“Every book its reader”:
recommendations in Aleph

Matthew Phillips
Systems Librarian
University of Dundee
m.e.phillips@dundee.ac.uk
Every book its reader

“While well thought out shelf-arrangement is necessary, it is by no means sufficient to get for EVERY BOOK ITS READER. The catalogue can also be of immense help in this matter.”


What Ranganathan is emphasising in his Third Law of Library Science is that librarians need to be active in promoting stock, in helping the books to discover their readers.

No doubt today Ranganathan would be referring to the Integrated Library System rather than the catalogue.
How can we recommend books?

- reader’s subject of study; course; modules
- reader’s past borrowing history
- the e-shelf as a clue to reader interests
- readers’ search patterns collectively
- reader’s saved searches; alerts / SDI
- reader ratings of material
- usage of electronic material

This slide is really about “what do we know about the reader which might help us recommend stuff?”.

By “readers’ search patterns collectively” I am thinking that we can keep logs of what search terms users as a whole have used. It is then possible to recommend to users search terms to try which are similar to the search terms they have entered, but which yield broader or narrower result sets. A good example of this in practice is the University of Huddersfield OPAC. Try http://webcat.hud.ac.uk/ and search for something vague like “ethics” or “management”. At the top of the results the system will suggest refinements based on successful searches made by a number of other users.

Usage of electronic material is a problem, as we cannot easily identify the materials which have been used by the same individual. This makes it harder to provide recommendations on a personal level. Nevertheless, if done properly, such as via a gateway service like SFX, useful recommendations can be made (e.g. bX service).
What does Aleph already provide?

• SDI (selective dissemination of information)
  = search alerts
• But that’s about all
Problems with Aleph SDI

• Reporting of items which aren’t actually new
  – e.g. if process status set and then cleared
  – but it’s a lot better from Aleph 18 onwards

• Not reporting items which are new
  – e.g. if several copies are received at once

• Not reporting material without items (for example, e-books)

• Reporting new journal issues but not telling you which issue is new

• RSS feeds have no <pubDate> for new items: cannot sort in IE

• Alert e-mails are not XSL based, so inflexible
Aleph SDI technical background

When an item is added or updated, and its process status is clear, a Z324 is created if it doesn’t already exist. New items are recorded at each sub-library and globally.

<table>
<thead>
<tr>
<th>Z324_REC_KEY</th>
<th>BIB no. + Z324 sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z324_ENUMERATION_A to _D</td>
<td>Enumeration of highest numbered item (serial)</td>
</tr>
<tr>
<td>Z324_LOCATION</td>
<td>ALL / ADM / sub-library</td>
</tr>
<tr>
<td>Z324_DATE</td>
<td>Date item was new</td>
</tr>
<tr>
<td>Z324_HOUR</td>
<td>Hour item was new</td>
</tr>
</tbody>
</table>

The infrastructure for the Z324 which lies behind the alerts system is good, but its implementation is a little faulty. If we can get the Z324 working right, we can exploit the data in many ways.
Fixing the SDI problems

- Custom service to run just before p_sdi_01
- Deletes the Z324 for items
  - with very old arrival dates
  - with very old open dates
  - in closed collections undergoing retrospective cataloguing
- Creates Z324 records for new e-books
- Resulting system is much more usable.

This custom service is available on EL Commons.
New books pages and RSS

- Use Z324 to discover the new material
- Custom service to build pages for each subject of study
  - HTML for inclusion in OPAC
  - Atom feed (similar to RSS)
  - updated via the job_list several times a day
  - XML-based configuration allowing testing of MARC fields, item data and order records
  - uses SFX API to link to new e-journal issues

We have written a custom service which uses the Z324 information to provide new books lists for each subject.

This custom service is not yet available on EL Commons. Please contact me if you are interested.
Here is an example from our web site. (The URL shown is from our development web-site. Go to http://www.dundee.ac.uk/library/subjects/media.php to see the public version.)

The blue panel “Latest titles added to the catalogue” is based on the RSS feed we have generated for that subject. Links from the titles lead to the catalogue. The “More…” link leads to the page shown on the next slide.
This page is generated by the custom service and is stored on our Aleph server, using the same templates and layout as our OPAC.
For journals, the custom service interrogates SFX when it builds the page. If the SFX API identifies that we have full text available for a journal issue, a link is provided. Of course, depending on the target, this does not always lead to exactly the right issue.
For journals, the custom service interrogates SFX when it builds the page. If the SFX API identifies that we have full text available for a journal issue, a link is provided. Of course, this does not always lead to exactly the right issue, depending on the target.
This slide shows examples of the many configuration options available in our new book script, allowing testing of item data, orders and budgets, as well as the MARC record.
This slide shows examples of the many configuration options available in our new books script, allowing testing of item data, orders and budgets, as well as the MARC record.
We don’t use LC classification but the 050 field appears in most e-book records. Example shows testing for values within a range.

