University of Glasgow | Experimental Particle Physics

GLAS-PPE/2010-05
26th October 2010

School of Physics and Astronomy
Experimental Particle Physics Group
Kelvin Building, University of Glasgow
Glasgow, G12 8QQ, Scotland
Telephone: +44 (0)141 330 2000 Fax: +44 (0)141 330 5881

Towards a Virtual Research Environment for Language and Literature Researchers

M S. Sarwar (1), R.O. Sinnott (1), T. Doherty (1), J.P. Watt (1).

1 National e-Science Centre, University of Glasgow, Glasgow, G12 8QQ

Email: m.sarwar@nesc.gla.ac.uk or t.doherty@physics.gla.ac.uk

Abstract

Language and literature researchers make use of variety of data resources in order to conduct their day-to-day research. Such resources include dictionaries, thesauri, corpora, images, audio and video collections. These resources are typically distributed, and comprise non-interoperable repositories of data that are often license protected. In this context, researchers conduct their research through direct access to individual resources. This form of research is non-scalable, time consuming and often frustrating to the researchers. The JISC funded project Enhancing Repositories for Language and Literature Researchers (ENROLLER, http://www.gla.ac.uk/enroller/) aims to address by provision of an interactive, research infrastructure providing seamless access to major language and literature repositories. This paper describes this infrastructure and the services that have been developed to overcome the issues in access and use of digital resources in humanities. In particular, we describe how high performance computing facilities including the UK e-Science National Grid Service (NGS, http://www.ngs.ac.uk) have been exploited to support advanced, bulk search capabilities, implemented using Google's MapReduce algorithm. We also describe our experiences in the use of the resource brokering Workload Management System (WMS) and the Virtual Organization Membership Service (VOMS) solutions in this space. Finally we outline the experiences from the arts and humanities community on the usage of this infrastructure.

Towards a Virtual Research Environment for Language and Literature Researchers

[1]Muhammad S. Sarwar, [2]Richard O. Sinnott, [1]T. Doherty, [1]J.Watt
[1]National e-Science Centre,
University of Glasgow
Glasgow, UK
m.sarwar@nesc.gla.ac.uk

[2]Melbourne eResearch Group
University of Melbourne
Melbourne, Australia
rsinnott@unimelb.edu.au

Abstract— **Language and literature researchers make use of variety of data resources in order to conduct their day-to-day research. Such resources include dictionaries, thesauri, corpora, images, audio and video collections. These resources are typically distributed, and comprise non-interoperable repositories of data that are often license protected. In this context, researchers conduct their research through direct access to individual resources. This form of research is non-scalable, time consuming and often frustrating to the researchers. The JISC funded project Enhancing Repositories for Language and Literature Researchers (ENROLLER, http://www.gla.ac.uk/enroller/) aims to address by provision of an interactive, research infrastructure providing seamless access to major language and literature repositories. This paper describes this infrastructure and the services that have been developed to overcome the issues in access and use of digital resources in humanities. In particular, we describe how high performance computing facilities including the UK e-Science National Grid Service (NGS, http://www.ngs.ac.uk) have been exploited to support advanced, bulk search capabilities, implemented using Google's MapReduce algorithm. We also describe our experiences in the use of the resource brokering Workload Management System (WMS) and the Virtual Organization Membership Service (VOMS) solutions in this space. Finally we outline the experiences from the arts and humanities community on the usage of this infrastructure.**

Keywords– Humanities, Language and Literature, MapReduce, HPC, Grid.

I. INTRODUCTION

Consider a scenario where a researcher wants to search for a word, say '*canny*', in the dictionary to find its meaning; in a thesaurus to look up associated concepts and categories it is found and used in, and in a corpus of work to find the documents containing it. Researchers may also want to see the concordances (context where the term was used) and word frequency of the word in each found document and undertake further analysis. The ability to save the different result sets and analysis of those results for later comparison between many different resultant data sets and with different researchers is compelling to this community. This scenario becomes especially interesting and challenging when multiple dictionaries, thesauri and text corpora need to be cross-searched or differences between the textual resources exist. For example, searching for the word 'canny' in the Oxford English Dictionary (OED) [1], Scottish National Dictionary (SND) [2] and Dictionary of Older Scottish Tongue (DOST) [3] at the same time will have slightly different results on the definition of the term. When compared with other resources such as the Historical Thesaurus of English (HTE) [4] to look up the related concepts and categories and/or the Scottish Corpus of Text and Speech (SCOTS) [5] and/or the Newcastle Electronic Corpus of Tyneside English (NECTE) [6] the multitude of definitions and their historical context is especially challenging to establish. The problem scales further if the researcher decides to search for multiple, possibly hundreds, of words at once and do all of the mentioned tasks. Currently, all of the language and literature resources provide scholars with independent resources. Licensing access to multiple resources is commonly required and the end user researchers are left with traditional internet hopping based research. An interactive research infrastructure that brings together all of the different provider's data sets together in a seamless and secure environment exploiting high performance computing infrastructures such as National Grid Service (NGS - http://www.ngs.ac.uk) and ScotGrid (http://www.scotgrid.ac.uk) where needed, is thus highly desirable.

The ENROLLER project [7] which began in April 2009 has been tasked with providing such a capability through the establishment and support of a targeted Virtual Research Environment (VRE) implementing secure and seam less data integration and information retrieval system for language and literature scholars.

This paper describes the challenges in implementing the VRE for language and literature researchers and the solutions put together thus far. Section II describes the background and data collections involved in the project. Section III describes the VRE and its overarching requirements. Section IV describes the design of the various components that make up the system. Section V explains the implementation details and outlines the problems faced and solutions implemented during the course of the work. Section VI presents typical use cases of the system. Section VII highlights the feedback of the work collected from the language and literature community. Finally section VIII draws conclusions on the work as a whole and areas of future work.

