

# Technical review of the systems developed by the JISC 'Building MLEs in HE' (7/99) programme

Paul Browning  
University of Bristol  
V3.1 30<sup>th</sup> April 2003

BACKGROUND.....	1
AIMS .....	2
METHOD OF WORKING .....	3
ANALYSIS .....	4
PROJECT OVERVIEWS .....	5
<i>DMU-MLE</i> .....	5
<i>GIMIS</i> .....	5
<i>INSIDE</i> .....	8
<i>MARTINI</i> .....	8
<i>SMILE</i> .....	11
<i>TISR</i> .....	11
RECOMMENDATIONS.....	14
<i>A Potential for exploitation within other institutional contexts</i> .....	14
<i>B Potential for integration into other portals</i> .....	14
<i>C Requirements – additional technical development</i> .....	14
<i>D Requirements – documentation</i> .....	14
<i>E Requirements – further costs</i> .....	14
<i>F Requirements – distribution</i> .....	14
<i>G Transference – for projects</i> .....	15
<i>H Transference – for JISC</i> .....	15
ACKNOWLEDGEMENTS.....	17
REFERENCES .....	17
APPENDICES.....	17

## **Background**

The JISC-funded 'Building Managed Learning Environments (MLEs) in HE' programme is of three-year duration and will conclude in July 2003. The aim of the programme has been to explore developments that test, evaluate, and prove (or in some circumstances disprove) the generic deployment of technology in support of improved learning. The programme has developed good practice and shared ideas and experiences across FE and HE sectors. The specific objectives are to:

- Demonstrate to the FE/HE community the potential of MLEs to support and enhance learning and teaching;
- Identify the generic and transferable aspect of the development projects;
- Demonstrate the benefit of a student-centred approach to service provision;
- Enrich the educational sector understanding of the cultural and organisation issues encountered when developing MLEs;
- Create technical and organisational guidelines to support the introduction MLEs.

The fourteen projects funded under the programme have been exploring issues in four areas: the development of learner-centred MLEs; the development of institutional systems to support MLEs; the use of IMS specifications to develop MLEs; and strategies to achieve organisational change to support MLEs.

Six of the projects have developed a variety of joined up systems that, in general, link together back-end administrative systems and present a user interface through a web front-end. In many cases, the system that has been developed shares many of the characteristics of an institutional portal (also commonly known as Educational or Campus portals):

"An institutional portal provides a personalised, single point of access to the online resources that support members of an institution in all aspects of their learning, teaching, research and other activities. The resources may be internal or external and include local and remote 'information resources' (books, journals, databases, Web-sites, learning objects, images, student information systems etc.), 'transaction-based services' (room bookings, finance, registration, assignment submission, assessment, etc.) and 'collaborative tools' (calendars, email, chat, etc.). Access to many of these resources is usually restricted to authenticated members of the institution. In some cases the institutional portal may provide a view of institutional resources to end-users outside the institution, for example alumni and prospective students." [1]

JISC wished to review the systems that were developed and to consider their potential for future use in the wider FE and HE community. An Invitation to Tender for a technical review of the systems developed by the six projects that developed a variety of joined up systems under the programme was published in August 2002 [2]; this report is the result of the tendering exercise.

## ***Aims***

The aims of this report are to review products developed under the programme, in the context of international technology developments, to ascertain their potential future in the FE and HE community.

The objectives of the report are

- to document the products that are being developed by the projects, including how successfully they are achieving their objectives;
- to make recommendations about the potential deployment of the products in the wider FE and HE sector or the integration of the products with current open source portal software/frameworks.

## **Method of working**

The following approach was taken:

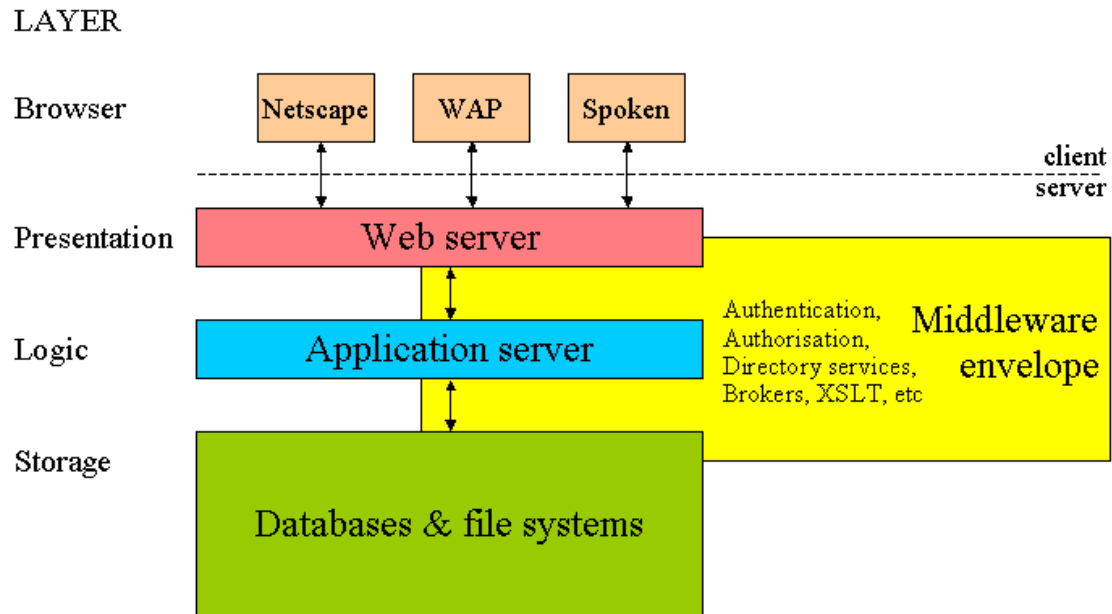
- A standardised template for projects to complete which would establish the product's functionality and context was developed using San Diego State University's "Rubric for Rating Commercial Portal Vendors" [3] and subsequently extended by the University of Bristol [4]
- The template was distributed via the JISC 7/99 Technical Review website [5]. The template contained sections to be completed by the projects and other sections to be completed subsequently by the reviewer [6].
- The responses were collated on the website for checking and comment by the projects. The completed templates are reproduced as Appendices to this report.
- Projects were invited to select a "Testing Week" during November and December 2003 by which time documentation, test datasets and software would be made available for installation and testing at the University of Bristol, and access provided to local development and production systems.
- During the Testing Week projects would ensure that a member of their team was available to handle queries from the reviewer.
- The Testing Week programme proved exacting for both projects and reviewer. Not all projects were in a position to release the raw materials requested (indeed for some projects it was never part of their original project plan to do so). As a result various compromises were adopted. The following table summarises the the various levels of access that had been made available to the reviewer by the time testing was completed at the end of January 2003:

