



Improving FOLIO Architecture

Julian Ladisch

Verbundzentrale des GBV (VZG) in Göttingen, Germany

Martina Tumulla

North Rhine-Westphalian Library Service Centre (hbz) in
Cologne, Germany

European Library Automation Group (ELAG) Conference

Berlin, May 8th, 2019



FOLIO

Overview and Project structure





- Aim
 - Development of an open source Library Service Platform (LSP)
 - Open, sustainable, innovative, flexible, expandable
 - Software for librarians to manage daily work
- Target Group
 - Academic and research libraries

FOLIO



- FOLIO is a product = software
- FOLIO is a community
- Development since 2016
 - Founded as open source project by the stakeholders EBSCO, Index Data and the OLE community (Open Library Environment)
 - Designed and developed cooperatively



Stakeholder – EBSCO

- Funding of contracted developer teams, e.g. 25 FTEs EPAM
- Bringing in own human resources: product management, product owners, developers, UX/UI designers
- Financial support of OLF infrastructure
- Funding of expert reports, e.g. technical evaluation (OTS-report), security audit (planned)



Stakeholder – Index Data

- Bringing in own developer teams, product owners and UX/UI designers
- Responsible for the basic technical architecture (Okapi)
- In addition: developers under contract with EBSCO



Stakeholder – OLE Community

- Funding by membership fees and [Andrew W. Mellon Foundation](#)
- Funding of developers and OLE staff
- Bringing in own human resources:
Product owners, developers, functional experts, project management
- Financial support of OLF infrastructure
- Support of OLE partner projects, e.g. ERM apps



OLE Community

- OLE Board
- OLE Steering Committee
- OLE Managing Director
- OLE Director of Strategies
- OLE Project Manager



