Workshop: «swissbib for the short distance runner»

Günter Hipler, system architect, project swissbib
University Library, Basel
Outline

- **Part I**
  *Dusty theory : Overview «Architecture of the swissbib solution» (~ 30 minutes)*

- **Part II**
  «Hands on» for short distance runners : setup your own presentation component based on the swissbib infrastructure within maximum 40 minutes

- **Part III**
  *Open Discussion –
  swissbib infrastructure as your « working bench » ?!*
swissbib architecture: A layered system with open interfaces!

The four independent layers of the swissbib solution and how to use them:

- Content collection (Open Source)
- Data Hub (Commercial)
- Lucene based Search Server (Open Source)
- Presentation of swissbib result for Humans / Machines (Open Source)

OAI
Webdav
Filepush
...

Directory API
("hot folder")

Online SRU
Catcher / Pusher

HTTP

OAI / SRU
(offline or online)

SearchEngine
HTTP Interface

HTTP

SRU
/ SPARQL
/ REST

service
dependent APIs

REST
/xServer
/ NCIP
/ SQL
Dive deeper - content collection

the four independent layers of the swissbib solution and how to use them
Purposes:

- fetches content from a repository (initial / frequently)
- Pre-processing of content (valid ? / transformation of structure)
- store the latest version of every single native record in datastore
- use Directory API to exchange single native records from repositories with DataHub
- extendable via plugins
Dive deeper - Data Hub
Data processing (refinement, add additional value to incoming content):

- 1st step: Single native records are virtually brought together if duplicates
- 2nd step: Result of 1st step is clustered to look for similar records (FRBR principals)
- Build additional relations between records
- Automatic process is highly flexible and customizable
- Enrich available content with external information (e.g. from WordCat)

Why do we call it a «Data Hub»?

- The result of the processing is used internally by swissbib and provided to external services
- We connect a multitude of single content resources on a national and international level
Dive deeper - To the heart of Search
Purposes:

«Heart» of Search (based on Lucene 4.x):

- If you search something: This is the place where it happens to be
- Search Server on top of Lucene is SOLR – could be extended or replaced by ElasticSearch
- Because of it's open character:
  - Library experts are able to adjust search characteristics by their own.
  - Could be used for educational purposes (teach students principals of IR on real systems)

Pluggable Document Processing (SDP):

- Full text enrichment
- Already Enrichment with GND variants (VIAF and MACS soon)
- Tailored for every single Index
Dive deeper - Join the user and their devices!
Part II. Hands on - « swissbib for the short distance runner »

- Create your own user presentation component on your local machine within maximum 40 minutes by using the swissbib infrastructure!
- Well-chosen institutions should be part of defined regions (virtual view) - use of « libadmin » tool (http://admin.swissbib.ch/libadmintest)
- As an example how it's used by swissbib: http://jus.swissbib.ch
- only single target within the workshop
- choose your colour within the swissbib design
Prerequisites:

- Linux recommended (CentOS / RedHat / Ubuntu)
- MacOs : possible (used by colleague)
- Windows : possible but not recommended
- Overview of software to be installed:
  Apache >= 2.2, MySQL >= 5.1, some php extensions, git

- Installation cookbook for Ubuntu:
- Cookbook for Mac is coming / Windows is not used by swissbib
Steps to install a local swissbib presentation component (1):

0. detailed instructions:

I. General installation (not tailored to the institutional requirements or desires)

• Fetch the source code from Github
  git clone https://github.com/swissbib/sbvf2.git .
  (use the git branch feature/epflworkshop)
• Setup and configure a locale MySQL database
• General configuration of the Application
• Ready for the first start!

  -> within the workshop these steps are done automatically via script
Steps to install a local swissbib presentation component (2):

II. Meet your institutional requirements

a) define your local virtual view

- Use admin.swissbib.ch/libadmintest to define a view (a prepared view with code «epflworkshopview» is already available)
- Adjust your component configuration
- Fetch the defined institutions from admin.swissbib.ch/libadmintest into your component
- Clear the cache and restart the local component to test it

b) define a search restriction

- Because you want to search only for the institutions part of the view
- Restart the local component again to test it
Steps to install a local swissbib presentation component (3):

**III. Change the design** (or create a new theme to use «VuFind language»)

a) the branch feature/epflworkshop already contains two additional themes epflworkshopblue and epflworkshopred

b) configure it and use / play around with the themes
Some basic principles of the View concept in swissbib / VuFind:

- VuFind provides standard themes which are inherited from each other
- customized views are inherited from VuFind views (or you can build your own theme from the scratch)

![Diagram showing the inheritance of themes in VuFind]

- VF root theme
  - VF jquerymobile theme
  - VF bootstrap theme
  - VF blueprint theme
  - swissbib theme
    - swissbibmulti theme
    - swissbibsingle theme
    - epflworkshopblue theme
    - epflworkshopred theme
Summary: Available swissbib (and workshop) resources:

- www.swissbib.ch (productive presentation service)
  testvf.swissbib.ch (Beta: next version based on VuFind2)
- sru.swissbib.ch (productive SRU interface)
  srutest.swissbib.ch (Test: next SRU version)
- www.swissbib.org (project wiki)
- swissbib.blogspot.ch/ (project blog)
- www.twitter.com/swissbib (Twitter account)
- https://github.com/swissbib/ (Open Source swissbib source code)
- Installation cookbook (prerequisites) Ubuntu:
  http://www.swissbib.org/doc/vf/cb.pr.ubuntu.odt
- Detailed installation steps workshop EPFL
  http://www.swissbib.org/doc/vf/cb.inst.sbvf.odt
Part III. Discussion!

1) Your comments, ideas and desires about the swissbib infrastructure and how to use it?

2) Possibilities to get involved in the further swissbib development?

Proposals:

• develop your own service based on the swissbib infrastructure e.g.:
  -> «hands on» in this workshop
  -> mobile swissbib from Lionel Walter
• help us to create better and more documentation
• speak with your «neighbour» about the pros and cons of swissbib
• make it known to your customers
• other fancy ideas? - it's up to you!
Thanks for your participation and thanks to the organizers of the «Free software in libraries» day!

National project swissbib
Hannes Hug, project lead
hannes.hug@unibas.ch

Tobias Viegener, project coordinator
tobias.viegener@unibas.ch

Günter Hipler, system architect
guenter.hipler@unibas.ch

swissbib Basel / Bern
Oliver Schihin
oliver.schihin@unibas.ch