Proceedings of the 2010 Winter Simulation Conference
B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Yücesan, eds.

USING WORKFLOWS IN M&S SOFTWARE

Stefan Rybacki
Jan Himmelspach
Enrico Seib
Adelinde M. Uhrmacher

University of Rostock
Albert-Einstein-Str. 21
18059 Rostock, GERMANY

ABSTRACT

The usage of workflows to standardize processes, as well as to increase their efficiency and the quality of the results is a common technique. So far it has only been rarely applied in modeling and simulation. Herein we argue for employing this technique for the creation of various products in modeling and simulation. This includes the creation of models, simulations, modeling languages, and modeling and simulation software modules. Additionally we argue why roles should be incorporated into modeling and simulation workflows, provide a list of requirements for the workflow management system and sketch first steps in how to integrate workflows into the M&S framework JAMES II.

1 INTRODUCTION

M&S applications may incorporate diverse products. These products may comprise models (or model components), simulations (experiments with a model), (modeling) languages, or M&S software. The importance of following well defined processes to achieve these diverse products is not new to M&S. A number of *M&S life cycle models* have been proposed to help developing and interpreting validated and verified models (Balci 1990, Sargent et al. 2006, Law 2007). However, the quality of simulation studies does not only depend on validated and verified models but also on the underlying validated modeling and simulation methods (Himmelspach and Uhrmacher 2009b), which in turn might require extensive simulation studies, e.g., to experimentally validate the accuracy or superiority of a simulation method in comparison to others (Jeschke and Ewald 2008). Thus, to the final product, e.g., simulation results that confirm a hypothesis about the dynamics of a given system, diverse M&S "workflows" involving different users have contributed. Also in the light that currently, simulation faces a credibility crises (Pawlikowski et al. 2002, Pawlikowski 2003, Kurkowski 2006), the desire emerges for more documentation of and guidance through these processes.

Workflow systems are aimed at providing software support for documenting, executing, and controlling workflows. They facilitate determining and eliminating problems in a production process that might induce errors in the final product (van der Aalst and van Hee 2004). Well defined workflows are considered to be mandatory for a repeatable production process (ISO 9001:2008 2008); repeatability being a salient feature of any experimental science.

Thus, integrating workflow management systems in M&S processes would increase the quality of products and strengthen credibility. In concrete, possible benefits would be improved repeatability, means to track "product" creation stepwise, intuitive task description, auto documentation and progress information, and a far reaching automation of M&S processes.

In the following we will give a short overview on workflows in science and M&S, discuss two use cases to illuminate the need for flexible workflow support, introduce different roles according to different types of M&S users, use these use cases, roles, and a set of established requirements for workflow management systems to define requirements for M&S workflow systems, and finally, identify a few crucial steps toward realizing such a workflow system, illuminated by the example JAMES II.

2 WORKFLOWS FOR SCIENTIFIC PROCESSES AND M&S

Workflows are recently receiving more and more attention in the scientific domain. So it is not surprising that more and more workflow systems emerge that support scientific processes. Project Trident (Barga et al. 2008, Barga et al. 2008),

978-1-4244-9864-2/10/$26.00 ©2010 IEEE

Taverna (Hull et al. 2006, Oinn et al. 2006), Kepler (Altintas et al. 2004, Ludäscher et al. 2006) to name just a few of such systems.

Depending on the specificity of the systems, different approaches are pursued in managing workflows. General systems as Project Trident, Taverna and Kepler allow the scientists to create their own workflows or to edit, enhance and reuse existing ones. Systems are developed to publish, share, store, and query scientific workflows and experiments, e.g., the so called myExperiment project (Roure et al. 2008) provides a repository for sharing research objects used by scientists, such as scientific workflows. It is also supported by Taverna (Roure and Goble 2009) and Project Trident.

While those systems are aimed to provide support for general scientific workflows including as in the case of Kepler support for specific M&S aspects by building on PTOLEMY II (Lee and Neuendorffer 2007, UC Berkley EECS Dept. 2010), other systems such as SYCAMORE (Weidemann et al. 2008), SWAN Tools (Perrone et al. 2008) and JAMES II (Himmelspach et al. 2008, Himmelspach and Uhrmacher 2009a) are dedicated toward supporting M&S processes.