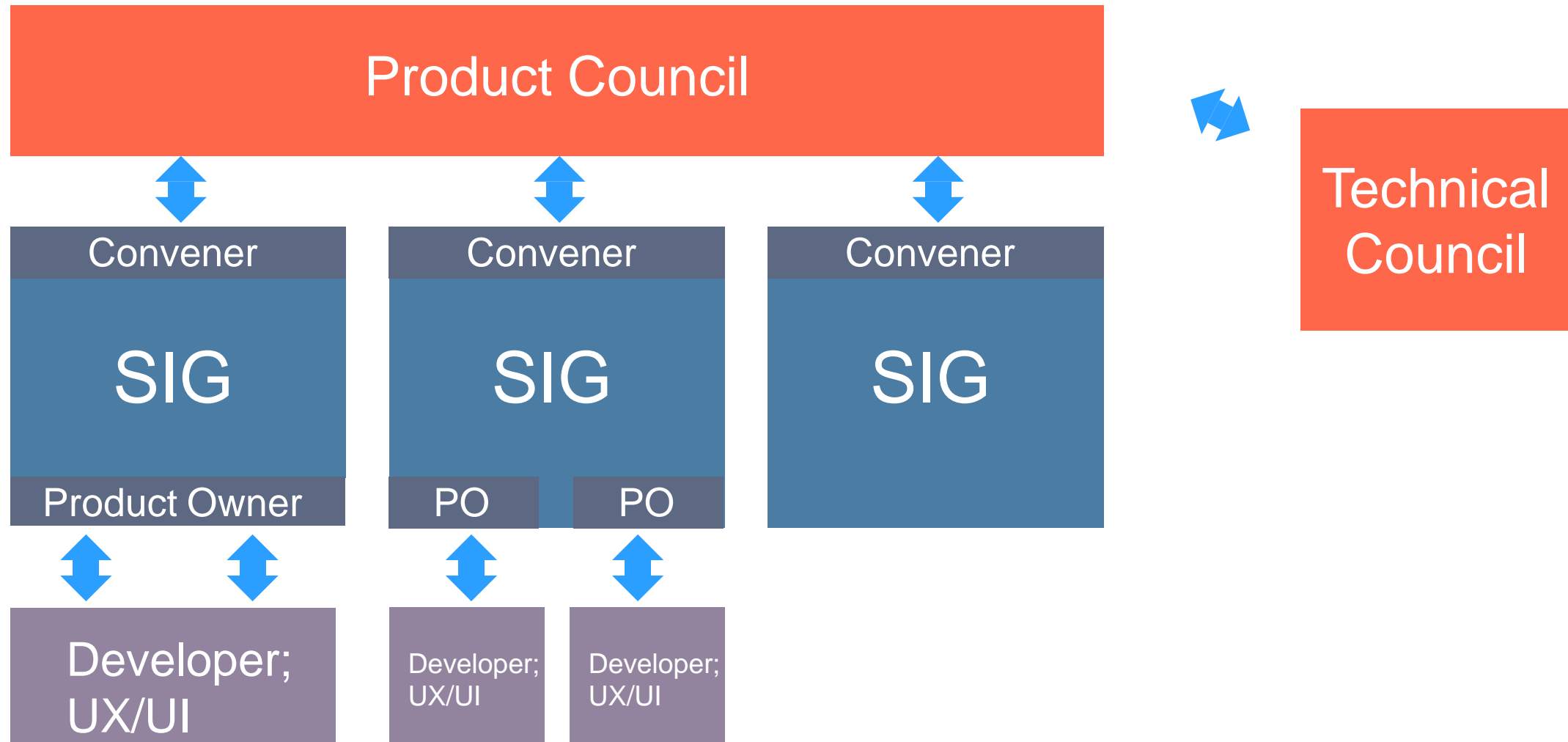


Open Library Foundation (OLF)

- New administrative home since 2016
 - Provides infrastructure (Confluence, Jira, Slack, ...) and secures open source code (GitHub) for projects in higher education
 - Projects: FOLIO, OLE, GOKb, ReShare ...



FOLIO – Committees





FOLIO – Special Interest Groups

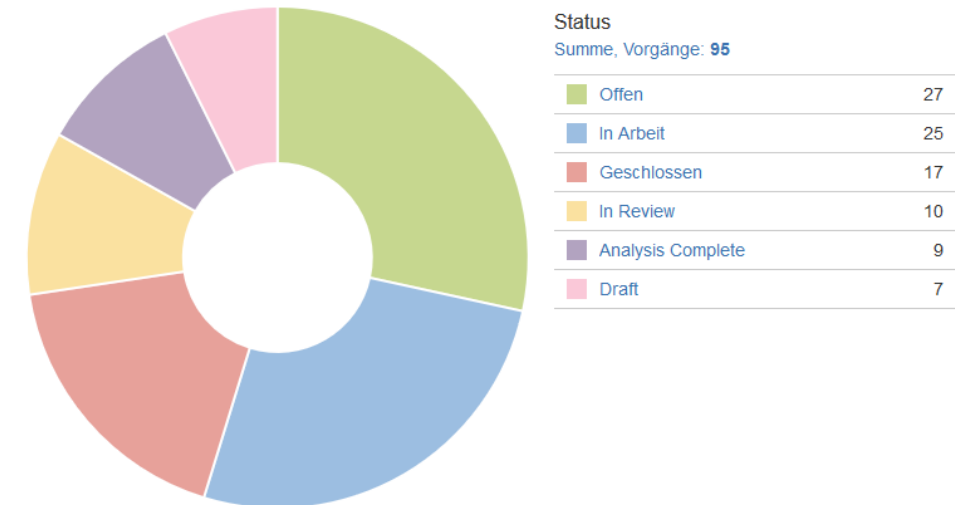
SIGs und subgroups

- | | |
|---|--|
| <ul style="list-style-type: none">• Metadata management<ul style="list-style-type: none">• Data import / export• MARCcat• Resource access<ul style="list-style-type: none">• Loans, reserves, requests• Printing slips, Patron notices, calendar• Off-site integration• Resource management<ul style="list-style-type: none">• Acquisitions small group• App interaction group• ERM subgroup | <ul style="list-style-type: none">• User management• Internationalization• Consortia• Reporting<ul style="list-style-type: none">• Reporting prototype subgroup• Privacy• Accessibility• System operations and management<ul style="list-style-type: none">• Data migration subgroup |
|---|--|

FOLIO Releases – Q1 Bellis, April 2019



- Enhancement of basic functionalities
 - Metadata management
 - Circulation
 - Import, note fields, document storage
 - Acquisition: ordering, receiving/check-in
 - ERM: usage statistics (eUsage app)
 - ERM: license-, agreement- and package management



