A NeISS Collaboration to Develop and Use e-Infrastructure for Large-scale Social Simulation

Tom Doherty, Sam Skipsey, Andy Turner, John Watt

Metadata
- Draft version 0.6 (2011-09-26)
- This document was created using Google docs
  - If you are reading an exported version, then consider that there may be a more recent version on the world wide web available via the following URLs:
    - https://docs.google.com/document/d/17YMzMRgL39AsjrKtiL5MKPS17z_r0Y_Wpot8UpjcqSw/edit?hl=en_US&authkey=CL2w9d4I&pli=1
    - http://bit.ly/o5ahF9
    - http://bit.ly/oAJT9j

Contents

Abstract

The National e-Infrastructure for Social Simulation (NeISS) project is focused on developing e-Infrastructure to support social simulation research. Part of the project is providing e-Infrastructure for running contemporary dynamic demographic social simulation models developed in the GENESIS project. There are several types of these GENESIS models. This work deals specifically with population change models driven by mortality and fertility rates that are stochastically applied. In the GENESIS models, each individual human in a population is represented as a distinct digital entity. This type of model is commonly referred to as an individual based model.

A portal based Graphical User Interface (GUI) to the model has been developed as a set of standard JSR-168 portlets. One portlet is for specifying model parameters and setting a simulation running. Another is for comparing the results of different simulation runs. Other portlets are for monitoring submitted jobs and for interfacing with an archive of results. A layer of programs enacted by the portlets stage data in and submit jobs to a Grid computer that enact GENESIS model programs. Once the job is submitted some details are communicated back to a job monitoring portlet. Once the job is completed, results are stored and some part of the results are made available for download and further processing. Collectively we call the system the Genesis Simulator.

This paper and the accompanying presentation at the [UK e-Science All Hands Meeting 2011](#) provide details of the work done to date and outline the next steps we are planning to take. We are aiming to further automate workflows and scale up to model larger populations in more complex ways.

keywords

NeISS, e-Infrastructure, GENESIS, social, simulation, population, model, stochastic, portlet, grid

1. Introduction

Social simulation is an attempt to model societies in a dynamical way in a digital computer [1]. "e-Infrastructure consists of social and technical arrangements around advanced, networked information and communications technologies that can enable new research practices and methods" [2]. This work is part of the National e-Infrastructure for Social Simulation (NeISS) project, which is attempting to develop e-Infrastructure to support social simulation research [3].

Section 2 provides some background. Section 3 details the current e-Infrastructure and workflow for generating results. Section 4 presents some results with discussion and details the quantities of Grid resource used in terms of CPU hours, model memory requirement and the amount of storage used for results. Section 5 outlines next steps we are planning for this research collaboration. Section 6 briefly concludes.

2. Background

Contemporary social simulation models have been developed as part of the Generative e-Social Science for Socio-Spatial Simulation (GENESIS) project. GENESIS is a second phase research node of the UK National Centre for e-Social Science funded by the UK Economic and Social Research Council (ESRC). Funded for three years and started in October 2008, the GENESIS project is a collaboration between the University of Leeds and University College London [4]. One focus of GENESIS has been the development of individual based dynamic demographic social simulation models.

This paper, concerns the development of e-Infrastructure for supporting GENESIS models focusing on models of demographic population change driven by mortality, fertility and

miscarriage probabilities and initialised with age by gender population count statistics. These GENESIS models are basic in that they do not incorporate a migration component that details the movement of people from one location to another. Although basic, the model does explicitly handle multiple births and miscarriage, which is reasonably sophisticated for a demographic model. The GENESIS model we focus on in this paper is perhaps an original attempt to model a human population at an individual level, for a daily time step on the scale of a modern contemporary city (a population of hundreds of thousands).

NeISS is a project funded for three years by the UK Joint Information Systems Committee (JISC) under its Information Environment programme [3]. NeISS started in April 2009 and involves partners from 8 different institutions. In the last year, one part of NeISS has focused on developing e-Infrastructure for a GENESIS demographic daily time step social simulation model (henceforth referred to as the model).

