



M A R K
L O G I C

Unlock Content™

MarkLogic Server Extensions to XQuery

Agenda



- Search Extensions
- Update Extensions
- Admin Extensions
- Mixing XQuery versions
- Navigating the Online Documentation

Search Extensions



What is “Search”?



- Selecting from a collection of items those "relevant" to some search condition
- Results are relevance ranked
- Relevance is determined statistically
 - Word frequency (%) in the fragment vs. word frequency in the full document set
 - Higher % than average = higher score
 - Increased scores for: more appearances, smaller fragments, rare terms

cts:word-query()



- Matches a word or phrase
- Affected by phrase through and phrase around
- No structural constraints

```
cts:word-query("now is the time")
```

Options for cts:query



- Each query-building function accepts an xs:string* options list
 - "case-sensitive", "case-insensitive"
 - "punctuation-sensitive", "punctuation-insensitive"
 - "stemmed", "unstemmed"
 - "wildcarded", "unwildcarded"
 - "diacritic-sensitive", "diacritic-insensitive"
 - "lang=xx"

```
cts:word-query(  
    "now is the time",  
    ("case-sensitive", "punctuation-insensitive")  
)
```

cts:element-word-query()



- Matches named element(s) *containing* words or phrases

```
cts:element-word-query(  
    xs:QName("function"), "math quiz") )]
```

- Matches “function” elements with the *phrase* “math quiz”
- Use a sequence of element QNames, for a logical OR across multiple elements
- Use a sequence of words or phrases, for logical OR
- Also cts:element-attribute-word-query()

cts:element-value-query()



- Matches named element(s) with specific value(s)

```
cts:element-value-query(  
    xs:QName( "function" ), "math quiz" ) ]
```

- Matches “function” elements with the *text value* “math quiz”
- Supports options for stemming, wildcards, etc.
- Also cts:element-attribute-value-query()

cts:near-query()



- Proximity search
- Match "MarkLogic" within three words of "Server", in that order

```
cts:near-query(  
    (cts:word-query("MarkLogic"),  
     cts:word-query("Server")),  
    3, "ordered"  
)
```

- Combine with other query types
 - Any number of query terms
 - Any kind of cts:query

cts:element-range-query()



- Matches value ranges within a named element

```
cts:element-range-query(  
    xs:QName( "page-number" ), ">", 10 )
```

- Matches “page-number” elements with a value greater than 10
- Requires a Range Index on “page-number”, with a numeric type
- Also cts:element-attribute-range-query()

cts:contains()



- Returns true if any of a sequence of nodes matches a query

```
cts:contains($nodes as node()*,  
             $query as  
             cts:query  
           ) as xs:boolean?
```

Metadata Queries



- `cts:document-query()`
matches by document-URI
- `cts:collection-query()`
matches by collection-URI
- `cts:directory-query()`
matches by directory-URI

Boolean Combinations



- Queries may be combined using booleans
- `cts:and-query(($q1, $q2, ...))`
 - Intersection of nodes from \$q1 and \$q2 queries
- `cts:or-query(($q1, $q2, ...))`
 - Union of nodes from \$q1 and \$q2 queries
- `cts:not-query(($q1, $q1, ...))`
 - Matches nodes that do **not** match the terms
- `cts:and-not-query($q1, $q2)`
 - Difference between queries, all \$q1 minus \$q2

Searching non-XML Content



- MarkLogic supports BLOB and CLOB documents
- BLOBs have binary() root nodes
 - No internal structure
 - Not fragmented
 - Not searchable
- CLOBs have text() roots
 - No internal structure
 - Not fragmented
 - Search still works

Update Extensions



Updates



- The requirements for XQuery 1.0 do not include an update mechanism
- Mark Logic provides update abilities with a set of update built-ins
 - Add and remove documents
 - Add and remove nodes
 - Add and remove document properties
 - Add and remove collections
 - Add and remove permissions

