

# A Semantic Workflow Mechanism to Realise Experimental Goals and Constraints

Edoardo Pignotti, Peter Edwards  
School of Natural & Computing Sciences  
University of Aberdeen  
Aberdeen, AB24 5UE, Scotland  
{e.pignotti, p.edwards}@abdn.ac.uk

Gary Polhill, Nick Gotts  
The Macaulay Institute  
Craigiebuckler  
Aberdeen, AB15 8QH, UK  
{g.polhill, n.gotts}@macaulay.ac.uk

Alun Preece  
School of Computer Science  
Cardiff University  
Cardiff, CF24 3AA, UK  
A.D.preece@cs.cf.ac.uk

## Abstract

*Workflow technologies provide scientific researchers with a flexible problem-solving environment, by facilitating the creation and execution of experiments from a pool of available services. In this paper we argue that in order to better characterise such experiments we need to go beyond low-level service composition and execution details by capturing higher-level descriptions of the scientific process. Current workflow technologies do not incorporate any representation of such experimental constraints and goals, which we refer to as the scientist's intent. We have developed a framework based upon use of a number of Semantic Web technologies, including the OWL ontology language and the Semantic Web Rule Language (SWRL), to capture scientist's intent. Through the use of a social simulation case study we illustrate the benefits of using this framework in terms of workflow monitoring, workflow provenance and enrichment of experimental results.*

## 1. Introduction

In recent years researchers have become increasingly dependent on scientific resources available through the Internet, including computational modelling services and datasets. This is changing the way in which research is conducted with increasing emphasis on 'in silico' experiments as a way to test hypotheses. Scientific workflow technologies [22] have emerged in recent years to allow researchers to create and execute experiments given a pool of available

services. However, the current generation of technologies can only capture the experimental method and not the associated constraints and goals, which is essential if such experiments are to be truly transparent.

Many different workflow languages exist including: MoML (Modelling Markup Language) [14], BPEL (Business Process Execution Language) [2], ScufI (Simple conceptual unified flow language) [21]. A number of tools are available for creating and enacting workflows most notably Taverna [20] and Kepler [15]. Taverna (based on the ScufI language) is a tool developed by the myGrid<sup>1</sup> project to support 'in silico' experimentation in biology. It provides an editor tool for the creation of workflows and the facility to locate services from a directory via an ontology-driven search facility. Semantic support in Taverna allows the description of workflow activities but is limited to facilitating the discovery of suitable services during the design of a workflow. Kepler [15] is a workflow tool based on the MoML language; Web and Grid services, Globus Grid jobs, and GridFTP can be used as components in the workflow. Kepler extends the MoML language by introducing the concept of a Director, to define execution models and monitor the workflow.

These languages and tools are designed to capture the flow of information between services (e.g. service addresses and relations between inputs and outputs). We argue that in order to fully characterise scientific analysis we need to go beyond such low-level descriptions by capturing the experimental conditions. The aim here is to make the constraints and goals of the experiment, which we describe

---

<sup>1</sup>[www.mygrid.org.uk](http://www.mygrid.org.uk)

as the *scientist's intent*, transparent. We argue that this is particularly important as there is an increasing need to capture the provenance associated with experimental workflows. Provenance (also referred to as lineage or heritage) aims to provide additional documentation about the processes that led to creation of a resource [12]. Goble [10] expands on the Zachman Framework [29] by presenting the '7 W's of Provenance': *Who, What, Where, Why, When, Which, & (W)How*. While some progress has been made in terms of documenting processes [11] (*Who, What, Where, When, Which, & (W)How*), little effort has been devoted to the *Why* aspect of research methodology. We feel that by capturing scientist's intent we could provide more information about the *Why*.

In this paper we discuss a framework [23] for capturing scientist's intent, based upon rules which operate on workflow metadata. Others, most notably the SEEK [17] project have identified the need to develop metadata-driven tools to support complex multi-domain workflow experiments [5]. Our framework requires that both the workflow environment and the services invoked by the workflow have rich metadata support. The Kepler workflow environment is ideally suited for our framework as it uses OWL ontologies to support semantic annotation of dataset schemas, activities and their corresponding input and outputs; to provide classification and browsing of workflow activities; to check if the workflow is semantically consistent; and to search for contextually relevant activities during workflow design. Moreover, Kepler can make use of available Grid and Web services as part of the workflow. However, traditional service description languages such as WSDL<sup>2</sup> lack the semantic support required by our framework. For this reason we have developed a service infrastructure which is designed around the vision of the Semantic Grid [27] which combines Semantic Web and Grid technologies. Where Grid technologies [8] provide an infrastructure to manage distributed computational resources, the vision of the Semantic Grid is based upon the adoption of metadata and ontologies to describe resources (services and data sources) in order to promote enhanced forms of collaboration among the research community. Two major technologies has been considered in this respect. The first one is WSMO (Web Service Modelling Ontology) [26] which provides ontological specifications to describe the core elements of Semantic services and the goals associated with the use of the services by a client. The second one is OWL-S [16] which is an ontology of services based on OWL. OWL-S is designed to enable automation of Web Service discovery, invocation, composition, interoperation and execution. Both ontologies have been used in the context of Grid services [6] [4] and both can be integrated into our framework.