3. e-Infrastructure and Workflow

A web portal based Graphical User Interface (GUI) has been developed as a set of standard JSR-168 portlets. One portlet is for specifying model parameters and setting a simulation running. Another portlet is for specifying simulation results to compare and setting a comparison job running. Other portlets are for monitoring submitted jobs and for interfacing with an archive of results. Currently there is only support for deleting results from the archive, but work is on-going to allow the full results to be transferred elsewhere out of the archive. The portlet and the underlying programs that have been developed for this work are made available under the GNU Lesser General Public License (LGPL) [5] via Tom Doherty's NeISS Source Code Web Page [6].
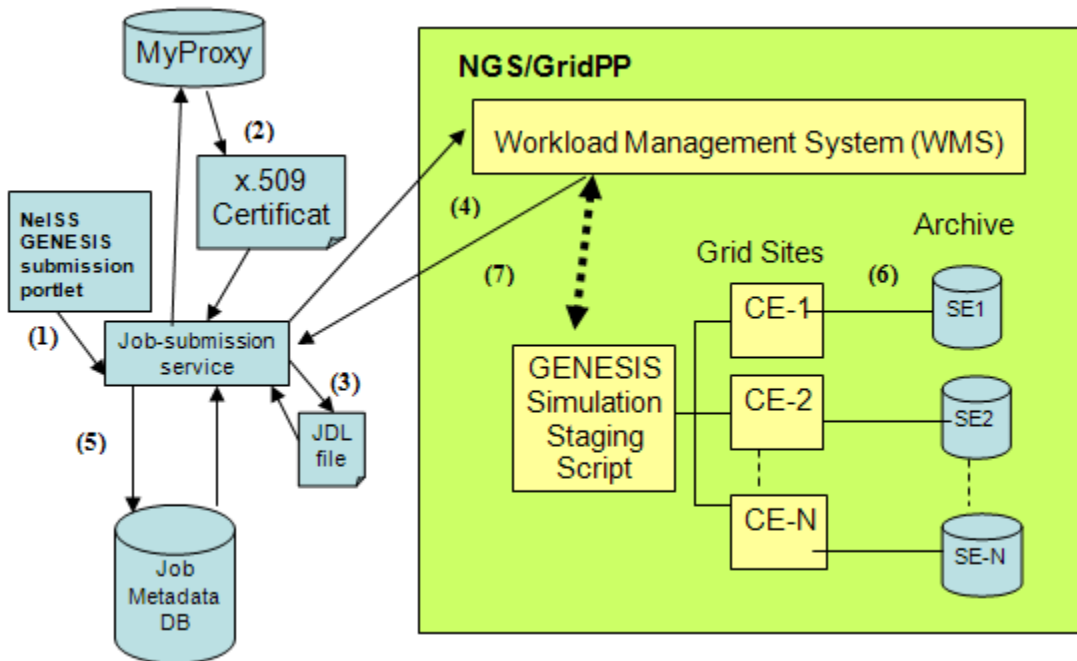
Programs enacted by the portlets stage data in and submit jobs to a Grid computer. Once a job is submitted some details are communicated back to the user via a job monitoring portlet. Once the job is completed, the full result is archived and smaller parts of the results are stored and made available for download. Collectively the system is called the Genesis Simulator.

The portlets have been developed and tested on the DAMES Application Portal hosted at the National e-Science Centre (NeSC) in Glasgow [7]. Being standard JSR-168 portlets means they can be readily migrated to work within other types of portal [8], however, the model will only run with the other required programs in place.

The model itself is implemented in Java and is made available as open source under the LGPL from Andy Turner's GENESIS Source Code Web Page [9]. The GENESIS source code forms a library of packages with dependencies on a number of third party open source libraries. It contains code for running other types of geographical models and some of the dependencies for the library are not required for reproducing the results presented in Section 4. For the Genesis Simulator two programs were compiled from the library. One is the 'simulation model program', the other is the 'simulation model results comparison program'. These compiled programs are passed with the data in workflow execution and require Version 1.6 or later of the Java Runtime Environment (JRE) to run.

The results in Section 4 are for 10 years (1991 to 2001) of simulation for Leeds Local Authority District in the UK. For each year, four simulations are run using different pseudo random number sequences that produce a range of results. These simulations are run in pairs. The Genesis Simulator simulation workflow is shown in Figure 1.

Figure 1. The Genesis Simulator Simulation Workflow

(1) After logging in and authenticating using shibboleth [10] the user initiates job submission via the portlet and the portlet then invokes the job-submission service
(2) Job submission service pulls user's proxy from MyProxy service [11] and creates VOMS proxy
(3) Job-submission service creates JDL file drawing in user input provided via the portlet interface
(4) jLite [12] API used for Java representation of glite-WMS commands – job submitted to WMS
(5) Job ID and associated Job metadata stored in Job metadata database using JDBC API [13]
(6) Model output saved in archive using WLCG tools and registered in LFC
(7) Model result metadata saved as output from worker node and passed back to portlet. Including GUID associated with population file (saved in archive) so that this file can be used as input to future jobs.

One of the results (from the two pairs of simulation results for each year), is selected for use as input for the next year to be simulated. The selection is based on a comparison program output which details and summarises differences between simulated and input mortality and fertility rates for each simulation. For the results described in Section 4, the simulation result with mortality and fertility rates closest to those input is the selected for continued simulation in the next year of simulation. The Genesis Simulator results comparison workflow is essentially very similar to the workflow shown in Figure 1.

