



## **LarKC**

*The Large Knowledge Collider*

*a platform for large scale integrated reasoning and Web-search*

**FP7 – 215535**

---

# **D7b.3.3b Final Prototype**

---

**Coordinator: Mark A. Greenwood**

**With contributions from: Niraj Aswani, Mark A. Greenwood, Mattias Johansson, Angus Roberts, James McKay, Jon Wakefield, Valegntin Tablan, Ian Roberts, Hamish Cunningham, Paul Brennan**

**Quality Assessor: Bosse Andersson**

**Quality Controller: Angus Roberts**

Document Identifier:	LarKC/2008/D7b.3.3b/V1.0
Class Deliverable:	LarKC EU-IST-2008-215535
Version:	version 1.0
Date:	September 14, 2011
State:	final
Distribution:	public



## EXECUTIVE SUMMARY

The Large Knowledge Collider (LarKC) project is building a platform for scalable reasoning over terabytes of scientific data, using massive distributed incomplete reasoning. One of the use cases is carcinogenesis research. This has two scenarios, as described in D7b1.1a *Requirements summary*. First, improved literature search is required to assist with carcinogenesis reference production (Monographs). Second, literature knowledge mining is required to assist with gene-disease involvement in Genome Wide Association Studies (GWAS).

During the life of the project, we built prototype software that uses LarKC to assist with the GWAS scenario. The final version of this use case software is documented in this report. We give a brief non-technical description of the use case, describe how the software works, and give instructions for use. Brief technical notes on installation and use of data are also provided. Useability and evaluation of the software is described in D7b.3.3a *Version 3 final iteration report*.



## DOCUMENT INFORMATION

<b>IST Project Number</b>	FP7 – 215535	<b>Acronym</b>	LarKC
<b>Full Title</b>	The Large Knowledge Collider: a platform for large scale integrated reasoning and Web-search		
<b>Project URL</b>	<a href="http://www.larkc.eu/">http://www.larkc.eu/</a>		
<b>Document URL</b>			
<b>EU Project Officer</b>	Stefano Bertolo		

<b>Deliverable</b>	<b>Number</b>	7b.3.3b	<b>Title</b>	Final Prototype
<b>Work Package</b>	<b>Number</b>	7b	<b>Title</b>	Carcinogenesis reference production
















<b>Date of Delivery</b>	<b>Contractual</b>	M42	<b>Actual</b>	30-Sep-11
<b>Status</b>	version 1.0		final	<input checked="" type="checkbox"/>
<b>Nature</b>	prototype <input checked="" type="checkbox"/> report <input type="checkbox"/> dissemination <input type="checkbox"/>			
<b>Dissemination Level</b>	public <input checked="" type="checkbox"/> consortium <input type="checkbox"/>			

<b>Authors (Partner)</b>	Mattias Johansson, Paul Brennan, James McKay (all International Agency for Research on Cancer); Jon Wakefield (University of Washington); Niraj Aswani, Mark A. Greenwood, Ian Roberts, Valentin Tablan, Hamish Cunningham, Angus Roberts (all University of Sheffield)			
<b>Resp. Author</b>	Mark A. Greenwood	<b>E-mail</b>	m.greenwood@dcs.shef.ac.uk	
	<b>Partner</b>	University of Sheffield	<b>Phone</b>	+44 (114) 222 1920



<b>Abstract (for dissemination)</b>	<p>Given advances in human genome sequencing, genetic testing, and the availability of samples from large population studies, it is now possible to carry out new types of study on the association between genes and diseases - Genome-Wide Associations Studies. In these, samples are tested from thousands of subjects with the disease in question, and thousands of disease-free controls. Each sample is tested with many hundreds of thousands of gene markers. If a marker is found more frequently in disease samples as opposed to control samples, then perhaps genes close to that marker are associated with the disease. Analysis of raw experimental data uses common statistical models to find the relevance of each marker, and rank them in order of relevance to the disease. Genes close to the top few markers are then studied in more depth. This last bit is expensive, and improving rankings could improve both the efficiency and the economics of the technique. Analysis could be improved if we incorporated knowledge we might already have about genes - prior knowledge. Such knowledge is available e.g. in the vast numbers of research databases and research publications that now exist in the Life Sciences. The software documented here is a prototype version of software to assist with this, produced in the context of the Large Knowledge Collider (LarKC) project.</p>
<b>Keywords</b>	Evaluation, Genome Wide Association Study, GWAS, carcinogenesis, Bayesian statistics, Bayesian False Discovery Probability, Single Nucleotide Polymorphism, SNP, genif



## PROJECT CONSORTIUM INFORMATION

Participant's name	Partner	Contact
Semantic Technology Institute Innsbruck, Universitaet Innsbruck	 	Prof. Dr. Dieter Fensel Semantic Technology Institute (STI), Universitaet Innsbruck, Innsbruck, Austria Email: dieter.fensel@sti-innsbruck.at
AstraZeneca AB		Bosse Andersson AstraZeneca Lund, Sweden Email: bo.h.andersson@astrazeneca.com
CEFRIEL - SOCIETA CONSORTILE A RESPONSABILITA LIMITATA		Emanuele Della Valle CEFRIEL - SOCIETA CONSORTILE A RE- SPONSABILITA LIMITATA Milano, Italy Email: emanuele.dellavalle@cefriel.it
CYCORP, RAZISKOVANJE IN EKSPERI- MENTALNI RAZVOJ D.O.O.		Michael Witbrock CYCORP, RAZISKOVANJE IN EKSPERIMEN- TALNI RAZVOJ D.O.O., Ljubljana, Slovenia Email: witbrock@cyc.com
Höchstleistungsrechenzentrum, Universitaet Stuttgart		Georgina Gallizo Höchstleistungsrechenzentrum, Universitaet Stuttgart Stuttgart, Germany Email : gallizo@hlrs.de
MAX-PLANCK GESELLSCHAFT ZUR FOERDERUNG DER WISSENSCHAFTEN E.V.		Dr. Lael Schooler, Max-Planck-Institut für Bildungsforschung Berlin, Germany Email: schooler@mpib-berlin.mpg.de
Ontotext AD		Atanas Kiryakov, Ontotext Lab, Sofia, Bulgaria Email: naso@ontotext.com
SALTLUX INC.		Kono Kim SALTLUX INC Seoul, Korea Email: kono@saltlux.com
SIEMENS AKTIENGESELLSCHAFT		Dr. Volker Tresp SIEMENS AKTIENGESELLSCHAFT Muenchen, Germany Email: volker.tresp@siemens.com
THE UNIVERSITY OF SHEFFIELD		Prof. Dr. Hamish Cunningham, THE UNIVERSITY OF SHEFFIELD Sheffield, UK Email: h.cunningham@dcs.shef.ac.uk
VRIJE UNIVERSITEIT AMSTERDAM		Prof. Dr. Frank van Harmelen, VRIJE UNIVERSITEIT AMSTERDAM Amsterdam, Netherlands Email: Frank.van.Harmelen@cs.vu.nl
THE INTERNATIONAL WIC INSTI- TUTE, BEIJING UNIVERSITY OF TECHNOLOGY		Prof. Dr. Ning Zhong, THE INTERNATIONAL WIC INSTITUTE Mabeshi, Japan Email: zhong@maebashi-it.ac.jp
INTERNATIONAL AGENCY FOR RE- SEARCH ON CANCER		Dr. Paul Brennan, INTERNATIONAL AGENCY FOR RE- SEARCH ON CANCER Lyon, France Email: brennan@iarc.fr
INFORMATION RETRIEVAL FACILITY		Dr. John Tait, Dr. Paul Brennan, INFORMATION RETRIEVAL FACILITY Vienna, Austria Email: john.tait@ir-facility.org



TECHNICAL UNIVERSITY OF CLUJ-NAPOCA <a href="http://www.utcluj.ro/">http://www.utcluj.ro/</a>	 The logo of the Technical University of Cluj-Napoca, consisting of a stylized 'T' and 'U' in red and grey.	Prof. Dr. Eng. Sergiu Nedevschi TECHNICAL UNIVERSITY OF CLUJ-NAPOCA Cluj-Napoca, Romania E-mail: <a href="mailto:sergiu.nedevschi@cs.utcluj.ro">sergiu.nedevschi@cs.utcluj.ro</a>
SOFTGRESS S.R.L. <a href="http://www.softgress.com/">http://www.softgress.com/</a>	 The logo for Softgress, featuring the word 'Softgress' in white text on a blue rectangular background.	Dr. Ioan Toma SOFTGRESS S.R.L. Cluj-Napoca, Romania E-mail: <a href="mailto:ioan.toma@softgress.com">ioan.toma@softgress.com</a>



# TABLE OF CONTENTS

LIST OF FIGURES	8
1 INTRODUCTION	10
1.1 Introduction . . . . .	10
1.2 Delivery of the software . . . . .	10
1.3 Associated documentation . . . . .	10
1.4 Outline of this report . . . . .	11
2 THE GWAS SERVICE: AN OVERVIEW	12
2.1 Introduction . . . . .	12
2.2 Background . . . . .	12
2.2.1 Traditional techniques for studying gene disease associations . .	12
2.2.2 Genome wide associations studies . . . . .	12
2.2.3 Ranking markers . . . . .	12
2.2.4 Improving analysis . . . . .	13
2.3 Practicals: how does it work? . . . . .	13
3 THE GWAS SERVICE: HOW DOES IT WORK?	17
3.1 GWAS Prototype V1 . . . . .	17
3.2 GWAS Prototype V2 . . . . .	18
3.2.1 Optimization . . . . .	18
3.2.2 Why MIMIR? . . . . .	19
3.2.3 Keywords Suggestions . . . . .	19
3.2.4 Semantic Searches . . . . .	20
3.3 Final GWAS Prototype, V3 . . . . .	21
3.4 Performance Statistics . . . . .	21
4 USING THE PROTOTYPE	22
4.1 Introduction . . . . .	22
4.2 A small example . . . . .	22
4.3 A bigger example . . . . .	23
4.4 Using the prototype . . . . .	23
4.4.1 Single SNP search mode . . . . .	23
4.4.2 Service mode . . . . .	27
5 TECHNICAL DOCUMENTATION	32
5.1 Introduction . . . . .	32
5.2 LarKC plugins and workflow . . . . .	32
5.3 Installation . . . . .	33
5.3.1 Introduction . . . . .	33
5.3.2 MIMIR Incides . . . . .	33
5.3.3 Linked Life Data repository . . . . .	34
5.3.4 UMLS repository . . . . .	34
5.3.5 Random Indexing service . . . . .	34
5.3.6 GWAS application . . . . .	34
5.4 Further documentation . . . . .	35



5.5	Computing BFDP from priors . . . . .	36
6	FUTURE WORK AND CONCLUSION	37
6.1	Future work . . . . .	37
6.2	Conclusion . . . . .	37
	REFERENCES	38
A	APPENDIX - SAMPLE DATA	39
A.0.1	Introduction . . . . .	39
A.0.2	Sample data . . . . .	39



## LIST OF FIGURES

2.1	Gene marker study in lung cancer . . . . .	13
2.2	Gene markers . . . . .	14
2.3	Associated knowledge . . . . .	14
2.4	Associated abstracts . . . . .	15
2.5	Searching for keywords . . . . .	15
2.6	Semantic annotation . . . . .	16
2.7	Relational information . . . . .	16
4.1	Configure a single SNP search. . . . .	24
4.2	Automatic Keyword Aquisition . . . . .	25
4.3	Single SNP Search Results . . . . .	26
4.4	Shows the Progress of an Experiment . . . . .	28
4.5	Configuring an Experiment in the Service Mode . . . . .	29
4.6	Experiment Details . . . . .	30



## LIST OF ABBREVIATIONS

<b>API</b>	Application Programming Interface
<b>B FDP</b>	Bayesian False Discovery Probability
<b>DNA</b>	Deoxyribonucleic Acid
<b>GWAS</b>	Genome Wide Association Study
<b>IARC</b>	International Agency for Research on Cancer
<b>LLD</b>	Linked Life Data
<b>LarKC</b>	The Large knowledge Collider project
<b>RI</b>	Random Indexing
<b>SNP</b>	Single Nucleotide Polymorphism
<b>TFIDF</b>	Term Frequency Inverse Document Frequency
<b>UMLS</b>	Unified Medical Language System
<b>WAR</b>	Web Archive
<b>WHO</b>	World Health Organisation



## 1. Introduction

### 1.1 Introduction

This report accompanies the software delivered as **LarKC deliverable D7b.3.3b, final prototype**, software for LarKC WP7b Carcinogenesis use case. Two previous reports, **D7b.3.1b Version 1 prototype** [1] and **D7b.3.2b, Version 2 prototype** [2], documented versions 1 and 2 of this prototype. This report documents the final version of the prototype. The prototype provides support for data analysis in Genome Wide Association Studies (GWAS).

