

JISC DEVELOPMENT PROGRAMMES

Project Document Cover Sheet

APIS PROJECT PLAN

Project

Project Acronym	APIS	Project ID	
Project Title	Assessment Provision through Interoperable Segments		
Start Date	10 th May 2004	End Date	29 th October 2004
Lead Institution	University of Strathclyde		
Project Director	Niall Sclater		
Project Manager & contact details	Rowin Cross Tel 0141 548 2298; email rowin.cross@strath.ac.uk		
Partner Institutions			
Project Web URL	http://ford.ces.strath.ac.uk/APIS		
Programme Name (and number)	Elearning Programme: Technical Framework and Tools strand		
Programme Manager	Tish Roberts		

Document

Document Title	<i>APIS Project Plan</i>		
Reporting Period			
Author(s) & project role	Rowin Cross, coordinator		
Date	8 th June 2004	Filename	APIS ProjectPlan.doc
URL			
Access	<input type="checkbox"/> Project and JISC internal		<input type="checkbox"/> General dissemination

Document History

Version	Date	Comments
v0a	8 th June 2004	

JISC Project Plan Template

Overview of Project

1. Background

Summarise the background to the project (and how it builds on previous work) and the need for it (and why it's important).

The project will implement a modular assessment system in line with the IMS Question and Test Interoperability (QTI) version 2 Lite specification. It will also scope the operations required by tools to make use of this system, and put forward a new API which will accommodate these needs.

The core deliverable of the project will be the simple APIS (Assessment Provision through Interoperable Segments) runtime assessment rendering engine constructed in accordance with the IMS QTI Lite specification version 2.0. This will provide the minimum functionality necessary for an assessment engine, by providing the basic libraries required for constructing, delivering and assessing multiple choice and multiple response items. The functionality covered by the Lite specification can be regarded as the unambiguous core of a large and complex specification, and the availability of an open source implementation of this will provide an ideal starting point for future implementations of the full specification which address the less clearly defined or more problematic areas. The engine will be developed on the principle that additional modules providing additional functionality can be incorporated into the core system. It will be possible to produce a series of modules (or 'segments'), each targeted at a specific interaction type as defined by the QTI v2.0 specification. Users will be able to incorporate these segments into APIS or any other QTI assessment system.

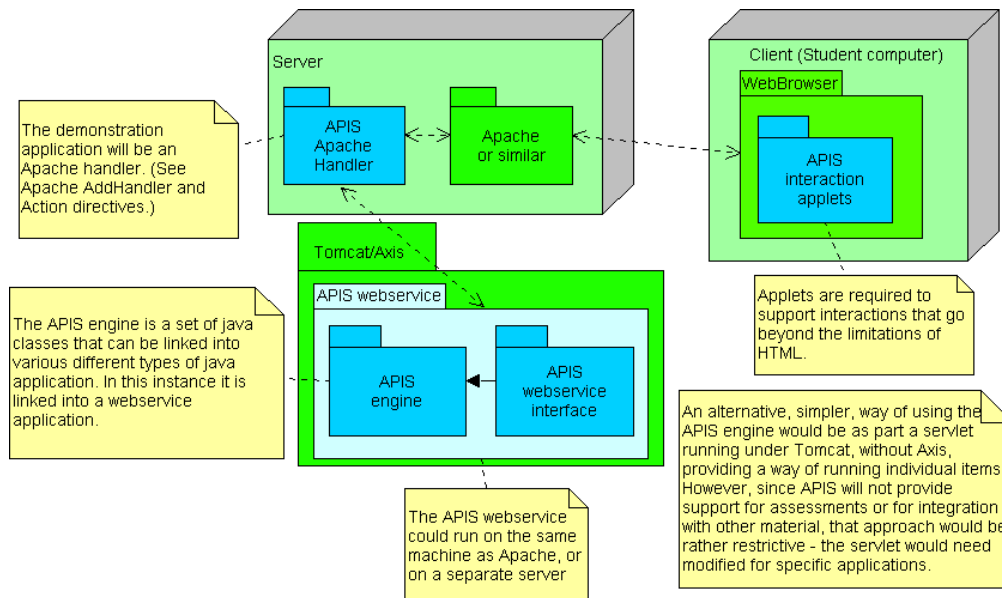


Diagram 1: Deployment of the APIS engine.

