Iridium: Data Catalogue Specification

Author: Paul Thompson 07 June 203

Project Goals

- To make publicly available information about the location, format and subject of research data created within the university to other researchers to facilitate sharing and preservation.
- To allow for the audit and reporting on such data to meet the requirements of funding bodies, the university and other third parties.

Some Principles

- **Data entry** work should not be replicated, the system should be able to speak to primary source systems for all data.
- **Data Structures** should be made available in best practice standard formats, readable by humans and machines.
- **The user interface should be readily available** to the researcher at the most appropriate location and not require logging into *yet another system*.

Data Structures

Data Flows and Sources

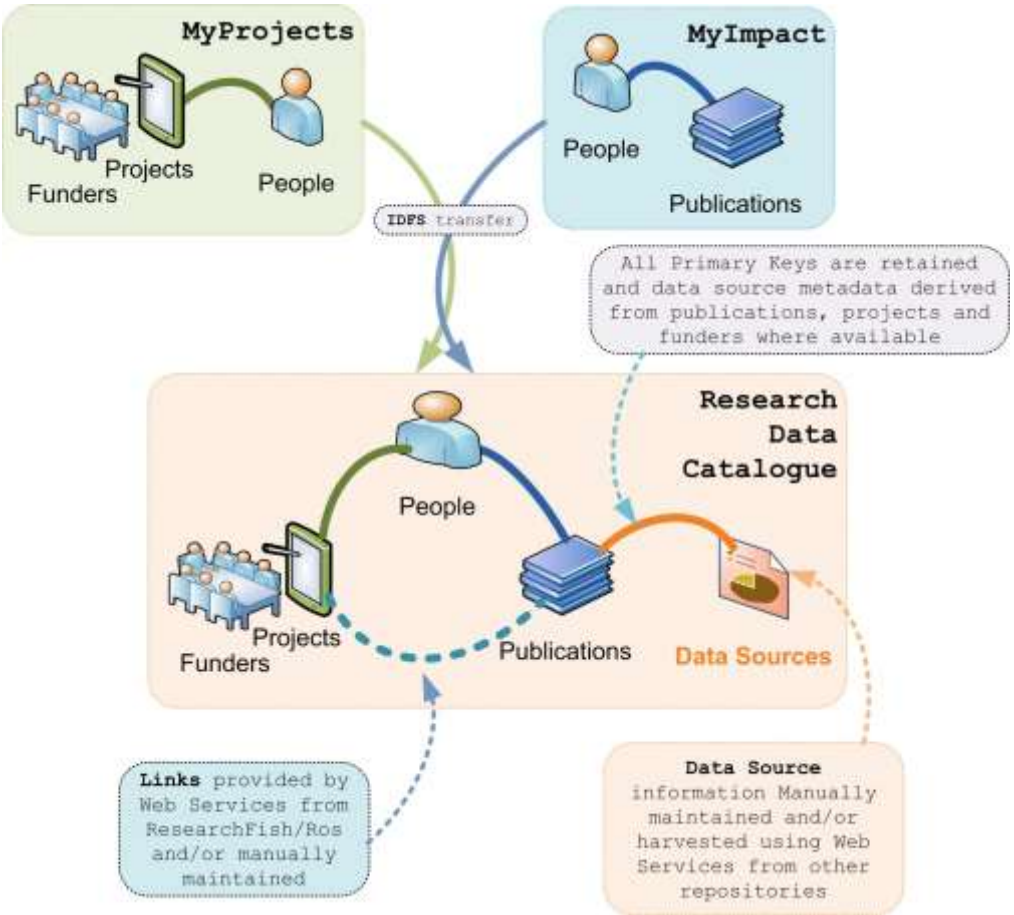
The system **must** allow researchers to adequate **record** information about their **data sources** and to **link** this to their **publications**, **research projects** and **funders**.

In Newcastle University, these data are currently held in different systems:

- Publications and Researchers Profiles are held in **MyImpact**
- Projects and Funding bodies are held in **MyProjects**
- Links between these currently exist in external systems such as **ROS, ResearchFish, PubMed**

MyImpact and **MyProjects** data can be obtained via the University's Institutional Data Feed Service (IDFS).

