Introductions

• Paul McBride

Paul joined the North American Primo Support Team in 2010. Prior to joining Ex Libris, he spent ten years doing distributed application development and support. Paul is an SME on Primo’s APIs.

• Troy Gnat

Troy joined the North American Tier 1 Support Team in 2013. Prior to joining Ex Libris, he spent four years working as a Linux Administrator and Support Specialist. Troy is experienced in Primo web, CSS, UI, Open Interfaces.
Objectives and Target Audience

• Brief Session Description: Integrate Primo into other software! Primo RESTful APIs can be used to bring the power of Primo to your applications, web portals, library site and more. Different Primo APIs and their syntax will be explained and discussed. Usage of the API Explorer to develop Primo REST APIs will be demonstrated as part of this presentation.

• Session Objective(s)
  • By the end of this sessions you will know, understand and/or be able to:
    • Understand the differences in Primo APIs
    • Get started with Primo REST APIs using the API Explorer
    • Perform a brief search using Primo REST APIs

• Session Target Audience:
  • New and Experienced Customers
  • Local and Direct SAAS Customers will have advantage over TotalCare

Agenda

1. Introduction
2. Developer Network
3. Primo API Overview
4. REST APIs
5. API Gateway
Why use APIs?

“People don't want to buy a quarter-inch drill. They want a quarter-inch hole.”

Theodore Levitt
Economist & Professor at Harvard Business School

Why use APIs?

- Build your own UI
- Integrate new workflows into Primo
  - Plug-ins
  - Send To
  - RTA
- Integrate Primo functionality into your site
New Portal -- Developer Network

• All content from El-Commons moved to “Developer Network”
• http://developers.exlibrisgroup.com/primo
**Developer Network - Sections**

The pleasure lies not in discovering truth, but in searching for it. (Leo Tolstoy, 1828-1910)

**API**
- Learn in-depth documentation of our APIs
- Deep Links
- WEB SERVICES

**Plugins**
- Primo has many hooks in its code allowing you to integrate your business logic into Primo's workflow. Read the in-depth documentation of Primo's plugins. Learn more...
- Front End Plugins
- Back Office Plugins
- Search Engine Plugins

**Code contributions / Developer's Blog**
- Share your experience and read other success stories at the Primo Developers Blog
- Latest entries:
  - 24 Jun 2014: How we're building APIs at Ex Libris
  - 02 Jul 2014: Primo 4 latest slider enhancements

---

**Developer Network – Code contributions / Developer’s Blog**

- **Last Time I Counted...** Over 70 contributions

**Feedback Button example**

Help the library improve its discovery search engine by answering this quick 5 questions survey!
Front End

- RTA – Real Time Availability
- Push To - social bookmark services (Delicious, RefWorks...)

Back Office

- File Splitters
- Enrichment

Search Engine

- Deep Search

Primo API Overview
1. Deep Links (Classic UI)
2. Web Services
   ✓ X-Services (non-standard REST)
   ✓ SOAP
   ✓ REST (stay tuned!)
Deep Links

**Input:**
A URL with parameters:
http://myprimo.hosted.exlibrisgroup.com/primo_library/libweb/action/dlSearch.do?tab=default_tab&vid=01UNIV_main&institution=01UNIV&query=any,contains,economics&search_scope=UNIV_ALMA

**Output:**
Your Primo site Classic UI

(Works best with Classic UI)

Deep Links in New UI

- Institution Wizard
- “New UI Enabled” is checked
- Deep Links Redirected to New UI

Replaced by Primo Search URL

WYSIWYG
Web Services - SOAP

*Supported but deprecated*

**Input:**
A Primo search request XML
Wrapped in SOAP envelope
Uses POST

**Output:**
A Primo search result XML

**Types of SOAP Web Services**

- Search
  - searchRequest
  - fullViewRequest
  - getItRequest

- eShelf
  - addFoldertoE-Shelf
  - addToEshelfRequest
  - eshelfStructureRequest
  - getEshelfRequest
  - removeFromEshelfRequest

- Tags
  - addTagRequest
  - getAllMyTagsRequest
  - getTagsForRecordRequest
  - getTagsRequest
  - removeTagRequest
  - removeUserTagsRequest

- Reviews
  - addReviewRequest
  - getAllMyReviewsRequest
  - getReviewsByRatingRequest
  - getReviewsForRecordRequest
  - getReviewsRequest
  - removeReviewRequest
Web Services – X-Service