JAMES II, SYCAMORE and SWAN Tools hide the workflow aspect by only supporting fixed predefined workflows that can be executed by the scientists to assist them.

JAMES II, e.g., provides predefined workflows to create a basic simulation experiment by guiding the scientist through the steps of selecting a formalism, creating, loading or editing a model, parameterizing the model, selecting a simulation algorithm and a visualization. This results in an experiment description that can be executed afterwards. Another predefined workflow in JAMES II is aimed at supporting the experimental validation of models. It refines the basic experiment workflow integrating additional steps to select the type of experiment (e.g., Optimization Experiment, Parameter Scan, Sensitivity Analysis, etc.) to be performed (Leye, Himmelspach, and Uhrmacher 2009).

SYCAMORE is a web based front end supporting, e.g., COPASI (Hoops et al. 2006) as simulation engine, different online resources like databases, and locally available tools. SYCAMORE provides a fixed workflow. It guides through the process of setting up a model by selecting kinetic data from a connected database, adjusting parameters, model checking, parameter estimation, sensitivity analysis, and simulation execution using COPASI. However, experiences with SYCAMORE have revealed that whereas novices appreciate this strict guidance more experienced users found the solution too restrictive and wished for workflows that balance guidance and sufficient flexibility for adaptations.

The SWAN Tools provide a web based framework for the automation of the entire simulation workflow with SWAN (Liu et al. 2001). It assists and guides the scientist when configuring models with parameters by using an information rich interface that should enhance the understanding of what each parameter does. It also helps the scientist to manage and create simulation experiments and their configuration by letting the user define data for simulation runs and also to select or provide a specific simulator. Additionally simulation runs are generated for the scientist and can automatically be distributed in, e.g., a cluster. The simulation results are provided via a database interface as well as using different visualizations.

3 USE CASES

This section focuses on presenting two potential use cases for workflows in the M&S systems like JAMES II. They form the basis to define requirements that are essential in a workflow system intended to provide support for applying and evaluating M&S methods. The former refers to typical simulation studies, in this case we focus on the phase of creating a model, the latter deals with the problem how simulation algorithms can be evaluated.

3.1 Creating a Model for Simulation

Usually process models for M&S found in literature (e.g., Sargent 2008, Balci 2004, Law 2007) are described at a rather abstract level and just contain general task descriptions like "create qualitative model". This coarse grained description leaves a high degree of freedom for interpretation on how to execute each step but results in a hardly traceable process execution which makes it difficult to find potential sources of errors or to ensure specific requirements and therefore a specific quality. Thus, these steps need to be refined for realizing a suitable support via workflows.

The workflow in Figure 1 is based on the process model for model creation introduced in (Balci 1990). The process of creating a simulation model can be described by the following steps:

System Investigation The *System Investigation* deals with the investigation of system properties used for system definition and modeling. It also defines the simulation study's *Objectives*, i.e., the questions to be answered. System properties are identified that are of relevance for answering those questions. Typically as part of *System Investigation* data are collected and relevant input parameters are identified.

Model Formulation During the *Model Formulation* step the system under study is described as conceptual model taking the previously identified *Objectives* and the resulting requirements into account.

Model Representation The *Model Representation* turns the previously formulated conceptual model into a communicative model. The communicative model allows other parties (Researchers, Project Lead, etc.) to comment and to discuss the model.