The data catalogue will store information about the data, but will **not** be used to **store** it, since it recognises that many and various repositories exist. None digital data sources (filing cabinets, audio tapes) and disconnected data sources (external hard drives and legacy disks) therefore remain in scope for the purposes of discovery at least. The system **should** allow researchers to indicate the location of their data sources in a manner which may allow the data catalogue to automatically **harvest metadata**.



Data Storage and Format

Publication and Project primary keys should be retained from the source systems and used as primary keys by this system if possible.

The Primary key for People is their **nid** – the benefits of this are that it is short, but also familiar to the user and links with our authentication system (Shibboleth).

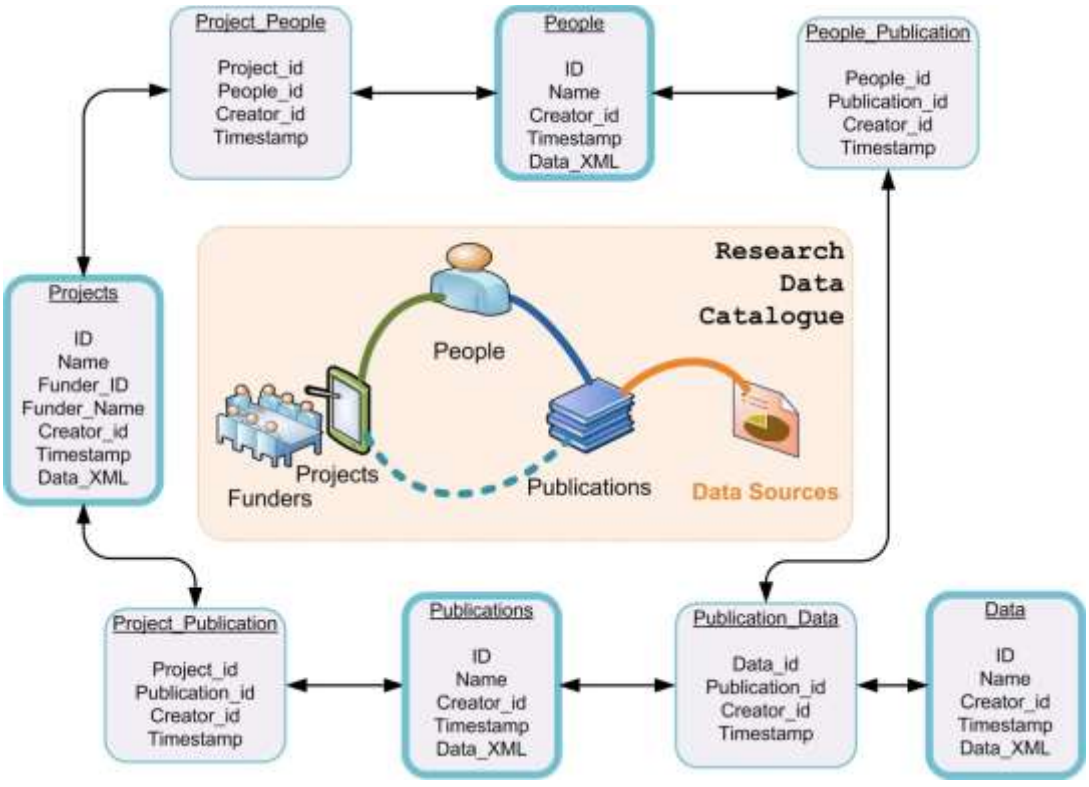
A relational database structure will be used to store the links between objects.

Object metadata should be stored in the database as XML, JSON and/or other serialized forms so that should the metadata need expanding, the database does not need to be reconfigured. This will also mean that the data in the system is stored as ready to publish requiring minimum server scripting.

Dublin Core naming conventions should be used to enable ease of understanding and interrogation.

Since many of the objects in the system will be of interest to more than one editor, each will contain references to the system or person who created or made each change, the time and the nature of the change. All versions are stored.

Database Structure:



Note: In additional to or instead of Data_XML, a Data_JSON or another serialization may also be used depending on the convenience of the server side language in use, provided it can encode quickly to and from XML and JSON.