II. DATA SETS

The ENROLLER project is currently working with numerous major data sets from a variety of data providers. These include:

A. The Historical Thesaurus of English (HTE, http://libra.englang.arts.gla.ac.uk/historicalthesaurus/aboutproject.html)

The HTE contains more than 750,000 words from Old English (c700 A.D.) to the present. HTE has been published by the Oxford University Press since 2009 and offers a new and significant development in the historical language studies. HTE data is currently available in XML format.

B. Scottish Corpus of Text and Speech (SCOTS – www.scottishcorpus.ac.uk)

The Engineering and Physical Sciences Research Council (EPSRC, www.epsrc.ac.uk) and the Arts and Humanities Research Council (AHRC, www.ahrc.ac.uk) funded SCOTS resource offers a collection of text and audio files covering a period from 1945 to present. The SCOTS corpus is currently available in a Text Encoding Initiative (TEI, www.tei-c.org) -compliant XML format. Data can also be made available through a PostgreSql relational database.

C. Dictionary of Scots Language (DSL – www.dsl.ac.uk/dsl)

The AHRC funded DSL resource encompasses two major Scottish language dictionaries The Scottish National Dictionary (SND) and The Dictionary of Older Scottish Tongue (DOST). DSL data is currently available in XML format. SLD (Scottish Language Dictionaries) hosts the data on their servers in Edinburgh.

D. Newcastle Electronic Corpus of Tyneside English (NECTE - www.ncl.ac.uk/necte)

The AHRC funded NECTE is a corpus of dialect speech from Tyneside in Northeast England. This corpus aggregates the work of two existing corpora, Tyneside Linguistic Survey (TLS) created in late 1960s and Phonological Variation and Change in Contemporary Spoken English (PVC) created in 1994. NECTE corpus is encoded in TEI-compliant XML format. The encoded data is available in four different formats: audio, orthogonal text, phonetic and parts of speech tagged.

E. Corpus of Modern Scottish Writing (CMSW – www.scottishcorpus.ac.uk/cmsw/)

The EPSRC and AHRC funded CMSW is a collection of letters (mostly texts and images) from the period 1700 A.D to 1945 A.D. (This is regarded as ‘modern’ writing to the language and literature community).

F. Oxford English Dictionary (OED – www.oed.com)

The Oxford English Dictionary (OED – www.oed.com) is a commercial resource published by Oxford University Press and is widely regarded as the primary authority on the current and historic version of the English language vocabulary.

G. Hansard Collection

The Hansard Collection is a collection of transcribed texts of UK’s parliamentary speeches from period 1935 to 2010. The Hansard data is available in XML format.

All of these data resources collectively represent significant independent investments and efforts in capturing and cataloguing the history of the English and Scots language.

III. VIRTUAL RESEARCH ENVIRONMENT

A VRE is generally regarded as an online environment offering a set of tools aimed at providing a collaborative research environment for researchers that may be geographically dispersed. Typically this involves offering a portfolio of services and data sets through a single, uniform web portal. Successful VREs will typically minimize the level of detail required by end user researchers in the back end processes in interacting with services and dealing with issues such as heterogeneity of data resources.

The requirements of the VRE for this project can be broadly divided into three major categories:

A. Computational Requirements

1) Parsing and Indexing

Before any information can be extracted from the data collections, where feasible, data needs to be extracted from the collections. Extracted data then needs to be parsed and indexed. Since every collection follows a different approach of encoding the data, different approaches to data extraction, parsing and indexing are required targeted to the remote data provider data models.

The VRE has been designed to be extensible and allow incorporation of other data sets, and indeed for individual researchers to upload their own data sets of interest to the community. Processes to automate the indexing of uploaded collections are thus highly desirable.

2) Information Retrieval

Executing simple word, multiple word and phrase queries on the indexed collections are a basic requirement of the project. Queries should be executed against any number of available and selected collections. Furthermore being able to perform cross-collection searches on the indexed collections is an essential requirement to this community – since this is one of the primary benefits of having a VRE.

3) Grid-based Information Retrieval

For many queries that the researchers wish to run – especially bulk based querying, it is essential to exploit high performance computing resources. As such, an advanced grid-based search facility is required. It is also desired that the system be able to execute complex and computationally intensive linguistic interactions over the grid.

4) Linguistic Analysis Tools

Enabling the researchers to be able to perform the linguistic analysis on the search results obtained from multiple providers requires development and deployment of linguistic analysis tools such as concordance, frequency analysis and collocation clouds into the VRE, i.e. offering the one stop shop for the language and literature research community.

B. Security Requirements

1) Authentication and Authorization

The VRE should support seamless access to multiple data resources. The Grid tenet of single sign-on is thus highly desirable. This should overcome the need for creating multiple data provider specific username and password combinations. Furthermore, it is essential that individual users should be exposed to the associated Grid technologies to the minimal extent possible, i.e. having them acquire and maintain their own X.509 certificates should be avoided.

2) Communication

Secure communication channels are very important to this community, since they are often dealing with data sets that have been collected over many decades and have considerable intellectual property. This should prevent unwanted eavesdropping and avoid transfer of confidential information such as usernames and passwords to and from data providers.

C. Usability Requirements

A well-designed easy-to-use search system providing secure and seamless access to the distributed data collections is essential. The user interface of the search system is key. Complex interfaces that require degrees of IT '*savvyness*' were to be avoided. The interface itself should ideally provide user intuitive options and engage the community directly. In this regard, personalization is an important feature to this community. Thus users of the VRE should be able to personalize their home pages and be able to perform collaborative research by being able to share the results of their individual researches.

All of these factors have been taken into the design and development of the ENROLLER VRE and associated software platform.