	Template completed	Documentation – publically available	Documentation – restricted availability	Test data	Code	Access to development or production systems by role		
						SysAdmin	Staff	Student
DMU-MLE	*	*		*	*			10 accounts
GIMIS	*	*		Reviewer generated anonymised dataset on Oracle instance in Bristol.	GIMIS staff developed application on local server which queried remote dataset.	*	*	*
INSIDE	*	*		Extensive HyperCam videos demonstrating use of live systems.		VideoConference combined with NetMeeting allowed reviewer to see behind the scenes.		
MARTINI	*	*	*				*	
SMILE	*	*		Not applicable	*	*	*	*
TISR	*	*	*	*	*			*

- A draft report was circulated to projects in early February 2003 in advance of a workshop held on 11<sup>th</sup> February 2003 to discuss the initial findings of the report and suggested recommendations.
- The report was revised in the light of the workshop and circulated to members of JCLT in advance of their meeting on 21<sup>st</sup> May 2003.

## Analysis

Essentially all project products were underpinned by an architecture summarised in Figure 1:



*Figure 1: A generic MLE Architecture?*

Whilst common across the JISC 7/99 projects that were reviewed, Figure 1 is not really a generic MLE Architecture; rather it is typical of any N-tier Web application and contains easily identifiable storage, logic and presentation layers on the server side of the application.

Harder to pigeon-hole was the “glue” that allows interoperability between the various layers to occur – the term “middleware envelope” is used here to bracket the integration magic that holds it all together. It was principally in the nature of middleware envelope that projects differed from a technical standpoint – and this is also where the major innovations of this strand of the JISC 7/99 programme can be identified.

## ***Project overviews***

Details of the projects can be found in the supporting appendices. The overall impression is that the six projects had more in common than they had differences. This commonality is conveyed in Figures 2-13; the architecture diagrams have been constructed using the same schema as that used in Figure 1.

### **DMU-MLE**

This project sought to integrate the student information system, the timetabling system, key documents (e.g. student handbook, regulations) and a news/messaging system. The product was primarily aimed at students of De Montfort University.

The project started development using Java and XML but switched to the uPortal framework when Version 2 became available.

The potential for exploitation within other institutional contexts is very wide; the uPortal framework is agnostic and standards-based. The degree to which other systems can integrate is limited only by whether they offer adequate APIs and whether sufficient expertise is available to undertake the task (and maintain the service subsequently).

In general the architecture is flexible. However, the most innovative component of the product - the MLE-QLS Broker - is also the most proprietary in terms of the toolkit (Visual Basic) and the platform (Windows) required. For such middleware to be adopted widely the approach needs to be recast in a more open, less platform dependent alternative.

### **GIMIS**

Integration of Writtle College's four primary databases (finance, library, student records and timetabling) was the primary goal of this project. In addition, about twenty-five ancillary databases and discrete data sources used throughout the College were also integrated within the GIMIS core (e.g. locations database, FE Attendance). The product was aimed at both staff and students.

The architecture is flexible if working within a Cold Fusion environment. The potential for exploitation within other institutional contexts is limited by this factor. The current prototype exposes an extraordinarily rich range of information drawn from many content sources within the institution and demonstrates what can be achieved with a high-level of commitment to an Information Strategy.