Throughout this paper we use a social simulation case-

study to highlight some of the limitations of current workflow technologies, and to illustrate how these can be addressed using our framework. This case study is based on FEARLUS (Framework for Evaluation and Assessment of Regional Land Use Scenarios) [25], an agent-based model developed to investigate land-use change in rural Scotland. Agent-based social simulation (ABSS) has been mooted as a third way to study social systems [19][3] with representations that are more descriptive than traditional analytical approaches, whilst still retaining their formality. Output can consist of hundreds of megabytes of data, and thorough exploration of parameter spaces can require significant CPU resources. Also, the heterogeneity of computing environments can make modelling software hard for others to install or use. These issues have led to calls for greater openness in the modelling community [1]. An earlier project involving the authors (FEARLUS-G) demonstrated the benefits that Semantic Grid technology can bring to ABSS [25], but only for one particular model. More general solutions are needed to enable ABSS model builders to capitalise on these benefits.

This paper is organized as follows: Section 2 discusses some of the limitations of current workflow technologies through the use of a social simulation case study. In section 3 we present a framework for capturing scientist's intent and a semantic workflow infrastructure which implements this framework. Section 3 continues by discussing some examples of how scientist's intent can be used to enrich workflow results, monitor and control workflow execution and to enhance workflow provenance. Finally, in section 4 we discuss future work and conclusions.

## 2. Deeside Case-Study

The focus of the Deeside case-study is on land use change patterns in the Upper Deeside region of North East Scotland between 1988 and 2004. Both qualitative data from interviews, and quantitative data from existing datasets, are used to build, calibrate and validate a case-study specific model. This is based upon a refined version of the pre-existing FEARLUS modelling framework. Specific foci of the case-study are on the drivers and processes of land use change, and the particular role of social networks in these processes. Once validation is complete, the intention is to use the model in policy-relevant, scenario-based studies of the future of Upper Deeside and similar regions, over the period to 2050.

Qualitative research is used to inform a series of refinements to the FEARLUS modelling system to create a framework capable of modelling the scenarios with an acceptable level of detail. Overall, the method takes an iterative approach, in which questions to be addressed in qualitative interviews are derived from issues arising from model de-

<sup>2</sup><http://www.w3.org/TR/wsd1>

velopment, and changes to the model are suggested by findings from qualitative interviews. This is in line with the TAPAS (Take A Previous model and Add Something) approach advocated by Frenken [9], who points out that incremental modelling strategies are more successful, faster to build, and easier to understand by others (presumably familiar with the previous model).

For calibration and subsequent validation of the macro-level outcomes of the FEARLUS Deeside case study model over the period 1988-2004, quantitative information is required. Available data on the changes in land use in the Grampian region of Scotland (and information on farm size change so far as this can be obtained) are used for input calibration and macro-validation of the model. Experiments assess whether the model is able to reproduce the direction and magnitude of the trends found in the data concerning land use and farm size, given the best available data relevant to model inputs.

The general approach being taken is as follows:

- Select those aspects of the world that can be represented in some way by inputs or outputs of the model. Some of these aspects (e.g. farmer decision-making procedures, climatic and economic conditions, available land uses) are inputs to the model; others (land use distribution and farm size) are outputs.
- For each of these aspects, determine what data are available for the period from the mid-1980s to the present. Farmer decision-making procedures in the model have been validated, as far as this is possible, using qualitative data from semi-structured interviews, as discussed above.
- Where there are data relevant to input parameters, determine how it can best be encoded in those parameters.
- Where there are no data good enough to be worth using for a particular input parameter, select a range of plausible combinations of parameter values with which to run the model.
- Explore a combination of parameter values by creating many runs of the simulation model for each parameter set. The best parameters are then selected based on how the simulation results match the real-world data and will be used in the qualitative validation phase.

The workflow shown in Figure 1 is designed to perform the model calibration process using a number of computational and data services. A range of possible combinations of parameter values are explored, e.g. combinations of Aspiration Threshold, Off-Farm Income, Approval Weighting, etc. The exploration of such parameters is based on close examination of the currently available quantitative data on

changes over time in land use and farm size. Real-world data from 1992, 1996, 2000 and 2004 (Calibration Data) is compared with values from the model for the same years. As many runs as possible are carried out for each parameter set (e.g. 50) depending on available computational resources. Results from the first calibration phase are then used to produce the best parameter sets for use in the quantitative validation phase.

The experimental workflow in Figure 1 has some limitation as it is not able to capture the goals and constraints associated with the experiment. For example, it is not clear from the workflow that the goal of this experiment is to obtain at least one match where the real data falls within 95% of the confidence interval of the model value. The researcher knows that if in a simulation run, one land manager owns more than half of the land, the entire simulation can be discarded. The researcher might also be concerned with the platform on which the comparison test runs, specifically if the platform is compatible with IEEE 754<sup>3</sup> as this could change the results of the simulation model. It may also be important to record special conditions, for example whether a variable's real-world value is within the range of values produced by the model runs; any range outside 95% confidence limits would suggest either a problem with the data, or flaws in the model, and merit detailed investigation. We argue that existing workflow languages are unable to convey such intent information as they are designed to capture low-level service composition rather than higher-level descriptions of the experimental process.