Adding Documents



- `xdmp:document-load($location as xs:string, $options as node()) as empty-sequence()`
 - Loads documents from files or web servers
 - Options for uri, collections, etc.

```
xdmp:document-load("/tmp/file.rss",
  <options xmlns="xdmp:document-load">
    <uri>file.xml</uri>
    <repair>full</repair>
    <collections>
      <collection>feeds</collection>
    </collections>
    <format>xml</format>
  </options>)
```

Loading Multiple Files



- What if you want to load more than one file?

```
for $file in $filesToLoad
return
try {
<li>
{ xdmp:document-load($file) },
{ $file } loaded
</li>
} catch ($exception) {
<span>Problem while loading { $file }:
<br/><i xmlns:e="http://marklogic.com/xdmp/error">
{$exception/e:message/text()}</i>
</span>
}
```

Loading Options



- Supported options (see docs for defaults)
 - URI
 - Permissions
 - Collections
 - Quality
 - Default-namespace
 - Repair
 - Format
 - Default-language
 - Forests
 - Encoding
- Not all loading methods offer all options
 - xdmp:document-load and XCC support full range of options
 - Other methods are limited

- Content comes in three formats:
 - XML: Structured text
 - Text: Unstructured text, as single text() node
 - Binary: Byte-oriented, as single binary() node
- Load format depends on file extension
 - See "Mimetypes" configuration
 - Careful: .html is text, unknown is binary!
- Specifying the type for a URI without a file extension
 - How do you load “foobar” as an xml?
 - Change the mimetypes definition to XML for unknowns (not recommended)
 - Use xdmp:document-load and specify the type

Content Format

M A R K
LOGIC

MarkLogic Server
STANDARD EDITION
4.1-1

I-16898.synapse.com
November 25, 2009
11:46 AM

Configure Create Help

MIME Types Configuration

File extension to MIME type specifications

name <i>A MIME type name.</i>	extensions <i>A list of file extensions.</i>	format <i>The document format.</i>	
application/rtf		binary	delete
application/xml		xml	delete
application/vnd.openxmlformats		binary	delete
application/x-unknown-content	All others	binary	delete
application/postscript	ai,eps,ps	binary	delete
audio/x-aiff	aif,aiff,aifc	binary	delete
text/plain	asc,txt	text	delete
audio/basic	au,snd	binary	delete

ok cancel

Encoding



- Content stored in UTF-8
- Loading non-UTF-8 content:
 - Specify encoding during ingestion (or exception is thrown)
 - Server attempts to translate to UTF-8
 - Options to:
 - xdmp:document-load
 - xdmp:document-get
 - xdmp:zip-get
- UTF-8 is the default if unspecified by loading option or HTTP header

Adding Documents



- `xdmp:document-insert($uri as xs:string, $root as node()) as empty-sequence()`
 - Loads from an in-memory node
 - Lets you insert a generated document

```
xdmp:document-insert("/data/ranks.xml",
<ranks>{
  for $i in (/book/@fpi)
  let $rank := string(xdmp:http-get(
    concat("http://webservices.amazon.com/onca/xml?",
      "Service=AWSECommerceService",
      "&AWSAccessKeyId=1B1TT&",
      "Operation=ItemLookup&ItemId=", string($i),
      "&ResponseGroup=SalesRank"))/*:SalesRank)
  return <rank>{ ($i, $rank) }</rank>
}</ranks>)
```

Deleting Documents



- **xdmp:document-delete(\$uri as xs:string)
as empty-sequence()**
 - Deletes a generated document after use
 - Deleting a document when it is deleted from another source

```
xdmp:document-delete( "foo.xml" )
```

Adding Nodes

- Nodes are added relative to an existing node
 - `xdmp:node-insert-before($ sibling as node(), $ new as node())`
 - `xdmp:node-insert-after($ sibling as node(), $ new as node())`
 - `xdmp:node-insert-child($ parent as node(), $ new as node())`

```
(: transaction 1 :)
xdmp:document-insert("example.xml", <a/>)
```

```
(: transaction 2 :)
xdmp:node-insert-child(doc("example.xml")/a,
    attribute b { "bbb" })
```

```
(: query the result :)
doc("example.xml")
= <a b="bbb" />
```