The e-infrastructure uses the National Grid Service (NGS) [14] Workload Management System

(WMS) [15] to provide resource brokering-based job scheduling across all the Grid sites that support the NeISS VO [16]. The NGS user interface (UI) [17] provides a gateway for WMS job submission to NGS and GridPP [18] nodes via a command line interface. Virtual Organisation Membership Service (VOMS) [19,20] proxy certificates allow users to submit Grid jobs using a Job Description Language (JDL) [21] file that specifies the job parameters. These parameters detail the type of file to execute on the worker node, abstract file paths for uploading data via the input sand box, and abstract file paths for retrieving data from the output sand box.

Currently, the job staging script is executed on a single worker node and it handles the running of the model and pushing of results into a data archive organised as Logical File Catalog (LFC) [22]. On successful submission, a job id is returned which is used to monitor the job status from the portal interface which calls the glite-wms-job-status command in the background. Once the job has run, result files are pushed automatically via the output sandbox to be stored in the archive. Summary results are also passed to a portlet for display and download.

Without the job staging script and portal based GUI, a user would have to issue the appropriate glite-wms-job-submit command to submit the job and associated JDL file to the WMS and associate a delegated VOMS proxy with it. To fetch the job output once the job completed, they would also have to issue an appropriate glite-wms-job-output command. All this command line work is now hidden from the user behind the simple and easy to use portal based GUI.

The complexity of Grid middleware coupled with grid certificates has proven to be a barrier to entry for researchers in some disciplines [23]. As portals were already being used within the NeISS project, it was fitting to develop and use portlets to facilitate job-submission, status-monitoring, output retrieval and job output comparison. For the SARoNGS[27] implementation the OMII SPAM-GP Shibboleth module [24] is used to login to the portal framework, and an iFrame [25] within one of the portal pages provides a link to the NGS Credential Translation Service [26] GUI from which users are to generate and download their SARoNGS certificate. This is still to be integrated as we await a certificate fix to be rolled out to GridPP sites. A Registration Portlet allows the information provided in the portal account and the generated SARoNGSs proxy to be used to contact the VO administrator to request membership. Job metadata and monitoring is recorded in a database which makes it possible to track and manage each job submitted. A Management Portlet allows for the deletion of results data stored in the database and the associated archived data. The main Job Submission Portlet is designed with a 'wizard' type flow where the user enters the necessary information for the simulation in a step by step fashion. The certificate and JDL configuration is transparent to the user.

Archive data management leverages the existing WLCG [28] gLite [29] infrastructure, in order to reduce the amount of additional work needed to support it across potential sites . Files generated by a job are stored at the local Storage Element (for the UK, most often a DPM [30] in front of a disk pool), and registered in the UK LFC at RAL [31]. Later jobs can be directed to sites holding local copies of required data, and copies of data can be replicated at other sites, as the files are managed entirely in terms of their Globally Unique Identifier (GUID) [32] assigned by the LFC.

A video has been produced and made available on-line to show the use of the Genesis Simulator as described above [33]. The video is being shown as a basis for the demonstration of the Genesis Simulator at the [UK e-Science All Hands Meeting 2011](). The video speeds viewers through the workflow execution when in reality considerable time can be spent in queues and waiting for the result to process. In the presentation slides for the talk that accompanies this paper there are some snapshots of the GUI at different stages of workflow execution for the results presented in Section 4. All this dissemination is linked together and referred to from Andy Turner's NeISS Web Page [34].

4. Results

This section describes the results of a 10 year (1991 to 2001) simulation for the Leeds Local Authority District in the UK using the workflow detailed in Section 3. The size and format of the results are described and some pictures of them are presented as examples of the types of demographic output that can be automatically produced. Estimates are provided for the amount of resources used in the development of the Genesis Simulator to date and the amount of resources required to reproduce the best fit result.

The simulation outputs are the individual level population detail, a set of aggregated statistics, and various other metadata. We call these Type 1, Type 2 and Type 3 data outputs. The Type 1 data are the metadata details for recreating a simulation result. Type 2 data include the Type 1 data, some text output written as the model executes, and some generalised statistical outputs. The statistical outputs include age by gender population summary data and images that depict the population alive at the end of a yearly time step and the population that died during it. Examples of these images are shown in Figures 2 and 3 respectively. Type 3 data include the Type 2 data and the individual level population data. Currently this cannot be downloaded from the Genesis Simulator, however work is being done to allow for this. The result can be readily recreated at another site supporting the workflow execution using as input all the Type 1 data for each year of simulation from an initial starting population summary.

