JISC

JISC Final Report

Project Information			
Project Title	Virtual Infrastructure with Database as a Service		
Project Hashtag	#vidaas		
Start Date	01/05/2011	**End Date**	31/03/2012
Lead Institution	University of Oxford		
Project Director	Professor Paul Jeffreys		
Project Manager	Dr. James A J Wilson		
Contact email	james.wilson@oucs.ox.ac.uk		
Partner Institutions	n/a		
Project Web URL	http://vidaas.ox.ac.uk/		
Programme Name	*UMF Shared Services and the Cloud*		
Programme Manager	David Utting		

Document Information			
Author(s)	James A. J. Wilson, with contributions from VIDaaS Team members		
Project Role(s)	Project Manager		
Date	03/04/2012	**Filename**	VIDaaS_FinalReport.pdf
URL	http://vidaas.oucs.ox.ac.uk/docs/VIDaaS_FinalReport.pdf		
Access	public		

Document History		
Version	**Date**	**Comments**
0.2	29/03/2012	Structured document & added output & some narrative text
0.3	03/04/2012	Pre-Easter almost complete draft
0.4	03/04/2012	With corrections and Business Plan added as an appendix
0.9	20/04/2012	Pre-publication draft, including amendments by team members. For circulation to Steering Group
1.0	11/05/2012	Final version for JISC

Table of Contents

1 Acknowledgements

We would like to thank the JISC and HEFCE for funding the Virtual Infrastructure with Database as a Service (VIDaaS) Project under the UMF Shared Services and the Cloud Programme. We would also like to thank the JISC for funding the project's predecessor, the Supporting Data Management Infrastructure for the Humanities (Sudamih) under the Managing Research Data Programme, which enabled us to begin the development of the Database as a Service. The guidance we have received over the last two years from the members of the Sudamih and VIDaaS Steering Groups has been of enormous benefit, and we would like to thank all of the members of those boards. Finally, I would like to thank all of the members of the VIDaaS Project Team who have all put in a considerable effort to bring the project to its successful conclusion within some very challenging timescales.

2 Project Summary

The Virtual Infrastructure with Database as a Service (VIDaaS) Project developed software enabling researchers to create, edit, search, and share research databases via a simple Web interface. The Database as a Service (DaaS) software is open source and intended to be easily adopted and deployed by HE institutions beyond the University of Oxford.

VIDaaS has also developed a private cloud infrastructure at Oxford on which to host the software. Ultimately it is intended that the software can be hosted on any VMware-based cloud provider, whether public or private. The approach at Oxford has been to develop a 'hybrid' infrastructure that offers a private cloud (suitable for researchers with sensitive data which must be kept within the physical bounds of the University), but which enables bursting out into the wider public cloud as required (when cost is a priority, for example). We have tested the DaaS on both the private and public clouds, and successfully migrated an instance of the DaaS between the Oxford cloud and the Colt public cloud.

In order to be effective, the DaaS software is designed to be deployed as part of a wider service incorporating cloud infrastructure hosting and user support. The VIDaaS Project has planned such a service, which will initially be available to research projects where the Principal Investigator is a member of Oxford University. This service will be called the Online Research Database Service (ORDS) and is due to be launched in the summer of 2012.

This DaaS is intended to realize significant cost savings over current database-supported research practices and to form part of a broader institutional research data management infrastructure supporting the research data life-cycle from planning to long-term curation and re-use. At present the DaaS enables researchers to work with relational databases, although other types of database will also be supported in the future.

3 Main Body of Report

3.1 Project Outputs and Outcomes

Output / Outcome Type	Brief Description and URLs (where applicable)
Virtual Infrastructure (VI) based at the University of Oxford upon which SaaS can be hosted	The University's Private Cloud, based on VMware infrastructure is now built and tested and will be launched as a service during April 2012. http://www.oucs.ox.ac.uk/sis/cloud/
Database as a Service (DaaS) software enabling researchers and the wider public to create (and import) databases, collaboratively add and edit data, perform searches, and export data for further analysis or visualization	All core functionality is now in place, however the software needs to be migrated to a more secure application platform before it can be released officially. The DaaS source code is available from Google Code at http://code.google.com/p/vidaas/.
'Shared Services' that will enable SaaS (such as the DaaS) to run on the virtual infrastructure. This will include tools for: - Capacity management - Identity and access management (IAM) - Monitoring usage, accounting, and billing	We have produced IAM software enabling the DaaS to be securely accessed via Single Sign On with Shibboleth. There are various 'roles' which can be assigned to users, from Project Owner to Viewer, each of which grants different access rights. We have also developed billing software that automates payments and the sending of reminders. We have decided to use VMware's default software for capacity management and usage monitoring, although we intend to develop our own service monitoring tools for the forthcoming Online Research Database Service (ORDS), which will be developed during a follow-on project (the ORDS Maturity Project).
A project website and blog, hosting news of findings and developments and acting as a publicly-accessible repository of project reports and other outputs.	The VIDaaS Project website is available from http://vidaas.oucs.ox.ac.uk/. Significant reports and presentations are available from http://vidaas.oucs.ox.ac.uk/outputs.xml. The blog is hosted at http://blogs.oucs.ox.ac.uk/vidaas/.
A literature review summarizing the current state of research into academics' use of research databases.	Available from http://vidaas.oucs.ox.ac.uk/docs/VIDaaS%20Literature%20Review%20v1.0.pdf We have also created literature review bibliography groups on Zotero and Mendeley, which can be accessed at https://www.zotero.org/groups/vidaas_project_bibliography/items and http://www.mendeley.com/groups/1356623/vidaas-project-bibliography/ respectively.

	We have been using Diigo to manage bookmarks for relevant websites: http://groups.diigo.com/group/oucs-vidaas. This is a publicly-accessible tool.
A brief user-requirements report	http://vidaas.oucs.ox.ac.uk/docs/VIDaaS%20Researcher%20Requirements%20Report.pdf
A prioritized requirements list that will help the software developers focus on key user requirements	This was completed, covering 71 requirements in total, which were rated according to the MoSCoW prioritization system (Must haves, Should haves, Could haves, Won't haves). The document was restricted to internal use during the project, but we will be considering the remaining functional requirements and producing a public road-map of future developments once the DaaS has been migrated to a new platform. This will appear on the website in due course.
A draft service-level description (SLD) for the forthcoming Online Research Database Service (ORDS).	http://vidaas.oucs.ox.ac.uk/docs/VIDaaS%20SLD%20v1.pdf
A DaaS installation and set-up guide	http://code.google.com/p/vidaas/wiki/ViDaasSetup. n.b. these installation guidelines will change once the DaaS is migrated to a new platform.
Training materials relating to the use of the DaaS, aimed at academic researchers	During VIDaaS we have updated several of the training materials produced initially for the Sudamih Project to refer to VIDaaS developments and the forthcoming ORDS service: http://sudamih.oucs.ox.ac.uk/documents.xml. We have also produced how-to guides to performing common database actions using the system, which are available from http://vidaas.oucs.ox.ac.uk/documentation/index.xml (these will require further updating before ORDS is launched, so the documents are not fully public). We also ran two face-to-face courses using the new materials.
Survey of researchers who have set up research databases in the past	Towards the end of the VIDaaS Project we conducted a second researcher survey in order to capture information about researchers' experiences with research databases to feed into the benefits assessment. This attracted 34 complete responses. Whilst the results of the first survey were written up in the Researcher Requirements Report, the second survey contains data that has not been fully analysed and published. Anyone who is interested in looking at this data (in anonymized form) should request it from vidaas@oucs.ox.ac.uk.
An updated Return On Investment (ROI)	This forms part of the business case appended to this report. See Appendix A - VIDaaS Sustainability and

	Business Case
A business plan for the operation of the DaaS beyond the lifespan of the VIDaaS Project	The full business and sustainability plan is appended to this report. See Appendix A - VIDaaS Sustainability and Business Case
A final Project Report summarizing the achievements of the project and lessons learnt, with recommendations for future developments	This document is the final report.
Improved awareness of good data management practice	We have engaged in several activities during the VIDaaS Project that have required people to consider the processes involved in data management and how they might be improved. These most obviously include interviewing and presenting to researchers, but also presentations to and conversations with other support staff working in the Libraries, Research Services and our colleagues within the Computing Services.

3.2 How did you go about achieving your outputs / outcomes?

The VIDaaS Project effectively followed on from the Sudamih Project, which was funded under the JISC's Managing Research Data Programme and ran from October 2009 to March 2011. Sudamih had two main aims – to develop a pilot 'Database as a Service' (DaaS) and to produce training materials to help improve the data management practices of researchers. Sudamih focussed on the needs of the Humanities Division at Oxford University, whereas VIDaaS has sought to extend the disciplinary scope to look at requirements across the research divisions. By the end of the Sudamih Project we had the basic DaaS system in place, albeit without a search capability or other advanced functionality, and only running on a conventional physical server.

During the first months of the VIDaaS Project we pursued a number of strands in parallel. Whilst we had a firm knowledge of what our humanities researchers wanted from the DaaS, we could not be certain that these requirements were representative across the University as a whole, so an additional researcher requirements analysis was conducted. We also needed to work on developing the University's private cloud infrastructure in order to host the service. As the new hosting requirements would also require some significant changes to the back-end of the DaaS software, we also began work on that in parallel.

3.2.1 Requirements Gathering

The VIDaaS requirements gathering process had two main phases: the collection of qualitative information through a series of interviews with researchers, and the collection of quantitative data via an online survey. Both interviews and survey were designed to explore two main areas: researchers' current projects (including details of any datasets they had created or managed), and their potential interest in and user requirements for the DaaS. As the DaaS is designed to form a service that will allow researchers to share and publish data, questions gauging researchers' attitudes to data sharing were also included. In June and July 2011 we interviewed nine University of Oxford researchers currently working with structured data. Potential interviewees were identified via personal profiles on the University website or elsewhere online, and were then emailed to ask if they would be willing to participate.

In the second half of July 2011, we conducted an online survey. To gain a wider range of perspectives, two versions of this were provided: one aimed at researchers, and the other at IT support staff. 62 responses were received in total (although one of these appeared to be spam and was excluded from the analysis), approximately two thirds of the responses were from researchers, and one third from IT support staff.

The results of both exercises are explained in detail in the VIDaaS Researcher Requirements Report (http://vidaas.oucs.ox.ac.uk/docs/VIDaaS%20Researcher%20Requirements%20Report.pdf), but the main findings included:

- Researchers liked the idea of the DaaS
- Researchers in the Humanities and the Social Science Divisions seemed the most enthusiastic
- It would be best to catch researchers just before they embark on a new project when encouraging them to use a tool such as the DaaS
- Cost and functionality, unsurprisingly, would be the most important factors in deciding whether to use a tool such as the DaaS. Other key requirements included:
    - Automated back-up
    - The ability to import and export data in a range of common formats
    - The ability to view and present data in different ways
    - The ability to set different permissions levels for different users/collaborators
    - The ability to make data (or subsets of data) publicly available via the Web in a straightforward manner
- Researchers were particularly interested in the promise of the DaaS to simplify collaboration, especially where that could include colleagues at other universities, and in the possibility of publishing datasets to accompany research publications in a way that rendered them citable (i.e. with a persistent URL or DOI).

3.2.2 Virtual Infrastructure development

With regards to setting up the Virtual Infrastructure, VIDaaS was fortunate to have two members of its team with many years worth of experience of working with VMware to design the initial infrastructure. They approached key hardware vendors to evaluate the solutions available for cloud platforms based upon the VMware technology stack and after several discussions selected a vendor and procured a suitable hardware solution. In parallel to this activity they engaged closely with VMware directly and the project was fortunate enough to be selected as one of two EMEA (Europe, Middle East, Africa) projects for the VMware Activate program. Essentially, VMware agreed to provide the University with significant help setting up the infrastructure in exchange for using the VIDaaS Project as one of their early adopters for their own marketing purposes. A side effect of this work was that the project gained significant publicity at both the US and European VMworld conferences, amongst the largest IT conferences in the world. Press interviews given by the VIDaaS Project staff appeared in Computer Weekly and Tech Week Europe amongst other magazines.[1]

[1] http://www.computerweekly.com/news/2240106539/VMworld-2011-Oxford-University-demos-VMware-based-database-as-a-shared-service;

As a further benefit of the project, the University of Oxford has built relationships with the team within VMware who are developing mechanisms to connect clouds together, and a similar database as a service offering (VMware vFabric Data Director). We are considering ways in which we can collaborate more closely in the future to support our data management infrastructure.

