XQuery in SQL Server stores values as a typed representation, so pattern facets that are not encapsulated in the storage of a typed value are not round-trippable.

2.2.3 [XDM/2] Section 6.2.4, Element Nodes, Construction from a PSVI

C00003:

The specification states that the processor MAY use a sequence of nodes that are processing instruction nodes or comment nodes, in the children collection. XQuery in SQL Server supports processing instruction nodes or comment nodes in the collection of children nodes.

If an element has simple typed content and contains either processing instruction or comment nodes or both, XQuery in SQL Server moves all the processing instruction children nodes and all the comment children nodes to the front of the collection of children nodes, preserving order, and combines all the text children nodes into one typed text node value that follows the other children nodes.

2.2.4 [XQuery1.0/2] Section 5.2, Optional Features

2.2.4.1 [XQuery1.0/2] Section 5.2.1, Schema Import Feature

C00004:

XQuery in SQL Server does not support the schema import feature. However, XQuery in SQL Server implicitly supports schema import by using the XML schema collection that is associated with an XML data type.

2.2.4.2 [XQuery1.0/2] Section 5.2.2, Schema Validation Feature

C00005:

XQuery in SQL Server supports the schema validation feature.

2.2.4.3 [XQuery1.0/2] Section 5.2.3, Static Typing Feature

C00006:

XQuery in SQL Server supports the static typing feature.

2.2.4.4 [XQuery1.0/2] Section 5.2.4, Full Axis Feature

C00007:

XQuery in SQL Server does not support the Full Axis feature

2.2.4.5 [XQuery1.0/2] Section 5.2.5, Module Feature

C00008:

XQuery in SQL Server does not support the Module feature.

2.2.4.6 [XQuery1.0/2] Section 5.2.6, Serialization Feature

C00009:

XQuery in SQL Server does not support the Serialization feature.

2.2.5 sql:column Function

E00001:

XQuery in SQL Server extends the [XQuery1.0XPath2.0/2] specification with the **sql:column** function. See [MSDN-sqlcolumnXQuery].

2.2.6 sql:variable Function

E00002:

XQuery in SQL Server extends the [XQuery1.0XPath2.0/2] specification with the **sql:variable** function. See [MSDN-sqlvariableXQuery].

2.3 Error Handling

None.

2.4 Security

None.

3 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

4 Index