GWAS use bioprobes (SNPs - gene markers) to look for higher levels of association between genes in a diseased subjects as opposed to controls. The large numbers of markers mean that huge numbers of samples are needed to achieve sufficient statistical power.

Analysis of raw experimental data uses common statistical models to find the relevance of each marker, and to rank them in order of relevance to the disease. Genes close to the top few markers are then studied in more depth. This last bit is expensive, and improving rankings could improve both the efficiency and the economics of the technique. Analysis could be improved if we incorporated knowledge we might already have about genes - prior knowledge. Such knowledge is available e.g. in the vast numbers of research databases and research publications that now exist in the Life Sciences, sometimes known as the data-ome and bibli-ome.

LarKC WP7b aims to apply LarKC technology to this problem, scaling knowledge discovery across the large amounts of biomedical knowledge now encoded in the data- and bibli-ome, and applying it to the millions of data points in a typical GWAS. We have prototyped a technique with the WHO's cancer research unit, IARC, to combine prior knowledge about a gene with experimental data, thus improving statistical power. A web interface has also been built for the prototype. The prototype is use-able for analysis of SNPs from a full GWAS study.

### 1.2 Delivery of the software

The software that this report accompanies has been placed in the LarKC version control repository <sup>1</sup>.

### 1.3 Associated documentation

This report documents the final version of the prototype for the LarKC WP7b Carcinogenesis use case. It is paired with a second report, **D7b.3.3a Version 3 final iteration report** [7], which reports on the use of the prototype, and its evaluation.

---

<sup>1</sup>The SVN URLs for the individual components are given along with the component descriptions later in this deliverable.



## 1.4 Outline of this report

This report starts with an overview of GWAS, in Chapter 2. It gives a high level summary of the GWAS problem and analysis, and describes the approach developed for incorporating prior knowledge into the analysis. It gives background information for users of the software, and puts it in the use case context. This chapter is largely repeated from the report accompanying the previous version of the service, **D7b.3.2b Version 2 prototype** [2].

Chapter 3 goes on to provide more detail on how the software works, and details changes since version 2.

Following this, Chapter 4 gives detailed instructions for using the prototype interface. Instructions are provided as a step-by-step worked example, with annotated screen shots.

Chapter 5 provides pointers and references to more detailed technical information, details of the use of LarKC plugins and workflows, and gives instructions for installing the prototype. Additionally, details of how to calculate SNP rankings from the GWAS service output are given.

Finally, Chapter 6 gives ideas for future work, and concludes the report.



## 2. The GWAS service: an overview

### 2.1 Introduction

This chapter is largely repeated from the report accompanying the previous version of the service, **D7b.3.2b, Version 2 prototype** [2].

This Chapter gives background information, to help the non-expert understand the task that the GWAS prototype is tackling. The below description is aimed to provide pertinent background material to experts from other domains. It is hoped that it will put the software in context, and provide some useful explanation. It is not intended to be a full theoretical treatment of GWAS and the approach embodied in the software, merely to provide a high level description of those parts felt most pertinent to the LarKC project. Further detail and references to other material can be found in the LarKC deliverable [8].

The first section gives background information on the GWAS technique. The second section describes the approach used in the LarKC use case prototype, to assist with the analysis of GWAS data.

### 2.2 Background

#### 2.2.1 Traditional techniques for studying gene disease associations

In the past, studies of gene disease association either concentrated on looking at those genes in particular families susceptible to the disease, or at those genes for which we had some strong hypothesis based on prior knowledge. This is problematic, as the search for a gene is based on the availability of specific family groups, and on the scientist's own preconceptions and biases.

#### 2.2.2 Genome wide associations studies

Given advances in human genome sequencing, genetic testing, and the availability of samples from large population studies, it is now possible to carry out new types of study in the association between genes and diseases. In these, samples are tested from thousands of subjects with the disease in question, and thousands of disease-free controls. Each sample is tested with many hundreds of thousands of gene markers (SNPs). If a marker is found more frequently in disease samples as opposed to control samples, then perhaps genes close to that marker are associated with the disease.

#### 2.2.3 Ranking markers

Of course, analysis of raw experimental data is not quite as simple as that and uses common statistical models to find the relevance of each marker, and rank them in order of relevance to the disease. Genes close to the top few markers are then studied in more depth. This last bit is expensive, and improving rankings could improve both the efficiency and the economics of the technique.

Figure 2.1 shows the results of such a study. The horizontal axis gives position on the human genome. The vertical axis gives relevance (more correctly, significance).

Each dot represents a marker. Those above the threshold line are the ones considered important enough to warrant further investigation. These markers turned out to be clustered near two genes that are now shown to be associated with lung cancer.

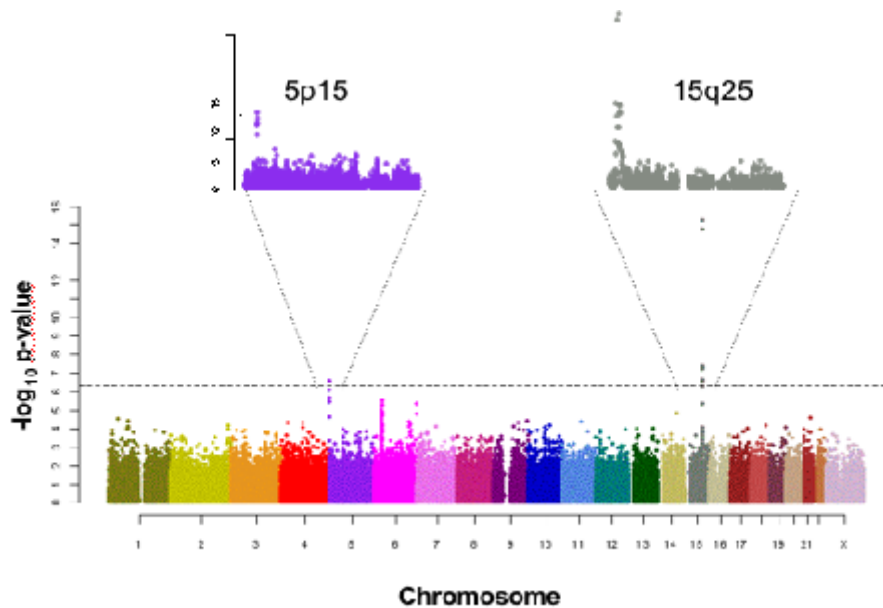


Figure 2.1: Gene marker study in lung cancer

#### 2.2.4 Improving analysis

Analysis could be improved if we incorporated knowledge we might already have about genes - prior knowledge. For example, if we are studying lung cancer, and if we already know that a marker is close to a gene expressed in lung tissue, then we could boost the ranking of that marker.

Full details of the statistics behind this are given in **LarKC Deliverable D7b.3.1a, Version 1 Iteration Report** [1].

### 2.3 Practicals: how does it work?

We first implemented this idea in scripting languages over flat data files. We have now implemented this on LLD for:

- A single data source, Entrez Gene - Simple keyword search of both GeneRIF texts and MEDLINE abstracts - Automated keyword selection using multiple techniques - Semantic search using UMLS concepts - Search using linguistic features - Large sets of markers

The diagrams below explain the technique.

**Where do we get the prior knowledge from?** From the vast numbers of research databases and research publications that now exist in the Life Sciences.

**We calculate a *distance metric* (or *relevance*)** from a set of keywords describing the problem, to each gene marker. We then use this distance to boost the rankings from the gene marker experiments. This is explained below.

**Does it make a difference?** Yes, using the study data shown in the graph above, this new method successfully predicts two markers known to be relevant that were missed by the methods with no prior knowledge. To put this in perspective, if used, it may have saved several hundred of thousand Euros, and found two genes relevant to lung cancer. Of course, these predictions are based on old, retrospective data. We are now running over new, previously unseen data.

### Step 1

**Figure 2.2.** Several genes may be in the region of a gene marker. We retrieve the IDs of all genes within a certain distance of the marker (100 000 base pairs).

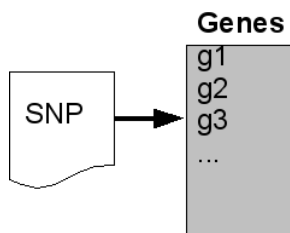


Figure 2.2: Several genes may be in the region of a gene marker

### Step 2

**Figure 2.3.** Each gene has structured knowledge associated with its ID, in several different knowledge sources in LLD.

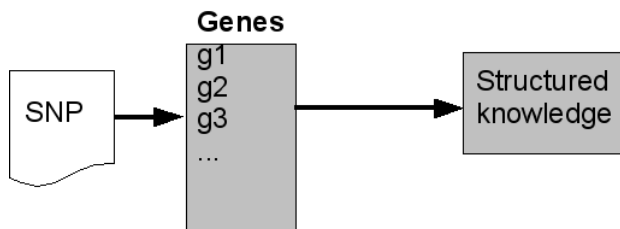


Figure 2.3: Each gene will have structured knowledge associated with it

### Step 3

**Figure 2.4.** Some of these knowledge sources also contain references to research paper abstracts of relevance to the gene.

### Keyword search: step 4

**Figure 2.5.** We collect a set of keywords of relevance to the disease, either chosen by a domain expert, or generated automatically from data in LLD. E.g. for lung cancer, these might be words such as "lung", "cancer", "tobacco". We search for these keywords in the abstracts and calculate a distance from the gene marker to the keyword set, based on the presence or absence of these keywords.

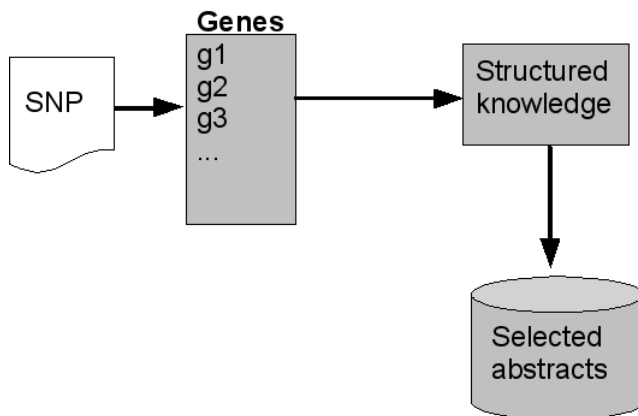


Figure 2.4: There may be research paper abstracts linked to this structured knowledge

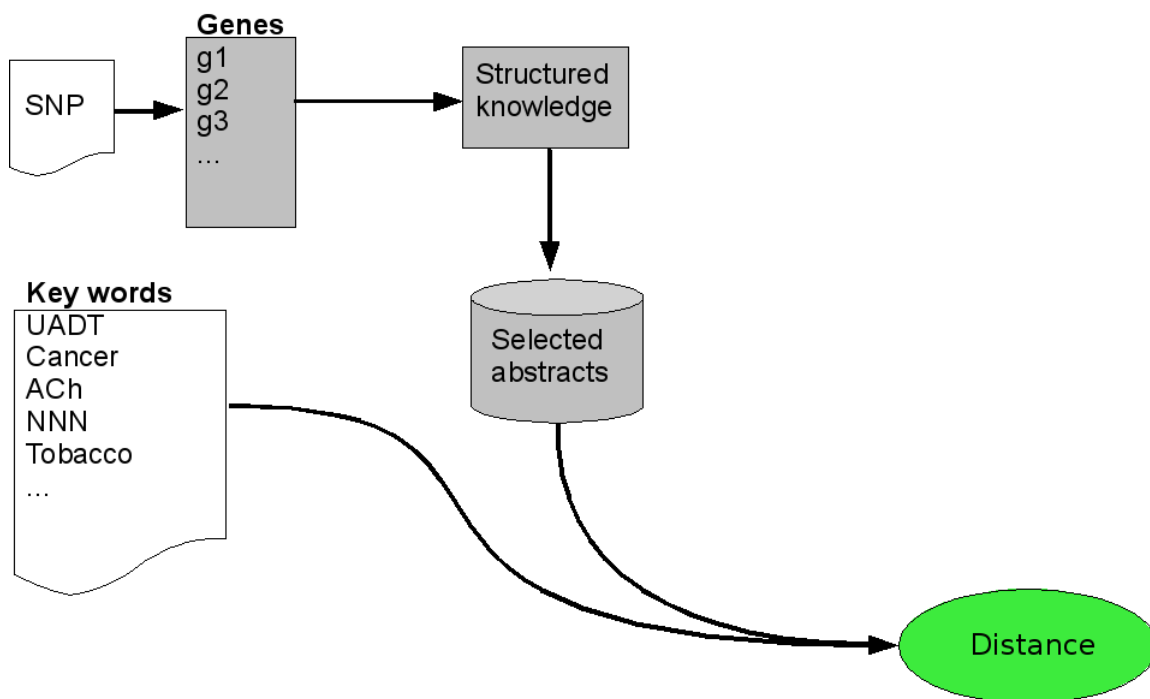


Figure 2.5: We can search for keywords in these abstracts and calculate the distance from these keywords to the gene marker

Semantic search: step 5

**Figure 2.6.** Alternatively, if we semantically annotate the abstracts then we could calculate such a distance from key concepts. For example, we could search not just for tobacco, but for the concept "tobacco product", which might include cigarettes and pipe tobacco.