3.2.3 DaaS Software Development

In terms of the DaaS software development, the early stages of the project involved preparing the software to run on the VMware back end. In July 2011, based on the results of the requirements gathering work we had undertaken, the project produced its prioritized list of user-facing requirements. In consultation with the software developers, we went through each of the requirements and decided whether it was feasible to implement within the scope of the project, and if so how important it was to the project's success. Some requirements were felt to be outside of the scope of the project, whilst others, although they could be facilitated by technical development, had workarounds that meant they could more efficiently be dealt with via documentation for the time being, so that development effort could be concentrated elsewhere. The second requirements list categorized each of the original requirements according to the MoSCoW prioritization system, with the requirements being assigned to the 'must have', the 'should have', the 'could have', and the 'won't have' categories. Considering the full list of user requirements (71 in the first instance) brought home the fact that our timescale for development had been extremely ambitious, and many of the non-essential aspects of functionality could only realistically be added after the VIDaaS Project itself had concluded if we were going to have time to ensure the basic elements were appropriately secure and robust.

Even though the Prioritized Requirements List provided a clear road map of required user-facing functionality, we were aware that we would need to remain flexible during the actual implementation of the functionality. We adopted an 'Agile' approach to development, with a high-level plan regarding which aspects of functionality should be implemented when. Within this the development team was expected to flesh out the detailed implementation schedule and produce new software releases every fortnight for the user testing team to test and provide feedback on. In practice this approach proved difficult to adhere to (see section 3.3, lessons 1 and 2).

By the end of the project we had implemented the following:

- Project registration interface
- Database creation interface
- Database designer drag-and-drop interface (for building new database structures or modifying existing ones)
- Database export facility (as .csv files)
- Data editing interface (for adding, editing, deleting, and linking records)

---

http://www.techweekeurope.co.uk/news/oxford-university-builds-vmware-private-cloud-43123; also see the VMware blog post at http://blogs.vmware.com/vcloud/2012/04/another-vmware-cloud-oxford-university-runs-their-hybrid-cloud-on-vmware.html.

- Ability to create test and milestone versions of databases (and swap them with the ‘live’ version)
- Data filtering interface
- SQL-search function
- Ability to create XML databases and add documents to them (but not yet to search them)
- Metadata editing interface
- Ability to add and remove project members and set viewing/editing permissions

We had also adapted the software to run on cloud infrastructure, spinning out new virtual machines as required.

The DaaS software is open source and licensed under the Lesser GPL licence. Other institutions wishing to set up their own instance of the DaaS are free to do so (although they too may wish to wait until after we have completed the migration process before launching a service based upon the software). The software and its associated documentation and issues log are freely available from the VIDaaS Google Code repository at http://code.google.com/p/vidaas/.

3.2.4 Designing the future service

We were aware from the beginning of the project that for the DaaS to be taken up and run elsewhere we would need to prove it could be provided as a sustainable service within the University of Oxford, and we had already known since our initial data management requirements surveys of researchers at the University that there should be strong internal demand.

We produced a draft Service Level Description relatively early in the project (August 2011), partly so that the team could get a better sense of what the service would actually need to do whilst it was still in development, and partly to start getting a sense of support requirements. The outline governance document followed shortly afterwards, again giving a better sense of what shape the eventual service would need to take.

We also invested significant time in planning and documenting requirements relating to Identity and Access Management (IAM), which arguably paid off in the subsequent relatively rapid development of the shared services during the first quarter of 2012. Our final system consists of an integrated Shibboleth/Single-Sign-On system that should be able to deal securely with researchers registered at UK HE institutions.

The roles available to users of the DaaS are as follows:

- VIDaaS administrator – able to administer the VIDaaS system but not view project data.
- Project owner – able to create and administer new projects within their VIDaaS view. Only a single project owner is allowed per project. The project owner has the authority to add other project members as administrators, contributors or viewers.
- Administrator – able to perform administrative operations on a project on behalf of the project owner. An administrator has the authority to restructure a database.
- Contributor – allowed to create, edit, and delete data within databases.
- Viewer – able to view all data within all databases for their assigned project.
- General public – if desired by the project owner or administrator, the general public shall have access to databases or to subsets of databases (the outputs of queries) defined by project owners, administrators, or contributors.

The registration and billing processes were actually the last parts of the service to be formally designed – partly because we wanted to get an idea of how the pricing model should work first (which depended in turn on the costing model), partly because we needed to understand exactly what metadata we could and should try and capture during the process (which had a partial dependency on the user roles and IAM aspects of the service) and partly because we wanted to get user feedback regarding the ‘dummy’ registration process that our developers had mocked up earlier in the project. With hindsight, we might have attempted to tackle the registration process more fully earlier in the project, as it needed to be substantially re-worked, which took development resources away from finishing other aspects of functionality at an important time (shortly before our public workshop at the beginning of March 2012).

3.2.5 Software testing

We started the project planning to rely mostly on regular testing by a volunteer testing group of researchers based in Oxford and at other UK Universities. The idea was that with each development cycle we would email the members of the testing team and ask them to devote an hour or two to testing the current release and feeding back any bugs or issues they encountered via our Google Code repository. Assembling the testing team went very smoothly. We concluded the early workshops and training/induction events that we participated in with a plea to any researchers working (or planning on working) with databases to get involved with the project – and many agreed to do so. By August 2011 we had 15 volunteers in place, which we hoped would be more than enough to winkle out even the most obscure bugs.

In practice however, two issues arose with this approach. Firstly, our technical development team did not adhere rigidly to fortnightly releases. When introducing new functionality we invariably also introduced new bugs, and there were few windows of opportunity for us to send out testing scripts where we knew that the testers wouldn’t very rapidly run into some major bug or another that we were already fully aware of and already working on. Secondly, all of our testers were busy with their own research and had little time to put aside to testing a half-built system. Consequently we received very little feedback even when we did send out emails. It was generally more effective for the project team to look for bugs themselves. Furthermore, as we did not wish to risk losing the buy-in of our testing volunteers, we were reluctant to send out testing scripts unless there were major new developments to demonstrate which would keep them interested.

In order to overcome these issues, and still get helpful user feedback regarding the interfaces and processes, we switched our attention instead to organizing user observation sessions in controlled conditions. After researching user observation testing online and seeking advice from colleagues, our project analyst opted to follow the methodology outlined in Steve Krug’s *Rocket Surgery Made Easy* (New Riders, 2010). This involves focused testing with a small number of users. Each tester is asked to explore the system under review and carry out some tasks, and is invited to ‘think aloud’ while doing so: this produces a lot of in-depth information about their reactions to the system, what they find confusing, and what can be improved. Two half-day testing sessions were organized, using testers from a mixture of academic backgrounds, and including people who already knew something about the DaaS (including members of the larger testing group) and those who were new to it. The results were then written up into categorized issues lists. This method proved to be significantly more effective than our emailed testing scripts.

3.2.6 Running the DaaS on the Virtual Infrastructure

Thanks to the early work on adapting the DaaS to run on virtual infrastructure, and the assistance of VMware in constructing the Oxford Private Cloud, we managed to start testing the DaaS on actual live virtual infrastructure relatively rapidly. By August 2011 we had an instance of the DaaS hosted on the University VI and successfully migrated it onto the Colt Public Cloud and back – an important proof of concept. By January 2012 we had successfully tested the DaaS on the Eduserv cloud, and in February we moved the in-development ORDS service to the now fully-implemented Oxford Private Cloud. Our live demonstration of the software in the VIDaaS/DataFlow workshop on the 2nd March 2012 involved running the system on the private cloud, spinning up a new virtual machine and moving a database to it in front of a live audience.

3.2.7 Documentation and training

During VIDaaS we updated the training materials we had originally produce for the Sudamih Project (including induction materials for new doctoral students and post-doctoral staff) as well as producing contextual help for the DaaS software and a set of how-to guides. The fact that DaaS feature development took longer than we had initially hoped led to delays to the documentation and training work-packages, and forced the project to adapt its initial plans accordingly.

Project staff drew up a list of the ten most crucial activities that a user was likely to undertake when using the DaaS. Each of these activities would then form the basis of a How-To, a step-by-step guide describing how to undertake that particular activity. For each How-To, the first step was for our analyst and technical writer to learn how to achieve that task within the application themselves. The steps were then documented and screenshots taken. Each draft How-To was stored as a Word document in the Project SharePoint site, which allowed staff members to collaborate on each other's documents. Once a How-To was finished, it was then converted into XML and made available as a Web page on the VIDaaS website. We also implemented a basic search function on the VIDaaS Web pages using Oxford's Google Search Appliance.

The user observation testing sessions provided some particularly useful feedback on potential improvements to the on-screen labelling and assisted the technical writer in drafting revised labelling text and explanatory text to be used as pop-up contextual help, which was fed back to the developers for implementation.

3.2.8 Costing the ORDS

One of the most important aspects of taking pilot software to a sustainable service is getting the costing and pricing right. We will not know with any certainty whether we have done so until the launch of the ORDS in the summer, but significant effort was put into this aspect of the project. The first step was to cost the Oxford Private Cloud infrastructure, to be hosted by the University of Oxford's new Shared Data Centre (SDC) (with the old data centre used to provide dual-site resilience). As this was being equipped during the early stages of the VIDaaS Project, we could get a clear sense of the hardware costs involved and likely rates of depreciation and therefore future capital renewal. The ORDS is being regarded as a significant driver of uptake of the new private cloud service provided by the SDC, so we based our ORDS cost estimates on the assumption that it would account for about 50% of the occupancy of the cloud and contribute 50% towards its capital renewal and staffing costs. At this level, the service would be expected to contribute £32,250 per annum to capital renewal, and £73,900 per annum to the cloud service costs, including power, back-up, licensing, and so forth.

Given that it will be our Systems Development team at the Computing Services that will be expected to take on ownership of the ORDS post-project (at least until the University has a dedicated research data management team in place), we consulted with the head of SysDev regarding future service staffing requirements, both at service launch and as the number of users increased. The initial required staffing levels amounted to upwards of 1.4 FTEs, with 0.4 FTE of that time dedicated to further feature development – something that will inevitably be necessary over the next few years and which will help speed the rate of uptake as a 'long-tail' of researchers with less generic requirements can be catered for. The full costs of service staff at this level amount to £142,717 over the first year.

The variable cost components of the ORDS service worked out as proportionally low, consisting of increased help-desk support, additional hardware beyond a certain level of uptake, and the costs of VMware licences. A more detailed analysis of the cost components can be found in Appendix A - VIDaaS Sustainability and Business Case.

Once we had established the costs of ORDS, we could then get a sense of the prices we would need to offer the service at in order to recoup the costs. Due to the high fixed-cost component of the costs profile, this was not an easy task. We began by creating a best-guess uptake model, so we could see where the break-even point would fall at various prices. The challenge was to find a price that would enable the service to break-even within a reasonable timescale whilst being low enough for researchers to confidently enter into their research proposals. We settled on a price of £600 pa for a 'regular' VM model, and £2,000 pa for the 'large' model, although these may need to be adjusted once the service is launched and we get a better idea of the actual take-up and the prices that people are willing to pay.

The final element of the costing work was to assess the value of the benefits that the ORDS would bring to the University. At around the middle of the VIDaaS Project we had staged a brainstorming session to get all members of the project team to come up with as many costs and benefits as they could think of. The benefits were then compiled and rationalized into a list, which was included in early drafts of the Project's sustainability plan. Due to time pressures, however, we only really started to try to quantify the benefits towards the end of the project. This was one of the most difficult aspects of the project, as many of the solutions that researchers currently employ when creating databases are not directly comparable to what ORDS offers, and also because many of the benefits of the ORDS are inherently difficult to measure in financial terms. The results of our analysis are reported in Appendix A.

The ORDS will be offered by the University as a 'Small Research Facility' which researchers can cost into grant applications (see http://www.admin.ox.ac.uk/researchsupport/facilities/).

3.2.9 Communication and dissemination

The communications work package began by considering in some depth the stakeholder groups that VIDaaS needed to be involved with and how exactly the project should reach them to the best effect. An expanded stakeholders list was placed on the project SharePoint site identifying 13 groups in total, and a 6-page communications plan was produced indicating how each group should be involved. Besides the standard project website and blog, we set up groups using Zotero, Mendeley, and Diigo to share our bibliography and research conclusions, maintained internal contact with our training staff, Infodev team,[2] and research support teams, compiled a mailing list of individuals and organizations to be updated via a quarterly email bulletin, and identified public dissemination channels.

