



Final Report

IDMAPS

**Institutional Data Management for Personalisation
and Syndication**

Janet Wheeler, Project Manager

April 2010

Table of Contents

Acknowledgements	3
Executive Summary	3
Background	4
Aim and Objectives	4
Methodology	4
Implementation	5
<i>In the beginning</i>	5
<i>The data audit</i>	6
<i>The data architecture</i>	7
<i>Getting personal</i>	7
Outputs	9
Outcomes	10
Conclusions	11
Implications	11
References	11
Appendix A: personalisation in more detail	12
<i>Module information</i>	13
<i>Defined RSS standard</i>	13
Appendix B: the data flow makeover	15

Acknowledgements

This project was funded under the JISC Institutional Innovations Programme. It represents a collaboration between Information Systems and Services (ISS) and Learning Technology for Medical Sciences (LTMS) at Newcastle University. The project gratefully acknowledges the support of JISC and our parent institution.

We would also like to acknowledge the contribution of the many people who have been involved in making the project a success.

The project team:

Rob Booth, Alan Cecchini, Dave Churchley, Gary Davison, Jon Dowland, Clare Johnson, Jan Hesselberth, Andrew Martin, John Moss, Jonathan Noble, Cal Racey, Sunil Rodger, John Snowdon, Dave Teasdale, Paul Thompson, Steve Williams, Dave Wolfendale

The supporting cast:

Chris Franks, Paul Haldane, Geoff Hammond, John Hills, Richie James, Tony McDonald, Jo Robinson, Dave Sharples, Rog Sillito, Mike Stephenson, Carol Summerside, Lawrence Thompson, Andrew Vickers.

For JISC:

Andy Dyson, Isobel Falconer, Lawrie Phipps, George Roberts, Mitul Shukla

Last, but not least, we would like to acknowledge the support given to us by Talend open data solutions.

Executive Summary

The aim of this project was to produce an exemplar institutional data architecture and to evaluate its benefits in the context of Web 2.0 systems. The overall approach taken was to conduct an audit of institutional data and use this to inform the specification and implementation of a revised data architecture. The implementation of standard interfaces to data combined with student information supplied from the new data architecture would enable personalised information to be delivered to an authenticated individual.

This approach has proved to be successful. The project has produced an Institutional Data Feed Service which is based upon a new data architecture and which holds definitive records of institutional data flows, allowing the coordination of data from disparate systems into a cohesive whole and promoting the rapid development of new feeds.

Following on from this, a student homepage which displays personalised information based on modules being taken by the logged in user has been implemented. This approach found favour with student focus groups when it was presented to them.

The project has shown the benefits of implementing a flexible information architecture of core user data which can be adapted and extended to meet specialised application needs. It has also shown that deployment of standardised interfaces to data in combination with Shibboleth and a groups management system can successfully be used to display personalised information. Data quality issues have been highlighted but the work has also provided a means of mitigating them by exposing data and offering the potential for data feedback loops.

Background

The origins of IDMAPS lie in **iamsect**, a JISC Core Middleware Development project investigating the use of Shibboleth for inter-institutional authorisation (2004-6). In order to make authorisation decisions, access to institutional data is needed; this proved to be rather more difficult than might have been anticipated – there was no catalogue of available data, existing data feeds had been implemented ad hoc in response to specific needs, and there was no procedure for requesting data.

Some three years later it had become clear that the distribution of reliable data was a major issue facing the University and that data distribution mechanisms that had grown organically over time were no longer adequate for the purpose of integrating different systems across teaching, learning, research and administration into a cohesive user experience for staff and students. This is the problem that the project set out to solve.

Data has become an integral part of the individual's life, resulting in an increasing demand for it to be reused, redeployed and accessed in a variety of formats. The required system integrations in turn require timely and reliable data. The display of personalised information that illustrates the benefits of improved data architecture builds in part on the work done by GFIVO, a JISC e-infrastructure project investigating the use of Shibboleth in conjunction with the Grouper group management toolkit (2007-9).