IV. DESIGN

A. System Architecture

The system architecture has adopted an n-tier architecture. The system has been broadly divided into four distinct tiers from top to bottom. At the top is the presentation layer, which provides the user interface to the system. The second tier is the messaging layer which implements methods to communicate with business logic and data access components. Business logic and data access components form the third tier of the system. Business logic components implement the processes and workflow activities of the system. Data access components interact with underlying persistent data stores. The third tier also contains a set of web and grid services that interact with distributed data and computational resources, such as accessing the Oxford English Dictionary (OED) and NGS. Persistent data stores form the fourth layer of the system. Figure 1 shows the high level layering architecture of the system.

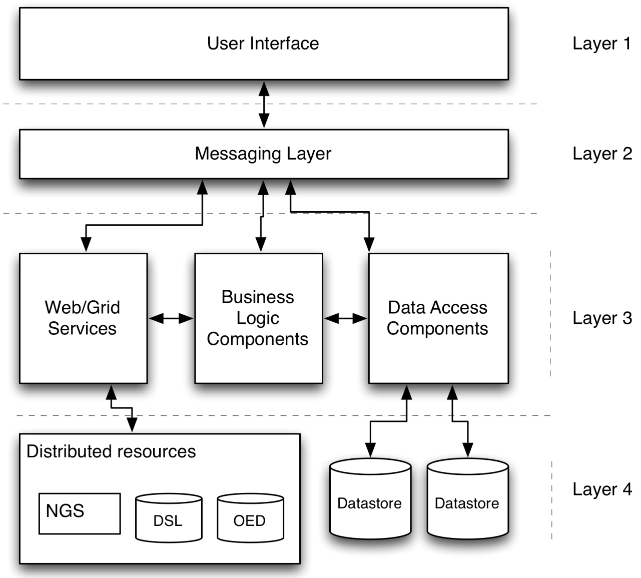


Figure 1: System Architecture

The business logic and data access components are responsible for data extraction, parsing and indexing. Information retrieval, transformation and application of linguistic analysis algorithms are also performed by these components. Figure 2 shows the flow of activities of parsing and indexing components. Figure 3 shows the flow of activities of information retrieval, transformation and language analysis components that the system currently supports.

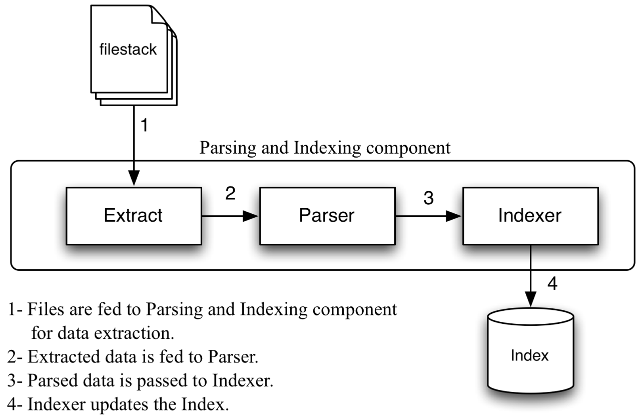
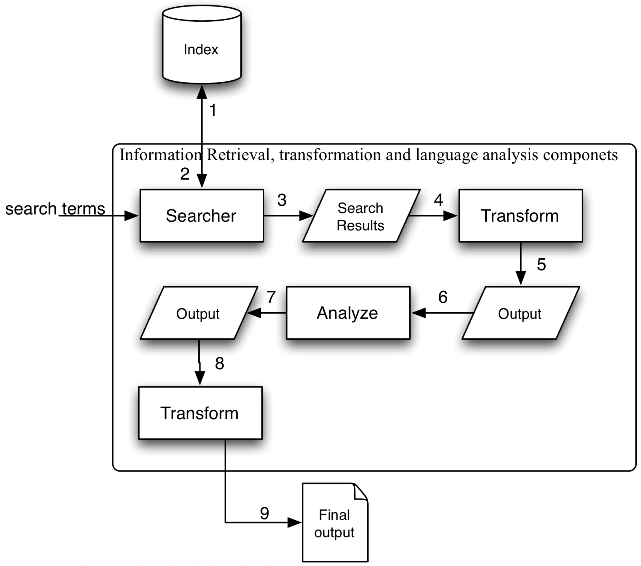

filestack
1
Parsing and Indexing component
Extract
2
Parser
3
Indexer
4
Index
1- Files are fed to Parsing and Indexing component for data extraction.
2- Extracted data is fed to Parser.
3- Parsed data is passed to Indexer.
4- Indexer updates the Index.

Figure 2: Parsing and Indexing Components

It is noted that the business logic and data access components have been implemented as standalone plug-and-play software components to increase the reusability and simplify the maintenance of the system.

B. Authentication and Authorization

The Internet2 Shibboleth [8] framework has been used to provide user-oriented secure access to the portal. This eliminates the need for users to create usernames and



1,2- Searcher searches the index for the 'search terms'.
3- Search results are produced.
4- Search results are fed to transformation module.
5- Transformation module produces the transformed output.
6- Output is fed to Language Analysis module.
7- Analyze module produces output.
8- Output is sent for transformation.
9- Final output is written to a file.

Figure 3: Information Retrieval, Analysis and Transformation

passwords to login to the portal. Instead users are taken to their institutional homes for authentication and once they are authenticated they are brought back to the portal and based upon individual's authorization attributes are allowed to use the portal features. This security-oriented personalization of portal contents exploits software capabilities from the SPAM-GP project and is described in [9]. Figure 4 shows the flow of activities of the whole process.