Representatives from high-priority stakeholder groups, such as the DCC, Eduserv, JISC, the academic divisions, and UCISA were invited to sit on the Project Steering Group, whilst others, such as the OSS Watch team and our sister projects in the UMF programme were involved in specific activities such as workshops. We spoke about the project and our ambitions for the DaaS at a number of internal training and induction sessions, and external workshops, and conferences (see the list below). We have worked on two papers during the project, the first of which will be presented at the 2nd International Conference on Cloud Computing and Services Science in Porto in April 2012, and the second, less technical paper, will be submitted to an international journal in due course. We have coordinated our outputs with the JISC Communications Team where appropriate.

3.2.9.1 List of communications and outreach activities during the VIDaaS Project

Project website: http://vidaas.oucs.ox.ac.uk/

Project blog: http://blogs.oucs.ox.ac.uk/vidaas/

Reports:

- VIDaaS Researcher Requirements Report (http://vidaas.oucs.ox.ac.uk/docs/VIDaaS_OriginalProposalnb.pdf)
- Researchers' Database Behaviour and Use: A Literature Review (http://vidaas.oucs.ox.ac.uk/docs/VIDaaS%20Literature%20Review%20v1.0.pdf)

Presentations:

- 'Database as a Service: a Tool for Researcher' (Presentation to Project Bamboo and SurfNet). Oxford, 9th May, 2011.
- 'Improving Research Data Management at the University of Oxford' (presentation to OUP). Oxford, 18th May, 2011.
- 'Managing Data in your Institution' (part of 'Discovering Babel' workshop). Oxford, 24th June, 2011.
- 'Virtual Infrastructure with Database as a Service (VIDaaS) Project' (UMF Implementation Group Workshop). London, 4th July, 2011.
- ‘Real-World Cloud Experiences: Enterprise IT Leaders Discuss Public and Hybrid Cloud Deployments’ (VMworld Conference). Las Vegas, 31st August, 2011.
- 'Using the USDC to support research activities: The Virtual Infrastructure with Database as a Service (VIDaaS) Project' (USDC for Researchers Workshop). Oxford, 7th September, 2011.
- 'Database as a Service - an institutional tool for research data management' (DCC Roadshow, Oxford). Oxford, 14th September, 2011.
- 'Research data management: tools and resources', 'Introduction to the DaaS', and 'Institutional Data Management Tools: Maximizing the Benefits for All'

[2] http://www.oucs.ox.ac.uk/infodev/

(training session and software demonstration at DCC Roadshow, Oxford). Oxford, 16th September, 2011.

- 'Virtual Infrastructure with Database as a Service (VIDaaS) Project' (UMF Workshop, part of All Hands 2011). York, 29th September, 2011. http://vidaas.oucs.ox.ac.uk/docs/UMF29thSeptember_VIDaaS.pdf
- 'Real-World Cloud Experiences: Enterprise IT Leaders Discuss Public and Hybrid Cloud Deployments'. Las Vegas, 31st September 2011 & Copenhagen, 18th October.
- 'Data Management at the University of Oxford' (presentation to the Humanities Postdoctoral Induction Day, including explanation of VIDaaS and appeal for early adopters). Oxford, 5th October, 2011.
- 'Real-World Cloud Experiences: Enterprise IT Leaders Discuss Public and Hybrid Cloud Deployments' (VMworld Europe Conference). Copenhagen, 19th October, 2011.
- VIDaaS Presentation to VMworld Europe press conference. Copenhagen, 19th October, 2011.
- 'Virtual Infrastructure with Database as a Service (VIDaaS) Project' (Knowledge Exchange Workshop). Bonn, 14th November 2011. http://vidaas.oucs.ox.ac.uk/docs/VIDaaS_BonnKnowledgeExchange.pdf
- 'VIDaaS/VMware demonstration'. Oxford, 22nd November, 2011.
- 'Creating Electronic Research Resources at Oxford' (English Graduate training session). Oxford, 29th November 2011.
- 'Virtual Infrastructure with Database as a Service (VIDaaS) Project' (RDM Launch Workshop). Nottingham, 1st December 2011. http://vidaas.oucs.ox.ac.uk/docs/RDMlaunchworkshop_VIDaaS%20update.pdf
- 'Assessing the Costs & Benefits of RDM Projects : Sudamih & VIDaaS' (RDM Launch Workshop). Nottingham, 1st December, 2011. http://vidaas.oucs.ox.ac.uk/docs/MRD_CostsBenefitsApproaches.pdf
- VIDaaS Poster presentation (International Digital Curation Conference 2011). Bristol, 5th December, 2011.
- 'Virtual Infrastructure with Database as a Service (VIDaaS) Project' (first BRISSkit workshop). Leicester, 19th December, 2011.
- 'What our researchers want and what they need: Research Data Management (OUCS staff day)'. Oxford, 12th January, 2012. http://vidaas.oucs.ox.ac.uk/docs/OUCSstaffday2012_DataManagement.pdf
- 'The Online Research Database Service' (PowerPoint presentation displayed at Research Skills Toolkit live events). Oxford, 17th January, 2012.
- ***VIDaaS/DataFlow joint workshop, Saïd Business School, Oxford***. Included the following VIDaaS presentations:
  - Data Management at Oxford (http://vidaas.oucs.ox.ac.uk/docs/Paul%20Jeffreys%20-%20data%20management%20at%20Oxford.pdf)
  - Introduction to and live demonstration of DaaS software (http://vidaas.oucs.ox.ac.uk/docs/VIDaaS_Workshop_morning_NoScreenshots.pdf)
  - The Oxford Cloud Infrastructure (http://vidaas.oucs.ox.ac.uk/docs/VIDaaS_Cloud.pdf)
  - o VIDaaS Future Developments (followed by question and answer session) (http://vidaas.oucs.ox.ac.uk/docs/VIDaaS_Workshop_afternoon.pdf)
- 'The Online Research Database Service' (PowerPoint presentation displayed at Research Skills Toolkit live events). Oxford, 6th March, 2012.

Other outputs:

- Initial VIDaaS Project award press release
- VIDaaS Web Bookmarks (http://groups.diigo.com/group/oucs-vidaas)
- VIDaaS Literature Review (http://vidaas.oucs.ox.ac.uk/docs/VIDaaS%20Literature%20Review%20v1.0.pdf)
- Research Skills Toolkit article on ORDS/DaaS

3.2.10 Project follow-up work

Whilst the VIDaaS Project has achieved almost all of the goals it set out to achieve, there is still much to be done in order to establish the ORDS as a sustainable long-term trusted research data management service and a fully integrated part of a complete-lifecycle research data management infrastructure.

The immediate follow-up work will involve the migration of the DaaS software from the JBoss 5 platform to something more secure. We chose JBoss 5 as our software development platform at the start of the Sudamih Project as it seemed like a good means for our developers to get the DaaS up and running within the short period of time we had available. Unfortunately, over the last couple of years, the company behind JBoss 5 was bought up by Oracle, who released two new versions in rapid succession. The current version of JBoss (version 7) is not backwards-compatible with JBoss 5, and the earlier version is no longer being supported. We had initially hoped to take the DaaS to service running on JBoss 5, and then migrate the software to the new version behind the scenes. Unfortunately, security flaws have since emerged with JBoss 5 that are not going to be patched. Obviously we cannot expose our researchers' data to known risks of this kind so we must perform the migration *before* the ORDS can be formally launched.

Once the DaaS has been migrated to a more secure platform, we can then return to ironing out the remaining bugs and officially launch the service to early adopters (during the summer of 2012). With the core functionality currently in place we believe that the service will meet the requirements of approximately 50% of the research community. In order to expand this potential market we will have to add additional user functionality. We plan to do this via an internally-funded follow-on project called the 'ORDS Maturity Project'. Funding is already confirmed for this project, which will commence as soon as the migration is complete. A road-map of the additional functionality this project will introduce will be published shortly after its commencement.

We hope to acquire additional funding to incentivize uptake from early adopters (by granting them a subsidy) and conduct a more detail study into the benefits of the service once we are in a position to ask our early adopters about this.

3.3 What did you learn?

Completing a project of the size and scope of VIDaaS within a year was not a straightforward task, and it imposed challenges of a kind that stretched most members of the project at one time or another. One of the main lessons that we learnt during the course of the project was that taking pilot software to a full production service requires a lot more work, and a lot more time than developing a prototype, and simply adding extra resources cannot in itself overcome some of the issues faced.

Due to the timescale of the project we had to run a lot of activities parallel to one another which would have been easier to manage could they have been performed in a more sequential manner. Although there were few 'hard' dependencies between the work packages, it was in practice very difficult for the training and documentation work to proceed whilst changes and improvements were being made to the software, but these changes were the inevitable outcome from the ongoing user testing sessions.

In compiling this section of the report I have asked each of the members of the VIDaaS Project team to contribute the lessons that they felt they had learned from participating in the project and assimilated their responses.

1. *Involve development staff early in the planning process; insist that sufficient effort is spent creating good estimates of the development time required for each aspect of functionality; plan a regular software release schedule based on these estimates, and then stick to it.*

Although we had begun the software development work intending to stick to a fortnightly release schedule, in practice this was allowed to drift as the developers responded to bugs or realized additional details were required that only became apparent during the development work itself. We also took too much of an *ad hoc* approach as to the order in which functionality was implemented. Functionality within the application should have been more clearly divided into coherent chunks from an early stage in the project, and signed off when completed. This would have eased some (although not all) of the problems that the testers and documenters faced. Getting the software developers to give a regular (public) demonstration of what they had implemented every two weeks would be a good idea for the future, in order to provide more of a ‘rhythm’ to development work and to better communicate developments to the rest of the project team.

2. *Have a single system for monitoring progress, ensure it makes sense to everyone, and revise and add detail as necessary*

The diverse range of work that VIDaaS covered, along with the large number of people involved in the project, called for more detailed monitoring than is required by most JISC-funded projects. We did actually set up a single GANTT-style spreadsheet to help monitor progress and highlight dependencies, but not everybody stuck to it, partly because it was based on the original project plan and the tasks were too high-level to be really meaningful on a day-by-day basis. It may have been better to have asked each project member to break down the tasks they were responsible for into smaller chunks, providing greater detail each month for the tasks that they were about to move on to.

3. *Google code is good for open source software development, but needs a defined workflow for dealing with bugs on top of the basic tools*

On the whole, the decision to use Google Code as the open-source code repository for the DaaS software was regarded as a good one, although one of the developers felt that the built-in bug-reporting mechanisms were a little too simplistic. We tried to overcome this by assigning one member of the team to review all issues as they were added and re-prioritize them as necessary, but in future we might look to refine this process further.

4. *Working closely with VMware was beneficial to the project and such partnerships would be worth exploring again in the future*

Working closely with VMware helped our virtual infrastructure staff acquire an excellent working knowledge of the software and allowed us to get the cloud aspects of the project up and tested more swiftly than would have been the case otherwise. Whilst their involvement came at the expense of helping them with their promotional activities, this actually handed us the opportunity to promote our own work on VIDaaS at the same time.

5. *Scope the software development platform you will be using thoroughly before committing and ensure it is supportable once in service*

We were rather unfortunate regarding the obsolescence of JBoss 5 (the application server we began developing the DaaS with) during the project, but even disregarding the unforeseeable security problems when we adopted the platform, it was not a unanimously popular choice amongst the developers that we brought in to the team. We had originally chosen to work with JBoss as it offered a relatively rapid way for us to get a prototype DaaS system up and running that researchers could evaluate, and because our lead developer was already familiar with the development environment. This approach became problematic, however, once we had to start considering the long-term support of the software as a service. We are currently assessing whether we should upgrade the software to the latest version of JBoss, which might require staff re-skilling, or whether the long-term sustainability would be better served by migrating at least the back-end of the software to a common platform with the other services that our SysDev team already support.

6. *Small user observation workshops may prove better for software testing than larger but less focussed groups*

We had wanted to use our testing processes as a means to familiarize a wide range of early adopters with the software and involve them closely with its development in a longitudinal capacity. However, when we asked the testing group for feedback (by email) the response rate was very low, mostly (we suspect) due to the other higher-priority claims on our researchers' time. Realizing that we were not getting the in-depth involvement we had hoped for, we adjusted our strategy and set up a number of small user-observation workshops instead, drawing both from people who had already signed up to the testing group as well as new volunteers who responded to emails circulated around academic faculties. This turned out to be very effective, generating a large amount of useful feedback from a relatively short testing session, both confirming some things we already suspected, and flagging up new issues.

7. *Begin user testing early – even if it's just with paper prototypes*

The earlier user testing begins, the better, as the earlier problems are identified, the easier it is to fix them (or better still, avoid them entirely). Beware that technical prototypes take time to develop, and that once they have been developed it becomes psychologically difficult to tear up significant parts of them and start again. Were we to re-run the project we would probably talk our testers through processes such as the registration process before actually coding it, as re-doing the forms we developed for the prototype proved more time-consuming and complicated than had been anticipated.