Aim and Objectives

The overall aim was to produce an exemplar institutional data architecture with associated data access policies and interfaces and to evaluate their benefits in the context of Web 2.0 systems. Specific objectives were to:

- investigate existing structures, requirements, and the fitness for purpose of existing solutions;
- specify a flexible information architecture of core user data which can be adapted and extended to meet specialised application needs;
- implement data infrastructure and supporting systems;
- specify and deploy interfaces to enable data exchange and reuse across a range of systems;
- demonstrate, document and disseminate all appropriate policies, models and findings.

Aim and objectives did not change during the course of the project.

Methodology

Reflecting the aim and objectives, the project consisted of three main phases:

- **Conduct a data audit**
The methodology was to design a template to capture the requirements and interdependencies of existing data flows. The owners of applications consuming data were individually interviewed, resulting in a diagrammatic representation of their data flows and a completed template.
- **Specify and implement data architecture**
The methodology was to use the audit results to inform a requirements analysis and to reconcile this with the characteristics and features of available ETL (Extract, Transform, Load) software.
- **Leverage the data architecture to provide personalised information**
The methodology was to use the results of the data audit to identify data that would be

useful to students in a personalised form and to implement standard interfaces to that data utilising RSS or SOAP as appropriate. Combining this with student information supplied from the new data architecture via Shibboleth and Grouper would enable personalised information to be delivered to an authenticated individual.

Implementation

In the beginning

In addition to the business of setting up the project web site and writing and submitting the formal project plan, the project team needed to be assembled.

Because of the potentially wide-ranging implications of the project outcomes, it was vital that all stakeholders internal to the project partners were represented on the team and that representation was at an appropriate level of seniority. Thus it was that the final assemblage consisted of 14 people encompassing Business Applications, Infrastructure Services, Academic Services Support and Development, Learning Technology for Medical Sciences, Information Applications and Delivery, Middleware and the ISS senior management team. Just under half of these were at team manager level or above; the remainder were senior technical staff.

To this team were added 4 project officers. We had two very successful recruitments for a communications officer and a data specialist, and co-opted two further members of staff internally.

The eventual size of the team posed some interesting organisational problems, not least of which was the lack of a big enough meeting room. Initially meetings of the whole team were held every two weeks but once workpackage 1 (data audit) was almost complete and we had a clearer view of the remaining work, the project was split into two halves: specification and implementation of the data architecture led by Cal Racey, and Personalisation and Data Integration, led by Gary Davison and Paul Thompson. A monthly management meeting achieved coordination.

It should be emphasised that every member of the team has contributed to the project outcomes, and that its size is a reflection of the scope of the undertaking.

Stakeholder engagement

Once we had the project team, wider engagement was vital in order to achieve an appropriate level of buy-in.

Our first move was to brief stakeholders and potential stakeholders in the wider University. An invitation-only briefing event was held for senior representatives from areas including the Library, Business Development Directorate, Student Progress, Quality in Teaching & Learning, HR, LTMS, Computing Science, Netskills, Alumni, Audit and the Executive Office. This was followed up by a discussion of the project and its aims, objectives and potential outcomes at the University Information Strategy Committee (chaired by the Registrar).

The briefing was repeated in a more technical form for members of `ISS staff; we have also presented the project as part of the Computing Science seminar series, and briefed School Computing Officers.

We learned two important lessons in the course of our stakeholder engagement.

- Senior University Management don't do detail

The lesson is: take out the technical detail. Then look at what you've got left and take out the technical detail. Ensure that you are able to present the project in terms of risks and benefits, where the benefits are tangible enhancements to the business of the university. A rationalised and well-governed data architecture is really only of interest to the computing service. The University is interested in potential improvements to data quality, increased student satisfaction due to the delivery of personalised information, reduction of risk in the area of information security, and increased ability to support new applications.

- The developers on the ground are just as important.

