FOLIO & VuFind® at Trinity College Cambridge

08 Sept 2022
Trinity College

- Founded in 1546
  - by King Henry VIII
- 34 Nobel Prizes
- ~700 undergraduate students
- ~300 graduate students
- ~180 Fellows (academics)
About the Library Collection

- ~300,000 volumes
  - ~1250 western medieval manuscripts
  - ~750 books printed before 1501
- Capell collection of Shakespeariana
- Books from library of Sir Isaac Newton
- Rothschild collection of 18th-century literature
- Kessler collection of livres d’artistes
- >70,000 books printed before 1820
- Also, an ever-expanding modern collection
Why FOLIO

• Driven by need to replace old LMS (Innovative’s Millennium); very dated OPAC
• Experience of being independent from central university library (Alma);
• Single vendor model vs new model
• FOLIO: future-proofed(?); community-based; transparent; scalable

FOLIO is an open-source library services platform that integrates print and electronic resource management. It includes core library management functionality, is built on a modern architecture to ensure continuous innovation, and supports the integration of applications from third-party providers. EBSCO provides implementation, hosting and support for FOLIO.
Implementation Experience

• 9 month period, very intense. Hardest thing: to explain our library, its intricacies, our workflows. Decisions, decisions, decisions.

• Staff awareness/perception: intuitive, but different from Millennium, some good some bad; modern

• Challenges of using such a 'young' LMS e.g. connecting via SIP to self-issue machines; exporting and importing. But great community support.

From the Implementation Consultant’s (IC) view:

• Trinity’s catalog had the best data quality our IC had ever come across

• Everyone involved seemed to know the in- and outs of the catalog very well

• Client knew **what** they wanted but at the same time were flexible about the **how**
Why VuFind

• Open source, like FOLIO
• Blacklight first(!). Similar functionality, but more polished appearance (Stanford, Cornell) vs VuFind examples. But, VuFind highly customisable.
• Modern vs special collections. Challenges: call numbers, geared for LC. Boundwiths.
• Browse-type searches vs Google-like single search box
• Cater to different users: students, academics, external researchers
• Possible interaction with University Library's discovery layer, iDiscover (Primo)
Feature Spotlight

• I’ll show you features that do work and made an impact
• All of these items should see improvements
• I welcome suggestions and questions

• Thanks in advance!
Mapping Rules
Mapping Rules

- marc_local.properties for the win
- My head is spinning whenever I even think about it
  - Without Ben, I would be lost here

- Very powerful logic is possible
  - See https://github.com/solrmarc/solrmarc/wiki

- Sometimes you’ll need to do a Java function
  - Not nearly as bad as it sounds
New Items
New Items

Subjects are configured in Folio.ini
New Items
New Items

• Solr Index vs ILS

• Folio doesn’t have an endpoint to show newly added items

• Query against /orders/order-lines
  - receiptDate + receiptStatus="Fully Received"

• Downside:
  - Shows “in process” items
Boundwith

Sammelbände
Boundwith

Boundwith is a term used by catalogers when referring to pamphlets & tracts published separately but subsequently bound together…
Boundwith: The Problem to be solved

• A Boundwith in Folio is currently represented as:
  – Separate Instance for each element
  – Each Instance has a holding & call number
  – Only the first Instance has an item (= status)

• The problem:
  – When the first Instance was checked out, the other Instances would still show as available
Boundwith

- holdingsTypeld = 7b94034e-ac0d-49c9-9417-0631a35d506b
- Current State

```
"callNumber": "Crewe 1.10[1]",
"administrativeNotes": [],
"notes": [
  {
    "note": "This Record is a Bound-with. It is bound with 3 instances. Below is a json structure allowing you to move this into the future Bound-with functionality in FOLIO\n\n\n\n\n"aec5b5d2-55c1-4b77-b127-eaf3bc8b3e1d",\n\n\n\n\n"4b17cecc-9b7a-86b9-85dc-3f2720a9aad",\n\n\n\n\n"4739324a-f6e1-4b9b-b004-ed1a6c84f1d"\n"},
  "staffOnly": true,
  "holdingsNoteTypeId": "e19eabab-a85c-4ae-f-a7b2-33bd9acef24e"
},
  {
    "note": "This Record is a Bound-with. It is bound with 3 other records. In order to locate the other records, make a search for the Class mark, but without brackets."
  },
  "staffOnly": false,
  "holdingsNoteTypeId": "e19eabab-a85c-4ae-f-a7b2-33bd9acef24e"
],
```
Boundwith

• Availability for “boundwith” can be challenging
  – Imagine Crewe 1.10[2] – it has a holding, but no item
  – This would lead to Crewe 1.10[2] showing as available, despite the fact that the bound with (1.10) was checked out