8. *Involve your interface designers in the initial stages of the software development process*

VIDaaS hired in some design consultancy to help improve the appearance of the Web interfaces, but the amount they could achieve within the time available was limited, primarily due to the fact that they needed to familiarize themselves with the development platform and that the design and content had not always been kept properly separated before that point. The use of Ajax-based interfaces also threw up some accessibility issues which we had not foreseen. Seeking advice earlier in the project would probably have avoided some of these issues.

9. *Keep user interface text separate from the main software code*

Text seen in the user interface should be kept in a single (or small number of) messages file(s) to allow the text there to be referenced from all the different parts of the application. This approach has three benefits: it allows for multiple language support (easy internationalization is important to get new contributors for the open source project); it centralizes all end-user text for easy maintenance (so that we know which text is shown where and don't define a label multiple times); and it allows a UI designer to easily work on and improve the application.

10. *Think about how you will measure benefits at the beginning of a project rather than the end*

We effectively ended up creating a second survey of research practices near to the end of VIDaaS because we hadn't captured very much data that would enable us to cost the benefits of the ORDS during the earlier requirements analysis stage. In practice it may not have been viable to capture all of the information we eventually used in just one set of questions anyway – one can only take up so much of a researcher's time with interviews and surveying in any one go, but thinking through what we needed to measure earlier in the project could have helped inform discussions regarding user requirements when speaking to potential early adopters.

11. *Envisage how the finished service will function and communicate this*

The project was helped by the fact that we considered the software development in terms of a user-facing service from an early stage. Creating the SLD and service governance documents early in the project helped focus attention on what it was we were supposed to be delivering, and combining this with an understanding the research context made it faster to establish the general type of pricing model that would be required by the service and therefore how we needed to go about the costing work package and the billing process.

12. *Plan exactly how each stakeholder group should be informed of developments*

We feel that our communications strategy worked relatively well. A significant effort was made at the start of the project to identify precisely which stakeholder groups should be informed of which aspects of project progress and the best way to communicate with that group. Our early focus was on reaching researchers (potential early adopters) via presentations at academic workshops (rather than events specifically organized by the project). The quarterly email bulletin sent to a long-list of named stakeholders also worked well to reach a wider audience than JISC project sometimes do and ensure that they were reminded of the project at regular intervals.

3.4 Immediate Impact

The VIDaaS Project has raised awareness of many of the issues surrounding institutional research data management amongst a much wider audience than was possible in earlier projects. Whereas Sudamih had been a small pilot project limited to the humanities, VIDaaS has touched a much wider range of people within the University of Oxford and beyond. Through the interviews, surveys, testing and workshops a wider range of researchers have learned about the service we are developing and been required to think about how they manage research databases at present and how their practices could be improved. The involvement of a number of teams within the Computing Services (along with the University's data management programme being a major focus of the Computing Services staff day) has meant that many more people involved in providing support and services to researchers are also aware of what we are implementing. The fact that more and more researchers are actively thinking about their data management requirements is evidenced by the increase in inquiries about database support that the department is receiving.

The major impact of the project within the University will be felt after the summer, when the ORDS will be launched as a new public-facing University service. We have identified a number of projects who are likely to be interested in using the service, and the divisional research services will encourage new projects to use the ORDS where appropriate. The service should transform the way that research data is managed and preserved within the University, and will form a core part of the larger research data infrastructure that we are building.

We have received informal expressions of interest in adopting the DaaS software from the universities of Essex and St. Andrews, and we anticipate that this list will grow once the software has been migrated and properly integrated. Our workshop and public demonstration of the software attracted delegates from 22 different universities, so awareness of the project in UK HE should be high.

3.5 Future Impact

Our user requirements analysis suggests that demand for the ORDS amongst researchers will be high, and conversations with research services and other staff indicate a commitment to nudge researchers into using the service where appropriate to enable the University to keep better track of its research data. Our initial best-guess uptake figures assume that once the service is fully mature (after additional functionality has been implemented and confidence in the security of the system assured) it will attract between 80 and 100 new projects each year at the University of Oxford. If scaled up to provide a national service this figure would be significantly greater. We will monitor uptake and use closely and adjust the pricing model if required so as to ensure that researchers who would benefit from the system are able to do so.

Whilst any university or other organization is free to download the DaaS software and use it in whatever way they see fit, we hope that they will get involved in the ongoing development of the open source software (even if just to report bugs and feature requests) and work with the University of Oxford to help maximize the value of the DaaS. We can in that way track who is using the software and for what purpose.

4 Conclusions, recommendations, and implications for the future

At present researchers use a wide range of tools and processes for managing their research databases. In many instances these are sub-optimal, either with respect to what the researchers wish to be able to do with their data, or with respect to its longer-term management and curation, or both. Even where research databases are well managed there is generally significant scope to reduce the associated costs. The DaaS offers significant advantages over many existing practices, especially with regards to the collaborative editing and sharing of databases. These benefits are discussed in more detail in Appendix A.

Many researchers see the advantages of using the DaaS over their existing solutions. The system does need to be easy to use and well documented, however, and priced in such a way that researchers can access it.

From an institutional perspective, there is a clear need for better, centralized systems that can ensure that the research data produced in the University is known and managed in an infrastructure that meets the requirements of the major funding agencies. The ORDS helps meet these requirements during the data collection and analysis stages of the research data lifecycle.

4.1 General recommendations

During the VIDaaS project we have learnt a number of lessons which may be of benefit to future projects developing software tools for research data management (listed in section 3.3). One recommendation worth re-iterating here is simply not to underestimate the amount of time it can take to bring a pilot project to a full, properly documented and supported service.

4.2 Recommendations for the wider community

As the Research Councils refine their data management policies and begin to devote more energy to monitoring institutions and researchers to check that they are complying with those policies, all UK research universities will need to ensure that they have an appropriate infrastructure to meet funders' requirements, else risk losing research income.

It is becoming increasingly clear that whilst many of the Research Councils are prepared to offer funding towards the management of research data during the lifetime of funded research projects, universities will be expected to support the infrastructure for long-term preservation and re-use of research data. By including tools such as the DaaS in their infrastructures, universities will be able to, 1) make a strong case for data management funding to pay for the cost of using the tool during the lifetime of a project, and 2) streamline the full lifecycle management of research data and reduce the associated costs.

We would recommend that other HE institutions involved in developing research data management infrastructure review the DaaS and consider how it can benefit them. Work on the DaaS will continue at Oxford, where there is a clear (and funded) commitment to provide a service based upon the software. In order to help secure the long-term future of the software it would be beneficial if any institution thinking of adopting the DaaS could work with Oxford, whether formally or informally, on improving the open source software – even if this is merely to report bugs or suggest future improvement. By building an active development community around the DaaS the community can maximize the benefits derived from the software and ensure it evolves to meet the changing research landscape.

4.3 Recommendations for JISC

Although the terms of the UMF Shared Services and the Cloud funding call were well understood and the funding vital in terms of bringing the DaaS to service standard, it is worth noting that the short timescales dictated by the funding did place risks and limitations on the project which would have been mitigated had the same funding been available over a longer period:

- Although only one position in the VIDaaS project team was not filled by the planned recruitment date (the Shared Services Developer), projects with large staffing requirements that must be completed within 12 months are always at a serious risk of failing to deliver their expected outcomes due to complications in recruitment. VIDaaS succeeded as key team members were able to transfer from the earlier Sudamih Project.
- Short-term funding awards encourage projects to adopt available pre-fabricated open-source software solutions and then try to adapt them, rather than developing better-suited tools from scratch. This may slow later development work. Some of the DaaS components adopted during the

Sudamih project, whilst essential to the success of that project within its timescale, have proved more time-consuming to adapt during VIDaaS than we had hoped.
- Longer-term funding would have enabled us to respond better to user feedback. As it was the project needed to move rapidly on to new aspects of functionality as each was released, otherwise risk falling behind schedule. Being able to develop documentation and training materials consecutively to the software development, rather than simultaneously, would have allowed the project to produce more robust support materials.
- VIDaaS delivered against the remit set by the UMF Implementation Group at Brettemham House on 1 November 2011. However, the original UMF vision was to share services across the community, and for the four research-focused projects funded by the UMF to share best practice. The UMF Implementation Group did not meet again, and VIDaaS did not get the opportunity to share the very significant work it has completed on defining the process of taking a prototype service to a full production service, on service costing and service subsidy, and on calculating Return-on-Investment.

In addition it is worth noting that short notice of funding opportunities can impact on the quality of JISC project work:

- Staff working on JISC projects often build up very useful and beneficial skills sets and domain knowledge, but can be difficult to retain when future funding is still in doubt as a project nears its conclusion. Calls for funding, particularly programme continuation funding, should be timed so that future funding can be announced well before the conclusion of major projects and universities can plan to retain staff on short-term contracts.

JISC/HEFCE should now turn its attention to investigating how the ORDS and other Software-as-a-Service tools can best be scaled up for provision at a national level. Our costing work has concluded that there are large economies of scale to be gained by offering the service nationally, rather than on the institution by institution approach we have adopted thus far, but there are also greater risks involved. We hope to investigate the process of turning a local service into a national service with some follow-on funding to VIDaaS, but additional support from JISC in terms of case studies and guidelines would be very welcome, not just from our own perspective, but also, I am sure, for other software/services that have found themselves in similar circumstances in the past, or will do so in the future. Cloud-based services should be uniquely well suited for this sort of transition, but there appears to be little best-practice information available at present.

Possible models for national provision of the ORDS service are outlined in Appendix A - VIDaaS Sustainability and Business Case.

5 Appendix A - VIDaaS Sustainability and Business Case

N.B. The VIDaaS Sustainability and Business case is available as a separate document, but included as an appendix to the final report in order to support the conclusions drawn in that document.

5.1 Output details

Name *[3]
Virtual Infrastructure with Database as a Service (VIDaaS)
Description
The principal deployable output of the VIDaaS Project is the 'Database as a Service' (DaaS) software. The DaaS is intended for researchers across all academic disciplines who require a relational or XML database to store and analyse their research data. Essentially the DaaS provides databases on demand, enabling researchers to: • Create a new relational database or XML database from scratch • Import an existing database in common formats (including Microsoft Access) • Design or re-structure databases via a drag-and-drop interface • View, add, edit, and filter data • Construct queries to perform sophisticated data searches • Set different access permissions levels • Share data with selected colleagues or open it to public searching & viewing • 'Publish' subsets of data which can be linked to other research outputs • Straightforwardly export data for analysis and visualization in commonly-used software tools. • Access DaaS-hosted data via custom-built websites • Access online user documentation for help and support As the DaaS system is build to be institutionally supported and hosted on cloud infrastructure, other benefits include: • Centralized back-up of data and disaster recovery • Centralized maintenance and updating • Automatic metadata capture • Smooth integration with long-term data archives The DaaS software is intended to be provided to researchers as part of an institutional service, the Online Research Database Service (ORDS). The University of Oxford intends to officially launch the ORDS during the Summer of 2012.
Aspiration
The DaaS is intended to improve current data management practices amongst

researchers whilst saving the UK HE sector money (compared with current practices). The DaaS is one component that can be used in an integrated infrastructure to ensure the long-term preservation of research data and facilitate its re-use. By making research data more open and publicly accessible it may also help improve research integrity.

Benefits

Description	**Indicator**	**Measurement**
Quicker initiation of database from request to use	Reduction in latency	See section 5.1.3.1 below
Lower cost to deploy databases	Reductions in time / staff / hardware costs	See section 5.1.3.2 below
Reduced hosting costs	Reduced cost of data/Web hosting	See section 5.1.3.3 below
Economies of scale from centralized software updates and support	Reduction in time during which system is unusable / improvements in user support	See section 5.1.3.4 below
Improved data back-up, disaster recovery, and data security	Reduction in costs of data reproduction	See section 5.1.3.5 below
Improved ease of data sharing/publication	Increased public sharing of research data	See section 5.1.3.6 below
Automation of metadata capture	Reduced data curation costs	See section 5.1.3.7 below
Builds up data expertise amongst researchers and improves research management awareness	Improved data management expertise amongst researchers	Difficult to meaningfully measure. May get indications via survey post-project
Metadata can be used in a local registry of research data	Improved data rediscovery & reuse	Difficult to meaningfully measure. May be attempted during DaMaRO Project
Data can be linked to research publications, increasing impact	Improved impact	Difficult to meaningfully measure. May be attempted post-project
Greater efficiency (time saved) finding data	Reduced search time / improved search results	Requires analysis once ORDS is in service
Increased opportunity to verify data leads to improved data integrity, improved research integrity	Improvement in research standards	Almost impossible to measure. May get qualitative opinions post-project
Improved institutional	Better ability to capitalize on valuable data & manage	Difficult to meaningfully measure. May be

awareness of data assets	reputational risk	attempted post-project
Standardized format facilitates data mash-ups	Increased re-use of data & ideas for new research	Difficult to meaningfully measure. May get indications post-project
Better awareness of other datasets should reduce risk of data duplication	Reduced duplication of data	Difficult to meaningfully measure. May get indications via survey post-project

5.1.1 Cost structure

The VIDaaS Project has conducted an in-depth study of the costs involved in providing the ORDS Service and the uptake required before the service can recover its costs.

