Earth System Modelling with Windows Workflow Foundation

Matthew J. Fairman [a] Andrew R. Price [a,*] Gang Xue [a] Marc Molinari [a] Denis A. Nicole [b]
Timothy M. Lenton [c] Robert Marsh [d] Kenji Takeda [a] Simon J. Cox [a]

[a] *School of Engineering Sciences, University of Southampton, Southampton SO17 1BJ, UK*
[b] *School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK*
[c] *School of Environmental Sciences, University of East Anglia, Norwich NR4 7TJ, UK*
[d] *National Oceanography Centre, University of Southampton, Southampton SO14 3ZH, UK*

Abstract

The GENIE project has built a Grid-enabled Earth system modelling framework that facilitates the integration, execution and management of component models for the study of the Earth system over millennial timescales. The existing framework supports collaborative study of GENIE models across heterogeneous compute grids through scripted workflows in the Matlab environment. While the scripting approach achieves simplicity and flexibility, it suffers from an essentially passive approach to work unit management and from a heavy reliance on a central database to provide fault tolerance. The Windows Workflow Foundation (WF) technology provides a rich set of features to support the authoring and execution of workflows, tracking services that enable the monitoring of a running workflow, and state persistence services that allow workflows to be recovered and resumed upon failure. We demonstrate how the Windows Workflow Foundation has been applied to build a complementary simulation management system which provides rapid composition, event driven logic and reliable hosting of the scientific workflows while interfacing to existing infrastructure. We also describe how the adoption of WF enables the application of a number of associated technologies to provide better interoperability and accessibility for the simulation system. These improvements are demonstrated through a parametric study of the bi-stability of the oceanic thermohaline circulation in a GENIE model where the effects of a new carbon cycle are studied.

Key words: Workflow management, Earth and atmospheric sciences, Windows Workflow Foundation
PACS: 92.70.Pq, 92.10.af, 92.20.Sg

1. Introduction

Advances in science and engineering are increasingly reliant upon building collaborations between people and digitally connected entities such as computers, data, applications, sensors and networks. Grid computing technology [8] is enabling these interactions across distributed, heterogeneous domains and is driving the field of e-Science. However, harnessing Grid technology in a particular problem domain requires scientists to become familiar to some degree with the software supporting the Grid. Workflow tools are therefore important in e-Science because they help abstract away the complications of Grid middleware and leave the user with the task of composing the logic and data flows to achieve their scientific aims. Workflow technologies have

* Corresponding author.
Email addresses: `mjf@soton.ac.uk` (Matthew J. Fairman), `andrew@soton.ac.uk` (Andrew R. Price), `gx@soton.ac.uk` (Gang Xue), `mm2@soton.ac.uk` (Marc Molinari), `dan@ecs.soton.ac.uk` (Denis A. Nicole), `t.lenton@uea.ac.uk` (Timothy M. Lenton), `rma@noc.soton.ac.uk` (Robert Marsh), `ktakeda@soton.ac.uk` (Kenji Takeda), `sjc@soton.ac.uk` (Simon J. Cox).

become a key element to e-Science systems, helping to orchestrate service interactions so that they can be seamlessly knitted together to implement the desired behaviour of the system.

An example of an e-Science system is the Grid Enabled Integrated Earth system modelling (GENIE) project [12]. The GENIE project has developed a framework for the composition, execution and management of integrated Earth system models. Component codes (e.g. ocean, atmosphere, land surface, sea-ice, ice-sheets, biogeochemistry, etc.) of varying resolution and complexity can be flexibly coupled together to form a suite of efficient climate models capable of simulation over millennial timescales. The project brings together a distributed group of environmental scientists with a common interest in developing and using GENIE models to understand the Earth system. Grid computing technology supports the virtual organization to collaboratively study the models from the framework [28].

Climate models are both computationally and data intensive processes. Typical studies involve evaluation of large multi-member ensembles requiring weeks of continuous CPU time. The client software deployed to enable the GENIE science team to manage Earth system simulations on the Grid is built from the Geodise Toolboxes for Matlab, a product of the UK e-Science pilot project GEODISE [10]. These toolboxes provide Grid functionality to extend the Matlab and Jython (Java Python) problem solving environments that are familiar to scientists and engineers. Simple linear and cyclic scripted workflows have been developed to enable collaborative ensemble studies of GENIE models [12,26]. The scripting approach has proven an accessible means of providing scientists with powerful core functionality for model management on the Grid and the means to coordinate collaborative studies mediated by data services. However, there are significant limitations to the nature of the workflows that can be expressed; their re-usability; and the robustness with which they can be executed in the Matlab environment.

The Microsoft Windows Workflow Foundation (WF) [31] is a general, extensible framework for developing workflow solutions. As an integral part of the .NET Framework 3.0 [19], WF provides a common development model for creating workflows, a set of tools that facilitate workflow design, and a robust workflow hosting environment. It also allows seamless integration with other .NET technologies to provide support for distributed communication and rich user experiences.

We have applied WF to the GENIE framework to provide a robust long-running workflow hosted on a Windows Server to manage GENIE Earth system models. The application is complementary to the existing Matlab client and interfaces to existing infrastructure. The WF solution provides a robust and reliable environment for executing GENIE workflows for the management of Earth System simulations. By exploiting event driven logic available in WF the efficiency with which the workflows manage simulations is greatly improved. A rich user interface, delivered through an internet browser, enables the scientist to monitor and manage experiments with greater ease and convenience than is possible in the Matlab problem solving environment.

In this paper we discuss and demonstrate how Microsoft Windows Workflow technologies are applied to build an efficient and flexible workflow with the GENIE framework. In section 2 we review existing workflow technologies for e-Science and Grid computing. In section 3 we describe the functions of the GENIE workflows and explain why WF technologies have been adopted. Section 4 gives a detailed summary of the features of WF. Section 5 describes the design and running of our WF based GENIE workflow. Results from a study of bi-stability in the thermohaline circulation of a GENIE model are presented in section 6. Conclusions are drawn in section 7.

2. Related Work

Workflows are the expression and enactment of connectivity, in both execution control and data flow, between functional entities (variously referred to as actors, processors, agents, services) to solve a problem. Fox and Gannon [9] identify a hierarchy of workflow technologies based upon a simple complexity scale. This is an instructive methodology for characterising workflows in the context of the GENIE Earth system modelling project.

At its simplest a workflow is just a linear sequence of tasks where the output from one activity is provided as input to a following activity. A series of activities are invoked one after the other to achieve the job at hand. The expression of linear workflows is perhaps most commonly programmed in scripting languages like bash/tcsh, Perl and Python. Indeed, the GENIE project has adopted the Geodise Toolboxes [4] which augment the Matlab problem solving

environment with scriptable Grid functionality. Activities for the management of Earth system models on the Grid have been scripted by chaining simple tasks in Matlab function scripts. For example, the submission of a single simulation to a remote compute resource involves a linear sequence of simple tasks that coordinate the creation of a unique directory, the transfer of model binary, input files and configuration files and the submission of the compute job to the job manager. A large number of similar sequential workflows have been developed for the project to enable a range of studies including task farming for parametric study [14], systematic tuning of model parameters to improve fit to observational data [27,17] and collaborative ensemble studies [12,26]. While linear workflows enable significant work to be achieved, and within PSEs provide an intuitive extension to the functionality of an existing familiar environment, there are limitations in their scalability and expressiveness. To extend the workflows requires an additional framework to support them. For example, collaborative studies with the GENIE workflows in Matlab rely extensively on database services to provide fault tolerance and job brokering [26]. We discuss the short-comings of the Matlab workflows for GENIE in the next section but first discuss workflow technologies that operate higher up the complexity scale.

The next level of complexity involves acyclic workflows where task graphs are defined in terms of explicit dependencies between activities. Workflows are built from activities that depend upon the prior execution of other activities. The Java CoG kit provides GridAnt to express workflow and Karajan to enact [33]. Grid computing is naturally supported through the Java Globus API. Inspired by the Java build tool Ant, GridAnt exploits the dependency graphs that Ant naturally expresses to implement acyclic workflows. The Java CoG kit extends the workflow language with conditions, loops and exceptions to support more complex constructs. Similarly, Condor DAGMan [32] expresses execution plans for job scheduling in the Condor high throughput computing system. UNICORE [29] defines Abstract Job Objects (AJO) with the necessary synchronization between the elements reflected by forming a directed acyclic dependency graph. A drawback of these systems is that they can require the user to work with unfamiliar languages such as XML. Although elementary GUI functionality exists the scientist would have to spend significant time learning how to construct the workflows.

Many workflow tools for e-Science avoid burdening the user with expressing their desired system behaviour in a programming language by providing a graphical composition environment. Almost all such systems provide a set of 'components', typically representing computational tasks or services, that are added to a workspace or assembly where connections are made between their typed inputs and outputs. The resulting graph is then executed using a workflow enactment engine. Such systems enable cyclic workflow expression and introduce support for asynchronous activities.