Figure 2. A Demographic Depicting the Living Population at a Particular Time Step

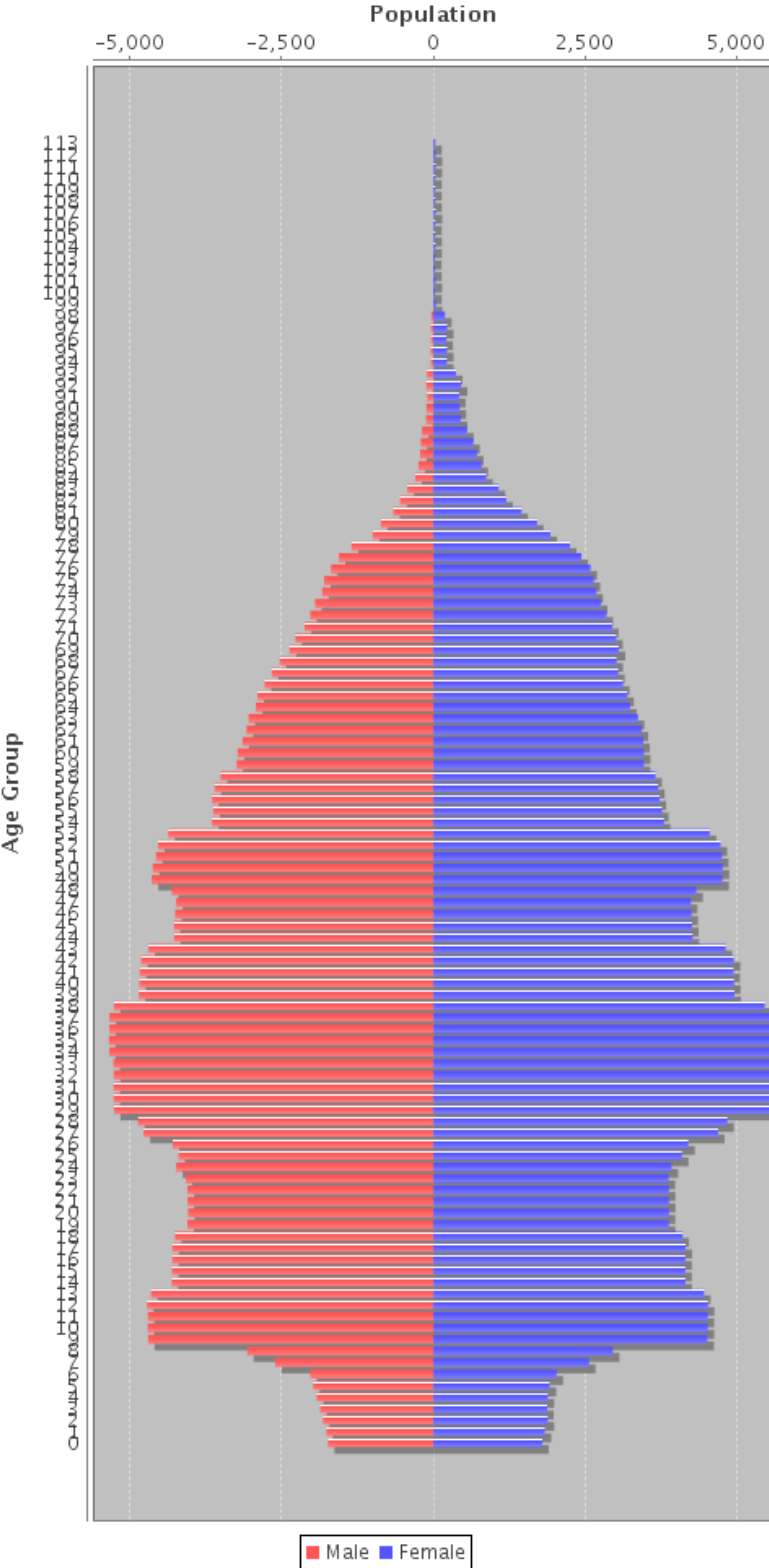