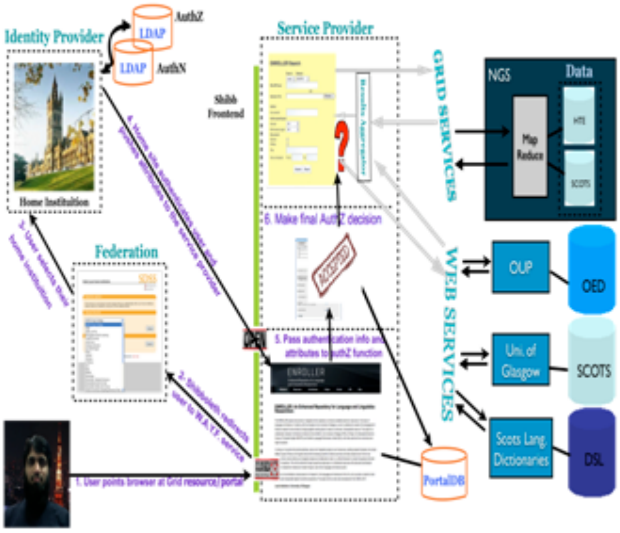

Identity Provider
LDAP
AuthZ
LDAP
AuthN
Home Institution
Service Provider
Shibb Frontend
Results Aggregator
GRID SERVICES
NGS
Data
Map Reduce
HTE
SCOTS
4. Home site authenticates user and provides attributes to the service provider
3. User selects their home institution
Federation
SDSS
6. Make final AuthZ decision
ACCEPTED
WEB SERVICES
OUP
OED
5. Pass authentication info and attributes to AuthZ function
OPEN
ENROLLER
Uni. of Glasgow
SCOTS
2. Shibboleth redirects user to W.A.Y.F. service
Scots Lang. Dictionaries
DSL
PortalDB
1. User points browser at Grid resource/ portal

Figure 4: Shibboleth-based LogIn to Portal

It is worth noting here that the signed SAML assertion that is sent back from the Identity Provider includes encrypted information that is subsequently used as part of the process of creating a secure session in the portal. Ideally this would include sufficient information to dynamically create an X509 proxy credential for the particular user. This extra information is not typically made available through the UK Access Management Federation (www.ukfederation.org.uk) however. We are working closely with the UK Federation and the NGS in this regard to explore solutions that allow direct translation of SAML assertions, to create the associated proxy credentials - building on results of the SARoNGS project (http://cts.ngs.ac.uk).

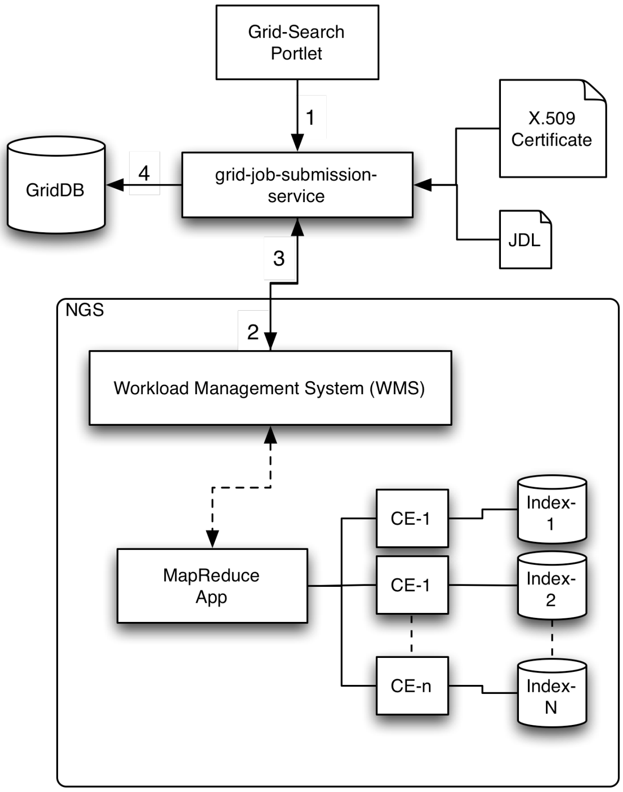
C. Bulk Searching Over the Grid

To support larger scale searches where several thousand query terms are possible and need to be searched across multiple large scale data resources (although we note that this does not include license protected data resources), the project is exploiting the NGS. In particular the project is exploiting the Virtual Organisation Membership Service (VOMS) solution [10] in accessing the NGS where pooled ENROLLER accounts are used by researchers accessing these resources through a targeted project portal. This includes use and exploitation of the Workload Management System (WMS) [11] to provide resource broking-based job scheduling across all of the NGS nodes. This job scheduling is targeted currently to supporting large-scale searching based upon the Google's MapReduce [12] application. A set of job-submission and status-monitoring services support the job submission, status monitoring and output retrieval. All of these capabilities have been iteratively developed in close co-ordination with the language and literature community.

1) Job Submission

A job submission service provides the facility for job submission directly from the portal. Once the user submits a *grid-search* request, the *grid-job-submission-service* is invoked and parameters for the grid-search are provided. Parameters for the grid-search include the search-terms themselves (or a file including these search terms), the user_id, and an encrypted version of the MyProxy username and password. The Grid-job-submission-service decrypts this username and password and subsequently contacts the MyProxy [13] server to retrieve the necessary proxy certificate of the user who initiated the job submission process using the provided username/password information. It is worth noting that the returned credentials already include the VOMS attribute certificate extensions (stating what role and privileges the end user has as part of the ENROLLER VO) as part of the X509 proxy certificate.

A job-submission-script has been written to launch the MapReduce Java application on the grid. To support this, the Grid-job-submission-service generates the Job Specification Description Language (JSDL) [14] for the job. The job-submission-script itself is also staged to the grid. Once all the configurations are completed successfully, the NGS WMS service is contacted and a request for job submission is made. The WMS automatically matches the job requirements with the available pool of resources available to the ENROLLER VO and schedules the job for execution accordingly. Upon successful scheduling of job, a job-id is returned. This job-id is then stored in a database for general job tracking and updates. It is noted that this job-id can have numerous sub-jobs associated with it, i.e. when bulk jobs are supported. In this case the WMS service can schedule jobs across multiple distributed NGS resources. Figure 5 shows general the job submission process.