# DMU-MLE

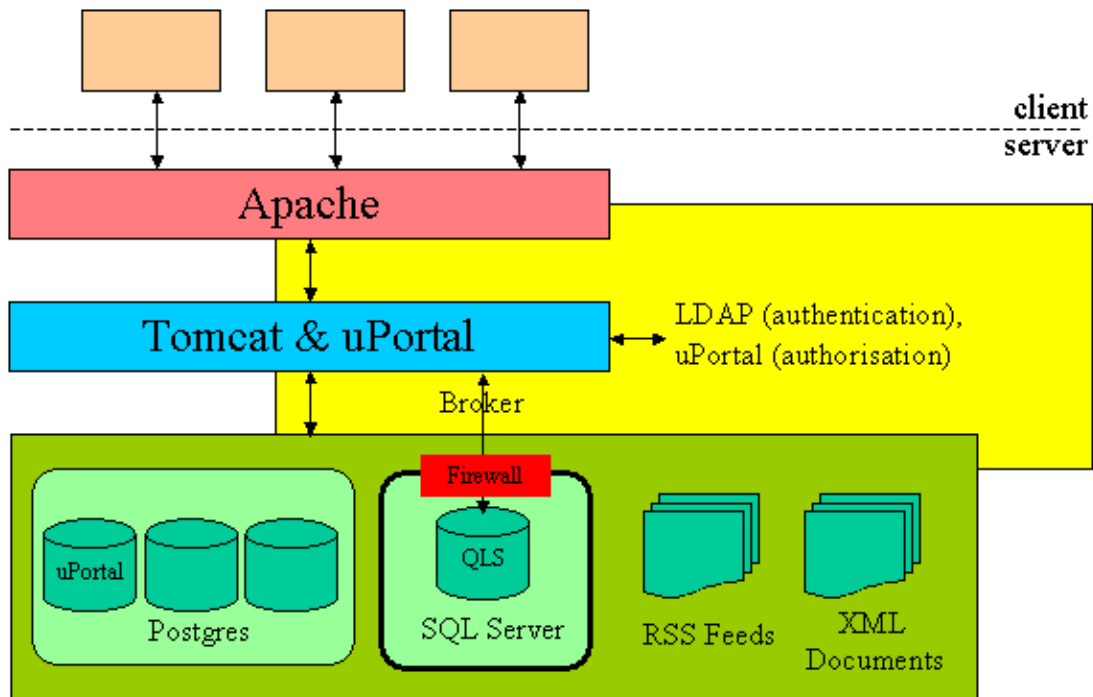


Figure 2: The DMU-MLE architecture

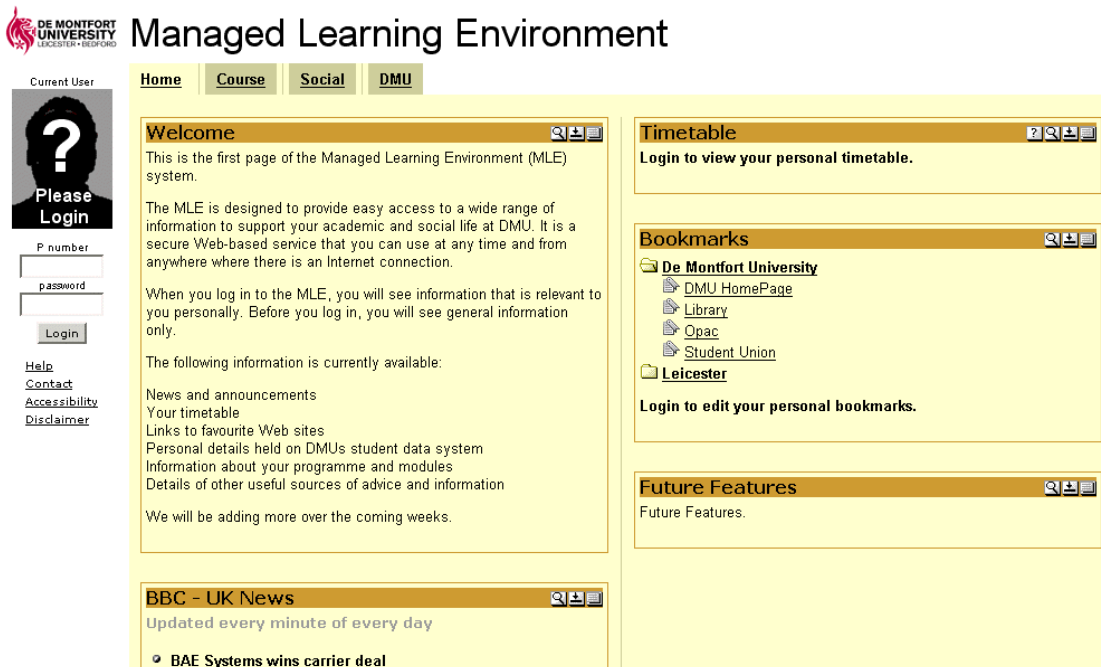


Figure 3: Screenshot of DMU-MLE – the Home Tab (before login)

# GIMIS

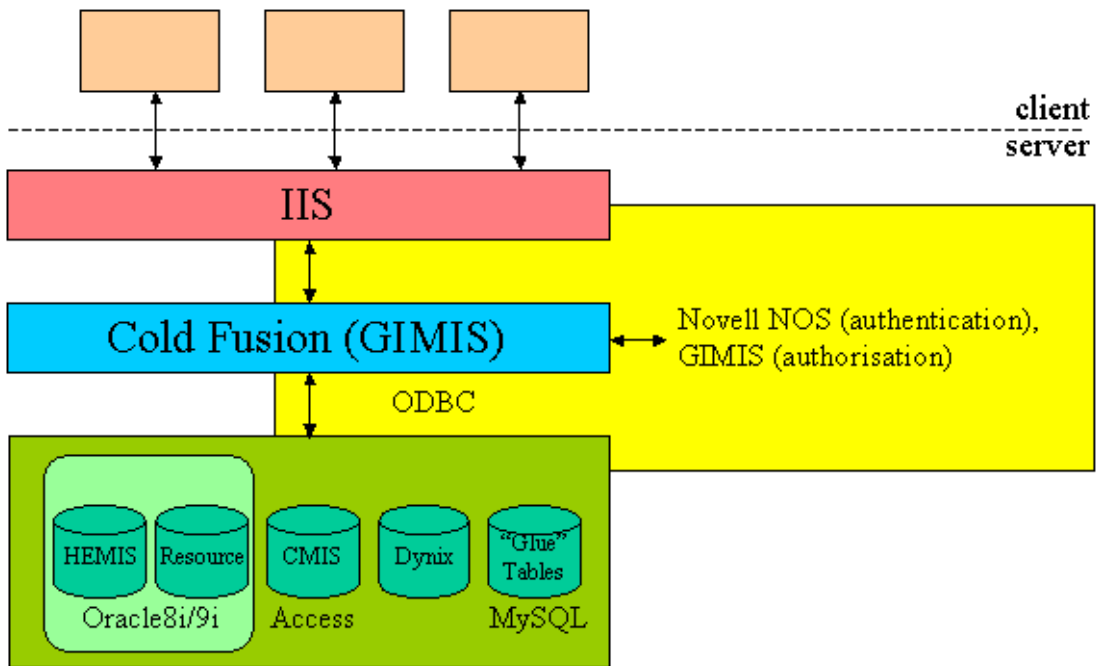


Figure 4: The GIMIS architecture

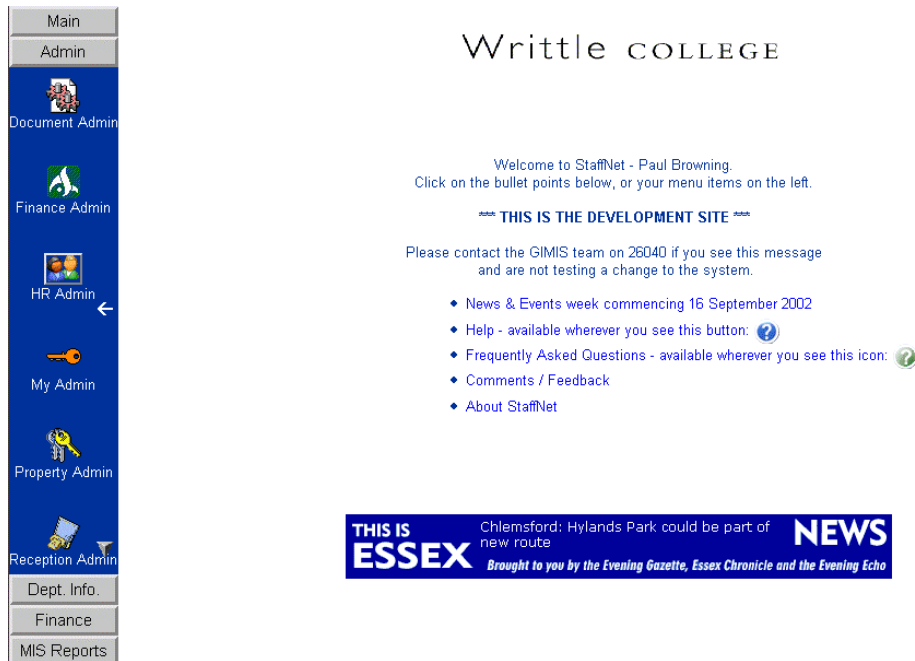


Figure 5: Screenshot of GIMIS StaffNet - Admin Menu

#### **INSIDE**

Perhaps the distinguishing feature of this project (a joint venture between St. Andrews and Durham Universities) within the JISC 7/99 programme was its recognition that information maintained by departments, and the work involved in doing so, are important components in the overall institutional information base.

