

## Fachbeitrag

Kirstin Kemner-Heek und Maike Osters

# FOLIO – ein Open-Source-System für das Bibliotheksmanagement

<http://doi.org/10.1515/abitech-2018-2007>

**Zusammenfassung:** FOLIO ist ein plattformbasiertes Open-Source-Bibliotheksmanagementsystem, dessen Entwicklung seit 2016 maßgeblich von EBSCO, Index Data und der OLE-Community vorangetrieben und unter einer Apache-2.0-Lizenz auf der Github-Instanz der Open Library Foundation bereitgestellt wird. Ziel ist die Entwicklung einer zukunftsfähigen, effizienten und flexiblen Software, die sowohl aktuelle als auch zukünftige Anforderungen von Bibliotheken erfüllen kann. FOLIO und seine Plattform Okapi werden technisch und funktional vorgestellt und ihre Einsatzszenarien auch für deutsche Bibliotheken erläutert.

**Schlüsselwörter:** FOLIO, Bibliotheksmanagementsystem, Open Source

## FOLIO – an open source library management system

**Abstract:** FOLIO is a platform-based open source library management system. Since 2016 its development has been significantly advanced by EBSCO, Index Data and the OLE community. It is published on the Open Library Foundation's Github under Apache 2.0 license. Goal is the development of a sustainable, efficient and flexible software to fulfill the needs of today's library requirements as well as future ones. FOLIO and its platform Okapi are introduced technically as well as functionally, and its implementation scenarios are explained.

**Keywords:** FOLIO, library management system, open source

## 1 Das Bibliotheksmanagementsystem FOLIO

Das Open-Source-Projekt FOLIO wurde 2016 von EBSCO<sup>1</sup>, Index Data<sup>2</sup> und der OLE-Community<sup>3</sup> angestoßen. Ziel ist ein zukunftsfähiges, offenes, innovatives und flexibel erweiterbares Bibliotheksmanagementsystem auf einer neuen technischen Plattform zur Verfügung zu stellen. Die Organisation der Anwendungen auf der Plattform über sogenannte „Apps“ ermöglicht sowohl die Erfüllung heutiger funktionaler Anforderungen von Bibliotheken als auch die Entwicklung zukünftiger und zusätzlicher Services.

FOLIO als Bibliotheksmanagementsystem ist das erste Projekt auf der neuen „Library Service Platform“ (LSP) Okapi und bereits für das Jahr 2018 in einer ersten Version, V1 Beta, angekündigt.<sup>4</sup> Zielgruppe sind in erster Linie wissenschaftliche Bibliotheken unterschiedlicher Größe und Ausrichtung. Das System wird so flexibel und individualisierbar konzipiert, dass es sich in die Strukturen einer Hochschule integrieren und auf die individuellen Bedürfnisse von Bibliotheken anpassen lässt. Zudem lässt die strikt modulare Struktur eine schrittweise Ablösung bestehender Bibliothekssysteme zu, was die notwendigen Übergangsprozesse beherrschbar und mit möglichst wenigen

<sup>1</sup> Vgl. EBSCO Information Services. <https://www.ebsco.com/> (06.04.2018).

<sup>2</sup> Vgl. Index Data. <https://www.indexdata.com/> (06.04.2018).

<sup>3</sup> Vgl. OLE-Community. <https://www.openlibraryenvironment.org/> (06.04.2018). OLE-Entwicklungspartner sind aktuell Auburn University Libraries, Cornell University Libraries, Duke University Libraries, Fenway Library Organization (FLO), Lehigh University, SOAS (UK), Texas A&M University Libraries, University of Chicago Libraries, University Libraries at Colorado Boulder sowie die beiden deutschen Verbände hzb und GBV mit ihren Verbundzentralen und die UB Leipzig. Assoziiert sind die Andrew W. Mellon Foundation, die Global Open Knowledgebase (GOKb), FOLIO und die Open Library Foundation (OLP).

<sup>4</sup> Vgl. Kaplanian, Harry. FOLIO. The Future of Libraries is Open. A new Open Library Collaboration. Präsentation FOLIO-Informationstag, Stuttgart, 2017. Fol. 7. <https://www.ole-germany.org/wp-content/uploads/2016/04/2017-10-17-FOLIO-Stuttgart-roadmap-kaplanian.pdf> (06.04.2018).

Brüchen gestaltbar macht. Die gesamte Systemarchitektur ist konsequent über offene Schnittstellen entworfen, sowohl innerhalb der Plattform zwischen den Apps als auch nach außen. Unterschiedliche Systeme, Services von Drittanbietern oder auch Eigenentwicklungen können an FOLIO angebunden werden.

Die Plattform Okapi bildet die technische Basis u. a. für die funktionalen Module des Bibliotheksmanagementsystems. Ihre Entwicklung begann bereits im November 2015 durch die Firma Index Data. Sie ist in zweierlei Hinsicht offen, sowohl bezüglich der Beteiligung weiterer Partner, wie Bibliotheken, kommerzielle Anbieter oder Service Provider, als auch bezüglich funktionaler Erweiterungen und neuer Services.

Das Konzept sieht dabei vor, dass kein Baustein so groß sein soll, dass er nicht wieder ausgetauscht werden kann (Mikroservice-Idee<sup>5</sup>). FOLIO ist voll mandantenfähig konzipiert und von Grund auf cloudfähig. Ein hervorzuhebendes Alleinstellungsmerkmal ist die Wahlmöglichkeit des Betriebsmodells: FOLIO kann wahlweise in einer (kommerziellen oder public service) Cloud, als Hosting-Lösung (SaaS) oder auch lokal betrieben werden.