1- Portlet invokes the gLite-WMS-Client-App to submit the job to the grid.
2- gLite-WMS-Client-App uses the user's X.509 certificate and JDL for the job and submits the job to grid through WMS.
3- WMS returns the JobId.
4- JobId is stored in a database.

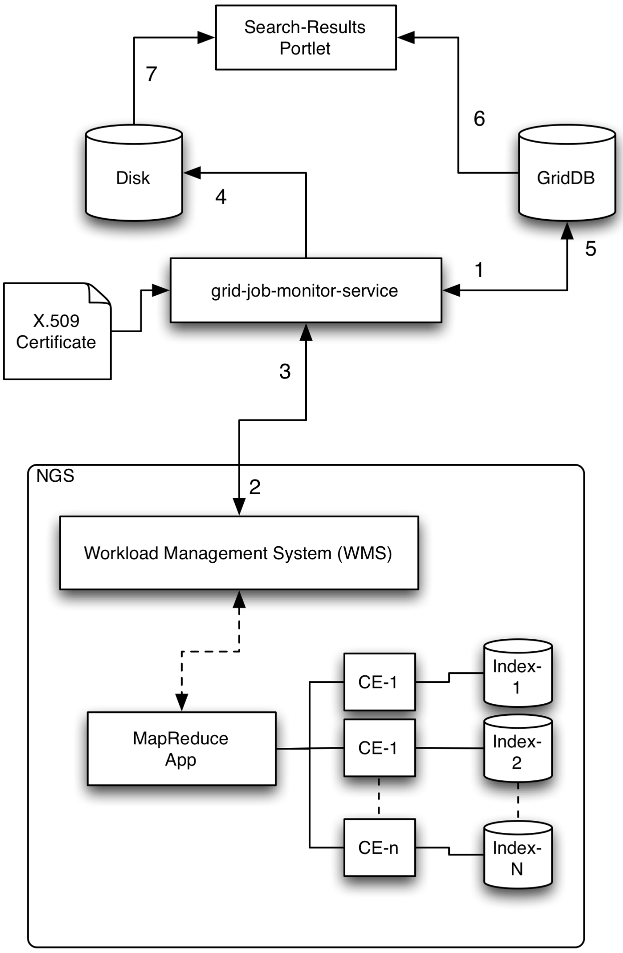
Figure 5: Grid-based Job Submission Process

a) Realisation of Grid-based Job Submission

When a job is successfully scheduled for execution by the WMS, the staged job-submission-script is initiated. The job-submission-script sets up the necessary paths to indexes and other necessary libraries. After setting up the paths to the indexes and libraries, the multi-threaded distributed MapReduce service, which itself implements Google's MapReduce algorithm, is started. Upon successful completion, the application outputs a file containing the search results. Once the job has finished execution WMS clears the job from memory and makes the output available for consumption.

2) Job Status Monitoring and Output retrieval

The job-status-monitoring-service is started as soon as a job is submitted to the grid. The job-status-monitoring-service fetches the job-id of the job from a database and keeps checking for the status of the job. When the job has finished executing the output is copied back to local server and status of job is updated in the database. Also the location of output files is inserted in the database. Once the output of a job becomes ready a download link is provided to the user to download the job output. Job output is formatted in XML format for interoperability reasons. Figure 6 shows the flow of activities through the system.



1- Get the JobId from database
2- Check status of the Job
3- Get job state
4- Save the output to disk when job is done.
5- Update the job status in database.
6- Fetch the 'done' jobs from database.
7- Get the output from Disk and make it available on search-results-portlet.

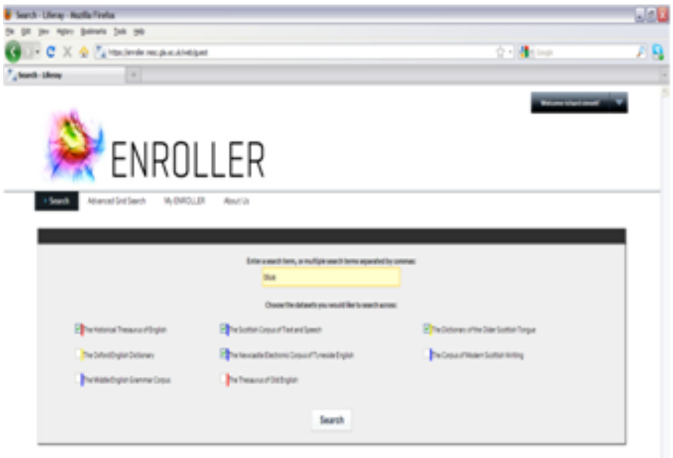
Figure 6: Grid Job Status Monitoring and Output Retrieval Process

D. Issues

In the realization of this system, numerous issues and challenges have arisen. VOMS proxy credentials are generated and remain valid only for a period of 12 hours that means if a job is going to take more than 12 hours to run, the proxy credentials are going to run out of time and the job will be terminated. One solution to this problem is to re-generate the proxy credentials if they are near to run out of time and the job is still running. In order to implement this solution, MyProxy username and password of the user needs to be saved in an encrypted format in a database for subsequent re-generation of the proxy credentials. This solution is still to be implemented. An alternative to this of course is to create a proxy credential with a much longer time period than required. This is a security danger however and has not been adopted. It is noted that software from the Proxy Credential Auditing project (PCA, www.nesc.ac.uk/hub/projects/pca) is exploiting case studies from the ENROLLER project to address precisely these kinds of issues.

V. IMPLEMENTATION

In the course of the ENROLLER project, we have largely adopted an agile approach to software development based on rapid prototyping and component-based software engineering for the current project artifacts. Maven has been used to manage the lifecycle of the project. For the user interface and VRE itself we have adopted the Liferay portal [15] which provides a platform for creating both the user interface components of the VRE and tools to support the back end provisioning and support of services. Liferay itself is a JSR286 [16] compliant platform that makes it a perfect candidate for creating and deploying JSR268 compliant portlets. Ajax [17] has also been used to develop interactive web 2.0 compliant user interfaces. All the communication is done over https to keep the flow of information secure. Figure 7 shows the current (basic) user interface of the search system. A more advanced Grid search interface is also available that supports larger scale searches including uploading of files with relevant search terms. .