There are two major cost components to the ORDS service: the staffing of the service; and the infrastructure that it will be provided on. The service can either be delivered via the Oxford private cloud (good for sensitive data or researchers of a nervous disposition), or on the public cloud, which is likely to be cheaper (see section 5.1.3.3). Both the service staffing and the private cloud option are associated with significant fixed costs. Using the public cloud for hosting results in a more variable infrastructure cost component, but is not a viable option for all research at present. A proportion of the infrastructure cost component needs to go towards the capital renewal costs of the Oxford cloud infrastructure.

The costs of providing the ORDS service are as follows:

	Fixed costs	**Variable Costs**
ORDS service	£142,717	c. £10k per 100 projects above first 100
Oxford Private Cloud Costs:-		
Capital costs	£32,250	c.£8k extra per 100 VMs above initial 400 capacity
Service costs	£73,900	
+ *software licences*		c. £40-£80 per VM
TOTAL	**£248,867**	

Above 100 projects the ORDS service costs increase due to the need to provide more helpdesk support, etc. Our initial hardware capacity can cater for up to 400 ‘standard’ VMs, after which investments in additional hardware is required.

The anticipated staffing requirements for the ORDS at launch are summarized below:

Role	FTE	Grade
Systems admin & monitoring	0.2	8
Technical development	0.4	8
OSS Governance	0.1	8
Platform upgrades etc.	0.2	8
Adding and supporting new IaaS	6-7 weeks pa	8
Updating training & documentation	0.1	7
Training researchers, staff, & support	0.1	7
Helpdesk support 2nd/3rd line	0.1 per 100 projects	8
Helpdesk / operations issues relay (1st year only)	0.2	8
TOTAL	**1.4 + FTE**	

5.1.2 Pricing & billing model

We have set our initial service prices for Oxford cloud hosting at £600 per year for a ‘regular’ virtual machine (VM), and £2,000 for a ‘large’ VM providing more CPU power, RAM, and storage space. We anticipate that the regular model will be fine for around 80% of users. Prices have been set at this level for now to try and recoup costs against a timetable that the University might realistically accept without pricing the service out of the market (see section 5.1.4). We will need to review our prices regularly to react to changes in demand.

Prices for public-cloud hosted instances of the ORDS will need to be calculated at the time of launch, as they are liable to change rapidly.

Model	CPU	RAM	Storage	Price per annum	Anticipated demand
Regular Oxford-hosted	½	1GB	25GB	£600	80%
Large Oxford-hosted	2	4GB	100GB	£2,000	20%
Regular Nationally-hosted	½	1GB	25GB	n/a	-
Large Nationally-hosted	2	4GB	100GB	n/a	-

Custom	?	?	?	?	-

Obviously we cannot confirm our best-guess uptake at these prices until we have a cohort of early adopters who understand the ORDS well enough to assess it against alternative solutions which they could use. Furthermore, once we better understand the financial benefits derived from using the service we may be able to offset these against the price charged to the user. Potentially the value of the benefits to the University from its researchers using ORDS as opposed to alternative solutions may be such that it becomes worthwhile for the University to offer the service at a reduced price, or even free. Until we can assess the experiences of early adopters, however, the financial risks to the University of offering the service at a price below that which reflects its running costs and estimated uptake involves too great an element of risk.

The pricing models are intended to be as simple as is realistically possible for researchers to cost into research funding bids. Cloud services are often charged on a pay-as-you-go basis, so that customers are only charged for what they actually use. This is appealing for most users as they do not end up paying for more space or compute power than they actually need. However, this model is not helpful for researchers, who are unlikely to have a clear idea how much use they will make of the service at the point at which they need to submit their funding bids. Providing relatively straightforward options with clear prices attached is likely to prove more convenient to the research community.

The current pricing model, whilst fine for researchers funded by Research Councils, is less well adapted to 'unfunded' research, where a researcher simply conducts research in their own time and is reliant on their salary or the largesse of their department to provide the required hardware and software. The ORDS prices are likely to exclude such researchers initially, so we will need to work further with the departments to find a way in which they can provide the ORDS service to those researchers who would benefit from it, whilst still contributing to its costs. A departmental subscriptions model is one potential solution that we will be considering.

The ORDS allows users to pay for the service either up-front or on an annual basis at the price agreed at registration. Projects will be encouraged to pay up-front, partly in order to ensure that cash-flow does not become a problem, and partly to minimize any issues that might result from the service needing to make significant changes to its pricing models in subsequent years.

5.1.3 Benefits measurement

Whilst there are many identified benefits deriving from the use of the ORDS service, costing these benefits is difficult in most cases. This is because the solutions that researchers currently employ when creating databases are generally not directly comparable to what ORDS offers, and also because many of the benefits of the ORDS are inherently difficult to measure in financial terms. Many of the existing costs of database creation (and curation, insomuch as this happens at all) are hidden, as we have established through a survey of more than thirty researchers who have created research databases in recent years. Establishing a solid baseline for comparison is therefore fraught with danger, and likely to be misleading rather than enlightening. In the analysis below I have used such data as we have been able to gather to support the benefits listed above. I have attempted to quantify benefits where possible, although such quantifications are estimates at best.

The best way to measure the benefits of the ORDS would arguably be to calculate the value of the data that the service will preserve and enable access to which would otherwise be inaccessible, incomprehensible, or unreusable. Unfortunately, this is not possible, as the value of any given data set to future researchers is almost impossible to measure or predict, and as yet we cannot even estimate what or how much data the ORDS will preserve that would not be accessible via other means.

5.1.3.1 Quicker initiation of database from request to use

The following steps must be addressed before one can really begin to add data to a research database:

1. Reaching a basic level of understanding regarding how databases work and how one could be appropriately used to address the research questions at issue
2. Deciding whether building a database is indeed the most effective way to address the research questions at issue
3. Establishing which database tools or software would be most appropriate, which may involve consulting with colleagues or technical experts
4. Acquiring the most appropriate software & installing it
5. Learning how to get the software to perform the tasks you wish to undertake
6. Designing and structuring the database

The precise length of time a researcher will take to go through these steps varies tremendously according to the importance and complexity of the required database.

Our survey of 34 researchers found that 29% took less than a week from deciding to use a database for their research to having their database structured, set up and ready to receive data. For 23% of respondents, this process took more than a year. Projects that spent a significant amount of time on this were more likely to set out with the intention of making their data publicly available via a website, but 4 out of the 10 responses from projects that had taken under a week to go through this process also either had already made their data available via a website or intended to do so. Of course, there is more than one way to make one's data available via a website, so generalizations regarding any 'standard' process are extremely hard to make.

Using the ORDS could save researchers time in some aspects of the initiation process, but not all. Most obviously, using an online on-demand service cuts step 4, although this is usually the step that takes least time and effort in any case. Given that the ORDS is quite a simple system in comparison to packages such as Microsoft Access, it may also reduce the time spent on step 5 of the above process, although it is unlikely to save any time on steps 1-3. The simple database structuring interface that the ORDS offers may also save time on step 6, but we have not had the opportunity yet to test this with real users under anything like controlled conditions. We may be able to get a better picture of the savings in this regard once the service is in use.

In situations where researchers want their data to be available online, the ORDS can save significant time provided that the generic interfaces are deemed good enough for their purpose (i.e. the project does not require their own branded custom-designed Web interface) (see 5.1.3.6).

5.1.3.2 Lower cost to deploy databases

As already mentioned, most research databases are not directly comparable with the ORDS. This is partly because researchers have very diverse needs, some easier to cater for than others, but also partly because many researchers do not really understand what databases are capable of when they start a project and deploy them in a sub-optimal manner. Our survey of researchers who had experience of setting up research databases asked what they had found to be the most frustrating aspect of the process. Responses included: 'uncertainty about the best approach', 'the time it took', 'lack of experience using database software', 'learning to use the software', and 'lack of capacity for collaborative working'.

Sharing a database between multiple editors seems to be a particular problem for many projects. We have come across the following 'solutions':

1) Email a database file to one another. One researcher can then edit it and email the next version to the others. This can get very confusing and does not allow multiple researchers to edit the database at the same time.
2) Put a database on a departmental server which researchers can then download via ftp, edit, and upload. Again, this is a poor way to go about things if multiple researchers want to edit the database at the same time.
3) Have one researcher in charge of a main database. Get the other researchers to work on their own bits of the database, then send them to the coordinator who can copy and paste them into the central database. This is another poor solution.
4) Custom-build a Web interface enabling selected editors to edit a database hosted on a departmental server remotely.
5) Work with text files or spreadsheets instead and use, for example, Googledocs with Google Fusion Tables.
6) Use SharePoint. SharePoint "lists" are actually little databases that can be used in some fairly complex ways [n.b. we haven't actually found any researchers using SharePoint in this way].
7) PostgreSQL or MySQL on shared server and desktop clients (e.g. Pgadmin3 or LibreOffice Base).

Obviously the first few options on the list are generally only 'chosen' by researchers who have begun their research using a particular piece of software (often one that came as standard with their machines) and whose needs have gradually grown without them having had the time or wherewithal to step back and reconsider which tools and methods would better suit their needs.

The costs involved in employing different database management options and combinations of software vary tremendously, and combinations that work effectively for one project may be unsuitable, or offer poor value for money for another.

Furthermore, the costs of various database management solutions can be invisible to researchers. No researcher (who doesn't also have significant departmental IT or accounting responsibilities) is likely to know how much their database solution is costing in terms of power, for example. The costs of software provided as standard with hardware are difficult to factor in, and hardware may be used for many purposes other than database management, so is also likely to be omitted by researchers considering their costs.

Our survey of researchers who have in the past created research databases found that half had spent nothing on hardware, 41% had spent nothing on software, and 5 out of the 7 projects who were already making their data available via a website said they had spent nothing on hosting. This suggests that these costs were not visible to the researchers, not that they didn't exist and were being paid by somebody, somewhere. The costs of providing the ORDS service, on the other hand, are well understood and factored in to the price (see 5.1.1).

The upshot of this is that we cannot say how much the ORDS service would financially save a researcher in comparison to any standard method or straightforward baseline – there is no standard method, and researchers don't know the costs of their databases.

We do, however, know that many researchers are working inefficiently (if sometimes quite imaginatively), and that the ORDS would in many cases greatly improve their efficiency.

Miko Flohr, a researcher in the Classics and Ancient History Department and the database coordinator for the Oxford Roman Economy Project (OXREP) estimated in 2011 that had his team been using the ORDS service during 2010 (it was of course not ready at that point), the efficiency savings to his project deriving from it use would have amounted to about 21% of his time (or £23,700 in FEC terms). At that time the project was essentially working according to the model in option 3 above.

To get a better sense of the costs and potential savings for projects working to a more sophisticated model, we asked our Infodev team at Oxford University Computing Services to estimate how much it would cost at present to develop and host an online research database able to cater for multiple editors and public searching. Infodev frequently create custom-built databases for members of the University and have a great deal of expertise in requirements gathering and building bespoke solutions. They estimated that a no-frills database of the type required might take about 6 days to produce. Within this time, the division of effort would typically be divided between different steps. The Infodev team charge £450 per day (FEC rates) for their time, so the actual cost of each stage can be included.

Database development service action	Effort	Cost (FEC)
Consulting with the customer	20%	c. £540
Structuring the database	20%	c. £540
Developing data editing tools	30%	c. £810
Developing user front-end website	30%	c.£810
TOTAL	100%	c. 2,700

In many cases researchers, after seeing what can be done, request additional work and customization.