Die technische Architektur sowie der Funktionsumfang von FOLIO werden im Weiteren ausführlich erläutert. Für eine erste Version werden Apps in den Bereichen Ausleihe, Erwerbung und Verwaltung aller Medientypen, Metadatenmanagement, Nutzermanagement sowie Budgetkontrolle, Lieferantendatenverwaltung und Systemadministration entwickelt. Die weitere Entwicklung und Integration neuer Services, z. B. in den Bereichen „Resource Sharing“ oder Forschungsdatenmanagement, sind in Vorbereitung.

## 2 Organisationsstruktur

Nachdem die OLE-Community bereits in den Vorjahren das Bibliotheksmanagementsystem OLE unter der Dachorganisation Quali Foundation<sup>6</sup> entwickelt und in drei Partnerbibliotheken zum Einsatz gebracht hatte, wurde Ende 2015 deutlich, dass die Quali Foundation in die Kommerzialisierung strebt und außerdem die Softwarebasis technisch keine Zukunft mehr hatte. Gemeinsam mit der Firma EBSCO, die bereits seit 2013 als Entwicklungs-

partner innerhalb der OLE-Community mitarbeitete, entschied sich die Community für einen Neuanfang unter einer ebenfalls neu zu gründenden Dachorganisation. Dieser Schritt erfolgte Mitte 2016 unter dem Namen „Open Library Foundation“.<sup>7</sup> Die Foundation stellt die Infrastruktur für innovative, Open-Source-basierte Projekte im Bereich „Higher Education“ zur Verfügung, sichert als „safe haven“ über GitHub-Repositories den entstehenden Code nachhaltig unter einer Apache 2.0-Lizenz und hat sich zum Ziel gesetzt, die Zusammenarbeit zwischen Bibliotheken, Entwicklern, kommerziellen Anbietern sowie Service Providern zu fördern. Als Non-Profit-Organisation wird die Foundation von einem unabhängigen Board geleitet. Jedes Projekt, das der Dachorganisation beiträgt, hat dabei wiederum seine eigene Organisationsstruktur. Einige dieser Projekte stellen zum jetzigen Zeitpunkt die OLE-Community, FOLIO und die Global Open Knowledgebase (GOKb)<sup>8</sup> dar.

Das FOLIO-Projekt wird aktuell von EBSCO, Index Data und der OLE-Community getragen. EBSCO finanziert und unterstützt das Projekt im Rahmen seiner spezifischen Non-Profit-Fördertradition im Bereich „Higher Education“ auf einer strikten Open-Source-Basis. Darüber hinaus werden Product Owner und Managementkapazitäten zur Verfügung gestellt. Die Firma Index Data entwickelt in EBSCOs Auftrag die Plattform Okapi und einige zentrale FOLIO-Apps. Ebenso übernimmt sie die Entwicklungsleitung und koordiniert die verschiedenen Entwicklerteams.

Die OLE-Community beteiligt sich ebenfalls maßgeblich mit finanziellen und personellen Ressourcen an der Entwicklung. Aus den Mitgliedsbeiträgen und aus eingeworbenen Mitteln der Mellon Foundation<sup>9</sup> werden eigene Entwicklerteams sowie ein kleines Organisationsteam (Managing Director, Director of Strategic Planning, Project Manager) bezahlt. Aus den Partnerbibliotheken kommen sowohl das fachliche Knowhow in Form von Mitarbeit in Arbeitsgruppen als auch die Bereitstellung eigener Entwickler, Product Owner und Project Manager. Weitere Entwicklerteams wurden von EBSCO extern angeworben. Darüber hinaus gibt es einige weitere Initiativen von interessierten Bibliotheken und Firmen, die an FOLIO mitarbeiten. Alle zusammen bilden die FOLIO-Community.

Die Entwicklung der Software wird über verschiedene Gremien organisiert. Den übergeordneten Zusammenhalt leistet hierbei der Product Council: Er setzt sich aus der

<sup>5</sup> Vgl. Lewis, James, Martin Fowler. Microservices – a definition of this new architectural term, 25.03.2014. <https://martinfowler.com/articles/microservices.html> (06.04.2018); und vgl. Wolff, Eberhard. Microservices: Grundlagen flexibler Softwarearchitekturen. Heidelberg 2015, 2. <http://microservices-buch.de/> (06.04.2018).

<sup>6</sup> Vgl. Quali Foundation. <https://quali.org/> (06.04.2018).

<sup>7</sup> Vgl. The Open Library Foundation. <http://www.openlibraryfoundation.org/> (06.04.2018).

<sup>8</sup> Vgl. Global Open Knowledgebase. <https://gokb.org/> (06.04.2018).

<sup>9</sup> Vgl. The Andrew W. Mellon Foundation. <https://mellon.org/> (06.04.2018).

OLE-Community, EBSCO und Index Data zusammen und ist offen für weitere Mitarbeitende. Er fungiert als Informationsvermittler, bildet das administrative sowie organisatorische Abstimmungsgremium, koordiniert und begleitet die fachlichen Arbeitsgruppen (Special Interest Groups – SIGs) und ist verantwortlich für die Roadmap.