Replacing Nodes



- You can also replace a node in place
 - `xdmp:node-replace($old as node(), $new as node())`
- Or delete a node
 - `xdmp:node-delete($old as node())`

```
(: transaction 1 :)
xdmp:document-insert("example.xml", <a><b>bbb</b></a>)

(: transaction 2 :)
xdmp:node-replace(doc("example.xml")/a/b, <c>ccc</c>)

(: query the result :)
doc("example.xml")
= <a><c>ccc</c></a>
```

Saving a Document



- You can save a document to disk after it's been changed in memory
 - `xdmp:save($path as xs:string, $node as node())`
- `xdmp:save()` supports:
 - Sending a document to another party
 - XML, text, or binary content

```
xdmp:document-insert("example.xml", <a/>)|  
(: some time later... :)|  
xdmp:save("/tmp/foo.xml", doc("example.xml"))|
```

Adding to a Collection



- To add a document to a collection
 - `xdmp:document-load()` **collections** option
 - `xdmp:document-add-collections($uri as xs:string, $collections as xs:string*)`
 - `xdmp:document-get-collections($uri)`
 - `xdmp:document-remove-collections($uri, $collections)`

```
(: set collection at load time :)
xdmp:document-load("data.xml",
<options xmlns="xdmp:document-load">
  <collections>
    <collection>coll</collection>
  </collections></options>)

(: or add the collection later :)
xdmp:document-add-collections("ex.xml", "coll")
```

Deleting a Collection



- To remove a document from a collection
 - `xdmp:document-remove-collections($uri as xs:string, $collections as xs:string*)`

```
xdmp:document-remove-collections("ex.xml", "coll")
```

- To remove from the database all documents in a collection:
 - `xdmp:collection-delete($uri as xs:string)`

```
xdmp:collection-delete("coll")
```

Document Properties



- Every document can have unlimited properties
- Especially useful for binary, text docs
 - `xdmp:document-set-properties($uri as xs:string, $props as element()*)`
 - `xdmp:document-add-properties($uri as xs:string, $props as element()*)`
 - `xdmp:document-remove-properties($uri as xs:string, $props as xs:QName*)`
- Access via `xdmp` functions, XPath “property” axis

Document Properties: Example



```
xdmp:document-add-properties(  
    "test.xml",  
    (<status>good</status>, <deleted>0</deleted>)|  
)|  
  
doc('test.xml')/property::node(..  
  
<prop:properties  
    xmlns:prop="http://marklogic.com/xdmp/property">  
    <status>good</status>  
    <deleted>0</deleted>  
    <prop:last-modified>  
        2005-06-16T08:04:03-07:00  
    </prop:last-modified>  
</prop:properties>
```

Document Properties



■ Querying document properties

□ //MedlineCitation

[property::last-modified gt current-dateTime()]

□ xdmp:directory-properties(\$uri as xs:string)
as element(prop:properties)*

□ xdmp:collection-properties(\$uri as xs:string*)
as element(prop:properties)*

```
for $p in xdmp:collection-properties("nyt1970")  
  /prop:properties[ content-length gt 1024 ]  
return element result {  
  attribute uri { base-uri($p) },  
  $p/content-length  
}
```

Collections, Directories, Documents, and Properties



- Properties mark documents with XML elements
 - Good for binary, or big fragmented docs
- Collections organize documents into sets
 - Fast retrieval, fast deletes
- Directories organize documents in trees
 - Fast retrieval, fast deletes
- Document URIs as a list of references
 - `cts:document-query()` - with uri lexicon

Add and remove permissions

- Adding permissions on the document or directory

```
define variable $uri as xs:string external  
  
xdmp:document-add-permissions(  
    $uri, xdmpp:permission("reader-role", "read")  
)↑
```

- Removing or setting new permissions on document or directory

```
xdmp:document-set-permissions() and  
xdmp:document-remove-permissions()
```