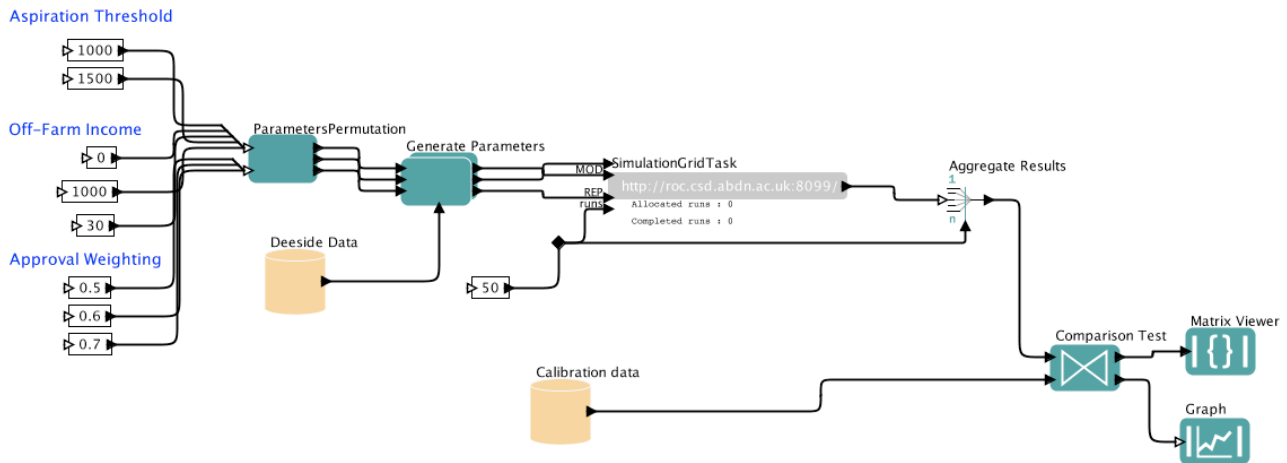
### 3. Scientist's Intent Support

As mentioned earlier, we have developed a framework [23] for capturing scientist's intent based upon rules. These rules act upon metadata generated from workflow activities (e.g. inputs, outputs, service execution). Details of the intent are kept separate from the operational workflow, as embedding intent information directly into the workflow representation would make it overly complex (e.g. with a large number of conditionals) and limit potential for sharing and re-use. We have chosen SWRL<sup>4</sup> (Semantic Web Rule Language) to represent such rules. SWRL enables Horn-like rules to be combined with metadata. The main challenges are to represent scientist's intent in such a way that:

- It is meaningful to the researcher, e.g. providing information about the context in which an experiment has been conducted so that the results can be interpreted;
- It can be reasoned about by a software application, e.g. an application can make use of the intent information

<sup>3</sup><http://grouper.ieee.org/groups/754/>

<sup>4</sup><http://w3.org/Submission/SWRL>



**Figure 1. Example Workflow for the Deeside Case Study (Calibration Phase).**

to control, monitor or annotate the execution of a workflow;

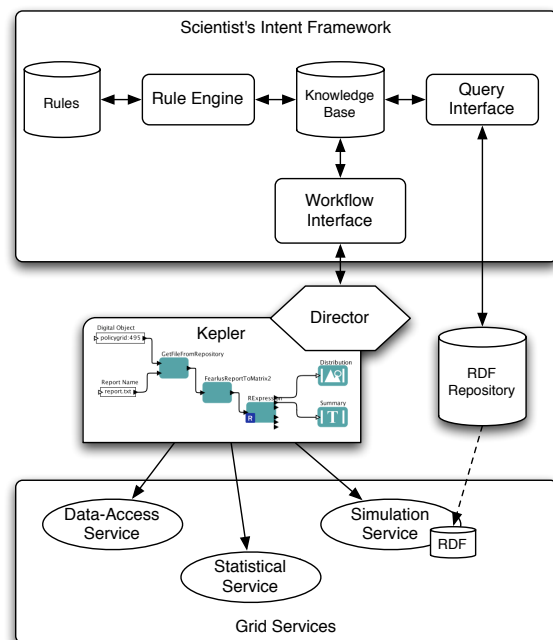
- It can be re-used across different workflows, e.g. the same high-level intent may apply to different workflows;
- It can be used as provenance (documenting the process that led to some result).

Figure 2 shows a semantic workflow infrastructure based on the Scientist's Intent framework. At the centre of this infrastructure we have the Kepler workflow tool which allows the user to design and enact workflows from local and remote services.

A crucial aspect of our framework is that the workflow and its component activities (e.g. *ParameterPermutation*, *SimulationGridTask*) must have supporting ontologies and should produce metadata that can be used against scientist's intent to reason about the workflow. We have identified the following possible sources of metadata:

- metadata about the result(s) generated upon completion of the workflow;
- metadata about the data generated at the end of an activity within the workflow or sub-workflow;
- metadata about the status of an activity over time, for example while the workflow is running.