Enter a search term, or multiple search terms separated by commas: blue

Choose the datasets you would like to search across:

☑ The Historical Thesaurus of English
☑ The Scottish Corpus of Text and Speech
☑ The Dictionary of the Older Scottish Tongue
□ The Oxford English Dictionary
☑ The Newcastle Electronic Corpus of Tyneside English
□ The Corpus of Modern Scottish Writing
□ The Middle English Grammar Corpus
□ The Thesaurus of Old English

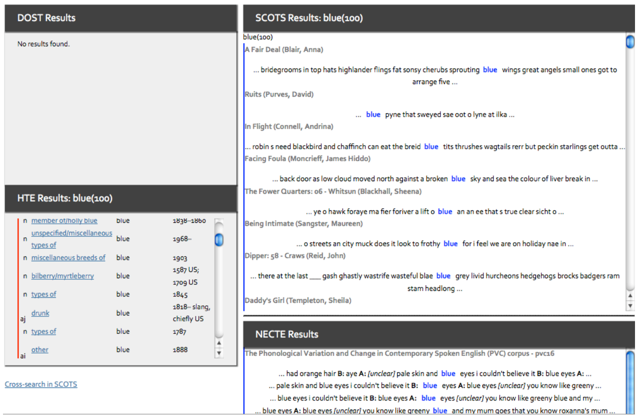
Figure 7: Basic Search Interface

In this interface, the design has been to deliberately offer a Google-like look and feel. The users simply enter the terms they are interested in and the resources they wish the search to run over (through selecting the appropriate boxes). More complex interfaces have also been developed, e.g. to reduce the time period over which the user interested – 18th century for example. The majority of the end users exploit the basic search capability however.

As mentioned previously, participating data collections are heterogeneous in nature therefore devising an identical parsing and indexing algorithm for all resources is not possible. The StAX API [18] is used throughout to parse the XML documents. The JDBC API [19] is used to interact with relational databases. The Lucene API [20] has also been chosen, to index the parsed data. Lucene was selected due to its flexible and powerful indexing and searching capabilities. Moreover since the advanced Grid-based searches require data to be placed over the Grid, Lucene based indexes can be archived and copied over the Grid easily using GridFTP [21]. This practice results in using the same index on local servers and on the Grid and produces identical search results. When simple searches are performed, indexes placed on local servers are searched and when a Grid-based search or workflow execution is invoked, indexes placed on the Grid are used.

A distributed search application based upon the Google's MapReduce algorithm has been written and exploited over the Grid for larger scale searches. This is a multi-threaded application and is responsible for carrying out searches across the data collections available on the Grid. Search results are saved in a file in XML format. The actual job submission service itself uses Globus Toolkit (GT) [22] – a necessary requirement when interacting with facilities such as the NGS.

Figure 8 shows the results of the executing of a search for the term "blue" and the associated results returned from the SCOTS resource, HTE and NECTE resources.



DOST Results

No results found.

HTE Results: blue(100)

n	member of/holly blue	blue	1838–1860
n	unspecified/miscellaneous types of	blue	1968–
n	miscellaneous breeds of	blue	1903
n	bilberry/myrtleberry	blue	1587 US; 1709 US
n	types of	blue	1845
aj	drunk	blue	1818– slang, chiefly US
n	types of	blue	1787
ai	other	blue	1888

Cross-search in SCOTS

SCOTS Results: blue(100)

blue(100)

A Fair Deal (Blair, Anna)

... bridegrooms in top hats highlander flings fat sonsy cherubs sprouting blue wings great angels small ones got to arrange five ...

Ruits (Purves, David)

... blue pyne that sweyed sae oot o lyne at ilka ...

In Flight (Connell, Andrina)

... robin s need blackbird and chaffinch can eat the breid blue tits thrushes wagtails rerr but peckin starlings get outta ...

Facing Foula (Moncrieff, James Hiddo)

... back door as low cloud moved north against a broken blue sky and sea the colour of liver break in ...

The Fower Quarters: 06 - Whitsun (Blackhall, Sheena)

... ye o hawk foraye ma fier foriver a lift o blue an an ee that s true clear sicht o ...

Being Intimate (Sangster, Maureen)

... o streets an city muck does it look to frothy blue for i feel we are on holiday nae in ...

Dipper: 58 - Craws (Reid, John)

... there at the last ___ gash ghastly wastrife wasteful blae blue grey livid hurcheons hedgehogs brocks badgers ram stam headlong ...

Daddy's Girl (Templeton, Sheila)

NECTE Results

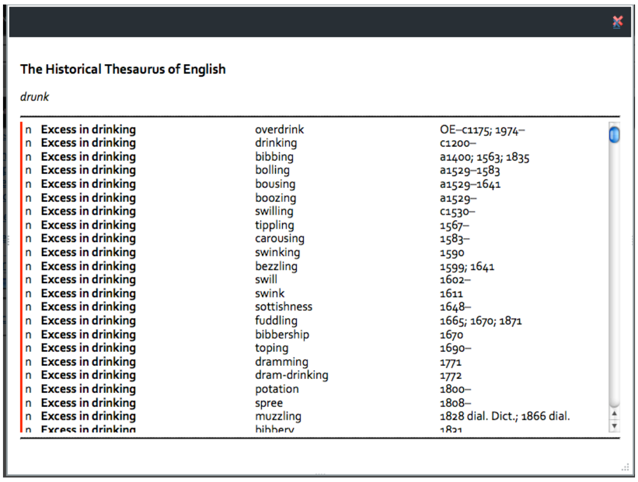
The Phonological Variation and Change in Contemporary Spoken English (PVC) corpus - pvc16

... had orange hair **B:** aye **A:** *[unclear]* pale skin and blue eyes i couldn't believe it **B:** blue eyes **A:** ...
... pale skin and blue eyes i couldn't believe it **B:** blue eyes **A:** blue eyes *[unclear]* you know like greeny ...
... blue eyes i couldn't believe it **B:** blue eyes **A:** blue eyes *[unclear]* you know like greeny blue and my ...
... blue eyes **A:** blue eyes *[unclear]* you know like greeny blue and my mum goes that you know roxanna's mum ...