Data Source Fields:

The fields for publications and projects are as of their parent systems. For the data source the following fields will be used, being derived from their parent publication where possible.

Field	Format	FieldName	Element URI (Standard)	Term URI (Refined)
Location	Free Text	Coverage / Location	http://purl.org/dc/elements/1.1/coverage	http://purl.org/dc/terms/Location
Creator	nid	Creator	http://purl.org/dc/elements/1.1/creator	http://purl.org/dc/terms/creator
Submission Date	Date/Time	Date Submitted	http://purl.org/dc/elements/1.1/date	http://purl.org/dc/terms/dateSubmitted
Resource Created Date	Timestamp		http://purl.org/dc/elements/1.1/date	http://purl.org/dc/terms/created
Last Access Date	Free Text		http://purl.org/dc/elements/1.1/date	
Description	Free Text	Description	http://purl.org/dc/elements/1.1/description	http://purl.org/dc/terms/description
Format	Free Text	Format	http://purl.org/dc/elements/1.1/format	http://purl.org/dc/terms/format
Size	Free Text	Format	http://purl.org/dc/elements/1.1/format	http://purl.org/dc/terms/SizeOrDuration
Unique Reference		Identifier	http://purl.org/dc/elements/1.1/identifier	http://purl.org/dc/terms/identifier
Terms and Conditions for Access	Free Text	Rights / Access Rights	http://purl.org/dc/elements/1.1/publisher	http://purl.org/dc/terms/publisher
Publication	PubID	relation / isReferencedBy	http://purl.org/dc/elements/1.1/relation	http://purl.org/dc/terms/isReferencedBy
Funder/Owner	Free Text	Rights / Rights Holder	http://purl.org/dc/elements/1.1/rights	http://purl.org/dc/terms/rightsHolder
Terms and Conditions for Use		Rights / Licence	http://purl.org/dc/elements/1.1/rights	http://purl.org/dc/terms/accessRights
Subject of Data	Free Text	Subject	http://purl.org/dc/elements/1.1/subject	http://purl.org/dc/terms/subject
Distinct Title	Free Text	Title	http://purl.org/dc/elements/1.1/title	http://purl.org/dc/terms/title
Type or Data	Free Text	Type	http://purl.org/dc/elements/1.1/type	http://purl.org/dc/terms/type
Status	List (TBC)			http://purl.org/dc/terms/provenance
Access Mechanism	Free Text	Mediator		http://purl.org/dc/terms/mediator
Cost	Free Text			http://purl.org/dc/terms/mediator

User Interface

A key aspect of the UI for researchers is that the data catalogue ought to be able to exist ***where it is needed***, so the person maintaining their data catalogue can do so when they make a deposit into a repository, when they are adding their profiles in **myimpact**, or while they are managing their project in **myprojects.**

We do not wish to create a new system for people to log into.

