

# On the Secure Sharing of Legacy Data

Douglas Russell, David Power, Mark Slaymaker and Andrew Simpson

## Service-oriented Interoperability Framework (sif) Middleware

### Drivers

- Secure sharing and aggregation of data
- Support for legacy systems
- Raising level of abstraction for application developers

### Philosophy

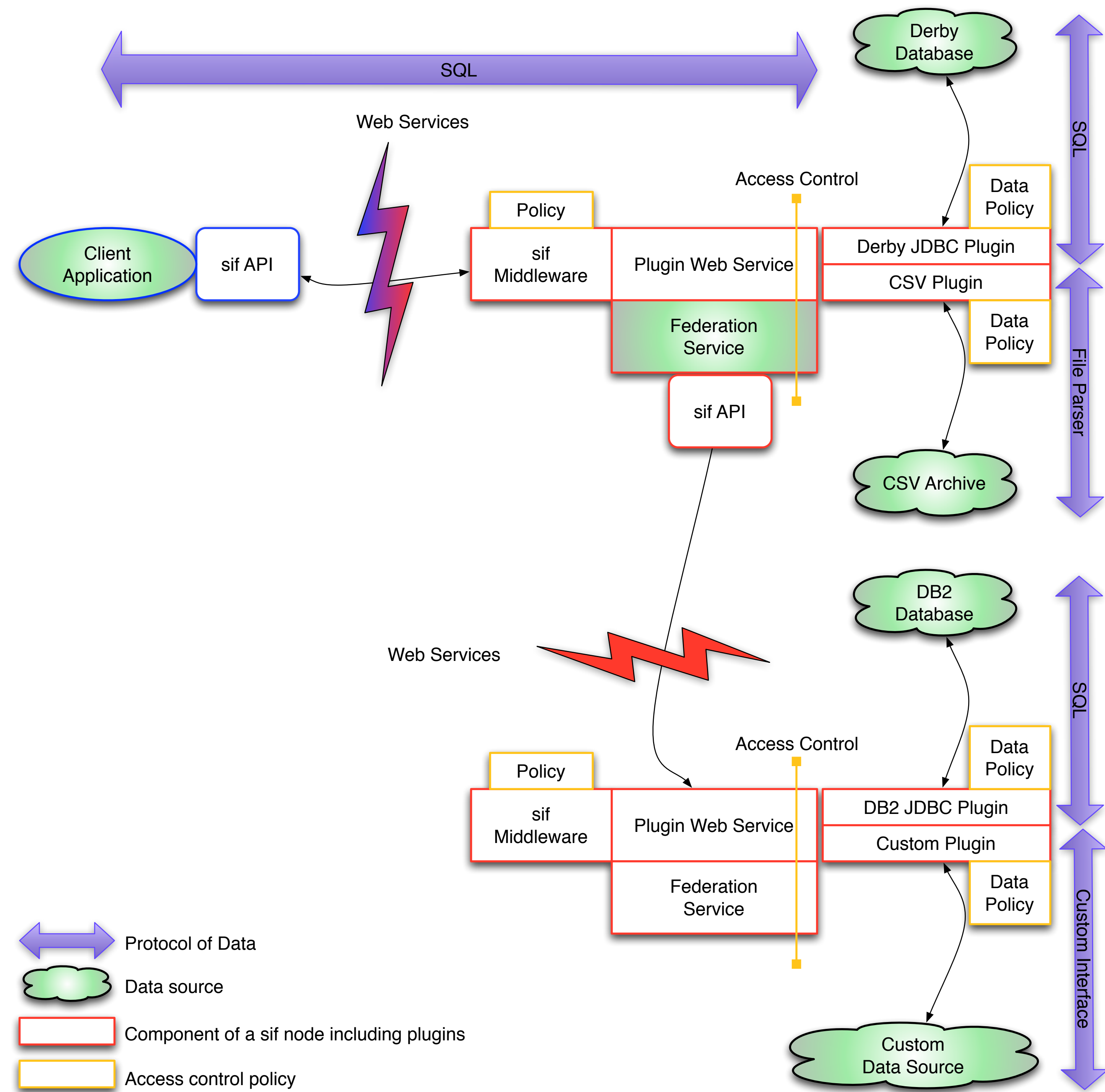
- Data-agnosticism
- Fine-grained access control
- Low barriers to entry for both data source providers and application developers

