Deliverable

D6.4

Final Evaluation of the Sensitive Data Test Bed

WP6. Sharing Sensitive Scientific Data Test Bed

February 2011
Version 1.0

Consequence

Context-aware data-centric information sharing

FP7-ICT-2007-1
ICT-2007.1.4. Secure, dependable and trusted Infrastructures
Grant Agreement 214859

LEGAL NOTICE

The following organizations are members of the Consequence Consortium:

Europäisches Microsoft Innovations Center GmbH,

BAE SYSTEMS (Operations) Limited,

Hewlett-Packard Italiana,

Imperial College of Science, Technology and Medicine,

The Science and Technology Facilities Council,

Consiglio Nazionale delle Ricerche,

Centre for Research and Telecommunication for Networked Communities.

Project acronym: Consequence

Project full title: *Context-aware data-centric information sharing*

Work Package: **6**

Document title: **Final Evaluation of the Sensitive Data Test Bed**

Version: **1.0**

Official delivery date: **28 Feb 2011**

Actual publication date: 09 Feb 2011

Type of document: Report

Nature: Public

Authors: Shirley Crompton, Michael Wilson

Approved by: David Golby

Version	Date	Sections Affected
1	Feb 2011	Full deliverable for project review

Table of contents

Executive Summary

This report describes the evaluation of an implemented research prototype application developed within the Framework for context aware data-centric information sharing produced in the Consequence project. The domain of the application is to provide controlled usage and secure access to STFC scientific data.

STFC operates large scientific facilities for the UK research community – e.g. synchrotrons, neutron sources, satellites, and petawatt lasers. These facilities produce vast volumes of scientific data under public and commercial funding. These facilities are used by collaborative teams of researchers from different universities and companies lead by principal investigators. Scientific funding bodies wish to maintain data security for the principal investigator to gain the scientific rewards and financial benefits which can result from their research, while also opening up data access to encourage:

- validation of analyses by other scientists,
- reuse of data for secondary studies,
- use of the data for other analyses not considered by its creator.

In order to open up data access the data management systems which support science have to implement more precise control over the data for which they are responsible. The Consequence Framework is based on legally binding Data Sharing Agreements (DSA) between enterprises such as facility providers, universities and companies. It provides a methodology, tools and application programmer interfaces (API) to allow the drafting of a DSA, and the technical enforcement of the data management policies in them. The policies address authorisation, prohibition and obligations on parties to the DSA. The enforcement addresses both on-line and off-line access to, and usage of the data.

The STFC prototype application supports searching, downloading and usage of scientific data within the Consequence Framework.

The application is evaluated for its conformance to security, usability, technical and business requirements.

The Consequence project took several technologies which have been recently reported in the computer science research literature and integrate them together into a single framework which the prototype incorporated. An ambitious aim was to move these technologies from a state of being reported individually in the research literature to being able to be incorporated by non-expert developers into demonstration projects. The evaluation shows that many of the technical requirements of the project have been met. The technologies which have been combined include ontology editors and software engineering formal methods tools which require explanation to most potential users not only at a detailed level but also at a deep conceptual level, since the functions they perform are completely novel to the target audience. Other technical issues (e.g. data policies for derived data) have been addressed in the Framework and as research topics which have been published in the computer science literature, but have not been included in the implemented prototype. As a result of addressing these ambitious technical objectives within the project, there are weaknesses in the usability of the system and in the completeness and accuracy of the technical documentation. These weaknesses have resulted in a prototype which demonstrates how the technologies operate together, but not a toolset which can be used by new users to develop their own demonstrators without additional support.

Recommendations are made throughout the report of future work which needs to be undertaken to move the technology forward towards adoption and assimilation into data management systems.

1. Introduction

This document describes the evaluation of the "Sharing Sensitive Scientific Data" test bed. The test bed uses a multi-organisation research project in structure-based biology to highlight the data sharing requirements in scientific research as gathered from practising scientists and providers of STFC large research facilities.

This section is a general introduction to the STFC, the scientific issues it addresses and how they relate to data management. The Science and Facilities Research Council (STFC) provides large scale research facilities for UK researchers to use in the UK and overseas. In either case the facilities themselves are usually international partnerships with research organisations around the world. UK researchers use these facilities to produce scientific data which they wish to analyse at a later date, and which they will use as part of scientific research studies which usually result in academic research publications and sometimes patents. The facilities provided include those used in Astronomy and Particle Physics as well as in the general structural sciences.

In the structural sciences facilities are used to determine the structure of materials which usually have an impact in microelectronics or biosciences. The most famous structure investigated using STFC facilities is the structure of the ribosome in all human cells. The ribosome is a very complex molecule which operates as a protein factory, obeying the commands of RNA messages from the cell nucleus to create proteins required by the cell.

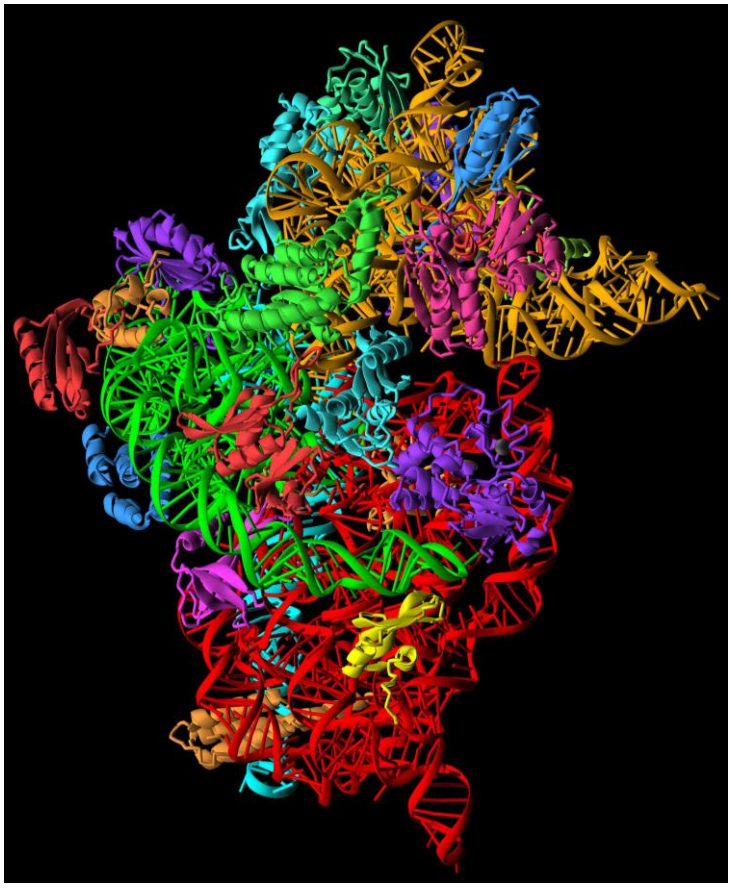

Figure 1: Structure of the human ribosome.

Venkatraman Ramakrishnan of the MRC Laboratory of Molecular Biology, Cambridge, was awarded the 2009 Nobel Prize for chemistry for determining the structure of the ribosome. He used several neutron and synchrotron light sources in Europe and the USA to collect his data, and then used several super-computers in the UK and USA to determine and model its structure. The data had to be shared by a large team of support teams in different institutions, including what is now the STFC Daresbury Laboratory where he used the Synchrotron Radiation Source (SRS) and supercomputer provision.

In astronomy, the discovery in 2004 of the dwarf planet 136108 Haumea in the Kuiper belt with a mass one-third the mass of Pluto illustrates another aspect of the sensitivity of scientific data. Its discovery was first announced by José Luis Ortiz Moreno and colleagues from the Sierra Nevada Observatory in Spain who laid claim to the scientific achievement. Michael Brown of the California Institute of Technology originally indicated his support for Ortiz's team being given credit for the discovery of Haumea. However, further investigation showed that a website containing archives of where Brown's team's telescopes had been pointed while tracking Haumea had been accessed eight times in the three days preceding Ortiz's announcement, by computers with IP addresses that were traced back to the website of the Instituto de Astrofísica de Andalucía (CSIC, Institute of Astrophysics of Andalusia), where Ortiz works, and to e-mail messages sent by Ortiz and his student. These website accesses came a week after Brown had published an abstract for an upcoming conference talk at which he had planned to announce the discovery of Haumea; the abstract referred to Haumea by a code that was the same code used in the online telescope logs; and the Andalusia computers had accessed the logs containing that code directly, as would be the case after an internet search, without going through the home page or other pages of the archives. The scientific community generally recognises Michael Brown as the discoverer of the dwarf planet, and considers the actions of Ortiz and his team to have been unethical.

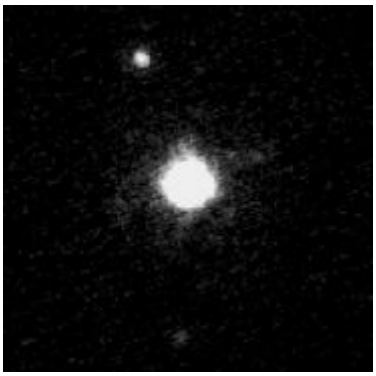

Figure 2: Keck telescope image of Haumea and its two moons. Hi'iaka is above Haumea (centre), and Namaka is directly below.

In the case of Venkatraman Ramakrishnan, scientific data about the structure of the ribosome was shared with large teams who supported his work, yet there was no leak of the data, and he legitimately achieved scientific reward. In the case of Haumea, data did leak out and there was a questionable claim of discovery which has tarnished the reputation of all those involved – even those who are clearly innocent.

Scientific discoveries resulting from the analysis of data collected at facilities of the class operated by the STFC can result in considerable scientific value, and further down the value chain, considerable business profits from new drugs and new microelectronics materials.

Historically scientific data has been protected by the limited information technology controlled by the principal investigator on each individual study who has maintained it themselves. Although this local approach has maintained security, it has also limited access to the data, preventing validation of analyses by other scientists, reuse of data for secondary studies, and use of the data for other analyses not considered by its creator. Today scientific funding bodies wish to open up access to data to promote these three uses, while maintaining data security for the principal investigator to gain the scientific awards and financial benefits which can result from it. This need to open up data access while controlling its usage has lead to large scientific data centres which require policy based management of the data. Consequence provides a policy based solution to this problem. This report describes the evaluation of the Consequence framework and tools as a method to control the data, rather than either totally limiting access or totally opening it to all.

Details of the STFC application and the three use cases were presented in the deliverable D6.1 in December 2008 [1]. The details presented there will not be repeated here unless they have changed over the two intervening years, or are necessary to explain the evaluation.

This report continues with a brief description of the problem addressed and the scenario to be addressed in section 3, followed by a description of the prototype architecture in section 4, then the evaluation procedure and results in section 5. The report ends with a conclusion of the evaluation and recommendations for future work in Section 6.

2. Problem Statement and Scenario

In a knowledge-based society, access to exclusive information confers competitive advantages. Public funding agencies and research organisations, including STFC, are keen to foster on-line community research resources in order to maximise access, to promote inter-disciplinary and cumulative research. In the academic domain, data arising from publicly funded research is considered public asset. According to guidelines of the Organisation for Economic Co-operation and Development (OECD), this resource should be made openly available to the maximum extent possible. This principle underpins the agenda of the coalition of UK Research Councils (RCUK) to enrich society and contribute to the national economy through leveraging research to promote knowledge transfer and innovation. However, to encourage a free flow of research data, there must be sufficient safeguards to protect a contributor's intellectual claims or property rights – which, after all, motivate research and innovation. Moreover, there may be legislation such as the Data Protection Act (DPA) 1998 which governs the publication of information. For instance, research involving the use of National Health Service (NHS) patient data must be anonymised before public release to protect the privacy of the human data subjects. Similarly, stakeholders to the research data may themselves have specific policies or requirements which limit how the data can be shared. For example, the STFC ISIS and Diamond Light Source (DLS) facilities require that experimental data obtained on their facilities by publicly-funded projects be released into the public domain after an initial period of exclusive use (embargo period). It is sharing data during this exclusive time window that presents the biggest challenge to data owners and providers. The data owners may have commercial patents or scholarly articles pending and these would be invalidated by prior publication of the data. The challenge is how to protect data owners' intellectual property right (IPR) without imposing excessive restriction that may stifle data exchange and impede research activities. Once academic data is released into the public domain, there is generally little concern regarding access. The emphasis is shifted towards securing data integrity and to track derived data for IPR or quality management purposes.

As a pragmatic solution, stakeholders commonly use legally binding data sharing agreements to control how their data is shared and disseminated. These agreements contain policy statements on the access, usage conditions and obligations for specific sets of data as well as references to external data sharing policies or protocols, like those of the funding agency and university hosts. Such agreements are usually drafted by senior managers and lawyers to express what can be decided in court should a breach occur. Enforcement is generally left to the discretion of the data owners, publishers and providers. In the academic domain, enforcement may range from simple mutual trust between individual researchers on one end of the spectrum, with data consumers expected to voluntarily observe the ethical and legal obligations pertaining to the data; to a complete lack of trust at the other end, with sensitive data secreted away on private repositories accessible to the selected few. A system based on mutual trust is simple to operate but not adequate to prove compliance as obligated by many data sharing policies or regulatory legislation. For instance, DPA requires an organisation to deploy appropriate technical and organisational security measures when processing personal data. The data processor must be able to demonstrate that such measures exist or risks prosecution. Similarly, there are many aspects in data sharing agreements relating to usage that are not addressed by a simple grant or no grant type access control system. For instance, a well-known drug company only uses approved, self-contained High Performance Computing (HPC) infrastructures to run simulations involving proprietary data; computing grids with nodes distributed across administrative domains are considered too risky.

Inter-disciplinary study and collaborative research with industry is changing the way researchers interact, share data and manage IPR. With increasing commercial exploitation and ambitious international experiments tackling grand research challenges, research data is becoming too expensive or even impossible to replace. To promote a free flow of research data in this complex environment, there is a need for a secure data sharing and dissemination framework that addresses issues such as context-aware usage and obligations, data integrity, derived data, privacy and confidentiality.

2.1. Business Context

In 2010 there were many industry commentators predicting the forthcoming data tsunami:

> "We managed to create 800,000 petabytes of digital information last year according to a study released today by IDC and EMC Corp.. The annual survey forecasts that the creation of digital data will increase to 1.2 million petabytes — or 1.2 zettabytes — by the end of this year. (A petabyte is the equivalent of a stack of DVDs stretching from here to the moon.) And by 2020, we'll have created 35 trillion gigabytes of data — much of that in the "cloud.""[4]

Specifically in science this data deluge has been projected by several sources [6][7], based on the growth in data from large scientific facilities such as those operated by the STFC – see the figure below for the NICE synchrotron.

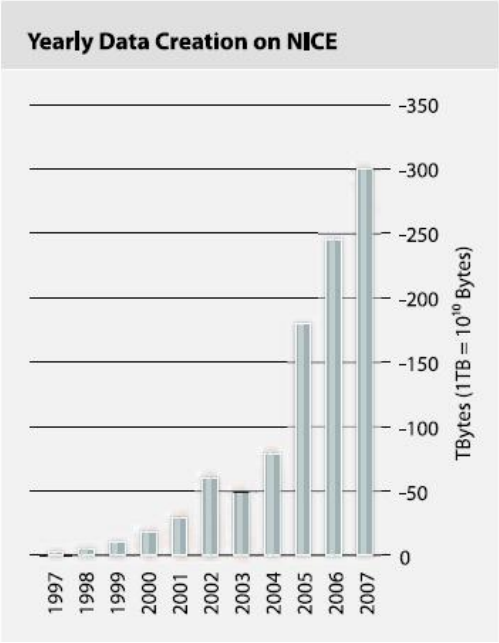

Figure 3: In Grenoble, the European Synchrotron Radiation Facility is a supermicroscope studying anything from the propagation of cracks in steel to the surface proteins on the influenza virus. in the decade to 2007, its annual data output rose more than a hundred-fold. And it is just one of about 50 synchrotrons world-wide.

Given the consensus that scientific data management needs are going to grow exponentially in the next decade, STFC has an urgent need to plan its infrastructure to manage this effectively so that the data curation costs are not overwhelming.

One aspect of scientific data management that is of concern is security. Data security (as illustrated in the introduction to this report) has not been a primary concern of scientific data management so far. But with the planned centralisation of scientific data management and the opening of scientific data to wider access (as argued in the introduction) this becomes a concern. This change for scientific data is not unique. As illustrated by the figures below, there is not only a projected growth in data volume, but also a growing demand for higher security levels on the bulk of the data in the digital universe over the next decade (from [5]).

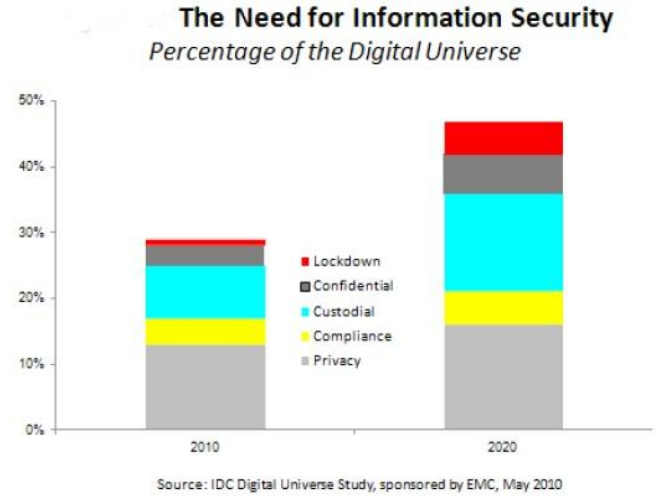

Figure 4: Projected changes in future security needs

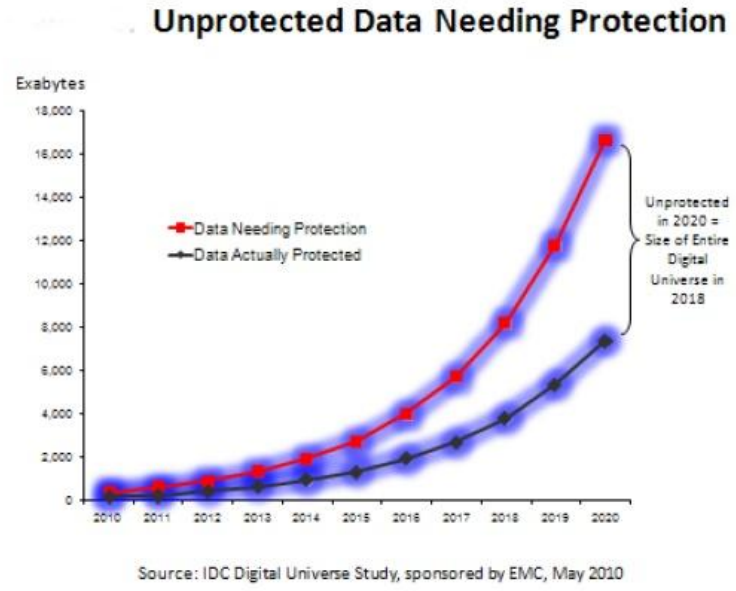

Figure 5: The growing gap between the data that needs protecting and that which actually is.

This growth in both data volume and security requirements requires an approach to data security which is more efficient than previously used to control costs.

Policy based security appears to meet this need. However, as the figure below shows, the problem being addressed is not only a technological one, but a human issue.

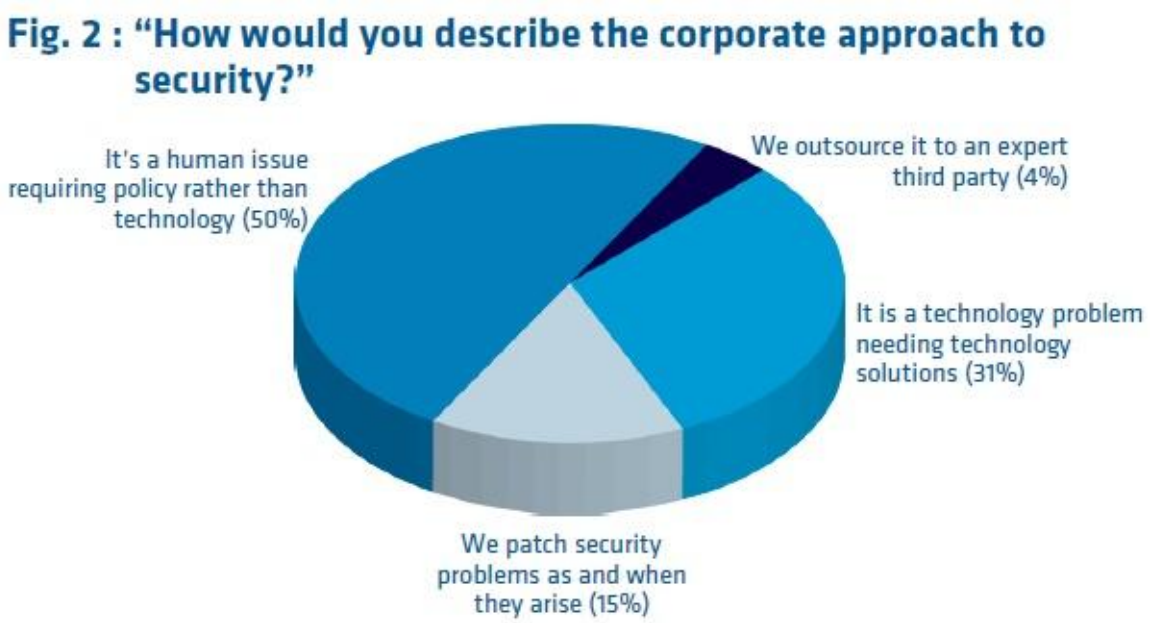

Figure 6: Survey of respondents to a *Computing* survey of 115 IT professionals working in enterprises of all sizes, with responsibility for security issues. [2]

From the perspective of the business of the STFC these projections make it clear that a solution to data management security which addresses both the technological solution and a human social one is crucial. Section 5.4 near the end of this report will evaluate whether the Consequence framework and tools meet that business need.