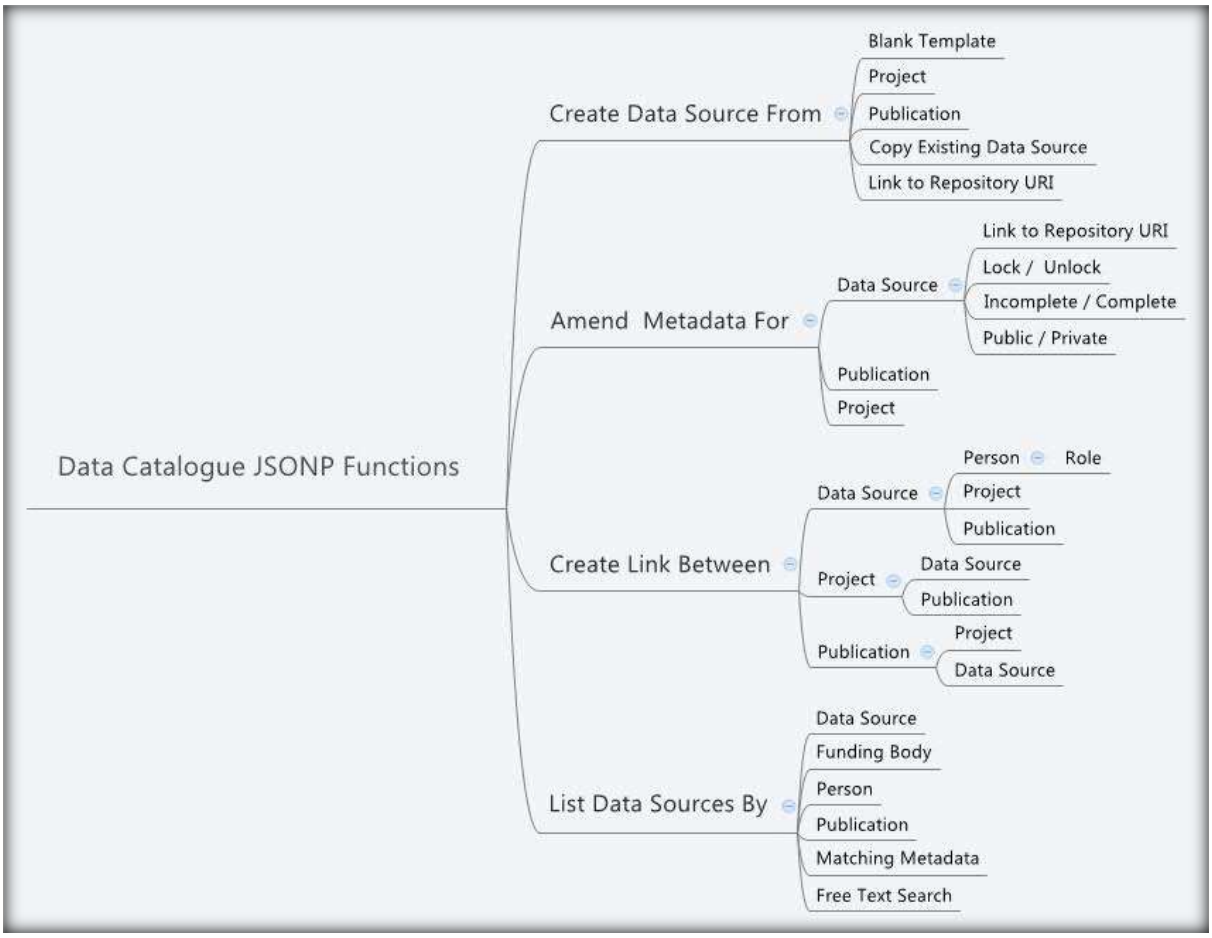
Ideally the user interface is available within the system they are currently using and also populated by that system wherever possible.

JSONP Interfaces

With this in mind, the primary interfaces between the data store and the *outside world* is via JSONP interfaces.

ALL functions of the data catalogue should be obtainable via JSONP, and be authenticated using the same authentication as these other systems – in our case, *Shibboleth*.

Because Shibboleth uses cookie authentication, a person logged into one university system will *also* be logged into the system providing the JSONP (which, unlike regular JSON, allows messages to be sent across different subdomains).



HTML/JQuery/CSS: Web Form Code

The actual user interface can *then be written with no server side code at all*. This will mean that it can easily made available to appear in our business systems (.NET) or any existing repository or CMS (Java in the case of our CMS or Sakai, Python in the case of CKAN, PHP in the case of Wordpress).