Future plans: step 6

**Figure 2.7.** We could also include relational information from the associated knowledge sources in our search, and enhance the search with reasoning. For example, we

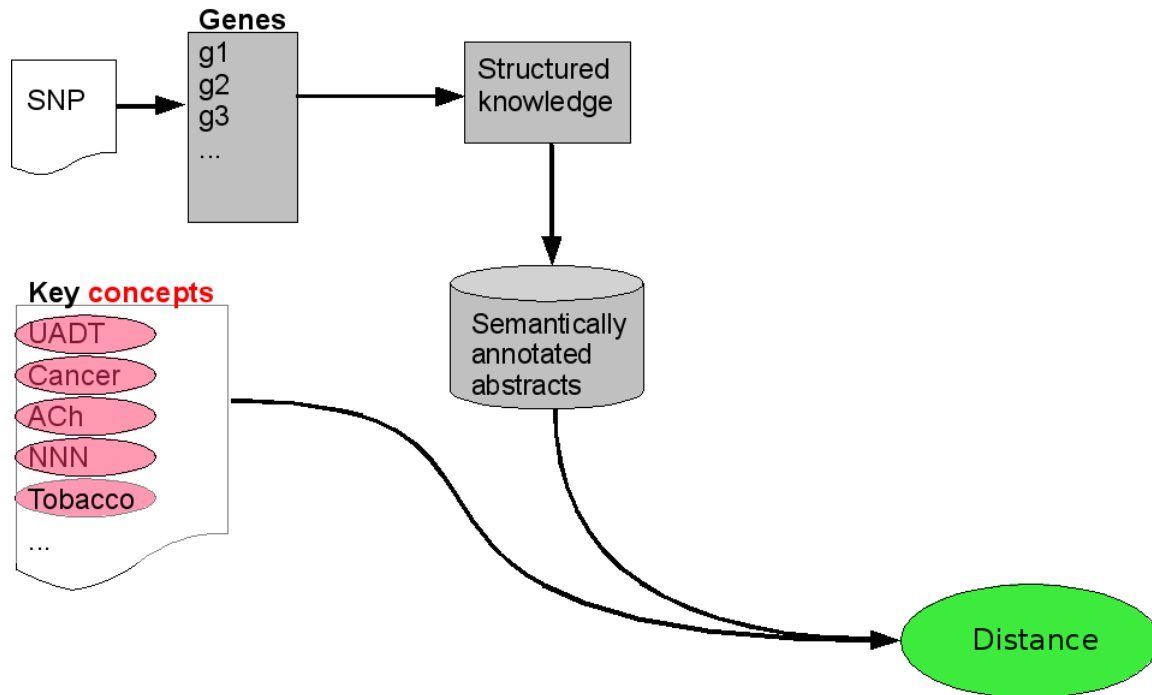


Figure 2.6: In the future: if we semantically annotate the abstracts then we can calculate distance from key concepts

could search for abstracts that mention anatomical parts of the lung, rather than the lung itself.

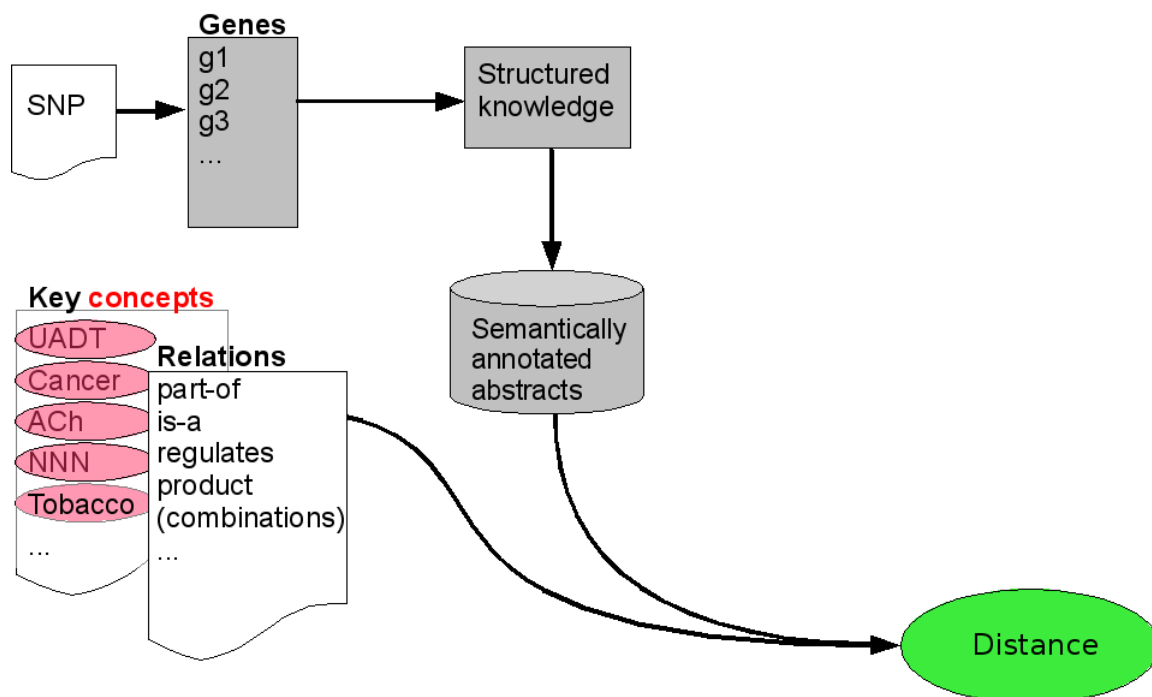


Figure 2.7: In the future: We could include further relational information from other associated knowledge sources



### 3. The GWAS service: how does it work?

In this chapter we describe how the system executes search queries submitted by users. First, we describe the two modes in which it can be used. We then briefly revisit the previous version, and follow this details of the newer version. We give details of the optimization carried out and how new features are implemented.

The GWAS application serves in two modes: single SNP search mode and service mode. In the first mode, it helps retrieve background knowledge associated with genes that are related to a SNP. The application, given a SNP ID and up to three sets of keywords, obtains the SNP's position and a set of genes which appear within a certain window of the SNP's position on the same chromosome. By searching in the background knowledge associated with these genes and in either matching abstracts or rif-texts with user-specified keywords, it calculates a prior score indicating how relevant the keyword groups are for the given SNP. The interface produces a list of background knowledge texts with matching keywords highlighted.

The second mode is a service mode. This is same as the first mode, except that in this mode, the user is allowed to specify more than one SNP per search and set more than one experiment to run simultaneously in the background. The GWAS application calculates a prior score (using the same logic as explained above) for each SNP in the query. In this mode, users only sees prior scores for each SNP, and not background knowledge texts.

The new version offers several improvements:

1. It is faster;
2. It allows matching keywords with different morphological inflections (i.e. “synthesis” will also match “synthesised” and “synthesising”).
3. It allows searching for search of MEDLINE abstracts in addition to in GeneRIFs
4. It allows for the expansion of a list of keywords with the use of UMLS and provides three different methods of obtaining related keywords.
5. It allows users to issue time-bound queries (e.g. search records before/after some date).
6. On completion of experiments, users are notified with emails containing links to download results of their respective experiments.

#### 3.1 GWAS Prototype V1

Given a SNP id, sets of keywords and a window in which to look for relevant genes, the earlier version of the GWAS application queried a large repository of life science information, Linked Life Data (LLD) to obtain a SNP's position and a list of genes that appeared within a specified number of base pairs (window) of the SNP's position. It also obtained all the background knowledge associated with these genes.

In order to find the relevance of keywords, each item in the background knowledge was searched to locate keywords and a score for each keyword group was calculated.

Querying LLD to obtain a list of genes that appeared within a certain window was found to be a bottleneck in the system. LLD is based on OWLIM. All indices in



OWLIM are hash based (e.g. values are not sorted). For this reason, when looking for genes that appear within a certain window of the SNP's position, the LLD had to probe every Gene to find out whether it is a relevant Gene or not every time a query for new SNP was issued. When searching for more than a million SNPs in the service mode, it made it almost impossible to carry out this operation.

Additionally, keyword lookup had problems too. The system had to have access to all background knowledge that was stored in LLD to check whether a specific keyword appeared in it or not. Also, because an exact keyword match was performed, PubMed articles with different inflections of keywords in the query were overlooked (so “synthesised” would be missed in a search for “synthesis”).

## 3.2 GWAS Prototype V2

From the analysis of the previous version, it was clear that there was a need for further indexing to speed up the system. We provided this with a system called MIMIR<sup>1</sup>. The MIMIR system allows indexing of linguistic metadata and document content. It is based on MG4J<sup>2</sup> (for free-text search) and OWLIM<sup>3</sup> (semantic search). MG4J is a Java library that allows managing (indexing) huge amount of data and is used for full-text search. OWLIM is the technology behind LLD, and enables us to index the linguistic metadata (annotations and annotation features) and allows semantically enabled rich queries. The new indices therefore combine LarKC's LLD with full text indices and semantic annotation indices.

### 3.2.1 Optimization

Optimization was achieved in two ways:

First, we pre-calculated associations among SNPs and genes that appear within a certain window (i.e. a certain number of base pairs) of one other. For the prototype version 2, we use a window of 100,000 (i.e. 50,000 on each side). We create a document in memory for each gene id listing all the associated SNPs. These documents are then indexed with MIMIR. As the MIMIR indices are inverted indices, given a SNP id, it helps us to obtain a list of gene ids that are associated with the SNP.

Secondly, for each gene we retrieve PubMed articles that have been associated with them in Entrez Gene. We then process these articles and index them with MIMIR. The metadata of these articles such as their publication date, associated gene ids and their titles etc. are stored in the index. While the information such as publication date helps in executing time-bound searches, information such as associated gene ids are helpful in figuring out genes that are related to the PubMed articles with the user specified keywords in them.

Additionally, we processed all the MEDLINE abstracts and GeneRIF texts with GATE [3, 5] to obtain the root form for each token in the text. We create separate MIMIR indices for PubMed articles and GeneRIF texts indexing not only surface forms of words but their root forms as well. This allows us to query not only on surface forms (like in the earlier version) but also on root forms enabling us to match documents

<sup>1</sup><http://gate.ac.uk/family/mimir.html>

<sup>2</sup><http://mg4j.dsi.unimi.it/>

<sup>3</sup><http://www.ontotext.com/owlim/>



with different inflections of query terms (i.e. “synthesis” will also match “synthesised” and “synthesising”). These indices allow us to easily and quickly identify genes whose background knowledge contain a given keyword. This is much faster than querying LLD to obtain all the background knowledge and then looking for every keyword in each.