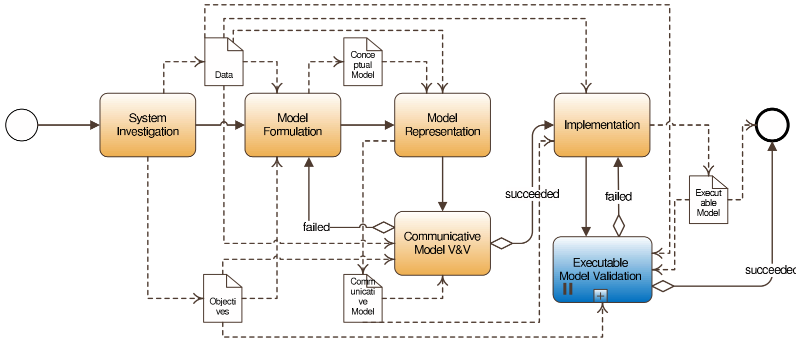


Figure 1: Workflow description for the creation of a model

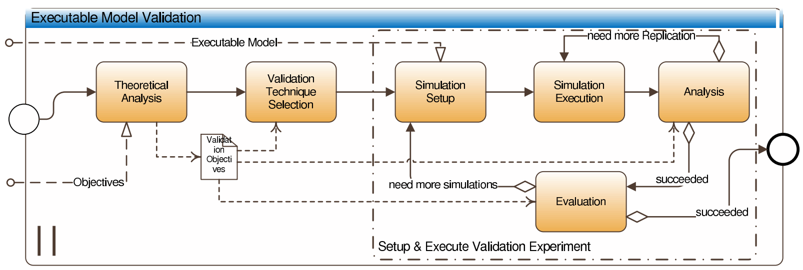


Figure 2: Sub Workflow description for validation of an executable model

Communicative Model V&V Before implementing the communicative model it should be verified and validated during the *Communicative Model V&V* step. This step verifies and validates the created communicative model. If the model does not comply to the objectives and requirements the *Model Formulation* step needs to be redone.

Implementation After verifying and validating the previously created communicative model the *Implementation* step turns the communicative model into an executable model. This can either be achieved by describing the model in a simulation specific or a high level programming language.

Executable Model Validation As the communicative model does not require a description in a formalism with a clear semantics, *Communicative Model V&V* does not necessarily involve the application of formal methods, this is different when looking at the phase of *Executable Model Validation*. The executable model can be validated using different techniques, e.g., sensitivity analysis, parameter scan or optimization techniques. Thereby, also requirements can be formally specified and formally verified against the behavior of the model. If the executable model does not validate according to the *Objectives* and techniques used, it needs to be reimplemented, hence the *Implementation* step needs to be revisited. A variety of validation techniques can be applied to the model for validity checking before continuing with the simulation study.

One of the key products of the shown workflow are the *Objectives* which describe the questions the simulation study intends to answer and thus the motivation behind developing the model. They also determine the requirements for validating and verifying the model (Leye and Uhrmacher 2010).

To execute such a workflow some of the steps shown need to be specified further. As an example Figure 2 shows the *Executable Model Validation* step sub divided into more specific tasks describing how validation could be described further as sub workflow.

Simulations can be used to explore the validity of models. Sensitivity analysis can be used to determine how sensitive models react on parameter changes, other experiments can be used in the spirit of units tests to find out whether a model will behave as expected given a set of inputs. Such experiments can only be made if a part of the model which can be executed standalone is sufficiently completed, and thus this type of validation can only happen at a late stage of the development process. Such a stage is the *Executable Model Validation* and described in more detail in the following.

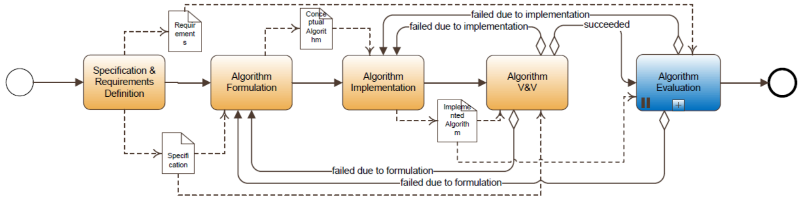


Figure 3: Workflow description for the evaluation of an algorithm

A possible executable model validation workflow in consensus with (Leye and Uhrmacher 2010) might look as shown in Figure 2 and consists of the following steps:

Theoretical Analysis The *Validation Objectives* are defined during the *Theoretical Analysis* step and are derived from the *Objectives* of the model creation.
Validation Technique Selection After defining *Validation Objectives* an appropriate validation technique according to them is selected in the *Validation Technique Selection* step. Such techniques could be, e.g., sensitivity analysis, face validation or even simulation based model checking (Rizk et al. 2008, Kemper 2009).
Simulation Setup Following the *Validation Technique Selection* the *Simulation Setup* takes place and is used to setup a simulation run using the selected technique as well as the model to validate. It is also part of the *Setup & Execute Validation Experiment* group.
Simulation Execution As part of the *Setup & Execute Validation Experiment* group the *Simulation Execution* step executes the previously set up simulation.
Analysis The *Analysis* performs analysis on the simulation results, e.g., data aggregation, statistical analysis and Monte-Carlo variability. An analysis might need further replications to perform correct analysis in which case the simulation is executed again until a sufficient amount of replications are available.
Evaluation The last step is the *Evaluation* step which performs evaluation tasks on the analysis results while the task involved depends on the selected validation technique. An evaluation might need more analysed data in which case it invokes another set of simulation runs by reentering the workflow at *Simulation Setup*. This is done until the evaluation task has sufficient analysed data available so it can perform the actual evaluation on that data. Eventually the evaluation returns a *succeed* or *failed* according to the previously defined objectives.

The *Setup & Execute Validation Experiment* group could be represented as a step in a coarser workflow rather than a group and would be defined as the shown sub workflow consisting of *Simulation Setup*, *Simulation Execution*, *Analysis* and *Evaluation*.

3.2 Experimental Algorithmics

To achieve a certain level of quality and credibility in M&S it is crucial to not only use validated and verified models for simulation studies but also validated and correct simulation algorithm.

Experimental algorithmics is concerned with the experimental evaluation of algorithms. Newly developed algorithms are evaluated in comparison to alternatives, usually with the purpose of demonstrating that the newly developed algorithm is superior. In the realm of algorithms for modeling and simulation an experimental evaluation implies the execution of simulations and thus, a simulation study aimed not at creating a (valid) model (as in the previous use-case) but at evaluating (new) algorithms. The software development cycle of M&S algorithms, involving the implementation of algorithms, their validation, and their evaluation can be supported by workflows.

Figure 3 shows the workflow dedicated to designing and evaluating simulation algorithms, or modules used such as event queues and random number generators. The workflow comprises the following steps.

Algorithm Formulation The *Algorithm Formulation* is used to specify an algorithm that can later on be implemented.
Implementation This specification of the algorithm is implemented in the *Implementation* step.
Algorithm V&V After being implemented the algorithm needs to be verified and validated. To show the correctness of the implementation the implemented algorithm can be verified against its specification, or the results of sample simulations using the newly implemented algorithm can be compared with results of another already validated and verified algorithm. Most often a failure will be due to an implementation problem which implies redoing the *Implementation* step. However, the comparison of the verified algorithm to a verified and validated algorithm might also reveal problems in the specification which implies revising the *Algorithm Formulation* step.

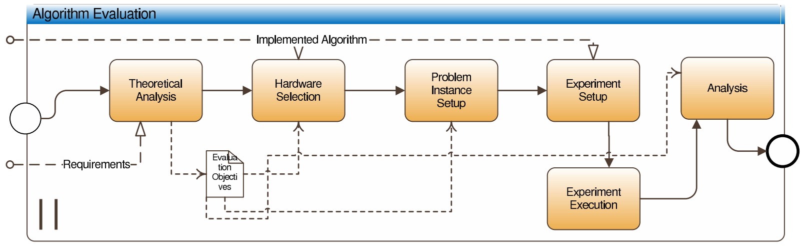


Figure 4: Sub Workflow description for the evaluation of an algorithm

Algorithm Evaluation Having a verified and validated algorithm it can be evaluated in the *Algorithm Evaluation* step to determine whether the newly algorithm fulfills specific requirements, such as the algorithm should outperform another algorithm. In case some of the requirements can't be fulfilled it might be necessary to refine the specification of the algorithm and go back to *Algorithm Formulation* or to change the implementation by going back to the *Implementation* step.