The Taverna Workbench [23] has been developed by the ^{my}Grid project for the composition and enactment of bioinformatic workflows. In Taverna, workflow is considered to be a graph of *processors* each of which transform data inputs to a set of data outputs. The simple conceptual unified flow language (scufl), an XML based language, is used to represent the workflow and is enacted by the Freefluo engine. Triana [3] provides a graphical composition tool to link functional *components* via synchronous or asynchronous *communication pipes* resulting in a directed cyclic graph. Similarly, in Kepler [13] workflow is considered in terms of connections between independent components called *actors* with communications channelled through *ports*. Kepler introduces the concept of a model of computation (MoC) which is determined by *providers* that are associated with each actor using the underlying Ptolemy II system. The execution and communication semantics of the workflow are determined by the choice of directors and this enables *behavioural polymorphism* whereby the same actors can be used in both control driven and data driven workflows. Workflow projects such as these are typically focussed on the problem domain for which they were developed and integrating new services from a novel domain can be time consuming. In the absence of established standards for the expression of Grid workflows the investment in migrating to these systems needs to considered carefully.

Alternatively, workflow environments in the business community are well established. Eswaran [5] discusses the application of commercial workflow products in e-Science and compares Microsoft BizTalk (XLang), OracleBPEM and Microsoft Windows Workflow Foundation. The key challenges that business workflow tools must address for e-Science are identified as: providing support for large data flows, enabling execution of large numbers of parameterised tasks and supporting dy-

namic environments in which resources come and go. Paventhan *et al.* [24] discuss the application of Windows Workflow Foundation to the management and processing of large volumes of data from wind tunnel aeroacoustic measurements on a Grid system. The graphical composition environment of Visual Studio with code beside features enable the workflow designer to exploit the full power of object oriented languages. Visual Studio provides comprehensive support for consuming web services, databases (ODBC, etc.) and third parties have developed Grid client services (myCoG.NET) [25] and native Globus clients [7].

Microsoft Windows Workflow Foundation is not yet a common tool in the scientific community but is well established in the document processing community in products such as Microsoft SharePoint. WF provides a mature framework for the reliable enactment of long running workflows and is an integral part of release 3.0 of the Windows .NET framework. As such, WF applications can be deployed and hosted on any recent Windows platform. WF is well established in a number of important Microsoft product lines and will therefore be well maintained and supported into the future. WF supports synchronous and asynchronous communication, event driven logic and provides an extensible model for workflow development. Microsoft Visual Studio provides a comprehensive environment for the design and construction of workflows. These provide compelling reasons to consider WF as a competitive alternative to dedicated systems such as Taverna, Kepler and Triana. Workflows fundamentally coordinate the work performed by people and software and it is important that users can interact with long-running processes. WF solutions can leverage other .NET technologies to build user interfaces to the workflow applications.

Members of the GENIE project have a need to execute large ensembles of long-running Earth system models. To support this need, a number of linear scripted workflows have been manually constructed in Matlab to manage simulations through numerous checkpoint and restart cycles to progress ensembles to completion on Grid computing resource. These workflows are critically reliant on a central database to maintain the state of the study. The workflows on the client side are operated in a speculative manner and there is no coordinating process to guarantee completion. Support for the reliable and robust enactment of a long-running workflow would therefore benefit the GENIE community greatly. In practice however, Matlab provides no "out-of-the-box" support to sustain long running processes and does not provide the means to coordinate the actions of a distributed user base. We have therefore applied to WF to the GENIE framework to provide a complementary simulation management system capable of reliably managing long running ensemble simulations.

3. Workflow in the GENIE Framework

The GENIE framework has been designed to facilitate cooperation among project team members to perform large ensemble studies of Earth system models that would be impossible for a single user. In contrast to system likes GridSolve [34] and Nimrod/G [1], in which a central manager controls all available resources and makes decisions about where to assign work units, the GENIE framework devolves responsibility for the management of work units to the client side, and pools compute resources contributed by members of the project to carry out large parametric studies. A central shared database accessible from all clients has been set up to provide the means to upload model configurations and experiment definitions, and be queried against to find available work units.

A study is initiated by a scientist uploading an experiment definition to the shared central database. The system provides a unique identifier for that experiment which is then distributed amongst the scientists participating in the study. The experiment definition comprises descriptive metadata and contains references to a collection of configuration files for the GENIE simulations that comprise the ensemble to be studied. Typically, each simulation requires many days of CPU time to complete and will typically need to be checkpointed and restarted on multiple occasions. The database system provides the means to manage the execution of ensemble simulations by automating the handling of the checkpoint files.

A GENIE workflow running at the client side is therefore responsible for the following tasks:
– Accepting user input with details of new resource availability
– Querying the central database for available work
– Submitting the work to the compute resources specified by the user
– Monitoring the status of the submitted jobs
– Retrieving and validating the results once jobs are finished

4

– Post-processing the results and archiving them in the database

A scripted workflow in the Matlab problem solving environment is invoked by the participating scientists to progress the ensemble of simulations. The Matlab workflow is shown in figure 1. The script accepts as input from the user the unique experiment identifier (*exptID*), a data structure describing a computer resource (*resource*), the number of compute jobs to submit to that resource (*njobs*) and the amount of work (*timesteps*) that each job should perform. The script invokes three activities in sequence that attempt to progress the simulations of the experiment. A post-processing script is invoked first in order to poll the status of all active job handles and process any work units that have completed since the last invocation of the workflow. If compute jobs have completed then their output is validated and archived to long-term storage. This ensures that all simulations are up to date before querying the database for details of new work to perform. New work is found by performing a number of queries on the database and returning a list of the incomplete simulations in the experiment that do not have active compute jobs associated with them. The third phase of the workflow submits a new compute job for each simulation to the resource specified by the user. This workflow is invoked at regular intervals until all simulations are complete. Collaborative study is achieved when the distributed user community invoke this workflow on multiple client systems and submit jobs to heterogeneous compute resource.

Standard Grid-enabled GENIE activities are implemented using the Matlab scripting language and hosted in the Matlab environment. While simplicity and flexibility are achieved through scripting, a few problems remain. The scripting approach for workflow tends to be less efficient when dealing with external events. When running a GENIE simulation, the user needs to keep running the script all the time and query the central database periodically to find work items, submit and monitor jobs, and post-process data from completed work items. Time intervals between two scheduled tasks can only be set in a speculative mode. Events such as new work units being added and compute tasks completing are likely to remain unhandled for a lengthy period of time. Callback mechanisms or publish/subscribe mechanisms would be an optimal choice for improvement. For example, the database could publish new work as it becomes available and the clients would subscribe for these events. However, it is difficult to make such modifications in the Matlab scripting environment.

A further problem with hosting workflows in the scripting environment is the lack of support for persisting the state of the workflow. While it is possible for the users to manually save some of the state information, restoring the running of a workflow in the cases of a system or network failure is always difficult. The GENIE workflow relies entirely on the database for fault tolerance. The database maintains the state of the study and the workflow is invoked at regular intervals to make any progress that it can. In the event of failure the workflow exits - the transactions on the database recover to a consistent state and the processing will be attempted again on the next invocation. This relies upon a user to schedule a regular invocation of the scripted workflow. There is no persistence in the operation of the workflow.

Finally, implementing the GENIE workflow in the Matlab script effectively ties the system to a single user environment. Workflows in scripting languages are also less well-structured and might require individual human knowledge to interpret or modify. This makes it hard to maintain or to try to reuse the workflows.

To address these problems, we have exploited the WF technologies to develop a dynamic and robust workflow implementation which actively manages the operations of the GENIE simulations with improved efficiency and stability.

4. Windows Workflow Foundation

Windows Workflow Foundation (WF) [31] is a Microsoft technology for quickly building workflow enabled applications. It consists of a programming model, a host engine and tools for defining, executing, and managing workflows. WF significantly enhances a developer's ability to model and support sophisticated science and engineering processes. Being part of the recently released .NET Framework 3.0, WF can be seamlessly integrated with other .NET technologies to provide a solid base for the design and running of Earth system simulations.