This project sought to integrate departmental and centrally maintained systems. In particular it focussed on the exchange and management of data about students and modules (the “Module Management System” - MMS). The product was mainly aimed at academic staff.

There are many components to the overall system; it is not clear to what extent they would be portable to other institutions (this was never an explicit goal of the project). The diversity of applications suggests that the system is indeed flexible within the local departmental context providing in-house expertise is available to customise and develop the system.

#### **MARTINI**

This project sought to provide access to a wide range of systems: student records, unit enrolments and marks, course data, accommodation records, debtors and payments data, library borrower information, IT Service user data and timetabling information. It was primarily aimed at students of the University of East Anglia.

The project started development using WebObjects but switched to the uPortal framework in its final stages. The architecture is flexible; the IMS - Generator and XML - IMS conversion programs can work with either a WebObjects-based or uPortal-based system.

The use of XML to act as the interface with institutional data allows this middleware to sit between any data and any web application that can import and read XML. If you know your institutional data structures and have the expertise to modify a few (less than five) XML documents then the middleware can work in your context.

The potential for exploitation within other institutional contexts is very wide (see comments above concerning the use of uPortal in the DMU-MLE project). It is not yet clear that “deep” integration of systems has been achieved with the current prototype; there is not yet passing of authentication credentials between component systems.