2.2. Scenario Story

The story behind the scenario is that STFC provides facilities to teams of scientists who bring samples of materials (often a crystal which a PhD student has spent 2 years purifying) to our facilities. The sample is loaded into one of the instruments on one of the beamlines. Subatomic particles are bombarded at the sample, which are then detected by the instrument which gives information about the structure of the sample material. This data is stored in the STFC scientific data infrastructure for members of the scientific team to download so that they can analyse it locally, and produce models of its structure which they will usually publish in the scientific literature. There may be a delay of several years before such publication while drug discovery or other work to exploit the result is undertaken. During this period, the members of the scientific team will download data to their local machines to use different pieces of software on it. The main piece of software will be the iCAT client tool, although Mantid and other analysis tools may also be used.

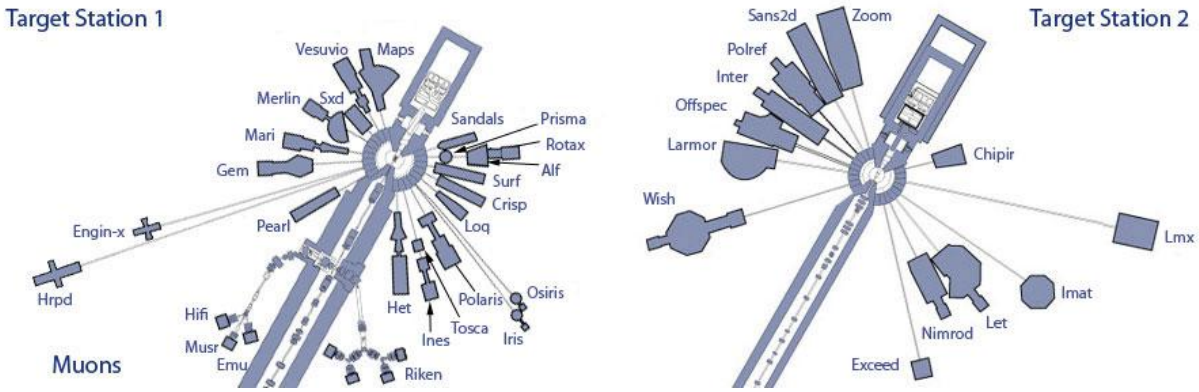

Figure 7: 33 instruments on the beamlines emanating from the 2 target stations of the ISIS facility.

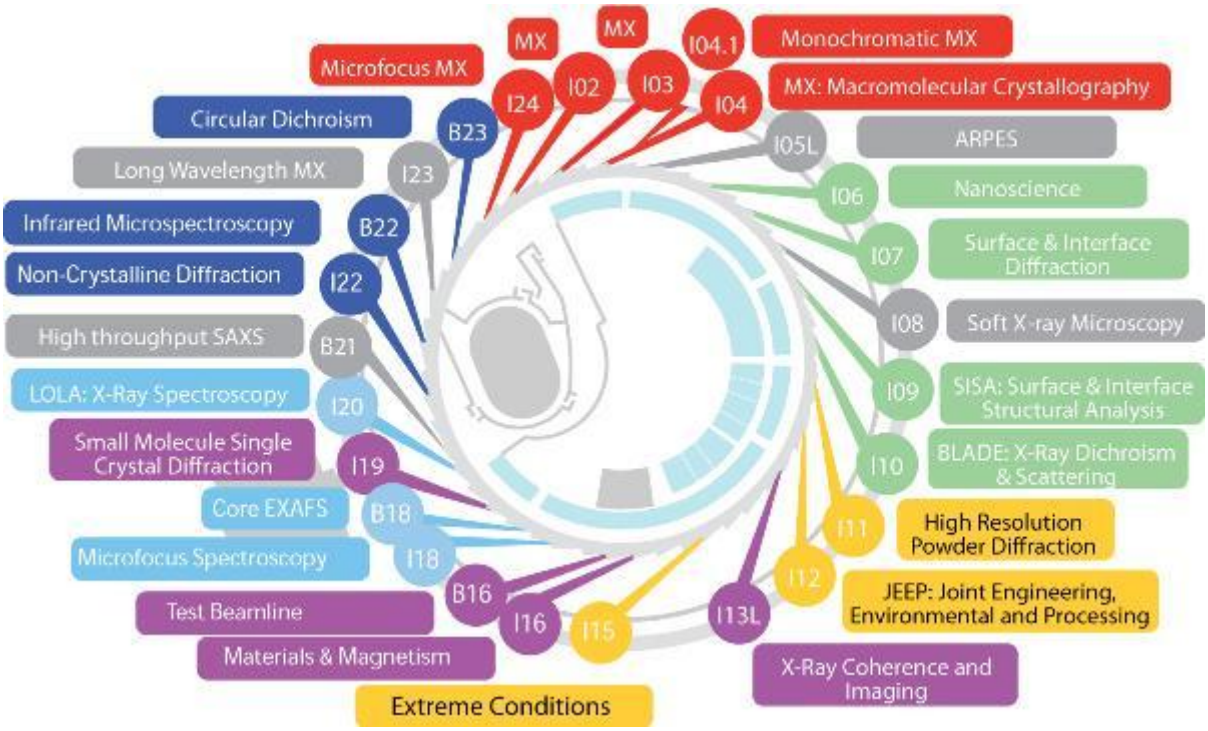

Figure 8: 26 instruments on the beamlines of the Diamond Light Source (DLS) synchrotron facility.

Before the experiment can take place a Data Sharing Agreement (DSA) must be in place between the STFC and each organisation involved in the research team to ensure the appropriate policies are in place to manage the data from the experiment. These usually include a period during which access to the data is embargoed for anybody except the project team. This period is usually 3 years, although this can be altered in the specific DSA.

After the experiment and the embargo period have elapsed, the data usually reverts to being accessible to all users of the STFC facilities and other guests. Guest users may be anybody who provides enough information that their identity can be identified and their system usage monitored.

2.3. Stakeholders

When the ISIS neutron and muon facility or the Diamond Light Source (DLS) synchrotron (photon) facility are used to provide data on the structure of a material, there are several stakeholders who need to be considered from different parts of the scientific lifecycle:

1. Principle Investigator (PI) of experimental team
2. Co-investigator in the experimental team – partner organisation
3. Beam-line scientist for STFC facility
4. Manager at PI organisation
5. Manager from partner organisation
6. Manager of STFC facility organisation
7. Lawyers for the managers
8. Manager at funding body of the PI

The first three in this list are individuals who will require access to data. The remaining five are individuals who will be involved in the drafting of the DSA.

2.4. Use Cases

Three use cases have been examined and used to provide both initial requirements for design and the evaluation of the Consequence prototype. The three cases address three stages in the lifecycle of the DSA and its usage:

- DSA Agreement Specification,
- searching a metadata index and downloading a data file, Server Based Data Sharing,
- using the data file on-line and off-line, Peer to Peer data sharing.

Each of the three use cases is described below, providing details of the problem addressed, how Consequence solves the problem and the use of context information in the evaluation.

2.4.1. Use Case 1: DSA Agreement Specification

This use cases addresses the process of creating a legally binding DSA, and establishing the technical mechanism to enforce policies which are incorporated in it.

The current method used in STFC to do this involves two main stages: managers and lawyers draft data sharing agreements, then technical staff examine the agreement to derive security policies that can be entered into existing data access control systems.

The use case breaks down into three stages which are addressed below in turn:

1. Authoring the DSA
2. Analysing the DSA
3. Mapping and deploying the DSA for enforcement

There can be an iteration between the first two of these stages to edit the DSA if the results of the analysis are unsatisfactory.

2.4.1.1. Authoring the DSA

The stage of authoring the DSA in turn breaks down into three stages, each of which is addressed below:

1. Creating the DSA vocabulary
2. Mapping the DSA vocabulary to the application vocabulary
3. DSA Policy Authoring using the defined vocabulary

Creating the DSA vocabulary

The vocabulary for the DSA must be created using the protégé tool V3.1.1 to produce an OWL V1 file.

It is unreasonable to expect a manager to precisely define a vocabulary within an ontology tool. If an ontology and vocabulary are created by a technical expert then they can be slightly modified if required by a technically able manager but this is the most which can be expected.

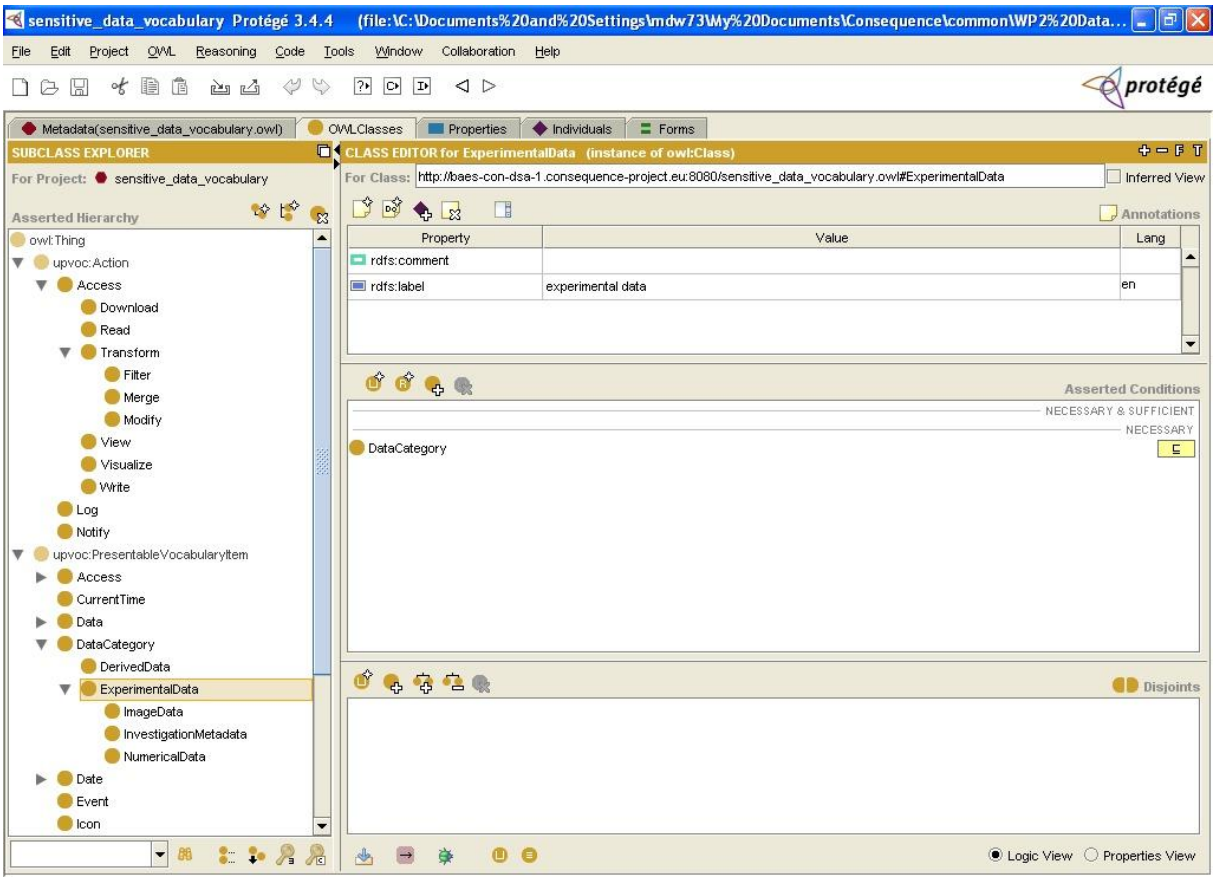

Figure 9: GUI to the Protégé ontology editor showing the object "experimental data" which is a type of "Data Category"

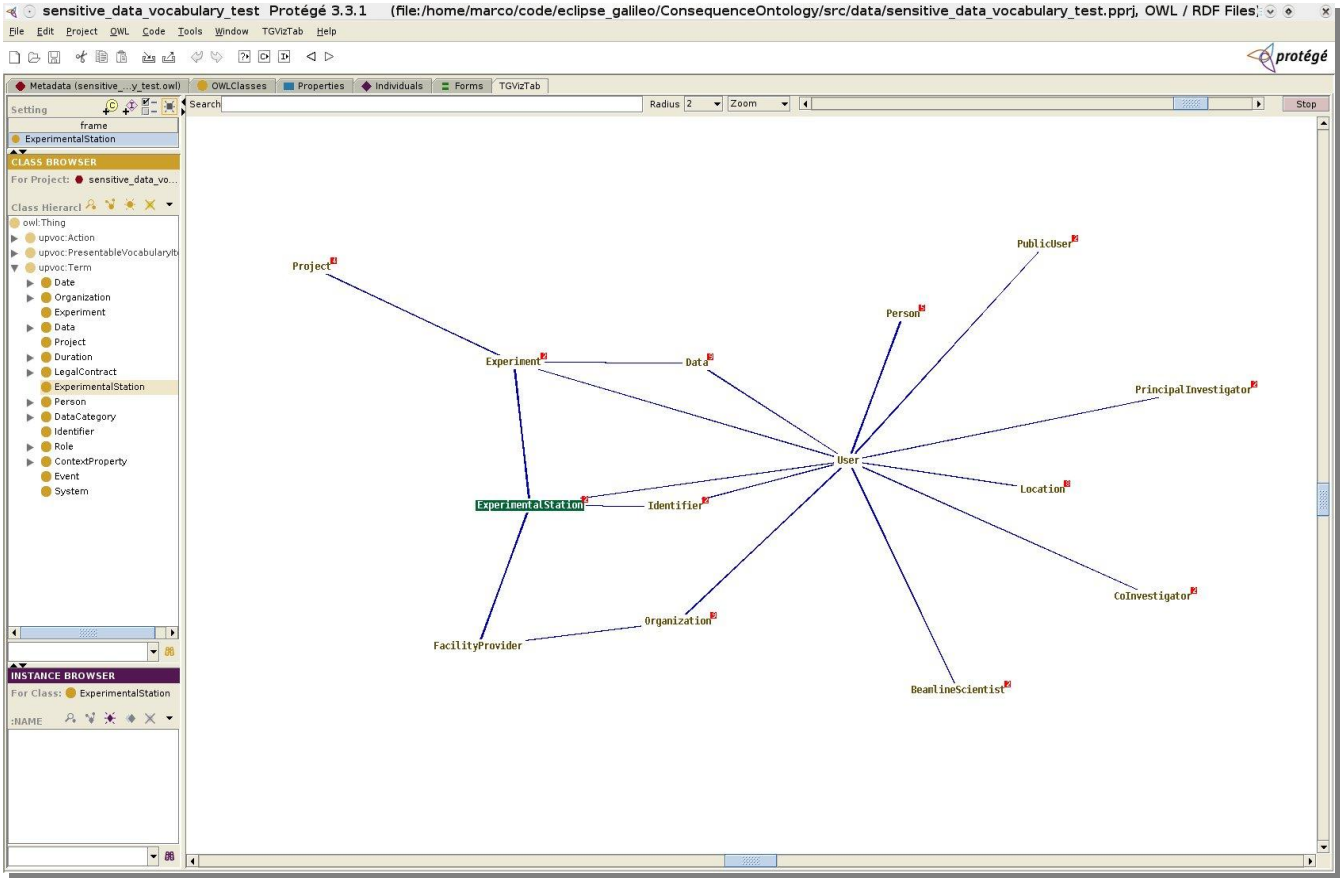

Figure 10: Example of a concept in the ontology in the Protégé tool: *Experimental Station* (in green in the figure) is hidden from the user vocabulary but is used to define terms which are visible which it is linked to in the figure.

The Protégé editor will allow the concepts in the domain to be defined, and a subset of those concepts will be made visible as vocabulary items which can be used in the DSA itself. Non-visible concepts are used to constrain the meaning of terms in the ontology.

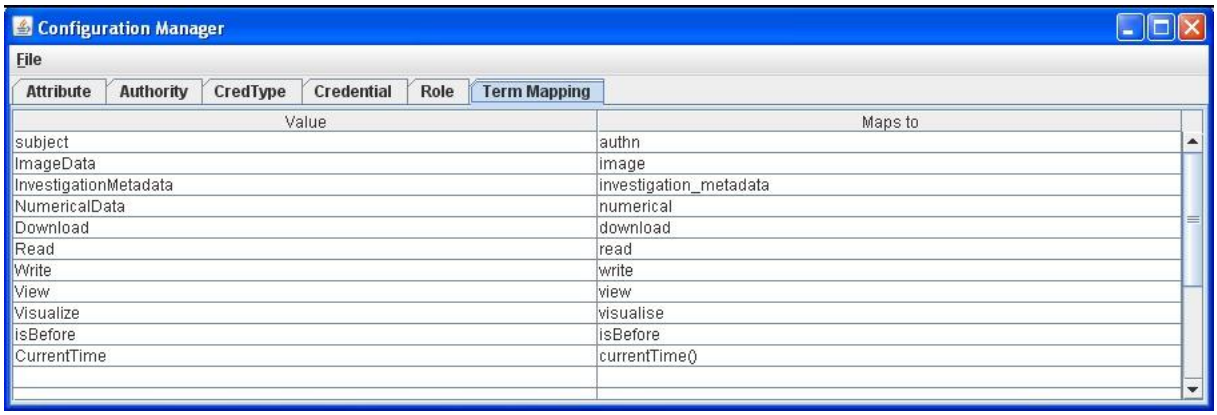

Figure 11: GUI to enter mappings from DSA vocabulary defined in the Protégé ontology to the application vocabulary defined in the metadata manager.

Once the domain ontology and DSA vocabulary are defined it is necessary to define the mapping of these terms to the actual terms used in the application. Since the application may be pre-existing, it is not reasonable to expect that it will use the terms in the DSA. A GUI is provided to support the entry of this mapping.

Authoring the DSA

Once the DSA vocabulary is defined it is possible to draft the DSA policies using that vocabulary.

To do this the DSA Lifecycle Manager is opened and *create new STFC DSA* is selected to open the DSA Authoring tool.

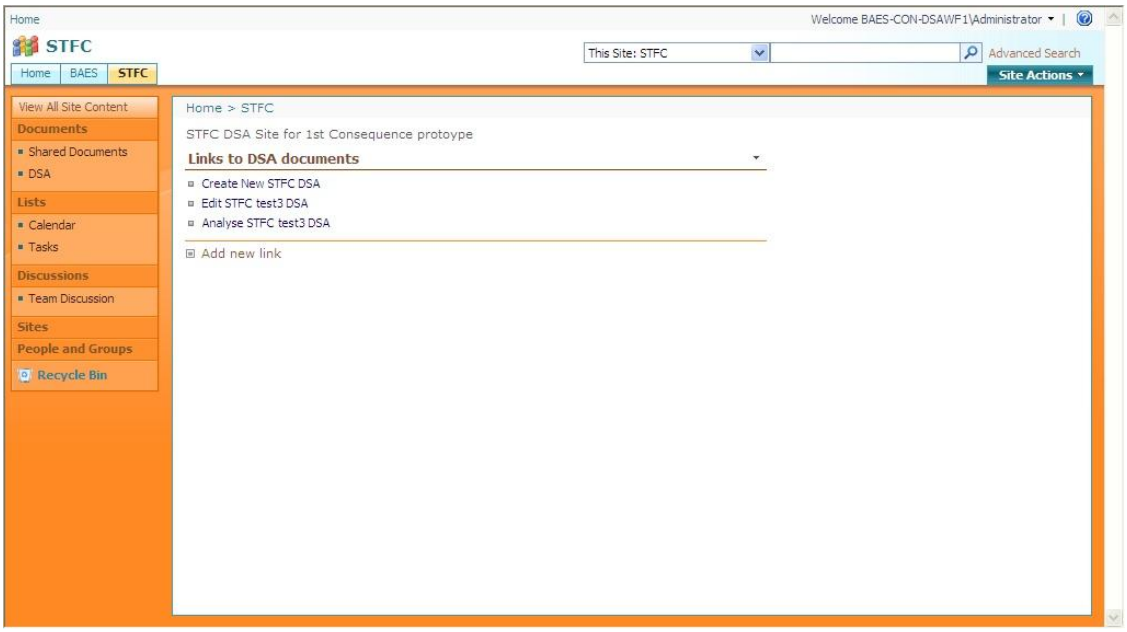

Figure 12: The DSA Lifecycle Manager

The DSA Authoring tool allows the user to enter declarations of the start and end date of the DSA, the parties etc.., and to draft the DSA policies as authorisations, obligations or prohibitions.