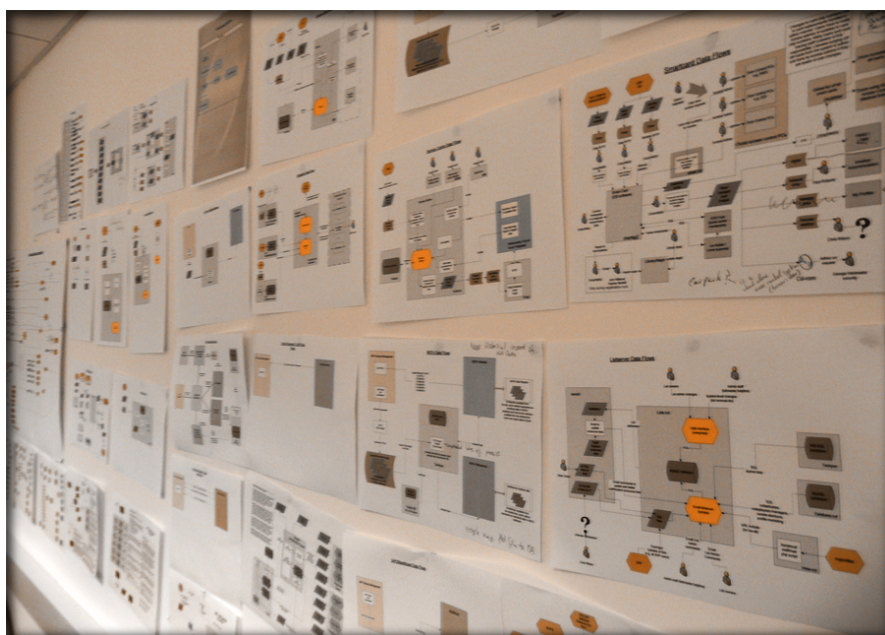
With 20/20 hindsight we now realise that engagement with application developers could and should have been more wide-ranging. Fortunately there is considerable scope for remedying this in the process of embedding the project outputs.

The data audit

The data integration template used is based on work done by the OpenEAI foundation. Its purpose is twofold: to document existing institutional data flows, and to be used by anyone requesting a new data feed. The information gathered by the template is detailed in the project output *Guide to Completing the Data Integration Template*. The template underwent significant evolution in the course of the audit as we tried to strike a balance between the information required and ease of use. Change control is established through the use of a Subversion repository to contain completed templates.

However, merely requesting that application owners completed a quite complex document was never going to produce speedy results. The strategy was therefore to interview them and to produce diagrams of data flows as a result of those interviews, completing the template at a later date.

When we started, we thought that we already had a good overview of existing data flows, and that this stage of the project would not take too long as it was only a case of formalising what we already knew. We were wrong. By the time that the audit was finished we were able to paper a wall with over 30 data flow diagrams, some of which were very complex and some of which were suspiciously similar to others.



In short the data audit gave us clarity on:

- what data is collected and from which source(s);
- where and how recorded data is stored;
- what the data is used for, and how it passes both between systems and to data consumers;
- who is responsible for the data at both an operational and a strategic level.

The data architecture

Once we finally had a picture of what we were dealing with, a requirements analysis was conducted in order to find a suitable tool on which to base the new data architecture. This was principally based on work done by LTMS, who have possibly the largest and most complex data requirements in the University as a result of their remit to support teaching and learning across Medical Sciences. It is available as the project output *Requirements Analysis*. In the course of our investigations we also published an ad hoc snapshot of the data architecture tools marketplace.

After extensive investigations, Talend was selected as our preferred ETL (Extract, Transform, Load) tool. How it fulfils the requirements is documented in detail in the project output *Report on Data Infrastructure setup*. It is, however, worth pointing out a couple of its broad advantages.

- It follows the model of open source software that has optional paid support and additional features. There is no financial lock-in.
- It generates Java code so that anything built with it can be used whether Talend itself is retained or not. There is no technical lock-in.

The data audit gave us a clear view of what data was available, how it was being used and who the data *custodians* are. This has been embodied in a data dictionary which presents available data to potential consumers.