4.1. *Workflow Design with WF*

WF supports two built-in workflow styles: sequential workflows, which carry out a series of operations in a pre-defined pattern, and state machine workflows, which are driven by external events. This en-
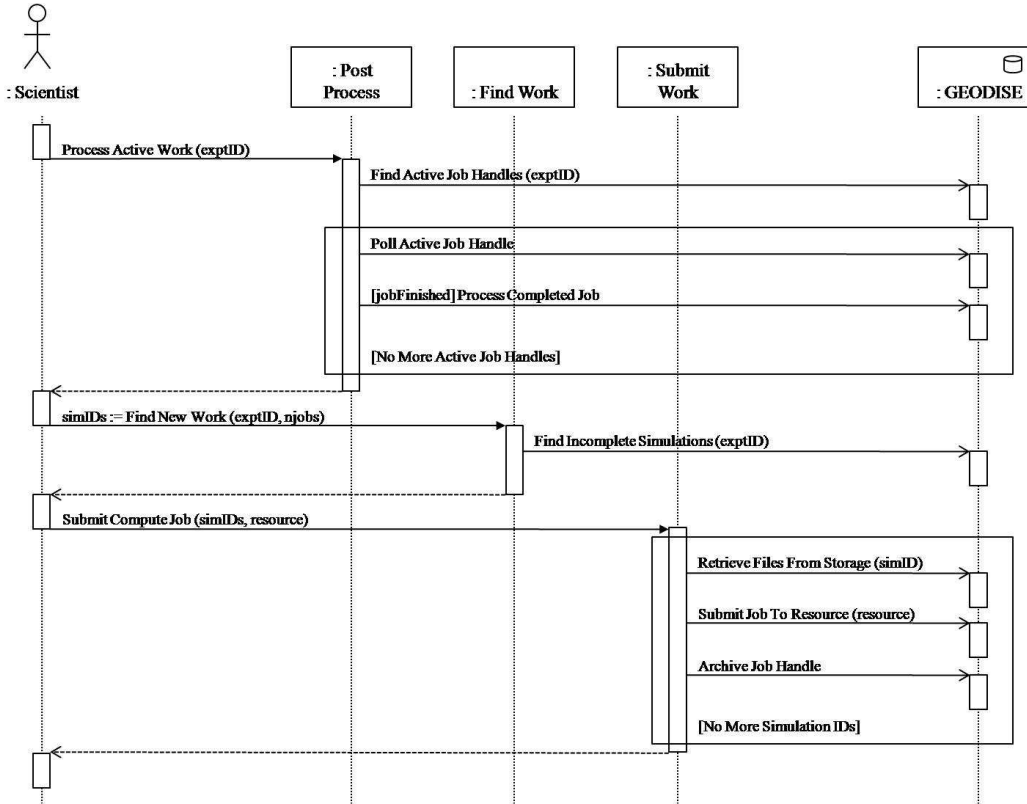
Fig. 1. UML sequence diagram of the workflow enacted in the Matlab problem solving environment.

ables WF to accommodate both the well structured, automated system workflows and the loosely defined, flexible human workflows. WF adopts a component like approach, where each step of a workflow can be implemented by a specific piece of software called 'activity'. A base activity library is supplied which contains a group of pre-defined activities most commonly used in both sequential and state machine workflows. Users are also allowed to build their own customised activities, build new activities by aggregating existing ones, and reuse activities in different workflows.

WF is a general framework not limited to a single language or a single tool. Workflows can be created using any programming language that is compliant with the .NET Frameworks Common Language Specification, such as C#, Visual Basic and Visual C++. In addition, WF supports defining workflows in XAML (eXtensible Application Markup Language) [22], an XML based object initialization language, so that different workflow designers can be used to generate or parse WF workflows. WF also provides its own graphical designer, usually hosted in Visual Studio 2005, to support visual design of workflows.

4.2. Workflow Hosting with WF

WF workflows can be hosted in any Windows process ranging from simple console applications to large, sophisticated enterprise services. The workflow execution is carried out within the WF runtime engine, which provides important services, such as persisting workflow state, tracking the execution, and supporting transactions. The WF runtime can make decisions to unload long-running workflows that have been inactive for a period of time, and load them back when necessary. It also works as a proxy for all interactions between the workflow and external software, including Web services, in an asynchronous style.

Being an integrated part of the .NET Framework, it is possible for WF workflows to take advantage of other Microsoft technologies including Windows Communication Foundation (WCF) [21], Windows CardSpace [2] and MS SQL Server 2005 [20] to address problems such as communication in a dis-

tributed environment, federated security control and data access. Application of some of these technologies together with WF can be found in our work on the Earth system modelling workflows, which is discussed in the following section.

5. A Composite Workflow for GENIE Simulations

A number of key components have been developed for the GENIE project to facilitate the running of Earth system simulations. These include a shared database hosted in Oracle 10g which contains the definitions, states and results of the experiments, a number of gateway systems to integrate compute resources from different HPC platforms including Microsoft Windows Compute Cluster Server [18], Condor [32] and Globus [11], and a shared file store managed by using the Geodise Toolkit [10]. In order to bring these components together in a single system that performs the Earth system simulations, a composite workflow has been implemented using the WF technologies. Figure 2 shows the structure of the WF based Genie system.

The WF application complements the scripted Matlab clients and provides a constantly running process to progress the simulations in the experiment. The workflow runtime engine enacts the workflow of Figure 3 (which we discuss below). WF uses a Microsoft SQL Server database to persist the workflow. The *Resource* and *Job* services manage the interface between external events and the running workflow. Windows Communication Foundation and Windows Presentation Foundation are used to provide a robust interface to the workflow via a web browser for the end user. An Internet Explorer 7 browser is required for our implementation but in principle the interface could be made available via the web service to an Ajax enabled browser.

The design of the linear scripted workflow from the Matlab environment has been re-factored for asynchronous execution in an event driven environment. Figure 3 shows the sequence diagram for the WF system deployed to manage GENIE simulations. There are a number of key improvements to the design of the system:

(i) The user can interact with a continuously running workflow and is able to add and remove resource on which the compute jobs are performed. The workflow now manages the entire experiment through to completion and only terminates once all simulations are complete. WF provides the robust environment in which the workflow can remain operational for the duration of the experiment and therefore guarantees completion. This provide a much more reliable solution because it removes the dependence on participants maintaining cron jobs or scheduled tasks to regularly invoke the Matlab client.

(ii) Asynchronous calls are made to the methods of the *Find And Submit Work* and *Post Process* activities. This means that job polling, post-processing and the submission of new work can all be carried out concurrently. In an event driven environment this improves the efficiency with which compute jobs are managed and increases the throughput of the system.

(iii) The retrieval of job handles from the database is eliminated from the workflow because WF maintains its own register of active jobs. The active job handles are still archived after job submission in order to ensure that the Matlab clients can continue to function and operate along side the WF solution.

The Genie WF workflow implementation automates the operations and interactions on the system components. Figure 4 shows the main part of the design of the GENIE workflow in Microsoft Visual Studio 2005. The workflow is composed of activities that have been placed onto the design surface from the toolbox side bar. There are many types of activity that can be used to construct workflows in WF but we limit our discussion here to describing the main features exploited in the GENIE workflow. The toolbox provides access to a base activity library (BAL) comprising many of the basic logical building blocks required for most workflows and also to more complex activities, such as components for invoking web services. The toolbox also provides access to custom activities that are added by developers to extend the functionality of WF. For the GENIE application we have developed the *GenieActivityLibrary* that comprises four activities for the management of simulations.

These components are required to implement the functionality of the Matlab scripts from which the application is derived. In order to build these Grid-enabled activities the Matlab compiler was used to generate a binary executable form of each script. This had the great advantage of building the application with well tested, understood, reliable and ro-
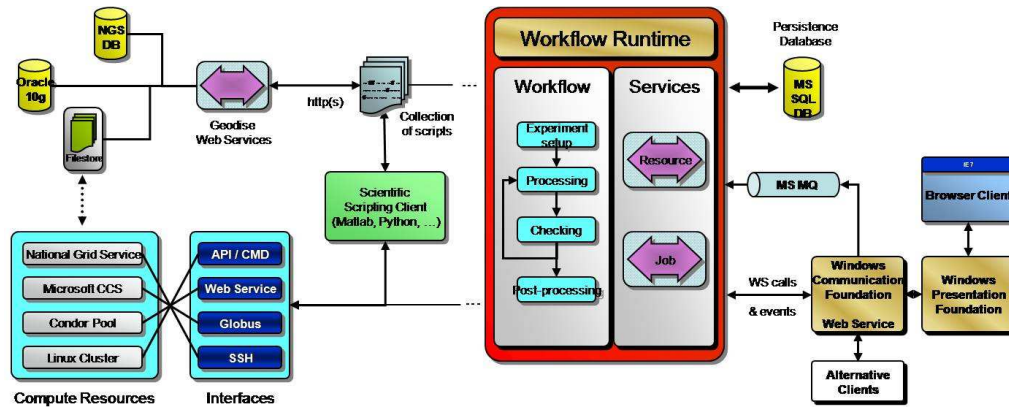
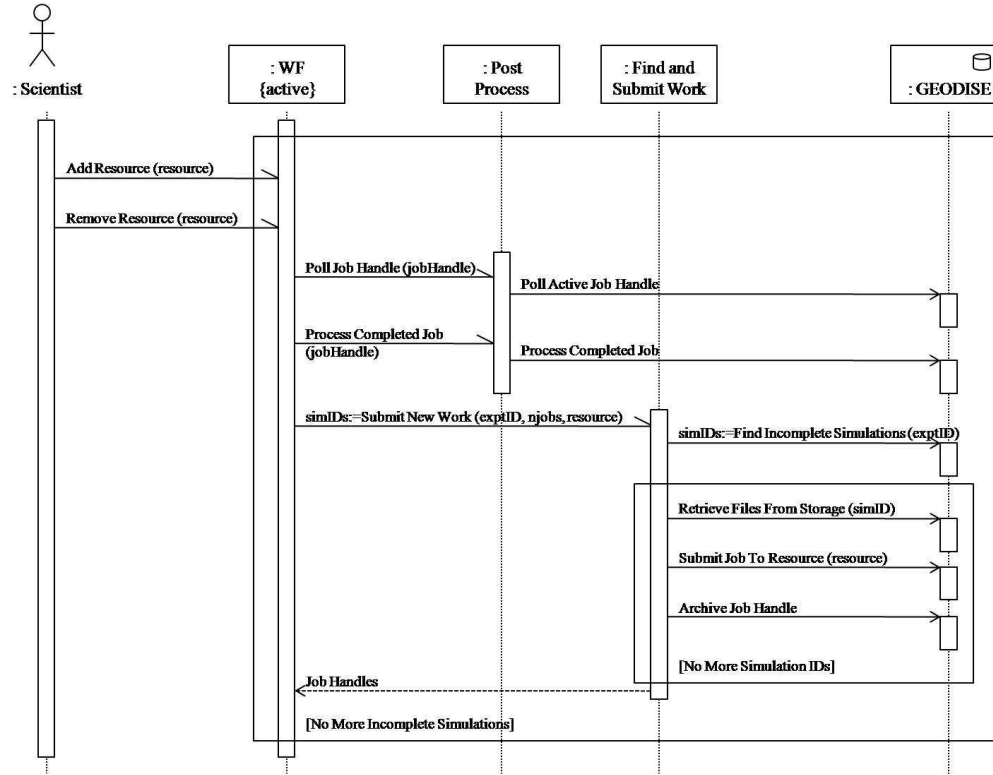Fig. 2. The Complete Structure of the WF based GENIE Simulation System.