Figure 13: DSA Authoring tool highlighting references made in policies.

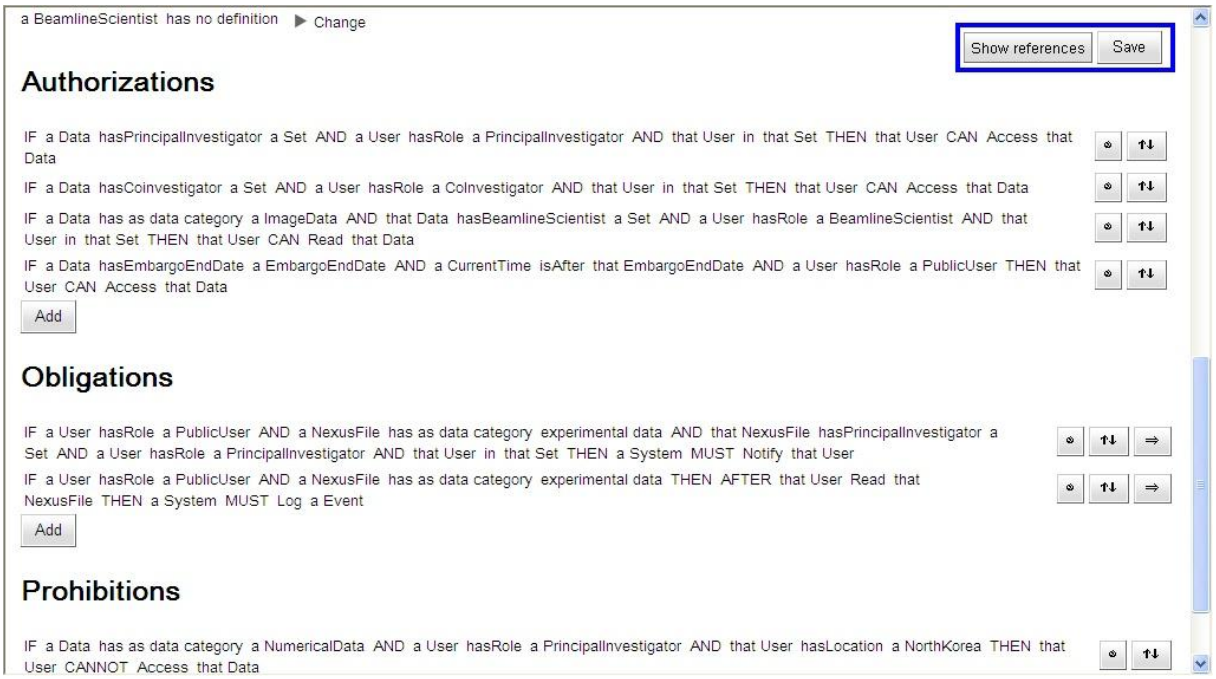

Figure 14: DSA Authoring tool showing authorisation, obligation and prohibition policies.

The use case is derived from the general real STFC operating procedures and policies. During the Consequence project STFC has developed a set of data policies which are included in Appendix 2. For the purpose of the use case a subset of these policies will be entered into a DSA authoring tool:

Authorisation Policies

1 Before the end of the three year embargo period, access to the experimental data is restricted to the principal investigator and co-investigators.

2 After the embargo period, the experimental data may be accessed by all users.

3 Beamline scientists can access image data produced on their experimental station.

Prohibition Policy

4 Access to numerical data should be denied to users in either Iran or North Korea.

Obligation Policies

5 After a public user downloads data then the system **must** notify the principal investigator.

6 After a public user reads any data then the system **must** log the event**.**

Usage Policies - prohibition

7 If a user does_not_use iCAT then that user cannot visualise experimental data.

2.4.1.2. Analysing the DSA

Once the DSA is drafted, it will be analysed to check for conflicts between policies and to determine the conditions under which access (that is read, write or download) are permitted.

The DSA Lifecycle Manager will launch the DSA Analyser with the DSA to be analysed as shown in the screen below. Then the user will set the context property values and query which the DSA should be analysed for, as shown in the figure below.

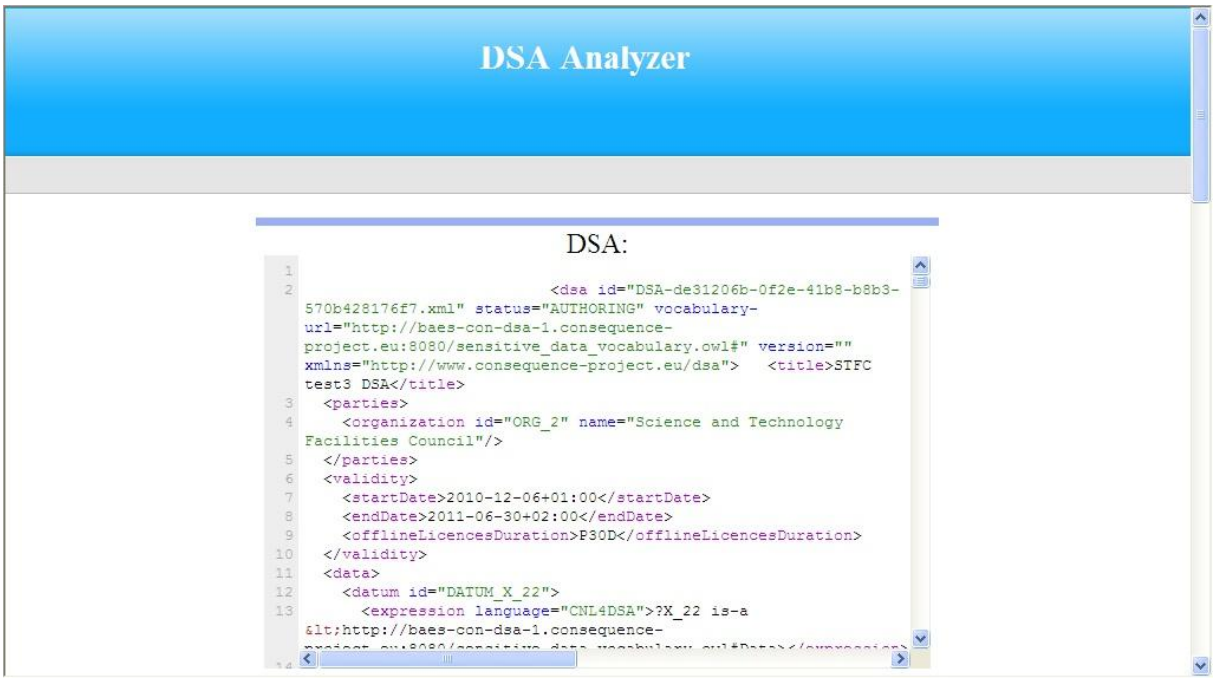

Figure 15: The XML version of the DSA in the analysis tool.

Figure 16: Setting the context properties and query in the analysis tool.

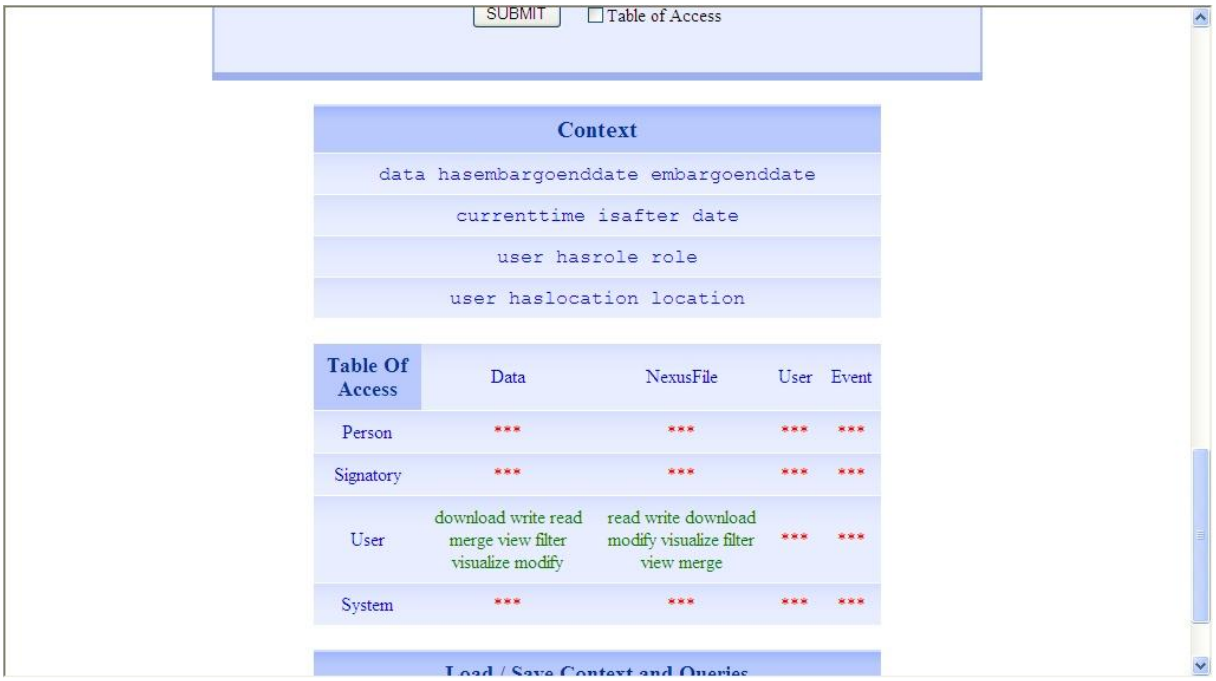

Figure 17: Results of the DSA analysis, showing the table of access.

If the analysis shows access can be achieved when it should not, or that access cannot be achieved when it should, then the author can revert to the DSA authoring tool and change the DSA policies.

Once the DSA authoring is complete the DSA Authoring Tool will save the DSA file in the DSA Lifecycle Manager ready for the next stage of the use case.

2.4.1.3. Mapping and deploying the DSA for enforcement

To perform the stages of mapping the DSA vocabulary to the application vocabulary and deploying the DSA for enforcement the user must select the workflows for the DSA in the DSA Lifecycle Manager, as shown in Figure 18 below.

Figure 19 shows the different workflows which are available. The first to be performed is the Authorisation of the DSA, which is a dummy operation since the distribution of the DSA, and the handling of XML signatures has not been implemented in the project. This was understood technology which will be required for an application, but which contained insufficient research interest to justify its development in the project.

The workflow which sends the DSA to the mapper should be called next as shown in Figure 20. The workflow calls one mapper per domain, since each domain could operate different applications which require their own terminology. This returns the DSA to the same place in the workflow tool DSA store, but with the addition of extra clauses in the XML file for the enforceable language version of the policies in EPL.

The workflow for deploying the DSA to the policy subsystem is also shown in Figure 19. When this is called the DSA is deployed.

The successful operation of these workflows is opaque to the user. The unsuccessful operation of these workflows will return error messages which are shown on the workflow status screen for each workflow. The usability of these error messages needs to be evaluated.

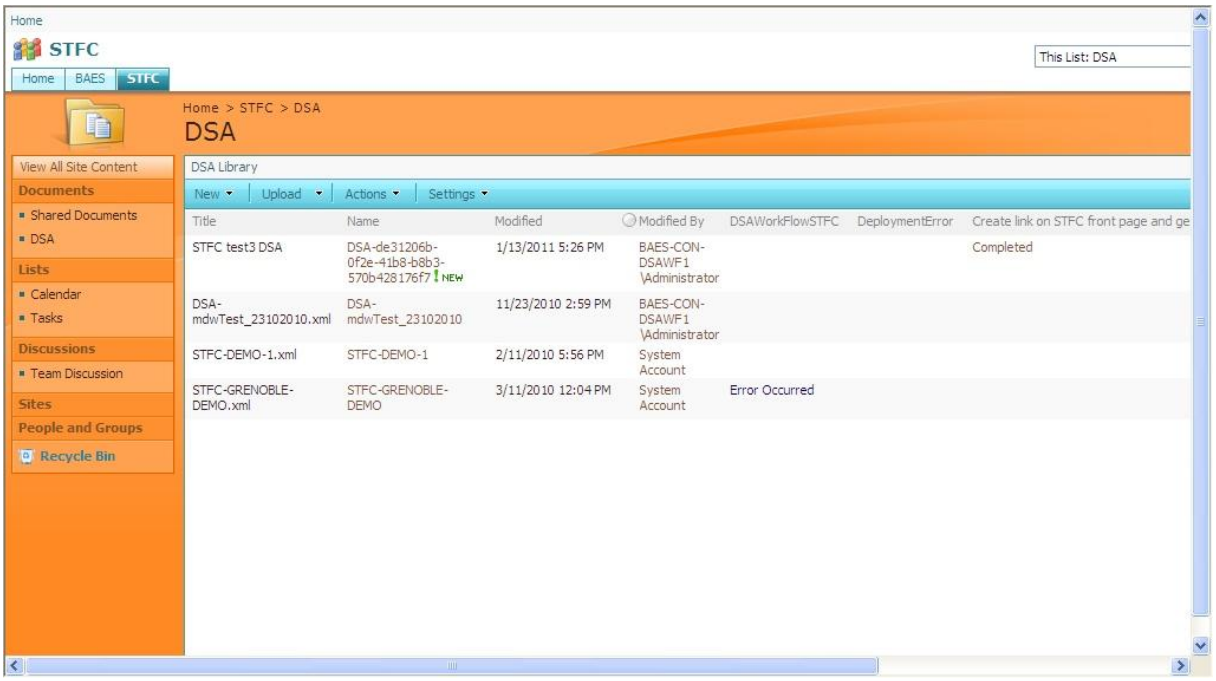

Figure 18: The authored DSA in the DSA Lifecycle Manager

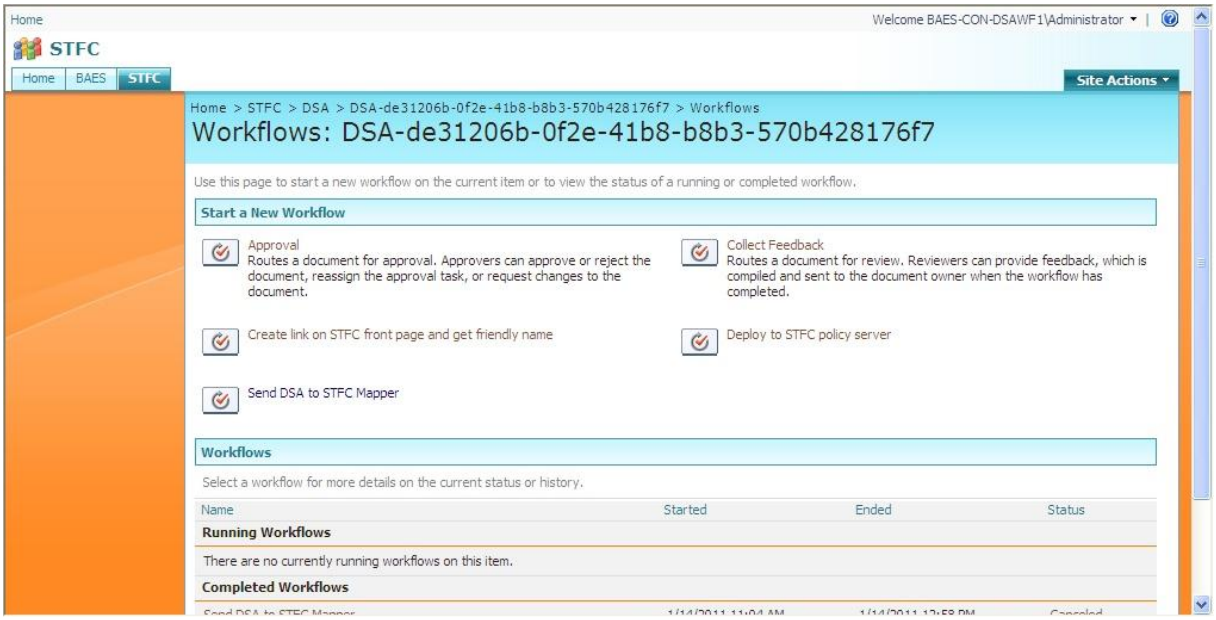

Figure 19: The workflows for the DSA

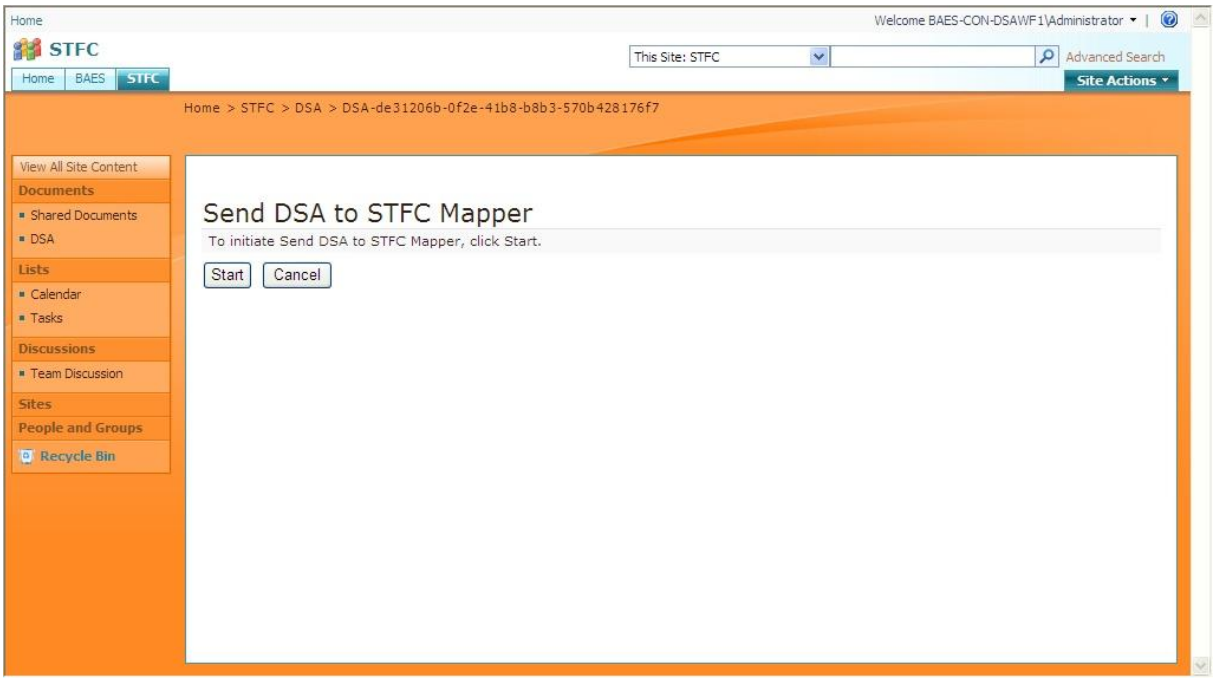

Figure 20: Operation of each workflow to send the DSA to the Mapper and to deploy it to the enforcement service.

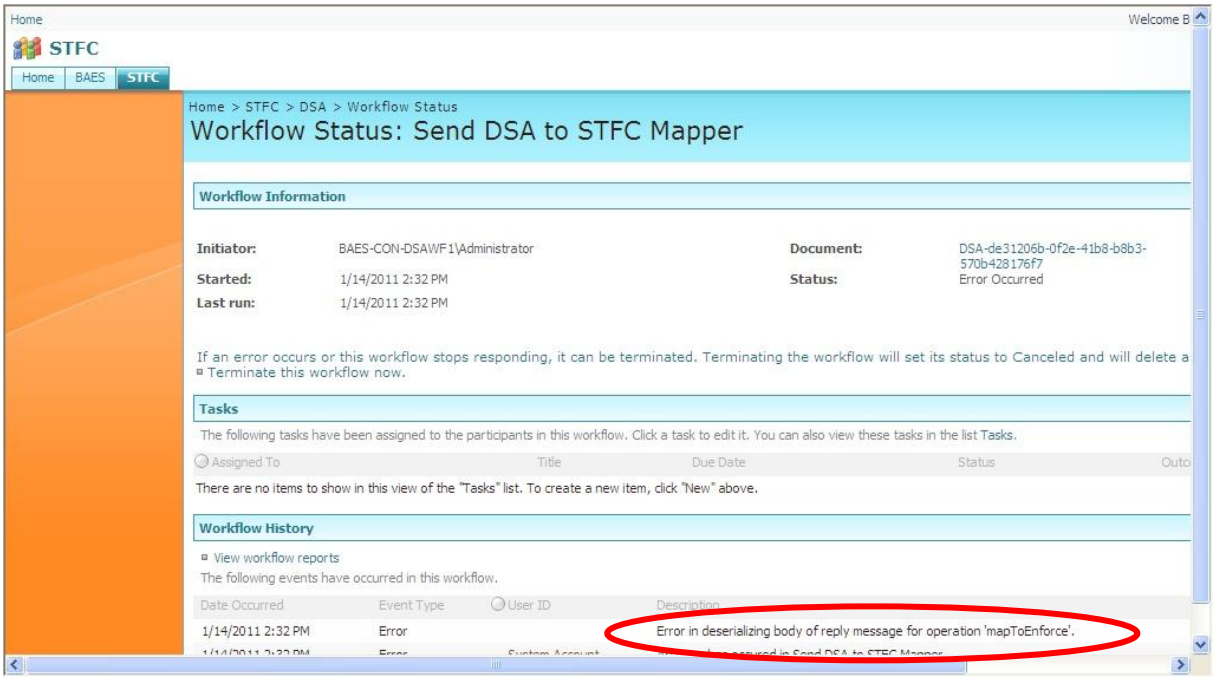

Figure 21: The report of the DSA Lifecycle Manager on the status of STFC Mapper workflow, reporting an error in the record.