The technical implementation of Talend proceeded with no hitches. Once this had been done, it was combined with the data integration template, the data dictionary and a set of procedures to create an Institutional Data Feed Service (IDFS) which was launched in October 2009 – www.ncl.ac.uk/iss/services/data-service/ IDFS is described in more detail in the project output *Specification of Data Architecture*.

Reimplementation of existing data feeds and new data feeds is ongoing, with multiple feeds having been implemented using the new IDFS service. The concentration of data flows through one service has helped to uncover institutional data quality issues. While current data is good enough to be used, a requirement to improve institutional data accuracy and reliability has been uncovered.

A final piece of work (which is outwith the scope of the project) will be to attempt to define the owners (as opposed to the custodians) of data sources.

Getting personal

This phase of the project was to use the data audit and infrastructure to make personalised information available to students. We chose to use academic data specific to a student's module choice that is held across various academic systems. The same principles might apply to more private or sensitive data but it makes sense to use a less controversial source, and one which we can control and completely validate.

The vehicle chosen to display personalised information was the University student home page (my.ncl.ac.uk/students/), and we gratefully acknowledge the cooperation of the steering group in permitting this. The personalised module-related information that is displayed is:

- reading lists;
- examination timetable;
- past examination papers;
- announcements from the in-house NESS course management system (which is also used as a VLE by one School);
- announcements from the Blackboard VLE;
- information from the Medical School's Networked Learning Environment.

We had intended to provide timetable information in addition to the above but were stymied by the late delivery of a new version of the University's timetabling software (Syllabus +) that would allow this. Implementation is now in progress but will not be completed before the project end.

Access to the student home page is authenticated by a combination of Shibboleth and SPNEGO (Simple and Protected GSSAPI Negotiation Mechanism), which automatically authenticates the user if they are already logged on to the campus managed desktop. This technical enhancement has allowed 10,000 logins a day to be achieved without requiring further user input. Using Shibboleth in conjunction with the Grouper group management system, which receives its data from IDFS, allows a basic identity to be supplemented with a range of metadata attributes such as email address, name, faculty, school, year of study and modules being taken; these are then used to retrieve and display relevant content via RSS or SOAP from a range of systems.

The system is described in more technical detail in Appendix A. Once it was running, we asked student focus groups what they thought of it and its potential usefulness. In summary the opinion of the focus groups was as follows.

- Personalised content was considered very useful. They would definitely make use of such a system, as it centralises information to one place in a convenient and easy-to-use manner. It was viewed as a major improvement on the current static homepage and worth the effort of logging in to access it from off-campus. They also wanted to know when it would be generally available, and were keen for this to happen soon.
- Improved visibility of data was welcomed. A feedback mechanism to allow inaccurate data to be corrected is needed.
- The layout of the data as presented on the page is sensible considering the relatively large amount of data displayed.
- Single signon throughout is essential. The fact that some source systems for data required an additional log-in was viewed as a weakness.

A significant issue that has come to light in the course of this part of the project is that inconsistent use of systems such as Blackboard by teaching staff – for example, where and how teaching materials are stored in the system – means that we have gone about as far as we can in terms of the personalised information we can extract and display with respect to student courses. Similarly, there is no University requirement for teaching materials to be

stored in any VLE, and so a significant fraction is in home-grown systems or static web pages of which there is no central record.

There are also two more technical issues to be tackled.

- Provision of a data feedback loop

Our current thinking is to provide a link from the home page to a partly pre-populated form that will allow for the correction of personal data. The form will submit an incident report to our helpdesk software and will be automatically assigned to the data custodian for action. Although cumbersome, this should be effective – we feel that providing a totally automated mechanism would be both difficult and somewhat dangerous, and that a mediated update is a good first step to being able to contemplate automated updates in the future.

- Single signon access to systems linked from the home page

We are investigating N-tier approaches to authentication to allow advanced secure integration. The GRAND project, recently funded by JISC under the Access and Identity Management Programme, will try to address this requirement.

Finally, it has become clear that a governance framework as regards the provision of web services interfaces to systems is required.

Outputs

The project has produced two major outputs, an Institutional Data Feeds Service (IDFS) and a personalised student home page.