# INSIDE

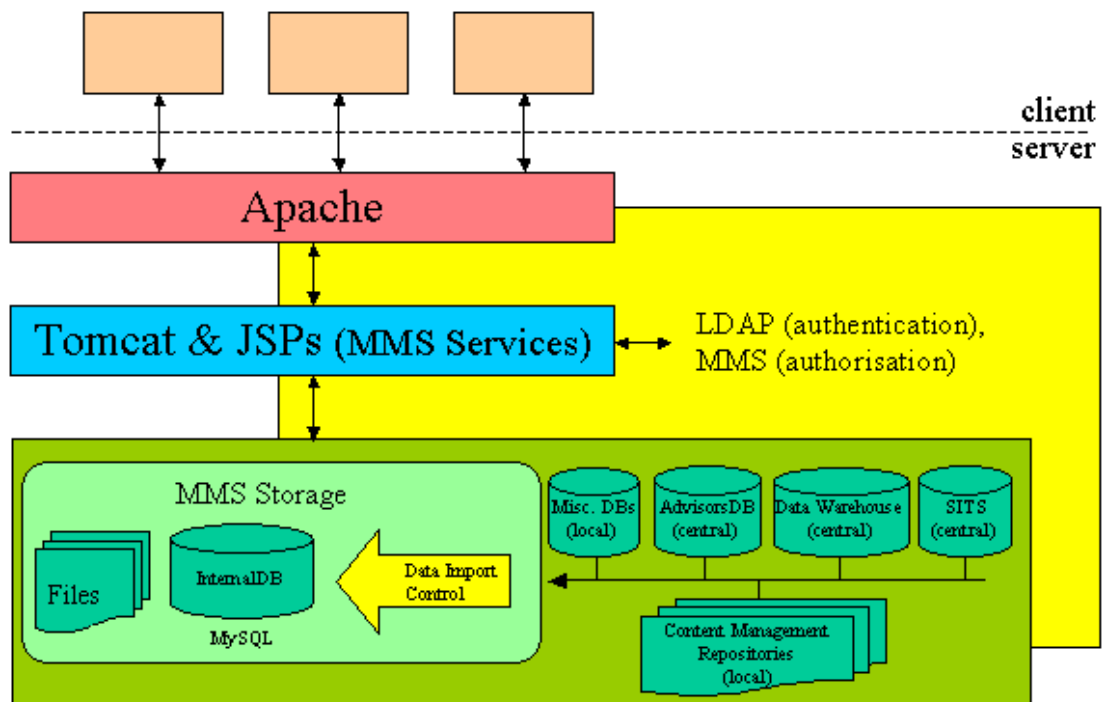


Figure 6: The INSIDE architecture

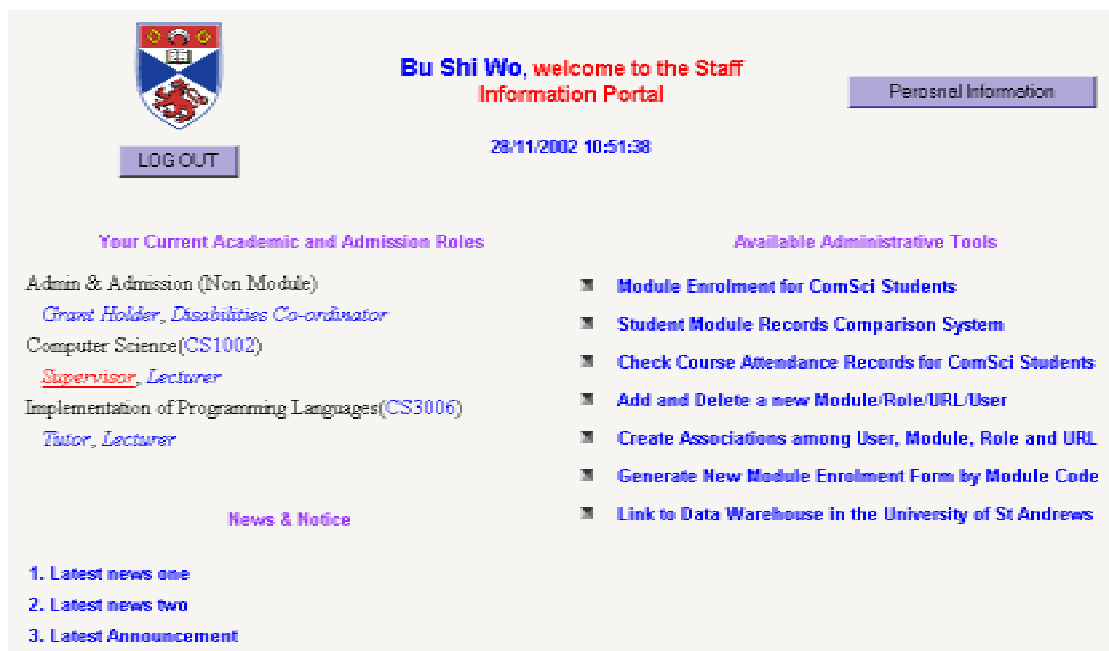


Figure 7: Screenshot of INSIDE Staff Information Portal

# MARTINI

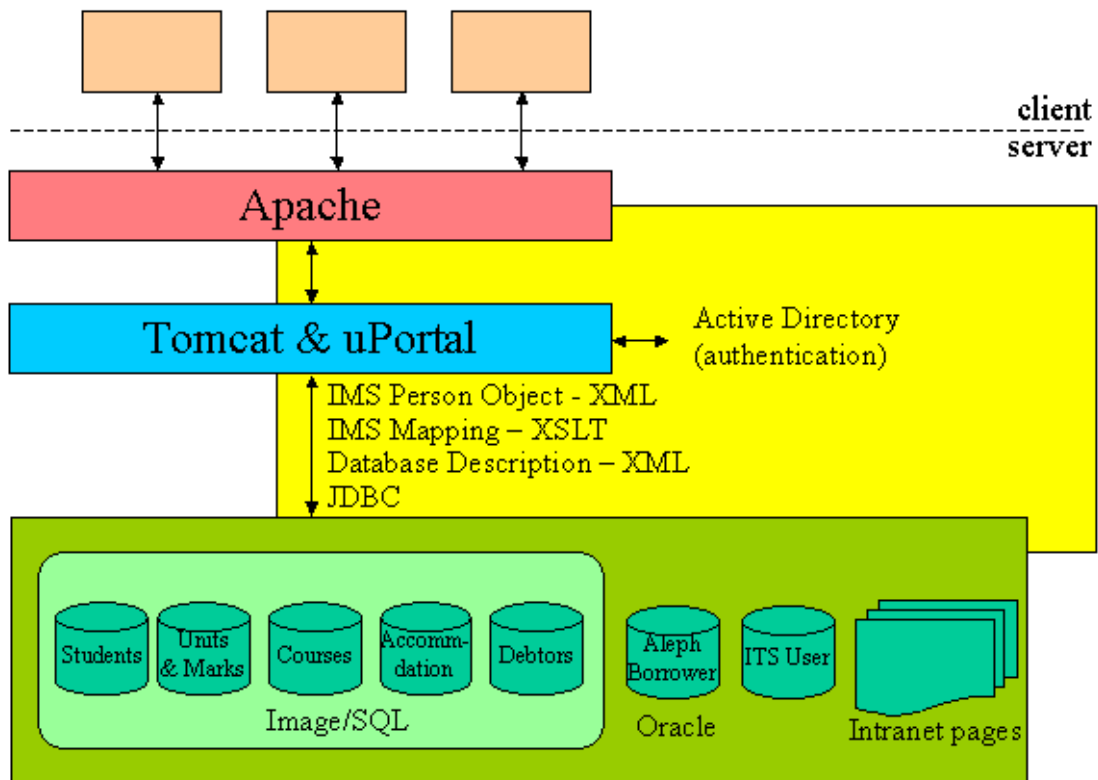


Figure 8: The MARTINI architecture

**University of East Anglia**

Student Details	
Description	Content
Registration Number	0136212
Full Name	Mr Hai , Hing Kwong
Title	Mr
Family Name	Hai
Given Name	Hing Kwong
Initials	HK
Gender	Male
Marital Status	
Nationality	HONG KONG
Date of Birth (dd/mm/yyyy)	15/10/1975
Disability	No known disability
Email Address	a136212@uea.ac.uk
WebSpace URL	http://www.uea.ac.uk/~a136212
School	School of Information Systems
Study Status	G
Study Method	Part-time
Year	02
Course	G400 ( R2 )
Course Length	48 Months
Qualification Expected	Master of Philosophy (Lab based)
Registered Data	01/10/2001
Finish Data	30/09/2005

MARTINI SYSTEM [2002]

Figure 9 Screenshot of MARTINI - Student Details screen (after login)

### **SMILE**

This project at Sunderland University attempted to integrate a virtual learning environment (WebCT) and student records system (SITS) within a portal. The product was aimed at both staff and students.

The project is distinctive in not having embraced Java as a development platform and instead adopting Zope (a Python-based Web application framework). Whilst this architecture is highly extensible (and portable), the project was able to re-use existing components (Plone - itself built upon the Content Management Framework) to create their product more-or-less “out of the box”.

SMILE has not yet achieved deep integration of its SIS and VLE systems. It has achieved a degree of being joined-up by providing a “one stop shop” at which users can find collected in one place the links they need to other campus. But there is no passing of authentication credentials; a separate login is required at the systems (e.g. E-mail, VLE) to which you are taken.

### **TISR**

The validating example of the TISR project is a model of the student record. The project shows how a student record can be embodied as an LDAP directory entry, and, using the TISR API, how the LDAP directory can be populated with student records whose individual components are derived from different sources.

TISR is described as a “lazy meta-directory”. Its role is to integrate the disparate data sources that typically make up the student record. Sample data connectors including JDBC, LDAP, EJB and XML have been developed. In short TISR will integrate whatever you want. The product aimed at systems administrators.

Aware of the international efforts to produce specifications for interoperability such as the IMS, EduPerson and LiPerson “student objects”, TISR focused on a pragmatic solution whilst such standards consolidate and manufacturers of both instructional and administrative software for the educational sector implement them.

The potential for exploitation within other institutional contexts is probably universal - what institution does *not* have more than one flavour of content repository? The TISR middleware is agnostic and standards-based.

