HIC/OMERO DATA MANAGEMENT PROPOSAL (DRAFT)

Version: 1.0 (last modified 01/03/2011)

Author: Jason Swedlow, Andy Judson

Document rationale: Initiate definition of pilot project for HIC research data centre infrastructure.

Project business case: The University of Dundee has a proven track record of facilitating high quality research in epidemiology and health outcomes work, particularly around diabetes. We have over recent years established a strong reputation within pharmacogenetics and pharmacoepidemiology. To support these research themes it is essential that researchers are able to work with large population based health datasets, such as those managed by The Health Informatics Centre (HIC) and the Scottish Diabetes Research Network (SDRN). In the wider public context however repeated losses of sensitive personal information have highlighted the importance of data security and how it is shared for research. Since potential risks arise because anonymised datasets are released to researchers for analysis on their own computers, the "Safe Haven" approach should help to address these concerns. By holding data centrally and thus reducing the number of copies of data in circulation, the risk of data loss is reduced, more powerful computing resources are available, and collaboration between researchers can be improved. The release of data to researchers (and as flat files) is a historical approach in health informatics research, but with more researchers working with larger datasets this model isn't scalable, nor is it as secure as technical solutions can now support.

In 2008, the Data Sharing Review report by Richard Thomas, the Information Commissioner and Mark Walport, the head of the Wellcome Trust, recommended Safe Havens as "environments for population based research and statistical analysis in which the risk of identifying individuals is minimised". Following this in 2009, the House of Lords (Science and Technology Committee) chaired by Lord Patel produced a report (paper 107) on the opportunities that genomic medicine could provide to the UK. This report was responded to by the previous government in December 2009 and was debated again with the new coalition government in June 2010. In the report and the recent debate, the University of Dundee is described (by Mark Walport) as an exemplar, having a very powerful environment for informatics research. Of particular interest in the report are the recommendations:

- R8.23 - establishing a new institute of biomedical informatics to address the challenges of handling the linking of medical and genetic information.
- R8.29 - use of safe havens for supporting data sharing as outlined in the UK Government (Ministry of Justice) Data Sharing Review report (June 2008) and the adoption of the governance approaches developed by UK BioBank.

Reading and listening to the House of Lords evidence, it is clear that genomic medicine has the potential to be an important area of health research. It is also true that Dundee University is already well placed in terms of health informatics and genomic research and thus has a good foundation to establish itself as centre for genomic-medicine. The previous and current governments both recognise the importance of this area of research but are unwilling (at this time) to fund such initiatives. Therefore, we have an opportunity to develop a proof-of-concept infrastructure that addresses existing issues around data security/governance and also demonstrates the linkages of genomic-phenotypic data, and thus puts us in a strong position for future funding streams.

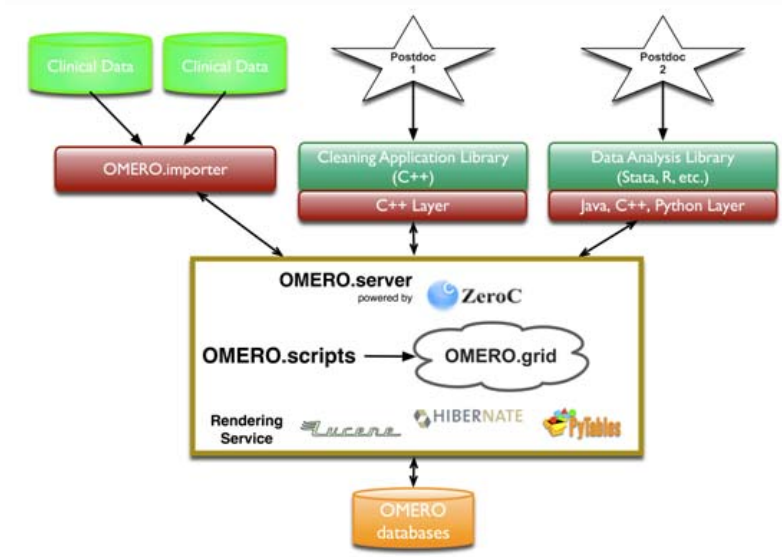
If this project is to succeed we need to have the support of the researchers. So, as well as the governance issues we have already discussed, we are keen that the safe haven environment is enabling to the research community. To help understand the needs of the researchers we held focus groups sessions with three research groups, the key points raised are listed in Appendix 3.

Scale of project: Initially, to enable data processing, management, collaboration and analysis for a set of research questions. Model assumes multiple post docs, with data cleaning and data analysis tools, in different environments, accessing data procured from existing HIC, SDRN and GODARTS data structures.

Major goals: Demonstrate the processing of anonymised linked clinical data based on HIC, SDRN and GODARTS repositories by ***existing, scientist-generated tools, through a secure software interface and a centralised data repository.*** Access to the data will be subject to user permissions and normal de-identification requirements.