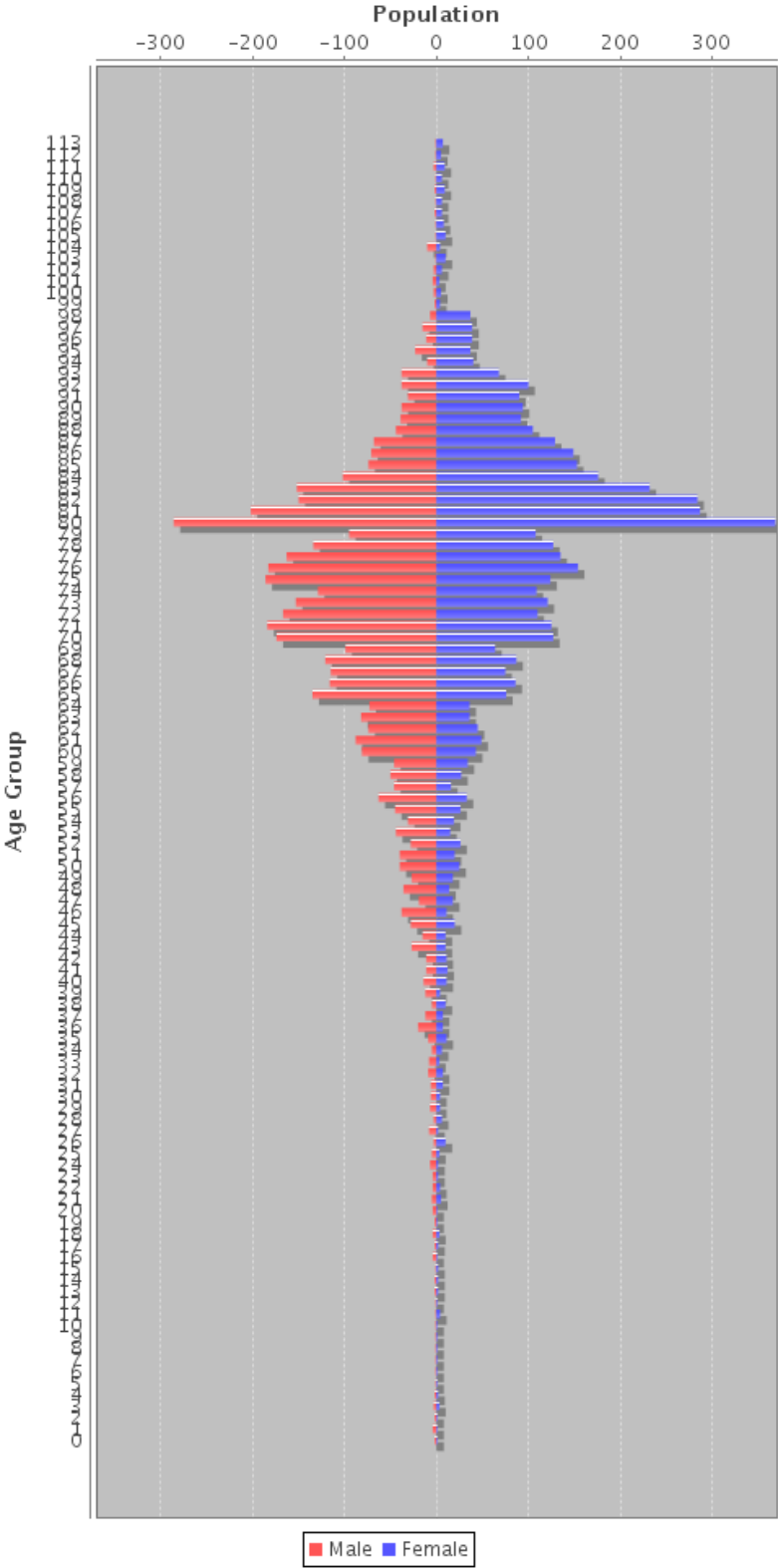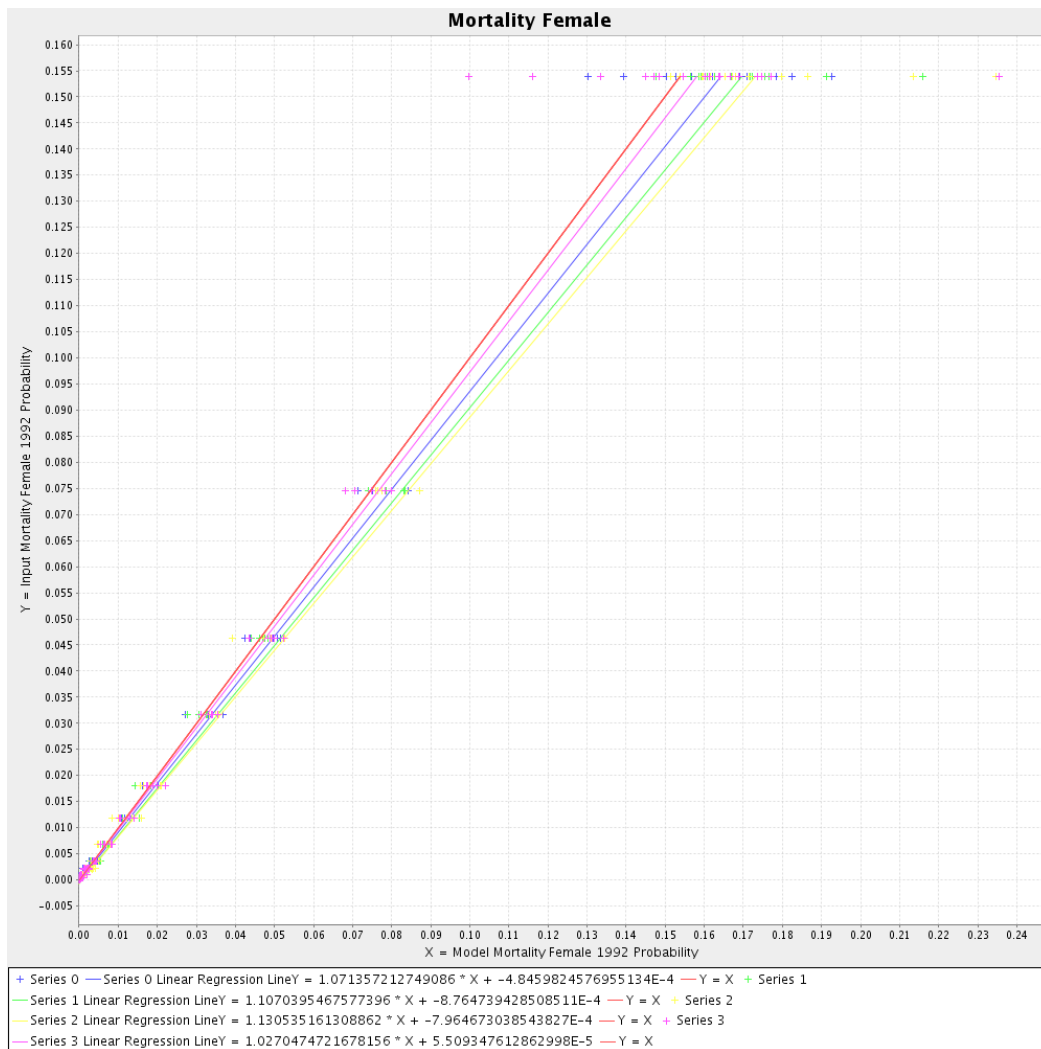Age Gender Plot Living Population Year 2000