Search & Filter

Search

^ Status

☐ Include inactive users

^ Patron group

☐ faculty

☐ graduate

☐ staff

☐ undergrad

Users

48 Records found

New

Active	Name↑	Barcode	Patron group	User
<div><div></div><div>Active</div></div>	Bailey, Khalid	676192847198080	faculty	n. ^
<div><div></div><div>Active</div></div>	Batz, Maddison Margie	144435285541195	staff	w
<div><div></div><div>Active</div></div>	Becker, Xavier Randal	447900165568253	graduate	je
<div><div></div><div>Active</div></div>	Bernhard, Sandra Name	246508223171616	undergrad	m
<div><div></div><div>Active</div></div>	Bradtke, Jonathan Randi	78243336520693	graduate	ai
<div><div></div><div>Active</div></div>	Champlin, Dagmar Emerald	731434363629411	staff	h
<div><div></div><div>Active</div></div>	Christiansen, Tomas	827217155887407	staff	pi
<div><div></div><div>Active</div></div>	Collier, Markus Paul	588195913950968	graduate	lc
<div><div></div><div>Active</div></div>	Connelly, Audie	103113883551642	undergrad	ai
<div><div></div><div>Active</div></div>	Cummings, Chester Gregorio	325861898332043	undergrad	sa
<div><div></div><div>Active</div></div>	Dooley, Newton Scarlett	403708804414844	faculty	la
<div><div></div><div>Active</div></div>	Ebert, Shaylee	180816517350763	staff	b
<div><div></div><div>Active</div></div>	Feeney, Margret Tavares	414876437022105	staff	te
<div><div></div><div>Active</div></div>	Feil, Aubrey Tia	262740097623773	staff	tf
<div><div></div><div>Active</div></div>	Ferry, Bernadette	1766653708351	graduate	b
<div><div></div><div>Active</div></div>	Fritsch, Telly Lillie	360419183882882	graduate	d
<div><div></div><div>Active</div></div>	Gerhold, Sedrick Jayson	420519690626914	undergrad	ju
<div><div></div><div>Active</div></div>	Gorczy, Anna Bernita	18447337064074	graduate	ay
<div><div></div><div>Active</div></div>	Gottlieb, Abbey Abbey	349246998317941	undergrad	ai
<div><div></div><div>Active</div></div>	Graham, Beaulah	265035487636620	undergrad	m
<div><div></div><div>Active</div></div>	Hilpert, Arthur Kaci	7465958630177	graduate	gr
<div><div></div><div>Active</div></div>	Hirthe, Era Trudie	655304524928436	faculty	ki
<div><div></div><div>Active</div></div>	Jewess, Kailyn	92362787956219	graduate	cl
<div><div></div><div>Active</div></div>	Jewess, Karli Bernard	662095559576421	undergrad	lii
<div><div></div><div>Active</div></div>	Kessler, Cade Treva	831099341670498	faculty	w
<div><div></div><div>Active</div></div>	Klute-Folio 2, Uschi	00180062573	undergrad	u:
<div><div></div><div>Active</div></div>	Kreiger, Barrett Devin	481573440960549	undergrad	ja
<div><div></div><div>Active</div></div>	Littel, Sofia	757563877102114	staff	n.
<div><div></div><div>Active</div></div>	Mayert, Lisette	661617218338733	staff	ju

Champlin, Dagmar Emerald

Champlin, Dagmar Emerald

Expand all

^ User information

Record last updated: 05/04/2019 11:38

Last name

Champlin

First name

Dagmar

Middle name

Emerald

Barcode

731434363629411

Patron group

staff

Status

active

Expiration date

19/02/2020

^ Patron Blocks

0

^ Extended information

^ Contact Information

^ Proxy/Sponsor

0

^ Fees/Fines

0

^ Loans

0

^ Requests

0

^ User permissions

0

^ Service points

2

FOLIO

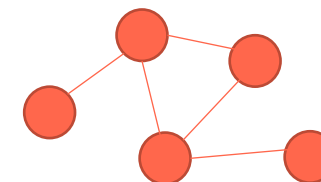
Architecture





Technical Concept

- Open platform: Library Service Platform (LSP)
- Platform provides infrastructure for functional modules
- Functional modules → self-contained programs
 - Can be developed independently from each other
 - Can be selected and installed one-by-one
 - Communication through interfaces
- Design based on micro-services idea