- By holding definitive records of previously implemented institutional data flows, the IDFS allows the coordination of data from disparate systems into a cohesive whole, promotes rapid development of new feeds and enables better prioritisation of development needs. Features include:
 - secure, consistent, accurate and timely bulk institutional data feeds;
 - single, reliable source of data;
 - data dictionary presenting what data is available;
 - fully documented, extensible, standards-based architecture;
 - data feeds tailored to the developer's requirements;
 - documented records of data feeds to help with Data Protection and Freedom of Information compliance.
- The student homepage displays personalised information based on modules being taken by the logged in user (see Appendix A for a picture). Metadata attributes supplied to Shibboleth/Grouper by IDFS are used to retrieve and display relevant content personal to the logged in user from a variety of academic systems.

Material published in the course of producing the above outputs is as follows.

- Data audit and planning:
 - *Guidelines on Performing an Institutional Data Audit*
 - Data Integration Template
 - *Guide to Completing the Data Integration Template*

- Data architecture
 - *Requirements Analysis*
 - *Specification of Data Architecture*
 - *Report on Data Infrastructure Setup*
 - *Brief review of the ETL marketplace*
- Data Feed Service and Policy Framework
 - *Considerations for Data Feed Governance*
 - *IDFS Example Service Definition*
 - *IDFS Example Policy*
 - *Case study: People Search*
- To provide an insight into how Newcastle University is using Talend, a series of short video clips has been produced.
 - *Module Provisioning into Grouper* demonstrates job creation using Talend to provision module information into Grouper. It gives an overview of key Talend components, explains how to use them to create a job, and shows the output of the job.
 - *Exporting Data from Grouper using Talend* illustrates using Grouper to provide data for Syllabus Plus room bookings.

All published outputs are available from research.ncl.ac.uk/idmaps/resources.php or research.ncl.ac.uk/idmaps/videos.php

Outcomes

The project has fulfilled its objectives in producing an exemplar institutional data architecture and exploring the possibilities that this brings to delivering personalised information. This work will be of use to any institution wishing to provide secure, consistent, accurate and timely bulk data feeds from a single reliable source utilising an extensible standards-based architecture.

The personalisation work that builds on the data architecture has thrown up both possibilities and further issues.

- Personalised information is a welcome addition to the student experience.
- Any application that is able to consume RSS can be used to display personalised information; this opens up the possibility of delivery to mobile devices and for the display of information from one learning environment in another.
- Data quality issues have been thrown into stark relief. The potential of vehicles such as the student home page as a source of a data quality feedback loop needs to be realised.
- The business case for single signon across all systems has been reinforced.
- A barrier to integrating information from multiple VLEs is the inconsistent way in which information is stored within them by their users. A lack of governance, or even information, regarding the location of teaching materials outside VLEs is also a factor. This is a potentially sensitive issue that will not easily be resolved.
- Attention needs to be paid to the governance of web services interfaces to applications.

Conclusions

- Auditing institutional data and if necessary re-implementing data architecture is an essential prerequisite for the provision of personalised information and improvement of data quality with concomitant enhancements to the staff and student experience and general efficiency.
- Deployment of standardised RSS interfaces in combination with Shibboleth and a groups management system can successfully be used to display personalised information.

Implications

Any institution that has deployed Shibboleth and has data available in a defined and structured manner can make personalised information available to its members as described here.

References

Resources produced by the G-FIV-O project on the use of Grouper and set up of SPNEGO:

<http://gfivo.ncl.ac.uk/resources.php>

GRAND (GRanularity, Audit, N-tier, and Delegation): <http://research.ncl.ac.uk/grand/>

