Edge APIs

- Overview
  - edge-common Framework
- Existing APIs
  - edge-rtac
    - Diagram
    - Demos/Presentations
  - edge-patron
    - Diagram
    - Demos/Presentations
  - edge-orders
    - Diagrams
    - Demos/Presentations
    - JIRA
  - edge-oai-pmh
    - Diagram
    - Demos/Presentations
  - edge-sip2
    - Diagram
  - edge-ncip
  - Reference Environments
    - folio-testing
    - folio-snapshot

Overview

The purpose of this page is to serve as a collection point for high level information about edge APIs.

edge-common Framework

The intent of edge-common is to simplify the implementation of edge APIs by providing much of the boilerplate code shared among these APIs.

Github: https://github.com/folio-org/edge-common

The README found at the link above contains general information that applies to many of the edge APIs, including the following topics:

- Dependency management
- API Keys: sources, format and generation
- Institutional users and secure storage of credentials
- Configuration

Existing APIs

A list of existing edge APIs and related information/links.

edge-rtac

Edge API to interface with FOLIO for 3rd party discovery services to determine holdings availability.

Github: https://github.com/folio-org/edge-rtac

API Documentation: https://dev.folio.org/reference/api/#edge-rtac

Diagram
Demos/Presentations

Sprint Review 36-39 @ ~1:07:10 (Demo)
https://drive.google.com/file/d/1xidmMs3anoB0SK1DYPOqMEp6CP2wpZIL/view

edge-patron

Edge API to interface with FOLIO for 3rd party discovery services to allow patrons to perform self-service actions (place holds, renewals, etc)

Github: https://github.com/folio-org/edge-patron

API Documentation: https://dev.folio.org/reference/api/#edge-patron

Diagram

Demos/Presentations

Sprint Review 40/41 @ 58:00 (Demo - Account Info)
https://www.youtube.com/watch?v=nhNdlyCMZVg

Sprint Review 42/43 @ 1:13:00 (Demo - Loans/Fees & Fines)
https://www.youtube.com/watch?v=xhHWmFKp2mY

edge-orders

The purpose of this edge API is to bridge the gap between 3rd party purchasing systems/vendors and FOLIO. More specifically, the initial implementation was built with EBSCO's GOBI (Global Online Bibliographic Interface) service in mind, though the aim was to make a general-purpose edge API that will work with other vendors as well.
Github: https://github.com/folio-org/edge-orders

API Documentation: https://dev.folio.org/reference/api/#edge-orders

Diagrams

GOBI

[Diagram showing GOBI integration with OKAPI and mod-gobi]

External System Integrations: Orders

GOBI

Harrassowitz

EBSCOnet

Enables
• Greater flexibility, e.g. schema and logic customizations
• Loose coupling of the vendor integrations, e.g. if one vendor needs a change, there’s little chance of it affecting the other vendor integrations

Requires
• Vendor modules that might have very similar functionality

Demos/Presentations

Sprint Review 42/43 @ ~1:14:40 (Overview)
https://www.youtube.com/watch?v=xhHWmFKp2mY
Vendor Integrations

Edge-orders is designed to support multiple vendors. Currently, adding a new vendor requires:

- Writing a vendor-specific module (see above) this is where mapping and translation happens between the vendor and FOLIO (mod-orders). Lookups of things like locations, material-types, organizations, etc. will also happen here.
- Minor changes in edge-orders to handle proper routing of requests to the proper vendor-specific module (e.g. mod-gobi, mod-ebsonet, etc.). There's essentially a decision tree that looks at the "type" query argument in the request and uses that to proxy the request to the appropriate API, e.g. POST /orders?type=GOBI proxies the request to POST /gobi/orders.
  - One way to make this easier would be to allow for these type API mappings to be defined via configuration. See EDGORDERS-TBD.

JIRA

- EDGORDERS-TBD
- UXPROD-2361 - Getting issue details... STATUS TBD

edge-oai-pmh

Edge API for Metadata Harvesting (OAI-PMH Version 2.0)

Github: https://github.com/folio-org/edge-oai-pmh

API Documentation: https://dev.folio.org/reference/api/#edge-oai-pmh

Diagram

Demos/Presentations

Sprint Review 50/51 @ 16:15
https://www.youtube.com/watch?v=wrXxqcCPz4

edge-sip2

Edge API for SIP2 (Standard Interchange Protocol Version 2.0) integration with a Self Check (SC) station. This edge module is unlike the other edge modules due to the fact that SIP2 uses TCP for message interchange. Once the SC is connected to edge-sip2, SIP2 messages are decoded and translated to existing FOLIO API messages. The FOLIO responses are encoded as SIP2 messages and delivered to the SC. Typically the SC will allow the patron to check out an item, check in an item, place a hold on an item, renew an existing loan and pay fees and fines they owe.