Ravensbourne College has delivered an exemplar project which should inform the future development of any “project standards” that the JISC may wish to consider in programmes involving technical development.

# SMILE

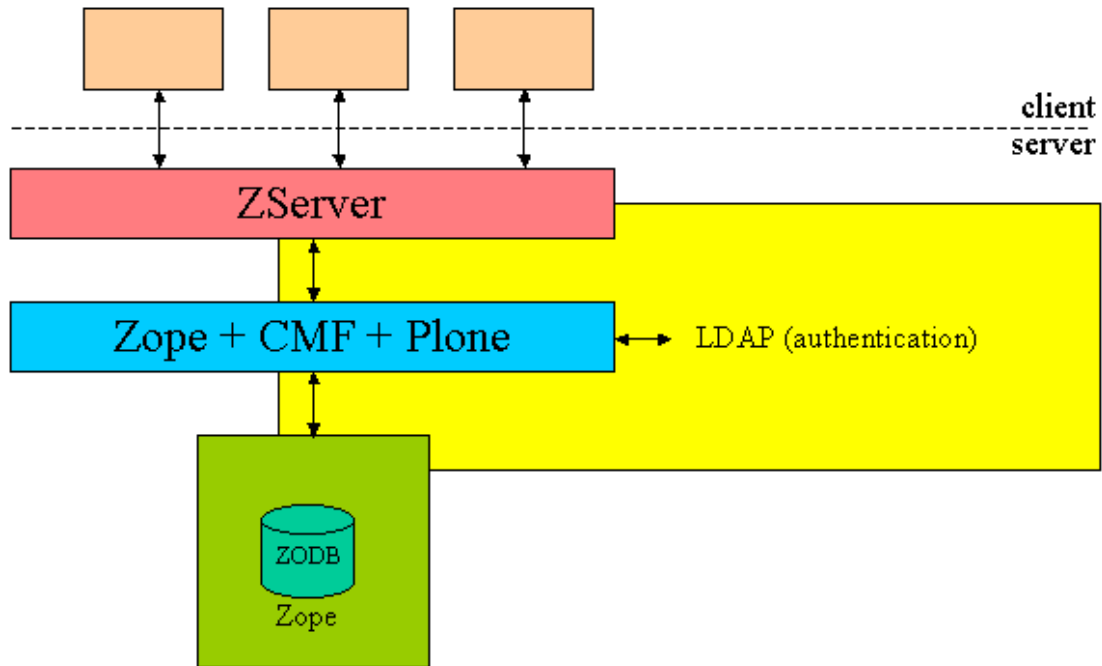


Figure 10: The SMILE architecture

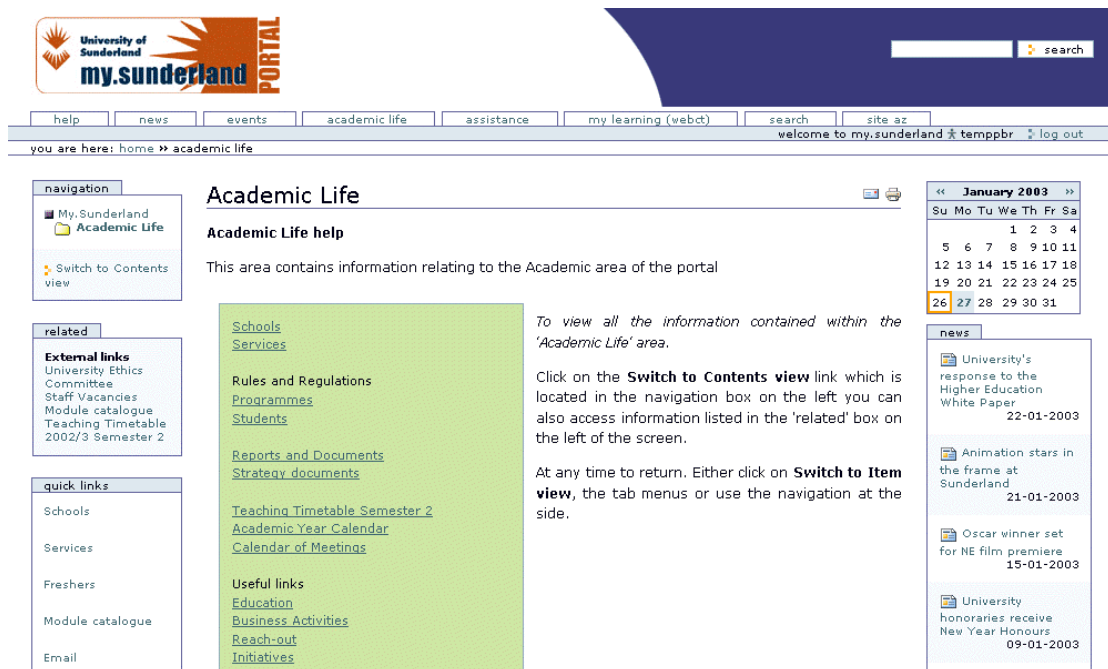


Figure 11: Screenshot of SMILE - Academic Life Tab

# TISR

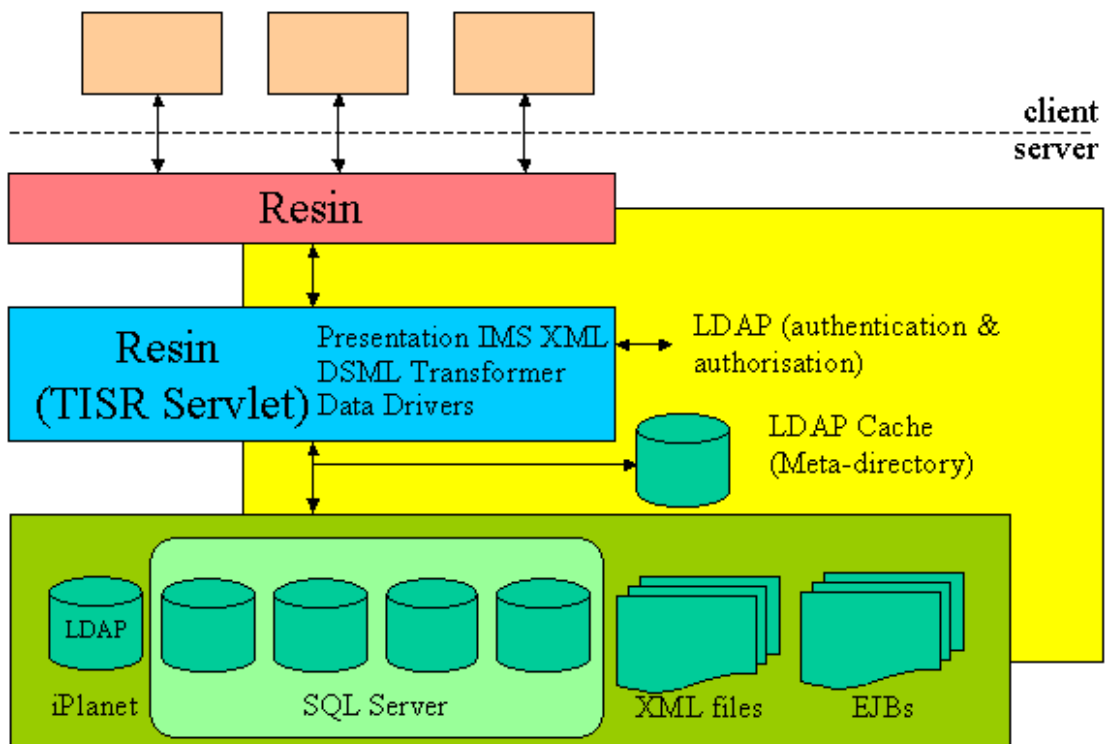


Figure 12: The TISR architecture


<b>diet tizr</b>	<b>Walt Disney</b>
	<pre> cn: Walt Disney displayname: DISNEY, Walt edupersonnickname: Walt edupersonprincipalname: 97990602@tizr givenname: Walt mail: w.disney@tizr sn: Disney tisraos: BA Product And Furniture Val-01 tisraoslevel: 1 uid: 97990602                 </pre> <p style="color: blue; text-decoration: underline;"><a href="#">Search again.</a></p>
<p style="font-weight: bold; margin: 0;">diet tizr</p> <p style="font-size: small; margin: 0;">The tizr Project at Rave tizr Demonstrator Overview tizr Demonstrator Downloads View Demonstrator sources</p> <p style="font-size: small; margin: 0;">IMS Project The Open Group OASIS</p> <p style="font-size: small; margin: 0;">OpenLDAP Netscape Directory Server iPlanet Directory Server</p> <p style="font-size: small; margin: 0;">Caucho Technologies Apache XML Projects</p>	
<b>JISC</b>	
JISC Website	

Figure 13: Screenshot from TISR demonstrator – Viewing search results

## **Recommendations**

Recommendations have been made on a project-by-project basis and are included verbatim in the relevant Appendix. This section provides a general commentary and offers some broader recommendations that can be distilled from this review.

### **A Potential for exploitation within other institutional contexts**

The rich diversity of applications in several projects already demonstrates the potential for exploitation – but in the context of the *local* institution. It is less clear how well the products would map on to the context of *other* institutions. The heterogeneity of back-end systems within and between institutions, and the need for specialised skill-sets to install, configure, develop and support the products, reduces the sector-wide potential for exploitation.

### **B Potential for integration into other portals**

Most projects had built products that manifested themselves to end-users as a portal – though the level of true personalisation and “deep” integration achieved was variable. The potential for integration into other portal frameworks will be limited by the degree of separation, in software engineering terms, that has been achieved between the code responsible for presentation from that handling the underlying logic and data.

### **C Requirements – additional technical development**

#### **C1 An assessment of middleware**

The analysis of the architectures employed (above) leads to the principal recommendation of the review: an assessment of the middleware that has been developed is required. There are many questions:

- Which middleware is best?
- Can one size fit all?
- To what extent can functions *within* the middleware envelope be defined in common ways across institutions?
- To what extent can functions *to and from* the middleware envelope be defined so as kept them to a manageable number?