As a starting point to this, and taking account of the limits of the proposed time scale, at least one such segment will be produced as a deliverable from the project. As well as providing additional functionality, this module (or modules) will offer a reference model for the development of further segments. A modular approach enables a variety of approaches to teaching and learning to be realised, and allows institutions and individuals to tailor their online assessment materials to suit their pedagogic methodologies.

The adoption of such a modular approach allows developers to ‘mix and match’ components to suit their own needs, and to integrate their own applets as they wish. This will have particular benefit for accessibility specialists, as it will be possible to add heavily customised segments to the main assessment module.

There is a clear need for a well documented open source implementation of the IMS QTI specification. Although there are a number of implementations of the specification currently available, including the TOIA (Technologies for Online Interoperable Assessment) X4L tool which is being distributed free of charge throughout UK Further and Higher Education, none of these implementations is open source; in other words, it is not possible for the developer community to take advantage of work already done in the development of assessment tools. APIS aims to increase interoperability in practice by providing liberally licensed open source implementations of non-controversial elements of the specification to the community as a whole.

The API which will be produced by the project will provide runtime support for the assessment engine. It will have a WSDL 1.1 definition and reference implementation using Apache AXIS. This activity is consistent with work identified by the IMS InterActive Content special interest group, which aims in future to address a web-services based API model for QTI, Learning Design and Simple Sequencing. As the project team are involved in the IMS InterActive Content SIG, it is hoped that the API produced by the APIS project will both inform and be informed by IMS activity in this area, ensuring greater interoperability.

APIS is intended to complement existing resources such as the CopperCore learning design engine¹ and Navigo², an open source implementation of IMS QTI and the OKI Assessment and Grading OSIDs which shares APIS's philosophy of a ‘pluggable’ architecture and modularization. The work will also be informed by the findings and experiences of activities such as the CETIS Enterprise Web Services toolkit, as well as the experiences of other projects in the present programme.

2. Aims and Objectives

The overall aim of the project is to implement a modular item rendering engine in line with the IMS Question and Test Interoperability (QTI) specification. It will also scope the operations required by tools to make use of this system, and put forward a new API which will accommodate these needs based on the work of the Open Knowledge Initiative (OKI) and IMS Web Services.

The specific objectives are to:

- Produce open source code libraries for IMS QTI v2.0 and an API to provide runtime support for assessment delivery.
- Enrich users' understanding of the issues around implementing the IMS QTI specification, and integrating QTI material with other learning resources through the use of the IMS Learning Design and Simple Sequencing specifications.
- Produce additional open source modules ('segments') to provide additional functionality to assessment systems.
- Create guidelines for the integration of QTI with Learning Design and Simple Sequencing, and explore best practice in this area.

3. Overall Approach

The project will implement a modular assessment system in line with the IMS QTI v2.0 specification. It will also scope the operations required by tools to make use of this system, and put forward a new API which will accommodate these needs. The engine will be developed on the principle that additional modules providing additional functionality can be incorporated into the core system. It will

¹ <http://coppercore.org>

² <http://navigo.sourceforge.net/>

be possible to produce a series of modules (or 'segments'), each targeted at a specific interaction type as defined by the QTI v2.0 specification. Users will be able to incorporate these segments into APIS or any other QTI assessment system. As well as providing additional functionality, these modules will offer a reference module for the development of further segments. A modular approach enables a variety of approaches to teaching and learning to be realised, and allows institutions and individuals to tailor their online assessment materials to suit their pedagogic methodologies. The adoption of a such a modular approach allows developers to 'mix and match' components to suit their own needs, and to integrate their own applets if they wish. This will have particular benefit for accessibility specialists, as it will be possible to add heavily customised segments to the main assessment module.

4. Project Outputs

- Open source code libraries for a core QTI v2.0 item rendering engine.
- UML model of assessment engine interactions.
- Documentation on the implementation of code libraries.
- UML model of API interactions required to support the delivery of functionality provided by APIS segments.
- API to give runtime support for assessment rendering engines.
- Documentation on the integration of IMS QTI, Learning Design and Simple Sequencing.
- APIS interaction applet code.
- Model of integration of applets with core system.
- Project information resources, such as project webpage and blog.
- Presentations to relevant community groups.
- APIS conformance statement.
- Final report on project.

5. Project Outcomes

List the outcomes you envisage, including their impact on the teaching, learning, or research communities, and what change they will stimulate or enable.