Admin Extensions



Mark Logic Builtins



- Mark Logic provides a wealth of custom functions
 - `xmp:hash32("string") = "206765238"`
 - `xmp:hash64("string") = "9637883543050871606"`
 - `xmp:quote(<a/>) = "<a/>"`
 - `xmp:describe(current-dateTime()) = xs:dateTime("2003-07-01T18:31:36-07:00")`
 - `xmp:random(100) = 47`
 - `xmp:eval("xmp:random(100)") = 5`
 - `xmp:estimate(/item[state='pending']) = 149823`
 - `xmp:strftime("%a, %d %b %Y %H:%M:%S",current-dateTime()) = Tue, 08 Apr 2003 17:21:37`

HTTP Server Builtins

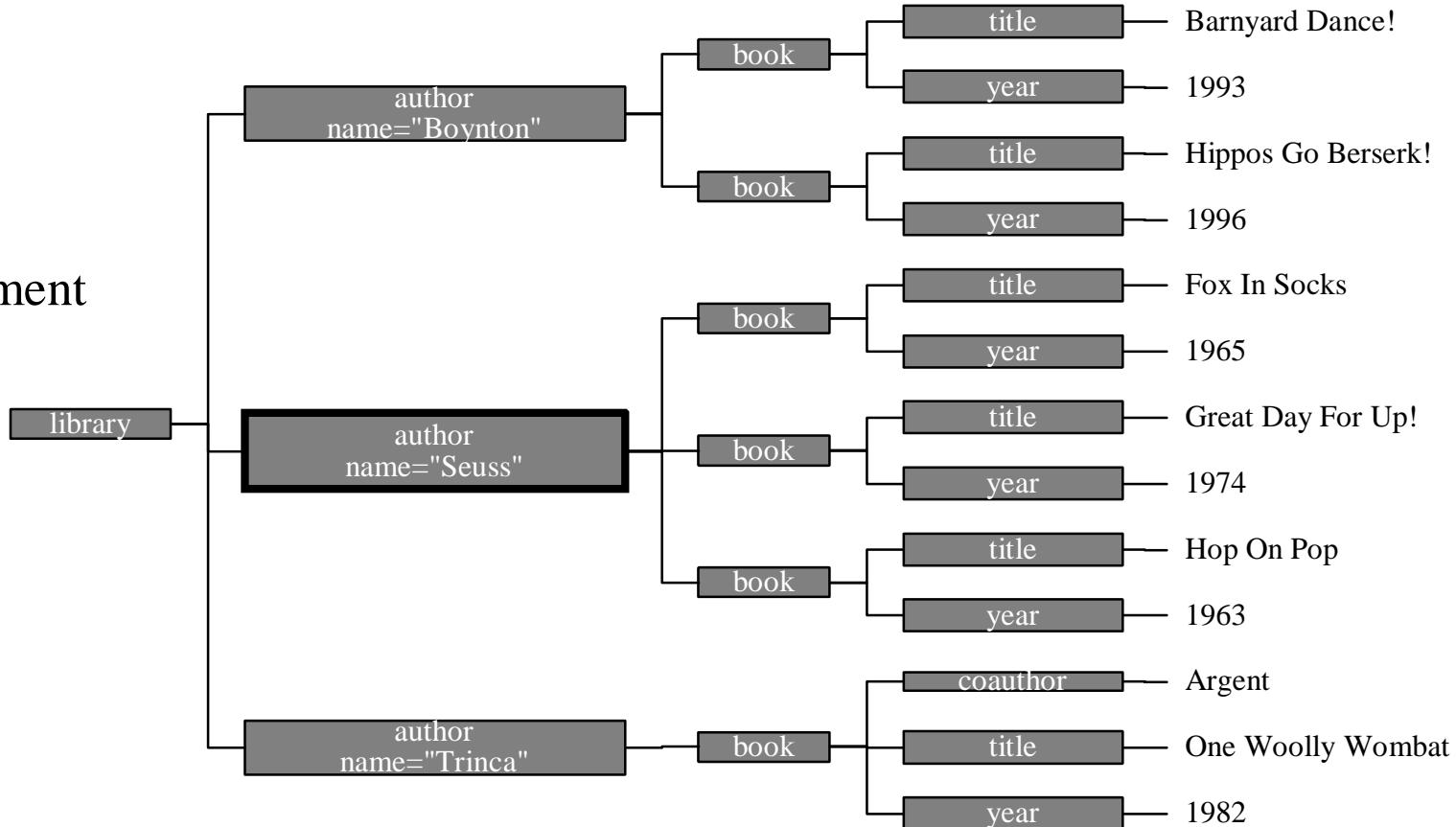


- The server provides custom HTTP-related functions to support .xqy files executing in the HTTP front-end
 - `xdmp:get-request-field("name", "default")`
 - `xdmp:get-request-field-names()`
 - `xdmp:get-request-header("name", "default")`
 - `xdmp:get-request-header-names()`
 - `xdmp:redirect-response("http://x.com/")`
 - `xdmp:set-response-content-type("text/xml")`
 - `xdmp:add-response-header("name", "value")`
 - `xdmp:url-encode("Why not?") = "Why+not%3f"`

Axis: property (MarkLogic only)

M A R K
LOGIC

Document



Document Properties

last-modified=
xs:dateTime("2005-01-01T12:34:56-08")

Prolog Variables and xdmp:set()

M A R K
LOGIC

- Use xdmp:set() to modify the value of a variable

```
declare variable $longest as xs:integer := 0;

declare function max-length($seq as xs:string*)
as xs:integer
{
  for $x in $seq
  return xdmp:set($longest,
                    max(( $longest, string-length($x) )),  

                    $longest
      );
}

max-length( ( "a" , "ab" ) )
```

xdmp:log()

- What's the value of \$n in my expression?
- XQuery doesn't have printf
...but we can log messages

```
let $list := /MedlineCitationSet/MedlineCitation
  [ cts:contains(Article/Abstract, $word) ]
let $d :=
  xdmp:log(text { $word, count($list), 'items' })
return $list
```

Mixing XQuery Versions



Extended Backus-Naur Form (EBNF)



```
Module           ::= Version? (MainModule | LibraryModule)
Version         ::=
                    "xquery" "version" StringLiteral Separator
MainModule      ::= Prolog QueryBody