Fig. 3. UML sequence diagram of the workflow enacted in Windows Workflow Foundation.

bust underlying functionality for interfacing to the GENIE database and managing simulations on heterogeneous Grid computing resource. The custom activities were generated by wrapping system calls to these Matlab executables. The separate activities provide the following functionality:

– *CreateExperimentActivity* enables simple one- and two-dimensional parametric studies to be defined in the database.
– *SpecifyResourceActivity* handles the addition and removal of resource definitions by the user.
– *FindAndSubmitWorkActivity* invokes the compiled binary version of the Matlab scripts that

query the database for incomplete simulations and submit compute jobs to available resource.
– *ImageActivity* generates plots to show the progress of the experiment.

The application coordinates these custom activities using a sequential workflow that consists of a while loop activity which contains a single parallel asynchronous event listening activity that waits for the following changes in the state of an experiment: newly specified resources, changes of job states and completion of work item post-processing. This not only provides a more responsive solution than the original scripted polling, but also achieves better ef-

ficiency and robustness of the system by allowing the runtime to persist workflow instances in a database and unload them during periods of idleness. All of the asynchronous events are indications that either compute resources have become available or subsequent work items are now available. Consequently, a *find work* activity is performed after any one of these events has occurred.

The Genie workflow employs a set of external data exchange services that run within the workflow runtime. These services provide the means by which events occurring within the experiment system are passed into the workflow. When compute jobs are submitted, the workflow registers their identifiers with the Job Service which is responsible for monitoring the jobs' running statuses. The Job Service notifies the workflow instance of changes to the job status by firing a batched 'jobStatusChanged' event. In addition, the Job Service also performs asynchronous post-possessing of results from completed jobs when the workflow receives a 'jobStatusChanged' event of either 'Completed' or 'Failed'. Another external data exchange service, the Resource Service, allows the users to instruct the workflow to create new experiment work units by registering additional compute resources with it.

Based on the Windows platform and the .NET 3.0 technologies, the GENIE workflow can easily be interacted with using secured and reliable communication supported by WCF. It is therefore possible to host the workflow on an independent server and steer the running of the workflow from a separate client, which can be disconnected from the GENIE system without affecting the running of the experiments. This mode is ideal for the Earth system simulations, which often run for a lengthy period of time. To facilitate the use of the GENIE system in such a disconnected client mode, we have developed a browser based client application using the Windows Presentation Foundation (WPF) [30]. The client can be loaded in a web browser on any computer with the .NET 3.0 runtime without installation. Figure 5 shows the running of the browser based client application in monitoring workflow status and displaying results.

6. Ocean Circulation Bi-stability Study

The world's oceans play an important role in determining climate both regionally and globally. The waters of the world's oceans circulate the planet in a system of surface and deep currents collectively known as the global ocean conveyor belt or thermohaline circulation. An important aspect of global ocean circulation is the strength of the meridional overturning circulation (MOC) in the Atlantic which affects the climate in North West Europe and elsewhere in the world. Under global warming scenarios, changes to atmospheric moisture transport properties could lead to a collapse of the Atlantic MOC from its present "on" state to an "off" state in which the northerly flow of warm surface water is dramatically reduced causing Europe to experience a relative cooling. Understanding the behaviour and sensitivity of the Atlantic MOC to changes in atmospheric moisture transports is therefore important in assessing the potential impacts of climate change. Of particular interest is the possibility of an irreversible transition between the "on" and "off" states. If anthropogenic climate change were to cause a collapse of the Atlantic MOC would there be a barrier to recovery to the present day "on" state? Furthermore, how would such a change in ocean circulation feed back on the level of atmospheric carbon dioxide (CO_2)?

To demonstrate the WF based GENIE system, we extend previous systematic 2-parameter studies of bi-stability in the Atlantic thermohaline circulation [14,6] using a new Earth system model from the GENIE framework. For this work we use a model that introduces atmospheric chemistry and ocean biogeochemistry modules and provides a full atmosphere–ocean carbon cycle. The study investigates the CBGOLDSTEIN composition constructed in the GENIE framework (*genie_eb_go_gs_ac_bg*), which consists of a 3D frictional geostrophic ocean coupled to a simple 2D energy-moisture balance model and a 2D sea-ice component. A carbon cycle is introduced by integrating atmosphere and ocean biogeochemistry components in the model. Following Marsh *et al.* [14] the study seeks to investigate the properties of the Atlantic MOC as a function of two of the key parameters that dominate atmospheric moisture transport in the atmosphere model:

– $\mathbf{FW}_x$: Zonal moisture transport from the Atlantic to the Pacific
– $\mathbf{K}_q$: Atmospheric moisture diffusivity - principally controls meridional moisture transport between the equator and poles

For this study each model is run for a total of $5,000$ simulation years requiring a total of ~ 10 hours of CPU time on an average desktop machine (e.g. Intel P4 2.6 GHz) compared to the ~ 2 hours
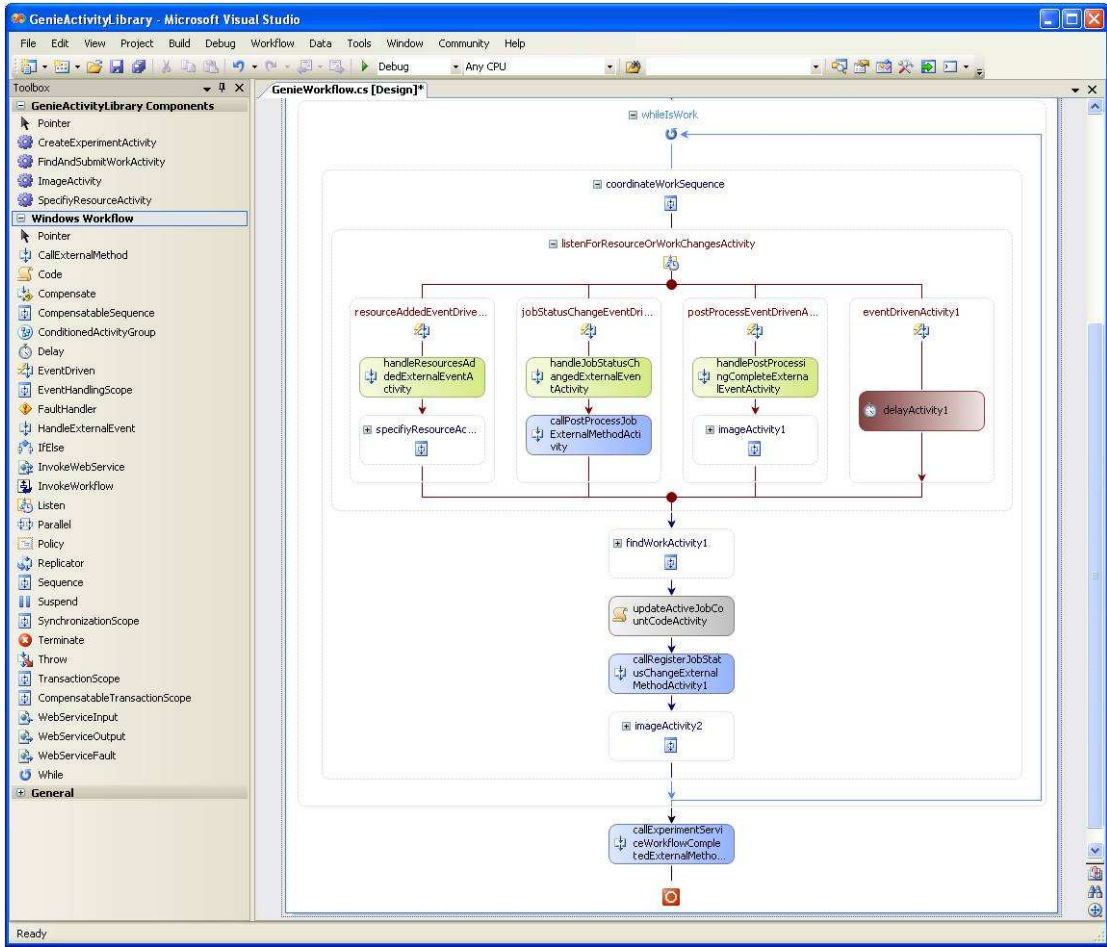
Fig. 4. Design of the GENIE workflow in the WF Designer with Visual Studio. (For clarity of presentation some activities related to WCF have been omitted).

for the previous study. This necessitates an alternative approach to the Condor cycle stealing methods used in the initial study [14]. We exploit the WF client in conjunction with Matlab clients to progress the simulations defined for the study using both institutional and national compute resources.