User experience: As much as possible, the experience of the user running their existing processing routines through the OMERO interface should be as similar as possible to running these on files on a local file system in their preferred analysis package. In this ideal, the research user is provided with a OMERO plugin for their preferred analysis package. This plugin is installed to the user's desktop instance of e.g. Stata. The analyst would then use Stata to load the data via the OMERO plugin, just as they would load a flat file or load data from an ODBC connection. The researcher would then just analyse the data in the same way they do now, writing scripts and deriving subsets of data or results. However, the actual processing of the data is to be performed on the server not on the analyst's client computer and the derived sets of data and scripts used are stored centrally on the server. Over time some cleaning scripts that analysts have developed should be translated into python so that the routines can be shared and/or pre-applied to the data - possibly with different assumptions between users. As much as possible, it is expected that the developers and not the analysts are writing the python scripts.

Proposed architecture: A summary of OMERO is in Appendix 1, and shown below are the key components colour coded to highlight where development work for this pilot is required. Objects in bright green are unchanged; as color changes towards red more work is estimated to be required.



Major work packages:

- *OMERO.importer: Automatic upload of existing clinical data into OMERO*. New scripts will be built to import the clinical datasets into an OMERO context.
- *OMERO.server: Modifications of tables and API to support HIC data.* The simplest method involves adding new tables to the OMERO database to support HIC data. However, this approach is less desirable as it means we create a custom version of the OMERO database and API which must be continuously maintained in the future, and it defines, before we know all possible queries, the set of queries that are possible or perform well. A preferred alternative uses so-called NOSQL approaches, which are currently being added to OMERO to handle a range of different data types. We will have preliminary implementations in place at project start that can synergise with the project proposed here.
- *Processing & analysis: adaption of legacy code to OMERO API.* A key goal of this project is the continued use of analysis tools and environments favored by HIC scientists. Indeed, the goal of the projects is to have two or more scientists, generating new datasets (from the underlying raw data), sharing that data with a colleague and enabling new analysis, with no fundamental change in the analysis application. To do this, the interfaces these applications will need to be adapted to pull data from OMERO, either by wrapping command line calls with Python, or adding in modules (especially in Stata) that mediate that connection.

Related (parallel / interim) local projects:

- **HIC Citrix environment:** to provide a secure remote-access environment, initially hosting undergraduate student projects and national Childsmile dental data. HIC proposes to test this environment over a year, in parallel with the main development (outlined above) in terms of the user and technical experience. This is not expected to be a long-term solution for the larger datasets but it may be used for smaller studies and/or to provide additional security constraints to the OMERO infrastructure.
- **SDRN Linux environment:** to provide a short-term / intermediate alternative to flat-file transfer and processing by federated research teams. The proposal is to install a Oracle SQL Server instance within the existing Linux server used by Helen Colhoun's team. Data will be loaded into this by the developers instead of being released as flat files. The developers (AJ/JK) will work with the analysts to migrate some example cleaning routines from their legacy code (Stata/R) to Python scripts as these should be reusable in the OMERO environment. Initially this will be setup to support a specific question, but if appropriate it could be scaled up to hold a central copy of the baseline SDRN dataset, accessible to the other SDRN analysts. As above, this is expected to be replaced by the OMERO model.

Proposed datasets to be included:

- The key datasets to be included (probably not all for the pilot) are listed below, with more information in Appendix 2.
    - **HIC** (CHI, GRO,RX, BIOCHEM, SMR01, SMR02, SMR06)
    - **SDRN** (SCI-DC, SMR01, SMR06, RENAL, …)
    - **GODARTS** (SCI-DC, GENETICS, RX, GRO, SMR01/06, …)
- In most cases these are national datasets, but for the purpose of the demonstrator we will work with Tayside subsets. Most of the datasets listed are single tables of patient event histories. The SDRN and GODARTS datasets are however purpose built linked datasets for diabetes research which include data from some of the other sources listed.
- In the case of the SDRN data, the research teams have access to the whole dataset. Whereas, with the HIC data, subsets are prepared for specific projects. In either case researchers may need to link to a new external datasets not defined above. For instance, one team of SDRN researchers may be working on a study involving just the SCI-DC (diabetes registry) data, another team may want to do a study that links to SMR06 (cancer registry), and another team may want to use another dataset which we've never seen before.

Synergies:

The problem of integrating heterogeneous critical data is one that is timely, and being addressed by many different projects. The current solutions have made some progress, and are good candidates for testing in this problem. For the next six months at least, 1-2 OME developers will be working on this problem, and at least one other developer will be working on the problem of distributed computation. OME's goal is always to build *reusable, multi-functional solutions*, so the project described here synergises with efforts across the project. This means that research, testing, documentation, and integration all can directly benefit from work on the rest of team, even though it is not directly dedicated to this project.