- Reference implementation of the IMS QTI v2.0 core specification.
- Increased understanding of the specification, including the opportunity to feed into specification development process.
- Increased understanding of approaches to the integration of QTI, Learning Design and Simple Sequencing.
- Reusable code libraries freely available to the community resulting in increased interoperability, and allowing developers to concentrate on more complex or specific areas of development.
- Greater flexibility for assessment producers to reflect varied pedagogies through the adoption of modular approach to engine development.

6. Stakeholder Analysis

Stakeholder	Interest / stake	Importance
CAA developers	Reusable open source resources	high
Standards and specifications community	Identification of issues in specifications	high
Learners	Availability of learning resources	medium
Teachers	Flexible resource for assessment development	medium
CETIS, particularly Assessment SIG	Identification and involvement with successful project	medium

7. Risk Analysis

Risk	Probability (1-5)	Severity (1-5)	Score (P x S)	Action to Prevent/Manage Risk
Staffing: <ul style="list-style-type: none"> · Niall Barr, Developer · Rowin Cross, Manager · Niall Sclater, Director 	1 1 1	5 3 3	5 3 3	All: good management practices, open dialogue, clear grievance raising procedures
Technical: <ul style="list-style-type: none"> · Delays in spec. release · Changes in spec. · Technical difficulties implementing spec. 	4 4 4	2 3 3	8 12 12	Delays not problematic as project team are members of IMS project team. Substantial changes in specification could cause problems given short timescale for APIS. Technical problems are likely, and would reduce the usability of the released code libraries; however, the modular nature of the engine would overcome this significantly.
External suppliers – consultant to work on integration model	3	5	15	Early identification of consultant to work on integration document. Formal contract for work agreed early.
Legal	2	3		Ensure that confidential information such as details of QTI v2 is kept private until formally released by IMS.

8. Standards

APIS will be one of the first implementations of the IMS Question and Test Interoperability specification version 2.0. It will take account of new developments in the QTI specification, such as the emphasis on Content Packaging assessment material and the use of IEEE LOM and QTI v2.0 metadata, through exploitation of a QTI content package viewer already released by the APIS project developer and a QTI metadata tool for ReLoad currently under development. APIS will reference services such as the OKI Assessment and Grading OSIDs and the IMS Web Services specification during the development of the runtime API in order to ensure maximum consistency with these initiatives.

The project will also examine the integration of QTI with the IMS Learning Design and Simple Sequencing specifications. This will be addressed in the *IMS Question and Test Interoperability: Item v2.0 Integration Guide*³. There are many use cases supporting the integration of these specifications, which enables adaptive teaching and assessment, and learning which is active and interactive rather than passive and computer focused. The outcomes of QTI activities can be used to determine future pathways through non-QTI learning objects; alternatively, learning activities undertaken within a learning design or simple sequencing activity can affect subsequent assessment activities presented to the learner.

³ This document is currently still in private draft form; the observations here draw on this work by IMS and, in particular, Colin Tattersall of the OUNL.

The lack of a clear method of integration with QTI is flagged in the best practice guides of both Learning Design⁴ and Simple Sequencing⁵, and these issues are currently being explored by the QTI v2 working group. The IMS Shareable State Persistence specification may be the most appropriate way of facilitating integration, and will be one option considered by the project. This will also involve looking at the Content Packaging specification as it is used to combine the other specifications.

In systems where Learning Design and QTI delivery is handled by a single product, the new guidelines, which will be included in the QTI v2 best practice guide, will be sufficient. In modular systems such as APIS, however, a further standardised communication mechanism will be required to allow Learning Design and Simple Sequencing modules to request outcomes from the QTI module. This information, for items, will be included in the Web Services definition for APIS. The APIS documentation will also provide suggested strategies and guidance on how to integrate the APIS engine with systems that support Learning Design and/or Simple Sequencing.

As a consequence of the scope of QTI v2, APIS will have no direct support for sections and assessments. However, it is envisaged that any system supporting a future QTI v2 section and assessment specification will be able to integrate with APIS in an identical manner to Learning Design systems.

9. Technical Development

Both the item engine and supplementary segments of APIS will be written in Java, and a UML editor will be used for modelling and documenting the system. The APIS item engine will be written as a module that can be linked with different façades for specific deployments. One façade will bind APIS to OKI OSIDs so that it can be used with other OKI applications, and another will provide a simple to use interface for developers of stand-alone Java applications and of servlets for use with more basic infrastructures.