LibraryModule   ::= ModuleDecl Prolog
ModuleDecl      ::= "module" StringLiteral
Prolog          ::= ((DefaultNamespaceDecl
                        |
                        | Setter
                        | NamespaceDecl
                        | Import) Separator)*
                        ((VarDecl
                        |
                        | FunctionDecl
                        | OptionDecl) Separator)*
Separator        ::= ";" 
QueryBody        ::= Expr
...etc
```

The Version



- An XQuery module may declare its version number at the beginning of the prolog
 - BNF tells you it comes after the "module"
 - No version statement?
 - HTTPServer or XDBCServer setting
 - Default is "1.0-ml"

```
xquery version "0.9-ml"
module namespace tr = "training-functions"

(: etc :)
```

- Processors throw an error when encountering an unknown version

MarkLogic Server v3.2 and v4.0



MarkLogic Server 4.0 includes a compatibility dialect to run 3.2 code without any changes using "0.9-ml"

3.2 (xquery version "0.9-ml")

- Semicolons (;) were limited to specific purposes like ending transaction
- Prolog declarations used 'define'
- Library module declaration use:
module "module-namespace"

4.0(xquery version "1.0-ml"/ "1.0")

- Semi-colons (;) are now required at the end of each prolog (like function, xquery version, global variable declarations etc) declaration
- Prolog declarations use 'declare'
- Library module declaration use:
module namespace ns = "module-namespace";

MarkLogic Server v3.2 and v4.0



3.2 (xquery version "0.9-ml")

4.0(xquery version "1.0-ml"/
"1.0")

- Within a module the functions could be called without any namespace
- The name space xdt was used in xquery version 0.9 and included/defined explicitly in the MarkLogic Server
- Library module declarations now require the namespace keyword and a prefix for the namespace, for example:
`module namespace my = "my-namespace";`
- The namespace used for *durations*, *atomic types* have changed to *xs* from *xdt*. For example, if an earlier code that uses *xdt:dayTimeDuration*, should be changed that to *xs:dayTimeDuration*.