*Supported but deprecated*

**Input:**
A URL with parameters:

http://primosb-pmtna.hosted.exlibrisgroup.com/PrimoWebServices/xservice/search/brief?institution=TRIAL_1_INST&indx=1&bulkSize=10&query=any,contains,science

**Output:**
A Primo search result **XML**

**JSON** is supported, add: &json=true

---

**Types of X-Services**

- Search
  - Brief Search
  - Full View
  - Browse Primo Records –SP4.1
  - Get Scopes Of View
  - GetIt

- eShelf
  - Add Folder to E-Shelf
  - Add to E-Shelf
  - Get E-Shelf
  - Get E-ShelfStructure
  - Remove Folder From E-Shelf
  - Remove from E-Shelf

- Tags
  - Add Tag
  - Get Tags
  - Get Tags for Record
  - getAllMyTags
  - Remove Tag
  - removeUserTags

- Reviews
  - Add Review
  - Get Reviews
  - getAllMyReviews
  - getReviewsByRating
  - getReviewsForRecord
  - Remove Review
X-Service – Understand the response

Brief search response structure:

Constructed from 3 parts:

```
<root:SEGMENTS>
  <root:JAGROOT>
    <root:RESULT>
      <root:QUERYTRANSFORMS>
      <root:FACETLIST ACURATE_COUNT
      <root:DOCSET IS_LOCAL="true" TOTAL
    </root:RESULT>
    <root:searchTokens>
    <root:JAGROOT>
  </root:SEGMENTS>
```

1. QUERYTRANSFORMS – Did You Mean
2. FACETLIST – Facets and counters
3. DOCSET – PNX records

WS and XS IP mapping table

- IP address restrictions for X-Services or SOAP Web Services
- Configuration is required for usage
- Open a Case to Ex Libris Support
Primo REST APIs

What is REST?

REST = REpresentational State Transfer

- architectural style and paradigm by Roy Fielding, developer of HTTP
- Using links to navigate and utilize an application to represent transferring from one state (page) to another state (page)

- Roy Fielding said of the process to standardize HTTP:
  - “That process honed my model down to a core set of principles, properties, and constraints that are now called REST”

https://developers.exlibrisgroup.com/blog/How-we-re-building-APIs-at-Ex-Libris
https://developers.exlibrisgroup.com/primo/apis/webservices/rest/linked_data
REST APIs – SaaS vs Local

- Hosted / SaaS Customers – use REST calls via API Gateway
- Local Customers – use REST calls directly to Primo server
  - Different from Xservices
  - API respond in JSON (not XML)

Pro-tip: To view JSON LD files in your Chrome browser, use “JSON Formatter” Chrome Extension.
REST API's & API Gateway

• Gateway Flow

Configuring the Gateway

• Getting Started with Primo REST APIs

1. Signup in Developers Network
2. Activate account - confirmation email
Configuring the Gateway

3. Add application
   - Select Platform = “Web Application”

4. Get API Key – MOST IMPORTANT STEP!
   - Edit Application (using [ ] ) to link to Production or Sandbox
   - Set read/write permission
Configuring the Gateway

5. Open Case to Ex Libris Support to connect your new Gateway applications to your relevant Primo environment

- Title – “Primo REST API registration”
- In the description please include the following information:
  1. Alma Institution code
  2. Primo Institution code
  3. Your Production Primo URL
  4. Your Sandbox Primo URL

Once Ex Libris support completes the registration process, you will be able to start developing using the API gateway.

PNX API Syntax

- SaaS customers:
cdd0d5ae3ac84b3f886ac6f443cfbdad

Note: “inst” parameter is required in local API calling - but “apikey” is used in SaaS syntax

- Local customers
  http://<primoserver>:<port>/primo_library/libweb/webservices/rest/v1/pnxs
  ?q=any,contains,book&inst=MY_INST
API Explorer

- Build and test API syntax structure - [https://developers.exlibrisgroup.com/alma/api-explorer](https://developers.exlibrisgroup.com/alma/api-explorer)

Eshelf API Methods

**Eshelf services/methods:**

- **GET** - Retrieve
- **DELETE** - Delete – Remove folder or item
- **POST** - Create – Add folder or item
- **PUT** - Update – Change note or folder name

**POST Method Example:**


{"folders": [{"name": "MyNonfiction"}]

{"basket-items": [{"pxnId": "ALMA00123245", "context": "L or PC"}]}
Eshelf API Syntax