Systematic programming and testing methods will be employed, with beta testing involving members of the QTI community not directly involved with the project team. Test plans will be developed for testing the extremes of the system.

10. Intellectual Property Rights

- IMS specifications: IMS makes these freely available for use.

Project Resources

11. Project Partners

The CETIS Assessment Special Interest Group will be involved with supporting and promoting the project. The APIS team will also work with associated ELF projects on the integration model.

⁴ IMS Learning Design Best Practice Guide page 5 - "The LDWG will explore with the QTI WG how best to integrate the QTI Specifications into the Learning Design Specification."

⁵ IMS Simple Sequencing Best Practice Guide page 7 - "More work is needed to determine how information flows between QTI and sequencing, but the mastery objective reference ID number should provide the basic link to make this happen."

12. Project Management

The project will be managed by the department of Learning Services, part of the Information Resource Directorate at the University of Strathclyde. The project team will report to Niall Sclater, project director, who will report to the Director of Learning Services.

Niall Sclater, Assistant Director – Learning Services, Room 5.02, Alexander Turnbull Building, 155 George Street, Glasgow G1 1RD; tel +44 (0) 141 548 3496, fax +44 (0) 141 548 4216, niall.sclater@strath.ac.uk.

Niall Sclater, Assistant Director of the Learning Services department at the University of Strathclyde, will be the project director. The department and institution has considerable experience in supporting project work, including several JISC projects such as CETIS SIGs, the TOIA X4L project and dissemination for the ReLoad X4L project. Niall will have overall responsibility for ensuring that the aims of the project are met and will act as an advisor on the project. Niall has significant project management experience, having lead the SHEFC-funded Clyde Virtual University project from 1995 – 1999, the Scottish Computer Assisted Assessment Network between 1999 and 2001, and currently leads the JISC X4L-B Technologies for Online Interoperable Assessment project.

Niall Barr, APIS Developer, Room 5.18, Alexander Turnbull Building, 155 George Street, Glasgow G1 1RD; tel +44 (0) 141 548 2298, fax +44 (0) 141 548 4216, niall.barr@strath.ac.uk.

Niall Barr will have primary responsibility for the progress of the project. He has been involved in computer based learning since 1993, and has nine years' experience in developing online assessment systems. He has direct and detailed experience of implementing the IMS QTI specification. Niall is part of the IMS QTI v2.0 project team, and has contributed directly to the development of the specification. He has also been the technical supervisor for a number of java-based student projects.

Dr Rowin Cross, CETIS Assessment SIG Coordinator, Room 5.18, Alexander Turnbull Building, 155 George Street, Glasgow G1 1RD; tel +44 (0) 141 548 2298, fax +44 (0) 141 548 4216, rowin.cross@strath.ac.uk.

The project will be managed by Rowin Cross, who will also provide administrative and dissemination support to Niall and will be responsible for coordination of and reporting on the project. As Coordinator of the CETIS Assessment SIG, she is committed to the sharing of knowledge and the development of best practice in online assessment. She is a member of the IMS QTI project team and has contributed to the development of the specification, and also sits on the British Standards committee which is currently reviewing BS7988, 'A code of practice for the use of information technology in the delivery of assessments'. The time spent on the project will vary at different stages of the project, and is likely to be greatest at the start and end of the project. It is anticipated that on average over the course of the project around five hours per week will be spent on this.

The project team are keen to receive training in UML.

13. Programme Support

- Training in UML
- Access to expertise across the programme, including discussion fora for projects members.
- Tools and resources.
- Training in WSDL and webservice.

14. Budget

See Appendix A.

Detailed Project Planning

15. Workpackages

See Appendix B.

16. Evaluation Plan

Timing	Factor to Evaluate	Questions to Address	Method(s)	Measure of Success
Months 5 - 6	Conformance with specifications	Does the engine adhere to relevant specifications?	Input from QTI community.	Successful evaluation by QTI community.
Months 5 - 6	Testing whether the engine actually functions	Does the engine work?	Test plans developed by project manager, beta testing from community	Engine performs as intended
Ongoing	Interest from the community	Are the CAA and QTI communities interested and involved in the project?	Discussion at CETIS Assessment SIG meetings; updates on SIG mailing list, maintenance of website	Community is involved and interested in project. Integration of functioning APIS modules by others into their software.