Work to come back to (depending on time):

- Example analyses including clinical-genomic data linkages.
- Metadata portal (dataset documentation, knowledge exchange).
- Data cleaning algorithms & libraries (port data cleaning / modeling scripts).
- Open access portal (similar to the zebra fish or 1958 birth cohort portals).

Resources:

- Three developers, Grade 7/8, starting Jan 1. These staff are full-time, completely, and working as developers on the project. They are also definitely new appointments - everyone currently involved in OMERO is working flat out on other aspects of the infrastructure, which we will need. In fact, we will be getting time from 2-3 others who are also working on the alternative storage stuff, for other projects, and can leverage the work of many others on the team. Our aim is to make a single solution that makes HIC happy, but also works for HCS, FLIM, and a variety for other things.
- Three laptops. (£1900 each).
- One server (£2900).

Schedule:

- Project start is March/April 2011 and will run for 12 months.

References:

- http://www.justice.gov.uk/reviews/datasharing-intro.htm
- http://www.parliament.uk/business/committees/committees-archive/lords-s-t-select/genomic/
- http://openmicroscopy.org
- http://trac.openmicroscopy.org.uk/omero/wiki/AlternativeStorage
- http://research.nesc.ac.uk/node/590
- http://research.microsoft.com/pubs/64537/tr-2005-10.pdf
- http://www.hdfgroup.org/why_hdf/
- http://www.pytables.org

APPENDIX 1: WHAT'S AN OMERO AND WHY SHOULD I CARE?

INTRODUCTION

Modern biology depends on the collection, management, and analysis of comprehensive datasets. Unlike the concerted efforts that drove the completion of the first genome projects, most current datasets derive from a specific experiment—they are generated to reveal molecular mechanisms, networks, interactions or phenotypes during a biological transition or after some perturbation. While the range of these experiments is diverse, they all require facilities for managing and analysing data. In many cases, resources for data management become limiting, and thus can define what is achievable in a data-dependent experiment.

Traditionally, most data intensive research projects have recorded data on filesystems, and built analysis and visualization applications that read, process, and present the data. This approach is extremely effective, but ignores two critical needs for long-term use of data-- the maintenance of data and the software that accesses it, and the ever-growing scale of the data and the demands on it. Maintenance refers to the needs to keep datasets and software current, aware of updates to the data itself, but also filesystem and operating system versions-- a very technical problem, but one inherent in any computational software infrastructure. Scalability strictly refers to an increase in size (i.e., more data records, more users, etc.). However, in scientific settings, a more expansive definition of 'scalability' is required. As experience with the data grows, more sophisticated questions and facilities are asked of the data—indexing of all text data, new queries that express new, unanticipated scientific questions, new analysis tools and approaches, and association and linkage with new data. These examples of scaling complexity and value must also be considered, and designed into any solution.

These challenges cannot be met by a solution that is 'brittle'-- where the analysis facility is hard-coded to the data. Inevitably, the analysis tool will be built for the scaling relevant at that time, and can't grow without significant rebuilding, with growing demands. To advance and handle increasingly heterogeneous data across larger teams and include facilities for large-scale computational analysis, a different approach must be implemented. That approach involves the use of *middleware*, software applications that abstract the data from the analysis and visualization, and at the same time embed extra value-- security, indexing, distribution of data and compute, and access to alternative means of data access-- all through a single common interface (technically referred to an *application programming interface*, or API). In addition, middleware enables the use combined data storage strategies -- relational schemes, flat files, annotations, and so-called NO-SQL approaches—which are all required to satisfy the broad requirements of modern scientific data analysis.

OMERO

Since 2005, we have developed and released OME Remote Objects (OMERO), a multi-component data management platform comprised of a number of individual modules. OMERO.server is a Java server application that mediates communication with a number of databases based on PostgreSQL (http://www.postgresql.org/), Lucene (http://lucene.apache.org), HDF5 (http://www.hdfgroup.org/), and an image file store. OMERO.server uses Hibernate (http://www.hibernate.org/) for object-relational mapping, Lucene for text indexing, and ICE (http://www.zeroc.com/) for object remoting. OMERO.server initially used JBOSS for remoting (http://jboss.org/), but the simplicity, stability and cross-platform support in ICE has meant that that this single framework could be used for all aspects of the OMERO application programming interface (API). This single API supports simultaneous access from Java, C++ and Python clients and can be extended to many other frameworks. This abstraction of the underlying data from the analysis allows the addition of additional functionality (indexing, data rendering, annotation), adding value to the data without any specific coding by the analysis developer. Furthermore, as the database modeling evolves, effects on the API can be minimised, to

ensure that legacy code still runs. OMERO was originally built to serve image data, but is increasingly used to serve non-image types of data as well.