Die bibliotheksfachliche Expertise kommt zum einen aus den am Projekt beteiligten Bibliotheken und zum anderen aus Serviceeinrichtungen wie den Verbundzentralen oder Konsortien. In themenbezogenen Arbeitsgruppen, den SIGs, treffen sich zumeist wöchentlich die jeweiligen Experten in Webkonferenzen, um über ein bestimmtes Modul, ein Fachgebiet oder verschiedene Fragestellungen zu diskutieren. Die SIGs werden von einem sogenannten „Convener“ geleitet, der als Schnittstelle zum Product Council fungiert, die Arbeitsgruppe moderiert und die Agenda erstellt. Des Weiteren wird die Arbeit von einem Product Owner begleitet. Dieser sorgt für die Umsetzung der funktionalen Anforderungen in technische Spezifikationen, bildet die Schnittstelle zu den Entwicklern und ist für die Einhaltung von Zeitplan und Entwicklungszielen für sein „Produkt“ (= Entwicklungsmodul) verantwortlich. Je nach Bedarf nehmen auch Entwickler an den Treffen teil. Aktuell gibt es folgende 10 Gruppen:

- Metadata Management (Katalogisierung),
- Resource Access (Ausleihe),
- Resource Management (Erwerbung inkl. Electronic Resource Management (ERM) und Bestandspflege),
- User Management (Nutzerverwaltung),
- Internationalization (Internationalisierung),
- Consortia (Konsortiale Anforderungen),
- Reporting (Berichtswesen und Statistik),
- Privacy (Datenschutz),
- Accessibility (Zugänglichkeit/Barrierefreiheit) und
- System Operations and Management (Datenbank-Administration, Installation, Debugging, Migrationsfragestellungen).

Das deutsche FOLIO-Team hat mittlerweile Vertreterinnen bzw. Vertreter in allen SIGs. Diese kommen aus den beiden Verbundzentralen von hbz und GBV<sup>10</sup>, aus den Verbundbibliotheken sowie aus weiteren Verbänden und Bibliotheken. Eine Mitarbeit in den SIGs steht allen Interessierten offen. Interessierte können sich beim Folio-Team der Verbundzentralen melden.

<sup>10</sup> Vgl. OLE Germany. <https://www.folio-bib.org/> (06.04.018).

### 3 Kommunikation

Die Kommunikation<sup>11</sup> innerhalb des Projekts und nach außen läuft über verschiedene Kanäle. Es gibt ein Wiki für das Gesamtprojekt und die Facharbeitsgruppen<sup>12</sup> sowie eine Website mit der Entwicklerdokumentation<sup>13</sup>. In beiden finden sich für verschiedene Zielgruppen (technisch und funktional) gute Übersichten zum Thema „Getting started in FOLIO“<sup>14</sup> sowie weiterführende Informationen zum System und dem Projekt selbst. Der Softwarecode wird über GitHub<sup>15</sup> zur Verfügung gestellt. Die technische Spezifikation sowie die Entwicklung wird über ein Ticketsystem (Jira)<sup>16</sup> gesteuert. Fachdiskussionen werden asynchron in „FOLIO Project Discussion“<sup>17</sup> geführt oder auch in den Channels im FOLIO-Workspace auf Slack<sup>18</sup>. Ergänzt wird das Informationsangebot um gruppenspezifische E-Mail-Listen und Videokonferenzen. Alle Tools bieten Interessierten die Möglichkeit der freien Anmeldung sowohl zu reinen Informationszwecken als auch zur aktiven Beteiligung. Neben den bestehenden Partnerbibliotheken kann jeder Interessierte der Community beitreten.

Als wesentlicher Beitrag zum Erfolg des Projekts tragen persönliche Treffen, sogenannte „Face-to-Face Meetings“ bei. Entweder in spezifischen Gruppen, wie einem Entwicklertreffen, oder als große FOLIO-Konferenz – genannt WOLFcon (World Open Library Foundation Conference).<sup>19</sup>

### 4 Funktionales Konzept

Die funktionale Entwicklung von FOLIO auf der LSP sieht die Konzeption der notwendigen Module für ein Bibliotheksmanagementsystem als Apps vor. Dabei sind die einzelnen Apps klein genug, dass sie jederzeit flexibel ergänzt oder ausgetauscht werden können. So besteht der

<sup>11</sup> Vgl. FOLIO-Wiki. FOLIO Communication Spaces. <https://wiki.folio.org/x/dgAB> (06.04.018).

<sup>12</sup> Vgl. FOLIO-Wiki. <https://wiki.folio.org/> (06.04.018).

<sup>13</sup> Vgl. FOLIO Developer. <https://dev.folio.org/> (06.04.018).

<sup>14</sup> Vgl. FOLIO-Wiki. Getting Started for Developers. <https://wiki.folio.org/x/fIAV> (06.04.2018); und FOLIO-Wiki. <https://wiki.folio.org/> (06.04.018).

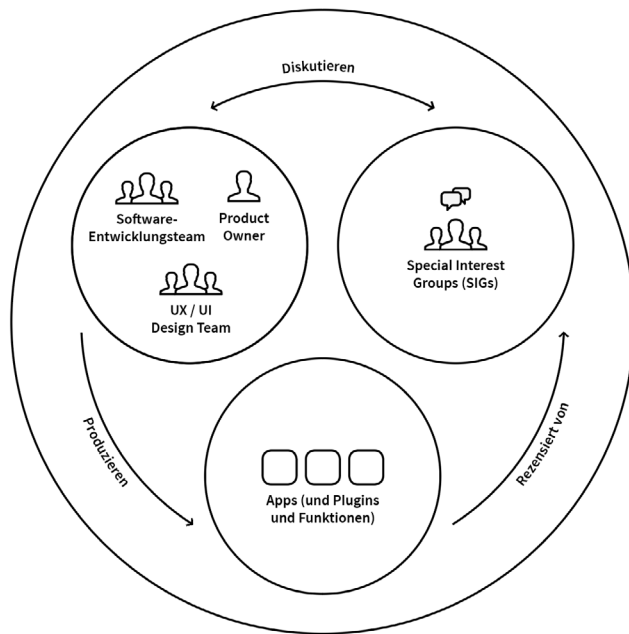
<sup>15</sup> Vgl. FOLIO-GitHub. <https://github.com/folio-org> (06.04.2018).