The experiment proceeds by performing an initial factorial sampling of the parameter space by defining an evenly distributed 11×11 model evaluations in the database. This comparatively coarse sampling is sufficient to identify a region of collapse in the Atlantic MOC. Taking the extreme end points from the "top-left" and "bottom-right" points of the initial sampling we obtain models with strong "on" and "off" states of the Atlantic MOC. Identical parame-

ter space samplings are then defined in the database using these end-states as initial boundary conditions for an identical set of simulations. These repeat ensembles therefore examine the stability of the "on" and "off" states as a function of the atmospheric moisture transport properties. Once the initial sampling is complete an analysis script in the Matlab client is used to add new evaluation points (simulation configuration files) in the parameter space in the regions of interest – in particular the emerging region of bi-stability of the model. The WF client manages the execution of the ensembles over the resources provided by the users. The addition of new ensemble members provides new work units for the workflow to manage and users resume the study at
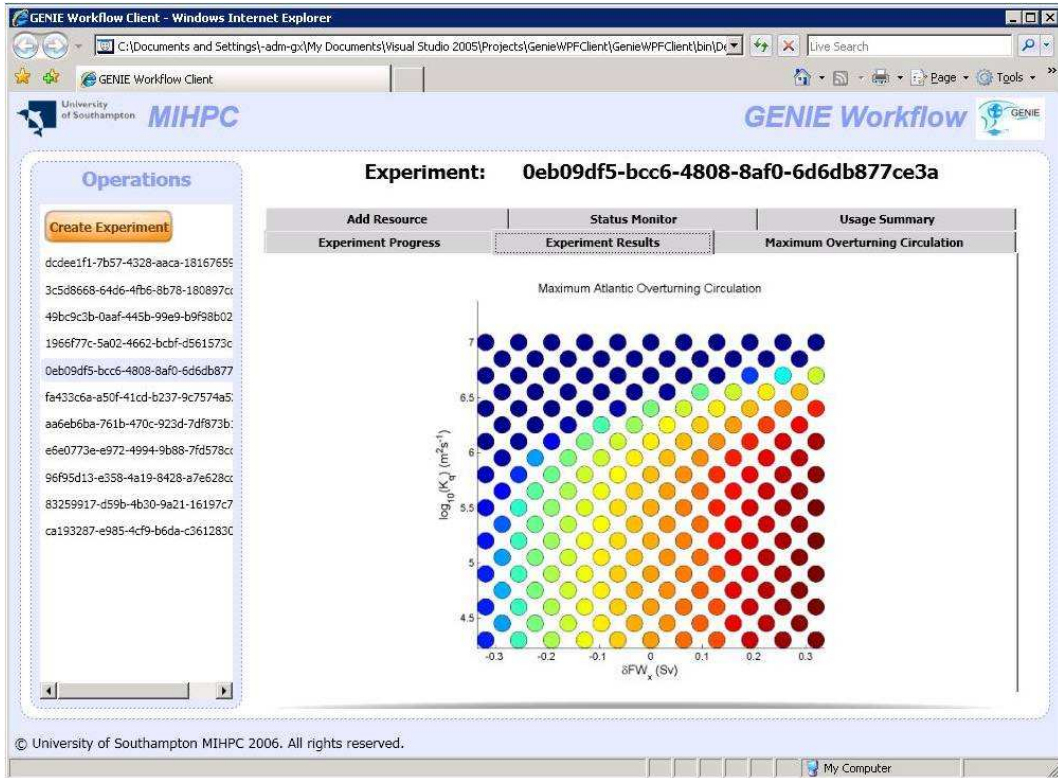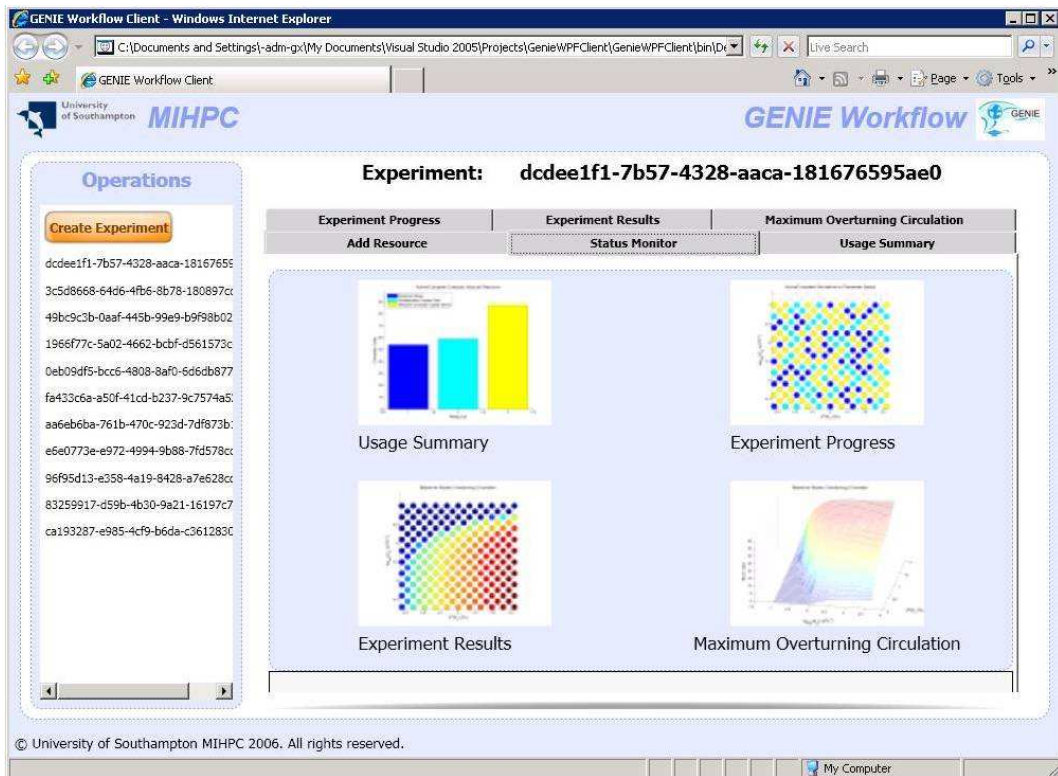
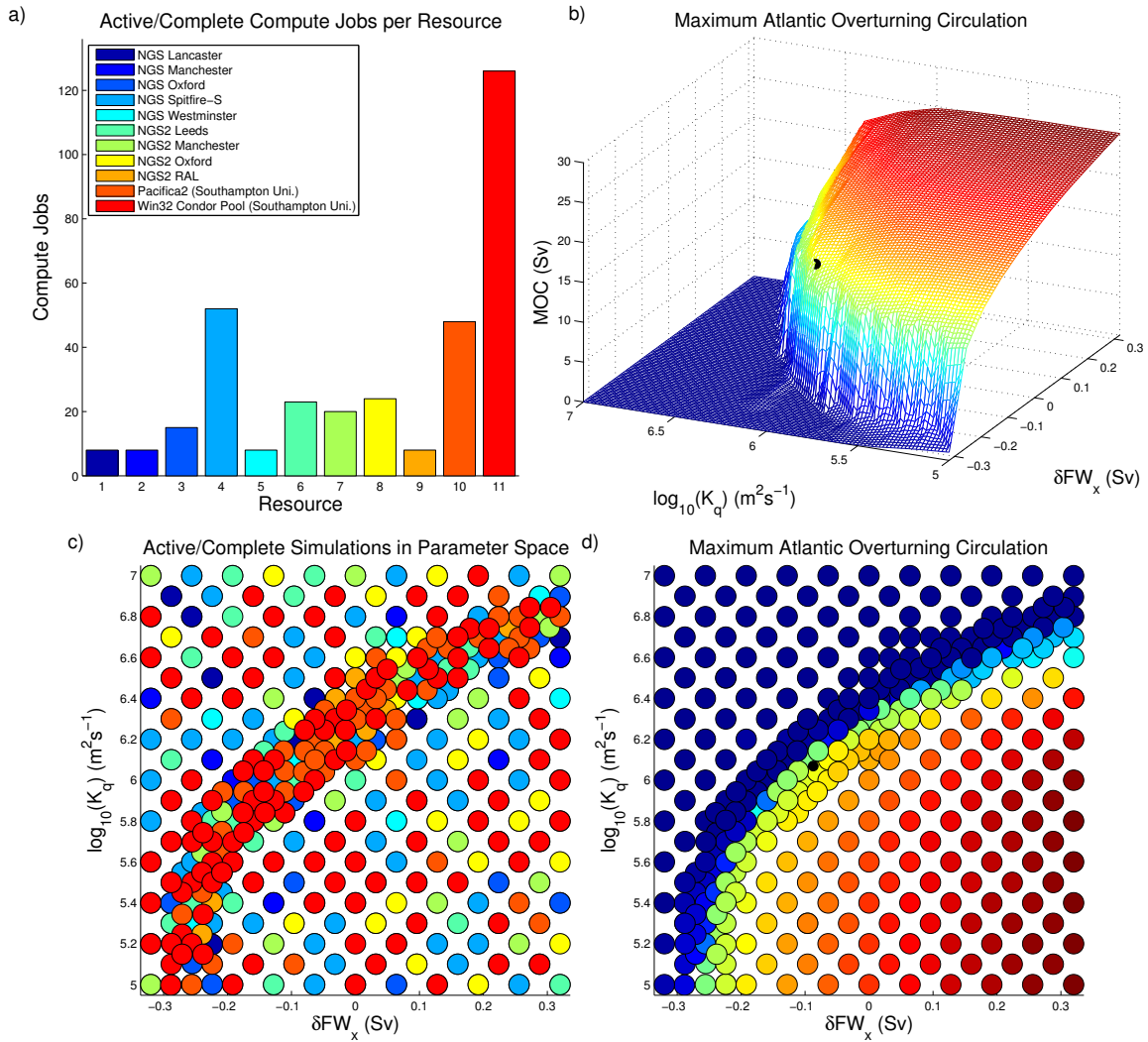Fig. 5. The GENIE Workflow Client Application Hosted in Internet Explorer 7.