THE IMPORTANCE OF A COMMON API

A common argument in data management systems is the balance between direct access to data, as files stored on a filesystem or in a database, and an abstracted view of data through an API, as in OMERO. Accessing an array of files on a file system or records in a database is usually easier in the short term, and provides the most flexibility for the developer. This approach is certainly ideal for a small research team where the requirements are completely defined and very little expansion of the system will ever be necessary, but includes substantial limitations on scaling and interoperability {Gray, 2005 #1592}. As data management systems grow, more developers and users join the system, the software that accesses the data becomes more heterogeneous, the requirements for controlling access grows, and the complexity for derived and associated data increases as well. The simple computational costs of searching millions of files (or more) grows as well, thus burdening the data management system. While sophisticated file storage and access mechanisms like HDF5 and netCDF so called 'nascent databases' {Gray, 2005 #1592} are now available, incorporating applications that add value to data— access control, organization, annotation, indexing, rendering, and processing— will always be necessary. As this sophistication grows, building and maintaining the custom linkages between the various applications that directly access files becomes ever more difficult. Tiered applications that abstract data access and processing and enable access through a common API are used throughout business and science for management, visualization and analysis of complex, heterogeneous data. In our experience, sophisticated data storage mechanisms like HDF5 can be incorporated into tiered applications, accessed through the API, and used to deliver specific functional enhancements, e.g., storage of large tabular arrays (>1M records).

ANNOTATING DATA WITH DATA

Analysis of large primary datasets inevitably generates new sets of data. Acquired images may be processed to improve contrast and generate a second set of derived images, or analysed to generate calculated features derived from the objects in the image—segmented objects, fitted functions, etc. In OMERO, generated images can be stored within the same hierarchy structures as the original image, and annotated appropriately. However, many analyses generate derived measurements that are not images themselves, but can be represented as defined data objects. In OMERO, structured files (.xml, .xls, etc.) can be stored as 'StructuredAnnotations', defined data structures stored as files in an OMERO file repository, and linked from the database (http://trac.openmicroscopy.org.uk/omero/wiki/StructuredAnnotations). The OMERO API exposes this linkage, allowing any OMERO client to access the original data and the derived measurements.

A specific use of OMERO's StructuredAnnotations involves the implementations of specific ontologies for data annotation. Biological image data is still heterogeneous, and there are not yet defined specifications for annotating microscope image data. To help begin this process, we have implemented support for defined OBO ontologies within OMERO.server, and access to them through the OMERO API. We expect that as the use of these ontologies develops and software tools for image annotation improve, the specific ontologies used can be harmonised to start to enforce common annotations across all images in different domains.

Many derived measurements are of the form of large tabular data and are normally stored as .csv or .xls files. In principle these data would be well-suited for inclusion in OMERO's database, but the data must be clearly defined for future recall and query. A very common use case is the storage of results from multiple segmentation algorithms, defined and chosen by a user. Our experience with the OME Server suggested that updates to the database by anyone not fully adept at data modelling (e.g., 'Segmentation Method 1', 'Segmentation Method 2', etc.) inevitably led to unusable data. In addition, a standard use case in high content screening (HCS) is the calculation of >1M measurements per plate (384 wells x 10 images/well x 25 cells/image x 20 calculated features/image). Importing many millions of calculated values into a relational database, in the absence of any specialised

hardware or database configuration, thus imposes an unacceptable performance burden. As each institution or imaging facility uses its own algorithms and structures for data storage, analysis and mining, we developed a simple flexible framework for storing tabular data and ROIs known as OMERO.Tables (http://trac.openmicroscopy.org.uk/omero/wiki/OmeroTables). This facility uses an HDF5-based data store and a PyTables interface (http://www.pytables.org) to provide a flexible structure for naming, storing and accessing data stored in tables within an OMERO server. The OMERO API has been updated to provide access to this facility. For example, to store and query measurements of counts of nuclei per well at specific cell cycle stages:

1. create the table columns

```
iId = ImageColumn('Image_id', 'Image ID', list())
nucCount = DoubleColumn('Nuclear_Count, '', list())
stage = StringColumn('cell_cycle_stage', 'Cell cycle stage', list())
```

2. populate values

```
iId.values.append(100L)
nucCount.values.append(47)
stage.values.append("Metaphase")
```

3. create the table

```
columns = [iId, nucCount, stage]
table = session.sharedResources().newTable(1,'/mytable.h5')
table.initialize(columns)
hdfFile = table.getOriginalFile()
```

4. query the table

```
array = table.readWhere('(nucCount < 50) && (cell_cycle_stage == "Metaphase")')
```

OMERO.Tables unifies the storage of columnar data from various sources, such as automated analysis results or script-based processing, and make them available within OMERO.