As MIMIR contains a full-text search engine, it is possible to issue a single OR query with all the keywords listed in it. The keyword query needs to be issued only once. This is a major optimization over the earlier version where the keyword search had to be performed for every SNP. We cache the results of keyword search and perform an intersection with the gene ids obtained for each SNP in the first step to obtain the final set of gene ids.

### 3.2.2 Why MIMIR?

One could argue against storing the information on SNP and gene’s association in MIMIR and suggest storing this information in LLD (in the form of triples). The problem is that doing so could result in millions of new triples. Although the repository size is not an issue with LLD, there are couple of other problems at this stage of the development:

The work has been at experimental level and should a user decide to change the window size, one would have to delete all these statements from the LLD and add new pre-calculated statements. This would place an administration overhead on the people managing LLD. On the other hand, building MIMIR indices is cheap and could be achieved within few hours. The indices are small in size too, which means one could have more than one index available for different window sizes at the same time.

### 3.2.3 Keywords Suggestions

Sometimes specifying keywords relevant to the disease can be difficult. When keywords are assigned by a domain expert, this can inevitably introduce bias. One of the future plans listed in the report accompaning version 1 of the GWAS service [1] was to provide an automatic way of identifying relevant keywords to the disease in question. We have implemented two methods namely, Random Indexing (RI) and TF-IDF to address this issue. The Random Indexing method has been described in the LarKC deliverable [4]. In this section we describe the TF-IDF approach.

TF-IDF is a popular method for identifying keywords by obtaining a product of term frequency (TF) and inverse document frequency (IDF). In other words, how frequently a term occurs in the corpus in relation to how many documents in the corpus have the term appearing in them. A term with the highest product score is said to be the most relevant term describing the corpus. In our case, where the goal was to obtain terms that are highly relevant to the user specified keyword, we adopted the following approach.

We use MEDLINE abstracts and pre-process them to locate high frequency nouns and noun phrases. When a user provides a seed keyword (usually the disease being considered, e.g. lung cancer), first, we query the MIMIR index to obtain all the documents with the user keyword (such as lung cancer).



The terms identified at the pre-processing step in these documents are considered as candidate terms. For each of these terms, a TF-IDF score is calculated and the top N terms are presented to the user.

For the calculation of TF score, we consider all the documents with the user term in them as a single large document. The TF score is calculated by dividing the number of times a term appear in the big document by the number of all terms in the big document. For the calculation of IDF score, we use all the documents in the dataset (i.e. all MEDLINE abstracts) as individual documents. First, we divide the total number of documents in the collection by the total number of documents with the term in the context. Secondly, we take a log of this number. Finally, the TF-IDF score is calculated by taking the product of the TF score and the IDF score for that term.

For example, for the query “lung cancer”, the method returns the following top five keywords:

1. tg+/fvb
2. hsaecells
3. number of clinical predictors
4. e-cadherin genes
5. egfr-tk mutations

### 3.2.4 Semantic Searches

As explained in the previous deliverable, we wanted to annotate MEDLINE abstracts to make it possible for users to benefit from semantic and relational information in the associated knowledge sources. For example, one could search for abstracts that mention anatomical parts of the lung, rather than the lung itself.

In order to achieve this, we processed MEDLINE abstracts with GATE [3, 5]. We used a GATE wrapper for MetaMap<sup>4</sup> to annotate those texts with relevant UMLS concepts. This allows us to expand keyword lists by including other keywords that belong to same or sub-concepts in the UMLS taxonomy. We indexed the MetaMap processed MEDLINE articles with MIMIR. For each UMLS concept found in the MEDLINE abstracts, we created a document containing all the keywords annotated as that concept. Indices of such documents help to identify a concept from a given keyword and also in obtaining all the keywords for a given concept.

The keyword expansion procedure can be described as below:

1. Obtain a keyword from the user;
2. Search the MIMIR index to obtain a concept for the keyword;
3. Use LLD to obtain all equivalent and subclasses of the concept;
4. Use these subclasses to query the MIMIR index and obtain all the keywords that are annotated with these concepts;

---

<sup>4</sup><http://mmtx.nlm.nih.gov/>



5. Finally, expand the respective keyword group list with the keywords obtained in step 4.

For example, for the query “lung”, the method returns the following five keywords (randomly chosen from total of 8000 terms returned by the search):

1. pulmonary alveoli
2. bronchiolar
3. primary hbe
4. venous thromboembolic diseases
5. lamina cribrosa

### 3.3 Final GWAS Prototype, V3

The final version of the prototype described by this document differs only marginally from the V2 prototype. Apart from minor bug fixes and performance improvements the main change between V2 and the final prototype is that the latest version now runs upon v2.5 of the LarKC Platform. The details of the plugins and workflow resulting from this refactoring work are detailed in Section 5.2

### 3.4 Performance Statistics

With a single thread process on a standard desktop, we are able to process between 600 and 700 SNPs per second when search is restricted to gene-rifs. This number goes down to between 350 and 400 SNPs per second when the search is issued over MEDLINE abstracts. This happens because the MEDLINE abstracts are bigger in size and far greater in number than the total GeneRIF texts.



## 4. Using the prototype

### 4.1 Introduction

This Chapter describes use of the GWAS use case prototype. We discuss web interfaces for both search modes (the single SNP search mode and the service mode) and explain how to use them with the help of a few screen shots. The first section gives examples of the type of input with which the prototype can be used. The second section describes how to use the software, screen by screen.

### 4.2 A small example

This section gives examples of the data that can be used with the prototype. It is not intended to imply that the prototype can only run with this input, only that these examples give interesting results. The prototype can be run with any SNP specified in the HapMap data (build 37).

#### SNP IDs

The SNPs considered most important for the lung cancer example we will use, are:

- rs8034191
- rs1051730
- rs4324798
- rs3117582
- rs2736100
- rs401681

In the examples in the next section, we will use rs1051730.

#### Keyword groups

We will use the following keyword groups, as used for analysis of a lung cancer data set.

- **Group 1**
  - lung
- **Group 2**
  - smoking
  - carcinogen
  - non-small cell carcinoma



- **Group 3**
  - DNA repair
  - genetic disease

## 4.3 A bigger example

Further sample data is provided in Appendix A. This data is suitable for use in GWAS service mode, and consists of approximately 1000 random SNPs combined with six SNPs known to be relevant in lung cancer.

Service mode can take many hours to run over a full set of SNPs. It is suggested that for demo and review purposes, the sample data given in the appendix is instead used. This data will run in minutes. It should be copied and pasted into a local text file for upload via the GWAS web interface.

## 4.4 Using the prototype

This section describes how to use the prototype for both the search modes. The following sub-sections run through using the prototype for a fully worked example, in the analysis of data from a lung cancer GWAS. The sub-sections describe using the prototype in sequence, and should be followed in the order given. It is intended that a user following these steps should be able to replicate the experiment shown, and then go on to try their own analyses.

### 4.4.1 Single SNP search mode

As the name suggests, the interface allows users to obtain a prior score for a single SNP. The figure 4.1 shows the web interface for the single SNP search. We will go through each caption in the image and explain.

### Configuring a single SNP search

The user is expected to provide a SNP ID for which he/she wants to obtain a prior score.

The user can also provide keywords in three different groups. The prior score is calculated based on which keywords from what groups are found in the background knowledge. The user should write only one keyword per line. The user can also write phrase expressions to match (for example, “non-small cell carcinoma”). Given such a keyword/phrase, the program will try to locate MEDLINE abstracts or GeneRIF texts with these keywords appearing in them. The new version also allows user to specify a root form of a word (by specifying e.g. “root:smoke” in the search box). This allows retrieving documents with different inflections of the word “smoke” appearing in them (i.e. smoke, smokes, smoking and smoked).

A user can now also specify if he/she wants to expand the keywords list by obtaining related keywords from the UMLS database. This is done by selecting the “Expand keyword list with UMLS” check box. For instance, when a user specifies “lung” as

a keyword, a user might want to include the term “alveolus”, which is part of the “lung”. Expanding keywords list with the UMLS makes it possible to achieve this.

The earlier version only allowed searching on GeneRif texts. The new version allows users to select one of the two sources: MEDLINE abstracts and GeneRif texts. The text to be searched is chosen via a drop down list.

The new version also indexes information such publication dates of the MEDLINE articles. Such information is used to allow users to issue time-bound searches. In other words, user can specify if they want to restrict their keyword searches to the MEDLINE articles published before or after a certain date.

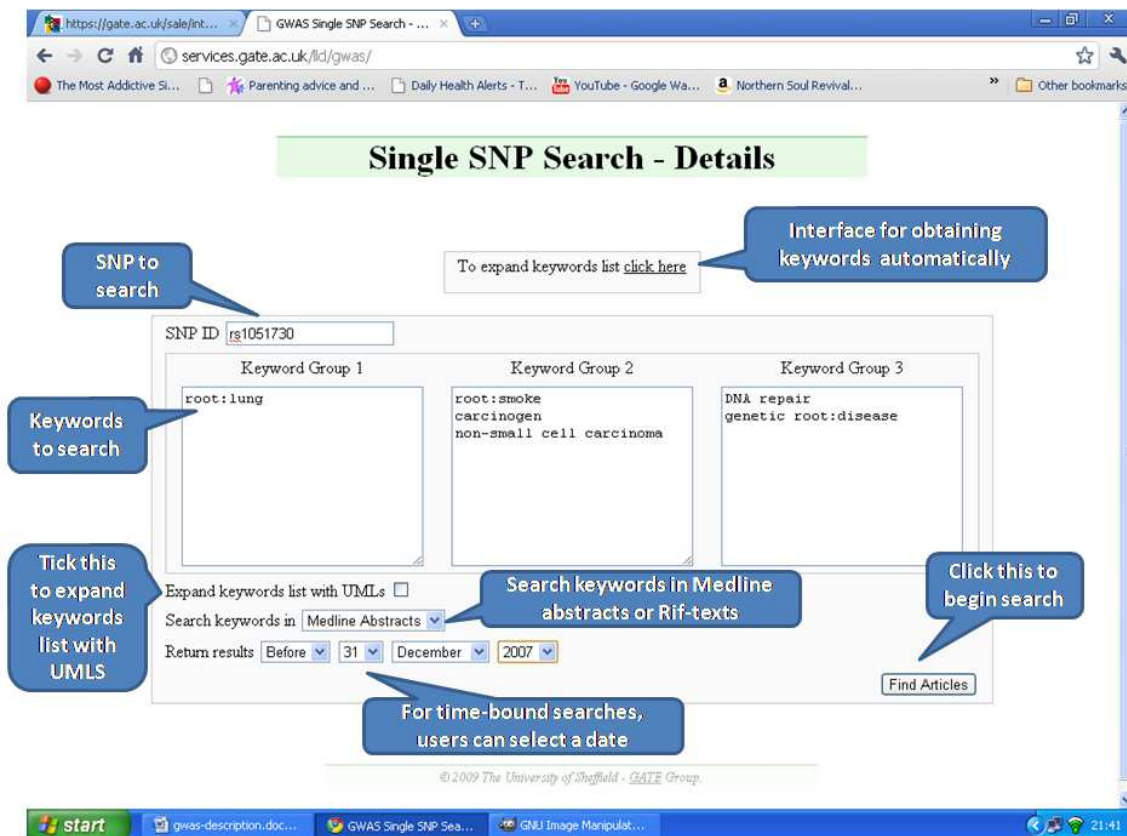


Figure 4.1: Configure a single SNP search.

## Automatic keyword acquisition

The interface has a new option that assists users in deciding keywords relevant to the disease name. This option appears at the top with caption “To expand keywords list click here”. Clicking on the link “click here” expands the interface (see figure 4.2).