• What to do?
  – Currently we look at the note, identify the first instanceId and try again
  – Those notes were added during data migration and not necessarily for newly added items
Boundwith

• Future State:

"boundWithParts": [
    {
      "id": "3a7642f7-bf54-4fe6-b407-a2ee2ad94be1",
      "holdingsRecordId": "b9d0c18d-b94a-4109-9fad-4c9a66f408b0",
      "itemId": "1a10d783-cf87-4ae4-a2f5-4fa2e92542cd",
      "metadata": {
        "createdDate": "2022-07-13T14:07:51.898+00:00",
        "createdByUserId": "28e02507-c772-446a-9f79-93065e7232a5",
        "updatedDate": "2022-07-13T14:07:51.898+00:00",
        "updatedByUserId": "28e02507-c772-446a-9f79-93065e7232a5"
      }
    }
],

• By using Folio’s Boundwith feature, we will find it easier to handle, it does require a change though
Boundwith

1

**Excellent opuscule de Plutarque, de la tardive vengeance de Dieu**
by Plutarch.
Published: A Paris : Pour Iean Dallier libraire, demeurant sur le pont Sainct Michel, à l'enseigne de la Rose blanche, 1563.
**Shelfmark:** Crewe 1.10[2]
**Located:** Reserve (Wren) - Ask at Enquiry Desk

**Holdings**

**Description**

**Comments**

**Staff View**

Reserve (Wren) - Ask at Enquiry Desk

**Shelfmark:**

Crewe 1.10[3]

Not for loan: Available

Notes:
- This Record is a Bound-with. It is bound with 3 other records. In order to locate the other records, make a search for the Class mark, but without brackets.

**To Top**
Granular Status in Result List
Result List: Granular Status

• By default: Checked-out or Available
  − What about “in Process”, “Missing”, etc?
• Config.ini
  − show_full_status = true
  − The generated table was deemed undesirable
• A bit of a hack
  − Adapt ajax/status-full.phtml
Result List: Granular Status

1. **John Selden and the western political tradition**
   by Haivry, Ofir
   Published: Cambridge : Cambridge University Press, 2017.
   Subjects: “...Selden, John, 1584-1654....”
   Located: Reading Room
   Book In process
Special & Modern Collections
Special & Modern Collections

• Trinity College has:
  - Special Collections
    • Identify those truly rare books
  - Modern Collection
    • Books for everyday student use

• Goal:
  - Help filter results
  - Browse the collection
Collections: The Challenge

• Two very different approaches

• Special Collection
  – Created Java Function to extract the “Collection Part” from 952$e
  – If 952$h == “Special Collection” extract first section of call number
  – NQ.17.37 => match against specialCollection_subject_map.properties
Special Collection: Browse from Homepage
Special Collection: Facet in result set
Collections: The Challenge

• Modern Collection
  - Created Java Function to extract the “Collection Part” from 952\$e
  - If 952\$h == “Modern” extract subject part of call Number
  - LL 082 K 100 => match against three properties files
    • Modern0:
      • 082 = 0/[006-108] ENGLISH LITERATURE: 1200-/  
    • Modern1:
    • Modern2:
      • 082 = 2/[006-108] ENGLISH LITERATURE: 1200-/ [080-085] More than one period/[082] History and criticism of prose; fiction; the novel; narrative/
Modern Collection (Browse by Subject Number)
Modern Collection (Browse by Subject Number)
Modern Collection (Hierarchical Facet)
Call Number Browse
Browse Indexes (Call Numbers)

• Trinity uses many different Call Number schemas
  – Crewe 1.10[1], LL 082 K 100 & NQ.17.37

• Call Number Browse did not do a good job at sorting
  – Crewe 1.1, Crewe 1.10, Crewe 1.2, Crewe 1.21

• Small Java Function normalizes these for sorting
  – Crewe 001 001, Crewe 001 010, Crewe 001 002, Crewe 001 021
    • => Crewe 1.1, Crewe 1.2, Crewe 1.10, Crewe 1.21
Browse Indexes (Call Numbers)

• Downside
  – Current implementation is a bit of a hack
  – Needs to be refactored

• Other Browse Indexes can suffer from a similar problem
  – Series Title Browse would benefit from a Natural Sort order
“It has been a great pleasure to work with EBSCO on implementing VuFind for the Trinity library catalogue. Once the full range of capability of the system became evident, we were able to install several additional features beyond our original expectations. It is very rewarding to see our researchers submitting complex searches and making full use of the faceted search options. We are particularly grateful to the FOLIO and VuFind communities for making all of this possible.”

Dr Nicolas Bell
Librarian, Trinity College Library
Questions?