METADATA FOR COMPUTATIONAL EFFICIENCY

There is general agreement that metadata preservation and access is a foundation for efficient and effective data analysis and mining {Schofield, 2009 #1528}{Field, 2009 #1530}. Definitions of the structure and history of data are needed to provide context for analysis calculations (size and dimensions, etc.) and also the ability to search and cluster data (source, date, sample, acquisition conditions and settings, etc.). However, an additional requirement for metadata, and efficient identification and recall of its components emerges as the size and complexity of analysis applications grows. As an example, for genotype imputing calculations, the computation load changes depending on the density of SNPs in any specific region of the genome (). As most compute resources are shared resources, either within a department cluster or in the Cloud, the distribution and monitoring of compute jobs depends on knowledge of the metadata, that describe characteristics of the data. A second example involves the HCS example above. One plate is a container for 3840 images, each 2 MB in size (1024 x 1024 pixels x 2bytes/pixel). Distributing the segmentation and feature calculations for one plate across many thousands of cores is possible, but the simultaneous I/O calls for thousands of image files can cause performance of many file systems to significantly degrade. Some throttling of job distribution is thus necessary, and again this requires access to metadata-- throttling strategies will differ for different sized images. Meta-calculations are therefore required to define the structure of and perform the main calculation. This can only be achieved if metadata are linked to the original data and can easily be recovered and used. Scanning, for example, many HDF5 files for individual quantities is inefficient, because of the overhead of open and closing many thousands or millions of files.

OMERO CLIENTS

The goal of OMERO is to enable interoperability, allowing access by software applications, developed without any awareness of OMERO, and to build data analysis and visualization applications that deliver real scientific value. To this end we have built our own OMERO clients that enable functionality that we require in our own work, and connected other open image analysis applications to OMERO. The functionality in these tools is not meant to be exhaustive, but rather to demonstrate how OMERO can be used, and to demonstrate integration and re-use of other, third party software tools.

We have designed and built three OMERO client applications. Two are Java applications (requiring Java 1.5 or 1.6). OMERO.importer is a desktop application that imports proprietary image data files using Bio-Formats from a user-visible filesystem into an installation of OMERO.server. OMERO.insight enables data management, annotation, visualization and analysis (Fig 2A and Suppl Fig 1). OMERO.web is a web browser-based client application that provides access to OMERO's data hierarchies, and enables annotation and visualization of data. OMERO.web exposes the annotation-based data sharing mechanism where user-defined sets of data can be shared and discussed with other users on the OMERO system (Fig. 2B). A full list of the functionality in these clients is available at http://www.openmicroscopy.org/site/products/feature-list.

To demonstrate OMERO's flexibility and interoperability, we have built specific bridges between OMERO and Image J (OMERO.ij; http://openmicroscopy.org/site/support/omero4/downloads) and Cellprofiler (http://trac.openmicroscopy.org.uk/omero/wiki/OmeroCellProfiler), two popular open tools image processing tools, thereby making these tools clients of OMERO. Moreover, we have used an existing Java gateway in Matlab, the popular scripting tool for data analysis, to read and write data from OMERO (http://trac.openmicroscopy.org.uk/omero/wiki/OmeroMatlab). This opens any of the functionality within Matlab to be directed towards data within an OMERO installation. As an example, Figure 3A shows a series of image segmentation and analysis tools implemented in Matlab, analysing data in OMERO. We have also integrated VolViewer, an open source GPU-based 3D volume rendering tool with OMERO (Figure 3B), thus enabling a powerful image rendering facility access to any data that be stored in OMERO.

APPENDIX 2: DATA MODELS

HIC DATASETS (TAYSIDE)

CHI (demography)

Every person in Scotland is assigned a unique identifier (Community Health Index) at the time of initial registration with a primary care doctor under NHS Scotland.

- 998,075 rows, 1 per patient, for Tayside.
- Fields: chi, master chi, sex, names (current & previous), address (current & previous), date address changed, date of birth and death, health board residence, registered gp code (current & previous), date in/out health board.
- Time period: 1994 - current.
- Monthly snapshots are received; 2% growth per year.

GRO (deaths)

Death data is obtained from the official government record. The General Register Office for Scotland is a department of the devolved Scottish Administration responsible for the registration of births, marriages and deaths. Note this data is not CHId. HIC auto-populates the CHI based on name/address/postcode etc and data entry complete any left using a purpose-built application.

- 105,106 records for Tayside
- Fields: sex, date of birth and death, social class, occupation, place of birth and death, country of birth and residence, cause of death (both written and coded), etc.,
- Time period: 1989 – current.
- Quarterly downloads are received, with a 3 month-lead time; 2% growth per year.

RX (prescribing)

Since 1993, HIC has been logging all Tayside community dispensed prescriptions against a persons' unique identifier (CHI). This longitudinal dispensed prescribing database is unique in the UK and makes HIC data an ideal resource for drug utilisation studies.