2.4.2. Use Case 2: Server-based Data Sharing

2.4.2.1. Details of the Use Case

This use case is the data access problem which can be managed by existing technology if it is sensitive to the context variables required in the applicable data policies. The use case was described in D6.1 and is still consistent with that description so that the user can perform the following five steps.

1. A user will log into the iCAT client application.
2. The user will issue a query for data on a topic of interest.
3. The server application will provide encrypted metadata relevant to the query which is permitted by the data policies in the DSA.
4. The user will navigate through the metadata to identify data sets which meet their requirements for download.
5. The user will download the datasets which meet their requirements.

Screen images for the five steps are shown below.

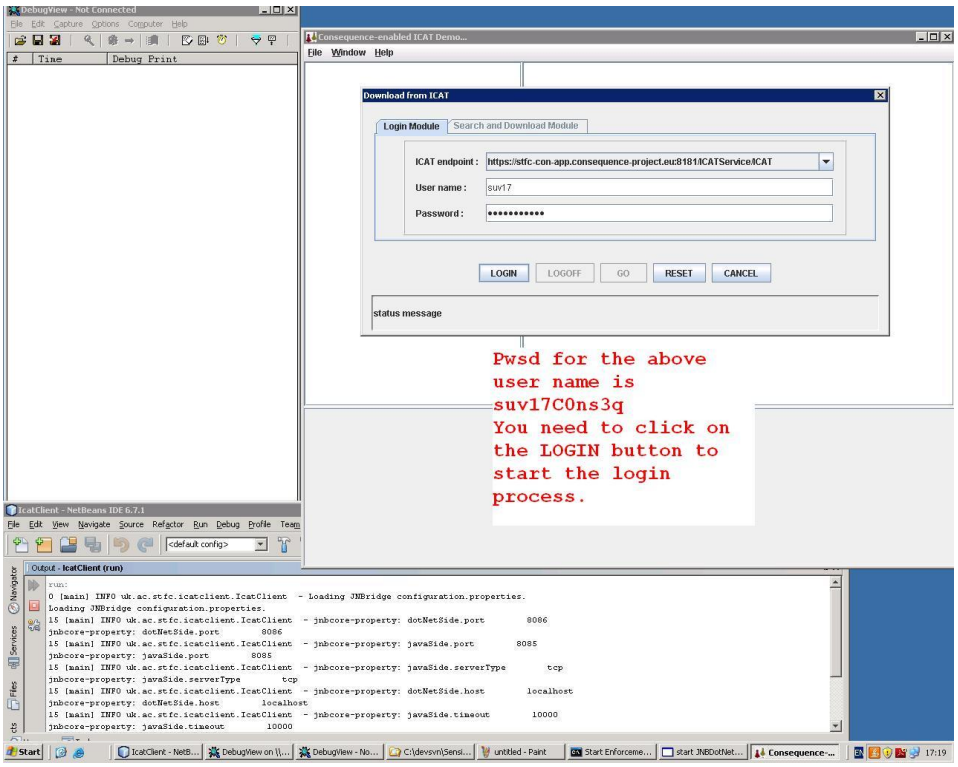

Figure 22: Screenshot of a user logging into the iCAT client application.

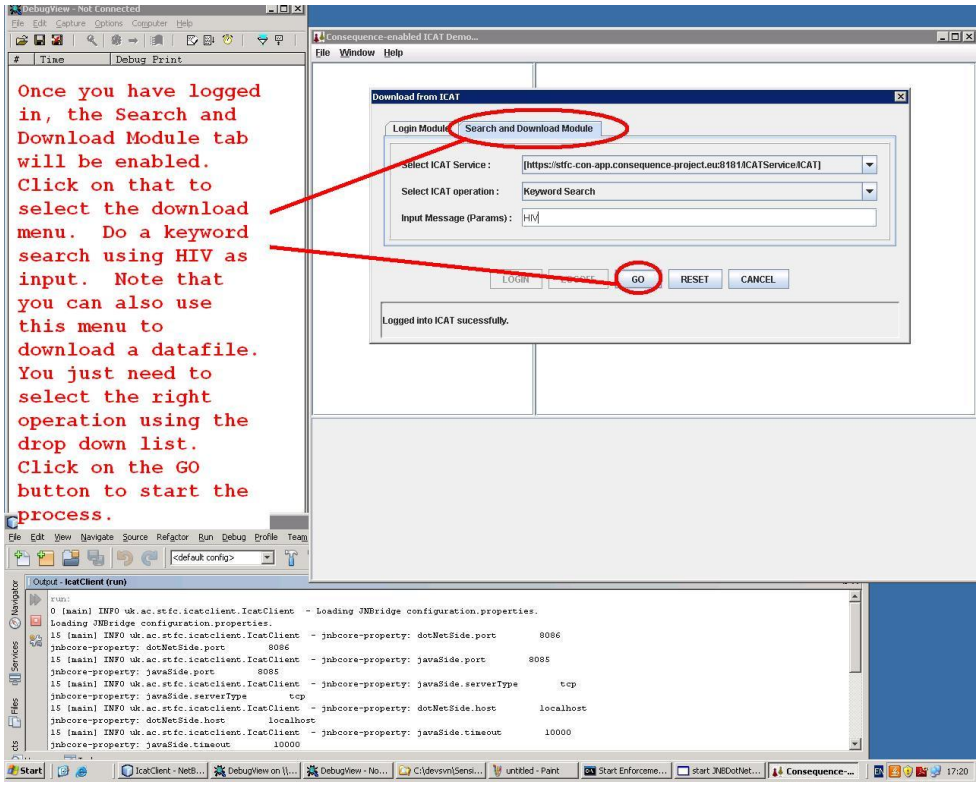

Figure 23: The user will issue a query for data on a topic of interest – HIV in this case.

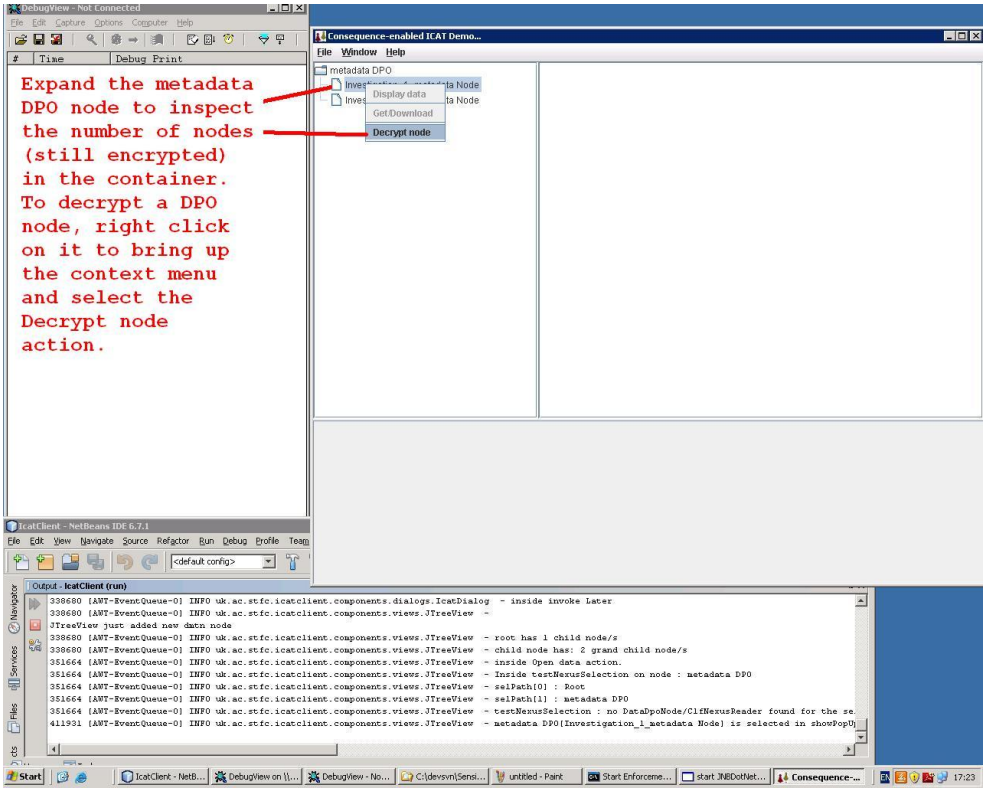

Figure 24: The server application will provide encrypted metadata relevant to the query which is permitted by the data policies in the DSA.

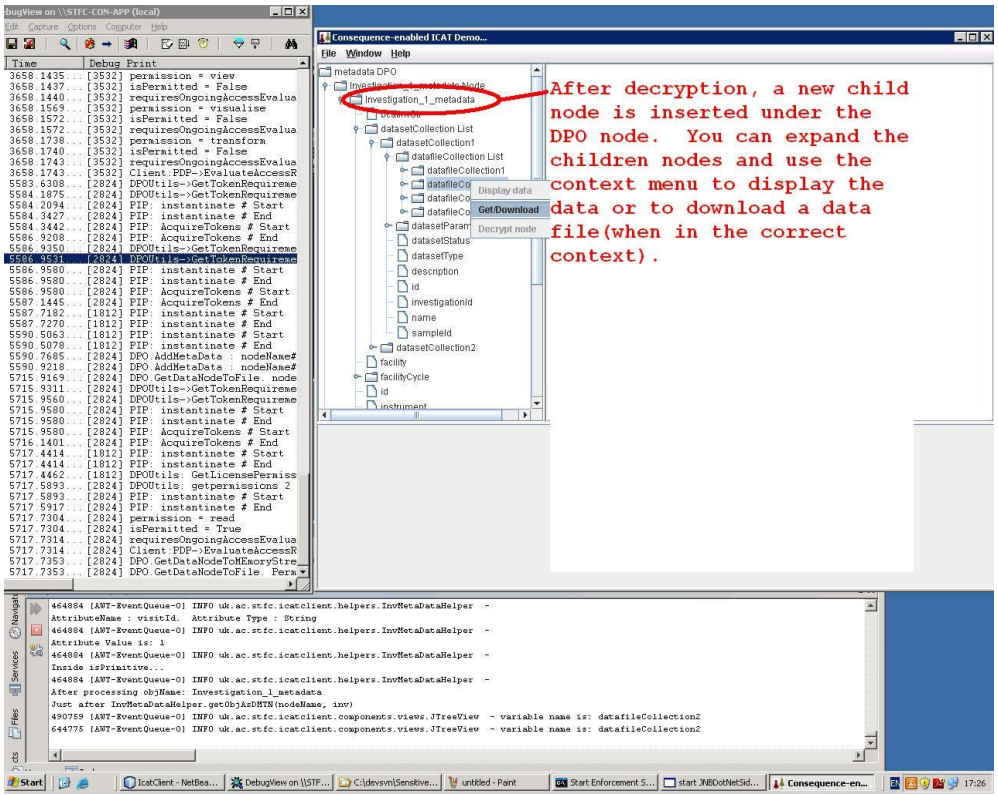

Figure 25: The user will navigate through the metadata to identify data sets which meet their requirements for download.

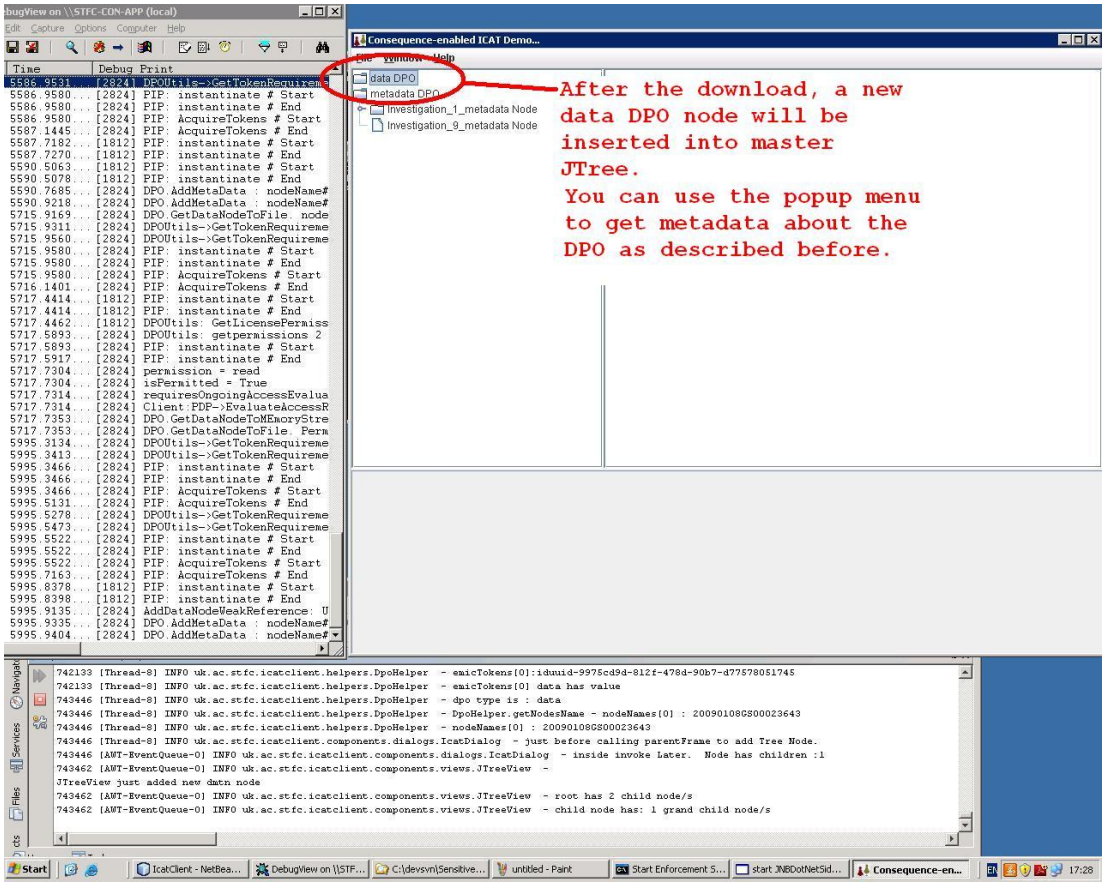

Figure 26: The user has downloaded the datasets which meet their requirements.

2.4.2.2. Contexts for the Use Case

Access to the metadata and data will depend on the following contexts:

- User identity
- User role in each investigation which has produced data
- Time of the query relative to the end of the embargo period

The application needs to be evaluated for various values of these context variables in order to test that the policies are being correctly applied to the data and metadata.

2.4.3. Use Case 3: Peer to Peer Data Sharing

2.4.3.1. Details of the Use Case

This use case is the conventional existing digital rights management problem with generalisations to the context variables used in the applicable data policies. The conventional approach is to issue a data object, usually a video or music file, with a license that only permits the presentation of the encrypted file on a particular machine, with a particular program, by a named user within the duration of the license. This approach is widely used in the music and broader media industries to protect copyright music and films.

There are two sub use cases for this one depending on whether the user is on-line or off line.

The use case is that:

- A user can browse the data file and display numerical data if the data policy permits.
- A user can browse the data file and display image data if the data policy permits.

Screen images of these two actions are shown below.

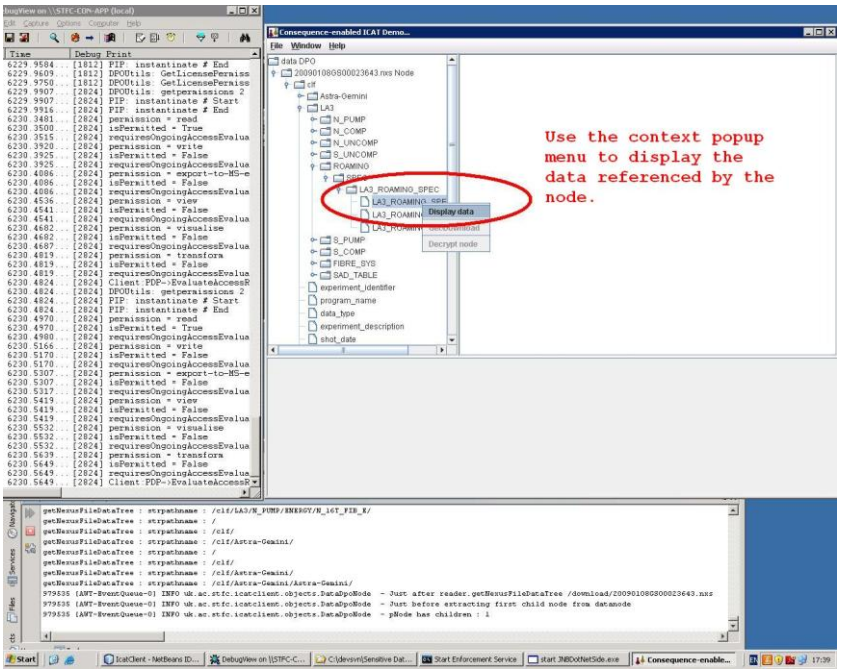

Figure 27: The user can navigate the internal structure of the data file to identify data items which they want to view.

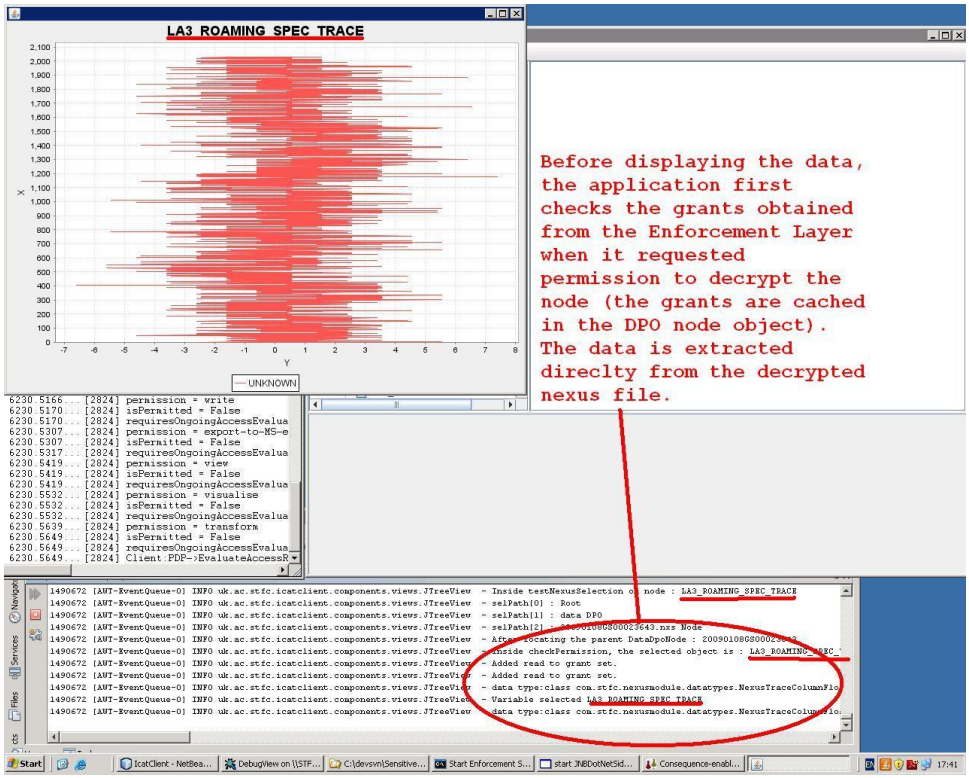

Figure 28: The user displays numerical data as a trace of the instrument detection.

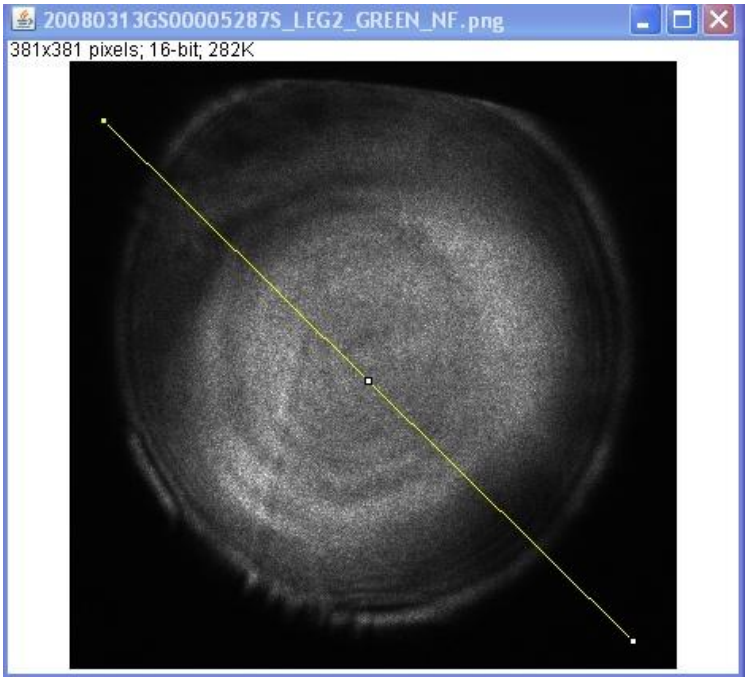

Figure 29: The user displays image data.