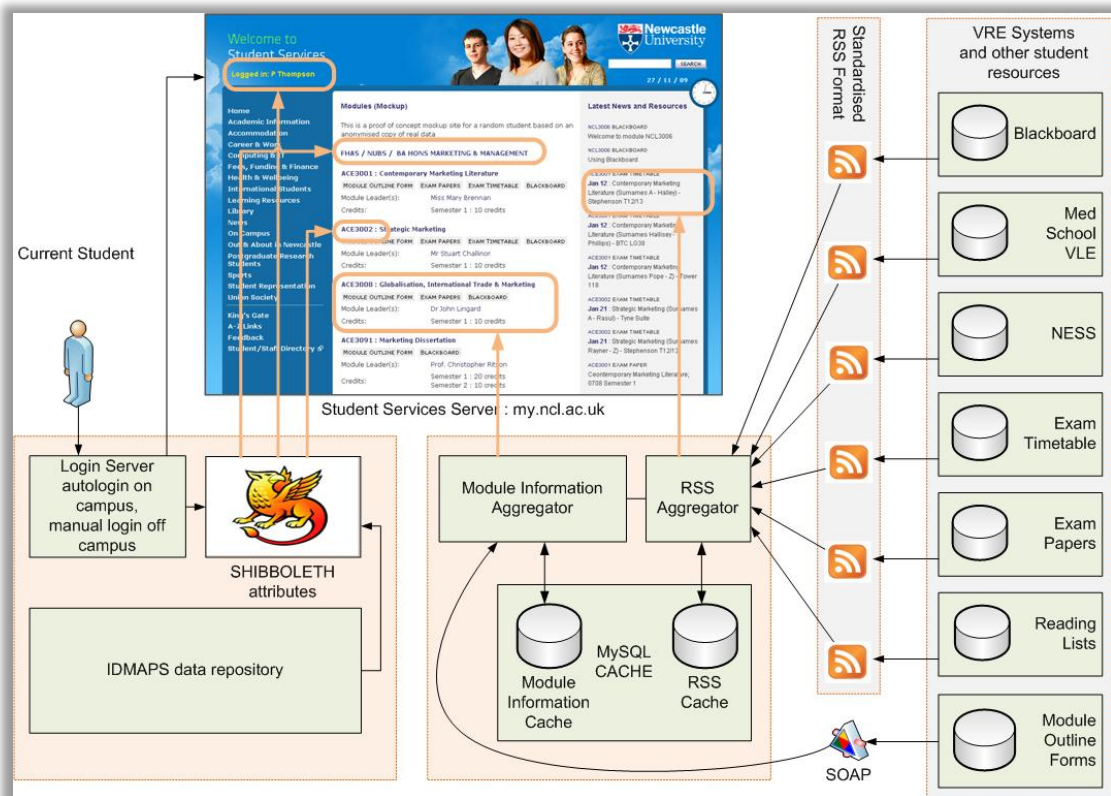
OpenEAI Foundation: <http://www.openeai.org/>

Appendix A: personalisation in more detail

The system was built to be displayed within the context of the existing Student Homepage, which set the parameters for the overall look and feel of the page. There are two content areas:

- a central pane with summary information for the student's modules, and links to the relevant resource page for each module within a number of systems;
- a news pane which displays individual news items collated from various systems.

SOAP and RSS were chosen as technical standards as they are simple and well-understood. This permits a range of future systems to be integrated relatively easily, with minimal custom development work. The diagram below provides an overview of the data flows.



When a user on campus using Internet Explorer arrives at the homepage, they are automatically logged on. If they're off campus or using another browser, they must click the log in button. Shibboleth makes available to applications relevant rich identity metadata relating to that user. The values for each field are comma separated, and in PHP the data is in PHP `$_SERVER` variables, e.g.:

- `$_SERVER["HTTP_SHIB_STUDENT_AWARDTITLE"];`
- `$_SERVER["HTTP_SHIB_STUDENT_COURSECODE"];`
- `$_SERVER["HTTP_SHIB_STUDENT_FACULTY"];`
- `$_SERVER["HTTP_SHIB_STUDENT_MODULES"];`
- `$_SERVER["HTTP_SHIB_STUDENT_SCHOOL"];`

Module information

The system takes the module codes and obtains some high level information about the module, such as module leader and credits, from the "Module Outline Forms" (MOFS) system. SOAP was chosen because the data is in an object structure (if it had not been, RSS would have been used).