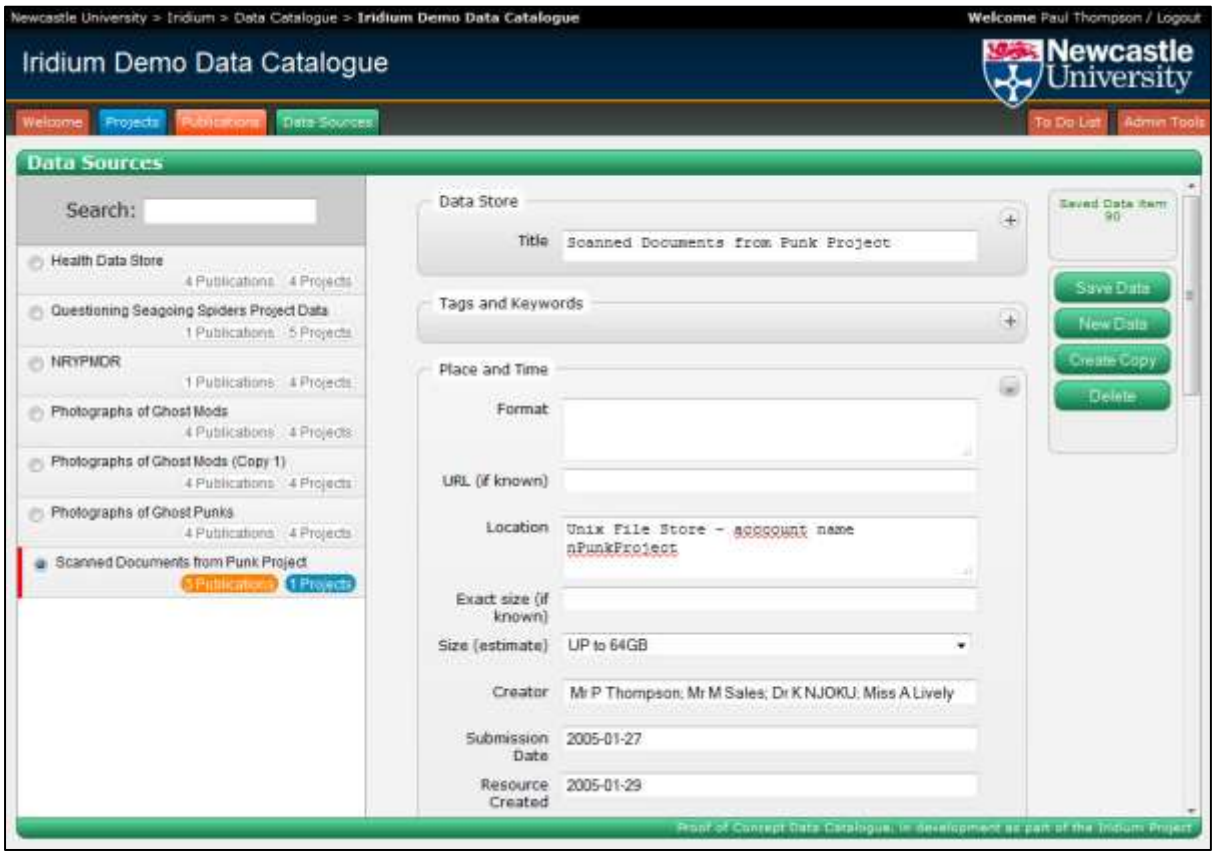
This can be via iFrame or by providing versioned code depending on the needs of the system. The HTML should be written in such a way as to be compatible with the University Style Guide CSS files when embedded in capable systems.

Using JSONP/Jquery to interact with the backend system should enable the UI to meet modern expectations for usability and responsiveness and not require constant web page posts and reloads.

Examples of Interface Design from the Proof of Concept Data Catalogue

As not only the interfaces and fields for Projects, Publications and Data sources, but also the Titles and top level metadata and were likely to be very similar, our proof of concept data catalogue differentiated these using coloured tabs, interfaces and buttons where each of the above has its own assigned colours.

These should be driven from colour schemes found in MyImpact and MyProjects or other appropriate source systems.



Metadata Harvesting from Data Repositories

NOTE: At time of writing it appears that **CKAN** is worth investigating for this role, and where a repository is entirely within the Universities control a product such as **Talend** (in our case via **IDFS**) would be used to bypass much of this process and maintain a database populated with Metadata ready to be linked to.

While specific standards do exist (SWORD, OAI-PMH) they do not appear to have much traction.