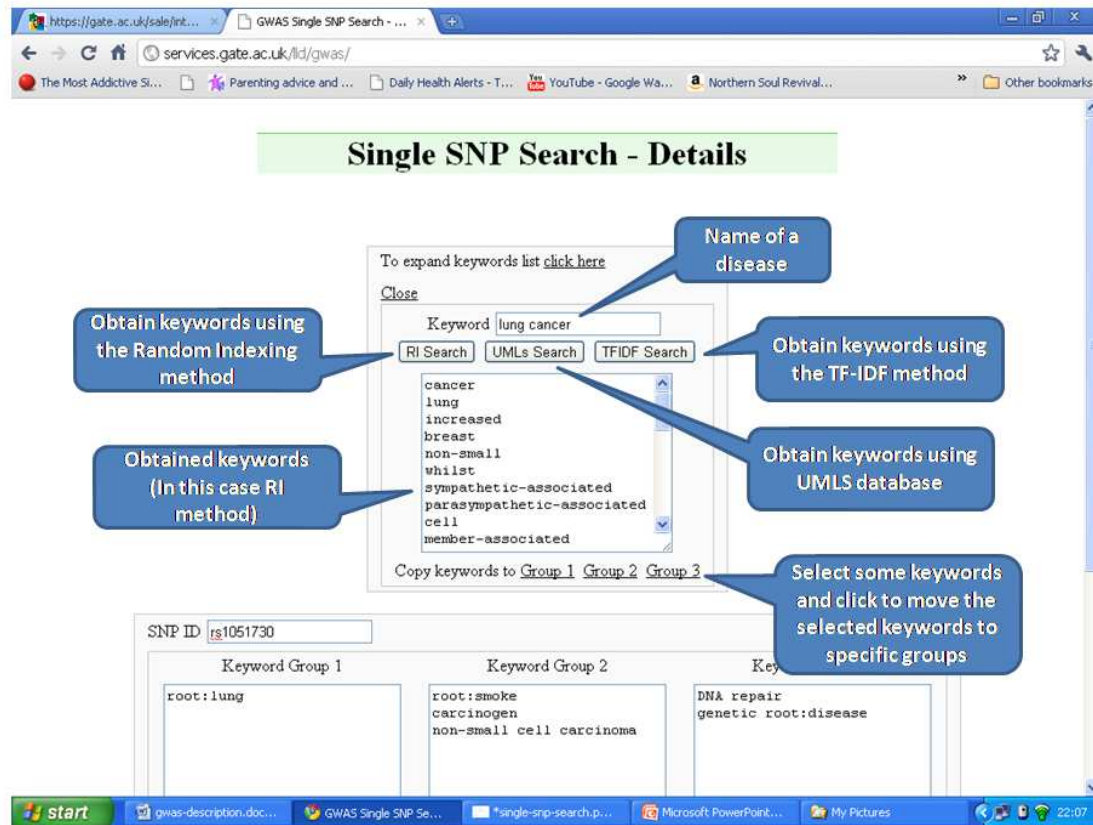
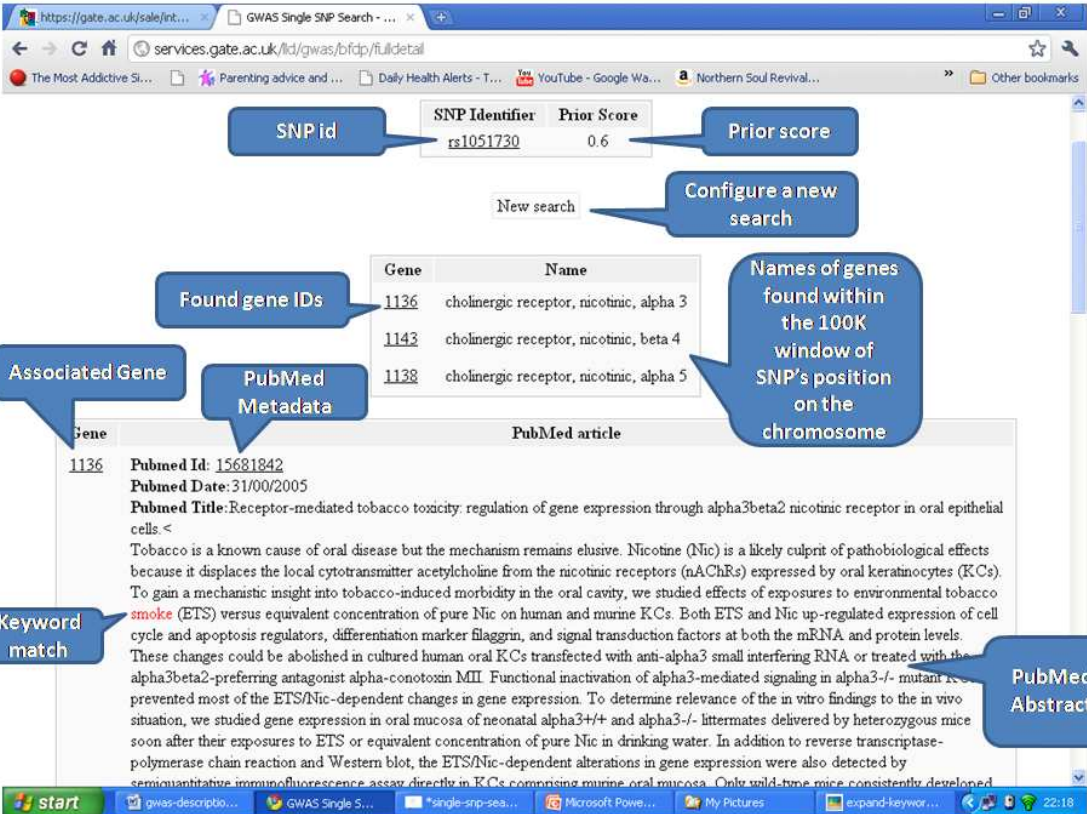


Figure 4.2: Automatic Keyword Acquisition

As can be seen in the figure (figure 4.2), the user needs to provide a name of the disease or important keyword related to a disease for which he/she wants to obtain keywords (e.g. lung cancer). There are three methods of obtaining keywords automatically: Random Indexing (RI) search, TF-IDF search and the UMLS search. The user can choose one of the three methods to retrieve a maximum of 30 keywords relevant to the user’s seed keyword. It also provides a way to move these keywords into different keyword groups. For example, a user can select the first five keywords and click on the link “Group 1”. This will move the selected words to the Keyword Group 1. If no text is selected, all the keywords are moved to the specified keyword group.

## Single SNP search results

On the submission of a search, the system, using the keywords and a SNP ID, tries to locate genes that appear within a window of 100K of the SNP's position. On the one hand, if no keywords are provided, the system obtains all such genes and asks index to return all the associated background knowledge to these genes. On the other hand, when keywords are provided, only the background knowledge with at least one of the keywords in them is selected.



The screenshot shows a web browser displaying the results of a GWAS Single SNP Search. The search parameters are: SNP Identifier: rs1051730, Prior Score: 0.6. A table lists the found genes:

Gene	Name
1136	cholinergic receptor, nicotinic, alpha 3
1143	cholinergic receptor, nicotinic, beta 4
1138	cholinergic receptor, nicotinic, alpha 5

Below the table, a PubMed article is displayed for Gene ID 1136. The article title is "Receptor-mediated tobacco toxicity: regulation of gene expression through alpha3beta2 nicotinic receptor in oral epithelial cells." The abstract text includes the keyword "smoke" highlighted in red. Callouts in the image identify the following elements:

- SNP id**: Points to the SNP Identifier field.
- Prior score**: Points to the Prior Score field.
- Configure a new search**: Points to the "New search" button.
- Found gene IDs**: Points to the Gene column in the table.
- Names of genes found within the 100K window of SNP's position on the chromosome**: Points to the Name column in the table.
- Associated Gene**: Points to the Gene ID 1136.
- PubMed Metadata**: Points to the PubMed ID, Date, and Title.
- Keyword match**: Points to the word "smoke" in the abstract.
- PubMed Abstract**: Points to the full abstract text.

Figure 4.3: Single SNP Search Results

As shown in the figure 4.3, it shows the SNP ID with the assigned prior score. The link on the SNP ID takes users to the relevant HapMap page. At the top of the page, it also provides a list of genes found for the SNP. Along with the gene IDs, the page also shows names of these genes. Clicking on the gene ID links, the interface links users to the articles associated with the gene, on the PubMed website. For each abstract on PubMed, the result page shows information such as its ID, article title, its publication date and the abstract text with the matching keywords highlighted in red.



#### 4.4.2 Service mode

This mode allows users to submit a list of SNPs and process them in a batch to obtain their prior scores. The mode is first described, and then a step-by-step example given using the sample data in Appendix A.

## Experiment progress reports

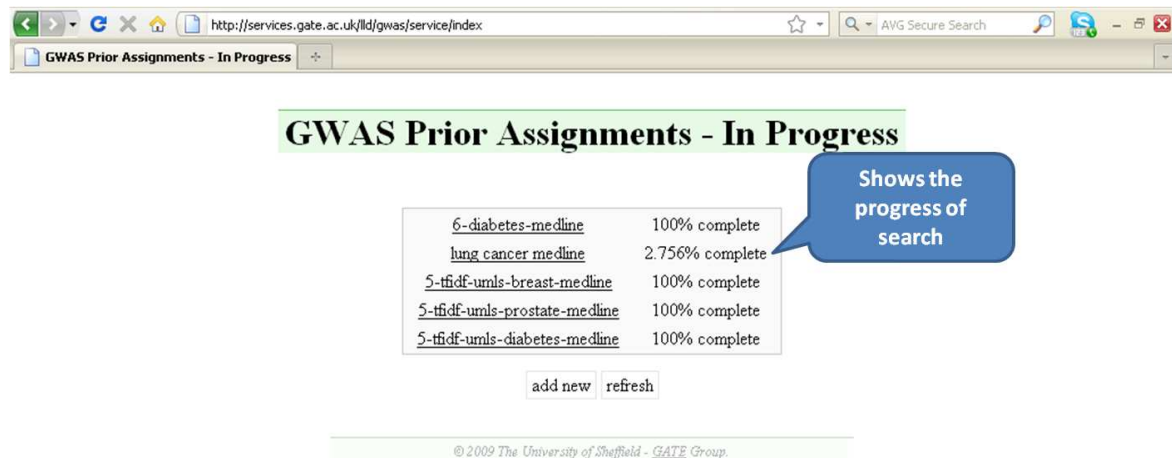


Figure 4.4: Shows the Progress of an Experiment

Figure 4.5 shows the first screen of the interface for the service mode. The screen lists all experiments in progress. The progress is reported in percentages. Figure 4.4 demonstrates this. User can click on the experiment name to see details of the experiment. A new experiment can be added by clicking on “add new”, as described in the next section.

## Configuring an experiment

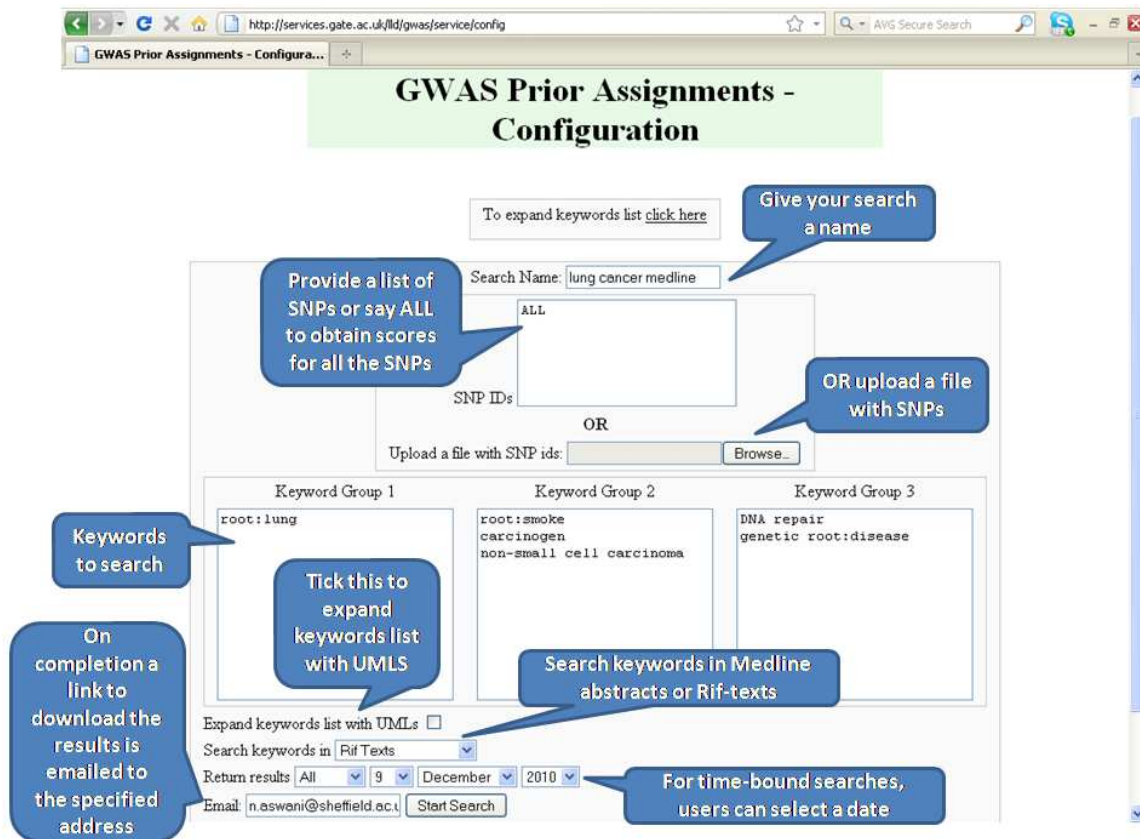


Figure 4.5: Configuring an Experiment in the Service Mode

If a user chooses “add new” on the previous screen, they are taken to a screen to configure a new experiment.