Technical Concept

- Promotes various support models
 - Cloud based, hosting, local
 - Commercial, library network, self
- Multi-tenancy
- Flexibly extendable, modular
- „Plug and play“ application
- Based on today's requirements and aiming to future needs



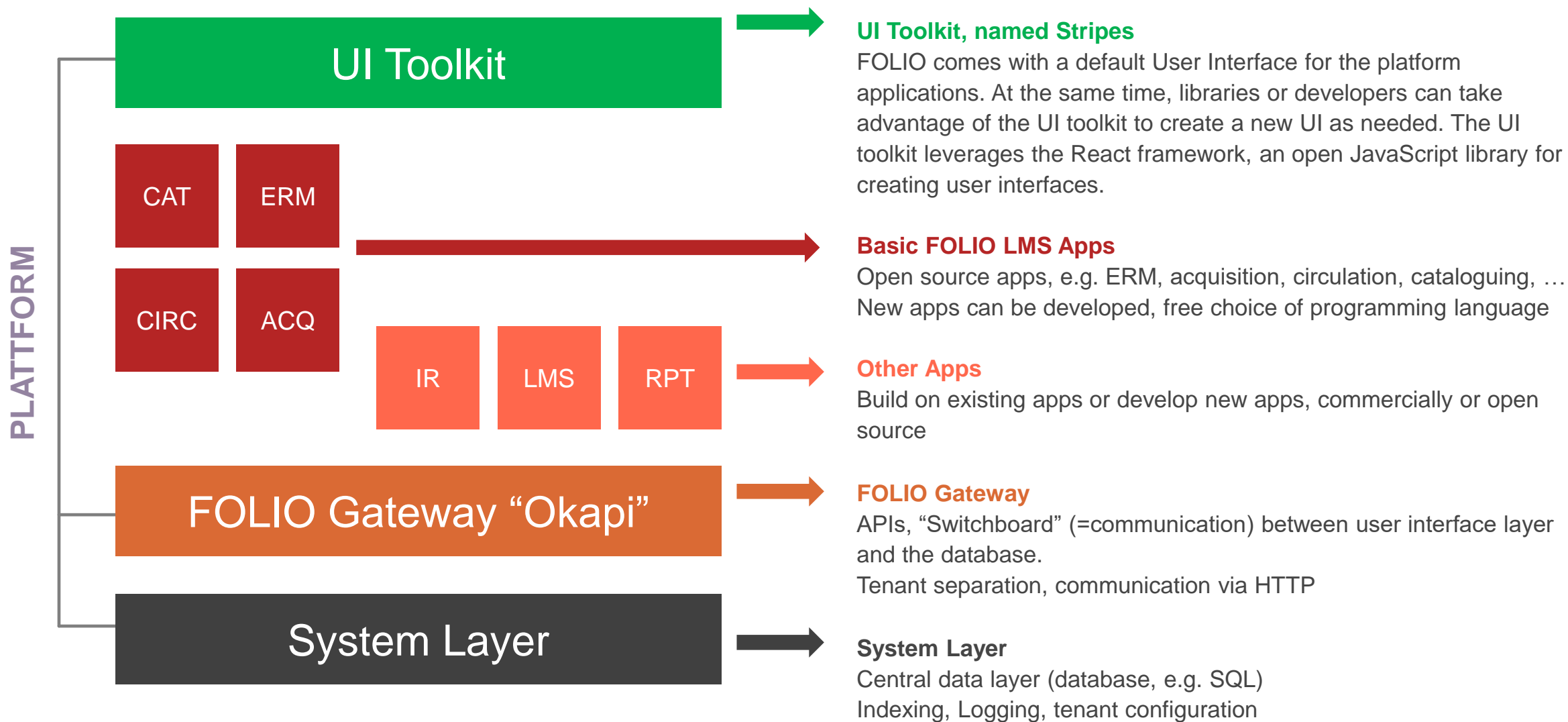
Platform design

„APIs all the way down“

- This means that
 - any developer can interact with any layer in the platform, and
 - no component is too big to be replaced



Technologies



Technologies

Modern software stack of proven components

Frontend (= in the browser)

- JavaScript (ECMAScript 6)
- React/Redux

Backend (= on the server)

- Java 8 (Java 11 [soon](#))
- Vert.x (asynchronous communication)
- RAML
- PostgreSQL
 - JSONB (NoSQL) and
 - Relational SQL