Prior to 2004, the paper prescriptions were scanned by a team of data entry staff (located within the Clinical Information Bureau) who used purpose-written software to manually enter the missing prescription details, including the patient CHI number, prescription date and medical directions (dosage).

Since 2004, all Tayside prescription data has been obtained electronically from Practitioner Services Division (PSD). PSD are responsible for the processing and pricing of all prescriptions dispensed in Scotland. All these prescriptions are dispensed by community pharmacies, dispensing doctors and a small number of specialist appliance suppliers. GPs write the vast majority of these prescriptions, with the remainder written mainly by nurses and dentists. They also include prescriptions written in hospitals that are dispensed in the community, but exclude drugs dispensed within hospitals themselves.

The data received from PSD includes both scanned images of the prescriptions and the data - the output of the Optical Character Recognition (OCR) process at PSD. This electronic feed has replaced the need for the physical transfer and processing of paper prescriptions. Purpose-written software automatically completes a large amount of the missing / incomplete data, although some manual

data entry is required, particularly for those handwritten scripts where the CHI and/or date was not picked up by the OCR process or was not available. Date and medical directions (dosage) can be added by the data entry team on a project-specific basis.

- 88M rows, 1+ per patient.
- Fields: chi, drug name, drug code, prescribed date, batch date, bnf code, line no, quantity, directions, no. of packs, strength, formulation, gp code, pharmacy code.
- Time period: 1993 – current (with gaps between 1997-1999 & 2002-2003).
- Monthly downloads are received of scanned images and data, with approximately a 3 month lead time from a prescription being issued; 15% growth per year.

BioChemistry (clinical chemistry & haematology)

The Tayside laboratory (MasterLab) system records all tests performed in surgeries, clinics, hospitals etc which have been sent to the Tayside labs for processing. This system covers biochemistry, haematology, immunlogy, microbiology and virology disciplines. The information recorded includes date of test, test(s) performed and the results of the test(s). HIC receives quarterly downloads of this data. Some common biochemistry tests include cholesterol, triglycerides, glucose, liver function, urea, HDL, LDL and thyroid. Some common haematology tests include FBC (full blood count), WBC (white blood count) and INR (international normalised ratio).

- 113M rows, 1+ per patient,
- fields = chi,lab number, test code, dt test, result, source, comments, units, hi/low range, sample type
- Time period: 1992 – current
- Daily downloads are received, 14% growth per year.

SMR01 (outpatients)

SMR01 is an episode-based patient record relating to all inpatient acute stay hospital admissions. A record is generated when a patient completes an episode of inpatient or day case care. Examples include discharge home from hospital, transfer to another clinician (either at the same or a different hospital), a change of specialty (either under the same or a different clinician), or death.

Data collected includes patient identifiable and demographic details, episode management details and general clinical information. Currently diagnoses are recorded using the ICD-10 classification and operations are recorded using the OPCS-4 classification. Information such as waiting time for inpatient/day case admission and length of stay may be derived from the episode management data.

- 2.5M rows, 1+ per patient.
- Fields: chi, date admission, date discharge, location, episode record key, speciality, significant facility, admission type, admission reason, admission transfer from/to, discharger transfer from/to, diagnosis codes (6), operation codes (8) plus dates, simd.
- Time period: 1981 – current.
- Annual downloads are received, but because of the extensive validation applied to this data set, there is approximately a 6 month lead time; 4% growth per year.

SMR02 (maternity)

SMR02 is an episode-based patient record relating to all maternity related hospital admissions. A record is generated when a patient completes an episode of inpatient or day case care. Examples include discharge home from hospital, transfer to another clinician (either at the same or a different hospital), a change of specialty (either under the same or a different clinician), or death.

- 140,000 rows, 1+ per patient,
- Fields=chi Mother, dt delivery, chiBaby(1-3), height baby, weight baby, sex baby, agpar baby, crown_heel baby, mode delivery, outcome of pregnancy, operation codes plus dates, LMP, dt conception
- Time period: 1975 – current
- Annual downloads are received, 4% growth per year.

SMR06 (cancer)

SMR06 contains information on all new cases of primary malignant neoplasms, carcinoma in situ (including grade III cervical intra-epithelial neoplasia), neoplasms of uncertain behaviour and (since 1 January 2000) benign brain and spinal cord tumours arising in residents of Scotland. For patients diagnosed up to 31 December 1996 a limited data set was collected. For patients diagnosed from 1st January 1997 onwards, an extended data set, including information on stage, (for breast, colorectal and cervical cancer) and information on treatment was collected for all patients.

Data quality is monitored using routine indicators, computer validation and ad hoc studies of data accuracy and completeness of ascertainment. Data assurance is also achieved through data exchange with specialist tumour registries such as those of the Scottish Melanoma Group and the UK National Register of Childhood Tumours.