Figure 3. A Demographic Depicting the Population that Died in a Particular Period

Age Gender Plot Dead Population Year 2000

Results comparison outputs include a text file containing comparison details written out as the program executes. Crucially this identifies the best fit result according to some aggregate statistics which compare the input or imposed mortality and fertility probabilities with those that are output or result in simulation. Anything from the best fit to the worst fit overall result can be selected for simulation continuation. What is appropriate to select depends on whether trend or more extreme results are wanted. Fertility and mortality graphs are plotted for results run for the same period revealing the difference between observed rates and those that were input. An example graph for Female Mortality is given in Figure 4.

Figure 4. A Plot for Comparing Input and Resulting Female Mortality Probabilities for 1991

The leftmost (red) line on the graph in Figure 4 is the $Y = X$ line. The other four lines show the regression line of best fit for four simulations. They are all to the right of the $Y = X$ line which suggests that there may be a data issue and/or a logic error in the model such that observed mortality rates are higher than they perhaps should be. If the model was run many times and there were no bias, then it is expected that the average distribution of regression lines would be

around the Y = X line.

The results we have generated to date are only for testing the models. The models are not yet ready to be used in support of applications. Already, some issues have been identified and are awaiting further investigation. Part of the investigation process is likely to involve the production of more results and in doing this greater automation will be sought to reduce the manual overhead in executing workflows.

In time, resulting population estimates from the simulation in 2001 will be compared with other available population data from the 2001 UK Population Census. Significant differences will be expected when doing this comparison as the model does not deal with migration which is known to have a large effect on the population of Leeds [36]. The effect of net migration (the difference between the residential population moving out of and into a region) on other regions of the size of Leeds will vary, but in general, for larger regions with more population, the effect of net migration reduces as net migration becomes smaller relative to the size of the entire population.

The minimum memory requirement for the model has not been calculated, but 1.2GB is set for the JVM in model execution in producing these results. As we scale to larger populations there is likely to be a larger memory footprint, but as most of the data is stored on slower access memory, it is effectively only the indexes for the data that grow in memory and these can also be organised for alternative storage on disk if memory issues are encountered. The largest result for any yearly simulation step is 2.2GB. All results (4 for each time step and 10 time steps) for Leeds requires 53GB.

In terms of resource usage to date. Just under 2160 hours (90 days) of CPU time has been used by the NeISS VO on Grid Computer resources. About 12% of this (just under 280 hours or 11 days) was for the generation of the results described above. This is the amount of resource in terms of CPU that would be needed to recreate all the result. However, less than 25% of that is required to create the results that are regarded as the best.

The comparison jobs run for about a minute or so, whereas the simulation runs for hours.