This slide shows examples of the many configuration options available in our new books script, allowing testing of item data, orders and budgets, as well as the MARC record.
We can use Perl regular expressions!

This slide shows examples of the many configuration options available in our new books script, allowing testing of item data, orders and budgets, as well as the MARC record.
New books at Edinburgh Napier University

Edinburgh Napier have saved searches which identify new material in different subjects. This page is then generated (possibly via the RSS feeds?) and enlivened with cover images licensed from Syndetics.
New books: techniques

• Complex search in OPAC saved as SDI
  – e.g. Edinburgh Napier: uses standard Aleph
• Complex script and versatile configuration
  – e.g. University of Dundee
• Automatic identification of class numbers based on past habits of students on the course
  – e.g. University of Huddersfield (Dave Pattern)

The various options for producing new books reports.

At Huddersfield, rather than asking subject librarians to identify the classification numbers appropriate to each course, Dave Pattern uses the borrowing behaviour of previous students to identify the class numbers relevant to each course automatically. You could try this in Aleph also. We have not done this yet because our set-up does not store enough information about the patron’s course. To do this historically you really need to have accurate patron type information corresponding to the precise course of study. Our patron types are much more vague.
Automatic patron interest profiles

- Analyse the user’s past borrowings
- Look at
  - subject headings (e.g. 600, 650)
  - classification numbers (e.g. 050, 080, 082)
  - authors (100, 700)
- Weight according to overall popularity of the heading, i.e., how specialised is this interest?
- Store interests in an Oracle table

Can we use the user’s previous borrowings to identify new material which is of interest automatically? Here are the steps we might follow.
Automatic patron interest profiles (2)

When a new book arrives:

• compare the headings on the record with the patron’s interests in the table;

• calculate a score;

• if it passes a threshold, notify the patron of the arrival of the new book.

Still at the very experimental stage!
This shows an RSS feed of new books based on interests identified from my personal borrowing history. Some of the suggestions are not of interest, because I tend to borrow random things when testing the system. However, I was genuinely interested to know about many of the new books listed, and intend to borrow some of them in the near future!

So the initial results of this experimental system are promising.
I was particularly interested in the book on library web site design!
Linking to related books

• Taking the user beyond the search results
• Other editions
• Works on similar topics
• Works sharing borrowers (people who borrowed this have also borrowed…)
• Helps to promote stock that users have not searched for
This is a real example from http://library.dundee.ac.uk/ which has been in production with this enhancement for two years.

Not yet loaded up on to EL Commons. Please ask if interested and I’ll see if I can find the time to document it!
<table>
<thead>
<tr>
<th>People who borrowed this have also borrowed:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Farrelly, Lorraine</td>
</tr>
<tr>
<td>Forty, Adrian</td>
</tr>
<tr>
<td>Forsch, Kevin</td>
</tr>
<tr>
<td>Laseau, Paul</td>
</tr>
<tr>
<td>Powell, Helen</td>
</tr>
<tr>
<td>Drpic, Ivo D.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Related works include:</th>
</tr>
</thead>
<tbody>
<tr>
<td>White, Edward</td>
</tr>
<tr>
<td>T.</td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
## Full View of Record

### Choose format: Standard format, Catalogue card and Citation

**Record 7 out of 15**

<table>
<thead>
<tr>
<th>Format</th>
<th>Text</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Holdings</th>
<th>Location</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Main Entry</th>
<th>Begg, David K. H.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Title</th>
<th>Economics / David Begg, Stanley Fischer, Rudiger Dornbusch</th>
</tr>
</thead>
</table>

|---------|---------|

|---------|-----------------------------|

<table>
<thead>
<tr>
<th>Physical note</th>
<th>Please ask at Duncan of Jordanstone issue desk for CD-ROM which accompanies this copy</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Subject</th>
<th>Economics</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Added entry</th>
<th>Vance, Stanley</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>ISBn</th>
<th>9780070139710</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Record number</th>
<th>000312290</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Holdings</th>
<th>Location</th>
</tr>
</thead>
</table>