We have implemented a number of Grid services and supporting ontologies: a data access service to enable access to large-scale data-sets; a service for statistical analysis based on R<sup>5</sup> and a number of simulation services running



**Figure 2. Semantic Workflow Infrastructure.**

<sup>5</sup><http://www.r-project.org/>

different versions of land-use and ecology simulation models. If the execution of a service produces a large amount of metadata at runtime (e.g. a simulation service), an RDF repository for each of the service instances is created. The core of the implementation is the knowledge-base repository where metadata from the workflow is translated into “facts” by the workflow interface component. The workflow interface component collects metadata every time it becomes available from the workflow. Such metadata is converted from RDF to facts represented as n-place predicates (e.g. `father(Alfred, Bob)`) and imported into the knowledge-base. A rule store contains all the rules generated by the user which are used to infer new facts (e.g. `IF father(?x, ?y) AND father(?y, ?z) THEN grandfather(?x, ?z)`). The rule engine processes such rules when new facts become available and stores the inferred facts back in the knowledge-base. The same engine is able to perform reasoning over an ontology, to infer additional facts. We have extended the Kepler Director component to communicate with the scientist’s intent framework. It is able to extract metadata from the workflow during execution, and can perform actions resulting from scientist’s intent rules. Finally, as some services generate a large amount of metadata, a query interface is used to extract only the metadata required by the intent rules from the associated RDF repositories. This is achieved by creating SPARQL<sup>6</sup> queries based on the scientist’s intent rules. This is facilitated by the fact the the rules are expressed in SWRL and the metadata required is explicitly referenced in the rule formalism.

To illustrate, in the Deeside case-study, the FEARLUS model implements a mechanism to describe the status of the agents during the simulation using RDF metadata [24]. This metadata can be used as the basis to define scientist’s intent rules. The example rule below defines the goal of the Deeside calibration experiment:

```
Pre Condition:
  ParameterSet( ?x1 ) ^
  DataSet( ?x2 ) ^
  ComparisonTest( ?x3 ) ^
  compares( ?x3, ?x1 ) ^
  compares( ?x3, ?x2 ) ^
  similarity( ?x3, ?x4 ) ^
  [more-than ( ?x4, 98%) = true]
```

This states that the goal is to obtain at least one match where the real data falls within 95% confidence interval of the model value. This is achieved when a specific precondition occurs based on the workflow metadata. Param-

<sup>6</sup><http://www.w3.org/TR/rdf-sparql-query/>

eterSet, DataSet and ComparisonSet refer to ontological classes, compares and similarity are properties in those classes and more-than is a built-in function used to test the value of the similarity property.

We will now present some examples of goals and constraints to illustrate the benefits of scientist’s intent in terms of enriching workflow results, support for monitoring and controlling the workflow, and workflow provenance support.

### 3.1 Scientist’s Intent for Result Enrichment

Using the framework presented above it is possible to describe constraints whose purpose is to enrich workflow results. For example, in the Deeside calibration experiment if the real data and the simulation data vary significantly it is interesting to explore why this happens. The constraint below adds a new property (`runToExplore`) to the Simulation instance.

```
PreCondition:
  ParameterSet( ?x1 ) ^
  DataSet( ?x2 ) ^
  Simulation( ?x3 ) ^
  hasSimulationRun( ?x3, ?x4 ) ^
  ComparisonTest( ?x5 ) ^
  compares( ?x5, ?x4 ) ^
  compares( ?x5, ?x2 ) ^
  similarity( ?x5, ?x6 ) ^
  [less-than ( ?x6, 10%) = true]
PostAction:
  runToExplore(?x3, ?x4 )
```

Using this new property, it is possible to explore the simulation data after the workflow has been completed by following the annotations provided by the scientist’s intent, e.g. `runToExplore`. The simulation instance contains a link to the repository containing the relevant simulation metadata. By exploring such metadata the scientist can gain insight into the simulation model status and understand the mechanism(s) which triggered a particular event. For example, this new information about the simulation model can be used to define new constraints that can be used during the validation process. Such constraints will inform the scientist if the events investigated during the calibration process occur during validation.

Another example constraint is presented below:

```
PreCondition:
  SimulationRun( ?x1 ) ^
```

```

hasLandUse( ?x1, ?x2 ) ∧
hasLandParcels( ?x2, ?x3 ) ∧
[more-than ( ?x3 80%) = true]
PostAction:
  isInvalidRun( ?x1 )

```

This specifies that if a specific land use is associated with more than 80% of the land parcels, we can ignore the simulation run.

### 3.2 Scientist’s Intent for Monitoring and Controlling Workflow

Using our framework it is also possible to control the execution of a workflow by specifying a post action from a number of options coded in an ontology, e.g. `stop workflow`, `pause workflow`, etc. Details of the ontology are presented later in this section.

The example constraint below is used to check if the simulation is running on a platform compatible with the IEEE 754 floating point standard:

```

PreCondition:
  GridTask( ?x1 ) ∧
  Simulation( ?x2 ) ∧
  runsSimulation( ?x1, ?x2 ) ∧
  neg runsOnPlatform( ?x1, 'IEEE754' ) ∧
  hasResult( ?x2, ?x3 )
PostCondition:
  hasInvalidResults( ?x2, ?x3 )
  ACTION:resubmitTask(?x1)

```

In this constraint the statement `neg runsOnPlatform( ?x1, 'IEEE754' )` is negation as failure based on the closed world assumption (what is not currently known to be true is false). As a consequence, if there is no information about the platform on which the simulation runs, such a statement is considered to be false.

Actions based on scientist’s intent (e.g. `resubmitTask( ?x1 )`) depend on the ability of the workflow to process events triggered by the scientist’s intent framework. In our case, the extended Kepler Director component is able to understand the above action and therefore re-submits the Grid task.