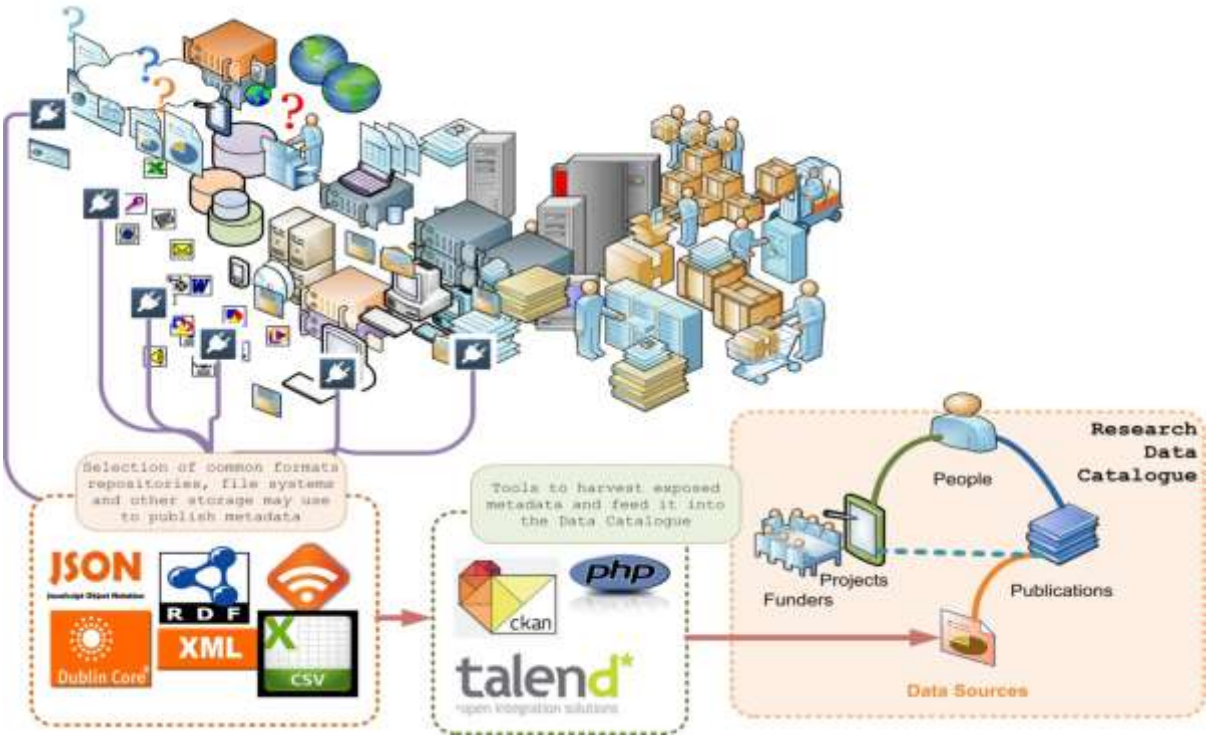
We can though, assume that any repository the university would procure or work with would support at least one of the following, and make it available at a URL.

- **XML** in RSS, ATOM or RDFa formats
- **JSON** or **JSONP** in a format based on the above.
- **Comma** or some other kind of delimited data.
- or at worst (bespoke or legacy systems) we have enough access to the underlying system to write the above.

So, provided the researcher enters a URL into the repository field, the Data Catalogue would be required to have a library of components able to:

- Identify the nature of the repository and the components required below from the URL
- Visit that URL periodically (authenticating if required), using GET, POST, REST or SOAP.
- Obtain the data and transform it into a format compatible with the Data Catalogues storage.
- Merge the data into the record for this Data Source.

Where **Talend** or a similar system such as **Rapid Miner** is not appropriate, and pending the development of **CKAN**, this is likely to be a job for any scripting language capable of handling all of the above formats (PHP has extensive libraries and is most appropriate when used as masking tape)



The Public Data Catalogue

At time of writing *only* the researcher interface exists in proof of concept for the Research Data Catalogue. The specification for the public website, what is shown and what is hidden, and how researcher or funding body gains access to it is likely to be refined based on a number of factors relating to policy acceptance and user testing of the researcher interface.

Assumptions:

- The most basic access to the Data Catalogue may not require an account but *would* require the user to be funnelled through a disclaimer checkbox to provide context to the information within.
- Researchers, Funding Bodies and the Public can be allowed to search the catalogue to a degree where they could see that Data for a particular publication exists and have a facility to request further information.
- The detail – size, format, location, project team, etc, may be restricted to a defined group of people for various reasons. The Data Catalogue would therefore require a field for managing this. This would be authenticated using *Shibboleth* –people can request a Shibboleth compatable ID via ***Protect Network***.
- The quantity of information in the data catalogue will be such that its contents should not be made available to any external search engine and not appear by default as part of the Newcastle University search. The quantity and keyword heavy nature of the data would otherwise risk damaging the universities search profile.

