INTEGRATION OF LIBRARY SERVICES WITH INTERNET OF THINGS TECHNOLOGIES

by Kyriakos Stefanidis & Giannis Tsakonas

The SELIDA framework is an integration layer of standardized services that takes an Internet-of-Things approach for item traceability in the library setting. The aim of the framework is to provide tracing of RFID tagged physical items among or within various libraries. Using SELIDA we are able to integrate typical library services—such as checking in or out items at different libraries with different Integrated Library Systems—without requiring substantial changes, code-wise, in their structural parts. To do so, we employ the Object Naming Service mechanism that allows us to retrieve and process information from the Electronic Product Code of an item and its associated services through the use of distributed mapping servers. We present two use case scenarios involving the Koha open source ILS and we briefly discuss the potential of this framework in supporting bibliographic Linked Data.

Introduction

Libraries are trying to identify potential applications for Internet-of-Things (IoT) technologies and a recent survey by OCLC (2015) highlighted that the anticipated uses of IoT are mostly related to intelligent uses of space and facilities. The same survey revealed that the most familiar IoT services to librarians were those designed for inventory purposes. Such services would require the employment of Radio Frequency Identification (RFID) tags as aids for increased visibility and unique identification. RFID have been envisaged as the appropriate technologies to fulfill the promise of IoT in libraries. Fortune (2012) describes a future state where books are tagged with RFID and “sensors are placed on the shelves to detect the removal of any item for consultation” making the shelves “active”.

In doing so, libraries must be equipped with the right technology infrastructure, such as sensors, readers, services and software. One of the main technical challenges is that, apart from the lack of appropriate technologies to coordinate these components, any intervention has to be easily adjusted to key software infrastructure, such as the various Integrated Library Systems (ILS). The existing technologies in libraries cannot be ignored or skipped, but on they should be able to easily link to these solutions. In our case this is achieved with the SELIDA framework, on which the main part of this article is focused.

The SELIDA framework has been developed in the frame of the self-titled project. Its name stands for the Greek equivalent for the acronym “Printed Material Management Using Radio Frequency Identification Technology” (“selida” in Greek means “page”). SELIDA is a public-private sector collaboration project, with the Library and Information Center at the University of Patras as host of the pilot implementation. The ILS used in this prototype is the open source Koha.

A second section of the paper is dedicated to the exploration of the Electronic Product Code (EPC) tags’ potential to store information about library holdings. As the current Linked Data efforts in the bibliographic ecology have not substantially covered the issue of holdings information, we can exploit technologies from other domains. One of the main principles in the Linked Data domain is to reuse already operating or standardized components to avoid ambiguity (Bizer & Heath, 2011) and to this end EPCs can be part of the linking of bibliographic records with holdings information.
Related technologies

The key component of the SELIDA framework is the use of RFID tags as an aid for identification of physical items, and EPC as the underlying building block for standardized tracking services. RFID tags are now commonplace in Libraries, with many advantages over other identification technologies, such as barcodes or QR-codes—namely in bulk detection, reading distance, alignment, capacity and so on. RFID tags are also preferred for supporting real time traceability and for being intrinsically connected to EPCs. According to a recent survey of the GS1 US—the US chapter of GS Global, an organization that promotes the adoption of its standards in industry—retail market stakeholders such as manufacturers and sellers are rapidly adopting RFID with EPC to increase the visibility of their products (GS1 US, 2015). Thus, the use of EPC Global, the GS1 suite of standards and specifications, enables the visibility and traceability of physical units. The EPC standard is mostly used in the supply chain sector and in our case it is adjusted to the physical items of a library’s collections, such as books, discs, etc. The use of EPCs together with the Object Naming Services (ONS) enables the circulation of uniquely tagged information; the items are notated by the tagging services of EPC Global and the information is exchanged with ONS.

An EPC can be written on a physical RFID tag and the number of encoded digits is determined by its type (in our case 24 decimal digits). It encodes a Serialized Global Trade Item Number (SGTIN), which is a Global Trade Item Number (GTIN) that is expected to be the same for a set of items, plus a serialization number for each physical unit. The EPC is split into different, fixed-length sections, with each section providing different information. The first section is the header that reports on the coding scheme; the second is the company/institution prefix, which in our case represents a library; the third section defines the item reference; and the final section refers to its unique serial number. When an RFID reader captures an EPC in binary format, it transmits it to a middleware layer and transforms it in a URI format. Figure 1 presents the sections of a typical EPC as a URI.