Users can provide IDs of one or more SNPs (one SNP per line) to analyse, or simply type “ALL” to obtain prior scores for all the SNPs in the database (approx. 15 million SNPs). It is **NOT** recommended that users do this for demo or review purposes, as results can take many hours to compute. Instead, it is suggested that the sample data in Appendix A is used. Users can also write a SPARQL query that is executed against LLD to obtain a list of SNPs. The interface also allows users to upload a file with SNP IDs listed one SNP per line (as in Appendix A).

As is the case with the single SNP search, users can configure their experiments by providing all the other parameters in exactly the same way, and as described for the single SNP search.

Users are asked to provide their email address. The email address is used for notifying users with updates on their experiments. On completion, users are sent email with links to download the experiments’ results. A user can then simply click on the link to initiate download of their results.

## Examining experiment details

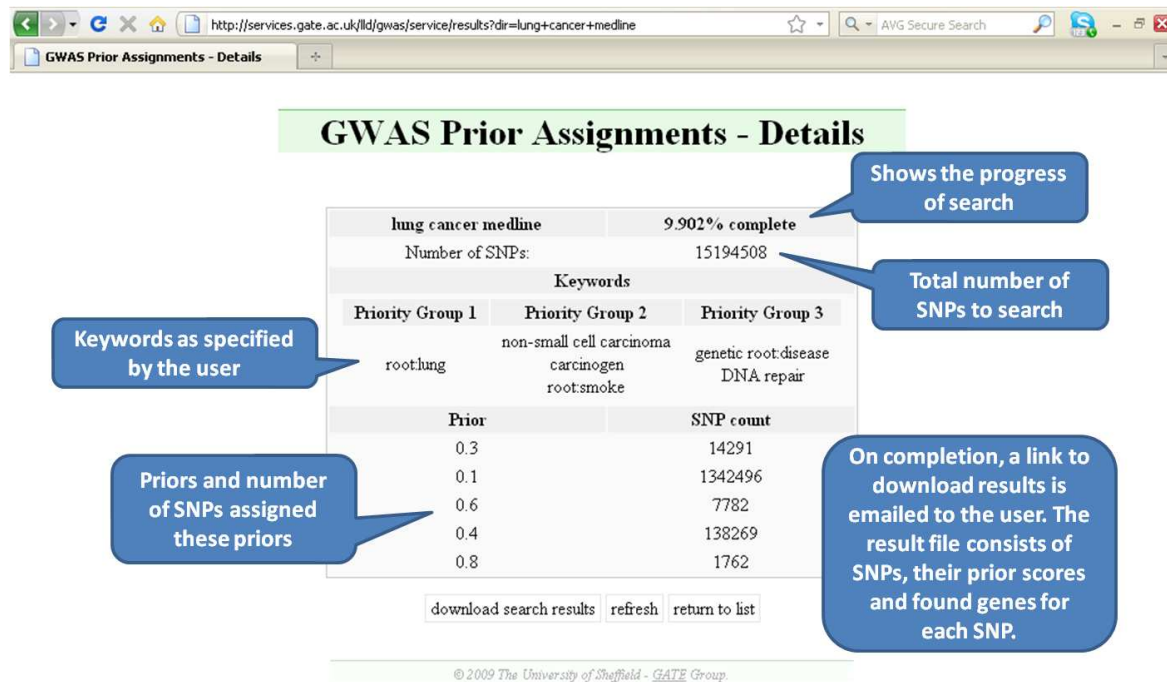


Figure 4.6: Experiment Details

Figure 4.6 shows the page with details of an in progress experiment. This screen is reached by clicking on the experiment name from the first screen. As shown in the figure, it gives the total number of SNPs the experiment is configured to process. It also shows the keywords that user provided for the experiments. It also gives distribution of SNPs for the prior scores they are assigned. In other words, it shows the number of SNPs for each prior score.

At the bottom of this page, a button is given to download the results of an experiment, once it is completed, labelled “download search results”. Results are provided one line per SNP, with commas separating fields. The user may then combine the prior probabilities with experimental GWAS results, to give a ranking of SNPs. Formulae for combining results in a Bayesian model are given in Chapter 5.

### A service mode example

This example uses the sample data given in Appendix A). In order to use the data, copy and paste it into a plain text file on your local computer.

1. Go to the first page of the GWAS service mode.
2. Click on “add new”.
3. Give your search a name in the first field, e.g. “test”
4. Click on the “Browse...” button and navigate to the file of sample data from Appendix A.



5. Enter keywords in the boxes for group 1, 2, and 3. Each keyword should be on a new line. Try the keywords given in Section 4.2.
6. Add your email address.
7. You will be returned to the first screen. Your experiment will appear in the list, with a percentage completion. You can refresh this every so often with the “refresh” button. You will also receive an email when 100% complete.
8. Once complete, click on the name of your search to go to the results page.
9. Download results with the “download search results” button. You can open results in a spreadsheet to examine them. Search for the SNPs known to be associated with lung cancer (see Appendix A), and note that they usually have higher prior probabilities.



## 5. Technical documentation

### 5.1 Introduction

This section reports on the installation and technical documentation provided for the GWAS prototype. First, we give a summary of the LarKC workflow and plugins used by the prototype. This is followed by details of installation and then pointers to other sources of documentation. The final section describes how the prior probabilities delivered by the GWAS service can be used to calculate a Bayesian statistic for use in ranking SNPs in a GWAS analysis.

### 5.2 LarKC plugins and workflow

The GWAS prototype web interface has been placed in the LarKC version control repository <sup>1</sup>.

The web interface is used to collect the relevant information from the user and to publish the results. Scientists are unlikely to be used to accessing a raw SPARQL endpoint and as such a web GUI was adopted. This service is, however, implemented as a LarKC workflow consisting of a number of plugins; including a dedicated GWAS plugin. The workflow is defined via RDF as:

```
# Workflow Description
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix larkc: <http://larkc.eu/schema#> .

# Define three plug-ins
_:plugin1 a <urn:eu.larkc.plugin.identifier.gwas.GWASIdentifier> .
_:plugin2 a <urn:eu.larkc.plugin.S0StoVBtransformer> .

# Connect the plug-ins
_:plugin1 larkc:connectsTo _:plugin2 .

# Define a path to set the input and output of the workflow
_:path a larkc:Path .
_:path larkc:hasInput _:plugin1 .
_:path larkc:hasInput _:plugin2 .
_:path larkc:hasOutput _:plugin2 .

# Connect an endpoint to the path
_:ep a <urn:eu.larkc.endpoint.sparql> .
_:ep larkc:links _:path .
```

As you can see from this description the standard workflow consists of just two plugins

- **GWASIdentifier:** the dedicated GWAS plugin which computes the score of each SNP. This plugin is available via SVN from <https://larkc.svn.sourceforge.net/svnroot/larkc/trunk/plugins/plugin.identifier.gwas>

<sup>1</sup><https://larkc.svn.sourceforge.net/svnroot/larkc/branches/wp7b/service-interface>



- **SOSToVBtransformer:** converts the set-of-statements produced by the GWAS plugin to a SPARQL variable binding for return by the endpoint. This plugin is available via SVN from <https://larkc.svn.sourceforge.net/svnroot/larkc/trunk/plugins/SOSToVBtransformer>

The first stage in the GWAS identifier is to run the SPARQL query against LLD in order to determine the set of SNPs to process. The SPARQL query can take any form as long as each result is a single variable binding containing the URI of the SNP to score. This information along with the keywords extracted from the original query is then used to drive the main service code as described above. This results in each selected SNP being given a prior score, and these pairs are then stored as a set of statements which are returned to the decider and back to the web interface for consumption by the scientist who initiated the search.

## 5.3 Installation

### 5.3.1 Introduction

This section gives a high-level description of installing the software. In order to install, a system administrator will need a good understanding of web application deployment and Java, and will need to refer to other documentation as referenced here.

The GWAS Prototype requires the following components to be installed, deployed or accessible:

- MIMIR indices
- Linked Life Data repository
- UMLS repository
- Random Indexing service
- GWAS application

### 5.3.2 MIMIR Incides

The prototype relies heavily on the MIMIR indices. It uses these indices to store information such MEDLINE abstracts, GeneRif texts, MetaMap annotations on MEDLINE abstracts, SNP to Gene associations and terms identified for the TF-IDF score calculations. The MIMIR library allows its indices to be accessed remotely, however, for the GWAS prototype, they are expected to live together with the GWAS web application.

There are five indices of approximately 2.9GB in size. Since the GWAS application requires access to these indices, the absolute location of the MIMIR indices (on the disk) is required to be specified in the GWAS configuration file. The steps to configure the configuration files are described later in the section 5.3.6.

In order to build these indices from scratch, one needs to have access to several huge datasets. For example, all of the MEDLINE abstracts, their GeneRif texts, SNP data sets, UMLS datasets, MetaMap installation etc. If a user can afford to download all these data, it is possible to rebuild all the MIMIR indices with the GWAS application. However, the easiest and recommended way to get access to the ready-to-use indices is to contact the authors.



### 5.3.3 Linked Life Data repository

It was a requirement of the previous version to install a copy of the LLD and customize it to use with the GWAS application. The current GWAS prototype utilizes the publicly available LLD instance (running at <http://www.linkedlifedata.com>).

If one wants to deploy and use his or her own copy, it is mandatory to set appropriate property values in the GWAS configuration files. A URL of the LLD along with any authentication details need to be specified in the configuration file. The section 5.3.6 provides more information on these parameters.

### 5.3.4 UMLS repository

As explained earlier in the deliverable, the GWAS application makes use of MetaMap annotations along with the UMLS dataset to exploit the relational knowledge available in these resources. Instead of storing UMLS related information in LLD, we have created a separate repository with all the relations among UMLS concepts. Given a concept, this allows querying the repository with a SPARQL query and obtain information such as the subconcepts of the concept. As it is based on the same technology used for the LLD, it scales to millions of triples and makes it possible to retrieve results in real-time.

The size of the UMLS repository totals to 33GB. As is the case with other components of the GWAS prototype, the GWAS application needs to know about the URL of this repository too and it can be set by setting the appropriate parameter in the configuration file. The section 5.3.6 provides more information on this parameter.

### 5.3.5 Random Indexing service

The LarKC Random Indexing service has two components: a web application and a vector space model. The web application provides an easy to use web interface to query the vector space model with a keyword or phrase and obtain related keywords.

A war file along with the vector space model based on the MEDLINE abstracts is distributed for the deployment of the random indexing service. One needs to simply drop the war file in a web application container and set the location of the vector space model in a file called “Services.properties”. The file is located inside the “WEB-INF/classes” folder. The parameter “vectorsFilePath” should be set to the absolute location of the vector space model on the server.

The GWAS application needs to know about the URL of this service. Once deployed, URL of the service should be set in the configuration file of the GWAS application.

### 5.3.6 GWAS application

The GWAS prototype that this report accompanies has been placed in the LarKC version control repository <sup>2</sup>.

The GWAS prototype has been built with the GRAILS application toolkit [6] and is distributed as a war file. The war file can be easily deployed in a web application container such as the tomcat or jetty.