React and Redux

- Are open source JavaScript web frameworks for single-page applications (SPAs)
- React provides a framework for rendering user-interface components
- Redux is a data container that makes reading from and writing to the backend easy
- <https://reactjs.org/> und <https://redux.js.org/>



Stripes

- JavaScript program library for frontend modules
- Based on React + Redux
- Customized to Okapi's and FOLIO's needs
 - Communication via Okapi to backend modules
 - Granular user rights
 - Locale (language, date format, ...)
 - Hotkeys (keyboard shortcuts)
 - Logging via Okapi
- <https://github.com/folio-org/stripes-core/#readme>



vert.x

- Library for Java
- Facilitates simple concurrency
- Avoids many problems of parallel programming
- Asynchronous communication
 - Vert.x wraps a synchronous HTTP REST request into an asynchronous interface
- Reactive programming
- Design pattern “Reactor”
- <https://vertx.io/>



RAML

- RAML = RESTful API Modeling Language
- Describes the interface of any module
- Generators take a RAML file and generate
 - Interface documentation: <https://dev.folio.org/doc/api/>
 - Java Code (Interfaces)
 - Validation, invoked by Okapi when calling an interface:
 - Sufficient user permissions?
 - Correct data format?
- <https://github.com/folio-org/raml-module-builder>



Database selection

- PostgreSQL
 - 2016 MongoDB proof of concept
- PostgreSQL became the DBMS of choice because it support both (!)
 - relational SQL database model
 - document based NoSQL database model
- NoSQL = Not-only-SQL, in this case document based (JSON documents)
- PostgreSQL can process JSON documents as JSONB, this is an efficient binary format where the JSON document is decomposed allowing indexing



JSON

- JSON = JavaScript Object Notation
- FOLIO stores most of the data as JSONB
- Data exchange format of most FOLIO APIs is JSON
- Vert.x offers extensive JSON support, JSON is vert.x' main exchange format
- JSON is a very common data exchange format for asynchronous browser server communication
 - This applies to Java as well



Database operation

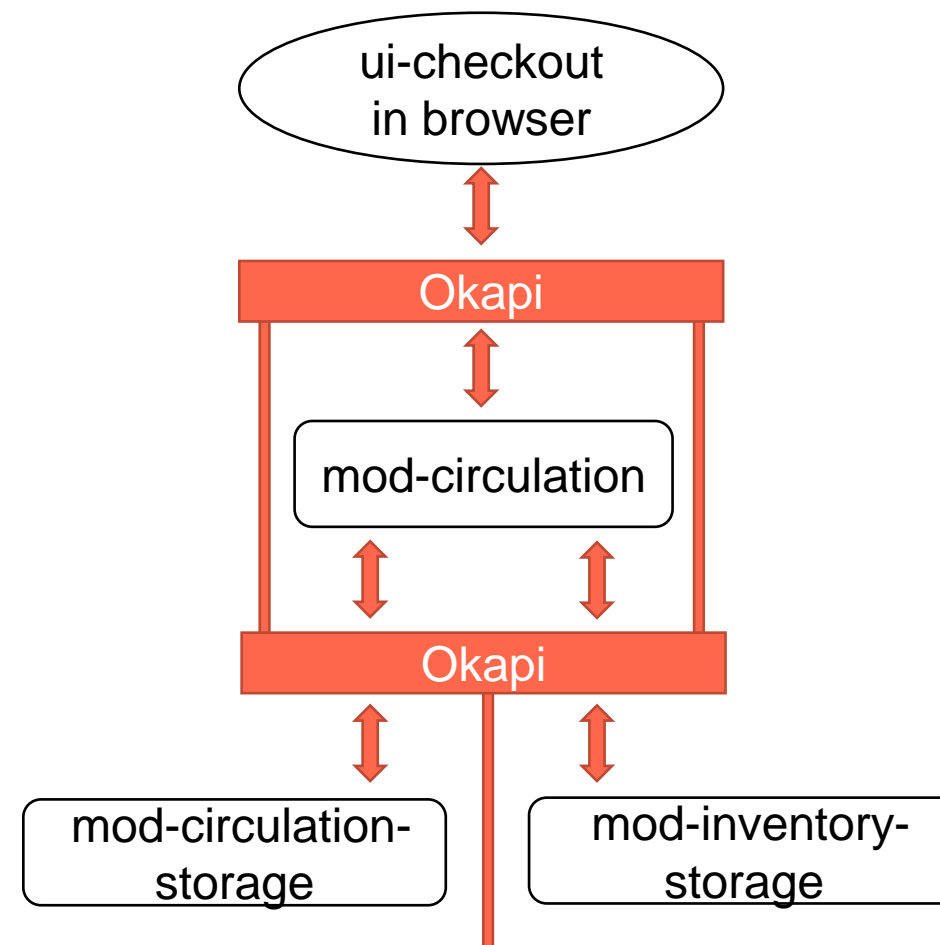
- Each storage module may start an own PostgreSQL instance
 - We use this for software development
- Use a parameter to connect an external PostgreSQL installation
 - We use this for our demo and test installations
 - Allows for high availability and replication with PostgreSQL cluster