- 80,000 rows, 1+ per patient,
- Fields: chi, date incidence, date of death, cause of death (4) diagnosis code, type of tumour, site of tumour, date of radiotherapy, date of chemotherapy
- Time period: 1980 – current.
- Annual downloads are received, but because of the extensive validation applied to this data set, there is approximately a 3 year lead time; 4% growth per year.

SDRN DATASET

The SDRN project has developed a linked dataset which can be used for Scottish-wide diabetes epidemiology research. The primary data set for SDRN is the SCI-DC clinical information system - Scotland's shared electronic health record for Diabetes. SCI-DC gathers data from primary, secondary and tertiary care and makes the electronic record available to relevant members of the healthcare community. Existing SDRN epidemiology studies have linked the SCI-DC data to SMR01, SMR06, and GRO.

SCI-DC was gradually rolled out across Scotland from 2001, by 2003 much of Glasgow, Lothian, Fife and Forth Valley regions were online, by 2006 all regions were online to some degree, and by 2009 all regions and almost all GP practices were linked (unconfirmed estimates of 99% of practices across Scotland). In January 2009, SCI-DC held data on approximately 228,000 patients (4.4% of the population) who are living with diabetes across all 14 Scottish health boards.

SCI-DC is a large and complex clinical database, therefore in SDRN we are deriving a specialised view of the data purely for research. In some cases this simply limits the fields that are available, but in other cases the derived view is a complex translation of the underlying tables to better fit the research analyses. Data variables (tables) of particular interest include diagnosis, demography, biochemistry, and prescribing.

In May 2008, a cohort was defined that returned 274,363 patients. Since then we have extracted data for various studies against this cohort. Each patient will have longitudinal data for most variables, and

some variables (e.g. diagnosis) will also have a summary record (best/most reliable). Below is a summary of some of the key tables (within the SDRN model).

Table	Summary / Longitudinal	Rows (Tayside)	Rows (Scotland)
patient_summary *(links to all other tables by serialno)*	summary	25,620	274,363
cddiabetes	longitudinal	398,626	3,478,215
biochem_hba1c	longitudinal	643,701	3,300,130
bp	longitudinal	1,032,631	8,980,146
misc	longitudinal	1,047,097	9,232,297
prescribing_drug *(links to prescribing_repeats by serialno & prescriptionid)*	longitudinal	3,660,122	29,760,563
prescribing_repeat	longitudinal	16,026,591	206,855,019
eyes	longitudinal	489,094	4,428,411

GODARTS DATASET

The GODARTS project have developed a high quality resource with the successful recruitment of consented patients with type 2 diabetes and matching controls (non diabetics) throughout Tayside. This resource is helping to define genetic factors related to diabetes including susceptibility, complications and response to treatment.

The patient consent at time of recruitment allows the data collected to be linked anonymously to datasets derived from consenting patients' medical records. These datasets include laboratory data (SCI-STORE), hospital admissions (SMR01) and outpatient appointments (SMR00) and the diabetes register (SCI-DC). DNA samples are analysed for their sequencing. Serum samples are analysed for chemical and biochemical components of either established or new biomarkers. RNA samples are analysed for the sequence and relative content of individual RNA molecules.

- Volunteers: approx 10,000 (cases) & 8000 (controls)
- Demographic data: chi, name, address, consent, date of birth, age, gender, ethnicity, ...
- Phenotypic data: type of diabetes, date of diagnosis, family history, fasting, treatment, smoking status, physical activity levels, sbp, dbp, height, weight, bmi, serum cholesterol, hdl cholesterol, ldl cholesterol, serum tryglicerides, serum creatinine, hba1c, glucose, insulin, ...
- Genetic data: imputing to 1 million (750K SNPs per person)
- Linked to: Prescribing (HIC & SCI-DC), SMR01, SMR06, GRO, TOPAS, SCI-DC (other tables), Stroke, etc...

APPENDIX 3 - KEY POINTS FROM RESEARCH USER FOCUS GROUPS

Below, is a summary of the key points from the research user focus groups that were completed as part of the scoping exercise leading to this paper.