- **SaaS customers:**
  https://api-na.hosted.exlibrisgroup.com/primo/v1/eshelf/johnt123?apikey=l7xx44f7c09f06ed47709aabe31fbf3bac57

  **https://api-na.hosted.exlibrisgroup.com/primo/v1/eshelf/johnt123/Basket/fishing?apikey=l7xx44f7c09f06ed47709aabe31fbf3bac57**

  **Note:** “inst” parameter is required in local API calling - but “apikey” is used in SaaS syntax

- **Local customers**

Analytics API

- https://developers.exlibrisgroup.com/blog/Working-with-Analytics-REST-APIs
- https://developers.exlibrisgroup.com/primo/apis/Analytics/Retrieve_Analytics_report

  - Returns a Primo Analytics (OBI) report in **XML** format only (not JSON)
  - Maximum 65,000 lines response
  - **GET** /primo/v1/analytics/reports
  - Uses the OBI Analytics URL-encoded path to your report
  - Sample: https://api-na.hosted.exlibrisgroup.com/primo/v1/analytics/reports?path=%2Fshared%2FPrimo+Lincoln+University+%28New+Zealand%29%2FReports%2FSearches+vs+clickthroughs&limit=50
API Governance Thresholds

- Maintain optimum performance
- Efficient and proper use of infrastructure resources
- Reduce the risks of malicious attacks

Two types (for Primo):

1. Daily API Request Threshold
   - (FTE x 5) REST APIs per day
   - Violation = error message for the day

2. Concurrent API Request Threshold
   - 10 REST APIs per institution per second
   - Violation requests are queued and may be slower

Since each request made to the API incurs a computational cost, it’s important to make economical use of API requests. If a common function of your application can be performed using fewer requests than it currently does, it should.
Analytics for REST APIs

• Sign in
• Dashboard > Analytics
• You can view statistics about your REST API usage
Useful Guides and Articles

- **Deep Links**
  - https://developers.exlibrisgroup.com/primo/apis/deeplinks

- **Xservices**
  - **Brief Search** https://developers.exlibrisgroup.com/primo/apis/webservices/xservices/search/briefsearch
  - **Deprecated** https://developers.exlibrisgroup.com/blog/Ex-Libris-API-Road-Map

- **SOAP**
  - **Brief Search** https://developers.exlibrisgroup.com/primo/apis/webservices/soap/search/briefsearch_soap
  - **Deprecated** https://developers.exlibrisgroup.com/blog/Ex-Libris-API-Road-Map

- **REST APIs**
  - **Concepts** - Part 1, Part 2, Part 3
    - https://developers.exlibrisgroup.com/blog/How-we-re-building-APIs-at-Ex-Libris
    - https://developers.exlibrisgroup.com/primo/apis/webservices/rest/linked_data
    - https://developers.exlibrisgroup.com/blog/Ex-Libris-API-Road-Map
  - **Getting Started/Registration** https://developers.exlibrisgroup.com/primo/apis/webservices/rest
  - **Thresholds** https://developers.exlibrisgroup.com/blog/reducing-the-number-of-API-calls
  - **Brief Search** https://developers.exlibrisgroup.com/primo/apis/webservices/rest/pnxs
  - **Analytics** - Part 1, Part 2
    - https://developers.exlibrisgroup.com/blog/Working-with-Analytics-REST-APIs
    - https://developers.exlibrisgroup.com/primo/apis/Analytics/Retrieve_Analytics_report
  - **Analytics of REST APIs**
    - https://developers.exlibrisgroup.com/dashboard/reports/api-reports

Useful Knowledge Articles

- **Eshelf APIs return “Invalid URI Syntax” errors when using API Explorer**
  
  [https://knowledge.exlibrisgroup.com/Primo/Knowledge_Articles/Eshelf_REST_APIs_return_Invalid_URI_Syntax_errors_when_using_API_Explorer](https://knowledge.exlibrisgroup.com/Primo/Knowledge_Articles/Eshelf_REST_APIs_return_Invalid_URI_Syntax_errors_when_using_API_Explorer)

- **Other helpful tools (Chrome Web Store)**
  - JSON Formatter
  - Advanced REST Client
  - XML Tree
Summary

1. Developer Network
2. Primo API overview
3. REST APIs
4. API Gateway
5. API Explorer

Q & A

Any Questions?
Session Survey Evaluation

Please use the following link, https://www.surveymonkey.com/r/techsem2017, to provide feedback on your sessions.

THANK YOU
Paul.McBride@exlibrisgroup.com
Troy.Gnat@exlibrisgroup.com