![EPC in URI format](http://journal.code4lib.org/articles/10897)

**Figure 1. An example of an EPC in URI format**

Lastly, the Object Naming Services are provided by a resolver, which translates a URI address to a domain name. By working in a way similar to DNS, the ONS resolver splits the binary format to different sections, each of which corresponds to a different server that holds the related information and services about its own section and a pointer to the next server in the URI sequence.

Architecture

We deploy a three-layer architecture, outlined from bottom to top:

1. The bottom layer is the integration layer that manages seamlessly all the processes relevant to the library services workflows. By seamless we mean that the actual services (i.e. the source code) of the ILS are not changed and the production system remains intact. Therefore any change, such as a version update, in either part does not affect the other. The integration layer is injected upon page load as a JavaScript file. In our
prototype, we use the mod_substitute directive on the ILS’s dedicated Apache web server. Each time a module/page of interest is requested by the server (i.e., checking out an item), the web server adds a script tag that loads the additional functionality. This layer, in the form of a JavaScript application module (henceforth the SELIDA module), adds the required user interface elements and handles all the necessary web service requests.

2. The middleware layer receives, analyses, processes and propagates the data collected by the RFID readers to the ILS. The middleware hides the complexity of the actual RFID infrastructure and provides only the information about the workflow events. The middleware is agnostic of how the data is handled afterwards.

3. The upper layer, called the services layer, provides secure access to the ONS infrastructure which consists of a mapping server and an ONS resolver. The mapping server manages all provenance data which are relevant to a single physical item, as well as the mapping between an EPC and its metadata. These data can contain information regarding item transactions within locations controlled by the owning library or any other interconnected library, as long as all those are part of the SELIDA infrastructure. Furthermore, through the services layer, the system accesses the ONS Resolver, which allocates specific services to specific types of users, such as inventory services for collection development librarians or history services for circulation librarians. This is accomplished by a web service layer that functions on top of the ONS resolver, which provides authenticated users with the capability to query the whole ONS infrastructure and discover the management services for any given EPC and point them to the correct mapping server.

![Figure 2. The layers of the SELIDA framework architecture](image-url)
**Implementation**

In our prototype version, the framework supports two types of services: first is a subset of basic library services, namely checking-out, checking-in, adding, and deleting an item. Secondly, the framework supports some item-related services, namely retrieving information about the history of an item (location of check-ins and check-outs across the interconnected libraries), as well as searching for the physical location of an item within a library. We describe one implementation case for each type of service.

**Checking-out**

When a circulation librarian navigates to the check-out module of the ILS by requesting the respective URL, the SELIDA module starts executing upon page load and adds the button “Scan” next to the button “Check-Out” of the Koha interface. When the user presses the button “Scan”, a web service request is launched from the SELIDA module, which starts up the RFID reader via the middleware services. The results, in the form of the items’ titles and codes that the reader captured, are sent back to the SELIDA module. The SELIDA module pops up a window that informs the user of these results. After this presentation, the regular check-out workflow resumes by sending the required POST requests to the Koha web server. Since Koha does not support (at the moment) multiple checkouts, the required multiple requests are sent via AJAX and the results of these requests (namely the errors) are gathered by the SELIDA module for subsequent presentation to the user. When the check-out process ends and the web server responds with the next web page, the SELIDA module sends a second web service request to the middleware indicating that the check-out is complete.

**History services**

History services are expected to be used among interconnected libraries. Information about items should adhere to the same API and should be easily stored by the networking parties for reviewing the required data. Therefore a lightweight data exchange mechanism is required and as such the SELIDA module services have been designed following the RESTful architectural style using **JavaScript Object Notation (JSON)**. In our framework, historical data can include a list of transactions only based on the dates of interaction with the middleware, together with a geospatial representation of the location of the respective transaction, i.e. the library where this transaction takes place. Any request for item status, like checking in items, can contain multiple EPC tags at a time, as well as error warnings (0 in case of no error) and relevant metadata for the identified books, such as notes on the purpose of transfer, the owning and/or the target library, etc. The following code snippets in JSON exhibit the request and response for history services.