### Other editions include:

- **Begg, David K. H.**
  - *Economics* - 7th ed., 2005

### People who borrowed this have also borrowed:

- **Upson, Richard G. (Richard George)**, 1928–
  - *Economics* - 10th ed., 2004

- **Johnson, Gerry**
  - *Explaining corporate strategy* - 7th ed., 2004

- **Graves, Michael K.**
  - *Practical business forecasting* - 2002

### Related works include:

- **Chalhoub, Marlo**
  - *Job search and the firm's wage offer: a model of exit options* - 1989

- **Taylor, Marcus H.**
  - *Exchange rate bands with price* - 1990
Lots of other editions shown. Also several other popular general economics texts identified via borrowings.
Linking to related books (2)

- JavaScript to fetch related books panel
- Uses pre-generated (cached) links
- If no cached links available, panel does not appear and BIB record is queued for link calculation
- If cached links are old (for any value of “old”) they are also queued, at lower priority, for recalculation.

Here’s how it works. Links have to be cached because of the processing effort required.
Linking to related books (3)

- Other editions generated using OCLC xISBN API (see http://www.worldcat.org/affiliate/webservices/xisbn/app.jsp)
- “People who borrowed this have also borrowed” calculated from z36h
- “Related works” found using headings and class numbers in common, e.g. LCSH, Dewey, LC, UDC.
10% of views come via a “related materials” link rather than a search

This enhancement is having a real impact: it is popular with users and people follow the links to explore our library.
The logs we gather enable us to detect how deep users are exploring. How many clicks do they move away from their initial search result? Sometimes up to thirty!
A logarithmic scale makes the expected exponential decline apparent.
The links are cached, so we can see how long it has taken for users to explore our catalogue and cause the links to be generated. The green line shows the links via classification and subject headings. When we started collecting statistics, two years ago, we had built up 50,000 links, so with up to 5 per book that’s about 10,000 books with these links. Two years later, around 100,000 books have onward links. That leaves 400,000 books where no-one has even looked at the full catalogue record in the last two years!
Possible developments

• Exploit the user’s e-shelf as well as borrowings to determine interests.
• Automatic suggestions for exploration beyond the reading list.
• Automatic suggestions for lecturers for new material to add to reading lists.

A reading list is a collection of books. We could see what subject headings and classification numbers are attached to the books, and find closely similar material to recommend further reading. And we can see what books have been borrowed by people who have borrowed stuff on this reading list.

By analysing a reading list we could also make suggestions for relevant new material which arrives in the library, so that lecturers can keep the reading lists fresh.
Possible developments (2)

“People who borrowed this…” at a personal level:

1. patron logs in to catalogue
2. system accesses entire (or recent) borrowing history of patron
3. finds other patrons with similar borrowings and sees what they borrowed next


• “Collaborative filtering”

Netflix is an American postal DVD library. The great thing about their recommendation system is that by recommending films in their back catalogue they encourage users to borrow more, and not just the high-demand latest films. This way they work their stock harder and make more money.

A clear application to libraries here: we can make our stock work harder too (providing our older books are not factually inaccurate, like older medical texts, for example).
Aggregation of usage data

- Problem of scale by comparison with commercial data-warehousing
- Sharing anonymised usage data across higher education community (e.g. bX)
- JISC MOSAIC project: http://www.sero.co.uk/jisc-mosaic.html
- For example, a search for “ethics” could rank the results according to your subject of study.

Only by sharing usage data across institutions will we have the critical mass for more advanced services. This could be done co-operatively, or through a system supplier. For example, bX is doing this, and the URM may have the scope for greater sharing of statistics.

In the UK JISC has sponsored a project, MOSAIC, to look at exploiting the data. One idea floated is that if you know that the user is studying nursing, say, then a search by that user for “ethics” could rank more highly the medical ethics books which have been used by other nursing students. And an MBA student would get business ethics ranked at the top.
Questions?

Ideas?
OPAC scripting enhancement

- IGeLU enhancement request no. 1764 – Server-side scripting support for OPAC
- Please vote!
Miscellaneous other slides in case required...

The following slides are not part of the presentation, but were included in case I needed them when answering questions.
This slide shows the standard Aleph alerts e-mails. Not very nice.
Here’s the e-mail we generate using a custom service.
It even includes an HTML multipart-alternative with even nicer formatting.
Linking to related books: problems

• xISBN being incomplete, and cataloguers cannot correct it.

• Borrowings don’t include e-books: need a way to record usage of these by person.

• “Related works” works well for e-books, but we don’t use LCSH for printed books (local subject heading scheme) and don’t assign local headings for imported e-book records.