Inter-module Communication

Example: Check-out app uses three backend modules – the module mod-circulation combines the loan data received from mod-circulation-storage with the title data from mod-inventory-storage and returns the merged data set.

*-storage = database abstraction layer





Okapi

- Okapi implements API gateway pattern
- Is the tenant allowed to access the module?
- If several versions for one module run:
 - Select the version that has been activated for the tenant
- Has the user sufficient access rights?
- Validating the parameters passed into the API
- Passing the API request to the module
- <https://github.com/folio-org/okapi>



Modules

- Modules communicate via interfaces *only*
- Independence
- Easy to maintain, easy to exchange
- License can be selected independently per module:
 - Proprietary
 - Viral license like GPL or AGPL
 - Permissive free license like Apache or MIT
- Select programming language and software libraries independently
 - Core modules use the same software stack (Java, Vert.x, ...)



App and Module Architecture

- Functionality is split into apps by business area
- This reduces inter-app data exchange
- Example: Check-out and Check-in are one app only
 - Even if there are two buttons on the user interface
- Not: Nanoservices with mini modules
- One developer team per app
- An app usually has a GUI module, a business login module and a data storage (access to database) module



Vagrant und Docker

- Install a complete FOLIO system using Ansible:
 - <https://github.com/folio-org/folio-ansible>
- Manual installation with explanations:
 - <https://github.com/folio-org/folio-install/blob/master/single-server.md>
- All modules as Docker containers:
 - <https://hub.docker.com/u/folioorg/>
 - <https://hub.docker.com/u/folioci/>
- Download a complete FOLIO system as a Vagrant box:
 - <https://github.com/folio-org/folio-ansible/blob/master/doc/index.md>
 - <https://app.vagrantup.com/folio>

FOLIO

Technical Evaluation





Technical Evaluation

- The technical basis and the architecture of the FOLIO platform have been evaluated for three times:
 - by members of the OLE-Community
 - by EBSCO
 - by [Open Tech Strategies](#) (OTS)
- All evaluations were successful, and the suggestions resulted in improvements or prioritized issues
- The [OTS report](#) from January 2019 is online

FOLIO

Accessibility





Accessibility in FOLIO

Accessibility = easy to use for all, including those with disabilities

FOLIO aims at WCAG 2.1 priority AA compliance

WCAG:

- Web Content Accessibility Guidelines
- International standard for accessibility
- Required by law in many countries
- European Union law requires new websites of public sector bodies to comply with WCAG 2.1 priority AA from 23 September 2019 on



Stripes Accessibility

- Stripes is FOLIO's GUI toolkit, provides reusable components
- Designed to be accessible
- Accessibility and usability is checked on a regular basis
 - In usability labs
 - During our monthly power hour
- Feedback improved Stripes components and the guidelines
- Accessibility architecture = built-in by design

<https://wiki.folio.org/display/A11Y>

<https://ux.folio.org/docs/guidelines/accessibility/>

FOLIO

Query language –
From CQL to
GraphQL





Query language – from CQL to GraphQL

- CQL = Contextual Query Language
 - is a DBMS agnostic query language
 - is used by the front-end and by back-end modules that query data records from other back-end modules
- CQL has limitations
- Solution: A GraphQL module was added
- GraphQL supports advanced and complex queries
- This architectural improvement was possible because of FOLIO's microservices-like architecture
- <https://dev.folio.org/reference/glossary/#cql>
- <https://github.com/folio-org/mod-graphql>