17. Quality Assurance Plan

Timing	Compliance With	QA Method(s)	Evidence of Compliance
	Fitness for purpose	Beta testing modules	Modules function as expected.
	Adherence to specifications	Review of APIS modules by SIG members	Modules agreed to adhere to specifications
	Adherence to standards	Adherence to W3C, etc, quality assurance and conformance testing systems	Modules agreed to adhere to standards
	Accessibility legislation	Input from TechDis, accessibility specialists, CETIS Accessibility SIG.	Modules agreed to meet or exceed accessibility requirements.

18. Dissemination Plan

Timing	Dissemination Activity	Audience	Purpose	Key Message
months 1 - 6	Promotion of the project through the CETIS Assessment SIG	UK Further and Higher Education	Awareness raising	
1 - 6	Information on release of modules	UK Further and Higher Education;	Sharing resources	

		worldwide QTI developer community		
--	--	-----------------------------------	--	--

19. Exit/Sustainability Plan

Project Outputs	Action for Take-up & Embedding	Action for Exit
Open source code libraries for a core QTI v2.0 item rendering engine	Uploading to development bay, development of guidance materials to support uptake, dissemination	Ensure all code is fully documented before being uploaded; upload code libraries.
UML model of assessment engine interactions	Publish and publicise to community	
Documentation on the implementation of code libraries	Publish and publicise to community	Ensure any feedback received is implemented; version tracking
UML model of API interactions required to support the delivery of functionality provided by APIS segments	Publish and publicise to community	
API to give runtime support for assessment rendering engines	Publish and publicise to community	
Documentation on the integration of IMS QTI, Simple Sequencing and Learning Design	Publish and publicise to relevant community groups; seek further input from developers	Ensure feedback received is recorded and disseminated; version tracking
APIS interaction applet code	Uploading to development bay, development of guidance materials to support uptake, dissemination	Ensure all code is fully documented before being uploaded; upload to development bay
Model of integration of applets with core system	Publish and publicise to community	
Project information resources such as project webpage and blog	Publicise to community	Ensure adherence with JISC maintenance and archiving requirements
Presentations to relevant community groups		
APIS conformance statement	Publish and publicise to community; seek input from community	Ensure feedback received is recorded and disseminated
Final report on project		Forward to JISC

List any project outputs that may have potential to live on after the project ends, why, how they might be taken forward, and any issues involved in making them sustainable in the long term.

Project Outputs	Why Sustainable	Scenarios for Taking Forward	Issues to Address
Modular assessment engine	Modular structure means system can be expanded to suit particular requirements such as mathematics extensions	Seek further funding to develop specialist modules such as mathematical functionality	No formal standard or specification available for mathematics extensions at present, although some attention is being paid to this in QTI v2.0
Integration guide with IMS Simple	Closer integration and co-	Further examine and implement the	No existing practice to learn from

Sequencing and Learning Design	implementation of specifications is likely in the near future	specifications together	
--------------------------------	---	-------------------------	--

Appendixes

Appendix A. Project Budget

	JISC Contribution Requested			Institution Contribution			Total
	YR1	YR2	YR3	YR1	YR2	YR3	
Staff							
<i>APIS Developer (ALC2 spinal point 13 x 6 months plus oncosts, 1.0 FTE)</i>	£18,126						
Travel & Subsistence	£1,500						
Equipment	£2,000						
Dissemination activities	£750						
Evaluation activities							
Other							
Consultancy work on integration of QTI, LD and SS	£5,000						
Total	£27,376						
Total requested from JISC	£27,376						

Appendix B. Workpackages

WORKPACKAGES	Month	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1: Development of APIS engine		■	■	■																					
2: Scoping of API			■	■	■																				
3: Development of API					■	■	■																		
4: Development of APIS interaction applet(s)						■	■																		
5: Dissemination		■	■	■	■	■	■																		
6: Exit strategy							■																		
7:																									
8:																									
9:																									
10:																									

Project start date: 10-05-2004

Project completion date: 29-10-2004

Duration: 6 months

Workpackage and activity	Earliest start date	Latest completion date	Outputs (clearly indicate deliverables & reports in bold)	Milestone	Responsibility
YEAR 1					
WORKPACKAGE 1: Development of the APIS item rendering engine. <u>Objective:</u> To produce and open source code for a core QTI v2,0 item rendering engine.	10/05/04	23/07/04	Code libraries for the APIS item rendering engine.	Uploading of core libraries to development bay	NB
1. Scoping of workpackage	10/05/04	21/05/04	UML model of engine interactions	Publication of UML model	NB - development RC - dissemination
2. Code development	24/05/04	09/07/04	Code libraries for the APIS item rendering engine uploaded to the JISC development bay.	Uploading of code to development bay	NB
3. Testing of engine	12/07/04	23/07/04	Documentation on the implementation of code libraries, known errors, deviation from QTI specification where applicable.	Release of documentation	NB - testing RC - dissemination
4.					
5.					
6.					
7.					