<sup>16</sup> Vgl. FOLIO-Jira. <https://issues.folio.org/secure/BrowseProjects.jspa> (06.04.2018).

<sup>17</sup> Vgl. Discuss.FOLIOProject. <https://discuss.folio.org/> (06.04.2018).

<sup>18</sup> Vgl. Slack. FOLIO-Chat. <https://slack-invitation.folio.org/> (06.04.2018).

<sup>19</sup> Z. B. vom 7.–11. Mai 2018 in Durham, North Carolina.



**Abb. 1:** Schematische Darstellung der Prozess- und Teamstrukturen (Quelle: Filip Jakobsen)

Bereich „Resource Management“ aus etlichen Apps, z. B. im Bereich Lieferanten- und Budgetmanagement, Rechnungswesen, Bestellung, Inventarisierung etc.

Der Entwicklung voraus ging ein sogenanntes „White Paper“<sup>20</sup>, das die minimalen Anforderungen an das System beschreibt und vom OLE Board im Juli 2016 bestätigt wurde. Die Entwicklung der funktionalen Module erfolgt seit Oktober 2016 entsprechend der vom Product Council verantworteten Roadmap.

Die Arbeiten beginnen zunächst mit der Spezifikation der Nutzeroberfläche. Die UX/UI-getriebene (User Experience/User Interface) Entwicklung setzt bei den Bedürfnissen des Anwenders an. Auf dieser Basis werden jedoch nicht nur die Oberfläche, sondern auch der Funktionsumfang, die Workflows und die Mensch-Maschine-Interaktion konzipiert. Die Ausarbeitungen erfolgen innerhalb der SIGs unter Anleitung der UX/UI-Designer. Erst wenn nach mehreren Feedbackrunden Arbeitsergebnisse in einem Prototyp abgestimmt wurden, werden diese Vorgaben vom Product Owner zu technischen Spezifikationen aufbereitet und an die Entwickler gegeben. Der Feedback-

zirkel beginnt erneut, bis der Code und seine Ausprägung an der Nutzeroberfläche von den SIGs bestätigt werden.

Auf diesem Entwicklungsweg ist die Ausleihe funktional bereits weit fortgeschritten: die Verbuchung an Ausleih- und Rückgabetheke, Bestellungen, Vormerkungen sowie das Anlegen des Ausleihreglements stehen zur Verfügung. Im Bereich des Resource Managements wurden Funktionalitäten im Bereich der Rechnungsbearbeitung, Lieferantendaten, Budgets und Workflows sowie das Dashboard entwickelt. Zudem wird aktuell die Anbindung der EBSCO-Knowledgebase via API realisiert. Erste Ansätze existieren bereits im Bereich der Administration des Systems. Derzeit werden in der Community die Rubriken Bemerkungsfelder, Benachrichtigungen, Nutzergruppen und -rechte und das Reporting diskutiert. Die Anforderung der Mehrsprachigkeit wird von Anfang an im FOLIO-Entwicklungsprozess mit umgesetzt.

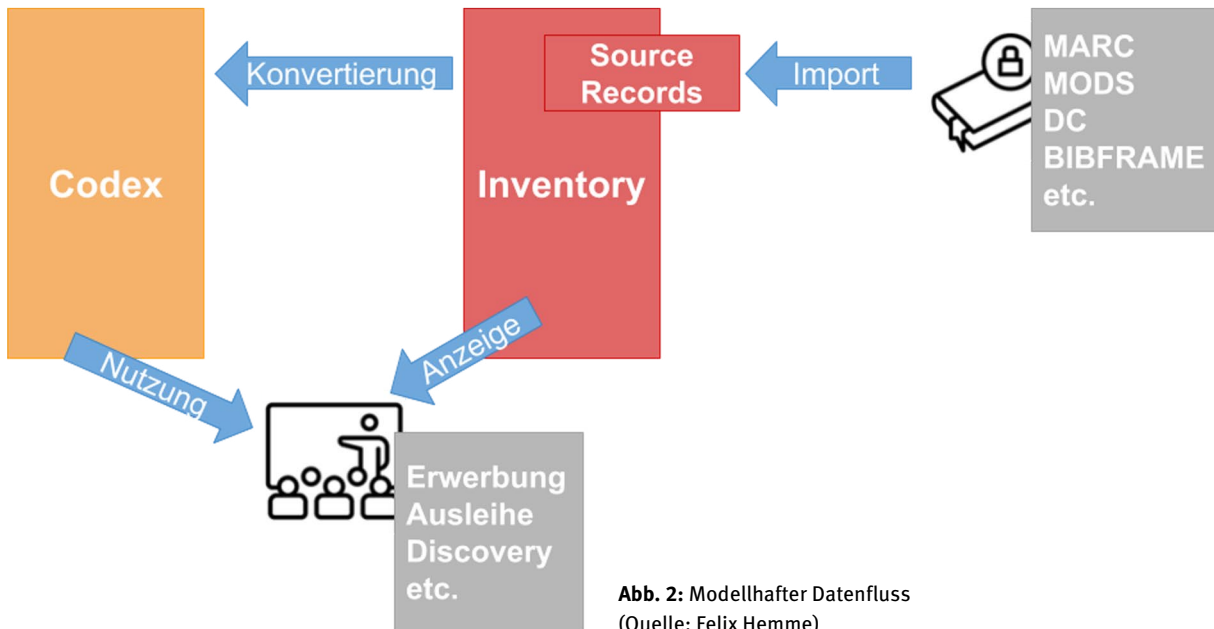
Parallel zu den klassischen Modulen Ausleihe und Erwerbung wird ein Metadatenkonzept modelliert, das formatoffen die Anbindung verschiedenster Datenquellen unterstützt. Den Kern der Metadatenverwaltung stellen der FOLIO Codex und das Inventory dar. Ein Metadatensatz in FOLIO wird in der Regel von einem sogenannten „Source Record“ abgeleitet, der in einem Standardformat, wie beispielsweise MARC 21, vorliegt. Während des Importvorgangs werden Mappings vom Ausgangsformat auf jedes Codex-Feld genutzt und es lassen sich zukünftig weitere Formate ergänzen. Der Import eines Source Records stößt die Generierung eines Datensatzes im Inventory an, der aus den Objekten Instance, Holding und Item besteht und die von der Metadata Management SIG definierten Properties<sup>21</sup> enthält.