URL Structure:

The data catalogue should contain a unique URL format for each Staff Member, Project, Funding Body, Publication and Data Source, listing the top level information about that item, for example:

- rdc.ncl.ac.uk/staff/paul.thompson
- rdc.ncl.ac.uk/publication/1234
- rdc.ncl.ac.uk/datasource/567890

The latter *may* meet the requirement for a permanent and consistent URI without registering for a DOI service membership, and as such the primary key of a data source and publication must remain constant. Where duplicate or merged records are found they *must* redirect.

Links into these lists could be embedded on university Staff Profiles, Publications pages or from the Research Projects Websites (such direct links would be directed to a disclaimer/context page before allowing the user to browse the data lists).

Web Page Coding:

The webpages should be written in an XML compatible version of HTML5 containing the metadata in both standard HTML meta tags (Dublin Core where possible) *AND* embedded RDFa, enabling *permitted* search engines and spiders to access the read only data without needing access to any of the backend systems or the JSONP interface.

Searching and Reporting on the Research Data Catalogue:

It doesn't seem likely that the data catalogue will exist in any kind of hierarchy that is comprehensible to all possible users of the site (university structure, REF categories for example), and also it's likely that part of the reason for arriving is specifically *because* the user doesn't know what is available, where or how it may have been categorised.

The default search should be a wide ranging free text search.

Tools allowing the user to filter and sort by any available fields should be made available but should never be the default. The user must also be able to sort by date of newly created or updated objects.

The Google Search Appliance

The Google Search Appliance cannot natively read RDFa, but **can** read, display and filter on standard

HTML meta tags which would allow it to search on the title, description or keywords of any other object in the system (Funding Body, Publication, Project, Person).

Examples of this are in place on a number of Newcastle university sites, filtering by Publications or News on our main Web Search uses the GSA and standard HTML meta tags, as does the DECTE project search.

At this point the system contains only test data – prototyping and user testing with real data would be required in order to determine the final specification for the search.

Reporting:

Any searches and filters that can be applied in the above search should be saveable by the user and given a unique URL for bookmarking.

The user should be given the option to receive by email a weekly or monthly digest of any changes to any search/report they have created.

In addition to a human readable search, all searches should be made available as RSS, CSV and RDFa so that tracking systems (belonging to either the University or to third parties such as Research Funding Bodies) can monitor the system.

A final note on Open Access Requirements

Open Access Requirements for publications were not part of the scope for this project, but it seems likely that such requirements would be easy to incorporate in such a system, since the data would be managed by the same people and need to be reported on by the same agencies.

Further Information:

University Systems:	• MyImpact: http://www.ncl.ac.uk/res/resources/myimpact/ • MyProjects: http://www.ncl.ac.uk/res/resources/myprojects
IDFS and Talend:	The Newcastle University **Institutional Data Feed Service** was put in place as part of the JISC IDMAPS project. • Project Site: http://research.ncl.ac.uk/idmaps/ • Service Site: http://www.ncl.ac.uk/itservice/idfs/ • Talend: http://www.talend.com/
Shibboleth:	Shibboleth Authentication is in use on most Newcastle University web applications as part of the JISC IAMSECT Project • Project Site: http://iamsect.ncl.ac.uk/ • Service Site: http://www.ncl.ac.uk/itservice/login-gateway/ • Protect Network: https://app.protectnetwork.org/registration.html
Data Transmission Formats:	• JSON: http://en.wikipedia.org/wiki/JSON • JSONP: http://en.wikipedia.org/wiki/JSONP
Metadata Syntax and Conventions:	• Dublin Core: http://dublincore.org/ • RDFa: http://en.wikipedia.org/wiki/RDFa
Webpage Coding Standards:	• RDFa +HTML5 http://en.wikipedia.org/wiki/RDFa#HTML_5_.2B_RDFa_1.1_example • Google Search Appliance and Meta Tags: https://developers.google.com/search-appliance/documentation/68/xml_reference#request_meta
Google Search Appliance	• University Search (Powered by the GSA) https://my.ncl.ac.uk/search/ • Google Search Appliance: https://developers.google.com/search-appliance/