Fig. 6. Study of Atlantic thermohaline circulation as a function of atmospheric moisture transport parameters. a) Summary of the distribution of compute jobs across the computational grid, b) maximum Atlantic MOC as a function of atmospheric moisture transport parameters, c) distribution of compute jobs in the parameter space and d) strength of maximum Atlantic overturning circulation in the parameter sweep.

their convenience.

The results of the initial sampling of atmospheric moisture transport parameter space are presented in Figure 6. Figure 6(a) summarises the resource usage in performing the initial ensemble. The distribution of compute jobs broadly reflects the relative numbers of single concurrent compute tasks that the schedulers on each system allow to a single user. The distribution of simulations across the resource pool is plotted in the parameter space in Figure 6(b). The user additions of model runs are focused along the "cliff-edge" of THC collapse (Figure 6(c, d)) where the narrow region of bi-stability emerges.

The EMBM atmosphere component of the GE-NIE model provides the option for the radiative forcing of the Earth system to be determined by the concentration of greenhouse gases in the atmosphere. We have performed this study using *genie_eb_go_gs_ac_bg* with both interactive and non-interactive CO_2 in the atmosphere. In the interactive CO_2 model, the energy balance at the surface of the planet is determined in part by the concentration of CO_2 in the atmosphere. Changes in the properties of the carbon cycle as a result of the moisture transport anomalies therefore feed back in to the radiation budget of the planet. In the
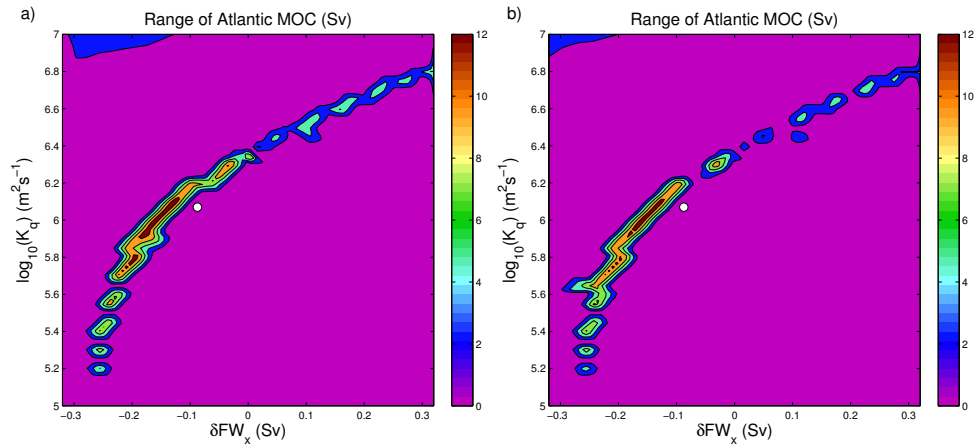
Fig. 7. Range of maximum Atlantic meridional overturning circulation as a function of the atmospheric moisture transport parameters for a) interactive CO_2 and b) non-interactive CO_2. The model with best fit to present day climate is indicated by the white dot.

non-interactive CO_2 model the radiative forcing due to CO_2 is fixed at 280 ppmv. Comparison of the bi-stability of the model with and without an interactive carbon cycle in the atmosphere provides insight into whether CO_2 enhances the stability of different states of the ocean circulation.

The bi-stability of Atlantic meridional overturning circulation is shown in figure 7 for interactive and non-interactive CO_2. The accompanying changes in the concentration of atmospheric carbon dioxide are shown in figure 8. Several new findings arise:

(i) The tuned 16-level model has a narrower region of bi-stability than the original 8-level model (comparing Fig. 7 with Fig. 8b in Marsh *et al.*, 2004 [14]).

(ii) Atmospheric carbon dioxide (CO_2) increases by about 20 ppmv when the Atlantic MOC collapses (see minima in Fig 8) because of a less effective 'solubility pump' of CO_2 from the surface to depth in the ocean, which is enhanced by a vigorous overturning circulation.

(iii) The range of maximum Atlantic MOC shows a slightly wider and more extensive region of bi-stability with interactive CO_2 (Fig. 7a) than without (Fig. 7b), indicating that the increase (decrease) in CO_2 caused by a collapse (recovery) of the Atlantic MOC acts to slightly maintain whatever is the initial state.

(iv) Significant differences in CO_2 are seen between different restarts in parts of the parameter space away from the 'threshold zone' (notably a 5–15 ppmv reduction restarting from the "off" state at $log_{10}(K_q) > 6.4$) - see Fig.

8e,f - because components of the circulation other than the Atlantic MOC are also changing, with implications for carbon uptake.

7. Conclusions

In this paper we have demonstrated how the Windows Workflow Foundation technologies can be exploited to implement scientific workflows for e-Science systems such as Earth system modelling in the GENIE project. We reviewed features of WF technologies and explained how they have been deployed in the GENIE framework to facilitate coordinated operations among various distributed components. The WF system improves on the previous Matlab scripting approach because it exploits an off-the-shelf workflow product and therefore benefits from support for event driven enactment, workflow persistence, improved fault tolerance and managed message queuing. The user experience in both design and management of GENIE workflows benefits from both the Microsoft Visual Studio IDE and the Windows Presentation Foundation which delivers a browser based client interface. The WF system is complementary to the existing software so the Matlab problem solving environment can still be used to perform numerical analysis of the results.