Aus dem Inventory fließen ausgewählte Elemente in den FOLIO Codex, um anschließend für die Suche, Ergebnispräsentation und alle „nicht-Katalogisierungs-Workflows“ zur Verfügung zu stehen. Die Codex-Objekte beschreiben ein Arbeitsformat, auf das alle weiteren FOLIO-Apps Zugriff haben. Bibliografische Daten aus externen Knowledgebases sind zunächst nur im Codex enthalten, nicht jedoch im Inventory. Die Anbindung externer Knowledgebases wird somit erleichtert, denn es entfallen aufwendige Mappings, da der Codex nur ein minimales Set an Elementen enthält.

Über die eHoldings-App ist zum aktuellen Zeitpunkt die EBSCO Knowledgebase lizenzpflichtig angebunden. Über eine entsprechend weiterentwickelte API sollen sich zukünftig weitere Knowledgebases integrieren lassen, z. B.

<sup>20</sup> Vgl. The Open Library Environment. Open, Flexible, Extensible, Sustainable. OLE Partners establish objectives for an Open-Source, Next Generation Library Management System. <https://www.openlibraryenvironment.org/archives/69> (06.04.2018).

<sup>21</sup> Vgl. FOLIO-Wiki. FOLIO Platform. The Codex Metadata Model. Inventory Metadata Elements. <https://wiki.folio.org/x/VZkV> (06.04.2018).



**Abb. 2:** Modellhafter Datenfluss  
(Quelle: Felix Hemme)

die Global Open Knowledgebase (GOKb). Die eHoldings-App unterstützt grundlegende Bestandsverwaltungsfunktionen, wie die Suche nach Anbietern, Paketen und Titeln, und ermöglicht den Zugriff auf detaillierte Informationen dieser spezifischen Knowledgebase. Mitarbeiterinnen und Mitarbeiter der Bibliothek können die für sie relevanten Pakete auswählen und die Daten in wenigen Schritten in den Codex einspielen. Dabei ist die Angabe von individuellen Bestandsangaben ebenso möglich wie die gezielte Aus- und Abwahl einzelner Zeitschriftentitel.

Im März 2018 neu hinzugekommen ist eine Entwicklungsinitiative der deutschen FOLIO-Partner GBV und hbz. Da aktuell noch kein Entwicklerteam die Vorgaben für ein vollständiges ERM-Modul umsetzt, sollen die bisher noch fehlenden Bereiche Paket-, Lizenz- und Zugangsmangement und die Anbindung einer herstellerneutralen Knowledgebase nun aus der Community heraus realisiert werden. Dafür wurde die ERM-erfahrene Entwicklerfirma Knowledge Integration (K-Int)<sup>22</sup> unter Vertrag genommen, die mit Personal aus der Community verstärkt wird. Die funktionale Arbeit geschieht unter Anleitung einer UX/UI-Designerin und eines Product Owners in einer eigenen Arbeitsgruppe, die als Untergruppe der Resource Management SIG in enger Verknüpfung mit allen bestehenden FOLIO-Entwicklungen arbeiten wird. Alle Funktionalitäten dieser Initiative sollen in den FOLIO-Kern integriert sowie der Code auf GitHub öffentlich bereitgestellt werden.

<sup>22</sup> Vgl. Knowledge Integration. <https://www.k-int.com/> (06.04.2018).

Den Fortschritt der FOLIO-Gesamtentwicklungen und die bereits vorhandenen Module sowie Funktionalitäten können im UI-/UX-Prototypen<sup>23</sup> und in einem Demosystem<sup>24</sup> verfolgt und selber getestet werden.

## 5 Technisches Konzept

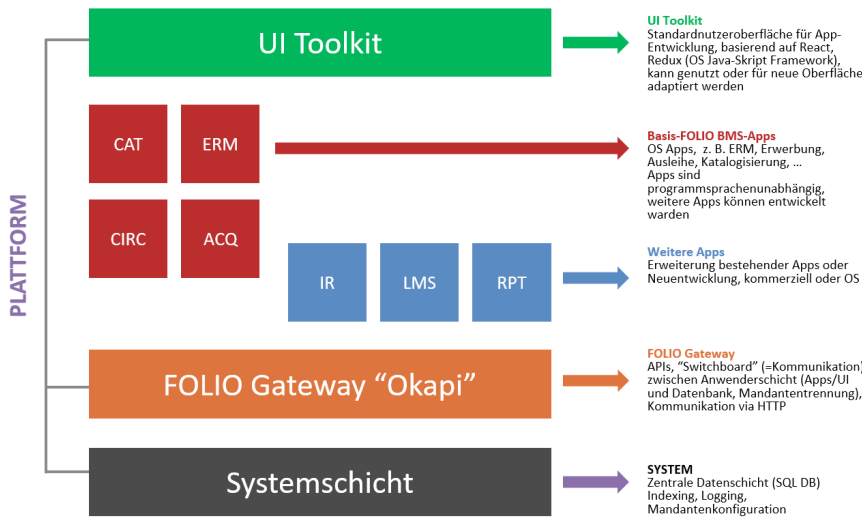
Die FOLIO zugrunde liegende technische Plattform Okapi ist als Herzstück der Entwicklung bereits weit fortgeschritten. Wie bereits erwähnt, orientiert sich das Konzept von FOLIO an der Idee der Microservices, welche unabhängig voneinander entwickelt und installiert werden können.