Figure 8: Results from Basic Searching Using ENROLLER VRE

The results of the HTE are shown in more detail in Figure 9. In particular this shows how multiple variant meanings of the term *blue* have been used throughout the centuries along (as a colour descriptor, for drunkenness, for sadness etc) with the periods of usage for variant of the term. The synonyms of blue when used to mean *drunk* are shown in Figure 9 along with the time and the period of usage.

When a user submits the Grid-based search request (through the advanced search portlet), the search terms from an uploaded CSV file are extracted. A Globus-based Grid-job-submission-service is initiated to run the job on the Grid. The JSDL contents are generated using functions from the jLite



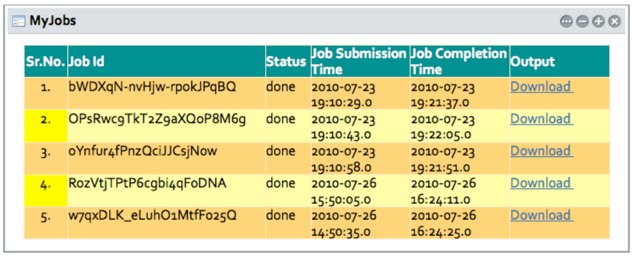
The Historical Thesaurus of English

drunk

n	**Excess in drinking**	overdrink	OE–c1175; 1974–
n	**Excess in drinking**	drinking	c1200–
n	**Excess in drinking**	bibbing	a1400; 1563; 1835
n	**Excess in drinking**	bolling	a1529–1583
n	**Excess in drinking**	bousing	a1529–1641
n	**Excess in drinking**	boozing	a1529–
n	**Excess in drinking**	swilling	c1530–
n	**Excess in drinking**	tippling	1567–
n	**Excess in drinking**	carousing	1583–
n	**Excess in drinking**	swinking	1590
n	**Excess in drinking**	bezzling	1599; 1641
n	**Excess in drinking**	swill	1602–
n	**Excess in drinking**	swink	1611
n	**Excess in drinking**	sottishness	1648–
n	**Excess in drinking**	fuddling	1665; 1670; 1871
n	**Excess in drinking**	bibbership	1670
n	**Excess in drinking**	toping	1690–
n	**Excess in drinking**	dramming	1771
n	**Excess in drinking**	dram-drinking	1772
n	**Excess in drinking**	potation	1800–
n	**Excess in drinking**	spree	1808–
n	**Excess in drinking**	muzzling	1828 dial. Dict.; 1866 dial.
n	**Excess in drinking**	bibbery	1821

Figure 9: HTE Results for term *blue* when used to describe *drunkenness*

API [23] library. The interface to the advanced Grid-search is very similar to the basic search given above with the additional option to upload a file of terms. It is noted that although it was a clear requirement for end users to not to want to know/deal with the intricacies of running jobs on the Grid, they are interested in seeing how their searches run on multiple distributed resources. A job status and monitoring portlet has been developed for this purpose as shown in Figure 10.



MyJobs

Sr.No.	Job Id	Status	Job Submission Time	Job Completion Time	Output
1.	bWDXqN-nvHjw-rpokJPqBQ	done	2010-07-23 19:10:29.0	2010-07-23 19:21:37.0	Download
2.	OPsRwc9TkT2Z9aXQoP8M6g	done	2010-07-23 19:10:43.0	2010-07-23 19:22:05.0	Download
3.	oYnfur4fPnzQciJJCsjNow	done	2010-07-23 19:10:58.0	2010-07-23 19:21:51.0	Download
4.	RozVtjTPtP6cgbi4qFoDNA	done	2010-07-26 15:50:05.0	2010-07-26 16:24:11.0	Download
5.	w7qxDLK_eLuhO1MtfFo25Q	done	2010-07-26 14:50:35.0	2010-07-26 16:24:25.0	Download

Figure 10: Job Status Tracking

VI. USE CASES

This section presents some typical scenarios to understand the interaction between various kinds of services and how the end users have been using the system.

A. Login

A user who wants to use the system, accesses the portal via a web browser. Upon reaching the portal Shibboleth's Where-Are-You-From (WAYF) service intercepts the user and asks for the selection of user's home institution. Once the user has selected their home institution they are redirected to the institution's login page. User enters their username and password for authentication. Once authentication is completed successfully, the user is redirected to the portal where the signed SAML assertion is used to allow them access and build up the portal session. At this point the user's authorization attributes (encoded as part of the eduPersonEntitlement attribute) are loaded into the portal and used to configure the portal contents, i.e. the portlets they are allowed to see/invoke.

B. Simple Search

Users exploit the ENROLLER portal's basic search interface to perform simple word, multiple words and phrase searches across any number of available collections. The search queries are made against the indexed data collections and ultimately returned to the portal. Users are able to download the results as CSV files for their own local use. Work is on-going to support data playgrounds where results can be stored longer term in the VRE and used by collections of researchers.

C. Cross-collection Search

Search results of a simple search can be further used as an input for cross-collection searches. For example, searching for a word '*excellence'* in SND and in HTE produces lists of synonyms. These lists of synonyms can further be used as input to searches of the SCOTS corpus, the NECTE corpus and the Hansard collection. This is the typical scenario used to feed the bulk search service. As before, data can be downloaded locally by researchers and used with their own local analysis tools, e.g. tools used for variant word spellings.