The WF application has been implemented by exploiting compiled versions of Matlab scripted activities. This ensures that the functionality of the simulation management system is identical to the client it complements. Future development of this system will implement a native interface to both the Globus
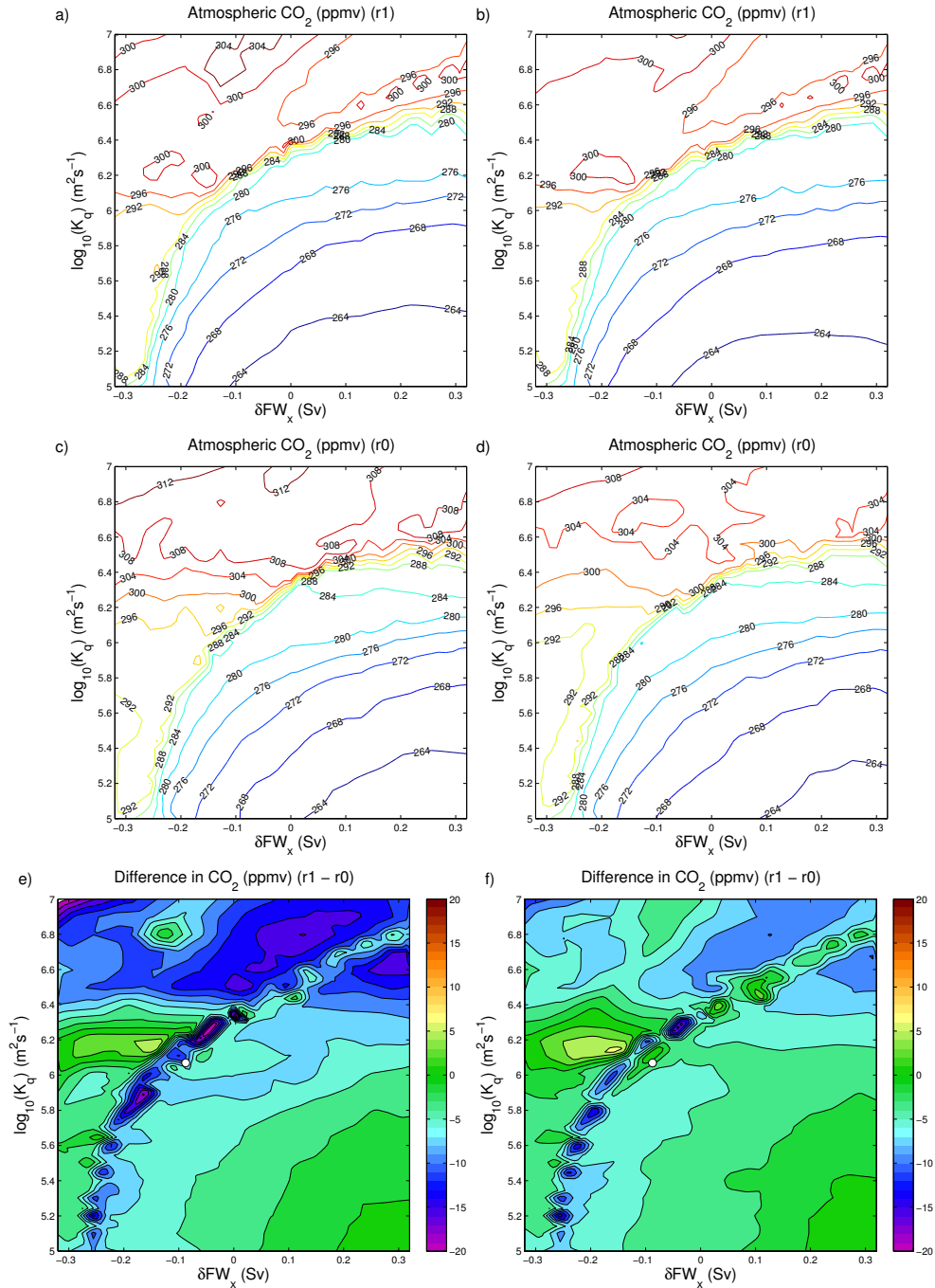
13

Fig. 8. Concentration of atmospheric CO_2 as a function of the atmospheric moisture transport parameters with interactive CO_2 and initialising the simulations in the a) "on" state (r1) and c) "off" state (r0); equivalent plots for non-interactive CO_2 initialised in the b) "on" state and d) "off" state. The differences in the concentration of atmospheric CO_2 due to the initial boundary conditions are shown for e) interactive CO_2 and f) non-interactive CO_2.

resources exploited for compute jobs and to the GE-NIE database web services interface. This will involve implementing the scripted Matlab workflows in WF. In addition to simplifying the overall deployment the implementation of the system in WF will also provide a more integrated error handling model

to deal with potential causes of failure in the system. With the Matlab activities the WF application cannot provide any additional error checking so any failures in the job life cycle can result in the work being abandoned and the simulation continued at a later invocation of the workflow. By providing an implementation in WF of the core activities we will be able to move away from the legacy persistence model where the Oracle database provides the fault tolerance.

A study of the bi-stability of the Atlantic meridional overturning circulation in a GENIE model has been carried out to investigate the effects of an interactive carbon cycle in the atmosphere. Atmospheric CO_2 feeds back (via the radiative "greenhouse" effect) on climate, with further implications for the ocean circulation and the carbon cycle. There is some evidence that the introduction of a full carbon cycle within the ocean and atmosphere components acts to enhance the bi-stability observed in the ocean circulation states. In a small range of 2-parameter space, where the two key parameters exert a strong influence on the hydrological cycle, the model Atlantic MOC is stable in two distinct states: "on" or "off". The Atlantic MOC plays an important role in the uptake of atmospheric CO_2: in the MOC "off" state, uptake is somewhat compromised, and atmospheric CO_2 levels are consequently higher by up to 20 ppmv. In conclusion, we note that such changes in CO_2, as a consequence of major changes in the hydrological cycle and the global ocean circulation, may have occurred under glacial climate, when these key elements of the Earth System were episodically disrupted [15]. It remains to see whether a similar positive feedback unfolds in the coming decades, as rising CO_2 levels drive climate change that may trigger a substantial reduction in the Atlantic MOC [16], leading to further increases of atmospheric CO_2.

Acknowledgements

References

[1] R. Buyya, D. Abramson, J. Giddy, Nimrod/G: An architecture of a resource management and scheduling system in a global computational grid, in: HPC Asia 2000, Beijing, China, 2000.

[2] D. Chappell, Introducing Windows CardSpace, Tech. rep., Chappell and Associates, http://msdn2.microsoft.com/en-us/library/Aa480189.aspx (Apr. 2006).

[3] D. Churches, G. Gombas, A. Harrison, J. Maassen, C. Robinson, M. Shields, I. Taylor, I. Wang, Programming scientific and distributed workflow with Triana services, Concurrency and Computation: Practice and Experience 18 (10) (2006) 1021–1037.

[4] M.H. Eres, G.E. Pound, Z. Jiao, J.L. Wason, F. Xu, A.J. Keane, S.J. Cox, Implementation and utilisation of a Grid-enabled problem solving environment in Matlab, Future Generation Computer Systems 21 (6) (2005) 920–929.

[5] S. Eswaran, D. Del Vecchio, G. Wasson, M. Humphrey, Adapting and evaluating commercial workflow engines for e-Science, in: P. Kellenberger (ed.), Proceedings of the 2nd IEEE International Conference on e-Science and Grid Computing (e-Science'06), IEEE Computer Society, Los Alamitos, CA, USA, 2006.

[6] M.J. Fairman, A.R. Price, G. Xue, M. Molinari, D.A. Nicole, T.M. Lenton, R. Marsh, K. Takeda, S.J. Cox, Building scientific workflows for earth system modelling with windows workflow foundation, in: Y. Shi, G.D.v. Albada, J. Dongarra, P.M.A. Sloot (Eds.), Computational Science - ICCS 2007, vol. 4489 of Lecture Notes in Computer Science, Springer, Berlin / Heidelberg, Beijing, China, 2007.

[7] J. Feng, L. Cui, G. Wasson, M. Humphrey, Toward seamless grid data access: design and implementation of GridFTP on .NET, in: Proceedings of the 6th IEEE/ACM International Workshop on Grid Computing, 2005.

[8] I. Foster, C. Kesselman, The Grid 2: Blueprint for a new computing infrastructure, Morgan Kaufmann, San Francisco, CA, 2004.

[9] G.C. Fox, D. Gannon, Special issue: Workflow in Grid systems, Concurrency and Computation: Practice and Experience 18 (10) (2006) 1009–1019.

[10] The GEODISE project, http://www.geodise.org/ (2007).

[11] The Globus Alliance, http://www.globus.org/ (2007).

[12] T.M. Lenton, R. Marsh, A.R. Price, D.J. Lunt, Y. Aksenov, J.D. Annan, T. Cooper-Chadwick, S.J. Cox, N.R. Edwards, S. Goswami, J.C. Hargreaves, P.P. Harris, Z. Jiao, V.N. Livina, A.J. Payne, A.J. Ridgwell, I.C. Rutt, J.G. Shepherd, P.J. Valdes, G. Williams, M.S. Williamson, A. Yool, A modular, scalable, Grid ENabled Integrated Earth system modeling framework: Effect of dynamical atmosphere model, surface grid, and ocean resolution on the stability of the thermohaline circulation, Climate Dynamics 29 (6) (2007) 591–613.

[13] B. Ludäscher, I. Altintas, C. Berkley, D. Higgins, E. Jaeger, M. Jones, E.A. Lee, J. Tao, Y. Zhao, Scientific workflow management and the Kepler

system, Concurrency and Computation: Practice and Experience 18 (10) (2006) 1039–1065.

[14] R. Marsh, A. Yool, T.M. Lenton, M.Y. Gulamali, N.R. Edwards, J.G. Shepherd, M. Krznaric, S. Newhouse, S.J. Cox, Bistability of the thermohaline circulation identified through comprehensive 2-parameter sweeps of an efficient climate model, Climate Dynamics 23 (2004) 761–777.