MarkLogic Server v3.2 and v4.0



Variable declaration syntax is slightly different, and now uses the ':=' syntax

e.g.

```
declare variable $is-it-january as xs:string
              := if (
                           fn:month-from-
date(fn:current-
                           date()) eq 1 )
                     then "it is January"
                     else "it is not January" ;
```

If this variable were defined in a library module named mylib.xqy, it would be accessed as

```
xquery version "1.0-ml";
import module namespace mylib="my.library.uri" at
"mylib.xqy";

$mylib:is-it-january
```

Navigating the Online Documentation



Download
the server

Documentation

Join developer mailing list



Mark Logic Developer Network

Welcome to the Mark Logic Developer Network where MarkLogic Server developers can find and share solutions, participate in discussions, and help build a community around the technology. As a Mark Logic developer, whether new or experienced, you're already ahead of the curve. Let us know what we can do to help you put Mark Logic to work for you.

For more on what the developer network is about, see the welcome [Letter From The Editor](#). You may also want to dive right in by reading our [Informal Technical Overview](#) that will introduce you to XQuery, MarkLogic Server and the site.

Tell us what you'd like to see here. Join the [Discussions](#) and let your voice be heard. Welcome to the community.

News from Mark Logic

Mark Logic User Group in New York!

[November 2, 2009] Come join us for the inaugural meeting of the Mark Logic User Group in New York City. The meeting will be held at the

Download
Now!

Search...

Download MarkLogic Server

What is MarkLogic Server?

Technical Overview

Online Tutorials

Reading Links Page

Mark Logic Bloggers

Dave Kellogg

Mark Logic TechBlog

Matt Turner

Micah Dubinko

Jason Hunter

MarkMail

Norm Walsh

Philip Easson

“docs” tab



[Mark Logic XQuery Function Reference \[html\]](#)

API documentation for the Mark Logic built-in and module extensions to the XQuery standard functions. It also contains API reference information for the W3C standard functions implemented in MarkLogic Server.

[Administrator's Guide \[pdf\]](#)

Provides procedures for administrative tasks such as creating servers, creating databases, backing up databases, creating users, setting up your security policy, and so on.

[Application Developer's Guide \[pdf\]](#)

Provides procedures, methodologies, and conceptual information about general application development tasks in MarkLogic Server.

[Search Developer's Guide \[pdf\]](#)

Provides procedures, methodologies, and conceptual information about developing search-related applications in MarkLogic Server. Includes information about the Search API and the built-in search features of MarkLogic Server.

[Application Builder Developer's Guide \[pdf\]](#)

Provides step-by step information on using the Application Builder to build a search application, and also provides information on customizing applications built with the Application Builder.

[Query Performance and Tuning \[pdf\]](#)

Provides performance-related information that is useful to application developers and administrators.

[Scalability, Availability, and Forest-Level Failover \[pdf\]](#)

Provides information on large-scale system architecture, clustering, and details on setting up forest-level failover.

[Content Processing Framework \[pdf\]](#)

Provides an introduction to the Content Processing Framework and procedures for installing the default content processing framework.

[Mark Logic XQuery Function Reference](#)

[Administrator's Guide](#)

[Application Developer's Guide](#)

[Search Developer's Guide](#)

[Application Builder Developer's Guide](#)

[Query Performance and Tuning](#)

[Scalability, Availability, and Forest-Level Failover](#)

[Content Processing Framework](#)

[Understanding and Using Security](#)

[Getting Started With MarkLogic Server](#)

[XQuery Reference Guide](#)

[XCC Javadoc](#)

[.NET XCC API Documentation](#)

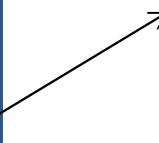
[XCC Developer's Guide](#)

[Installation Guide](#)

[4.1 Release Notes](#)

[ZIP file of all 4.1 Documentation](#)

Download all
the
documentation
on your
machine



“workshop” tab

Active Workshop Projects

Mark Logic CQ

CQ is a web-based XQuery tool, which allow developers to write ad hoc queries without managing .xqy files. Leveraging the web browser's built-in XML tree view, html view, and JavaScript engine, CQ is almost an IDE. You can view the results of your queries as XHTML, XML, or plain text. You can also profile the performance of your queries. You can work on multiple queries at once, browse your query history, change content sources, and save your queries to multiple, persistent sessions.