- What are the key functions to concentrate on to fit the greatest number of institutional environments?
- Is “a middleware toolkit” potentially a UK contribution to global portal framework projects such as uPortal?
- How might this link with projects such as ANGEL? Is this a task to be undertaken by CETIS?

### **D Requirements – documentation**

In general the existing or planned documentation was satisfactory or better.

### **E Requirements – further costs**

See relevant appendix.

### **F Requirements – distribution**

#### F1 Advice on software licensing

Projects may wish, or perhaps should be compelled in future, to release their software products into the public domain. There are a bewildering array of “open source” licenses available [7] and advice on which is most suitable is desirable. Alan Robiette (pers. comm.) has suggested that the MIT license may be a good choice for JISC-funded projects.

#### F2 Advice on software distribution

At present each project is maintaining its own web site. Might it be desirable for repositories such as SourceForge [8] to be used for this purpose?

A future JISC Advisory Service on Open Source Software [10] might be expected to offer guidance on both these fronts.

#### **G Transference – for projects**

See relevant appendix.

#### **H Transference – for JISC**

##### H1 Technical evaluation should be an expectation for any programme

This evaluation was an unexpected burden for project teams – who coped admirably and with good humour in dealing with the demands placed on them by the reviewer. However, projects should have been allowed to plan for the evaluation from the outset.

##### H2 Direction on software tools

Things get awkward if projects mix open source and commercial components in their products. If commercial components are used then an immediate activation energy is placed in the way of further investigation and use of any products that are developed. Should there be a stipulation in future programmes that only open source tools should be used?

##### H3 SSL from the word go

Many projects intend to secure their systems with SSL but towards the end of the project cycle. Aside from exposing users of prototype systems to the risk of plain text password interception, retrofitting SSL is not always straightforward. SSL should be used from the outset. Digest authentication is an alternative to SSL for password protection which may be easier to implement, as and when campuses can be sure that enough of their browser population supports this.

##### H4 LDAP as a common denominator

Every project is using an authentication service based on, or compliant with, LDAP. Is it time to stipulate the use of (at least) LDAP in future development projects?

##### H5 Test data sets

Test data sets are essential for software development and are also desirable when distributing software (to allow testing of a local installation). There seems to be a paucity of advice on how best to generate test data and guidance in this area would be welcome.

#### H6 Stress testing

Building “enterprise” Web applications may be becoming more straightforward but ensuring they work satisfactorily when loaded is not routine. None of the projects were able to report qualitative measures of likely scalability. Should stress testing be a standard element of any development project plan?