On top of the Infodev development charges, the researcher would also need to pay for hosting for their website and data. At present, this would be normally be offered by the Networked Systems Management Service (NSMS) team at the Computing Services, who offer a monitored Linux server with the equivalent spec to the ORDS at the following prices:

NSMS Online database hosting option	Effort	Cost (FEC) pa
Monitored Linux system, 1GB RAM, ½ CPU, 30GB storage (closest spec to a 'regular' ORDS)		£4,950
Monitored Linux system, 4GB RAM, 2 CPUs, 100GB storage (closes spec to a 'large' ORDS)		£8,445

The total cost to the researcher over five years for this kind of provision would therefore come to around £27,450 or £44,925 respectively. As with the ORDS, but unlike most departmental solutions, this figure covers all aspects of provision and doesn't hide elements such as power, depreciation, support, indirect staff costs, and so forth. A 'regular' ORDS instance would cost £3,000 over 5 years, whereas a 'large' ORDS instance would cost £10,000. These prices are, of course, not directly comparable, as researchers using the ORDS would normally expect to have to structure (and restructure) their own databases, work within the generic editing constraints of the system, and make do with a basic outward-facing front-end. Combining elements of the Infodev service with the ORDS could, however, offer substantial savings, mostly on hosting.

Imagine a researcher opting to use ORDS whilst benefitting from Infodev's consultancy. Infodev would still be responsible for consulting with the customer to precisely understand their needs, structuring their database, and providing an attractive and useful Web front-end and search facility, but built on top of the ORDS back-end and cloud hosting rather than the more traditional server hosting. Using the costing above, this would amount to £4,890 for a 'regular' 5-year project, or £11,890 for a 'large' 5-year project – savings of £22,560 (82%) or £33,035 (74%) respectively over the current standard solution.

Obviously, using Infodev's consultancy service and technical expertise may additionally reduce the time that inexpert researchers themselves spend structuring and restructuring their databases, thus saving money on this aspect of work as well, although that is not a benefit of the ORDS service *per se*.

Whilst we anticipate that our Infodev team will want to utilize the ORDS service as a back-end component in future projects, the service is also intended to be used directly by researchers who wish to control each step of the process. It is difficult to estimate the savings that will result from the creation-to-archiving use of the service directly by researchers, but we intend to try to assess this in follow-up work once we have early adopters actually working with ORDS.

5.1.3.3 Reduced costs due to cloud hosting

One of the key drivers of designing the DaaS for cloud hosting was the potential to minimize hosting costs.

The table below does not compare like with like exactly, but attempts to find an indicative cost of providing hosting for a moderately-sized research database using various hosting solutions.

Hosting solution	Cost	Notes
Departmental hosting (estimate from Classics department, University of Oxford, February 2011)	£10,494 per annum	Majority of cost is staff time for maintenance and management, server hardware costs of approx. £200 per annum
Oxford University Central hosting	£4,950 per annum	For a monitored, but not fully-managed Linux server (with 30GB space, 1GB RAM, ½ CPU) (as example in section 5.1.3.2)
Oxford University, VM for rent, centrally hosted	£768 per annum	Unmanaged, with Linux OS (30GB space, 1GB RAM, ½ CPU)
Public cloud hosting (Amazon EC2)	£386 per annum	For a small, always-on Light Utilization Reserved Instance (160 GB space, 1.7 GB RAM, 1 CPU)

Public cloud hosting (Rackspace Cloud Servers)	£352 per annum	Unmanaged Linux, 40GB space, 1 GB RAM, unspecified CPU, assuming 1GB per month bandwidth out
Private cloud hosting on Oxford Cloud	£313 per annum	Hosting cost of 'regular' instance of ORDS DaaS (25 GB space, 1GB RAM, ½ CPU) cost based on Shared Data Centre occupation at 50%
Eduserv vCloud Compute Pay-as-you-go	£230 per annum	Linux, 25 GB space, 1 GB RAM, 1 CPU.

The price models that commercial public cloud providers use are often very difficult to compare, and prices can change frequently (mostly downwards). The different terms and conditions offered means that some public cloud providers are unlikely to be able to meet the Service Level Description that the ORDS offers. We will address this in greater detail in follow-up work. The assumption is that we will generally be able to utilize public cloud provision at a lower cost than the Oxford private cloud (although besides the Eduserv model this is not apparent from the examples above).

5.1.3.4 Centralized software updates & support

Software developers tend to release patches and updates to their software on a fairly regular basis, as bugs and vulnerabilities are identified. Such updates tend to cause little inconvenience in an age of almost ubiquitous Internet connectivity (indeed they are often required as an indirect result of that ubiquity and the security issues it presents). Less frequently, completely new versions of software products are released to improve general functionality and restore market competitiveness, often (in the case of commercial software) requiring the purchase of a new software licence. Customers need not pay to upgrade to the most recent release of a software package, but if they choose not to, helpdesk support costs tend to increase (or quality suffers) as support staff need to be able to assist with enquiries relating to more and more extant versions.

As future upgrades to the DaaS software may all be applied centrally, behind the scenes, and at times when few people are using the service, they will not cause the users of the software any significance inconvenience, or cost them any money.

We estimate that the ORDS could eventually be used by about 450 projects within Oxford at any given time (these are very much 'finger-in-the-air' estimates, but we cannot produce anything more concrete at this time). If each of these projects involves an average of 3.35 researchers (at grade 7) entering data (the average from the 34 projects we surveyed) and each researcher loses just 5 minutes of productive time each month due to software updates or upgrades, then switching to ORDS would save the University in the region of £27,000 over the course of a year.

Our survey of researchers who have constructed research databases in recent years found that the 34 respondents used 14 different pieces of software to manage their databases. This diversity has costs attached to it. When researchers run into problems they must either spend their time finding out how to solve the problem themselves, or contact support staff for advice. But there is a limit to how many database management systems any member of support staff can expect to know intimately. A University must either spend more on training its support staff, employ additional staff with different expertise, or suffer inefficiencies as researchers flounder around trying to interpret past cryptic solutions that technical experts have left scattered around the Internet (software manuals may suffice for simple problems).

The ORDS service is never likely to be able to replace all of the various different tools that people use to manage research databases; there are simply too many special cases for any generic service to be able to do this, but if the University encourages researchers to use a single central service, it may over time reduce the burden on technical support staff and increase the efficiency with which they can respond to enquiries.

5.1.3.5 Improved data back-up, disaster recovery, and information security

Significant data loss (which need not necessarily be large-scale data loss) is understandably a concern for universities if they are to be able to fulfil research funders' data management requirements or respond to FOI requests. Researchers themselves tend to get upset if they lose months of painstaking research. For this reason, improvements to data back-up and disaster recovery processes tend to be regarded as good investments. It may therefore be concerning that only 44% of the researchers we surveyed backed their databases up to central university infrastructure. An additional 26% backed their databases up on departmental servers, but that still left 27% backing-up solely to local storage media (rather alarmingly, one of the surveyed researchers claimed not to back up his/her database at all).

The good news from our survey was that only 15% of the respondents had actually lost any data due to software or hardware failures, and it didn't take any of them more than 3 days to replace the missing data. No data had been lost due to malicious attacks. Our small-scale survey may not be the most representative from which to generalize, but if we assume the five researchers who had lost data were each employed at the top of grade 8, their data recovery costs in terms of wasted working hours would amount to £2,671. Given the number of years they had spent working on their databases, and the number of years all of our luckier respondents had worked on theirs without losing any data, it would seem that an average of just £20 per year is wasted in any given project recreating data. This does seem remarkably (unrealistically?) low. Of course, a single catastrophic incident of data loss would hugely alter this number, as the University of Southampton can doubtless attest after fire destroyed their Mountbatton building (which hosted their School of Electronics and Computer Science) in 2005.

Given that the ORDS backs its data up to tape every night, which is then stored in multiple locations, it is extremely unlikely that anyone using the system would ever lose more than 1 day's work. In practice, the risk of losing data stored in the Oxford private cloud would be minimal.

Given that we anticipate the ORDS service to support approximately 450 active projects per year in the University once it reaches maturity, it might be expected to save around £9,000 per year (450*£20) in avoided data recreation costs. It could, of course, end up saving much more – although this is unlikely ever to be known.

5.1.3.6 Easier to share/publish data

Referring to the example in section 5.1.3.2, our Infodev team estimated the cost of developing a Web front-end for a research database to be typically in the region of £810 for something fairly generic, with hosting costs on top. Whilst the generic data filtering and searching interfaces of the DaaS are no replacement for the design and branding that Infodev can create, for researchers that lack the funding to produce their own website, the DaaS essentially offers them the ability to expose their data to the public at no extra cost at the flick of a switch.

Of our surveyed researchers, 21% said that their data was already available, at least in part, from a website, and 35% said that they had no intention of making their data available in this manner. Of the rest, 32% reported that they were intending to make their data available via a website in the future, and the remaining 12% indicated that they might consider it. It is likely to be projects in these latter groups (none of which had specifically requested money for Web hosting in their funding bids) where this aspect of ORDS functionality might in practice influence behaviour.

If the 12% of researchers who would consider making their data public, but hadn't really planned to do so from the outset of their projects were to do so via the ORDS, this would result in a lot more data entering the public domain, and in a relatively standardized format which may in the future assist re-use or even data mash-ups.

The projects that indicated that they might consider putting their data on a website were all small one-researcher projects. Two of the projects indicated the investment that had gone into compiling their data: one costing around £35,000, the other around £2,000. Whilst it is almost impossible to measure the reuse value that these databases might have if made publicly available, the costs of their recreation are significant enough to suggest that the ease with which the ORDS facilitates public data sharing does provide a substantial financial benefit over existing solutions.

5.1.3.7 Better metadata capture for long-term archival

Research undertaken by Charles Beagrie Ltd. for the Keeping Research Data Safe 2 report concluded that 'the largest potential cost efficiencies [regarding the digital preservation of research data] will come from future tool development supporting automation of ingest and access activities for curation and preservation'.[4] With this in mind, the DaaS software has been designed to capture essential metadata describing the databases held in the system, which can then be passed to longer-term data repositories and archives.

The KRDS2 report assessed the costs involved in the various aspects of research data management. It defines the processes included under the 'ingest' phase as: receive submission; quality assurance; generate information package for archive; generate administrative metadata; generate/upgrade descriptive metadata and documentation; co-ordinate updates; and reference linking. The report notes that ingest costs are often one of the most expensive parts of the data curation process: 'One of the most salient features of ULCC's cost profile is the predominance of ingest staff costs as a fraction of overall annual costs. Ingest staff costs account for three-quarters of overall costs, or £3,936,82 per GB'.[5] The UK Data Archive, in their 2005-06 assessment of curation activities found that the most time-consuming aspects of the ingest process (or 'reception and processing' in their terminology) were the conversion of data and metadata to the dissemination format, and the production of the catalogue record, typically taking between 10 and 16 hours altogether.[6]

Upon ORDS registration, the project owner is required to give their details and provide some basic information about the project. Information is also captured about anyone else the project owner wishes to provide access to. When a database is created, the system requests a short description of the database and also prompts the project owner to describe the sources of data and the data gathering process –

---

[4] KRDS, p.83

[5] Ibid, p.45.

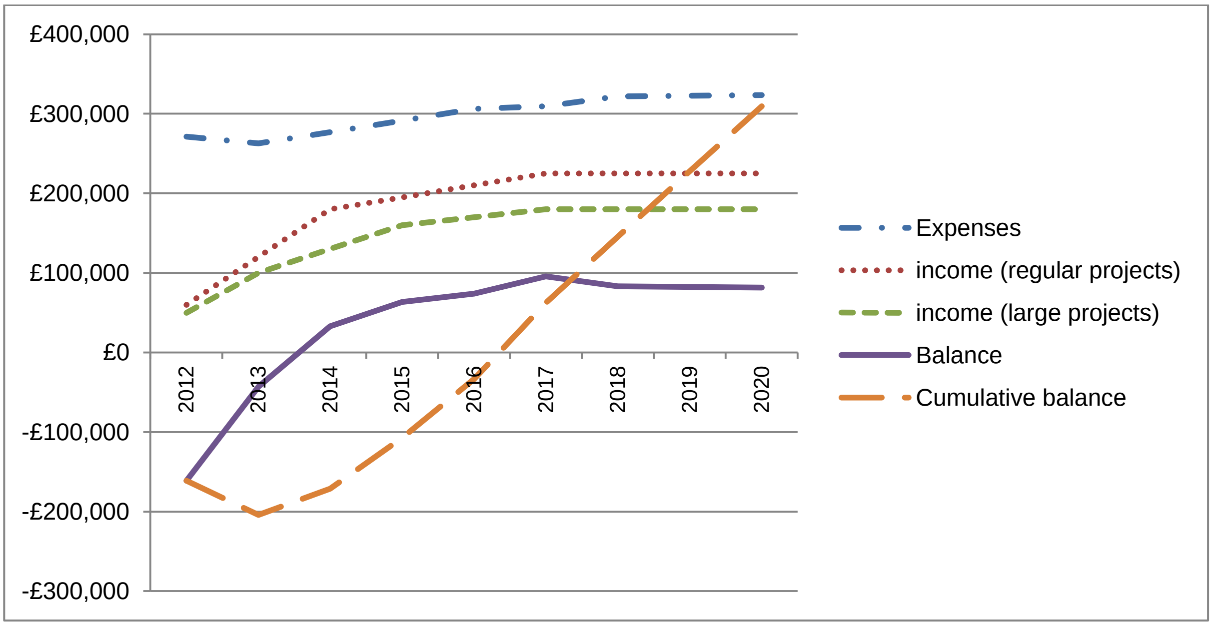
[6] Ibid, p.42.

useful information for anyone wishing to understand and re-use the data further down the line. The database designer also prompts the user to describe each table they create or adapt, although they are under no obligation to do so.