CQ is designed for use with MarkLogic Server. It must be installed within the root of an HTTP Application Server. CQ is tested with FireFox 3 and Internet Explorer 7. Some features may not work on other browsers.

The latest release requires MarkLogic Server 4.0-1 or later. Older releases are available for earlier versions of MarkLogic Server.

Current Version: 4.0-3

Subversion URL: <http://xqzone.marklogic.com/svn/cq/trunk>

Distribution URL: <http://xqzone.marklogic.com/svn/cq/releases>

MarkLogic Toolkit for PowerPoint®

The MarkLogic Toolkit for PowerPoint® allows you to quickly build information applications with MarkLogic Server that extend the functionality of Microsoft PowerPoint and leverage Office Open XML (OOXML), the native document format of PowerPoint 2007.

With the Toolkit for PowerPoint you can enable users to

- Search across your organization's PowerPoint files for specific slides and presentations
- Search across your organization's Word and Excel files for content to embed in slides
- Create new PowerPoint presentations from existing documents

MLJAM

MLJAM is an open source XQuery and Java library (written by Jason Hunter of Mark Logic and Ryan Grimm of O'Reilly Media) that enables the evaluation of Java code from the MarkLogic Server environment. MLJAM gives XQuery programs access to the vast libraries and extensive capabilities of Java, without any difficult glue coding. Example uses for MLJAM:

- Extracting image metadata
- Resizing and reformatting an image
- Running an XSLT transformation
- Generating a PDF from XSL-FO
- Calculating an MD5 hash
- Interfacing into a user authentication system
- Accessing a credit card purchasing system
- Connecting to a secure HTTPS web site
- Re-encoding content as UTF-8

MLJAM is distributed under the Apache License, Version 2.0.

RecordLoader

RecordLoader is a Java-based tool, designed to load any number of arbitrary-sized input documents into a Mark Logic database. To get started with RecordLoader, try the [tutorial](#). Capabilities include:

- load from input files or standard input
- input files may be xml files, or zip archives of xml files
- autosplit into new documents, using element name
- add read-permissions for one or more roles
- add collections
- generate document uris from a prefix, suffix, and element or attribute value
- up to one thread per input file
- optionally specify document forest placement
- optionally transform content using an XQuery module
- plug in subclasses to customize functionality

The current release requires Mark Logic Server 3.2 or later.

Current Version: see distribution URL

Subversion URL: <http://xqzone.marklogic.com/svn/recordloader/trunk>

Distribution URL: <http://xqzone.marklogic.com/svn/recordloader/releases>

Corb

Corb is a Java-based tool for content reprocessing in bulk. To get started with Corb, see the [README](#) file.

Corb requires Mark Logic Server 3.2-1 or later.

Current Version: 1.0.0

Subversion URL: <http://xqzone.marklogic.com/svn/corb/trunk>

Distribution URL: <http://xqzone.marklogic.com/svn/corb/releases>

Subscribe to general mailing list

M A R K
LOGIC

developer:network

home download docs how-to discussion workshop reading user groups help

Subscribe to the "general" Mailing List

Please enter your email address in the field below to subscribe to the general mailing list.

If you prefer to receive a daily digest of the messages posted to this list, please visit the generic [Mailman Signup Page](#) (Due to technical constraints, we haven't "skinned" all the Mailman pages yet, so it looks a little different).

Email addresses we collect for this mailing list are not disclosed to anyone else. Please review our [Privacy Policy](#) for details.