2.4.3.2. Contexts for the Use Case

There are four context variables which affect this use case for the test policies:

- On-line and off-line – duration of the off line license
- Location - country
- Time – before or after embargo period
- UserID and their Role with respect to the project which produced the data.

The application needs to be evaluated for various values of these context variables in order to test that the policies are being correctly applied to the data and metadata.

3. Prototype description

The overall Consequence architecture has been presented in deliverable D1.3 [3] as shown figuratively below. However, this is the architecture of the overall Consequence architecture showing the Consequence server and the Consequence client. Within the Consequence client are the application specific components shown in green.

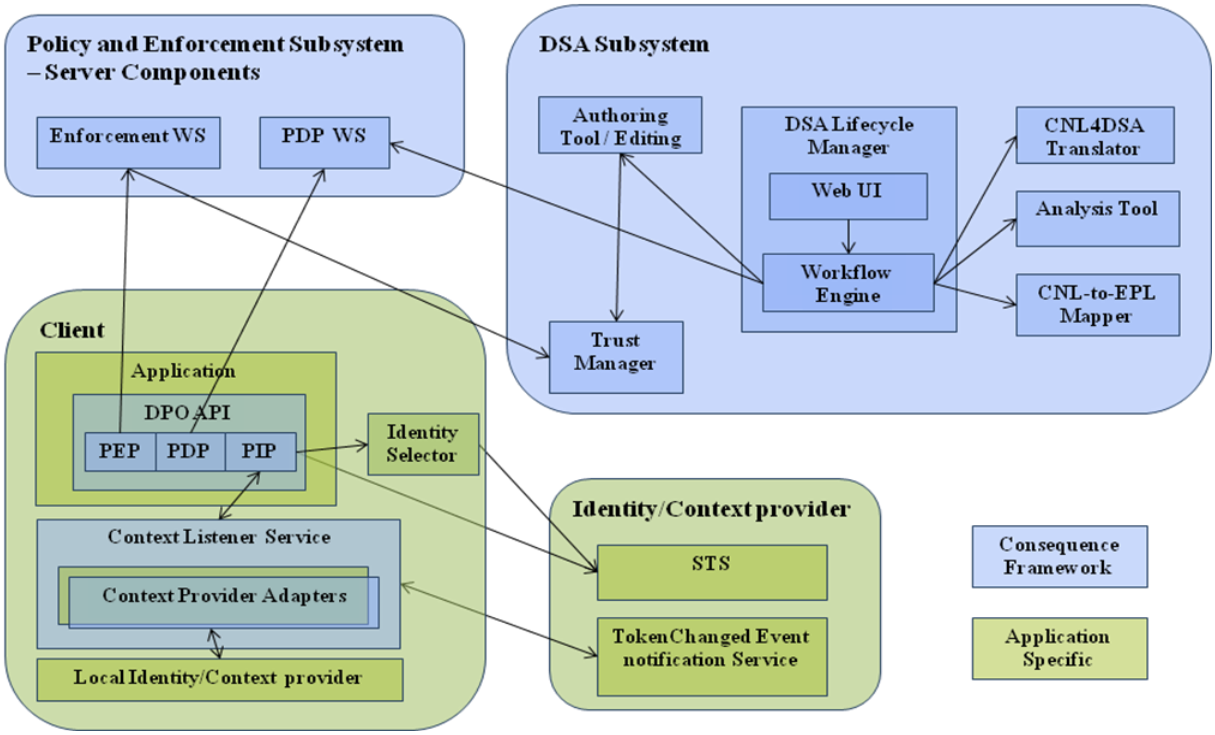

Figure 30: Consequence Architecture

The next section describes the architecture of the application used in this evaluation from the perspective of the application itself, to which the Consequence architecture provides interfaces. Following this description of the application architecture an analysis is described of the risks and threats which arise in this scenario from this application architecture.

3.1. Application Architecture

In Figure 30 the Consequence client can be broken down into the components shown in Figure 31 from the application perspective.

The main application includes both an application client (iCON) and an application server (iCAT). The server runs at STFC and is the only domain in the prototype application. iCAT is a data catalogue which stores scientific metadata about the scientific data, and not the data itself. The data itself is stored on a File Server, also on the server side. The users client runs the ICON client application which will download the metadata, the data files and will then view and analyse them. Two Secure Token Servers (STS) are used to provide the User ID (UID STS) and the user location using a GPS signal received by the user client machine (Location STS). The first four of these components (iCAT, iCON, File Server, UID STS) all

communicate with the Metadata Manger which presents the terms used in the application to the Consequence architecture. The Location Service does not need to do this since it only deals with the GPS location of the user client machine, and the country string to identify its location. These five components shown in green in Figure 31 constitute the application.

For the demonstration and evaluation of the prototype the application client is a Virtual Machine running in the Cloud called STFC-CON-CLI, while the server is a Virtual Machine called STFC-CON-APP. A third Virtual Machine STFC-CON-SRV runs the policy service of the Consequence architecture into which the DSA Lifecycle Manager deployed the DSA.

The interaction of the application with the Consequence architecture is entirely through the DPO-API, shown in Figures 32 and 33 below. The DPO-API is written in Microsoft's .NET while the application is written in Java. To communicate between these two, the intermediary product JNBridge has been used. Both the DPO-API and JNBridge should be evaluated.

Use case 1 requires the full Consequence architecture for DSA authoring, while use case 2 and the on-line sub-case for use case 3 require both the application client and server systems, while use case 3 second sub-case only requires the application client to evaluate the off-line performance of the application within the Consequence Framework.

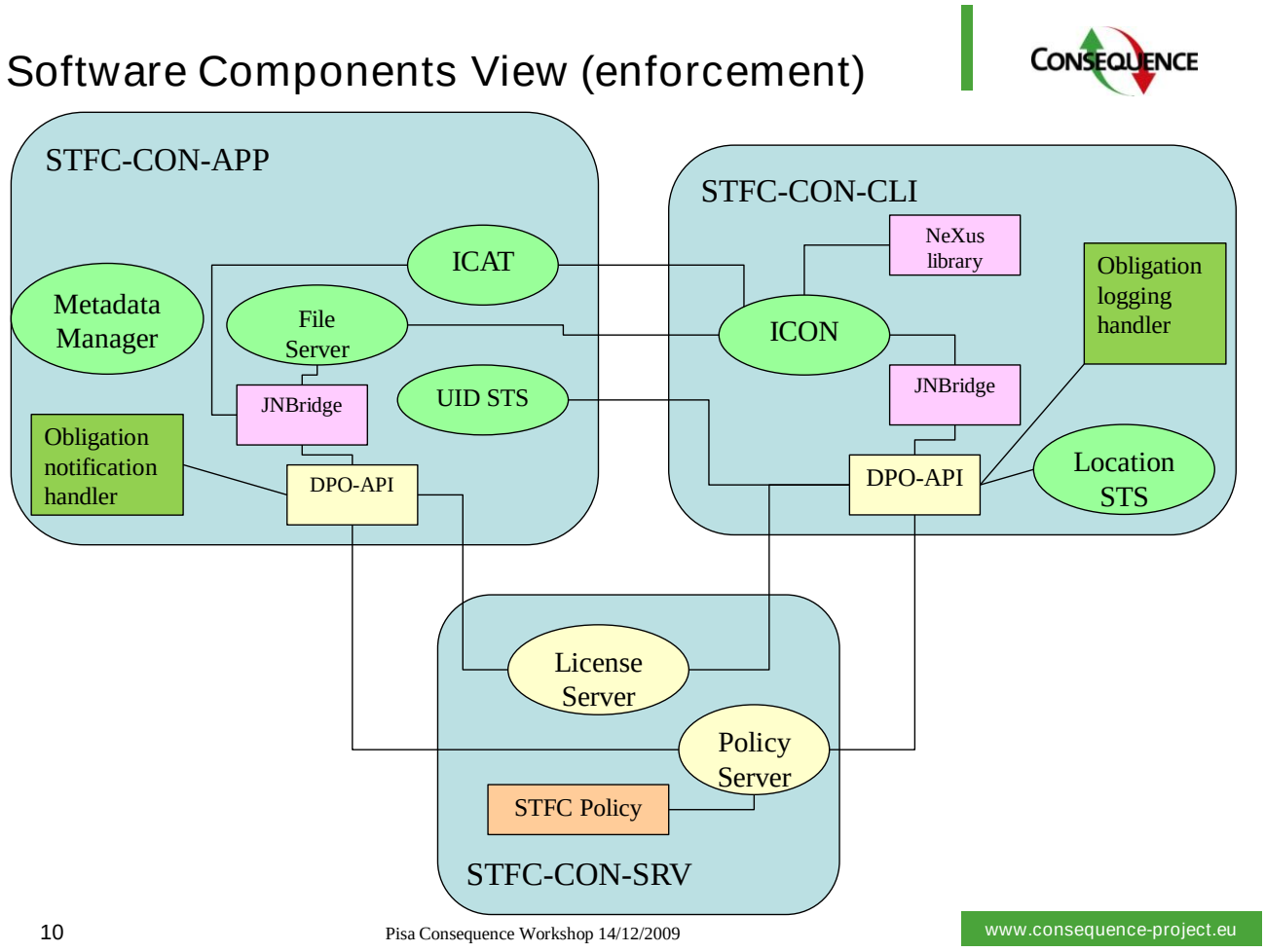

Figure 31: The architecture of the STFC Application

The final components in Figure 31 are the two obligation handlers. These are neither application specific, nor parts of the Consequence framework *per se*. They are required to

perform the obligations, and have been written for the prototypes, but can be used by both prototypes equally since they perform generic tasks. One, for logging events, resides on the client machine, while the other, for notification by e-mail, resides on the server machine.

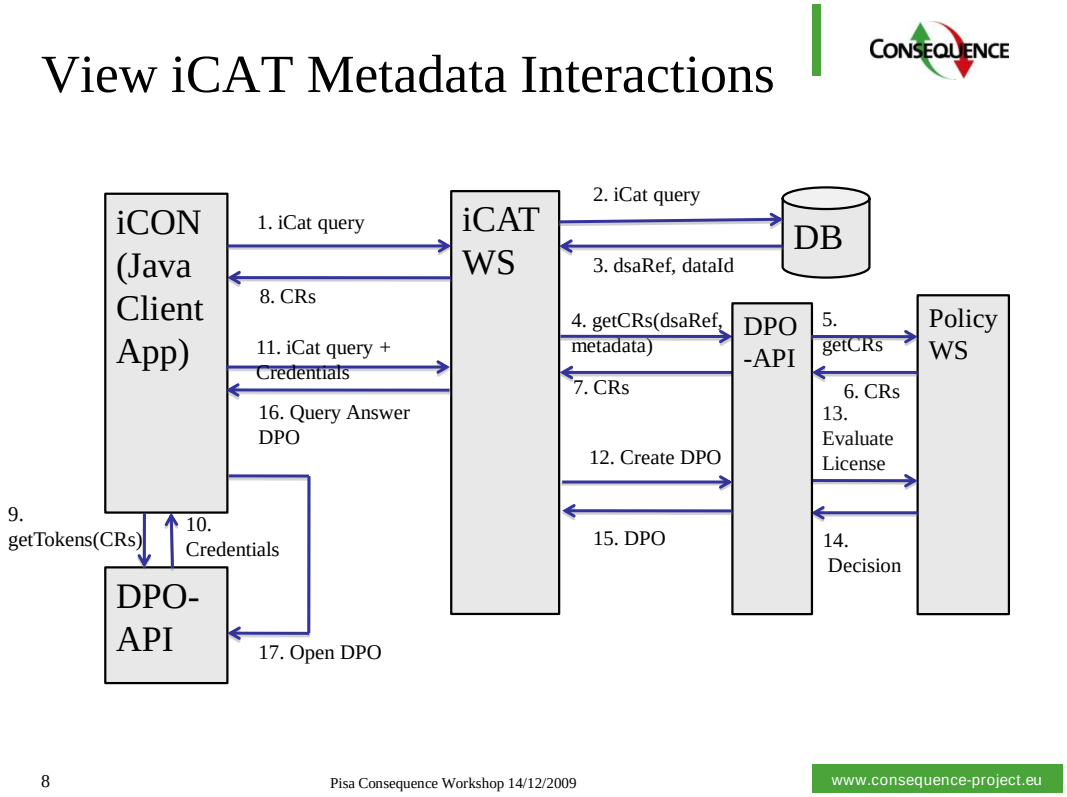

Figure 32: interactions between iCAT and the DPO-API to access scientific metadata.

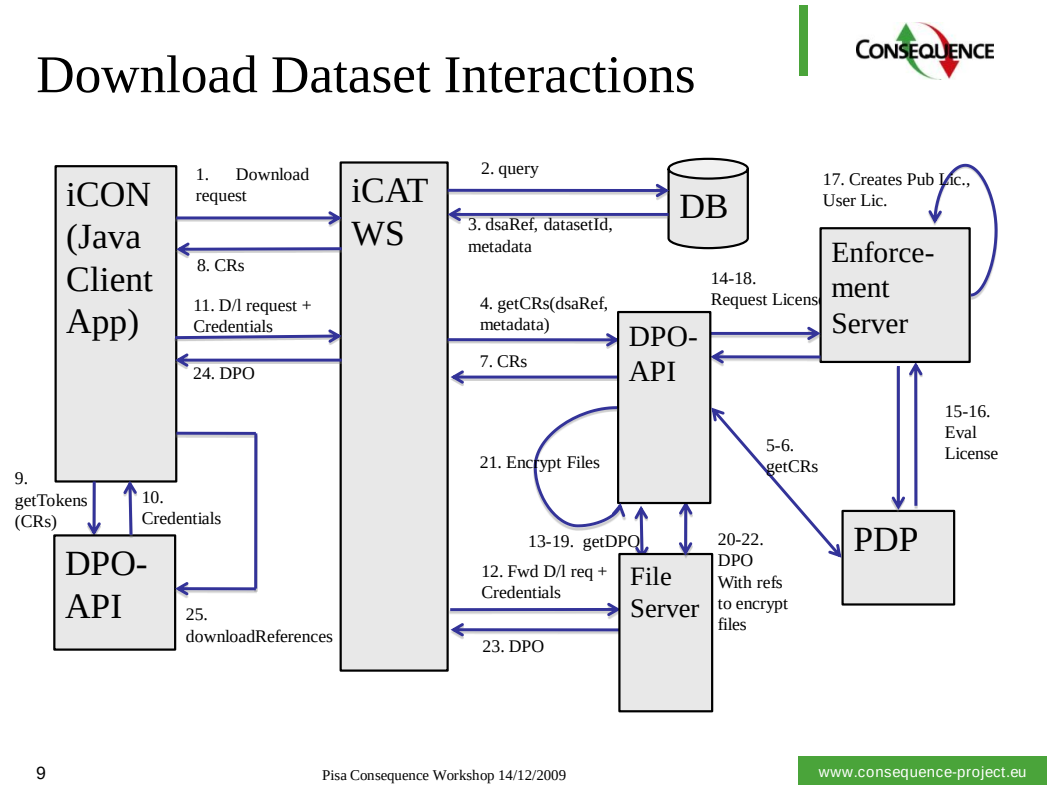

Figure 33: Interactions between the iCAT client and server to download datasets.

3.2. Risk assessment and threat analysis

The high level risks are shown in the table below. Threats arise from competitor science teams who will not use military levels of technology or effort to break into the system, although as the description in the introduction to this document illustrates, competing scientists can make efforts to unethically access and use scientific data. Threats also come from general hackers who see it as a challenge to break into large public systems such as those managing scientific data.

ID	Description
RI.1	The scientific data is released to people to whom it should not be.
RI.2	Not all stakeholders who should have access to data or metadata are granted access when they should be.
RI.3	Scientific metadata is released which allows competitors to determine that a particular chemical could have an impact on a technological problem – e.g. that a potential drug may impact on a specific disease.
RI.4	Data sharing agreements do not contain policies which address all realistic situations which arise for all stakeholders.
RI.5	The policy translation process enforces policies different from those stated in the DSA.
RI.6	The Consequence technologies do not record attacks on the Consequence infrastructure itself, or respond to attacks in such a way as to protect themselves against future attacks.
R1.7	The quality of metadata in the application maintained about user roles and data types is too poor to support accurate enforcement of policies which are based on these properties.

Table 1: Risk assessments

4. Evaluation process and results

4.1. Evaluation Criteria

This table below which was presented in D6.1 maps the use cases defined in section 3.4 to specific business, technical and administrative requirements. Highlighted items represent advanced (blue) and desirable (light green) requirements. Both categories have lower priority than the main requirements (not highlighted).

Requirement ID	Description	Use Case
BR1	A valid agreement must be in place before start of research.	Agreement Specification
BR2	A single party may have multiple agreements with different parties on the same dataset.	Agreement Specification; Server-based Data Sharing Mini Use Case 1
BR15	The formal data sharing policies applicable to a particular dataset must be unambiguously resolved between the different agreements with different end parties.	Agreement Specification;
BR3;	An agreement should include references to relevant external agreement/s and legislation.	Agreement Specification
AR5	The current version as at the agreement date will be used.	Agreement Specification
BR4	Template agreement shall be used.	Agreement Specification
BR5	The controlled vocabularies for building DSA clauses must be understandable to human users.	Agreement Specification
BR6	The controlled vocabularies must be suitable for expressing policy for analysis and refinement purposes.	Agreement Specification
TR6	The solution should include a mechanism for analyzing and resolving formal policy conflicts.	Agreement Specification;
BR14	The mechanism referred to in TR6 should provide human readable error messages to facilitate policy analysis and refinement.	Agreement Specification;
BR7	Formal language DSA may use identity or attribute-based description.	Agreement Specification
BR8	Formal language DSA should clearly define conditions of use and obligations for target dataset/s.	Agreement Specification
BR9; TR7	Capability to resolve the provenance of a formal policy back to the high level DSA that gives rise to it.	Agreement Specification;
BR10	Capability to propagate data sharing policies from parent to derived dataset/s.	Agreement Specification; Peer-to-Peer Data Sharing Mini Use Case 1
BR11, TR7	Capability to trace the origin of inherited policies in a derived dataset back to the parent dataset.	Agreement Specification; Peer-to-Peer Data Sharing Mini Use Case 1
BR12 BR13	An agreement should contain precise instructions on: - the procedure to obtain consent if this is a policy condition; or - obtaining intelligence about external events that trigger changes in policy states.	Agreement Specification;
BR16	An agreement may be updated by the owner if the resultant formal policy does not conflict with other formal policies in force on the same dataset. The agreement end parties will be informed and their consent sought.	Agreement Specification;
TR11	An audit trail available to support resolution of conflicts	Agreement Specification;

	and liability.	Off-line Data Sharing Use Case
BR17	An agreement must state the retention period and disposal process of the resultant audit trails.	Agreement Specification;
BR21;BR22; AR2; AR3; AR4	Each organization to establish clear organizational and reporting structure, guidelines and procedures for the administration and support of: - its portfolio of agreements and formal data sharing policies; - the related audit logs; - the technical infrastructure.	Agreement Specification;
BR23	The proposed framework should not bring about a detriment of the RCUK goal to achieve open access to publicly funded research data.	Server-based Data Sharing Mini Use Case 2
BR24	Demonstrate cost-benefit effectiveness to community data providers.	Server-based Data Sharing Mini Use Case 2
TR1; TR2 TR5	The proposed architecture: - must be compatible with STFC data management framework including the Data Portal, ICAT3 and CSMD; - should provide well-defined interface to facilitate integration with existing application architecture.	Server-based Data Sharing Mini Use Cases 1,2,3
BR18	The researchers will retain administrator rights to machines under their management.	
TR3	The proposed architecture should be platform independent.	Peer-to-Peer Data Sharing Mini Use Case 1
TR4	The proposed architecture could interoperate with a variety of software platforms.	Server-based Data Sharing mini use cases 1,2,3
TR8	The solution should support the notion of session.	Peer-to-Peer Data Sharing Mini Use Case 1
TR9	The policies are enforceable over varying types and volumes of scientific data.	
TR11	The data sharing policies are enforceable not just over data held centrally by the facility ICAT, but on disseminated data analysed in 3rd party locations.	Server-based Data Sharing Mini Use Case 1 Peer-to-Peer Data Sharing Mini Use Case 2
TR10	The policy infrastructure could actively monitor on-going environmental parameters to ensure the correct enforcement of context-aware data sharing policies.	Peer-to-Peer Data Sharing Mini Use Case 2
AR1	A data file will be the smallest unit of data objects being shared. This does not apply to metadata.	
TR13	The solution could support both manual and automatic deployment of low level policies.	Agreement Specification;
TR15	It is desirable for the solution to support secure data sharing in an environment where network connection is not always available.	Off-line Data Sharing Use Case
TR14	The proposed solution may include an efficient and reliable mechanism to obtain consent if this is an access condition.	Server-based Data Sharing Mini Use Case 1
BR19	Provision of software libraries or plug-ins to promote the development of Consequence-aware scientific applications.	Peer-to-Peer Data Sharing Mini Use Case 1
BR20	Capability for the framework to support different application levels.	Server-based Data Sharing Mini Use Case 2

Table 2: Application prototype requirements from D6.1

The evaluation criteria are whether the system meets the requirements above concerning the prototype, and whether it is usable by the target stakeholders listed in section 3.3, and addresses the risks identified in section 4.2.