By capturing this information during the data creation process, we hope to save time during ingest, as the person responsible for adding the metadata to the archived data will have fewer elements that he or she will need to follow up. If this saves just 1 hour over cataloguing and depositing a database where this information has not already been captured, that would amount to about a £45 (FEC) saving per database ingested. We conservatively project that once the service reaches full maturity and the project created during the early part of the service start to move to long term archives, savings from metadata efficiency will amount to at least £4,000-£5,000 per annum.

5.1.4 ROI

The ROI estimations below are based on our current estimates of uptake at the initial pricing levels. The costs of developing the service have been borne by the Sudamih and VIDaaS Projects are *not* included in the model. Due to the high fixed-cost component, the model is very sensitive to variations in take-up. We will therefore need to review prices and support levels regularly and adjust accordingly.



The model is one of cautious initial uptake followed by an increasing number of projects using the ORDS each year as new functionality is added (expanding the potential market) and confidence in the service grows (as early adopters help spread the word and their colleagues better understand the advantages of the ORDS). At present we anticipate demand for new projects levelling off at about 75 new regular projects per year (for an average of 5-years live hosting) and 18 or so new large projects per year.

We anticipate reaching a break-even point during 2014, and achieving a cumulative balance of operating income versus costs during 2017.

The efficiencies and cost-savings deriving from the use of the ORDS within the University will of course be in addition to the purely income-based returns indicated above. As is hopefully apparent from the benefits analysis above, we are not yet in a position to quantify these cost-savings and efficiencies with enough accuracy to factor them into the above graph, but we should be able to improve our initial estimates by monitoring and surveying ORDS users post-launch.

5.2 Market differentiation

At a very broad estimate we would suggest that the DaaS might eventually serve a use for about 20% of researchers in UK HE, although it will require significant further feature development to reach this level of penetration.

The notion of building a database as a service is not unique, although we have not encountered any other products build specifically for use in HE.

VMware's vFabric Data Director[7] also provides database as a service, but on commercial terms to corporate business users rather than specifically for researchers in the HE sector. Data Director is closed source, whereas the DaaS is available as open source software, enabling it to be modified to meet any particular requirements that different institutions might have and allowing it to be adapted to different infrastructures.

'Grubba'[8] is an example of a simple online 'database as a service' designed for use by the general public and small businesses. It is reliant on donations and the enthusiasm of its creator for sustainability. As it says on the service website, "Keeping Grubba running is expensive, so every donation is much appreciated. You can use Grubba up to 10.000 records and 5 forms." Whilst this may suffice for some small projects, the service is not built on open source software and is likely to be too limited for many research projects. It is not designed to be used as a component within a broader data management lifecycle infrastructure, and the long-term sustainability of the service is also hard to confirm at present.

We have worked with VMware during the development of the DaaS and will discuss opportunities for collaboration with their Data Director team after the project.

---

[7] http://www.vmware.com/products/application-platform/vfabric-data-director/overview.html

[8] http://www.grubba.net/

5.3 Operational Readiness

Dependencies
The DaaS is intended to be run on a VMware cloud infrastructure. An organization wishing to offer the DaaS as a service to its members would need to host the software on such an infrastructure. Commercial public cloud providers can of course be used for this purpose. Institutions would also need to connect the DaaS to other parts of their data management infrastructure if they wish to, say, transfer data and metadata held by the DaaS into their own long-term data archives, or make regular back-ups of user data to their institutional backing-up service, or so forth.

Operational readiness		
Function / facility	**Current**	**Required for operation**
Source code (for software installation)	DaaS source code is publicly available & in the process of being tested and toughened. We will need to migrate the software to a more secure application server than JBoss 5 before officially releasing it.	All software component source code online and packaged for easy deployment
VMware Cloud infrastructure	Software running on Oxford Private Cloud VMware cloud infrastructure, and tested on other infrastructures including Eduserv and Colt Public cloud	Already in place.
Helpdesk	Technical issues tracked via Google Code; project issues via project email address; enquiries dealt with by project team	Help-desk service consisting of email support – closely connected with development staff. Main OUCS helpdesk will point inquiries to the ORDS Service Team.
User/training materials	Step-by-step ‘how to’ guides in place, but will need revision before service launch.	Linked support documentation & accompanying training materials (user & support staff)

Staffing	Post VIDaaS there will be 1.2 FTE of staff available to complete the DaaS migration to a supported application server before the service launch in the Summer of 2012	Service Team (estimated 1.4 FTE+) (for Oxford local service only)
Delivery mechanism	Firefox Web browser	Firefox and Internet Explorer Web browsers (initially)
Organizational structure	Project Team	Service Team managed as part of OUCS SysDev
Accessibility standards	Working on interfaces to meet accessibility standards	Interfaces meet common accessibility standards
Open source software community	Source code is publicly available via Google Code, but only partially documented, and without external community	Source code publicly available via Google Code, with governance model in place and tools to facilitate community participation.

5.4 Business model and sustainability route

Options summary:

Option No.	Sustainability Route	Business Model	Transition Cost	Operational Cost*	Overall Risk Score	Preferred option?
1	Local Oxford ORDS Service	*Charge for core service / Cost recovery* Researchers pay to use service at a price that covers costs. Public access to data is free (if enabled via privacy settings). Charges included in funding proposals, possibly with departmental subscriptions to enable use by 'unfunded' researchers. Value added services may be build upon core service in time. Oxford continues to improve DaaS software.	£334,640 assumes 1st year of three is transitional	£918,432 Over 3 years	Medium	Yes
2	DaaS Software Only	*(coordinated) Community Model* DaaS software is packaged and available for free. Each institution can build whatever service they see fit on top of the software and host it on any VMware-based cloud infrastructure. There is nothing to stop any given service provider providing the service beyond their institution. One transitional year during which Oxford further develops DaaS.	£153,351 1 year to further develop DaaS & build community	£153,351 Over 1 year	Medium	No.
3	National Service – User Charge Model	*Charge for core service* One organization runs the DaaS as a national provider charging research projects to use the supported DaaS. Value added services may be build upon core service in time.	Approx. £566,694 1st year transitional	Approx. £1,735,110 over 3 years	Medium	Long-term 2nd choice
4	National Service – Subscription Model	*Membership / Consortium model* One organization runs the DaaS as a national provider offering a supported DaaS to subscribing institutions. Value added services may be build upon core service in time.	Approx. £566,694 1st year transitional	Approx. £1,735,110 over 3 years	Medium	Long-term preference

* This is the cost of providing the software/service for three years at zero revenue. See section 5.1.4 for revenue predictions.

5.4.1 Option details

5.4.1.1 Option 1 : Local Oxford ORDS service

Variation from standard Description given in section 1
Option 1 involves setting up a local service within University of Oxford – we have now completed the planning required to implement this option, although we will need to migrate the DaaS software to a supported application server before officially launching. The Oxford local instance of the DaaS will be entitled the 'Online Research Database Service' (ORDS) and utilize the Oxford private cloud infrastructure in the first instance (for data security), with public cloud operation options available via the Janet brokerage shortly afterwards. The DaaS software will be maintained and updated by University of Oxford staff as part of the service whilst we seek to develop a broader open source software community. This option assumes that besides providing the service, Oxford continues to coordinate the development of the underlying DaaS software and provides resources to fix bugs and implement additional functionality in the software (thus encompassing the work set out independently in option 2) The ORDS service is provided on a cost-recovery basis, with the project PI paying an upfront fee for their project to be allocated its own VM for a certain number of years. VMs are offered in two standard models: regular (half a CPU, 1GB RAM, 25GB disk space); and large (2 CPUs, 4GB RAM, 100GB disk space). Projects with specific requirements not met by these standard models will be able to negotiate customized VMs. The regular model will be available for £600 per year, and the large at around £2,000 per year. Long term archiving is not included in these prices and will be determined independently during the DaMaRO Project (which concludes in 2013). Whilst feedback suggests that these prices will be regarded as reasonable for funded research, it may be difficult for departments to bear these prices for unfunded research. This is an issue which we have begun to discuss with the Divisions within the University.
Transition timescale
This service will officially commence in the Summer of 2012. The University of Oxford has agreed to fund a follow-on project to improve functionality and promote the service within the University. We are currently applying for a second tranche of funding (from JICS/HEFCE) to accelerate adoption and consider how the service could be transformed into a national service (see options 3 and 4 below).

5.4.1.2 Option 1 risk analysis

Risks associated with this option					
Risk	**Impact**	**Probability**	**Risk level**	**Risk strategy**	**Mitigating action(s)**
Service too expensive for 'unfunded' research projects	Medium	High	4	Reduction	This is a known issue. We will need to come up with a second business model to reach such users – possibly by offering the service on a subscription model to departments
Slow uptake means service doesn't cover costs	High	Medium	4	Reduction	Publicize advantages of service; possibly provide subsidy to attract early adopters
System not intuitive enough to attract users	High	Medium	4	Reduction	Test interfaces with users; provide clear documentation; provide alternative front-ends to system
Potential users prefer other database management systems	Medium	Medium	3	Reduction / Transference	Enable users to use their preferred software as a front-end to the ORDS
Privacy/security issues put users off	Medium	Low	2	Reduction	Ensure good security; include security levels in SLD
Difficult to find appropriate staff given existing pressures on staff time at OUCS	Medium	Low	2	Reduction	Where possible assign resources to existing services for coordination (such as ITLP for training)
No transitional funding from JISC/HEFCE	High	Unknown	?	Contingency	Whilst the University is committed to going ahead with the ORDS, additional funding is likely to significantly improve the longer-term service sustainability, mitigate some of the risks identified above, and enable the University to further pursue options 3 and 4 below.

5.4.1.3 Option 1 costs

	Year 1 (transition)	**Year 2**	**Year3**	**TOTAL**
Directly Incurred Staff *(Post, Grade, No. Hours & % FTE)*				
Bug fixing, continued development, & OSS governance g.8, 1.0 FTE 1st year, 0.5 FTE thereafter	£56,252	£28,540	£28,968	£113,760
Platform upgrades, g.8, 0.2 FTE	£11,440	£11,968	£12,524	£35,932
SysAdmin & service monitoring, g.8, 0.2 FTE	£11,440	£11,968	£12,524	£35,932
Updating documentation & support, g.7, 0.1 FTE	£4,538	£4,608	£4,676	£13,822
Training provision, g.7, 0.1 FTE	£4,538	£4,608	£4,676	£13,822
Help Desk support, g.8, 0.2 FTE	£11,260	£5,720	£11,596	£28,576
Total Directly Incurred Staff (A)	**£99,468**	**£67,412**	**£74,964**	**£241,844**
Non-Staff				
Travel and expenses	£2,569	£2,639	£2,712	£7,920
Hardware/software (estimated cloud infrastructure costs + licences)	£112,338	£117,844	£124,902	£355,084
Dissemination (marketing, sales support, help desk etc)	£2,055	£2,112	£2,170	£6,337
Evaluation	£514	£528	£542	£1,584
Other (internal service to service costs)	£10,512	£10,801	£11,098	£32,411
Total Directly Incurred Non-Staff (B)	**£127,988**	**£133,924**	**£141,424**	**£403,336**
Directly Incurred Total (C) (A+B)	**£233,092**	**£207,056**	**£222,192**	**£662,340**
Directly Allocated				
Staff: Adding/supporting new IaaS, g.8, 7 weeks (280 hours)	£9,303	£9,443	£9,584	£28,330
Estates	£6,941	£4,958	£5,467	£17,366
Other	£0	£0	£0	£0
Directly Allocated Total (D)	**£16,244**	**£14,401**	**£15,051**	**£45,696**
Indirect Costs (E)	**£90,940**	**£64,977**	**£71,639**	**£227,556**
Total Cost (C+D+E)	**£334,640**	**£280,714**	**£303,078**	**£918,432**

5.4.2 Option 2: DaaS Software Only

Variation from standard Description given in section 1
Under option 2 the DaaS software will be packaged to be installed and run as a local service by individual institutions. Each institution wishing to offer a service based on the DaaS software will need to bear the responsibility of hosting it on appropriate cloud infrastructure and offering some level of user support and training. It is assumed that institutions will price the service independently according to their local costs and usage expectations. The University of Oxford is envisaged as continuing to take a governing and coordinating role in the future community development of the open-source DaaS software. Oxford would also continue to develop new features and 'harden' the DaaS code whilst bringing the software to the attention of the open source development community during a transitional year. Beyond that it is expected that code development will be driven by the open source community and institutions running the DaaS-based services. The University of Oxford offers no support to other institutions beside the documentation and training materials created in the course of the VIDaaS Project and the one-year transitional period. This option as described here assumes that the University of Oxford will NOT be running the planned Online Research Database Service (ORDS) – i.e. the software exists by itself, but does not (yet) have any service in place to provide it to users in a managed environment. There is, therefore, no long-term revenue stream attached to this option – a service would need to be added at a later point to begin recouping the costs of further software development.
Transition timescale
We will need to migrate the DaaS software to a currently-supported application server and update the installation instructions before this option is viable, but this is already in hand and should be complete by June/July 2012. Setting up the software at other institutions should not be a difficult or time-consuming process.