Die FOLIO-Plattform setzt sich aus verschiedenen Ebenen zusammen. Die Systemschicht umfasst die Datenbankabstraktion, zentrale Logging-Möglichkeiten und die Konfiguration der Mandanten. Das FOLIO-Gateway Okapi regelt die Mandantentrennung, es kontrolliert und steuert die gesamte HTTP-Kommunikation. Dies umfasst die Kommunikation innerhalb einer App mit mehreren Services, zwischen den Services unterschiedlicher Apps und zwischen einer App und der Systemschicht. Über das UI-Toolkit wird die Anwenderoberfläche generiert und eine vereinfachte Anbindung an die Services bereitgestellt.

Die auf der Plattform über APIs verknüpften Apps bilden die Funktionsmodule für den Anwender.

<sup>23</sup> Vgl. FOLIO Prototyp. <http://ux.folio.org/> (06.04.2018).

<sup>24</sup> Vgl. FOLIO-Demosystem. <https://folio-demo.gbv.de/> (06.04.2018). Login mit `diku_admin/admin`.



**Abb. 3:** Struktur der FOLIO-Plattform  
(Quelle: Harry Kaplanian, Julian Ladisch)

## 6 Moderner Softwarestack

FOLIO setzt einen modernen Softwarestack aus bewährten Komponenten ein. Im Browser läuft JavaScript, das in ECMAScript 6 (ES6) unter Verwendung von React<sup>25</sup> und Redux<sup>26</sup> programmiert wird. React bietet ein Grundgerüst für die Ausgabe von User-Interface-Komponenten in HTML, Redux stellt einen Datencontainer bereit, der das Lesen vom und Schreiben zum Backend vereinfacht. Alle Apps eines Mandanten werden zu einer Einseitenwebanwendung zusammengebaut (Single-Page-Applikation, SPA).

Stripes<sup>27</sup> ist eine auf React und Redux aufbauende JavaScript-Programmbibliothek von FOLIO, die auf Okapi zugeschnitten ist und bei der Kommunikation via Okapi zu den Backendservices granulare Nutzerrechte, eine Locale (Sprache, Datumsformat usw.)<sup>28</sup>, Hotkeys (Tastaturabkürzungen) und das Logging ermöglicht.

Die Backendservices auf dem Server sind in Java 8 programmiert, dabei werden Vert.x, RAML und PostgreSQL mit JSONB und relationalem SQL verwendet. Vert.x<sup>29</sup> bietet in Java die reaktive Programmierung in einfacher Nebenläufigkeit, ermöglicht also asynchrone Kommunikation, was viele Probleme paralleler Programmierung umgeht, aber die Performancevorteile gleichzeitiger Verarbeitung hat.

<sup>25</sup> Vgl. React. <https://reactjs.org/> (06.04.2018).

<sup>26</sup> Vgl. Redux. <https://redux.js.org/> (06.04.2018).

<sup>27</sup> Vgl. GitHub. Stripes Core. <https://github.com/folio-org/stripes-core/#readme> (06.04.2018).

<sup>28</sup> Vgl. Wikipedia. Locale. <https://de.wikipedia.org/wiki/Locale> (06.04.2018).

<sup>29</sup> Vgl. Vert.x. <http://vertx.io/> (06.04.2018).

Die Schnittstellen der Services werden mit RAML, der RESTful API Modeling Language, spezifiziert.<sup>30</sup> Aus dieser Schnittstellenbeschreibung werden automatisch die Schnittstellendokumentation<sup>31</sup> generiert und der Java-Code für die Interfaces erzeugt. Die RAML-Datei legt auch fest, welche Validierung Okapi beim Schnittstellenaufbau durchführt: Hat der aufrufende Benutzer die erforderlichen Rechte? Ist das Datenformat korrekt?

Als Datenbankmanagementsystem verwendet FOLIO PostgreSQL. Im Jahr 2016 wurden auch Tests mit MongoDB gemacht, jedoch fiel die Wahl auf PostgreSQL, da es gleichzeitig das relationale Datenbankmodell wie auch ein dokumentenbasiertes NoSQL-Datenmodell (Not-only-SQL) unterstützt. PostgreSQL kann JSON-Dokumente als JSONB verarbeiten, also in einem effizienten binären Format, bei dem das JSON-Dokument in seine Bestandteile zerlegt und dadurch indexierbar gemacht wird. Die bisherigen Apps verwenden überwiegend JSONB.

Bereits auf Datenbankebene werden die Daten unterschiedlicher Mandanten strikt getrennt. Unter einem Mandanten wird eine völlig unabhängige Institution verstanden; eine Institutsbibliothek ist üblicherweise kein eigener Mandant, sondern über granulare hierarchische Zugriffsrechte organisatorisch in die Universität ein- und an die Zentralbibliothek angebunden. Je Mandant und App legt FOLIO eine eigene logische Datenbank (PostgreSQL „Schema“) mit eigenem Datenbanknutzer (PostgreSQL „Role“) an. Beim Aufbau einer Datenbankverbindung mit diesem Datenbanknutzer garantiert PostgreSQL die strikte

<sup>30</sup> Vgl. GitHub. RAML Module Builder. <https://github.com/folio-org/raml-module-builder> (06.04.2018).