4.2. Evaluation Process

The overall evaluation process was to ask stakeholders to work through the use cases which related to them.

An ambitious aim of the project was to combine the individual technologies incorporated in the Consequence infrastructure from a state of being reported individually in the research literature to de demonstrated working together in the project prototypes, with the prospect of being able to be incorporated by non-expert developers into demonstration projects.

4.2.1. Evaluation of the authoring environment

Two managers were asked to enter pre-written DSA policies in the authoring environment, and to draft new policies.

4.2.2. Evaluation of the controlled natural language for policies

The controlled natural language (CNL) is designed to address not the technical problem of expressing policies that can be enforced, but the human problem of managers with little technical knowledge being able to draft enforceable policies. The enforceable policy language (EPL) used by the policy decision point can express the policies, but it is not intended to be user friendly, merely efficient to enforce. Therefore to evaluate whether the CNL does its job, it is necessary to evaluate whether the CNL is more usable by the target audience than the EPL is.

The evaluation was performed by using the questionnaire shown in Annex 1. This asks subjects to give a judgement as to whether the CNL is more like natural language than the EPL.

The target audience for the CNL consists of corporate managers and IT professionals. The questionnaire was answered by six individuals in each group to allow a comparison of whether the CNL is better than natural language for both groups, neither or both, and whether there is a difference between the two groups with respect to the CNL.

4.2.1. Evaluation of the policy analysis

The policy analysis tool was presented to two managers at STFC to analyse the pre-written DSA. Before using the tool the subjects were presented with a short video showing how the analysis tool operates to provide some context for this unusual tool.

4.2.2. Evaluation of the deployment and enforcement

The DSA for the STFC scenario was deployed and enforced. When these operate successfully there is no user interaction to evaluate. When they fail error messages can be evaluated for their usability.

4.2.3. Evaluation of the application using the Consequence framework

Two users who were not involved in the project, but who were knowledgeable about the application were asked to use the iCAT application to download and view data from studies which they were investigators on.

Users logged in as different roles in projects to check the variations in the context variables.

It was not possible to move between countries to test the location context variables using the GPS based service shown in Figure 34, but an emulator was added to the location service to test this variable as shown in the Figure 35 below.

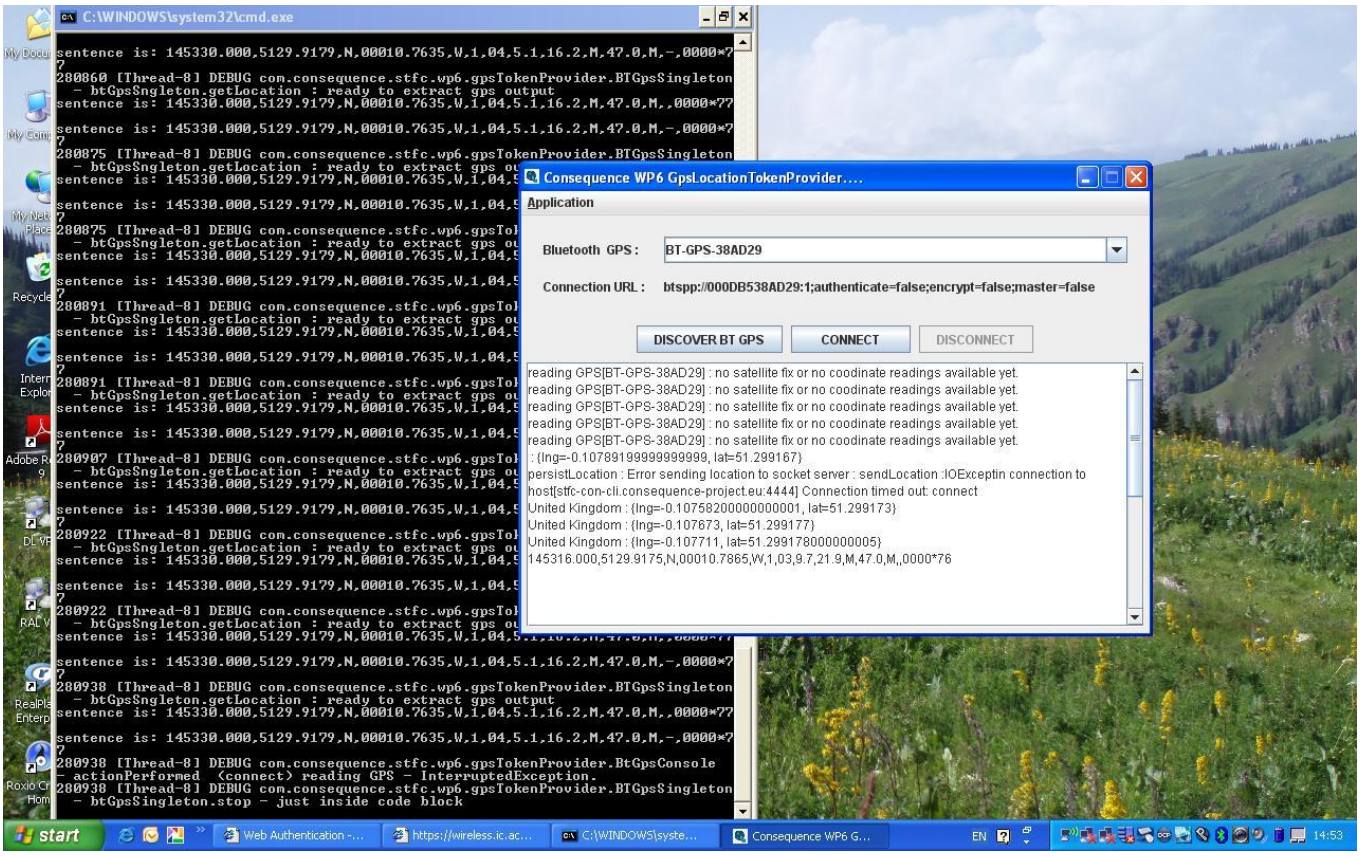

Figure 34: GPS based location service showing real data.

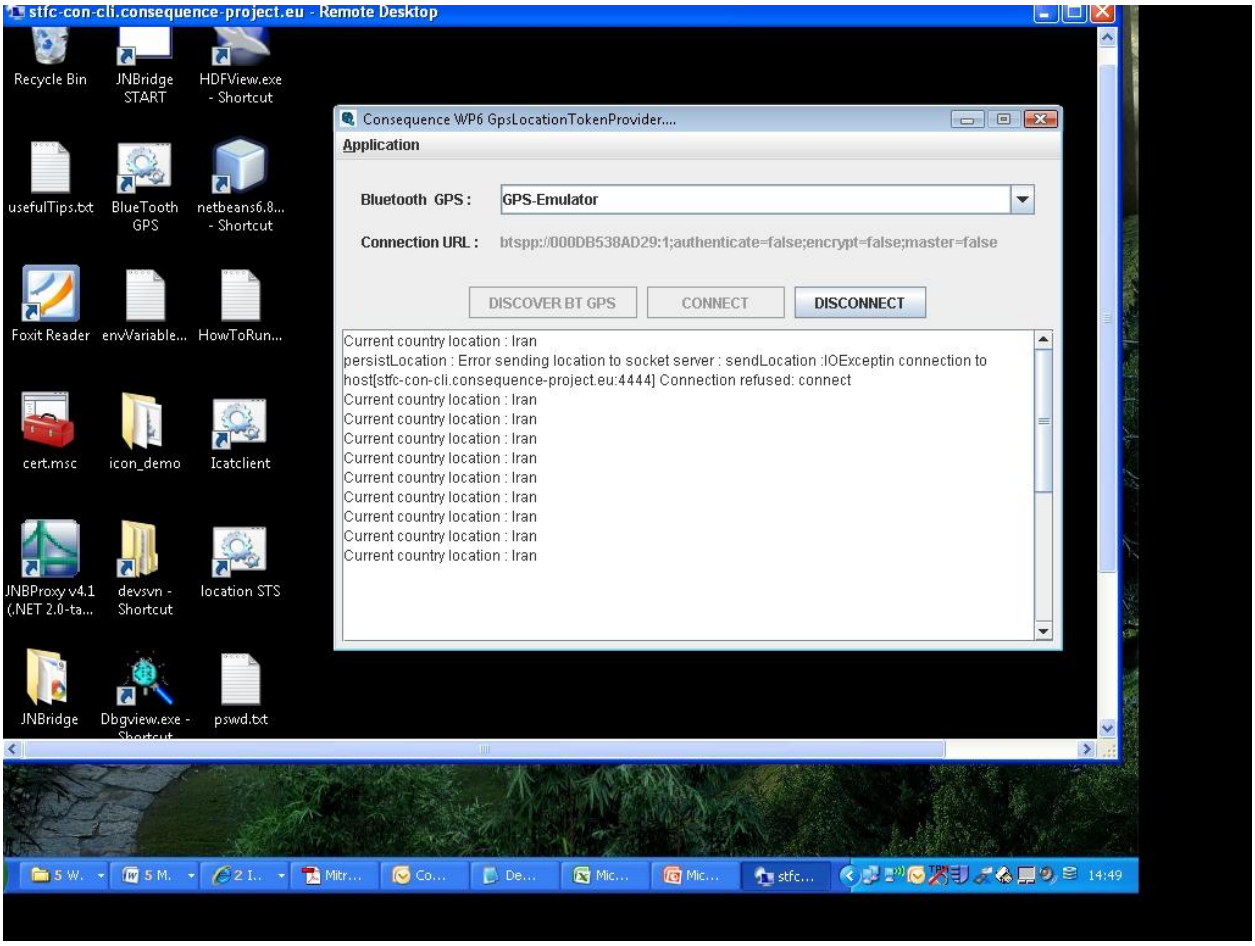

Figure 35: GPS emulator to set the location as countries where access is prohibited for evaluation.

4.3. Evaluation Results

4.3.1. Evaluation of the authoring environment

Overall, the different tools in the authoring environment conform to different "look & feel" and do not implement a common user model. The implementation being evaluated is a prototype, but in order to both designate it as a set of Consequence tools, and to increase their usability a common look and feel should be imposed on any industrialised version of the prototype.

Ontology Editor – Protégé

The use of an existing tool for this was a positive choice since there were staff available who already had knowledge of the tool who could easily edit simple ontologies of the form required for the Consequence DSA.

The Protégé tool is sufficiently complex that it is not approachable by senior managers who must rely upon ontology experts to provide appropriate vocabularies for the planned DSA.

There are some quirks in the ontology design which must be accepted by the user, for example the required distinction between *types* of *Person* and *roles* of *User* which is a sub-type of *Person*. A simple analysis would simply define the roles as sub-types of user. Policies differentiate access and usage on the basis of *roles* of *User*.

Unfortunately, the version of protégé used is an old one which produced OWL V1, while active users have moved on to V4.3 which produces OWL V2. Any further development of the ontology based authoring tool should upgrade to the latest version of both Protégé and OWL in order to maximise the transfer of skills from existing users.

CNL to EPL Mapper tool

The only documentation available for the mapper is that in the public deliverables which is insufficient to be able to use it reliably. The mapper GUI (see Figure 11) allows entry of terms in the CNL to map them to EPL. It has six tabs, one of which permits direct mapping of terms from CNL to EPL which can easily be used to enter mappings. The other five tabs also support the mapping of different classes of information from CNL to EPL but the lack of documentation, and examples makes their use unclear.

When the mapper runs some error messages seen in the DSA Lifecycle Manager report (see Figure 21) were impossible to interpret without a detailed knowledge of the implementation. It is unclear how to proceed from this point without advice from the author of the tool or improved detailed documentation.

When the mapper succeeds errors are often produced at deployment as a result of errors in the mapping. For example:

> Authz[AUTHORIZATION_26_read]: Unknown role: CoInvestigator

Which states that a role is unknown to the policy service, but which arose from a typing error by the user in the mapper GUI when entering “CoInvestigator” in the EPL instead of “Coinvestigator“ with a lower case “i”.

This form of error reporting is enough to identify such errors and correct them.

DSA Policy Authoring Tool

The policy authoring tool was used to enter a pre-written DSA without any severe problems by two middle managers with software experience.

The policy authoring tool uses the controlled natural language which does not permit the entry of real English. The policies presented in section 3.4.1.1 in English need to be entered in the form shown below. Managers had problems looking at the ontology and determining how to write the English versions of policies in the controlled natural language. Lawyers and contract staff would not accept controlled natural language as permissible in contracts themselves, although both managers and technical staff considered the language usable to represent policies. The consequences of the constraints imposed by this controlled English on usability are evaluated in general in the next section.

Authorisations

> IF a NexusFile has as data category experimental data
> AND that Nexus file hasPrincipalInvestigator a Set
> AND a User hasRole a PrincipalInvestigator
> AND that User in that Set
> THEN that User CAN Access that Nexus file
>
> IF a NexusFile has as data category experimental data
> AND that Nexus file hasCoinvestigator a Set
> AND a User hasRole a CoInvestigator
> AND that User in that Set
> THEN that User CAN Access that Nexus file
>
> IF a Nexus file has as data category a ImageData

AND a User hasRole a BeamlineScientist
AND that Nexus file hasBeamlineScientist a Set
AND that User in that Set
THEN that User CAN Access that Nexus file

IF a NexusFile has as data category experimental data
AND that NexusFile hasEmbargoEndDate a EmbargoEndDate
AND a CurrentTime isAfter that EmbargoEndDate
AND a User hasRole a PublicUser
THEN that User CAN download that Data

IF a NexusFile has as data category experimental data
AND that NexusFile hasEmbargoEndDate a EmbargoEndDate
AND a CurrentTime isAfter that EmbargoEndDate
AND a User hasRole a PublicUser
THEN that User CAN Access that Data

Obligations

IF a User hasRole a PublicUser
AND a Nexus file has as data category experimental data
AND that Nexus file hasPrincipalInvestigator a Set
AND a User hasRole a PrincipalInvestigator
AND that User in that Set
THEN AFTER that User downloads that Nexus file
THEN a System MUST Notify that User

IF a User hasRole a PublicUser
AND a NexusFile has as data category experimental data
THEN AFTER that User Read that NexusFile
THEN a System MUST Log a Event

Prohibitions

IF a NexusFile has as data category a NumericalData
AND a User hasRole a PrincipalInvestigator
AND that User hasLocation a NorthKorea
THEN that User CANNOT Access that NexusFile

IF a NexusFile has as data category a NumericalData
AND a User hasRole a PrincipalInvestigator
AND that User hasLocation a Iran
THEN that User CANNOT Access that NexusFile

When policies refer to two different users (as in the first obligation policy above) both users are referred to as "*that user*" which initially confused the subject authors, although the use of the reference highlighting feature does differentiate them.

When policies are being written and the user makes a mistake the tool does not support the deletion of the previous word entered. In this case the policy has to be continued so that it makes no sense and the system will respond with a message that: "*No further choices are available in the current vocabulary! It looks like the statement being composed has no well defined meaning and so it will be discarded*." After which the tool deletes the policy being created. When the user is aware of the error they have made the tool would be more usable if a delete function were available to allow the developing policy to be corrected.

The authoring tool is designed to present the user with a menu of the vocabulary terms which can be inserted next in the policy being constructed. While rules following the established structure of those shown above are written, policies can be created with the authoring tool – e.g. varying the data types, user roles in the policies. When writing policies with a novel grammatical form, the user frequently inserted a sequence of presented terms only to be shown the error message in the previous paragraph. The user must infer the structure of

permissible policies which have a well defined meaning as opposed to those which they can construct which the tool decides do not. Further guidance is required for users to make this decision than is currently available in order for the tool to be usable to non-specialists to create policies – either through detailed, but comprehensible, documentation, or through enhancements to the tool's grammar.

Constraints are also made on the composition of rules by the mapper and the policy service which are not evident in the Authoring Tool. For example, the mapper requires that policies declare the type of *users* (e.g. *principal investigator, co-investigator, public user)* and data (*experimental data, image data, numerical data)* in each policy which uses them. Such declarations are common (if not universal) in programming languages and are easy to create once they are understood.

A second example concerns authorisation obligations which must be linked to the appropriate authorisation rules for the policy service to apply them at the appropriate time. However, if an action type is used in an authorisation rule, then this rule will be expanded into separate rules for the policy service, one for each action. If obligations apply to sub-types of the action, they will be associated with all the rules (all the subtypes of action) generated from the authorisation rule. Once interactions like this are understood, then rules can be written to avoid this. For example, in the rules shown above, there are two rules allowing *access* to data by *public users* after the *embargo period*, one of which refers to the general action *access*, while the other refers to the specific action, *download*. This is so that the obligation for notification can be linked only to the specific authorisation for downloading data.

Another interaction is also illustrated by this example: since both authorisation rules authorise *access* to public users, their ordering becomes important in the policy service for them to be correctly applied. The more specific rule (that for *download*) should come before the more general rule (that for *access*).

It is easy to understand that when individual components of the system (e.g. authoring tool, mapper, policy service) have been developed by teams in different organisations that interactions and dependencies between components are not recorded in the documentation for the component with which users interact. The documentation for the authoring tool does not include these constraints which arise from component interactions. It would be easier for the user in any development of this prototype if these constraints were formally imposed within the authoring tool itself.

4.3.2. Evaluation of the controlled natural language for policies

Six research managers and six IT technologists completed the questionnaire in Annex 1 answering the question of whether the controlled natural language (CNL) had a usability closer to natural language (NL) than to the enforceable policy language (EPL) used in the policy service.

The raw results are shown in Table 3 below. In summary, a score of 0 states that the CNL usability equals the EPL, while a score of 10 judges the usability of the CNL to be equal to that of NL. The overall average score of 5.99 is significantly ($z = 3.75$, confidence > 95%) closer to NL that to EPL showing that the CNL does its job and is an improvement on EPL. Indeed one of the subjects clearly underlined this in one of the technologists additional comments of *"I hope I never need to read Enforaceable (sic) Policy Language again".*

	Q#/Subjects	Q1	Q2	Q3	Q4	Q5	Q6	average
technologist	**t1**	**6**	**3**	**4**	**3**	**4**	**5**	4.17
technologist	t2	7	4	5	3	5	6	5.00
technologist	t3	8	7	2	6	5	1	4.83
technologist	t4	3	2	5	6	4	5	4.17
technologist	t5	3	3	3	1	1	7	3.00
technologist	t6	3	5	3	7	2	8	4.67
manager	m1	8	7	7	8	7	7	7.33
manager	m2	7.5	7.5	7.5	7.5	7.5	7.5	7.50
manager	m3	8	8	8	8	8	8	8.00
manager	m4	8	8	8	8	8	5	7.50
manager	m5	8	8	8	8	8	8	8.00
manager	m6	8	8	8	8	8	6	7.67
	total	77.5	70.5	68.5	73.5	67.5	73.5	
	average	6.46	5.88	5.71	6.13	5.63	6.13	
	standard deviation	2.16856	2.317179	2.300774	2.450649	2.496589	2.00142	

Table 3: Raw scores answering the questionnaire on usability of the controlled natural language (rows: subjects; columns: questions).

More interestingly, there is also a highly significant difference (t =-9.74, confidence > 99.9%) between the answers given by the two groups of subjects. The average score of the technologists was only 4.31 while the managers gave a higher average score of 5.99 showing that they found the CNL more usable than the technologists who were more familiar with using computer languages.

Among the specific comments from subjects it is worth noting the introspection of one that "*the differences are related with the length of the controlled NL*". Clearly if the controlled NL appears to be significantly longer than the NL then there is a problem for managers, if not for technologists.

A second subject misunderstood the use of the sets in the CNL "*AND that Data hasBeamlineScientist a Set implies that some image data have this not set, and somehow contradicts that a beamline scientist can download all image data*". The use of sets in the CNL was previously avoided in the project by including special case code to manage the relationships where they are now used. This special case code was removed in the last year of the project to ensure that the Authoring Tool, Mapper and Policy Service would all be completely generic, and not tied to any particular application. To ensure the usability of the CNL, it may be necessary to find an alternative solution to the use of the set notation in the CNL where it is currently required.

A third problem identified by several subjects was with the last question in which the NL includes a simple disjunctive OR to separate list items, while the controlled NL was designed to require each item in the list to require the re-statement of the whole policy. This was done because with the use of both conjunction (AND) and disjunction (OR), an unnatural delimiter would have to be included (e.g. bracketing) to clearly define the limits of the scope of the operators. To avoid this problem the solution used was chosen. This design choice may have been a mistake, and future development of the controlled NL should include the use of the

disjunction with the bracketing notation to overcome the problem identified by users with the current approach.