In the Deeside calibration experiment, a wide range of possible combinations of parameter values are explored. It is interesting here to narrow the parameter space to be searched in order to save computing resources, and to gain understanding of the relative importance, and major interactions, between input parameters. A relatively simple conjunction of requirements for model output values concerning land use and farm size at the end of the case study period

are typically specified, which a run must meet in order to be considered plausible. For example: if in any of the five runs, one land manager owns more than half of the land, ignore this parameter set. The constraint below demonstrates how this can be achieved:

```

PreCondition:
  Simulation( ?x1 ) ∧
  hasSimulationRun( ?x1, x2 ) ∧
  hasLandManager( ?x2, ?x3 ) ∧
  ownsLandParcels( ?x3, ?x4 ) ∧
  [more-than( ?x4, 50% )]
PostAction:
  hasInvalidRun( ?x1, ?x2 ) ∧
  ACTION:stop(?x1)

```

The action `stop(?x1)` stops the entire simulation when one of the runs violates the pre condition.

### 3.3 Scientist’s Intent as Provenance

Earlier, we established that the provenance frameworks associated with existing workflow tools [7] are not sufficient to capture all aspects of the process. In particular, they are insufficient to understand *why* a particular step in the process has been selected. We argue that scientist’s intent can be used to provide the *why* context. For example, to answer *why* an experiment has been conducted we can look at what goal(s) have been defined (e.g. obtain at least one match where the real data falls within 95% confidence interval of the model value).

The scientist’s intent framework introduced in this paper is designed to interoperate with other eScience provenance frameworks (e.g. the provenance framework [7] developed by the PolicyGrid project (<http://www.policygrid.org>)) by providing information about the intent associated with a workflow experiment. In addition, we have attempted to align our scientist’s intent ontology (shown in Figure 4) with the core characteristics of the Open Provenance Model (OPM) [18]. OPM provides a specification to express data provenance, process documentation and data derivation, and is based on three primary entities:

- *Artefact*: an object that has a digital representation in a computer system;
- *Process*: a series of actions performed on artefacts and resulting in new artefacts;
- *Agent*: a contextual entity acting as a catalyst of a process.

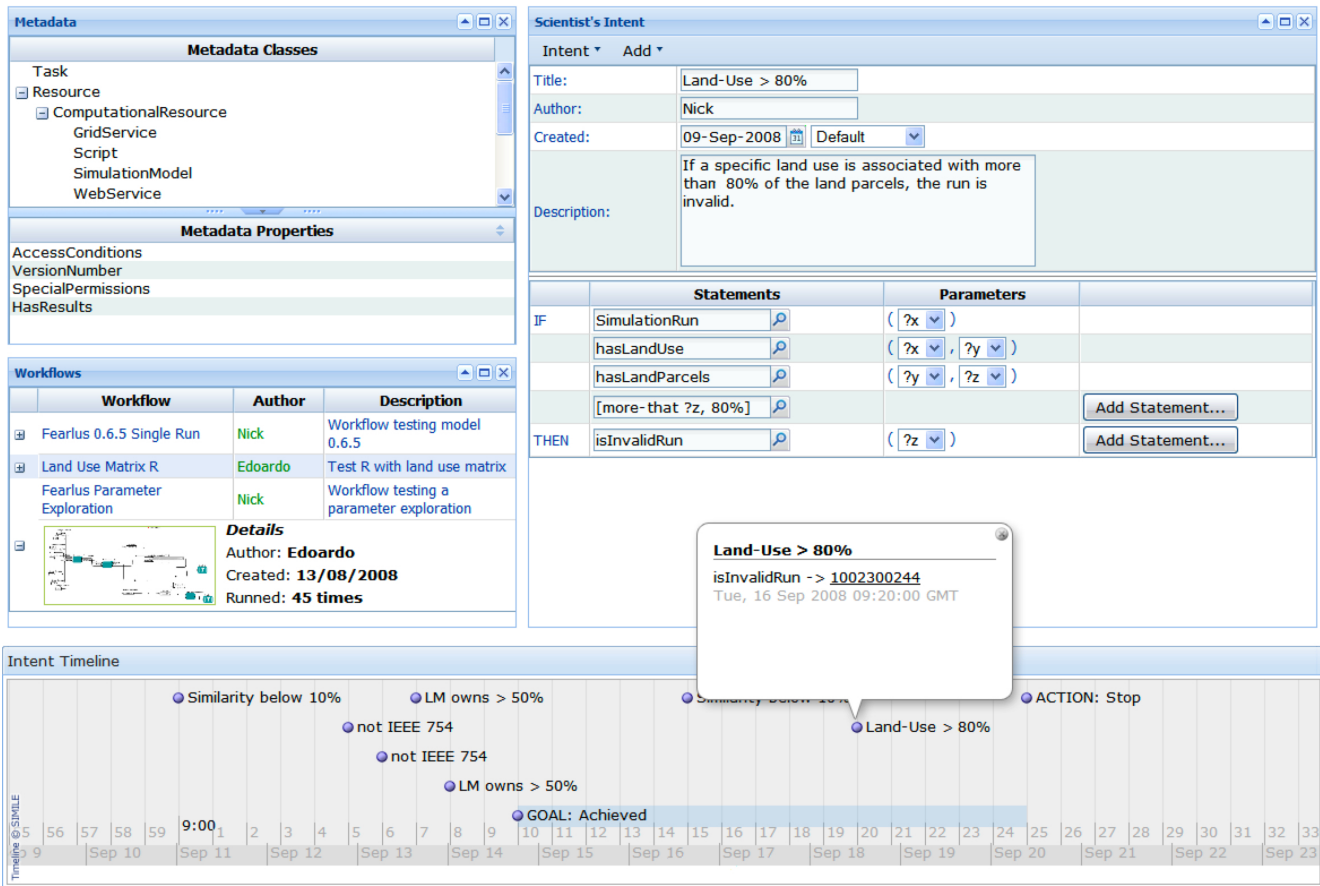


Figure 3. Scientist's Intent Interface.