Github: https://github.com/folio-org/edge-sip2

API Documentation: https://dev.folio.org/reference/api/#edge-sip2

Diagram
edge-ncip

The purpose of this module is to expose mod-ncip to external applications/3rd party software. It simply passes requests through to the NCIP module (after the API key is authenticated). It does not contain any other functionality.

Github: https://github.com/folio-org/edge-ncip

Reference Environments

The APIs above are deployed to the following reference environments:

See https://dev.folio.org/guides/automation/ for additional details

folio-testing

Build from HEAD/master nightly.

Base URL: https://folio-testing.aws.indexdata.com:8000

Example Request (edge-orders):

```
curl 'https://folio-testing.aws.indexdata.com:8000/orders?type=GOBI' \
-H 'Authorization: apikey eyJzIjoiNXNlNGdnbXk1TiIsInQiOiJkaWt1IiwidSI6ImRpa3UifQ==' \
-H 'Content-Type: application/xml' \
-XPOST \
-d '<PurchaseOrder>
   <CustomerDetail>
   <BaseAccount>8910</BaseAccount>
   <SubAccount>891010</SubAccount>
   </CustomerDetail>
   <Order>
   <ListedPrintMonograph>
   <collection>
   <record>
   <leader>00000nam a2200000u 4500</leader>
   <controldfield tag="001">99974828471</controldfield>
   <controldfield tag="003">Nh0cYBP</controldfield>
   <controldfield tag="005">20180905153857.0</controldfield>
   <controldfield tag="008">180905t20112011xx ||||||||||| eng d</controldfield>
   <datafield tag="020" ind1=" " ind2=" ">
   <subfield code="a">97805475757282</subfield>
   <subfield code="c">14.95</subfield>
   </datafield>
   <datafield tag="035" ind1=" " ind2=" ">
   <subfield code="a">(OCoLC)717297695</subfield>
   </datafield>
   <datafield tag="100" ind1="1" ind2=" ">
   <subfield code="a">DICK, PHILIP K</subfield>
   </datafield>
   </record>
   </collection>
   </ListedPrintMonograph>
   </Order>
 </PurchaseOrder>
```
<record>
  <datafield tag="245" ind1="1" ind2="0">
    <subfield code="a">MAN IN THE HIGH CASTLE.</subfield>
  </datafield>
  <datafield tag="260" ind1=" " ind2=" ">
    <subfield code="a">BOSTON</subfield>
    <subfield code="b">MARINER BOOKS</subfield>
    <subfield code="c">2011</subfield>
  </datafield>
</record>
</collection>

OrderDetail
<FundCode>USHIST</FundCode>
<Location>KU/CC/D1/A</Location>
<Quantity>2</Quantity>
<YBPOrderKey>99974828471</YBPOrderKey>
<OrderPlaced>2018-09-05T15:38:55</OrderPlaced>
<Initials>Mark</Initials>
<ListPrice>
  <Amount>14.95</Amount>
  <Currency>USD</Currency>
</ListPrice>
<NetPrice>
  <Amount>13.16</Amount>
  <Currency>USD</Currency>
</NetPrice>
<Description>LocalData1</Description>
<Value>Book</Value>
<Description>LocalData2</Description>
<Value>Notify requester upon receipt</Value>
<Description>LocalData3</Description>
<Value>Anne Esterhazy</Value>
<Description>LocalData4</Description>
<Value>signed-edition, vip-order</Value>
</OrderDetail>
</ListedPrintMonograph>
</Order>
</PurchaseOrder>

folio-snapshot
Build from HEAD/master nightly.

Base URL: https://folio-snapshot.aws.indexdata.com:8000

Example Request (edge-orders):

curl 'https://folio-snapshot.aws.indexdata.com:8000/orders?type=GOBI' \
-H 'Authorization: apikey eyJzIjoiNXNlNGdnbXk1TiIsInQiOiJkaWt1IiwidSI6ImRpa3UifQ==' \
-H 'Content-Type: application/xml' \
-X POST \
-d '<PurchaseOrder>
  <CustomerDetail>
    <BaseAccount>8910</BaseAccount>
    <SubAccount>891010</SubAccount>
  </CustomerDetail>
  <Order>
    <ListedPrintMonograph>
      <collection>
        <record>