5. Next steps

The next steps to improve on this work involve demographic model development and e-Infrastructure improvements.

The demographic model for which the results are presented is very basic. It does not have a migration component which represents the movement of people between different regions. It is to be expected that net migration will significant effect result and so in attempting to model what actually happens, incorporating a migration component is important. Indeed it is regarded as a key step if the model and results are to gain credibility in the demographic modelling community.

The very basic model does not have a concept of father nor does it try to group males and females into couples. Migration and parenting are related and these are also related to fertility. It is also known that the relationships between people or lack of them effect life expectancy and this might also be considered in model enhancement. Furthermore, people of the same

age exhibit a range of fertilities, some being very fertile and others perhaps infertile. Varying fertility rates for individuals is something that can be readily implemented. The way probability precision is handled in the model supports this such that each individual can be assigned their own fertility in a way so that the fertility of an age group as a whole stays average in a stochastic likelihood sense. It is the arbitrarily specifiable fixed number of decimal places precision numbers that are used for probabilities in the model that support the provision of such a feature. Similarly, for each individual, their fertility can be varied from day to day in such a way that the average over a period of days reflects the likelihood in general. There is a lot of work to be done demonstrating that these probabilities work.

The amount of detail that can be added to the model is endless. Health is a key thing to model. It effects mortality and fertility yet like migration, it is itself a very complex thing to model. The data on miscarriage that is currently used is quite general. Although it based on the age of females, it does not really take into account other factors such as the history of previous pregnancies which could be significant. Similarly it is common knowledge that multiple births run in familes, also pregnancy with twins is more common in females towards the end of their fertile years, and yet currently the likelihood of a multiple birth is the same for all females of the same age in years in the model.

A preliminary analysis of the results produced for this paper has raised further issues with the integrity of the model. Mortality rates seem bias and resulting fertility is lower than expected. This may be a consequence of the values used in the denominator of the rates. It may be a consequence of migration. It may also be a consequence of a logic error in the probability calculations. Investigation is needed and help should be sought from the demographic modelling community.

Iterative improvement of results should go hand in hand with model development and improved automation and flexibility of the Genesis Simulator.

Another key things to consider are how the results and the model can be validated.

We are in the process of scaling up to produce results for England, enhancing the Genesis Simulator as we go to readily make use of more computational resources across multiple sites.

Computational demands are currently being met by resources at the NGS/GridPP ScotGrid site [35]. In the next phase of work, we are hoping to use resource at other GridPP sites. GridPP have been key collaborators approving the NeISS VO and providing a scalable storage solution with 2TB of data for the Genesis Simulator to use. Hopefully other GridPP sites will collaborate with us with similar data storage provision. Hopefully in return we may be able to offer not only user testing, credit and thanks, but also some of our workflows may be generally useful for others.

The e-infrastructure uses the Virtual Organisation Membership Service (VOMS) solution [13] to handle access to NGS and GridPP resources. The aim is to use the SARoNGS approach [14], but while this is being organised, effectively a single certificate is being used with an appropriate accounting system in place. Hopefully the SARoNGS solution will be in place before we try to enact workflows at other sites.

6. Conclusions

The Genesis Simulator is an attempt to Grid enable some geographical models. All the source code of the model and e-Infrastructure is open source. The key to e-Infrastructure development is collaboration between the researchers and this has been working well for the NeISS work outlined herein. We hope that by the end of the NeISS project, the Genesis Simulator will have been used to produce some demographic model simulation results for England from 1981 until 2012 and that the results for 2012 will be in line for submission to the UK Data Archive Economic and Social Data Service [36]. We further hope that awareness of these results and the Genesis Model will be raised in the demographic modelling community and that we can begin to work more collaboratively with other demographic model developers to improve the model.

The paper presents an original attempt to support the development and use of demographic models that operate at the individual level at a daily resolution in a scalable way.

It is nearly always the case that in developing software for analysis or modelling that a large amount of resource is used in development. The estimated resource use given in Section 4 does not account for all the effort and computation that has gone into developing the Genesis model. The estimates given are for testing and configuring the models to run on specific Grid resources which although a considerable effort, is only a fraction of the overall effort that has gone into model development in GENESIS.

The results planned to be generated for all of England will further test the abilities of the e-Infrastructure. The demands on computational resources are expected to be around 100 times greater when simulating all of England compared to Leeds .

As the size of the population to be simulated increases and the number of steps in the simulation increases, the model demands more and more computational resource. A UK national simulation has requirements of input and output data in the size region of terabytes, and compute times for atomic model components that run for several days. We believe that the e-Infrastructure is capable of this, but the availability of large grid and cloud computing resources is not a certainty.