4.3.3. Coverage of DSA by CNL and the authoring tool.

The structure and scope of DSA were analysed in D2.1. DSA are constructed as legal documents which usually include the following sections which include clauses which can be expressed as declarations in the DSA authoring tool:

- Definition of terms
- Parties to the agreement
- Period of agreement

A DSA includes policies which can be expressed in the authorisation, prohibition and obligation modalities which are supported by the Consequence framework:

- Description of the data
- Data Quality
- Description of the data users
- Method of data access or transfer
- Location of data and custodial responsibility
- Restrictions on Data Use.
- Derived data
- Dissemination to third parties
- Confidentiality & Privacy
- Disposal of data

A DSA includes clauses which describe the intent of the agreement which cannot be addressed within the Consequence Framework:

- Purpose of the agreement
- Justification for access

A DSA includes textual clauses meeting legal requirements which cannot be addressed by the Consequence Framework:

- Administration of the Agreement
- Breaches to the agreement
- Applicable Law

As a legal document, a includes the requirement for the collection of:

- Signature

The DSA policies which the Consequence Framework can address in principle require appropriate vocabularies to be defined, and will probably require additions to the state transition grammar in the DSA authoring tool to be able to be written in that tool. Equally, many of these clauses will include obligations which will require obligation handlers to be written to enforce them within an implementation of the Consequence Framework. However, there should be no major issues with extending the implementation to address these within the Framework.

4.3.4. Evaluation of the policy analysis tool

It has not been evaluated whether the POLPA produced by the DSA authoring tool is syntactically or semantically correct, nor if the semantics of the original OWL ontology are preserved in the formal analysis. The Analysis Tool accepted the output of the authoring tool that it was presented with without error messages, and analysed it. Since no knowingly ill formed input was provided whether the tool provides appropriate error messages in this case was also not evaluated.

When the analysis tool is launched the user first sees the screen shown in Figure 15. This presents the XML version of the DSA that has been created. The two subjects who tried the tool were lost at this point. They neither knew what they were being shown nor what to do with it. They were instructed to select the option to continue with the analysis, but were immediately put off the tool by this apparently redundant presentation.

The next screen (see Figure 16) again presents the XML version of the DSA, below which is a box to enter context properties and below that queries. This screen cannot be used without considerable instruction in what the analysis does. A simple description that asserting the context properties are true, and then querying whether a user role can perform an action on some data was not enough. They had to know more about the representation of the context properties and the terms in the queries. No user documentation was available to provide this instruction, so verbal instruction was required by the evaluator.

After instruction subjects were able to enter context properties and queries as shown in Figure 16, to produce the analysis results shown in Figure 17. The table was interpreted as showing that users could access data and data files under the context set whereas signatories of the DSA contract could not. One user asked why the columns showed different types of agent (e.g. Person, Signatory, User, System) while the policies differentiated control by user role (e.g. public user, principal investigator) which were not mentioned. The evaluator was unable to explain this presentation. Both users asked why the columns included the titles “User” and “Event” since they could not understand how a person/user/signatory (shown in the rows) could be expected to access either of these. After discussion they accepted the possibility of the System accessing and “Event” but the role of the “User” column defied explanation.

Given the level of miscomprehension with what the tool was doing, the subjects did not want to continue with the analysis since they had no confidence in what it was doing. In order to build trust in the tool, there needs to be a clearer explanation of what it does than was provided by a five minute video of one example working.

The analysis tool provides no support for analysing policies from multiple DSA documents that apply to the same experiment between which there may be conflicts.

4.3.5. Evaluation of the deployment and enforcement

As would be expected, when the DSA is deployed and policies are correctly enforced the only feedback to the user by the lifecycle manager is the notification of success. The user only gets detailed feedback when there are errors. Errors could be real errors in the policy system itself, although none of these were identified while the system was being evaluated.

When the user has permission to access or use a data item denied, the user is merely told that alone. It was originally planned to provide more complex error messages to explain to the user the reason for the denial in order to improve the transparency of the user interface. But, following clear advice from the reviewers and experienced security staff in the project it was

accepted that this would provide potential hackers with too much helpful information and only simple denial statements were adopted.

The policy system did report errors during evaluation, but these had more complex origins than in the policy system alone, usually resulting from a syntax or lexical error in the DSA input for deployment. Although some of these are simple such as a report that the user is trying to declare a variable twice, they are sometimes complex to interpret. For example, when loading a DSA the message "*line 28:47 no viable alternative at input 'filter'*" states that the action “filter” was a reserved word in EPL itself while it was also being used as a planned action in the application. In practice this action was added to the ontology as a subtype of transform to produce derived data, and was entered into the mapper GUI, but the application did not support it. This is a clear identification of an error, and a clear report of it by the policy system. The interpretation of the error needs an understanding of the overall system including the ontology, mapper, policy system and application. No single component could provide such an explanation. However, this illustrates that there is a need for a technically informed user of Consequence enabled applications to debug such errors.

4.3.6. Evaluation of the DPO-API and JNBridge

The DPO-API has been used throughout the project by two experienced Java programmers. There evaluation after using it during the project is as follows.

The DPO-API provides the functionality required for this evaluation.

The DPO-API is used via the JNBridge proxy which is a Java jar file. There is no java doc in this file for the programmer to know what the calls are or their arguments. The only way to know what calls are is to look at the Java method signatures themselves. This may be a result of JNBridge producing the DPO-API project rather than the original .Net implementation, but it is still a problem for an implementer. The written textual documentation, as in all active development projects, is always out of date, and cannot be relied upon. This approach is manageable while the author of the DPO-API is available within the project but it limits the use of this component, and therefore the overall architecture, after the project when he is no longer available.

Performance of the DPO-API appears to raise no issues, including the encryption and decryption of large files which operates within acceptable user time.

As illustrated in Figures 34 and 35, there are many calls to different methods in the DPO-API from the application client. These should be simplified if wider use is planned after the project.

In 2008 when the project selected JNBridge as the tool to integrate Java and .Net implementations there were four products on the market - JNBridge, JIntegra, JNI and IKVM. JNBridge had most features supported at the time. There are two main weaknesses with JNBridge to do with the complexity of the configuration and its use for multi-layer applications.

Configuration requires manual editing of class paths which is very complex and error prone to manually type the 20 odd components in a class path. There should be a graphical tool for this, or it should be encapsulated and hidden from the user.

It is acceptable for single layer interactions between Java and .Net but when there are multi-layer calls between the two (e.g. Java to .Net and back to Java) JNBridge gets confused between function names in the different languages and in different Java packages, and

between the class paths of the different installations of Java. JNBridge was unable to locate java classes when 3 layers of interoperability were required within a J2EE server. Consequently, we had to run DPOAPI outside the J2EE server. To overcome this, the implementation required web services wrappers around the DPO-API so that Java could call .Net to then call Java in the PDP which is what the product is marketed for.

Consequently, users cannot encapsulate functions in one language into a product written in the other language since the end user has to generate a single proxy for multiple separate components, even for embedded Java written by other implementers about which they may know nothing.

JNBridge should be initialised once only per Java Virtual Machine, but in reality it has to be initialised once per instance of an application running which wants to use it. Application users should not have to initialise such embedded products explicitly.

It is very difficult to debug mis-configuration errors or a badly generated proxy file because the logging is poor. The developers also reported that they considered JNBridge to be expensive for what it does; although its price has reduced since the project originally purchased copies.

Although JNBridge still appears to be the best product available in the market to fulfil this role, in conclusion, it was a mistake to attempt such tight integration of Java calling .Net code and *visa versa* as we did in this project and in future we would be more conservative and use Web Services interfaces between code written in the two languages, or use one language alone on client systems.

4.3.7. Evaluation of the application using the Consequence framework

Use cases two and three include the use of the iCAT application to download and view data files respectively. Two technically knowledgeable users stepped through these two scenarios using the application. They had previous experience of the iCAT tool and its interface – without the Consequence framework modifications. They stepped through the screens shown in Figures 22-29 without user interaction problems, and without noticing that this tool was not the usual iCAT access tool with the exceptions discussed below.

In use case two the users had to download a data file and view the numerical and image data it contained. When logged in as a PI they were able to view the data, when logged in as a guest user they found that they were presented with less data to download – see Figure 36. They both commented that it was wrong not to show the guest user that the data existed but could not be downloaded since it would be available after the embargo period was over, and they should be told this. Meeting this request would be counter to the security principle explained in section 5.3.4.

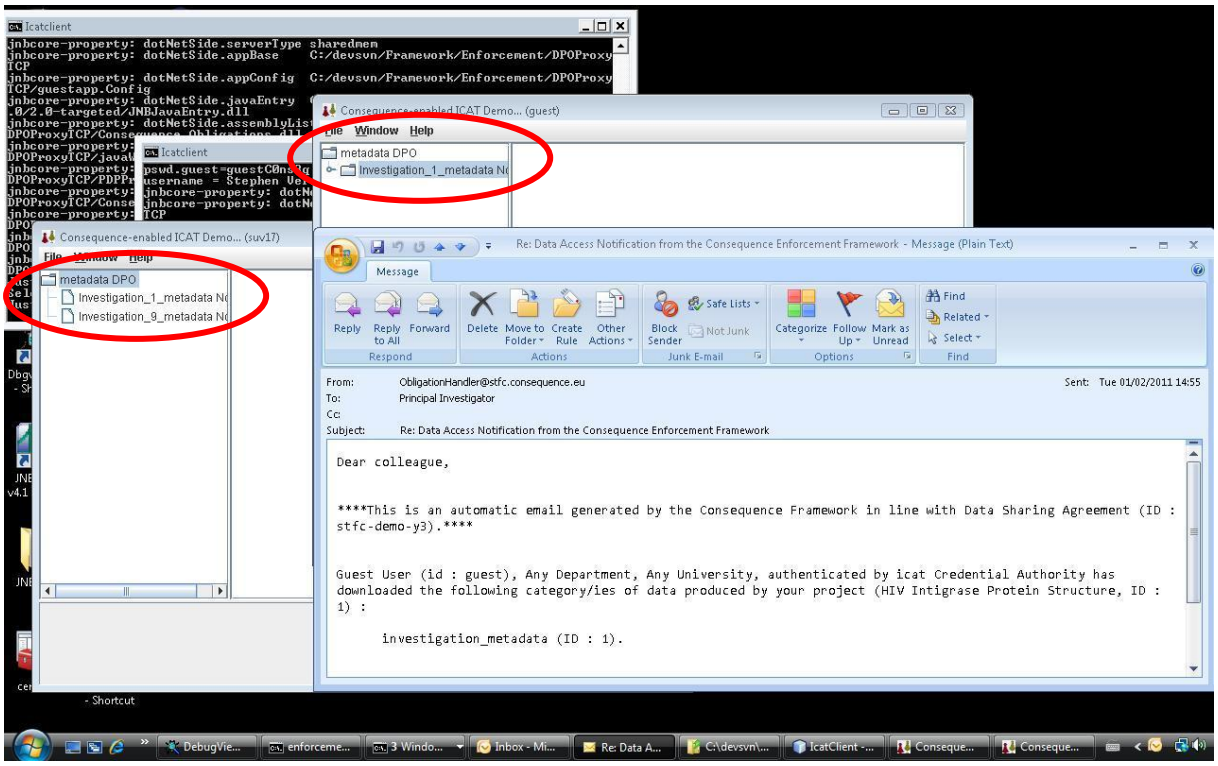

Figure 36: Both a PI and a guest user logged in two different instances of the client application issuing the same request to view data on HIV. The PI on the left sees 2 datasets while the guest on the right sees only one, since the other is under embargo. The PI is notified by e-mail that the guest has downloaded the available dataset.

When the subjects were denied access to data they both complained about the error message, wanting to know why, and what they could do to get access to the data they wanted. Meeting this request would be counter to the security principle explained in section 5.3.4.

When the location was set to Iran (see Figure 35) both users complained that it was unrealistic to restrict data access from a location. The data policy that enforces this constraint was invented for the demonstration of location context sensitive technology within the project and is not derived from the existing STFC data policy. However, similar policies have been adopted in the data policies of other organisations and may apply to some classes of data at STFC in the future.

These three problems the subjects identified with the application all arose from the introduction of policy based access and usage control which is what the Consequence framework is intended to do. In each case the subjects were constrained in their usual behaviour, which is what was intended. In conclusion, these observations are the result of the intended changes rather than problem with the application. They do indicate that the introduction of technology such as Consequence will have to address human issues in order to be adopted as well as the technical issues that it has already considered.

Both subjects also complained that the iCAT client did not allow them to do any real analyses which they would do with other client tools such as Mantid[1] once they had downloaded using iCAT. In order to introduce trusted applications such as iCAT it will be necessary to introduce a suite of trusted application versions of existing tools and not just one.

[1] http://www.scientific-computing.com/news/news_story.php?news_id=1354

4.3.8. Evaluation against requirements

This section presents an assessment of how well the system meets the requirements described in D6.1 at the start of the project, and reproduced in Table 2 above.

The majority of the requirements appear to be met satisfactorily: BR1, BR15, AR2, BR4, BR5, BR6, TR6, BR14, BR7, BR8, BR9, TR7, BR12, BR13, TR11, BR17, BR21, BR22, AR2, AR3, AR4, BR23, TR1, TR2, TR5, BR18, TR8, TR9, TR10, AR1, TR13, TR15, TR14, BR19, BR20.

The following requirements require further consideration:

BR2 A single party may have multiple agreements with different parties on the same dataset.

There is nothing in the Consequence Framework or Architecture to prevent this from being true, but there is no mechanism in the analysis to determine conflicts between multiple DSA unless the clauses are combined together into a single DSA document for the purposes of analysis.

BR3 An agreement should include references to relevant external agreement/s and legislation.

There is nothing to prevent the textual agreement to include a representation of external agreements or legislation, but they would have to be encoded, and cannot be included by reference to the existing natural language documents.

BR10 Capability to propagate data sharing policies from parent to derived dataset/s.

The implemented Consequence Framework evaluated in this report does not support derived data explicitly. Derived data has been addressed within the project as a research topic and developments in this area have been reported elsewhere in project deliverables and academic publications.

BR11, TR7 Capability to trace the origin of inherited policies in a derived dataset back to the parent dataset.

The implemented Consequence Framework evaluated in this report does not support derived data explicitly. Derived data has been considered within the project as a research topic and developments in this area have been reported elsewhere in project deliverables.

BR16 An agreement may be updated by the owner if the resultant formal policy does not conflict with other formal policies in force on the same dataset. The agreement end parties will be informed and their consent sought.

The Consequence Framework as implemented does not address the informing of parties or the seeking of consent. The technologies for this, like the digital signing of the DSA are well understood and can be added to the implementation evaluated if/when any partner takes the results of the project further towards a product.

BR24 Demonstrate cost-benefit effectiveness to community data providers.

This evaluation was intended to provide this demonstration. It does not explicitly address cost-benefit. The evaluation of business benefits in section 4.4 addresses the benefit part of this requirement, although it is not possible at

this time to realistically evaluate the costs of deploying and maintaining the prototype for managing data security.

TR3 The proposed architecture should be platform independent.

The project adopted JNBridge to support the running of components built on the Java and .Net platforms. The evaluation of JNBridge in section 4.3.6 above identified its limitations in achieving this.

TR4 The proposed architecture could interoperate with a variety of software platforms.

The architecture includes components which run on both .Net and Java platforms given the limitations of JNBridge identified above.

The failure to completely meet these requirements does not impact on demonstrating the STFC prototype. However, they may have an impact on the adoption and assimilation of a tool derived from the prototype if they are not addressed.

4.3.9. Evaluation against risks

The implemented prototype will be evaluated against the risks defined in section 3.2:

RI.1 The scientific data is released to people to whom it should not be.

The evaluation of the application has shown that the enforcement mechanism prevents access to scientific data unless the policy being enforced permits it.

RI.2 Not all stakeholders who should have access to data or metadata are granted access when they should be.

The evaluation of the application has shown that the enforcement permits access to those to whom the enforced policies grant it.

RI.3 Scientific metadata is released which allows competitors to determine that a particular chemical could have an impact on a technological problem – e.g. that a potential drug may impact on a specific disease.

The evaluation of the application has shown that the enforcement permits applies to metadata as well as to data.

RI.4 Data sharing agreements do not contain policies which address all realistic situations which arise for all stakeholders.

The evaluation of the coverage of the DSA by CNL and the authoring tool suggests that all clauses required can be written using the Authoring Tool and enforced by the Policy Subsystem. The analysis tool does provide a check that policies should not allow access in unexpected situations. However this risk still exists.

RI.5 The policy translation process enforces policies different from those stated in the DSA.

The example policies evaluated appear to be evaluated in the same way as the DSA authors intended, but since the policies are transformed through a series of languages from the controlled natural language to EPL rather than undergoing refinements which preserve their semantics. A semantics preserving transformation approach to reduce this risk was investigated as a research issue, but the risk still exists in the implemented prototype.

RI.6 The Consequence technologies do not record attacks on them, or respond to attacks in such a way as to protect themselves against future attacks.

The project has not addressed this risk. Any developments of the prototype or the Framework after the project would have to address this risk.

R1.7 The quality of metadata in the application maintained about user roles and data types is too poor to support accurate enforcement of policies which are based on these properties.

This risk arises from the data quality management procedures of the application user and not from the Consequence Framework or prototype implementation. This risk still exists.

In summary, the evaluation of the prototype has reduced these risks, but they still remain to be considered in any development or future exploitation of the project results.

4.4. Business Impact

Scientific funding bodies wish to maintain data security for the principal investigator to gain the scientific rewards and financial benefits which can result from it, while also opening up data access to encourage:

- validation of analyses by other scientists,
- reuse of data for secondary studies,
- use of the data for other analyses not considered by its creator.

As a result of awareness of the project and the analyses of data policies and DSA undertaken at the start of it, awareness of data policies and data sharing agreements has increased in STFC, so that it now has an agreed data policy for publically funded research across the institution and across European facilities as described in Appendix 2. This will have a considerable impact in the development of data sharing across European scientific facilities.

STFC is now working with its users to establish DSA so that this data policy can be enforced not only within STFC's facilities but also to the data elsewhere. This development will also have a significant impact on data sharing across European scientific facilities.

The authoring system for DSA demonstrated in use case one was not accepted by senior managers or lawyers because although it uses a controlled natural language it does not use complete natural language with every word and indeed comma serving its role as the lawyers demand in order to meet the requirements of legal precedent. However, it is accepted by senior managers as an intermediary language which they and technical staff can exchange to ensure that the policies which are enforced convey the intention, and the detail, of the binding data policy thereby reducing errors in enforcing policies.

The enforcement mechanism demonstrated in use case two for data access control demonstrates server side control that is required to enforce the data policies, which are in turn required to enable the desired broader data sharing. The inclusion of obligations in the data policies to inform research staff of potential re-use of their data that may result in citation of their work or economic impact has proved popular with those researchers, although it was not considered before the project. The exact implementation developed in this project has been demonstrated to STFC directors and technical staff and allowed them to understand what the technology has to offer and how it meets their needs. Large parts of the server side solution implemented in the project are expected to be adopted by STFC and their adoption is being encouraged by the other large European facilities which have adopted the common data policy.

The use of the trusted application approach as demonstrated by the iCAT client in use case three to support usage control both on and off line are not seen as feasible to adopt across those software tools which scientists currently use to analyse data. Individual researchers often develop their own tools to analyse data which incorporate new techniques they have developed. No mechanism is currently foreseen to force scientists to make their own developed applications into trusted applications which would support the Consequence

framework. However the demonstration of the Consequence tools links to other developments about establishing the provenance of scientific results produced by analyses which will also mandate changes to these tools. These developments may have a significant impact on the broader developing European common scientific data infrastructure in the future when such usage control systems can be enforced more generally.

Between them these developments will have a positive impact on the STFC business, and on that of our scientific collaborators across Europe.

It was intended that this report demonstrate cost-benefit effectiveness of the Consequence solution to scientific community data providers. The benefits of data security management have been described above. It is not possible at this time to realistically evaluate the costs of deploying and maintaining the prototype for managing data security.

5. Conclusion and Recommendations

The evaluation has considered how the prototype and the Consequence Framework on which it is built meet the requirements for the STFC scenario, how usable the system is to its different stakeholders, and how complete the expression of DSA is within the Framework.

The implementation that has been evaluated is a research prototype designed to demonstrate the Consequence Framework in action, and how it integrates many research technologies to address a real problem. The implementation is not a product or even the alpha release of a product.

An ambitious aim of the project was to combine the individual technologies incorporated in the Consequence infrastructure, from a state of being reported individually in the research literature, to be demonstrated working together in the project prototypes, with the prospect of being able to be incorporated by non-expert developers into demonstration projects.

The evaluation has shown both that the prototype meets many of its objectives, and fails to meet many requirements that it would have to in order to be used more widely than as a demonstrator within the project. Therefore the first part of this aim of demonstrating the combination of the technologies has been achieved.

The prototype meets many technical objectives, while those covering the usability & consistency of the user interface, and the completeness of documentation are particularly poorly achieved. Consequently the second part of the aim, to allow non-expert developers to use the Consequence infrastructure to produce their own demonstration projects without further assistance has not entirely been met. Since the project had no plan to release the infrastructure as an open-source toolkit which could be picked up by any developer, and the only way to gain access to the Consequence infrastructure has always been through one of the project partners, it is not a significant problem that project partners would be required to provide some additional guidance to those using the infrastructure.