#### H7 Load-balancing

Building portals is becoming easier but that running them successfully (in performance terms) is something that has still to be learned by the sector. There are about equal measures of interest and ignorance in the area of load-balancing (which has become standard practise in the commercial sector). JISC should consider funding projects that investigate, distil and disseminate best practise in this area.

#### H8 Web Services

We need loosely coupled components from which to build joined-up systems. Institutions should be able to pick and mix. Standards are obviously crucial. The JISC standards effort has been directed so far at content interchange – and rightly, as there was a near vacuum when the programme started. Recently an “JISC Information Environment Architecture Standards Framework” document has been provided [11]. A JISC Technology and Standards Watch Report on Web Services is also in preparation [12].

In time standards like WSRP [7] will arrive to allow inter-portal operability and/or portability. If one accepts that the inevitable expression of an MLE *today* is a portal, then what are the “standards” (which may be more “architectures” – i.e. a collection of standards) which should be given support to allow a modicum of code/product re-use *now* (whilst we wait for things like WSRP to mature)? JISC says it can’t be seen to endorse particular vendors – but are open source projects like Jetspeed, uPortal and Zope to be equated with “vendors”? Perhaps JISC should be bolder.

#### H9 Content Management Systems

MLEs/Portals have been tackling the easy bit – exposing relatively structured data in databases. But the majority of an institution’s data is much more unstructured information. This is where the next challenge may lie for many. Do Content Management Systems have a role to play? Most commercial VLEs are very poor CMSes. Meeting the reporting requirements of the Disability Discrimination and Freedom of Information Acts in a resource efficient way may see institutions investigating this technology anyway. Should JISC tap into this expertise?

#### H10 Write-enabled transactional systems

Most (all?) projects seemed to be based in institutions that were not yet culturally ready to consider the devolution of maintenance of part of the staff or student record to the user themselves. Clearly the notion of “self-service portals” demands such functionality. If other organisations are to benefit from the middleware being developed then the software needs to support information push as well as pull.

#### H11 Technology overlap

The JISC Portals FAQ [1] mentions ‘information resources’, ‘transaction-based services’ and ‘collaborative tools’ as characteristic of portals. The current projects are rich in the first category but examples of the more dynamic latter categories are harder

to identify. Here the overlap with VLEs, CMSes and collaborative groupware systems becomes substantial. Would some case studies provide some guidance in this area?

## **Acknowledgements**

It has been a privilege (and a learning experience) to see behind the scenes of the projects, their code and their host institutions - their co-operation is gratefully acknowledged. My thanks in particular to Tish Roberts (JISC Programme Manager) for the support and patience she has shown. Alan Robiette, Brian Davies and members of the JISC 7/99 Project Teams have provided helpful comments and suggestions on earlier drafts of this report.

## **References**

1. JISC Portals FAQ  
<http://www.jisc.ac.uk/dner/development/programmes/portalsFAQ.html>
2. Invitation to Tender: Technical review of the systems developed by the JISC 'Building MLEs in HE' (7/99) programme  
[http://www.jisc.ac.uk/pub02/7\\_99\\_tech\\_review.html](http://www.jisc.ac.uk/pub02/7_99_tech_review.html)
3. San Diego State University's "Rubric for Rating Commercial Portal Vendors"  
[http://its.sdsu.edu/portal\\_rubric.pdf](http://its.sdsu.edu/portal_rubric.pdf)
4. University of Bristol's "Technical assessment of uPortal and Oracle Portal"  
[http://www.bris.ac.uk/is/projects/portal/docs/uob\\_portal\\_rubric2.html](http://www.bris.ac.uk/is/projects/portal/docs/uob_portal_rubric2.html)
5. JISC 7/99 Technical Review website  
[http://www.bris.ac.uk/is/projects/portal/7\\_99](http://www.bris.ac.uk/is/projects/portal/7_99)
6. Technical Review Project Template  
[http://www.bris.ac.uk/is/projects/portal/7\\_99/template1](http://www.bris.ac.uk/is/projects/portal/7_99/template1)
7. OSI Certified Open Source Software - The Approved Licenses  
<http://www.opensource.org/licenses/index.php>
8. SourceForge.net  
<http://sourceforge.net/>
9. OASIS Web Services for Remote Portals (WSRP) - Whitepaper  
[http://www.oasis-open.org/committees/wsrp/documents/wsrp\\_wp\\_09\\_22\\_2002.pdf](http://www.oasis-open.org/committees/wsrp/documents/wsrp_wp_09_22_2002.pdf)
10. Circular 3/03: JISC Advisory Service on Open Source Software  
[http://www.jisc.ac.uk/index.cfm?name=circular\\_3\\_03](http://www.jisc.ac.uk/index.cfm?name=circular_3_03)
11. JISC Information Environment Architecture Standards Framework  
<http://www.ukoln.ac.uk/distributed-systems/jisc-ie/arch/standards/>
12. Technology and Standards Watch Reports  
[http://www.jisc.ac.uk/index.cfm?name=techwatch\\_reports\\_list#prep](http://www.jisc.ac.uk/index.cfm?name=techwatch_reports_list#prep)

## **Appendices**

1. DMU-MLE  
[http://www.bris.ac.uk/is/projects/portal/7\\_99/dmumle.pdf](http://www.bris.ac.uk/is/projects/portal/7_99/dmumle.pdf)
2. GIMIS  
[http://www.bris.ac.uk/is/projects/portal/7\\_99/gimis.pdf](http://www.bris.ac.uk/is/projects/portal/7_99/gimis.pdf)
3. INSIDE  
[http://www.bris.ac.uk/is/projects/portal/7\\_99/inside.pdf](http://www.bris.ac.uk/is/projects/portal/7_99/inside.pdf)
4. MARTINI  
[http://www.bris.ac.uk/is/projects/portal/7\\_99/martini.pdf](http://www.bris.ac.uk/is/projects/portal/7_99/martini.pdf)

5. SMILE  
[http://www.bris.ac.uk/is/projects/portal/7\\_99/smile.pdf](http://www.bris.ac.uk/is/projects/portal/7_99/smile.pdf)
6. TISR  
[http://www.bris.ac.uk/is/projects/portal/7\\_99/tisr.pdf](http://www.bris.ac.uk/is/projects/portal/7_99/tisr.pdf)