D. Bulk Search

If a researcher wants to search for tens or hundreds of words at once, instead of typing all the words into the query box they can upload a file of search terms. At present this has to be in CSV format. The system will automatically extract the words from this file and search them against the selected collections. Bulk searches are supported over Grid. In this case, the indexed data are distributed over the Grid for rapid searching. As noted only non-licensed data sets can be distributed and used like this.

E. Workflows

At present we provide a workbench where workflows can be manually driven. To explain this, consider the typical scenario where a researcher decides to search for all the relevant entries of the word '*timid'* in HTE and then wants to cross search the search results from HTE in Scottish Corpus to find all the documents that match against each of the input words. They may also wish to find all the concordances for each of the words. Augmenting this scenario to incorporate cycles of interactions where thesaurus search results, corpus search results, concordances for the words are used to define and shape future searches is a key scenario. At each stage the user is able to download the data locally, manipulate it and/or process it to help shape their future searches.

It may well be the case that the definition and enaction of such scenarios could well exploit established workflow environments and associated tools. However, at present the user community has adopted the solutions put forward and are not yet requesting such enhanced capabilities.

VII. Feedback from Community

The project has been specifically organized to be community driven. Email lists for networks of scholars in this field are established and used for updates and community feedback. The project also developed and rolled out a wiki as part of the VRE, however we have found this has made less of an impact than originally hoped/expected.

As part of the work itself, a colloquium was organized in April 2010 where over 30 academics and researchers from various institutions of UK and Europe participated and were shown the system and subsequently allowed to drive the system according to their own research needs and requirements. The overall response from the community was extremely encouraging and all users able to run large scale searches and undertake research that they could not easily do otherwise, i.e. without internet-based hopping from resource to resource. Participants gave numerous useful comments and suggestions for the further development of services and sustainability of this infrastructure in the longer run. We note that this user community also included the data providers themselves. We believe that the success of such efforts demands an inclusive model to help shape the resources and capabilities.

VIII. Conclusions and Future Work

Through the ENROLLER work, researchers in the language and literature domain now have access to large amounts of language and literature data from a single, easy-to-use portal; membership of an international network of scholars; increased knowledge of digital resources, and direct access to a portfolio of analysis tools.

The work is very much on-going however and numerous other challenges remain to be addressed. These include the development of enhanced data playgrounds where researchers can run queries and generate results that can subsequently be used by others as part of their own research or kept longer term for future usage. Data provenance is a key requirement that this community are keen on – knowing that they are dealing with the accurate historical resources and results from those resources.

Automated data deposition and automated indexing of deposited data collections are further items of work that we are also currently looking to support. We note that there are a huge number of researchers who have historically significant digital resources with no place to maintain this long term.

More work on the project as a whole is available at www.gla.ac.uk/enroller with the VRE itself available at https://enroller.nesc.gla.ac.uk.

Acknowledgment

We gratefully acknowledge funding from JISC for the work described in this paper. We also wish to thank all project partners for their input including Jean Anderson, Johanna Green and Marc Alexander. We also thank the network of scholars for their help and support in shaping the ENROLLER VRE.

References

[1] Oxfor English Dictionary Available http://www.oed.com

[2] Scottish National Dictionary Available http://www.dsl.ac.uk

[3] Dictionary of Older Scottish Tongue Available http://www.celtscot.ed.ac.uk/dost/

[4] The Historial Thesaurus of English, Available http://libra.englang.arts.gla.ac.uk/WebThesHTML/homepage.html

[5] The Scottish Corpus, Available http://www.scottishcorpus.ac.uk/

[6] NewCastle Electronic Corpus of Tyneside English, Available http://research.ncl.ac.uk/necte/

[7] ENROLLER Project http://www.gla.ac.uk/enroller/

[8] The Internet2 Shibboleth framework http://shibboleth.internet2.edu

[9] J. Watt, R.O. Sinnott, T. Doherty, J. Jiang, Portal-based Access to Advanced Security Infrastructures, UK e-Science All Hands Meeting conference, Edinburgh, September 2008.

[10] R. Alfieri, R. Cecchini , V. Ciaschini , L. dell'Agnello, A. Frohner, A. Gianoli , K. L ˝ orentey , and F. Spataro "VOMS, an Authorization System for Virtual Organizations" Available https://twiki.cnaf.infn.it/twiki/bin/viewfile/VOMS/WebDocumentation?rev=1;filename=VOMS-Santiago.pdf

[11] Workload Management System http://glite.web.cern.ch/glite/wms/

[12] Jeffrey Dean and Sanjay Ghemawat "MapReduce: Simplified Data Processing on Large Clusters", 2008, Communications of the ACM, 51:107-113

[13] MyProxy Available http://grid.ncsa.illinois.edu/myproxy/

[14] Ali Anjomshoaa, Fred Brisard, Michel Drescher, Donal Fellows, An Ly, Stephen McGough, Darren Pulsipher, Andreas Savva "Job Submission Description Language (JSDL), Specification, Version 1.0", 2005, Available http://www.ogf.org/documents/GFD.56.pdf

[15] Liferay portal Available http://www.liferay.com/

[16] JSR286 Available http://jcp.org/en/jsr/detail?id=268

[17] Ajax Available http://www.ajax.org/#home

[18] StAX api available http://stax.codehaus.org/Home

[19] Java Database Connectivity (JDBC) API, Available http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136101.html

[20] Lucene api Available http://lucene.apache.org/java/docs/index.html

[21] I. Mandrichenko, W. Allcock, T.Perelmutov," GridFTP v2 Protocol Description", 2005, www.ogf.org/documents/GFD.47.pdf

[22] Globus toolkit Available http://www.globus.org/toolkit

[23] jLite Available http://code.google.com/p/jlite/