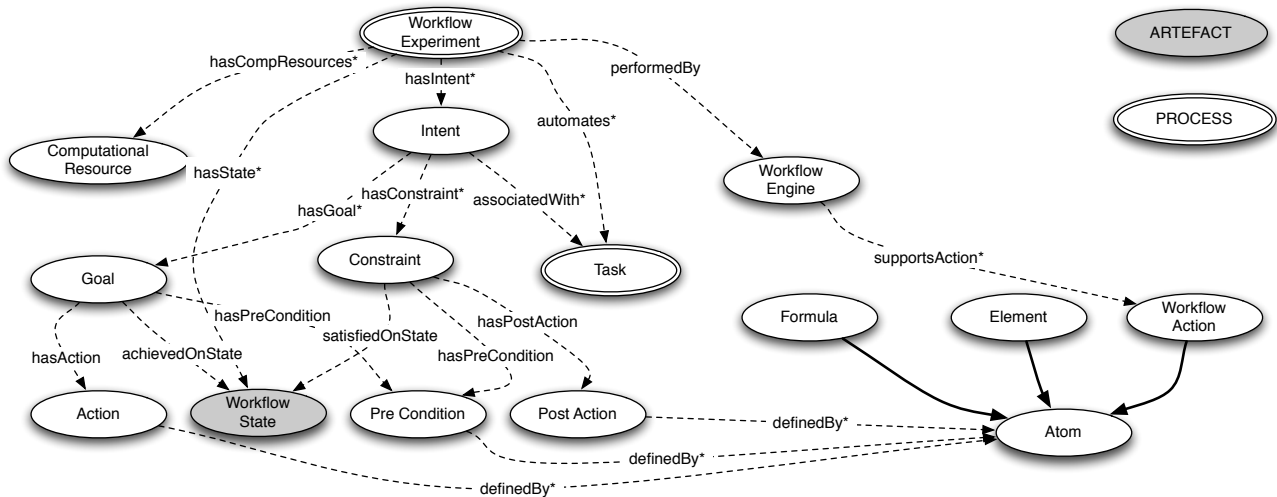
Our hope is that developers of provenance frameworks which implement the OPM specification will find it easy to integrate our scientist's intent solution.

Within our ontology we define the concept of `WorkflowExperiment` as a specific type of process which represents an instance of a workflow used to conduct a scientific experiment. A `WorkflowExperiment` automates one or more tasks (e.g. `DataAnalysisTask`, `DataCollectionTask`). A `WorkflowExperiment` also has associated `ComputationalResources`. The metadata properties associated with `ComputationalResource` instances are stored during the execution of the workflow as a sequence of state transitions. A state transition occurs every time the metadata about the `ComputationalResource` changes. Such transitions are represented as a set of `WorkflowState` instances. A `WorkflowExperiment` is performed by a `WorkflowEngine` which can implement `WorkflowActions`, such as `stop`, `resubmitTask`.

Central to this ontology is the concept of `Intent` which is characterized by a set of `Goals` and `Constraints`. A `WorkflowExperiment` can have zero or more `Intent`

instances. The `Goal` and `Constraint` class share the `preCondition` and `postAction` properties based on their constituent `Atoms`. Such `Atoms` can take the form of a metadata `Element`, a `Formula` or a `WorkflowAction`. In the case of a `Constraint`, when a `PreCondition` is achieved in a specific `WorkflowState` the `PostAction` is triggered. However, in the case of the `Goal` the `PostAction` is only used to inform the user that the goal has been achieved.

We have also implemented a user interface to create and explore scientist's intent. Figure 3 shows a screenshot of the interface where the user is defining one of the constraints presenter earlier. The interface provides the user with metadata classes and properties that can be used as part of a goal or constraint (*Metadata panel*). Such metadata is based on the ontologies used to describe the workflows created by the user (*Workflow panel*). The definition of a goal or constraint is specified by the user by dragging and dropping metadata elements from the metadata panel to text boxes forming the rule statement. Built-in functions (e.g. `more-than`, `less-than`) can also be selected from the dropdown menus associated with the rule statements. Work-



**Figure 4. Scientist's Intent Ontology.**

flows on the workflow panel can be associated with one or more intent definitions and executed. The metadata generated from the scientist's intent during the execution of a workflow is presented back to the user in the form of a timeline using a web-based timeline widget<sup>7</sup>. The timeline widget presents annotations derived from scientist's intent when they occurred during the execution of the workflow.

## 4. Conclusions

In this paper we have discussed how current workflow technologies have limited capabilities for capturing the experimental conditions associated with a workflow. We have presented a scientist's intent framework based on rules and workflow metadata to capture goals and constraints associated with a workflow experiment. We discussed the benefits of using our framework in terms of enriching workflow results, controlling and monitoring workflow execution, and enriching workflow provenance.