The system then queries the cache to find out if module information has been obtained recently for that module (currently every 5 minutes). If so, it is retrieved from the cache and displayed. If not, a separate, generic RSS Aggregator is called to query the module in each system. It first determines if there is any information held about that module in the given system at all.

Defined RSS standard

To query all potentially relevant systems, we defined a single RSS standard which we requested that all of our systems holding student resources adhere to. We kept this as basic as possible, using only one RSS extension.

The requesting URL might be

<http://system.ncl.ac.uk/resources.rss?module=ABCD123&type=videos>, though the module, type, and system details would obviously vary. An example of the response expected is provided in the diagram below:

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<rss version="2.0" xmlns:ev="http://purl.org/rss/1.0/modules/event/" xmlns:dc="http://purl.org/dc/elements/1.1/">
<channel>
  <title>Module Code / Module Title / Resource Type</title>
  <link>http://www.ncl.ac.uk/internal/module-catalogue/mof/2009/HIJK724</link>
  <description>If appropriate, information to show in the link - for example a forum resource might say "3 Posts"</description>
  <managingEditor>content.owner@ncl.ac.uk</managingEditor>
  <webMaster>system.owner@ncl.ac.uk</webMaster>

  <item>
    <title>May 24th : Hand in your Homework</title>
    <link>http://www.ncl.ac.uk/news-item-address</link>
    <description></description>
    <ev.startdate>2007-05-24T00:00:00</ev.startdate>
    <ev.enddate>2007-08-31T00:00:00</ev.enddate>
    <pubDate>2007-05-24T11:31:00</pubDate>
    <guid>http://www.ncl.ac.uk/news-item-address</guid>
    <category domain="http://www.ncl.ac.uk/news-page-address">Some Category of News</category>
  </item>