The recommendations noted throughout section 4.3 will have to be met to explain to stakeholders how the system works, and improvements to the technology that are required if it is to be developed further and exploitated. These are summarised below:

Component	**Recommendation**
All user interfaces	Impose a common look and feel to user interfaces
Protégé	Guidance documentation on style of Consequence ontology design
	Upgrade Consequence tools to OWL 2 so that latest version of Protégé can be used.
Mapper	Documentation, and examples to make mapper use clear
Authoring	Add delete last term functionality to editor
	Documentation on the accepted controlled language syntax, and on constraints on policies made by the mapper and set by the policy service
Analysis tool	Documentation to explain what the analysis does at a high level and ata low level context setting & query construction

	Remove first screen that presents XML DSA
	Enable analysis of conflicts between multiple agreements
Policy Service	Documentation on the error messages relating them to other processes in the infrastructure, e.g. what to change in the Authoring tool to avoid an error.
DPO-API	Accurate documentation on calls and their arguments
	Simplification of calls required
JNBridge	GUI editor for configuration
	Implement support for multi-layer calls between Java and .Net
Framework	Encode legislation as library for DSA to ease adoption
	Implement research results on inheritance of policies to derived data
	Implement secure DSA signature collection
	Implement reliable platform independence of solution
	Implement attack detection and protection on the Consequence infrastructure itself.

6. References

[1] Crompton, S., Aziz, B., Wilson, M., and Arenas, A. (2008) Consequence D6.1: Requirements Specification for the Sharing Sensitive Scientific Data test bed. Consequence project: http://www.consequence-project.eu/deliverables.html

[2] Star Consulting (2010) Data Security: a way forward in the cloud, Aug 2010, Computing. http://www.star.co.uk/Log/?pdfname=/Global/pdf/Data%20security%20a%20way%20forward%20in%20the%20cloud.pdf

[3] Orlov, A (Ed) (2010) Final Version of the Consequence Architecture, Consequence project.

[4] Higginbotham, S (2010) We Can't Squeeze the Data Tsunami Through Tiny Pipes, May 4th, http://gigaom.com/2010/05/04/we-cant-squeeze-the-data-tsunami-through-tiny-pipes/

[5] A Digital Universe Decade – Are You Ready? http://gigaom.files.wordpress.com/2010/05/2010-digital-universe-iview_5-4-10.pdf

[6] Wood, J. (2010) Riding the wave – How Europe can gain from the rising tide of scientific data, report of the EU High level expert group on scientific data, http://www.grdi2020.eu/Repository/FileScaricati/c2194260-3ddf-47bd-93e4-68f8912a3564.pdf

[7] Goth, G., (2010) Turning Data into Knowledge, Communications of the ACM, 53(11) 13-15.

7. Glossary

CNL, CNL4DSA – Controlled Natural Language used to define a DSA. It is English like but controlled in terms of the vocabulary, syntax and semantics which can be used.

DLS – Diamond Light Source, Synchrotron Radiation Source at the STFC Rutherford Appleton Laboratory, used to determine the structure of chemicals.

DPA – Data Protection Act, 1998 UK Legislation to protect the privacy of consumers

DSA – Data Sharing Agreement, an agreement between two or more parties about how data will be shared between them.

DSA Lifecycle Manager - module of the Consequence architecture responsible for managing the DSA while it is authored, mapped, deployed, modified and deleted.

EPL – Enforcable Policy Language used by the Consequence policy service to describe policies.

GPS – Global Positioning System, is a space-based global navigation satellite system (GNSS) that provides reliable location and time information in all weather and at all times and anywhere on or near the Earth when and where there is an unobstructed line of sight to four or more GPS satellites. It is maintained by the United States government and is freely accessible by anyone with a GPS receiver.

GUI – Graphical User Interface to a computer, usually incorporating mouse based input.

HPC – High Performance Computing, large computer system incorporating parallel execution of nodes with fast interconnection used to run system simulations which can synchronise after each cell is computed. Contrasts with High Throughput Computing or High Capacity Computing which do not incorporate fast interconnection between computing nodes, so simulations cannot synchronise.

iCAT – Information cataloguing software for data management of research results from scientific facilities – so far it only supports raw data resulting from the experiment itself, although it is being developed to address derived data resulting from analysis steps where the provenance of the analysis software must also be retained.

IPR – Intellectual Property Rights

ISIS – Neutron scattering facility at the STFC Rutherford Appleton Laboratory, used to determine the structure of materials.

Java – programming language supported by Oracle which allows developers to use the same set of skills to build programs on a wide range of platforms.

Mantid - Mantid (Manipulation and Analysis for ISIS Data) is an open source project for data reduction and analysis of neutron scattering experiments.

MRC – Medical Research Council, UK science funding body for medicine

Muon - An elementary particle similar to the electron, with a unitary negative electric charge and a spin of ½. Together with the electron, the tau, and the three neutrinos, it is classified as a lepton. It is an unstable subatomic particle with the second longest mean lifetime (2.2 μs).

.Net – programming framework supported by Microsoft which allows developers to use the same set of skills to build programs on a wide range of platforms.

Neutron - A subatomic particle with no net electric charge and a mass slightly larger than that of a proton. With the exception of hydrogen, nuclei of atoms consist of protons and neutrons, which are therefore collectively referred to as nucleons.

NHS – National Health Service in the UK, free at the point of use.

NL – Natural Language such as English, as contrasted with artificial programming languages, or controlled/constrained natural languages (CNL) which support a subset of NL.

OECD - Organisation for Economic Co-operation and Development

OWL - The Web Ontology Language (OWL) is a family of knowledge representation languages for authoring ontologies. The languages are characterised by formal semantics and RDF/XML-based serializations for the Semantic Web.

Petabytes – 1,000,000,000,000,000 bytes.

PI - Principle Investigator, or lead scientist on a study.

POLPA - POLicy Process Algebra: a process algebra based formal software engineering language for expressing policies.

RCUK – coordinating body for UK Research Councils – research funding bodies in different disciplines

STFC – Science and Technology Facilities Council, UK science funding body and large scientific facilities provider

SRS - Synchrotron Radiation Source at the STFC Daresbury Laboratory, now decommissioned and replaced by the Diamond Synchrotron at the STFC Rutherford Appleton Laboratory.

STS – Secure Token Service, a secure Web Service which provides tokens for values such as user identity (UID) or user location.

TRL – Technology Readiness Level, originally developed by the US DoD and adopted by NASA, is a measure used to assess the maturity of evolving technologies (materials, components, devices, etc.) prior to incorporating that technology into a system or subsystem.

UID – User Identity

Appendix 1: Policy Language Usability Evaluation

Introduction

The questionnaire includes six questions, each of which I would like you to answer by placing a “X” over the appropriate number on the scale of 1-10. Then return the edited file or the 6 numbers in order to michael.wilson@stfc.ac.uk

This very brief study is part of an EU funded project which has developed technologies to author and enforce data policies. Data policies are enforced using a predicate language which is considered difficult for managers to use. The project has developed a constrained natural language for policies whose usability I would like you to assess.

Scenario

The scenario behind the questions is that: experiments are undertaken at a large scientific facility, with many experimental stations called beamlines; teams of researchers including a principal investigator, co-investigators and a scientist who manages the beamline undertake each experiment; experimental data is stored in a format called Nexus and includes images and numerical data; access to data is embargoed for a period following the experiment after which it is available to the public.

Question Format

Each of the six questions overleaf has four components:

A statement in of a data policy clause in English natural language.

A statement of the same data policy clause in the enforceable policy language.

A statement of the same data policy clause in the constrained natural language – one or more clauses.

A scale of numbers from 1 to 10.

Judgement Reporting

You are asked to judge how usable you find the constrained natural language, and to record that judgement on the scale of numbers where 1 is less usable than 10.

The scale should be used to score:

1 – the constrained natural language is as usable as the enforceable language

10 – the constrained natural language is as usable as usable as the English natural language

2-9 – the constrained natural language is progressively judged to be closer in usability to the English natural language as the numbers increase.

DO NOT JUDGE WHETHER THE DATA POLICIES ARE APPROPRIATE FOR YOU

Question 1

English NL:

Any registered user can access any data after its embargo period has passed.

Enforceable Policy Language (EPL):

EPL policy is written with knowledge that data access can mean read or generateGraph action:

```
authorization aIread = allow read()
        target( data_category=="experimental_data" ) to any
        when( currentTime() > object.embargo_end_date);

authorization aIgenerateGraph = allow generateGraph()
        target ( data_category=="experimental_data" ) to any
        when ( currentTime() > object.embargo_end_date);
```

Controlled NL:

IF a Data hasEmbargoEndDate a EmbargoEndDate
AND a CurrentTime isAfter that EmbargoEndDate
THEN a User CAN Access that Data

Answer Scale:

1– the controlled NL is as usable as enforceable language (EPL)

10 – the controlled NL as usable as English natural language (NL)

1 2 3 4 5 6 7 8 9 10

Question 2

English NL:

Principal investigators and co-investigators can access their own data at any time.

Enforceable policy language (EPL):

EPL policy is written with knowledge that data access can mean read or generateGraph action:

```
authorization a2read = allow read()
        target ( data_category=="experimental_data" ) to principal_investigator
        when ( authentication.uid==object.principal_investigator_uid );

authorization a2generateGraph = allow generateGraph()
        target ( data_category=="experimental_data" ) to principal_investigator
        when ( authentication.uid==object.principal_investigator_uid );

authorization a3read = allow read()
        target ( data_category=="experimental_data" ) to co_investigator
        when ( authentication.uid==object.principal_investigator_uid );

authorization a3 generateGraph = allow generateGraph ()
        target ( data_category=="experimental_data" ) to co_investigator
        when ( in ( authentication.uid , object.co_investigator_uids ) );
```

Controlled NL:

IF a Data hasPrincipalInvestigator a Set
AND a User hasRole a PrincipalInvestigator
AND that User in that Set
THEN that User CAN Access that Data

IF a Data hasCoinvestigator a Set
AND a User hasRole a CoInvestigator
AND that User in that Set
THEN that User CAN Access that Data

Answer Scale:

1– as usable as enforceable language

10 – as usable as English natural language (NL)

1 2 3 4 5 6 7 8 9 10

Question 3

English NL:

Beamline scientists can access image data from experiments performed on their beamline at any time.

Enforceable Policy Language (EPL):

```
authorization a4read = allow read()
        target ( data_category=="image" ) to beamline_scientist
        when ( in ( authentication.uid , object.beamline_scientist_uids ) );

authorization a4generateGraph = allow generateGraph()
        target ( data_category=="image" ) to beamline_scientist
        when ( in ( authentication.uid , object.beamline_scientist_uids ) );
```

Controlled NL:

IF a Data has as data category a ImageData
AND that Data hasBeamlineScientist a Set
AND a User hasRole a BeamlineScientist
AND that User in that Set
THEN that User CAN Access that Data

Answer Scale:

1– as usable as enforceable language

10 – as usable as English natural language (NL)

1 2 3 4 5 6 7 8 9 10

Question 4

English NL:

Whenever a user downloads a data file that is not from their own project the system must log the event.

Enforceable Policy Language (EPL):

```
authorization a5download = allow download()
        target ( data_category=="experimental_data" and file_type=="nexus" ) to public_user
        preObligation log();
```

Controlled NL:

IF a User hasRole a PublicUser
AND a NexusFile has as data category experimental data
THEN BEFORE that User Download that NexusFile
THEN a System MUST Log a Event

Answer Scale:

1– as usable as enforceable language

10 – as usable as English natural language (NL)

1 2 3 4 5 6 7 8 9 10

Question 5

English NL:

Whenever a user reads a data file produced by an experiment in which they were not a participant the system must e-mail the principle investigator of the project whose data was read.

Enforceable Policy Language (EPL):

A new policy statement is not required, the previous authorization statement can be extended to include new obligation:

```
authorization a5download = allow download()
        target ( data_category=="experimental_data" and file_type=="nexus" ) to public_user
        preObligation log()
        postObligation notify(object.principal_investigator_uid);
```

Controlled NL:

IF a User hasRole a PublicUser
AND a NexusFile hasPrincipalInvestigator a Set
AND a User hasRole a PrincipalInvestigator
AND that User in that Set
THEN AFTER that User Read that NexusFile
THEN a System MUST Notify that User

Answer Scale:

1– as usable as enforceable language

10 – as usable as English natural language (NL)

1 2 3 4 5 6 7 8 9 10

Question 6

English NL:

Numerical data cannot be accessed by any user in North Korea or Iran.

Enforceable Policy Language (EPL):

```
authorization a6read = deny read()
        target ( data_category=="numerical_data" ) to any
        when ( location.country=="North Korea" or location.country=="Iran" );

authorization a6generateGraph = deny generateGraph()
        target ( data_category=="numerical_data" ) to any
        when ( location.country=="North Korea" or location.country=="Iran" );

authorization a6download = deny download()
        target ( data_category=="numerical_data" ) to any
        when ( location.country=="North Korea" or location.country=="Iran" );
```

Controlled NL:

IF a Data has as data category a NumericalData
AND a User hasLocation a NorthKorea
THEN that User CANNOT Access that Data

IF a Data has as data category a NumericalData
AND a User hasLocation a Iran
THEN that User CANNOT Access that Data

Answer Scale:

1– as usable as enforceable language

10 – as usable as English natural language (NL)

1 2 3 4 5 6 7 8 9 10

Thank you for your cooperation.

If your curiosity has been raised enough to want to know more about the Consequence project of which this questionnaire is part you can find more details on the project website:

http://www.consequence-project.eu/

Prof Michael Wilson, STFC e-Science, Rutherford Appleton Laboratory, UK.

Please add any comments on this study:

8. Appendix 2: STFC Data Policy

One of the objectives of the project for STFC was to promote awareness of data policies and data sharing agreements as they apply to scientific facilities. During the period of the Consequence project STFC developed a data policy for publically funded scientific data which has been the basis of a common data policy adopted by the PANData consortium of neutron and photon facilities across Europe. This data policy available[i], and is copied below.

8.1. General principles

1.1 This data management policy pertains to the ownership of, the curation of and access to experimental primary data and metadata collected and/or stored at the facility.

1.2 Acceptance of this policy is a condition of the award of beamtime.

1.3 Users must not attempt to access, exploit or distribute raw data or metadata unless they are entitled to do so under the terms of this policy.

1.4 Deliberate infringements of the policy may lead to denial of access to raw data or metadata and/or denial of future beamtime requests at the facility.

1.5 All data and metadata will be subject to the data protection legislation of the country in which the data and metadata are stored.

8.2. Definitions

For the purposes of this policy:

2.1 the term **facility** refers to one of the Photon and Neutron facilities participating in the PaN-Data initiative.

2.2 the term **raw data** pertains to data collected from experiments performed on facility instruments. This definition includes data that are created automatically or manually by facility specific software and/or facility staff expertise in order to facilitate subsequent analysis of the experimental data.

2.3 the term **metadata** describes information pertaining to data collected from experiments instruments, including (but not limited to) the context of the experiment, the experimental team, experimental conditions and other logistical information.

2.4 the term **principle investigator** (PI) pertains to the PI identified on the experiment proposal. For experiments outside of the facilities proposal system, the PI is the person initiating or performing the experiment.

2.5 the term **experimental team** includes the PI and any other person to whom the PI designates the right to access resultant raw data and associated metadata.

2.6 the term **public research** refers to research done through peer review and leading to publication(s).

2.7 the term **proprietary research** refers to research done through purchased (commercial) access to the research facility.

2.8 the term **on-line catalogue** pertains to a computer database of metadata containing links to raw data files, that can be accessed by a variety of methods, including (but not limited to) web-based browsers.

2.9 the term **results** pertains to data, intellectual property, and outcomes arising from the analysis of raw data. This does not include publications.

2.10 the term **long-term** means a minimum of 5 years and facilities will thrive for 10 years. This may obviously depend on the type and volume of data concerned and the economical consequences associated to long-term data storage. Thus the facility reserves the right to restrict the storage periods in consultation with the respective communities for high data rate instruments.

2.11 the term **open access** means belonging to the community at large, unprotected by copyright or patent and subject to appropriation by anyone. Open access to research data from public funding should be easy, timely, user-friendly and preferably Internet-based.

8.3. Raw data and associated metadata

3.1 Access to raw data and associated metadata

3.1.1 All raw data and the associated metadata obtained as a result of publically funded access to the research facilities are open access, with the research facility acting as the custodian.

This is a key point. Publicly funded data are open access. However IP arising from analysis of the data is not public.

3.1.2 All raw data and the associated metadata obtained as a result of proprietary research will be owned exclusively by the client who purchased the access. Proprietary users must agree with the facility management how they wish their raw data and metadata to be managed before the start of any experiment.

3.2 Curation of raw data and associated metadata

3.2.1 All raw data will be curated in well-defined formats, for which the means of reading the data will be made available by the facility.

3.2.2 Metadata that is automatically captured by instruments will be curated either within the raw data files, within an associated on-line catalogue, or within both.

3.2.3 Data will be read-only for the duration of its life-time.

3.2.4 Data will be migrated or copied to archival facilities for long-term curation.

3.2.5 It is planned that each data set will have a unique identifier. Anybody providing data with the same identifier must make sure that the copy is identical to the data in the facility database. Anybody publishing results based on open access data must quote the same identifier (and related publications if available & appropriate).

3.3 Access to raw data and metadata

3.3.1 Access to raw data and metadata in the facility is foreseen to be via a searchable on-line catalogue.

3.3.2 Access to the on-line catalogue of the facility will be either open access or restricted to those who are registered users of the on-line catalogue.

Registration may be necessary for certain access to open access data due to potential bandwidth problems with large data sets. The underlying AAI (Authentication and Authorisation Infrastructure) is being worked on within PaN-Data and other EU funded projects.

3.3.3 Access to raw data and the associated metadata obtained from an experiment is restricted to the experimental team for a period of 3 years after the end of the experiment. Thereafter, it will become openly accessible. Any PI that wishes their data to remain*restricted access* for a longer period will be required to make a special case to the respective facility management. If data can only be stored at the facility for less than three years, then access is exclusive to the PI up to the end of the storage period. Data can always be made openly accessible earlier on simple request of the PI.

3.3.4 It is the responsibility of the PI to ensure that the experiment number is correctly entered into the metadata for each raw data set, in order to correctly associate each data set with the PI. If this is not done, the experimental team will not be able to access the data via the on-line catalogue or other users may inadvertently be given access rights to the data.

3.3.5 Appropriate facility staff (e.g. instrument scientists, computing group members) has access to any facility curated data or metadata for facility related purposes. Every facility will undertake that they will preserve the confidentiality of such data.

3.3.6 The on-line catalogue will enable the linking of experimental data to experimental proposals. Access to proposals will only ever be provided to the experimental team and appropriate facility staff, unless otherwise authorized by the PI.

3.3.7 The PI has the right to transfer or grant parts or all of his rights to another registered person.

3.3.8 The PI has the right to create and distribute copies of his raw data.

8.4. Results

4.1 Ownership of results

4.1.1 Ownership of all results (intellectual property) derived from the analysis of the raw data is determined by the contractual obligations of the person(s) performing the analysis.

This avoids the need for the facility to define IP ownership.

4.2 Curation of results

4.2.1 Each facility will provide a means for users to upload results and associated metadata to the facility and enable them to associate these results with raw data collected from the facility.

4.2.2 The upload of results and associated metadata may be subject to volume restrictions.

4.2.3 These results will be stored long-term by the originating facility. It will not be the responsibility of each facility to fully curate this data e.g. to ensure that software to read / manipulate this data is available.

4.2.4 The facility cannot be made liable in case of unavailability or loss of data.

4.2.5 The facility cannot be made liable in case of unavailability or loss of data analysis software.

4.3 Access to results

4.3.1 Access to the results of analyses performed on raw data and metadata is restricted to the person or persons performing the analyses, unless otherwise requested by those persons. However, if the raw data being analysed is still restricted, access to the analysis results must be granted to the PI on request.

8.5. Good practice for metadata capture and results storage

5.1 The experimental team is encouraged to ensure that experiments metadata are as complete as possible, as this will enhance the possibilities for them to search for, retrieve and interpret their own data in the future.

5.2 Each facility undertakes to provide means for the capture of such metadata items that are not automatically captured by an instrument, in order to facilitate recording the fullest possible description of the raw data.

5.3 Researchers who aim to carry out analyses of raw data and metadata which are openly accessible should, where possible, contact the original PI to inform them and suggest a collaboration if appropriate. Researchers must acknowledge the source of the data and cite its unique identifier and any publications linked to the same raw data.

5.4 PIs and researchers who carry out analyses of raw data and metadata are encouraged to link the results of these analyses with the raw data / metadata using the facilities provided by the on-line catalogue. Furthermore, they are encouraged to make such results openly accessible.

8.6. Publication information

6.1 References for publications related to experiments carried out at the facilities must be deposited in a publications database within 3 months of the publication date, or during any new application for beamtime, whichever is the earlier.

[i] http://pan-data.nd.rl.ac.uk/PaN-Data_Scientific_data_Policy_Draft