[Download MarkLogic Server](#)

[Privacy Policy](#)
x-query.com

Brought to you by
Mark Logic Corporation

Copyright © 2004-2009 Mark Logic Corporation.
All rights reserved.
[About Us](#) | [Contact Us](#) | [Privacy Policy](#)

Powered by MarkLogic Server 4.1-3
Standard Edition

MARK Mail

Search 2 marklogic lists for: search

[Home](#) Want your own MarkMail? Tell us about it. [Sign In or Sign Up](#)

Messages per Month (Swipe to refine by date)

Sort by Relevance Actions... 1 to 10 of about 1106

[\[MarkLogic Dev General\] Fwd: help on updating node](#)
here is the flow after getting the result from `search` i call another query named `update-write.xqy` , whi...
Oct 12, 2009 - Rahul Saluja - com.marklogic.developer.general -

[\[MarkLogic Dev General\] Problem with update node](#)
here is the flow after getting the result from `search` i call another query named `update-write.xqy` , whi...
Oct 12, 2009 - Rahul Saluja - com.marklogic.developer.general -

[\[MarkLogic Dev General\] Fwd: Problem with update node](#)
here is the flow after getting the result from `search` i call another query named `update-write.xqy` , whi...
Oct 12, 2009 - Rahul Saluja - com.marklogic.developer.general -

[\[MarkLogic Dev General\] help on updating node](#)
here is the flow after getting the result from `search` i call another query named `update-write.xqy` , whi...
Oct 12, 2009 - Rahul Saluja - com.marklogic.developer.general -

What List? [View more](#)

...ogic.developer.general	1,106
---------------------------	-------

Who Sent It?

Danny Sokolsky	104
Geert Josten	75
Michael Blakeley	75
Mattio Valentino	35

Any Attachments? [View more](#)

xqy	8
jpg	6
gif	5
txt	2

Type of Message?

development	1,106
-------------	-------

Searching Dev General List

M A R K
LOGIC

MarkMail

Search 2 marklogic lists for: search

Home Refine Search Want your own MarkMail? Tell us about it. Sign In or Sign Up (V)

Sort by Relevance Actions... 1 to 10 of about 1106

[\[MarkLogic Dev General\] Fwd: help on updating node](#)
here is the flow after getting the result from **search** i call another query named update-write.xqy , whi...
Oct 12, 2009 - Rahul Saluja - com.marklogic.developer.general -

[\[MarkLogic Dev General\] Problem with update node](#)
here is the flow after getting the result from **search** i call another query named update-write.xqy , whi...
Oct 12, 2009 - Rahul Saluja - com.marklogic.developer.general -

[\[MarkLogic Dev General\] Fwd: Problem with update node](#)
here is the flow after getting the result from **search** i call another query named update-write.xqy , whi...
Oct 12, 2009 - Rahul Saluja - com.marklogic.developer.general -

[\[MarkLogic Dev General\] help on updating node](#)
here is the flow after getting the result from **search** i call another query named update-write.xqy , whi...
Oct 12, 2009 - Rahul Saluja - com.marklogic.developer.general -

[RE: \[MarkLogic Dev General\] stemmed searches](#)
cts:search(doc(), cts:or-query((cts:word-query('search', 'exact'), cts:word-query('search', 'stemmed')))) We're using lib-search to generate complex queries on a lot of fields an...
Mar 30, 2009 - Whitby, Rob, CMG - com.marklogic.developer.general -

[RE: \[MarkLogic Dev General\] search:search @total changes](#)

RE: [MarkLogic Dev General] Problem with update node 4 messages in com.marklogic.developer.gen...
Problem w...

From	Sent On	Attachments
Rahul Saluja	Oct 12, 2009 10:48 am	.xqy, .xqy, .xqy
Rahul Saluja	Oct 12, 2009 12:12 pm	.xqy, .xqy, .xqy
Geert Josten	Oct 13, 2009 8:28 am	
Geert Josten	Oct 13, 2009 8:32 am	

Subject: RE: [MarkLogic Dev General] Problem with update node [Action](#)
From: Geert Josten (Geer...@daidalos.nl)
Date: Oct 13, 2009 8:28:16 am
List: com.marklogic.developer.general

Rahul,

I have played around a bit with your files. I recommend you start with testing the form for entering the update info. Just pressing the submit button already makes clear only two out of 7 input fields are being passed along as request parameters. I do not see obvious flaws, apart from the fact that you declare a xhtml namespace, but don't use it. And you have non-empty br elements, which is not allowed. I only see calculateUri and Status return as request parameters in the browser url. It almost looks like the fields in between are not seen as part of the same form. Fix this, and your node-uri call will most likely no longer throw errors as well..

Thank You