<sup>31</sup> Vgl. FOLIO Developer. Reference. API documentation. <https://dev.folio.org/doc/api/> (06.04.2018).

Abschottung der Mandanten. Verwenden die Apps eine gemeinsame PostgreSQL-Installation, kann diese als PostgreSQL-Cluster Hochverfügbarkeit und Replikation sicherstellen. Für die Softwarequalität sorgen sowohl manuelle Code-Reviews als auch automatische Codeanalysen über SonarCloud.<sup>32</sup>

Die technische Basis der FOLIO-Plattform haben sowohl Vertreter der OLE-Community als auch EBSCO evaluiert. Die Einschätzung war – unter Berücksichtigung, dass FOLIO noch in Entwicklung ist – positiv. Ergänzende Verbesserungsvorschläge und Hinweise wurden inzwischen größtenteils umgesetzt.

## 7 Implementierungsmöglichkeiten und zukünftige Einsatzszenarien

Auch wenn FOLIO aktuell noch nicht produktiv eingesetzt wird, werden die Migrationspläne der OLE-Partner fortlaufend dokumentiert.<sup>33</sup> Zudem sind schon in dieser frühen Phase mögliche Einsatz- und Implementierungsszenarien erkennbar, die von unterschiedlichen Interessen geprägt sind. Diese Szenarien und die notwendigen Umsetzungsschritte werden in einer speziellen SIG „System Operations and Management: Data Migration“ erarbeitet.

EBSCO verfolgt das Ziel, FOLIO schnellstmöglich kommerziell als Software-as-a-Service (SaaS) verknüpft mit speziellen EBSCO-Services anzubieten. Von einem ähnlichen Interesse kann auch bei anderen Anbietern ausgegangen werden, z. B. ByWater Solutions<sup>34</sup> und Index Data. Zukünftig werden auf der Plattform unter der Nutzung der FOLIO-Basis-Apps verschiedene Paketlösungen kommerziell angeboten.

Daneben werden Service Provider aus dem öffentlich-rechtlichen Bereich, z. B. die Verbundzentralen von hzb und GBV, einen Service für ihre Verbundbibliotheken bereitstellen. Zusammen mit anderen deutschen Interessensparteien können hier Kräfte gebündelt und spezifische Servicepakete für deutsche Bibliotheken gemeinsam entwickelt und gepflegt werden. Die verschiedenen

Bibliothekstypen im deutschsprachigen Raum haben unterschiedlichste Anforderungen sowie strategische Ausrichtungen, welche auch die zukünftigen Bibliotheksmanagementsysteme erfüllen müssen. Ebenso verändern sich Funktionsumfang und Einsatzmöglichkeiten der Systeme analog zu den sich verändernden Aufgaben der Bibliotheken.<sup>35</sup> Eine besondere Stärke der Entwicklung von Funktionalitäten innerhalb der FOLIO-Community oder einer seiner, z. B. regionalen, Untergruppen bietet die direkte Einbeziehung der Expertise von Fachpersonal aus den Bibliotheken. Entwickelt wird entsprechend den Anforderungen der Bibliotheken, die aktiv an der Gestaltung des Systems mitwirken und somit gemeinsam mit den entsprechend beauftragten Programmierern und den jeweiligen Projektverantwortlichen eine offene und transparente Entwicklung ermöglichen. Die lokale Nutzung einer FOLIO-Instanz in Eigenregie einer Bibliothek oder eines kleineren Konsortiums ist ebenfalls ein Einsatzszenario, das vor allem bei den amerikanischen OLE-Partnern verfolgt wird.

Offen ist, wie die gemeinsame Entwicklung neuer Services, der Ausbau der bestehenden Apps und die Softwarepflege zukünftig zuverlässig, effektiv und nachhaltig organisiert werden können. Die bestehenden Anstrengungen der FOLIO-Community um Klärung dieser sogenannten „Governance-Fragen“ werden aktuell mit der Verabschiedung der FOLIO-Charter<sup>36</sup> und der OLE-Charter<sup>37</sup> unterstützt. Diese stärken die Community und sollen gemeinsam mit den zukünftigen Erfahrungen eines produktiven Einsatzes der Software in einigen Pilotbibliotheken die Basis bieten, auf der die weitere Entwicklung aufsetzen kann.

In allen bisher beschriebenen Businessmodellen sind die Weiterentwicklung und Pflege des FOLIO-Basisangebots, aber auch die Entwicklung neuer Services zu klären. Die Basisfunktionalität von FOLIO ist ebenso fertigzustellen, wie bereits jetzt neue Serviceideen geplant werden, etwa die nahtlose Integration in die universitären Strukturen (APIs zu Haushaltüberwachungssystemen, Einbindung auf eLearning Plattformen, Anbindung an Forschungsdatennachweise etc.) sowie die Unterstützung weiterer Wege des Bestandsaufbaus, wie Open Access.

<sup>32</sup> Vgl. SonarCloud. Folio-org. <https://sonarcloud.io/organizations/folio-org/projects> (06.04.2018).

<sup>33</sup> Vgl. OLE Partner – FOLIO Deployment Plans. [https://docs.google.com/spreadsheets/d/1y4ZE8jXcZ6FaVCA1tBE0300a42Vk\\_7PfwPUb5yBfxX0/edit#gid=0](https://docs.google.com/spreadsheets/d/1y4ZE8jXcZ6FaVCA1tBE0300a42Vk_7PfwPUb5yBfxX0/edit#gid=0) (06.04.2018).

<sup>34</sup> Vgl. ByWater Solutions. FOLIO Library Services Support Between ByWater Solutions and EBSCO Information Services. <https://bywatersolutions.com/2018/02/01/folio-library-services-support-between-bywater-solutions-and-ebco-information-services/> (06.04.2018).