WORKPACKAGE 2: Scoping of API <u>Objective:</u> Research design of API to provide runtime support for the assessment engine, including interaction with IMS Learning Design and Simple Sequencing.	07/06/04	06/08/04	UML model of API interactions required to support the delivery of functionality provided by APIS segments, including statement on the integration of QTI with IMS Learning Design and Simple Sequencing.	Publication of UML model and integration statement	NB - research and development RC - research and development, dissemination
8. Research into integration of IMS QTI, LD, and SS	07/06/04	09/07/04	Statement on the integration of the specifications.	Distribution of statement	NB RC
9. Research and review of APIs in elearning technology.	05/07/04	06/08/04	Paper on the use of APIs in the field.	Research paper produced	NB
10. Modelling of API			UML model of interactions	Distribution of UML model	NB
11.					
WORKPACKAGE 3: Development of API <u>Objective:</u> Release of proposed API for assessment runtime support	16/08/04	8/10/04	Proposed API to give runtime support for assessment rendering engines released to the community and to relevant standards and specifications bodies	Release of API	NB - development RC - dissemination
12. Development of API	18/08/04	24/09/04	Code libraries	Uploading of code to JISC development bay	NB
13. Testing of API	13/09/04	0/10/04	Code available	Uploading of code, dissemination of test results	NB - development RC - dissemination
14.					
15.					

<p>WORKPACKAGE 4: Development of APIS interaction applet(s)</p> <p><u>Objective:</u> To release at least one pluggable applet to provide additional functionality for the core APIS engine.</p>	(06/09/04)	variable	Applet code uploaded to JISC development bay	Uploading of code	NB
16. Development of applet for 'hotspot' item interactions	(06/09/04)	to be determined	APIS hotspot applet	Release of applet	NB
17. Identification and development of further applets	to be determined	to be determined	to be determined	Release of applets, if applicable	NB
<p>WORKPACKAGE 5: Dissemination activities</p> <p><u>Objective:</u> To share knowledge and experiences with the UK FE and HE communities and promote the JISC frameworks programme</p>	10/05/04	29/10/04	Increased awareness by the community of the programme as a whole and of the role and function of the APIS project in particular. To encourage use and sharing of resources produced by the project and distributed through the JISC development bay.	various	RC NB NS
18. Development and promotion of project webpages including project blog and links to JISC resources	10/05/04	29/10/04	Information resource	Resource made available	RC
19. Presentation to CETIS Assessment SIG quarterly meetings	to be confirmed		Presentations on the progress and to-date outcomes of the project	Presentations delivered	NB
20. Participation in programme demonstration event	26/10/04		Presentation on the outcomes of the project	Presentation delivered	NB
<p>WORKPACKAGE 6: Exit strategy</p> <p><u>Objective:</u> To review activities, ensure targets have been met and complete reporting.</p>	04/10/04	29/10/04			
21. Review of conformance with target standards and specifications	04/10/04	29/10/04	APIS conformance statement	Publication of conformance	NB - evaluation

				statement	RC - dissemination
22. Review of project documentation	04/10/04	29/10/04	APIS documentation	Release of APIS documentation	NB, RC, NS
23. Review of progress	04/10/04	29/10/04	Final report on project	Delivery of final report to JISC	RC, NB, NS
24.					
YEAR 2					
WORKPACKAGE 7:					
<u>Objective:</u>					
25.					
26.					
27.					
28.					
WORKPACKAGE 8:					
<u>Objective:</u>					
29.					
30.					
31.					
32.					
WORKPACKAGE 9:					
<u>Objective:</u>					
33.					
34.					

WORKPACKAGE 10:					
<u>Objective:</u>					
35.					
36.					
37.					

Members of Project Team:

NB = Niall Barr, Developer
RC = Rowin Cross, Coordinator
NS = Niall Sclater, Project Manager

JISC Project Management Framework
 22 December 2003