### Approach

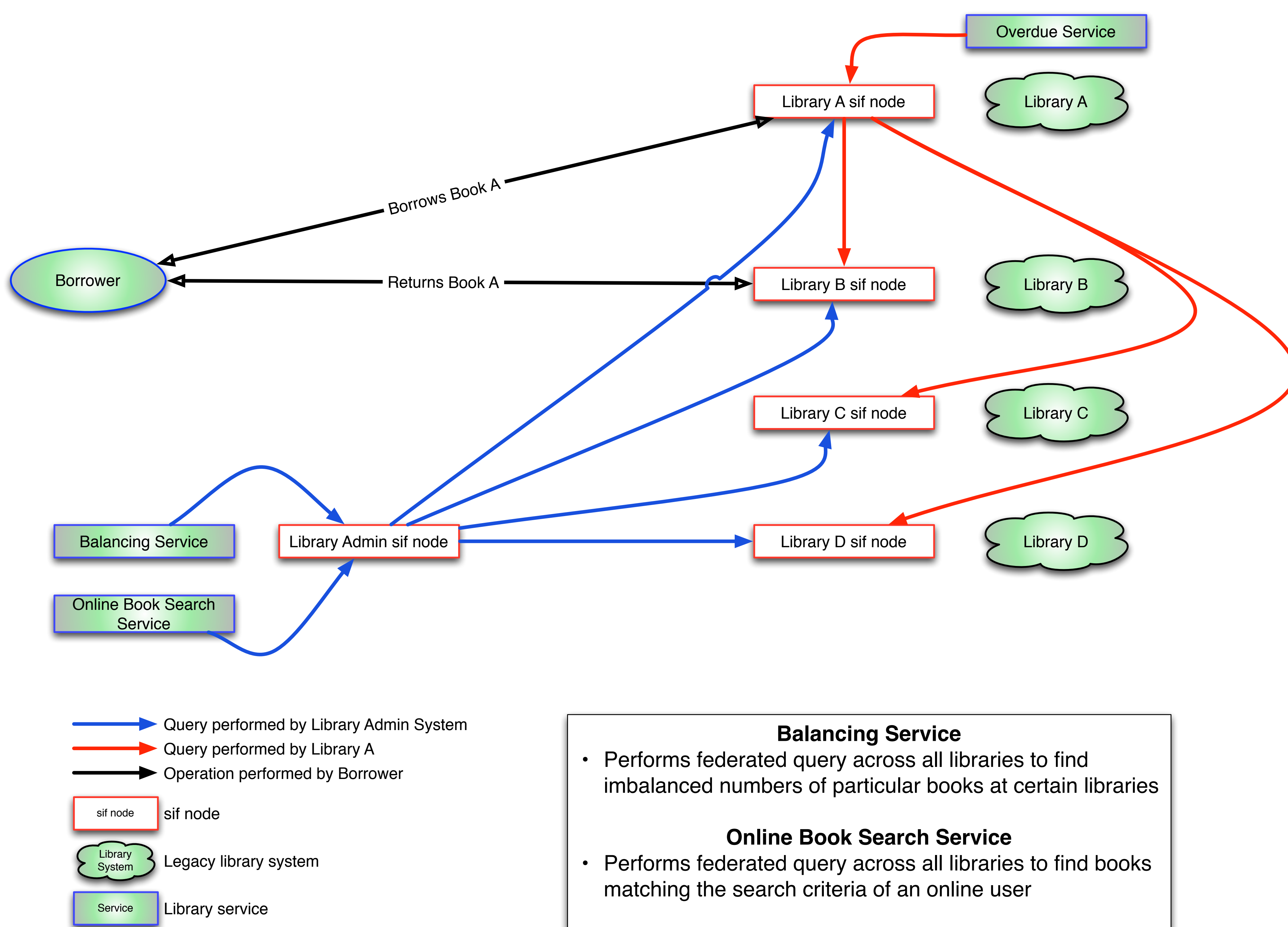
- Plugins are used to interface between the disparate underlying data sources and the standard SQL data model
- Through sif, a user or application developer can query for known (and accessible according to the security policy) data sources and return their data schemas
- End-to-end security with customisable security policies and audit
- Federation service across multiple instantiations of the middleware and multiple data sources
- Easy to use Application Programming Interface
- Based on Web Services for maximum cross-platform compatibility and programming language support

### Applications

- Training application for radiologists
- Self-management application for long-term medical conditions
- Image analysis for cancer care



## A Motivating Example



### Balancing Service

- Performs federated query across all libraries to find imbalanced numbers of particular books at certain libraries

### Online Book Search Service

- Performs federated query across all libraries to find books matching the search criteria of an online user

### Overdue Service

- Performs a federated query across all libraries for books deemed overdue according to the local library database in case they have been returned to a different library in the system

In this example, we demonstrate how sif could be applied to the problem of creating a unified library system out of four previously separate libraries with their own custom systems. Each library system is interfaced with a sif node using a plugin written specifically for that system (or a stock plugin if the system happens to be one for which we already have a plugin such as an SQL database with a JDBC connector).

When library A lends a book to a borrower, that is recorded in library A's local system; however in the new unified library system it may be possible to return a book to another library, say library B. When the book is about to become overdue according to the records of library A, they must now interrogate the systems at the other libraries to see if the book has been returned there. This is accomplished by sending a federated query to library A's local sif node, querying for that specific book. Results will be returned from the other libraries in the system to determine if it has been returned elsewhere in the system.

Now that the libraries have become inter-linked, it is likely that some designated administrator of the libraries will need to perform functions such as redistributing books when the balance of books in the libraries becomes imbalanced. This administrator will want to perform a federated query looking for multiple instances of certain books at one library and none at another. This data will likely be cross-referenced with data from each library about the type of books they hold. If Library C is a specialist science library then all fiction books which end up there will be redistributed to other libraries and similarly, certain specialist scientific books will always be repatriated there.

It would also be possible to easily create an online search service which could perform federated queries across the libraries at a user's request so they can find out if a book exists in the library system and at which library it is currently held. The information available to other libraries such as the identity of the current borrower would not be available to the user and this could be easily restricted by defining that the access rights of the online book search service are limited in the security policy at either each library system's sif node or at the library admin node.

### Contact Information:

Andrew Simpson  
 Oxford University Computing Laboratory  
 Wolfson Building, Parks Road, Oxford OX1 3QD, United Kingdom  
 andrew.simpson@comlab.ox.ac.uk / +44 (0)1865 83514