<sup>35</sup> Vgl. Wissenschaftliche Bibliotheken 2025. dbv, 2018. [http://www.bibliotheksverband.de/fileadmin/user\\_upload/Sektionen/sek4/Publikationen/WB2025\\_Endfassung\\_endg.pdf](http://www.bibliotheksverband.de/fileadmin/user_upload/Sektionen/sek4/Publikationen/WB2025_Endfassung_endg.pdf) (06.04.2018).

<sup>36</sup> Vgl.: FOLIO Project Charter: [https://docs.google.com/document/d/1Pc3KJ7VZISLe7yNG0DQLd\\_nX6ibKehcue3I41GXOSzU/edit#](https://docs.google.com/document/d/1Pc3KJ7VZISLe7yNG0DQLd_nX6ibKehcue3I41GXOSzU/edit#) (20.04.2018).

<sup>37</sup> Vgl.: OLE Community Charter: <https://docs.google.com/document/d/1xLjvUkCnHNNJMGSgojM4ukkoOIoW64rC9JjwpP51F4/edit#heading=h.yudfpayl4njp> (20.04.2018).

Die Verbundzentralen von GBV und hbz sehen in den sich abzeichnenden Entwicklungen rund um die FOLIO-Software ein großes Potential, Bibliotheken eine zusätzliche Alternative zu kommerziellen Produkten bieten zu können. Durch die jahrelange eigene, intensive Mitarbeit in allen Bereichen der Softwareentwicklung konnte die entsprechende Expertise aufgebaut werden.

Erste Ziele bei der Entwicklung eines konkreten Services sind:

- Analyse der Möglichkeiten und testweise Datenmigration der in den Verbänden aktuell eingesetzten Bibliothekssysteme nach FOLIO anhand von Beispielsbibliotheken
- Entwicklung von Migrationspfaden
- Umsetzung bestehender Datenflüsse, Verbundservices, Arbeitsabläufe auf der FOLIO-Plattform, z. B. Anbindung der Verbundkataloge, Fernleihe
- Gap Analysis – was fehlt für einen Produktivbetrieb?
- Entwicklung der fehlenden Funktionalitäten, bzw. Anbindungen an Drittsysteme in der Community oder in regionalen Konstellationen
- Aufbau eines Standard-Servicepakets FOLIO mit individualisierbaren Komponenten
- Pilotimplementierung und Weiterentwicklung

Ein weiterer Ansatz neben diesen großen Entwicklungslinien ist die schrittweise Anbindung von FOLIO-Modulen an das bestehende Softwareszenario. Hier wird vor allem an den Einsatz des neu zu entwickelnden FOLIO-ERM-Moduls in GBV-Bibliotheken gedacht.

Nach Einschätzung von GBV und hbz erlaubt die Entwicklung eines Open-Source-gestützten Bibliotheksmanagementsystems eine schrittweise und selbstbestimmte Ablösung bestehender Systeme unter Beibehaltung weiterhin gewünschter und erfolgreicher Verbundservices.

Die aktive Unterstützung des FOLIO-Projektes durch verschiedene kommerzielle Partner kann Risiken nicht ausschließen. Allerdings erzeugt dieses Engagement mit einer outcome-orientierten Zielsetzung einen entsprechenden Erfolgsdruck und sorgt für Einschätzbarkeit. Trotzdem muss die Community aus eigener Kraft sicherstellen, dass die Software auch ohne kommerzielle Treiber nachhaltig, effizient und zielgerichtet entwickelt wird und

Projekte nachnutzbar im Zeitplan fertiggestellt werden. Dabei wird die Sicherstellung des Rückflusses aller Entwicklungen als Open-Source Code auf dem Open Library Foundation GitHub essentiell sein. Darüber hinaus wird den insbesondere im Bereich SaaS geäußerten Anforderungen der Bibliotheken an Datenhoheit, Datensicherheit und Selbstbestimmtheit bei der Informationsversorgung Rechnung getragen, da diese bei dem beschriebenen Konzept im öffentlich-rechtlichen Bereich verbleiben können und auf bereits erprobte leistungsstarke, kollaborativ vorgehende Gemeinschaften aufgesetzt wird, die zukünftig international erweitert werden.

Das herausragendste Merkmal von FOLIO ist jedoch das Plattformkonzept. Neben der Migrationsmöglichkeit von aktuellen Bibliotheksmanagementsystemen auf ein neues System bietet die offene Schnittstellenarchitektur, eine aktuelle Softwaresuite, ein an Microservices angelehntes Technikkonzept und die Vereinbarkeit von Open-Source und kommerziellen Interessen eine echte, nachhaltige Weiterentwicklungsmöglichkeit bibliothekarischer Funktionalitäten und Services in einem innovativen Entwicklungskonzept selbstbestimmt „unter einem Dach“.

## Autoreninformationen



### Kirstin Kemner-Heek

Verbundzentrale des GBV (VZG)  
Abteilungsleitung Lokale Bibliothekssysteme  
Platz der Göttinger Sieben 1  
37073 Göttingen  
[kirstin.kemner@gbv.de](mailto:kirstin.kemner@gbv.de)  
[orcid.org/0000-0003-1957-3426](https://orcid.org/0000-0003-1957-3426)



### Maike Osters

Gruppenleitung Bibliotheksmanagementsysteme  
Hochschulbibliothekszentrum des Landes NRW (hbz)  
Jülicher Str. 6  
50674 Köln  
[osters@hbz-nrw.de](mailto:osters@hbz-nrw.de)  
[orcid.org/0000-0001-8495-6541](https://orcid.org/0000-0001-8495-6541)