---

<sup>2</sup><https://larkc.svn.sourceforge.net/svnroot/larkc/branches/wp7b/service-interface>



As explained earlier, the GWAS application relies heavily on several other components. In order for GWAS application to use them, details of these components need to be specified in a configuration file “gwas/grails-app/conf/Config.groovy”. Parameters in the configuration file must be set prior to building a war file for the final deployment.

Below, we list these parameters:

- **mimir.index.dir:** absolute location of the mimir indices’ root folder.
- **bk.endpoint:** a URL to access the LLD instance
- **repository.id:** id of the repository
- **bk.endpoint.username:** username to access the LLD instance
- **bk.endpoint.password:** password to access the LLD instance
- **umls.endpoint:** a URL to access the UMLS repository instance
- **umls.repository.id:** id of the UMLS repository
- **umls.endpoint.username:** username to access the UMLS repository instance
- **umls.endpoint.password:** password to access the UMLS repository instance
- **ri.endpoint:** a URL to access the random indexing service

The current version of the prototype provides an easy way to create a war file. It can be built by simply calling the “ant clean war” command in the “gwas” folder.

## 5.4 Further documentation

Those readers interested in the technical detail at a program code level may consult three sources of information:

- The GWAS prototype makes use of software written in LarKC WP2. This is described more fully in the previous sections.
- The above software is fully documented in the source code repository, using Javadoc.
- The interface is written as a standard webapp using the Grails application framework. The structure and content of the interface may be understood with an introductory level of Grails. See for example [6].



## 5.5 Computing BFDP from priors

The web service computes a prior  $\pi_0$  for SNPs, based on the occurrence of key words or concepts in text and biomedical knowledge resources. To use this with data from a GWAS,  $\pi_0$  must be combined with the experimental Odds Ratio ( $OR$ ) to give a Bayesian False Discovery Probability ( $BFDP$ ), as detailed in [9]. The smaller a  $BFDP$  is, the more likely the corresponding SNP is relevant to the disease. The notes below show how BFDP is calculated step by step from  $\pi_0$  and  $OR$ .

Given:

- Odds Ratio of the SNP,  $OR$
- Upper 95% confidence interval of the SNP,  $U_{95}$
- Prior of the SNP,  $\pi_0$

And defining the following:

- Standard normal deviate,  $N = 1.96$
- Upper odds ratio,  $U_O = 2$

Then the following steps may be used to calculate BFDP:

$$\hat{\theta} = \log(OR) \quad (5.1)$$

$$v_{root} = \left| \frac{\hat{\theta} - \log(U_{95})}{N} \right| \quad (5.2)$$

$$w = \left( \log\left(\frac{U_O}{N}\right) \right)^2 \quad (5.3)$$

$$r = \frac{w}{w + v_{root}^2} \quad (5.4)$$

$$z = \frac{\hat{\theta}}{v_{root}} \quad (5.5)$$

The Approximate Bayes Factor,  $ABF$ , is given by:

$$ABF = \frac{e^{-r \times z^2}}{\sqrt{1-r}} \quad (5.6)$$

and the prior odds is given by:

$$PO = \frac{\pi_0}{1 - \pi_0} \quad (5.7)$$

BFDP can now be computed as

$$BFDP = \frac{ABF \times PO}{ABF \times PO + 1} \quad (5.8)$$



## 6. Future work and conclusion

### 6.1 Future work

The software deliverable accompanying this report is a second version, and a prototype. Future work can be considered along three dimensions, as describe below.

**Use of the LarKC platform** The reported software uses LarKC plugins running on the LarKC platform, and the LarKC data layer. There are, however, some aspects of the platform that are not yet fully used. Mainly, the version of the platform available when the reported software was written, did not support full parallelisation of plugins. For the reported software, this is a highly desirable requirement, as each analysis may take hundreds of milliseconds on a commodity workstation, and analysis must be carried out over around fifteen million data points.

**Using more powerful search and analysis** The reported software uses a small number of biomedical knowledge sources, keyword and semantic search over biomedical literature. The technique, and LarKC platform, should support the use of multiple interconnected knowledge sources.

**Extending the range of prior knowledge** The last point considered the limited number of knowledge sources from a technology perspective, we may also consider the limited number of knowledge sources from a domain perspective. Now that end users have seen the approach in use, they have several ideas of how the theoretical approach can be extended.

### 6.2 Conclusion

This report has presented an approach to improving the analysis of data from Genome Wide Association Studies (GWAS). GWAS experiments look at the relative likelihood of a gene marker (SNP) being present in disease subjects and controls. Experimental data for a SNP is combined with prior probabilities, derived from the presence or absence of relevant information in the research literature and research databases.

The deliverable accompanying this report provides software that implements this approach. The approach has been implemented using LarKC plugins, a LarKC workflow, and the LarKC data layer. A web interface has been developed, with which end-users can use the prototype.

This report has documented the software, providing a step-by-step user guide, and technical documentation.

The prototype and its use are further described in the relevant LarKC deliverable [7].



## REFERENCES

- [1] A.Roberts, M. Greenwood, D. Damljanovic, H. Cunningham, T. Heitz, I. Roberts, Y. Li, M. Johannson, and J. McKay. D7b.3.1b version 1 prototype. Technical report, LarKC project deliverable, 2009.
- [2] Niraj Aswani, Mark Greenwood, Mattias Johansson, Angus Roberts, James McKay, Jon Wakefield, Valentin Tablan, Ian Roberts, Hamish Cunningham, and Paul Brennan. D7b.3.2b version 2 prototype. Technical report, LarKC project deliverable, 2010.
- [3] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics*, pages 168–175, Philadelphia, PA, USA, July 2002.
- [4] D. Damljanovic, J. Petrak, M.Greenwood, A.Roberts, H.Cunningham, I. Peikov, A. Kyriakov, Mihai Lupu, Jose Quesada Danica Damljanovic, and Hamish Cunningham. D2.2.2, 2.5.2. Month 24 Selection Components (report accompanying two software deliverables). Technical report, LarKC project deliverable, 2010.
- [5] Gate — general architecture for text engineering, 2009. <http://gate.ac.uk>.
- [6] Grails, 2009. <http://grails.org/>.
- [7] Mattias Johansson, Niraj Aswani, Mark Greenwood, Angus Roberts, James McKay, Jon Wakefield, Valentin Tablan, Ian Roberts, Hamish Cunningham, and Paul Brennan. D7b.3.2a version 2 iteration report. Technical report, LarKC project deliverable, 2010.
- [8] Angus Roberts, Kurt Straif, James McKay, Martin Stetter, and Hamish Cunningham. D7b.1.1a requirements summary and data repository. Technical report, LarKC project deliverable, 2009.
- [9] J. Wakefield. A bayesian measure of the probability of false discovery in genetic epidemiology studies. *The American Journal of Human Genetics*, 81(2):208–227, August 2007.



## A. Appendix - Sample data

### A.0.1 Introduction

This appendix provides a small sample of SNPs with which to try out and demonstrate the GWAS service. The data comprises IDs for 1000 SNPs. Six of the SNPs are known to be associated with lung cancer:

- rs1051730
- rs8034191
- rs4324798
- rs3117582
- rs401681
- rs2736100

These six are distributed amongst the remainder, which are randomly chosen from all known SNPs. A user can cut and paste these 1000 SNPs into a text file, and load them into the GWAS service via the web interface, for experimentation and demonstration. If successful, the resulting data file exported from the service will give higher priors for the above 6 SNPs than for most random SNPs. This can be verified by searching for the above six in the results file.

### A.0.2 Sample data

```
rs35748075  
rs35748076  
rs35748073  
rs10776870  
rs34970208  
rs10776871  
rs10776872  
rs10776873  
rs35748070  
rs34970207  
rs10776874  
rs34970205  
rs34970202  
rs16976076  
rs34970200  
rs35748077  
rs35748078  
rs16976069  
rs10776865  
rs10776864
```



rs10776867  
rs10776866  
rs10776868  
rs35748084  
rs10776880  
rs35748085  
rs10776881  
rs35748086  
rs34970217  
rs10776884  
rs35748081  
rs10776885  
rs10776882  
rs35748083  
rs10776883  
rs34970213  
rs16976087  
rs34970215  
rs34970216  
rs35748088  
rs34970210  
rs16976083  
rs10776878  
rs10776877  
rs10776876  
rs10776875  
rs10776879  
rs10776893  
rs10822184  
rs35748050  
rs10776894  
rs10822183  
rs34970228  
rs10776895  
rs10822186  
rs10776896  
rs35748053  
rs10822180  
rs35748054  
rs10776890  
rs10776891  
rs10822182  
rs35748052  
rs10776892  
rs10822181  
rs35748057  
rs34970222  
rs34970223



rs16976052  
rs35748055  
rs34970220  
rs35748056  
rs34970221  
rs34970226  
rs10822188  
rs10822187  
rs35748059  
rs10822189  
rs34970230  
rs10776887  
rs10776886  
rs10776889  
rs10776888  
rs34970239  
rs10822197  
rs35748060  
rs35748061  
rs10822194  
rs10822192  
rs35748064  
rs10822191  
rs35748065  
rs10822190  
rs35748066  
rs34970231  
rs17819303  
rs34970232  
rs17819305  
rs35748069  
rs34970234  
rs34970235  
rs34970236  
rs16976065  
rs34970238  
rs10822198  
rs17819300  
rs16976059  
rs34970241  
rs34970240  
rs10776899  
rs10776898  
rs10776897  
rs10822165  
rs16976033  
rs10822167  
rs10822168



rs10822169  
rs10822160  
rs10822161  
rs10822162  
rs10822163  
rs10822164  
rs10941081  
rs10941080  
rs10941085  
rs10941084  
rs10941083  
rs10941082  
rs10941089  
rs16976028  
rs10941086  
rs16976029  
rs16976025  
rs10822178  
rs10822179  
rs10822176  
rs16976044  
rs1051730  
rs10822170  
rs10822171  
rs10822174  
rs10822175  
rs10822172  
rs10822173  
rs10941090  
rs10941092  
rs10941091  
rs10941094  
rs10941093  
rs10941096  
rs10941095  
rs10941098  
rs16976039  
rs10941097  
rs10941099  
rs16976036  
rs10822147  
rs10822148  
rs10822149  
rs10822143  
rs10822144  
rs10822145  
rs10822146  
rs35748094



rs10822140  
rs35748092  
rs10822141  
rs35748091  
rs10822142  
rs35748098  
rs35748096  
rs35748095  
rs16976006  
rs10822158  
rs10822159  
rs10822156  
rs10822157  
rs16976020  
rs10822154  
rs10822155  
rs16976022  
rs10822152  
rs10822153  
rs10822150  
rs10822151  
rs16976014  
rs16976015  
rs16976018  
rs34970294  
rs10941042  
rs10941043  
rs34970296  
rs10941044  
rs10941045  
rs34970290  
rs10941046  
rs10941047  
rs34970292  
rs10941048  
rs34970291  
rs8034191  
rs10941040  
rs10941041  
rs34970286  
rs10822128  
rs34970289  
rs10822129  
rs10941055  
rs10941056  
rs10941053  
rs10941054  
rs10941058



rs2611098  
rs2611099  
rs2611096  
rs2611097  
rs10822131  
rs10822130  
rs10822133  
rs2611090  
rs2611091  
rs10822135  
rs10822134  
rs34970299  
rs10822137  
rs2611094  
rs10822136  
rs2611095  
rs10822139  
rs2611092  
rs10822138  
rs2611093  
rs10941064  
rs10941060  
rs10822101  
rs10941079  
rs10941078  
rs10941075  
rs10941076  
rs10941073  
rs10250600  
rs10941074  
rs10250601  
rs10941071  
rs10250602  
rs10250605  
rs10250606  
rs10250608  
rs10250609  
rs10822114  
rs10822117  
rs10250617  
rs10250615  
rs10250614  
rs10250613  
rs10250612  
rs10250611  
rs10941001  
rs34970252  
rs10941000



rs10941003  
rs10941002  
rs10941005  
rs10941004  
rs34970247  
rs34970246  
rs34970249  
rs34970243  
rs34970242  
rs10250625  
rs10250627  
rs10250621  
rs10250624  
rs34970262  
rs10941012  
rs34970263  
rs10941011  
rs34970260  
rs10941010  
rs34970261  
rs10941016  
rs10941015  
rs10941014  
rs10941013  
rs10941008  
rs34970259  
rs10941009  
rs10941006  
rs34970257  
rs10941007  
rs34970256  
rs34970255  
rs34970253  
rs10250629  
rs10250635  
rs10250633  
rs10250639  
rs10250638  
rs10250637  
rs10250636  
rs10941025  
rs10941024  
rs34970270  
rs10250631  
rs10941021  
rs10250630  
rs10941023  
rs34970274