**Request**

```json
1  
2  "EPCList": [
3   {"EPC": "961012345678910012345678"},
4   {"EPC": "961543210109876543210101"}
5  ]
6 }
```

http://journal.code4lib.org/articles/10897
Can Linked Data and IoT be Real in the Library?

The framework could be a promising start for the extension of linked data models of the bibliographic ecosystem to enable the accommodation of information about holdings. Whereas the Linked Data solutions are rapidly progressing in the library sector, thus creating a conceptual network of knowledge resources, IoT proposals are relatively limited and as previously stated have reached a dead end in linking concepts to holdings. Fast evolving linked data frameworks, such as BibFrame, are currently missing mechanisms for global and unique visibility, identification and traceability of items. For instance, BibFrame stores information about the holdings of an instance in annotation fields, but uses ambiguous and context-dependent information, such as the instance’s call number, which might differ from one library to another. The following JSON example shows how minimal, but concise EPC tagged information can be embedded in an annotation field of a BibFrame record:

```json
{
  "type": "HeldItem",
  "id": "0123456789012helditem1",
  "bf-id": "0123456789012helditem1",
  "uri": "urn:epc:id:sgtin:96.1.012345.678.910012345678",
  "urn:epc:id:sgtin:96.1.543210.109.876543210101",
  "bf-holdingFor": "9600215006instance1"
}
```
By parsing the EPC of an item in URI format, one could identify a specific physical unit from the collections of a library and retrieve information about its status and location. Thus, there can be an effective and dynamic connection between different “things”—such as works and entities, which have demonstrated that they can be in URI format—and actual things, such as physical items.

Discussion
The SELIDA framework demonstrates the potential of using already globally working standards and protocols with existing technologies to increase the visibility of physical items with minimal cost and disruption. It also underlines that there is a lot to be done in the future. The penetration of RFID technologies is still slow, especially in academic libraries, where the large number of volumes requires significant investment, while the EPC encoded tags are relatively unknown. Furthermore, there are other kinds of investments and efforts, notably in the area of GS1 company prefixes, which would require the cooperation of authorities to assign global prefixes to libraries. Finally, with the lack of wide application of the STGIN in libraries, there have to be firm decisions about the form of the sections of the 24 digit code. For instance, if a library desires to migrate from an existing item bar-coding scheme, then the number of available item reference digits is limited and will have to refer to another item relation, such as a category (e.g., books or discs). However, if a more global scheme is required, then the item reference will have to be occupied by a worldwide common scheme, like an edition’s barcode, and the rest of the digits used to identify the serialized number of copies of this edition in the library’s collection.

In this paper we described the architecture of the SELIDA framework and presented two typical implementation cases. We also raised the issue of using parts of the framework to merge Linked Data with the IoT efforts in the library in a more effective fashion. The services of the SELIDA framework alone can be appealing to networks of libraries, such as cooperating libraries or branch structures that circulate items among them. It is encouraging that IoT technologies are maturing and the potential they hold in identifying and tracing physical objects allow more confident visions for a global library catalog.

Appendix A: SELIDA Exhibition Video

Acknowledgements
The SELIDA project (#09SYN-72-646) was financially supported by the General Secretariat for Research and Technology (GSRT) of the Hellenic Ministry of Development in the frame of “Cooperation” 2009 Call.

About the Authors
Kyriakos Stefanidis holds a PhD in Computer Science and a BSc in Computer Engineering from University of Patras (Greece). He is a contracted researcher in University of Patras and ATHENA RC / Industrial Systems Institute. Since 2001 he is involved in a number of research and development projects both European and National. His research interests are
mainly focused on the field of computer and network security, cyber-physical security and high performance computing.

Giannis Tsakonas is Acting Library Director in Library & Information Centre, University of Patras. He holds a BA in Librarianship from the Department of Archives and Library Sciences, Ionian University, Greece and a PhD in Information Science from the same Department. More info at his website www.gtsak.info and/or at his Twitter feed @gtsakonas.

References