We are currently evaluating our framework with the help of a number of case-studies from different disciplines. User scientists are central to the evaluation process as they are using the tools we have developed to design and perform real experiments. Our evaluation process consists of two stages: In the first stage we ask the scientist to design an experiment using the Kepler tool, and to supply feedback on this process via questionnaires, interviews and through direct observation. During the second stage, the subject is asked to design and apply scientist's intent rules based on the experiment from the first stage.

The following are some of the questions that we put to

participants during the first phase of the evaluation:

- Can you describe the goals and sub-goals of the experiment in detail?
- Can you describe any constraint that applies to the experiment or part of it?
- Were you able to capture all the above information using the Kepler tool? If not, could you provide details of the cases where you had to compromise?
- Could you have saved time or computational resources by adding fine-grained controls on the workflows? If so, could you give details and some examples?
- Using the Kepler tool, does the workflow generated provide enough documentation about the methodology used in the experiment? If not, what kind of information is missing?
- Did you find it easy to use previous generated experiments as a base for creating new experiments?

The subjects who participated spanned several different disciplines: Land-Use Simulation, Computational Data Based Modelling, Urban Simulation, Health and Social Policy, Grid Application Development. Each of them described a typical experiment involving data and computational resources available to them. They were then asked to describe the experiment using a workflow formalism (boxes for activities, arrows for data pipelines) and we then conducted a recorded interview. All subjects agreed that workflow technologies could facilitate the execution of their experiments. However, some limitations of using workflow technologies were identified during the interview process:

<sup>7</sup><http://simile.mit.edu/timeline/>



- It is not possible to represent constraints regarding data aggregation;
- Contextual information about the experiment is missing;
- At the moment, detailed technical documentation is needed in order to fully understand a workflow.

The second stage of our evaluation is now underway. The following are the key criteria to be used:

- **Expressiveness of the intent formalism:** Is the formalism sufficient to capture real examples of intent? Were certain constraints impossible to express? Were some constraints difficult to express?
- **Reusability:** Can an intent definition be reused - either in its entirety or in fragments? Does our framework facilitate reusability?
- **Workflow execution:** Does the inclusion of intent information affect the computational resources required during the execution of a workflow? (This type of evaluation will be carried out in simulated conditions by monitoring the Grid resources required to execute example workflows with and without scientist's intent support.)

From a user perspective, creating and utilizing metadata is a non-trivial task; the use of a rule language to capture scientist's intent does of course provide additional challenges in this regard. We have addressed these issues by creating a web-based tool to compose scientist's intent rules from available metadata, to associate workflow with intent and to visualize intent information. Although we are using a timeline widget to present the intent information back to the user, there are still challenges associated with metadata browsing. Hielkema et al. [13] describe a tool which provides access to RDF metadata (create, browse and query) using natural language. The tool can operate with different underlying ontologies, and we are exploring whether it could be extended to explore scientist's intent metadata.

As described earlier in this paper, our framework allows the user to define goals associated with a workflow experiment. Our framework makes a limited use of goals by annotating workflow results so that when a goal has been achieved the event is recorded and displayed as part of the workflow results. This limitation is due to the fact that current workflow engines are not designed to reason about goals when planning the execution of the workflow. In the WSMO ontology, goals are defined as the objectives that a client may have when consulting a service. Such definition can be used to identify the services required to achieve a specific goal. Our definition of a goal could potentially

be utilised with an implementation of WISMO (such as WSMX [28]) to overcome the limitation of current workflow execution engines.

In conclusion, we aim to provide a closer connection between experimental workflows and the goals and constraints of the researcher, thus making experiments more transparent. While scientist's intent provides additional meta-data information for workflow results and provenance, its use should also facilitate improved management of workflow execution. In addition, scientist's intent provides more provenance information about the *why* context. However, much more work is needed if we are to truly capture the intent of the scientist; the framework described here is an important step towards that ultimate goal.

## References

- [1] L. N. Alessa, M. Laituri, and M. Barton. An "all hands" call to the social science community: Establishing a community framework for complexity modeling using agent based models and cyberinfrastructure. *Journal of Artificial Societies and Social Simulation*, 9 (4) 6, 2006.
- [2] T. Andrews. Business process execution for web services, version 1.1. <ftp://www6.software.ibm.com/software/developer/library/ws-bpel.pdf>, 2003.
- [3] R. L. Axtell. Why agents? on the varied motivations for agents in the social sciences. In e. Macah C M, Sallach D, editor, *Proceedings of the Workshop on Agent Simulation: Applications, Models, and Tools. Argonne, Illinois, Argonne National Laboratory*, 2000.
- [4] M. Babik, L. Hluchy, J. Kitowski, and B. Kryza. Generating semantic descriptions of web and grid services. In *Sixth Austrian-Hungarian Workshop on Distributed and Parallel Systems*, 2006.
- [5] C. Berkley, S. Bowers, M. B. Jones, B. Ludäscher, M. Schildhauer, and J. Tao. Incorporating semantics in scientific workflow authoring. In *Proceedings of the 17th International Conference on Scientific and Statistical Database Management (SSDBM'05)*, 2005.
- [6] C. Bussler, L. Cabral, J. Dominigue, and M. Moran. Towards a semantic grid service operating system. In *Proceedings of the Workshop on Network Centric Operating Systems Brussels, Belgium*, 2005.
- [7] A. Chorley, P. Edwards, A. Preece, and J. Farrington. Tools for tracing evidence in social science. In *Proceedings of the Third International Conference on eSocial Science*, 2007.
- [8] I. Foster, C. Kesselman, J. Nick, and S. Tuecke. *Grid services for distributed system integration*. Morgan- Kaufmann, Jan 2002.
- [9] K. Frenken. History, state and prospects of evolutionary models of technical change: a review with special emphasis on complexity theory. Utrecht University, The Netherlands, mimeo, 2005.
- [10] C. Goble. Position statement: Musings on provenance, workflow and (semantic web) annotation for bioinformat-