  <item>
    <title>May 24th : Hand in your Homework</title>
    <link>http://www.ncl.ac.uk/news-item-address</link>
    <description></description>
    <ev.startdate>2007-05-24T00:00:00</ev.startdate>
    <ev.enddate>2007-08-31T00:00:00</ev.enddate>
    <pubDate>2007-05-24T11:31:00</pubDate>
    <guid>http://www.ncl.ac.uk/news-item-address</guid>
    <category domain="http://www.ncl.ac.uk/news-page-address">Some Category of News</category>
  </item>
</channel>
</rss>
```

The Module Information section of the pilot examines only at the RSS header fields, to decide whether or not to display a link for each module.

- The **title** field contains the *Module Code* and *Title*, which enable us to check whether module names are accurate across multiple systems.
- The **link** field contains the URL for resources relating to this module.
- The **managingEditor** and **webmaster** fields are provided for support contacts, which allows data and systems feedback.

- The **description** field should contain the words '**Not Available**' if the system doesn't contain information for a particular module. It can otherwise be left blank, or contain information which would be useful to consumers of the field.

The cache maintains a record of whether a particular system contains any information for a particular module. This has two benefits: it prevents unnecessary queries where content does not exist, thus speeding up the page; and it allows us to easily obtain statistical data on *which* systems contain *how much* of *what kind* of information for *which* parts of our student population. Such data does not at present exist.

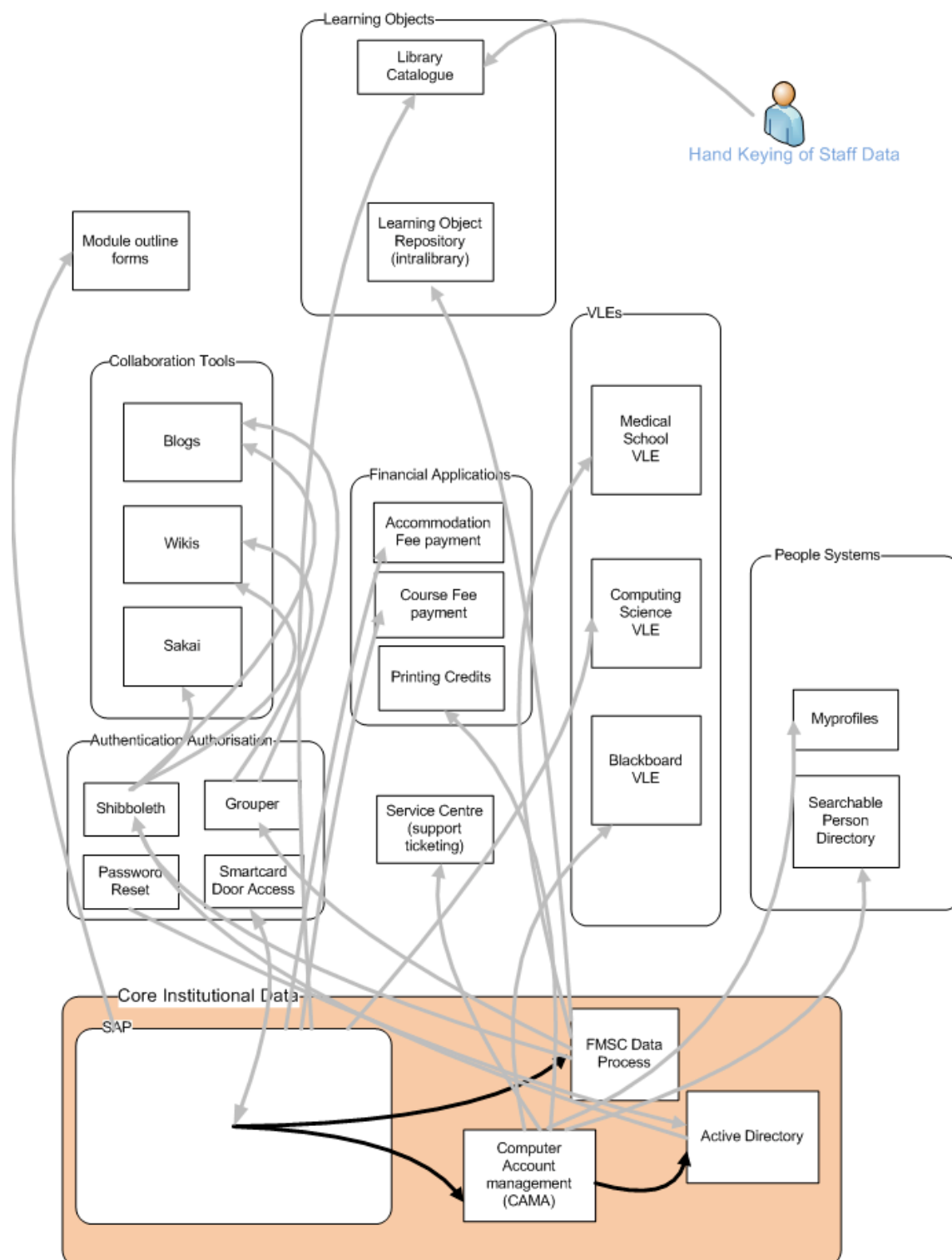
The news list panel uses the same RSS Aggregator with the same cache time. It makes use of the following fields (including the 'event module' to handle opening and closing dates) for each **item**.

- The **title** field should be a clear and intuitive. At present, different systems split information between **title** and **description** in different ways.
- The **link** field provides a link to the article described by the news item. Some systems don't hold an individual page per item, so would link to the system homepage and rely on the item being there as the person logs on.
- The **pubDate** field holds the publication date of the item.
- The **ev.startdate** and **ev.enddate** fields hold event start and end dates (where applicable). This is displayed on the news panel in the format "**Jun 16th : Item Detail**".

Ordering of events in the pilot is by **ev.startdate** in descending order, followed by **pubDate** in ascending order: future events first, followed by news items that don't have event dates.

Appendix B: the data flow makeover

The diagram below is taken from the project proposal and represents (as far as we were able to ascertain at the time) the institutional data flows.



This is what the institutional data flows looked like at the project end.

Amended Data Flows