rs10941022  
rs16976094  
rs34970264  
rs34970269  
rs10941017  
rs10941018  
rs16976099  
rs16976090  
rs10250646  
rs10250647  
rs10250649  
rs10250640  
rs34970285  
rs10250642  
rs10250641  
rs10941028  
rs2611133  
rs2611134  
rs2611135  
rs2611136  
rs2611130  
rs2611131  
rs2611132  
rs10250542  
rs10250543  
rs10250540  
rs10250546  
rs10250547  
rs10250544  
rs10250545  
rs2611138  
rs2611137  
rs10250548  
rs10250549  
rs2611139  
rs2611146  
rs2611147  
rs2611144  
rs2611145  
rs2611142  
rs2611143  
rs2611140  
rs2611141  
rs10250530  
rs10250532  
rs10250533  
rs10250534  
rs10250535



rs10250538  
rs10250539  
rs2611149  
rs2611148  
rs2611151  
rs2611152  
rs2611153  
rs2611154  
rs2611155  
rs2611150  
rs10250564  
rs10250565  
rs10250562  
rs10250563  
rs10250560  
rs10250561  
rs10250569  
rs10250567  
rs2611167  
rs10250551  
rs10250552  
rs10250554  
rs10250550  
rs10250555  
rs10250556  
rs10250590  
rs2611178  
rs2611179  
rs2611174  
rs17819282  
rs10250588  
rs10250582  
rs10250580  
rs10250587  
rs10250584  
rs2611183  
rs2611182  
rs2611181  
rs2611180  
rs17819276  
rs17819273  
rs17819270  
rs2611187  
rs2611186  
rs2611185  
rs2611184  
rs10250578  
rs10250577



rs10250579  
rs10250570  
rs10250571  
rs10250574  
rs10250576  
rs10250575  
rs2611194  
rs2611193  
rs2611196  
rs2611198  
rs2611197  
rs2611199  
rs17819299  
rs17819290  
rs17819294  
rs10250599  
rs10250596  
rs10250598  
rs10250592  
rs10250591  
rs10250594  
rs10776917  
rs10776916  
rs10776919  
rs10776918  
rs10776913  
rs10776912  
rs10776915  
rs10776914  
rs10776911  
rs10776910  
rs11200632  
rs11200633  
rs11200634  
rs11200635  
rs11200630  
rs11200631  
rs35748000  
rs35748001  
rs11200636  
rs35748007  
rs11200637  
rs35748004  
rs11200638  
rs11200639  
rs10776909  
rs10776908  
rs10776907



rs10776906  
rs10776905  
rs10776904  
rs10776903  
rs10776902  
rs10776901  
rs10776900  
rs11200645  
rs11200643  
rs11200644  
rs11200641  
rs11200640  
rs11200649  
rs11200647  
rs10776931  
rs10776930  
rs10776933  
rs10776932  
rs10776935  
rs10776934  
rs10776936  
rs11200650  
rs11200651  
rs11200652  
rs11200653  
rs11200654  
rs11200655  
rs10776929  
rs10776928  
rs10776927  
rs10776957  
rs10776958  
rs10776959  
rs10776953  
rs10776954  
rs10776955  
rs35748046  
rs35748045  
rs35748044  
rs17819204  
rs35748049  
rs10776960  
rs10776961  
rs35748043  
rs2611789  
rs2611788  
rs11498383  
rs2611787



rs2611786  
rs2611766  
rs2611767  
rs17819966  
rs17819969  
rs17819973  
rs2611759  
rs2611757  
rs2611758  
rs2611755  
rs2611753  
rs2611754  
rs17819978  
rs17819977  
rs17819985  
rs17819983  
rs17819945  
rs17819944  
rs17819946  
rs17819950  
rs17819964  
rs17819962  
rs2611721  
rs2611720  
rs2611719  
rs2611718  
rs2611717  
rs2611716  
rs2611715  
rs2611713  
rs11498331  
rs11498327  
rs11498329  
rs2611700  
rs2880981  
rs2880985  
rs2880984  
rs2880983  
rs2880982  
rs2880989  
rs2880988  
rs17819991  
rs4324798  
rs2880987  
rs17819994  
rs2880986  
rs17819999  
rs2880990



rs11498306  
rs11498305  
rs2880992  
rs11498304  
rs2880991  
rs2880994  
rs11498302  
rs2880993  
rs2880996  
rs11498300  
rs2880995  
rs2880998  
rs2880997  
rs2880999  
rs2880968  
rs2880969  
rs2880964  
rs2880966  
rs2880967  
rs28860525  
rs2880962  
rs2880963  
rs28860522  
rs28860520  
rs2880979  
rs2880977  
rs2880978  
rs2880976  
rs28860532  
rs2880970  
rs28860533  
rs28860539  
rs2880943  
rs2880944  
rs2880945  
rs2880947  
rs2880948  
rs2880949  
rs28860545  
rs28860543  
rs2880940  
rs28860546  
rs2880941  
rs2880939  
rs2880955  
rs2880956  
rs2880953  
rs2880954



rs28860554  
rs28860553  
rs28860555  
rs2880951  
rs2880952  
rs28860568  
rs28860567  
rs28860565  
rs2880925  
rs17819932  
rs2880924  
rs28860560  
rs2880927  
rs2880926  
rs2880921  
rs2880923  
rs2880922  
rs17819928  
rs17819929  
rs2880917  
rs17819924  
rs2880918  
rs2880919  
rs2880930  
rs2880938  
rs2880937  
rs17819940  
rs28860573  
rs2880936  
rs2880935  
rs2880934  
rs2880933  
rs2880931  
rs2880928  
rs2880929  
rs28860588  
rs28860589  
rs28860586  
rs2880901  
rs2880900  
rs28860584  
rs2880903  
rs2880902  
rs28860582  
rs2880905  
rs28860583  
rs2880904  
rs28860598



rs28860599  
rs28860590  
rs28860591  
rs2880910  
rs2880916  
rs2880915  
rs2880914  
rs2880909  
rs2880906  
rs17819913  
rs17819914  
rs17819919  
rs12109726  
rs12109728  
rs12109730  
rs12109733  
rs12109713  
rs12109718  
rs12109719  
rs12109716  
rs12109722  
rs12109720  
rs12109745  
rs12109746  
rs12109747  
rs12109753  
rs12109755  
rs12109754  
rs12109751  
rs12109750  
rs12109737  
rs12109735  
rs12109744  
rs12109743  
rs12109742  
rs3117582  
rs12109740  
rs12109767  
rs12109770  
rs12109771  
rs12109773  
rs12109774  
rs12109756  
rs12109789  
rs12109791  
rs12109798  
rs12109792  
rs12109793



rs12109794  
rs401681  
rs12109778  
rs12109780  
rs12109786  
rs12109784  
rs12109781  
rs12109782  
rs11498464  
rs2611695  
rs2611697  
rs2611696  
rs11498468  
rs11498469  
rs11498470  
rs2611698  
rs11498471  
rs2611699  
rs11498474  
rs11498453  
rs11498459  
rs17819796  
rs11498476  
rs11498481  
rs11498480  
rs2611650  
rs2611651  
rs2611652  
rs2611653  
rs2611655  
rs2611654  
rs2611657  
rs2611656  
rs2611667  
rs12109711  
rs12109708  
rs12109707  
rs12109705  
rs12109703  
rs12109702  
rs17819824  
rs2611629  
rs2611625  
rs2611626  
rs2611627  
rs2611628  
rs17819831  
rs17819839



rs2611649  
rs2611643  
rs2611644  
rs2611645  
rs2611640  
rs2611642  
rs2611641  
rs28860484  
rs17819851  
rs2611638  
rs2611639  
rs2611636  
rs2611637  
rs2611634  
rs2611635  
rs2611632  
rs2611633  
rs2611631  
rs2611630  
rs28860498  
rs28860496  
rs28860495  
rs11498406  
rs17819861  
rs28860492  
rs28860490  
rs17819872  
rs11498418  
rs11498417  
rs2880869  
rs2880860  
rs11498414  
rs11498416  
rs11498415  
rs2880862  
rs2880861  
rs11498430  
rs2880877  
rs2880876  
rs11498429  
rs17819886  
rs11498428  
rs11498427  
rs2880871  
rs2880870  
rs2880873  
rs2880872  
rs2880875



rs11498421  
rs2880874  
rs2611604  
rs2611603  
rs2611605  
rs11498440  
rs11498441  
rs2611601  
rs11498439  
rs2880888  
rs2880887  
rs2880886  
rs2880885  
rs11498452  
rs2880899  
rs2880898  
rs2880894  
rs2880897  
rs2880893  
rs11498447  
rs12109686  
rs12109683  
rs12109682  
rs12109685  
rs12109684  
rs2880826  
rs2880819  
rs12109699  
rs12109698  
rs2880830  
rs12109696  
rs2880831  
rs12109695  
rs12109694  
rs12109693  
rs12109691  
rs2880832  
rs12109690  
rs2880833  
rs2880838  
rs2880839  
rs2880829  
rs2880840  
rs2880841  
rs2880847  
rs2880848  
rs2880849  
rs2880844



rs2880845  
rs2880846  
rs2880852  
rs2736100  
rs2880853  
rs2880850  
rs2880851  
rs2880856  
rs2880854  
rs2880855  
rs28860469  
rs28860477  
rs28860479  
rs2880804  
rs2880803  
rs2880806  
rs2880802  
rs2880801  
rs17819809  
rs2880817  
rs2880816  
rs2880814  
rs2880812  
rs2880811  
rs2880810  
rs17819816  
rs2880809  
rs17819812  
rs2880807  
rs2880808  
rs34970627  
rs34970626  
rs34970625  
rs34970624  
rs34970620  
rs2880251  
rs2880254  
rs2880255  
rs2880253  
rs2880256  
rs34970637  
rs2880257  
rs34970632  
rs34970630  
rs34970605  
rs2880223  
rs34970608  
rs2880220



rs34970607  
rs2880222  
rs34970601  
rs34970600  
rs2880228  
rs2880229  
rs34970614  
rs2880236  
rs34970613  
rs34970616  
rs2880234  
rs34970615  
rs2880235  
rs34970618  
rs2880232  
rs34970617  
rs2880230  
rs34970619  
rs2880231  
rs34970610  
rs2880238  
rs34970611  
rs2880280  
rs2880281  
rs2880282  
rs17819694  
rs2880286  
rs2880285  
rs2880287  
rs17819692  
rs2880294  
rs2880295  
rs2880292  
rs2880293  
rs2611590  
rs2880299  
rs2880298  
rs2880297  
rs2880296  
rs10776500  
rs2880260  
rs17819677  
rs17819672  
rs2880268  
rs2880267  
rs2611584  
rs2880269  
rs2611585



rs2880264  
rs2880266  
rs2880265  
rs2880272  
rs2880273  
rs2880270  
rs2880271  
rs17819684  
rs17819683  
rs2880276  
rs2880275  
rs2611571  
rs2880274  
rs2611572  
rs11498962  
rs11498963  
rs11498966  
rs11498967  
rs17819700  
rs10776494  
rs10776493  
rs10776490  
rs11498941  
rs10776485  
rs11498942  
rs10776484  
rs11498943  
rs10776483  
rs11498944  
rs10776482  
rs10776489  
rs10776488  
rs10776487  
rs11498940  
rs10776486  
rs10776481  
rs11498947  
rs11498932  
rs10776471  
rs10776474  
rs11498931  
rs10776473  
rs10776475  
rs11498939  
rs11498937  
rs28860770  
rs28860772  
rs28860777



rs28860773  
rs11498915  
rs11498914  
rs11498913  
rs11498912  
rs28860768  
rs28860769  
rs28860790  
rs28860793  
rs28860795  
rs28860783  
rs28860786  
rs28860785  
rs28860789  
rs17819779  
rs34970699  
rs34970698  
rs34970696  
rs34970694  
rs34970693  
rs17819759