- ics. Workshop on Data Derivation and Provenance, Chicago, 2002.
- [11] M. Greenwood, C. Goble, R. Stevens, J. Zhao, M. Addis, D. Marvin, L. Moreau, and T. Oinn. Provenance of e-science experiments. In *Proceedings of the UK OST e-Science 2nd AHM*, 2003.
- [12] P. Groth, S. Jiang, S. Miles, S. Munroe, V. Tan, S. Tsasakou, and L. Moreau. An architecture for provenance systems. ECS, University of Southampton, 2006.
- [13] F. Hielkema, P. Edwards, C. Mellish, and J. Farrington. A flexible interface to community-driven metadata. In *Proceedings of the eSocial Science conference 2007, Ann Arbor, Michigan*, 2007.
- [14] A. Lee and S. Neuendorffer. Moml — a modeling markup language in xml —version 0.4. Technical report, University of California at Berkeley, 2000.
- [15] B. Ludäscher, I. Altintas, C. Berkley, D. Higgins, E. Jeager, M. Jones, E. Lee, and J. Tao. Scientific workflow management and the kepler system. *Concurrency and Computation: Practice and Experience*, pages 1039–1065, 2006.
- [16] D. Martin, M. Burstein, J. Hobbsa, O. Lassila, D. McDermott, S. McIlraith, S. Narayanan, M. Paolucci, B. Parsia, T. Payne, E. Sirin, N. Srinivasan, and K. Sycara. Owl-s: Semantic markup for web services. <http://www.w3.org/Submission/OWL-S>, 2004.
- [17] W. Michener, J. Beach, S. Bowers, L. Downey, M. Jones, B. Ludaescher, D. Pennington, A. Rajasekar, S. Romanello, M. Schildhauer, D. Vieglais, and J. Zhang. Seek: Data integration and workflow solutions for ecology. In *Workshop on Data Integration in the Life Sciences (DILS'2005), LNCS*, volume 3615, pages 321–324, 2005.
- [18] L. Moreau, J. Freire, J. Futrelle, R. McGrath, J. Myers, and P. Paulson. The open provenance model. Technical report, University of Southampton, 2007.
- [19] S. Moss. Relevance, realism and rigour: A third way for social and economic research. Technical report, CPM report, 1999.
- [20] M. Oinn, M. Greenwood, M. Addis, M. N. Alpdemir, J. Ferris, K. Glover, C. Goble, A. Goderis, D. Hull, D. Marvin, P. Li, M. Pocock, M. Senger, R. Stevens, A. Wipat, and C. Wroe. Taverna: lessons in creating a workflow environment for the life sciences. *Concurrency and Computation: Practice and Experience*, pages 1067–1100, 2006.
- [21] T. Oinn, M. Addis, J. Ferris, D. Marvin, M. Senger, M. Greenwood, T. Carver, K. Glover, M. Pocock, A. Wipat, and P. Li. Taverna: a tool for the composition and enactment of bioinformatics workflows. *Bioinformatics Journal*, 20(17):3045–3054, Jan 2004.
- [22] D. Pennington. Supporting large-scale science with workflows. In *Proceedings of the 2nd Workshop on Workflows in support of large-scale science, High Performance Distributed Computing*, Jan 2007.
- [23] E. Pignotti, P. Edwards, A. Preece, G. Polhill, and N. Gotts. Enhancing workflow descriptions with a semantic descriptions of scientific intent. In *Fifth European Semantic Web Conference, Springer-Verlag, Lecture Notes in Computer Science*, volume 5021, pages 644–658, 2008.
- [24] J. G. Polhill and N. M. Gotts. Evaluating a prototype self-description feature in an agent-based model of land use change. In *Proceedings of the Fourth Conference of the European Social Simulation Association, September 10-14, 2007, Toulouse, France, F. Amblard*, pages 711–718, 2007.
- [25] J. G. Polhill, N. M. Gotts, and A. N. R. Law. Imitative versus nonimitative strategies in a land use simulation. *Cybernetics and Systems*, 32(1-2):285–307, 2001.
- [26] D. Roman, U. Keller, H. Lausen, J. Bruijn, R. Lara, M. Stollberg, A. Polleres, C. Feier, C. Bussler, and D. Fensel. Web service modeling ontology. In *Applied Ontology, 1(1)*, pages 77 – 106, 2005.
- [27] D. D. Roure, N. Jennings, and N. Shadbolt. The semantic grid: a future e-science infrastructure. *Grid Computing: Making the Global Infrastructure a Reality*, Jan 2003.
- [28] T. Vitvar, A. Mocan, M. Kerrigan, M. Zaremba, M. Zaremba, M. Moran, E. Cimpian, T. Haselwanter, and D. Fensel. Semantically-enabled service oriented architecture: Concepts, technology and application. *Journal of Service Oriented Computing and Applications, Springer London*, 2007.
- [29] J. A. Zachman. A framework for information systems architecture. *IBM Syst. J.*, 26(3):276–292, 1987.