- Researchers see a real benefit of a centralised model would be research ready datasets – where cleaning routines have been applied and re-usable scripts/views can be applied easily by researchers and/or developers. These processing routines need to be developed in collaboration with the researchers, validated, documented and signed off. Researchers are then keen to just get the cleaned data (but must be able to go back to the raw data if needed). Idea of 3 stages - 1. Gathering the source clinical data (e.g. SCIDC), 2. Preparing this data for research, 3. Processing the research-ready data. Researcher currently having to spend too much time (estimates of 90%) doing data preparation (stage 2) rather than analysis and interpretation (stage 3) – thus if we can improve this, we can improve the rate of publication etc. So think of a safe haven as an enabling facility for researchers with improved management and governance.
- Researchers need to be able to link data from a range of sources - e.g. ISD, GRO & HIC/SDRN, so subsets of data must be able to move between safe havens.
- Researchers want the flexibility to use a range of analysis packages that could easily change over time. It is also important to them that they are able to upload/install custom modules or develop their own scripts (e.g. for STATA or R). This also includes the idea of working out of hours from a various sites, e.g. at home.
- Researchers recognise the limitations of flat files and are keen to have an improved data management / version control environment built around databases.
- Researchers see collaboration locally and nationally as essential, and that the current data management model is a key factor in blocking this, e.g. duplication of cleaning, knowing who is doing what, knowing who can help, synchronisation of data and cleaning steps, etc.
- Researchers need to be able to use external facilities, e.g. use HAPMAP for imputing genome sequences, or exporting direct genotypes to EBI for them to share with other researchers, or larger computing facilities.
- Researchers need a relational database for the genetic data to define a genotype for a person because so many dimensions. Want to be able to automatically implement data simulation (based on all the dimensions) to define what is best genome for a person at a position - probabilistic, multiple experiments, partial coverage of the population (difference genotypes for subgroups).
- Researchers want to be able to contribute back to community, e.g. open access web portal so others can do lookups across the data (e.g. similar to 1958 birth cohort).
- Researchers want good metadata, e.g. levels of detail, provenance, annotations, discussions, quality metrics, cleaning / processing algorithms, assumptions.
- Researchers want to build a community around the data, e.g. knowing who is doing what, who has worked with this data before, etc – a kin to a social network or stackexchange.
- Researchers working with large datasets, particularly with genomics data, need access to high performance computing (HPC) facilities – locally (e.g. CLS), but also externally (e.g. Oxford/Edinburgh). There is concern we can't guarantee invite resources (storage, bandwidth, processing) – many nodes shared by many users, prioritisation to cliques of researchers.
- Researchers want a remote access infrastructure – none can see how the secure physical model can work within the context of the data we support.

APPENDIX 4: FEEDBACK / DISCUSSIONS

1. What is this delivering for the work we do that cannot be delivered by simply using an SQL query interface?

Although a SQL based approach will work for some of issues, it can't solve all of the problems. A key problem is scalability and analysis performance - A relataion database can handle very large datasets, but when the number of records is over 1 million it does seem to be less efficient than alternative approaches which have been specifically designed for large scale data intensive analysis. The most notably alternative is based on a hierarchical data model (instead of a relational), and is know as HDF-5. The OMERO architecture uses HDF-5 data storage via a PyTables interface. For more information on HDF-5 see http://www.hdfgroup.org/why_hdf/.

You've used flat files to date, and are migrating to a straight database. Even with access via ODBC/JDBC and SQL, you have: a). no method to control access to data, other than giving someone permission to log on system or not; b). no method to include other data (e.g., SNP data, image data, or any others); c). no method to include even the most basic annotations on the data.

A SQL-based approach will also require substantial effort by someone to write, maintain and update the queries for you. Maybe in your project, maybe not, but there will be substantial work, and you will still have no control over the data, now way to directly link heterogeneous data, no method of bringing in external data visualization tools, and no framework for computing on the data.

2. Migrating cleaning routines from Stata/R to python will take up statistician time.

There are various stages to facilitating epidemiology research, sometimes described as a 3-room model. In room 1 - is the source data providers typically clinical systems on the NHS network. In room 2 - is the research data providers it is their role to extract data from the clinical systems, model and release the data for research analysis. In room 3 - are the research analsysts doing what they do. A key problem within the HIC/SDRN research teams is that too much data cleaning is done by the analyst not the data providers in room 2, or even room 1.

The migration of existing Stata/R scripts to python will be on a case-by-case basis and limited to just a few examples for the pilot. The key point though, is that the researcher can continue to develop and use Stata/R scripts and that the migration of the code to python will be done by the development team not the analyst. Migrating a researchers code will no-doubt require their involvement but it is to be kept to a minimum. It is essential though, that analysts try to structure their code in logical ways and use comments (a lot).

3. The architecture assumes that over the longer term cleaning code will be written in python.

The long term maintenance of the python code and developement / migration of new scripts is the responsibility of the developers not the analysts. That is not to say, it wouldn't be useful for any research team member to have python skills.

4. The metadata is key, how is it stored, maintained, and presented?

The researchers will want to know what data is available (e.g. what tables, fields, time period, cohort, data volumes, gaps / issues, etc). What is the provenance of the data (e.g. what source did it originate from, how has been mapped through the systems, how was it derived, etc).

At the moment we store this type of information on websites and or in text files provided with extracts. A big part of the problem is resource (funding / staff) to generate and maintain the information. How would is metadata stored in the OMERO model – is it embedded inside the HDF-5 files?

For the pilot we need to be able to demonstrate how this works.