[15] J.F. McManus, R. Francois, J.-M. Gherardi, L.D. Keigwin, S. Brown-Leger, Collapse and rapid resumption of Atlantic meridional circulation linked to deglacial climate changes, Nature 428 (2004) 834–837.

[16] G.A. Meehl, T.F. Stocker, W.D. Collins, P. Friedlingstein, A.T. Gaye, J.M. Gregory, A. Kitoh, R. Knutti, J.M. Murphy, A. Noda, S.C.B. Raper, I.G. Watterson, A.J. Weaver, Z.-C. Zhao, Changes in the Atlantic Meridional Overturning Circulation, in: S. Solomon, D. Qin, M. Manning, Z. Chen, M. Marquis, K.B. Averyt, M. Tignor, H.L. Miller (Eds.), Climate Change 2007: The Physical Science Basis. Contribution of Working Group I to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change, Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA, 2007, pp. 772–777.

[17] K. Matsumoto, K.S. Tokos, A.R. Price, S.J. Cox, GENIE-M: a new and improved GENIE-1 developed in Minnesota, Geoscientific Model Development Discussions 1 (1) (2008) 1–37.

[18] Microsoft Windows Compute Cluster Server 2003 product overview, http://www.microsoft.com/technet/ccs/overview.mspx (2006).

[19] Microsoft .NET Framework 3.0, http://msdn2.microsoft.com/en-gb/netframework/default.aspx (2007).

[20] Microsoft SQL Server 2005 books online, http://technet.microsoft.com/en-gb/sqlserver/bb428874.aspx (2007).

[21] Microsoft Windows Communication Foundation architecture overview, Tech. rep., Microsoft Corporation, http://msdn2.microsoft.com/en-us/library/Aa480210.aspx (Mar. 2006).

[22] Microsoft XAML overview, http://msdn2.microsoft.com/en-us/library/ms752059.aspx (2007).

[23] T. Oinn, M. Greenwood, M. Addis, M.N. Alpdemir, J. Ferris, K. Glover, C. Goble, A. Goderis, D. Hull, D. Marvin, P. Li, P. Lord, M.R. Pocock, M. Senger, R. Stevens, A. Wipat, C. Wroe, Taverna: lessons in creating a workflow environment for the life sciences, Concurrency and Computation: Practice and Experience 18 (10) (2006) 1067–1100.

[24] A. Paventhan, K. Takeda, S.J. Cox, D.A. Nicole, Leveraging windows workflow foundation for scientific workflows in wind tunnel applications, in: Proceedings of the 22nd International Conference on Data Engineering Workshops (ICDEW'06), IEEE Computer Society, Los Alamitos, CA, USA, 2006.

[25] A. Paventhan, K. Takeda, S.J. Cox, D.A. Nicole, MyCoG.NET: a multi-language CoG toolkit, Concurrency and Computation: Practice and Experience 19 (14) (2007) 1885–1900.

[26] A.R. Price, Z. Jiao, I.I. Voutchkov, T.M. Lenton, G. Williams, D.J. Lunt, R. Marsh, P.J. Valdes, S.J. Cox, Collaborative study of GENIEfy Earth system models using scripted database workflows in a Grid-enabled PSE, in: S.J. Cox (Ed.), Proceedings of the UK e-Science All Hands Meeting 2006, EPSRC, Nottingham, UK, 2006.

[27] A.R. Price, I.I. Voutchkov, G.E. Pound, N.R. Edwards, T.M. Lenton, S.J. Cox, Multiobjective tuning of Grid-enabled Earth system models using a non-dominated sorting genetic algorithm (NSGA-II), in: P. Kellenberger (Ed.), Proceedings of the 2nd IEEE International Conference on e-Science and Grid Computing (e-Science'06), IEEE Computer Society, Los Alamitos, CA, USA, 2006.

[28] A.R. Price, G. Xue, A. Yool, D.J. Lunt, P.J. Valdes, T.M. Lenton, J.L. Wason, G.E. Pound, S.J. Cox, Optimisation of integrated Earth system model components using Grid-enabled data management and computation, Concurrency and Computation: Practice and Experience 19 (2) (2007) 153–165.

[29] M. Romberg, The UNICORE Grid infrastructure, Scientific Programming Journal 10 (2) (2002) 149–157.

[30] C. Sells, I. Griffiths, Programming Windows Presentation Foundation, O'Reilly, Sebastopol, CA, 2005.

[31] D. Shukla, B. Schmidt, Essential Windows Workflow Foundation, Addison Wesley Professional, Crawfordsville, Indiana, 2006.

[32] D. Thain, T. Tannenbaum, M. Livny, Distributed computing in practice: the Condor experience, Concurrency and Computation: Practice and Experience 17 (2–4) (2005) 323–356.

[33] G. von Laszewski, M. Hategan, Workflow concepts of the Java CoG kit, Journal of Grid Computing 3 (3) (2005) 239–258.

[34] A. YarKhan, K. Seymour, K. Sagi, Z. Shi, J. Dongarra, Recent developments in Gridsolve, International Journal of High Performance Computing Applications 20 (1) (2006) 131–141.

16

Vitae

Matthew J. Fairman is a Research Fellow in the School of Engineering Sciences and a Software Architect with the Microsoft HPC Institute at the University of Southampton (UK). He holds a MEng in Computer Engineering and a PhD in Computer Science. His research interests include Grid computing and computational clustering. Since 2004 he has held several technical positions on Grid, e-Science, aeronautic, and commercial projects.

Andrew R. Price is a Senior Research Fellow in the School of Engineering Sciences at the University of Southampton. He received his doctorate in Electronics and Computer Science and degree in Physics at the University of Southampton. His research interests include Grid and e-Science technologies, distributed Problem Solving Environments and data management for Grid computing applications.

Gang Xue is a Research Fellow in the School of Engineering Sciences and Software Architect with the Microsoft HPC Institute at the University of Southampton. He received a PhD in Computer Science and a BSc in Electronics Engineering. His research interests include Grid computing, Web services and engineering design optimisation.

Marc Molinari is a Lecturer in the School of Engineering Sciences and Senior Application Engineer with the Microsoft HPC Institute at the University of Southampton (UK). He holds a PhD in Computer Science and received a first class BSc (Hons) degree in Physics with Electronics and Computing. His research interests include computational electromagnetics, engineering design search, and HPC. Since 2001 he has held several technical positions on Grid and e-Science related projects and is the author of the XML Toolbox for Matlab.

Denis A. Nicole received his BA in Natural Sciences from St John's College Cambridge in 1975 and a PhD in Theoretical Physics from the University of Southampton in 1978. He worked at the Universities of Pittsburgh and California (Santa Barbara) for six years before returning to Southampton in 1984. He was appointed a Reader in 1993. For the following ten years he was intimately associated with the development of Supercomputers based on the UK's Inmos Transputer and with the occam programming language, including the design of two ranges of machines from Parsys Ltd and a period of secondment to Inmos. Since then, he has concentrated on High Performance Computing software, including twice leading the technical work for UK national supercomputer procurements. He is a Chartered Electrical Engineer.

Tim Lenton gained a PhD (1998) in Environmental Sciences from the University of East Anglia, and then worked for 6 years at the Centre for Ecology and Hydrology in Edinburgh as an Earth system modeller, before returning to UEA in 2004, where he is now Professor of Earth System Science. His research focuses on understand the behaviour of the Earth as a whole system and developing the corresponding Gaia theory. He was awarded a Philip Leverhulme Prize in 2004, a European Geosciences Union Outstanding Young Scientist Award 2006, and gave the BA Charles Lyell Award Lecture 2006.

Robert Marsh is a Reader in Physical Oceanography in the School of Ocean and Earth Science at the University of Southampton, specializing in the development and use of Ocean, Climate and Earth System Models. High-resolution ocean models are evaluated alongside in situ and remotely-sensed observations, to investigate changes in the Atlantic circulation, heat content and water masses over recent decades. Earth System Models are developed in the new GENIE framework and subsequently used to quantify uncertainty in climate change over the next millennium and to gain a better understanding of glacial-interglacial cycles, through forward modelling of climate proxies such as oxygen isotopes.

Kenji Takeda is lecturer in Aeronautics at the University of Southampton. He teaches flight mechanics and race car design. He has a doctorate in Computer Science/Aeronautics on the subject of Discrete Vortex Methods. His research includes aerodynamics, aeroacoustics, flight simulation and supercomputing. He is a director of the Microsoft Institute for High Performance Computing and also runs the national UK airliner design competition at www.futureflight.org.

Simon J. Cox is Professor of Computational Methods in the Computational Engineering Design (CED) Research Group within the School of Engineering Sciences, University of Southampton. He is the Technical Director of the Southampton Regional e-Science Centre and Director of the Microsoft Institute for High Performance Computing. He has a PhD in Electronics and Computer Science, degrees in Mathematics and Physics and has published over 100 papers and regularly speaks at prestigious national and international conferences. He currently heads a team of PGs and RAs in the CED Group that is applying and developing high

performance computing in a variety of collaborative interdisciplinary computational science and engineering projects.