5.4.2.1 Option 2 risk analysis

Risks associated with this option					
Risk	**Impact**	**Probability**	**Risk level**	**Risk strategy**	**Mitigating action(s)**
Other institutions not prepared to take financial risk of building services around DaaS	High	High	5	Reduction	Without the example of the originating institution offering a successful DaaS service, other institutions are likely to be reluctant to take the risk of building their own services. If Oxford sets an example (and provides a model) this risk should be significantly reduced.
Other institutions develop DaaS software in an uncoordinated way, causing project to 'fork'	Medium	Medium	3	Prevention	In order to prevent every institution creating their own customized version of the DaaS (and therefore failing to pass the benefits of their development work on to the HE community), there needs to be a paid-for element of OSS governance and communication
DaaS software at April does not have enough functionality to address potential users' requirements	High	Medium	4	Reduction	We know that by April the DaaS software will include enough functionality to meet simple use cases, however many users have additional requirements, and they will not use the DaaS unless it can meet these requirements. We will need to fund additional software development in order to meet these requirements.
Idea of DaaS is not appropriately supported at senior level and fizzles out	High	Medium	4	Reduction	Under this option there is a real risk that enthusiasm for the DaaS simply fades and the software is effectively consigned to the dustbin of open source software history with no active development community, unless there is ongoing promotion work both within Oxford and to IT providers at other universities.

5.4.2.2 Option 2 costs

	Year 1 (transition)	Year 2	Year3	TOTAL
Directly Incurred Staff *(Post, Grade, No. Hours & % FTE)*				
Project management, g.8. 0.2 FTE	£11,260	£0	£0	£11,260
Bug fixing, continued development, OSS governance, promotion to development community g.8, 1.0 FTE 1st year	£56,252	£0	£0	£56,252
Updating documentation & support, g.7, 0.2 FTE	£9,076	£0	£0	£9,076
Total Directly Incurred Staff (A)	**£76,588**			**£76,588**
Non-Staff				
Travel and expenses	£2,569			£2,569
Evaluation	£514			£514
Dissemination	£4,110			£4,110
Total Directly Incurred Non-Staff (B)	**£7,193**			**£7,193**
Directly Incurred Total (C) (A+B)	**£83,781**			**£83,781**
Directly Allocated				
Staff: Paul Jeffreys (PI), 200 hours	£0			£0
Estates	£4,933			£4,933
Other	£0	£0	£0	£0
Directly Allocated Total (D)	**£4,933**			**£4,933**
Indirect Costs (E)	**£64,637**			**£64,637**
Total Cost (C+D+E)	**£153,351**			**£153,351**

5.4.3 Option 3 : National Service – Usage Charge

Variation from standard Description given in section 1
Option 3 envisages the DaaS provided as a centralized national service. Under this model there is one national organization that provides the DaaS as a service to any researchers in the UK with the resources to pay for it. This option (along with option 4) is likely to offer the greatest economies of scale as there is no duplication of support mechanisms across multiple organizations. It will also be the easiest model under which to coordinate software development and upgrades. Once the success of the DaaS has been proved at a local level it would make sense to invest in centralizing it. As with option 1, this option assumes that a charge is levied upon the users of the service which, as usage grows and fixed costs per project decline, reaches a point at which the costs of running the service is fully recovered or even exceeded.
Transition timescale
This is a longer-term option as it would involve the creation of a new national organization, which could be a JISC service or a company spun-out of the University of Oxford or some other organization.

5.4.3.1 Option 3 risk analysis

Risks associated with this option					
Risk	**Impact**	**Probability**	**Risk level**	**Risk strategy**	**Mitigating action(s)**
Lack of trust in service provider	Medium	Low	2	Reduction	Without the security of knowing that the service is being provided by the user's host institution, researchers may be reluctant to trust it with potentially sensitive research data. Universities should make it very clear that they support their researchers use of the service
Service too expensive for 'unfunded' research projects	Medium	High	4	Reduction	The same problem as identified in Option 1, but on a national scale. Will require some sort of secondary business model to cater for unfunded research
Slow uptake means service doesn't cover costs	High	Low	3	Reduction	Given the size of the potential market against the only slightly higher service costs of option 1 this risk is greatly reduced. Need to ensure advantages of service are well publicized
Unclear route to long-term archiving	Medium	Medium	3	Contingency	Without a particular long-term institutional data archive to move data that is no longer being frequently accessed to, it may not be clear where data should be placed once funding for live provision runs out. The National Data Centres obviously have a role, but negotiations will need to be conducted with institutions to establish routes for long-term data preservation and retrieval
Potential for data loss on a national scale	High	Low	3	Contingency	An agreement will need to be in place to determine what happens to the data hosted by the service in the event of its closing down
No financial support from JISC/HEFCE to set up national service	High	?	?	Acceptance	Without funding and support for creating a central service this option may never be taken up, and UK HE will not benefit from the significant economies of scale it offers. An individual institution *may* simply decide to start offering the DaaS as a national service, but there is of course no guarantee that they will wish to take on the risk, and there may be legal hurdles to negotiate.

5.4.3.2 Option 3 costs (These figures are very rough estimates – actual costs will depend on infrastructure used)

	Year 1 (start-up)	Year 2	Year3	TOTAL
Directly Incurred Staff *(Post, Grade, No. Hours & % FTE)*				
Service management g.9, 0.5 FTE	£32,298	£33,802	£35,378	
Bug fixing, continued development, & OSS governance g.8, 1.0 FTE	£56,252	£57,096	£57,956	
Platform upgrades, g.8, 0.2 FTE	£11,440	£11,968	£12,524	
SysAdmin & service monitoring, g.8, 0.5 FTE	£28,564	£29,990	£31,294	
Updating documentation & support, g.7, 0.2 FTE	£9,076	£9,216	£9,352	
Training provision, g.7, 0.5 FTE	£22,736	£23,072	£23,416	
Help desk support, g.8, 1.0 FTE	£56,252	£57,096	£57,956	
Total Directly Incurred Staff (A)	**£216,618**	**£222,150**	**£227,876**	**£666,644**
Non-Staff				
Travel and expenses	£5,138	£5,279	£5,424	
Hardware/software (estimated cloud infrastructure costs + licences)	£110,283	£115,732	£122,732	
Dissemination (marketing, sales support, help desk etc)	£10,275	£6,335	£6,509	
Evaluation	£514	£528	£542	
Consumables	£10,275	£5,279	£5,424	
Total Directly Incurred Non-Staff (B)	**£138,540**	**£135,265**	**£142,801**	**£416,606**
Directly Incurred Total (C) (A+B)	**£355,158**	**£357,418**	**£370,677**	**£1,083,250**
Directly Allocated				
Staff: Adding/supporting new IaaS, g.8, 7 weeks (280 hours)	£9,303	£9,443	£9,584	
Estates	£14,340	£14,732	£15,137	
Other	£0	£0	£0	
Directly Allocated Total (D)	**£23,643**	**£24,175**	**£24,721**	**£72,539**
Indirect Costs (E)	**£187,893**	**£193,059**	**£198,369**	**£579,321**
Total Cost (C+D+E)	**£566,694**	**£574,649**	**£593,767**	**£1,735,110**

5.4.4 Option 4 : National Service – Subscription Model

Variation from standard Description given in section 1
This option is essentially the same as option 3, but with a different charging model – instead of individual research groups paying to use the DaaS service, this model assumes a subscription model where universities pay a regular subscription in exchange for their staff to have access to the DaaS as needed. This would give the organization running the DaaS a more predictable cash flow, although there may need to be some sort of rationing implemented to prevent the over-consumption of the DaaS by individual researchers (variable costs per VM might start to become a sustainability issue).
Transition timescale
This is a longer-term option as it would involve the creation of a new national organization, which could be a JISC service or a company spun-out of the University of Oxford or some other organization.

5.4.4.1 Option 4 risk analysis

Risks associated with this option					
Risk	**Impact**	**Probability**	**Risk level**	**Risk strategy**	**Mitigating action(s)**
Universities unconvinced by economic case for membership	High	Medium	4	Reduction	Institutions may be reluctant to pay a flat subscription fee without a clear idea of how many of their staff would actually use and benefit from the DaaS. An introductory price or more flexible pricing model may be required
'Over consumption' by researchers	Low	Medium	2	Reduction	With the service provided free at the point of use, researcher may set up numerous different projects just to play around with, not appreciating that each requires a separate VM. This should, however, be relatively easy to avoid either via imposing limits or via changes to the underlying code to enable multi-tenancy and/or better scaling
Unclear route to long-term archiving	Medium	Medium	3	Contingency	(As with option 3) Without a particular long-term institutional data archive to move data that is no longer being frequently accessed to, it may not be clear where data should be placed once funding for live provision runs out. The National Data Centres obviously have a role, but negotiations will need to be conducted with institutions to establish routes for long-term data preservation and retrieval
Potential for data loss on a national scale	High	Low	3	Contingency	(As with option 3) An agreement will need to be in place to determine what happens to the data hosted by the service in the event of its closing down
No financial support from JISC/HEFCE to set up national service	High	?	?	Acceptance	Without funding and support for creating a central service this option may never be taken up, and UK HE will not benefit from the significant economies of scale it offers. An individual institution *may* simply decide to start offering the DaaS as a national service, but there is of course no guarantee that they will wish to take on the risk, and there may be legal hurdles to negotiate.

5.4.4.2 Option 4 costs (These figures are very rough estimates – actual costs will depend on infrastructure used)

	Year 1 (start-up)	Year 2	Year3	TOTAL
Directly Incurred Staff *(Post, Grade, No. Hours & % FTE)*				
Service management g.9, 0.5 FTE	£32,298	£33,802	£35,378	
Bug fixing, continued development, & OSS governance g.8, 1.0 FTE	£56,252	£57,096	£57,956	
Platform upgrades, g.8, 0.2 FTE	£11,440	£11,968	£12,524	
SysAdmin & service monitoring, g.8, 0.5 FTE	£28,564	£29,990	£31,294	
Updating documentation & support, g.7, 0.2 FTE	£9,076	£9,216	£9,352	
Training provision, g.7, 0.5 FTE	£22,736	£23,072	£23,416	
Help desk support, g.8, 1.0 FTE	£56,252	£57,096	£57,956	
Total Directly Incurred Staff (A)	**£216,618**	**£222,150**	**£227,876**	**£666,644**
Non-Staff				
Travel and expenses	£5,138	£5,279	£5,424	
Hardware/software (estimated cloud infrastructure costs + licences)	£110,283	£115,732	£122,732	
Dissemination (marketing, sales support, help desk etc)	£10,275	£6,335	£6,509	
Evaluation	£514	£528	£542	
Consumables	£10,275	£5,279	£5,424	
Total Directly Incurred Non-Staff (B)	**£138,540**	**£135,265**	**£142,801**	**£416,606**
Directly Incurred Total (C) (A+B)	**£355,158**	**£357,418**	**£370,677**	**£1,083,250**
Directly Allocated				
Staff: Adding/supporting new IaaS, g.8, 7 weeks (280 hours)	£9,303	£9,443	£9,584	
Estates	£14,340	£14,732	£15,137	
Other	£0	£0	£0	
Directly Allocated Total (D)	**£23,643**	**£24,175**	**£24,721**	**£72,539**
Indirect Costs (E)	**£187,893**	**£193,059**	**£198,369**	**£579,321**
Total Cost (C+D+E)	**£566,694**	**£574,649**	**£593,767**	**£1,735,110**