Example GraphQL query

```
query {  
  instance_storage_instances(query: "title=baby") {  
    totalRecords  
    instances {  
      title  
      holdingsRecords2 {  
        callNumber  
        holdingsItems {  
          barcode  
        }  
      }  
    }  
  }  
}
```

Join three tables:
instance, holdings, item

Return selected fields only:
totalRecords, title,
callNumbers, barcode

FOLIO

Tenant separation





Tenant in FOLIO

- Tenant = completely independent institution
 - Branch library is not a tenant and uses granular hierarchical access rights.
- FOLIO supports cloud installations
- Tenants share cloud hardware and cloud software
- Strict tenant separation required
- For each combination of tenant and module we create a database user and a logical database:

```
CREATE ROLE ${university}_${module} ...;  
CREATE SCHEMA ${university}_${module}  
    AUTHORIZATION ${university}_${module};
```

- We have a [tenantSeparation unit test](#)



Tenant in FOLIO

- Each module runs the CREATE ROLE and CREATE SCHEMA commands when a new tenant needs to be activated
- Okapi passes the credentials of a database superuser to the module
- This is an architectural deficiency for security reasons
- Better design:
 - Only a central service has superuser rights
 - and creates the role and the schema
 - and passes the information to the module
- Architectural change recommended by OTS report and [on the way](#)

FOLIO

Additional DBMS support





DBMS support: PostgreSQL + ?

- Any back-end module may use any DBMS
- RAML Module Builder (RMB) is a FOLIO software library
 - Supports only PostgreSQL
 - Supports PostgreSQL JSONB columns
 - Reduces boilerplate code for each module
- Using RMB is the most easy way
- Most modules use it
- Some use Grails with PostgreSQL instead
 - mod-licenses, mod-agreements



DBMS support: PostgreSQL + ?

- OTS report recommends additional DBMS back-ends
- FOLIO has postponed decision
- What do you think?
- How should FOLIO prioritize it?



Thank you!

Please visit us at our FOLIO booth

Speakers

Martina Tumulla

works as a systems librarian at the North Rhine-Westphalian Library Service Centre (hbz) in Cologne, Germany. She supports FOLIO's development as co-convener of ERM subgroup and is member of Resource Management SIG, Consortia SIG and Product Council.

tumulla@hbz-nrw.de

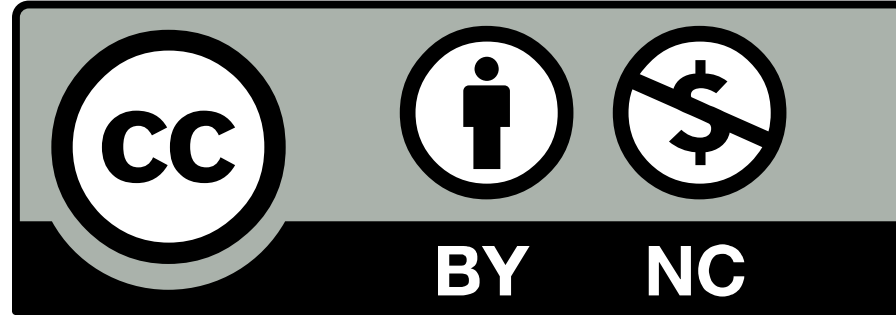
Julian Ladisch

works as a senior developer at the headquarters of GBV in Göttingen, Germany, and is active in the FOLIO project since its beginning in 2016. He is a member of the FOLIO platform core developer team.

julian.ladisch@gbv.de

FOLIO Links

- Code on GitHub
<https://github.com/folio-org>
- Dev Wiki
<https://dev.folio.org/>
- Demo Installation
<https://folio-demo.gbv.de/>
<https://folio-demo.hbz-nrw.de/>
(diku_admin / admin)
- FOLIO project website (in German)
<https://www.folio-bib.org/>
- FOLIO Wiki
<https://wiki.folio.org/>
- OLE Community
<https://www.openlibraryenvironment.org/>
- Open Library Foundation (OLF)
<http://www.openlibraryfoundation.org/>
- FOLIO
<https://www.folio.org/>



Attribution-NonCommercial 4.0 International

The text of this presentation is licensed under a
Creative Commons Attribution-NonCommercial 4.0 International License:
<https://creativecommons.org/licenses/by-nc/4.0/>