Producing larger (hopefully more impressive) simulation results should improve understanding of how well the model scales and produce results of greater interest to the demographic modelling community. We expect issues as we approach and push the boundaries of what is computationally feasible with available resources and what is implementable in terms of demographic modelling.

7. Acknowledgements

8. References and further information

1. Gilbert N., Troitzsch K.G. (2005) Simulation for the social scientist. Second edition. Milton Keynes: Open University Press. ISBN-13 978 0335 21600 0.
2. Voss A., Vander Meer E., Fergusson D. (2009) Research in a Connected World. http://www.researchconnect.org/book
3. NeISS http://www.neiss.org.uk
4. GENESIS http://www.genesis.ucl.ac.uk/
5. GNU Lesser General Public License http://www.gnu.org/licenses/lgpl.html
6. Tom Doherty's NeISS Source Code Web Page http://ppewww.ph.gla.ac.uk/~tdoherty/NeiSSCode/NeiSS_Source_Code.htm
7. DAMES Applications Portal https://dames.nesc.gla.ac.uk/
8. Java Specification Request 168: Portlet Specification http://www.jcp.org/ja/jsr/detail?id=168
9. Turner A.G.D. (2008) GENESIS Source Code Web Page http://www.geog.leeds.ac.uk/people/a.turner/src/andyt/java/projects/GENESIS/
10. The Internet2 Shibboleth framework http://shibboleth.internet2.edu
11. MyProxy http://grid.ncsa.illinois.edu/myproxy/
12. jLite Java gLite API http://code.google.com/p/jlite/
13. Java Database Connectivity (JDBC) API http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136101.html
14. NGS http://www.ngs.ac.uk
15. Workload Management System http://glite.web.cern.ch/glite/wms/
16. VOMS admin for VO: neiss.ac.uk https://voms.ngs.ac.uk/voms/neiss.org.uk
17. NGS WMS UI http://www.ngs.ac.uk/ui-wms
18. GridPP http://www.gridpp.ac.uk/
19. Alfieri R., Cecchini R., Ciaschini V., dell'Agnello L., Frohner A., Gianoli A., Lorentey K., and Spataro F. (2003) VOMS, an Authorization System for Virtual Organizations. https://twiki.cnaf.infn.it/twiki/bin/viewfile/VOMS/WebDocumentation?rev=1;filename=VOMS-Santiago.pdf
20. VOMS http://www.ngs.ac.uk/site-level-services/voms
21. JDL Specification https://edms.cern.ch/document/590869/1
22. Logical File Catalog http://www.gridpp.ac.uk/wiki/LCG_File_Catalog
23. Jensen J., Spence D., Viljoen M. (2007) A Scalable PKI for a National Grid Service. http://middleware.internet2.edu/pki07/proceedings/11-jensen-pki-national-grid.pdf
24. SPAM-GP http://www.nesc.gla.ac.uk/projects/omii-sp/index.html

25. iFrame http://en.wikipedia.org/wiki/HTML_element#Frames
26. NGS CA Hierarchy http://wiki.ngs.ac.uk/index.php?title=NGS_CA_Hierarchy
27. SARoNGS http://www.jisc.ac.uk/whatwedo/programmes/einfrastructure/sarongs.aspx
28. Wordlwide LHC Computing Grid (WLCG) http://lcg.web.cern.ch/LCG/
29. gLite Lightweight Middleware for Grid Computing http://glite.cern.ch/
30. Disk Pool Manager (DPM) http://www.gridpp.ac.uk/wiki/Disk_Pool_Manager
31. The Rutherford Appleton RAL http://en.wikipedia.org/wiki/Rutherford_Appleton_Laboratory
32. Globally Unique Identifier (GUID) http://en.wikipedia.org/wiki/Globally_unique_identifier
33. AHM 2011 Genesis Simulator Video http://ppewww.ph.gla.ac.uk/~tdoherty/AHMDemo/video-AHMDemo.html
34. Andy Turner's NeISS Project Web Page http://www.geog.leeds.ac.uk/people/a.turner/projects/e-ISS/
35. ScotGrid http://www.scotgrid.ac.uk/
36. Wu B.M., Birkin M.H., Rees P.H. (2008) A spatial microsimulation model with student agents. Journal of Computers, Environment and Urban Systems, volume 32, pages 440-453. DOI: 10.1016/j.compenvurbsys.
37. ESDS http://www.esds.ac.uk/
38. What happens when international migrants settle? Ethnic group population trends and projections for UK local areas under alternative scenarios http://www.esrc.ac.uk/my-esrc/grants/RES-163-25-0032/read
39. Ethnic group population trends and projections for UK local areas: dissemination of innovative data inputs, model outputs, documentation and skills http://www.esrc.ac.uk/my-esrc/grants/RES-189-25-0162/read