As for the use case in 3.1 the shown workflow is rather high level and needs to be defined further. An example for such a refinement is the sub workflow shown in Figure 4. Here the evaluation step is refined and describes the activities taken during the *Algorithm Evaluation* step in more detail:

Theoretical Analysis The *Theoretical Analysis* is used to define objectives and goals for the evaluation process which also take the requirements into account.

Hardware Selection A basic step the *Hardware Selection* which selects a platform the algorithm is able to run and which the actual evaluation is performed on.

Problem Instance Setup After selecting the target the *Problem Instance Setup* step defines a problem instance which takes objectives and goals of the *Theoretical Analysis* into account and is used later on in the evaluation experiment.

Experiment Setup The *Experiment Setup* creates an experiment on basis of the previously defined problem instance and selected platform.

Experiment Execution This step executes the evaluation experiment.

Analysis The *Analysis* step analysis results from the evaluation experiment which are used to determine whether objectives were fulfilled or not and therefore specify further steps in the main workflow.

4 REQUIREMENTS FOR M&S WORKFLOWS

An effective support of workflows in M&S has not only to take different scenarios, e.g., our two different use-cases, but also different types of users and their particular abilities and interests into account. Roles are usually representing types of potential users of a system, and assign specific responsibilities to them. As suggested by Balci in (Balci 1990) experts are needed to carry out special tasks, e.g., modeling should be done by modeling experts while validation should be done by validation experts and so on. Having the support of roles will improve overall product quality and credibility.

In the following we identify a set of roles that are important for an M&S system. These roles are not necessarily disjunct, e.g., a M&S researcher might wish to evaluate the performance of a newly developed simulation algorithm. Therefore, he will execute simulations and thereby take on the role of the simulation expert. Similarly, for designing benchmark models to test the expressiveness of a newly developed language, the role of a modeling expert might be assumed.

Role 1: the *modeling expert* The modeling expert's job is to create models based on data, a hypothesis, known facts or any combination thereof. Ideally, a scientific workflow system should ensure the quality of the data and facts that enter this process, e.g., (Hull et al. 2006). A workflow will direct the *modeling expert* through the development process. The interest lies in getting

- a correct model, that has been verified,
- the right model, that has been validated,
- a report, that documents the process of model generation, to facilitate assessing its quality and to increase its credibility.

Role 2: the *simulation expert* Simulation experts create simulations based on the right verified and validated models to prove the hypotheses of interest. The interest lies in getting

- the experiment definition, that allows to answer the hypothesis of interest,
- a correct and efficient execution of experiments,
- support in representation, exploration, and interpretation of simulation results,
- a report, that combines experiment definition, technical details of execution, and results, to facilitate reproducibility.

Role 3: the *M&S methodology researcher* There are different types of M&S researchers. Researchers in M&S can be interested in either modeling methods or computational methods to compute simulations. The latter group comprises researchers interested in any computation technique (i.e., simulation algorithms and accompanying data structures (e.g., event queues), specialized algorithms for getting parameter combinations (optimization, sensitivity analysis, etc.), in generating random numbers, a.s.o.).
Whatever the research interest is, a M&S researchers interest in workflows are in getting

- support in verification and validation,
- support in evaluation processes,
- and a documentation of the development efforts.

Role 4: the *V&V expert* The V&V expert cooperates with either the *M&S methodology researcher* or the *modeling expert* and assists in the validation and verification phases of modeling or simulation software module creation. His interests are in

- accessing all specifications and reports,
- support in V&V cycle,
- and in a documentation of the V&V steps undertaken.

Role 5: the *workflow designer* A workflow designer is interested in the ability to define well defined workflows meeting the requirement of the different, aforementioned, roles. Based on an sufficiently expressive workflow language, a sound execution semantics of workflows, and a workflow repository
In particular he is interested in support for

- designing, revising, sharing, storing, and retrieving workflows for diverse tasks and roles,
- workflow verification, and for
- workflow statistics.

Role 6: the *project manager* The project manager's job is to assign available resources to tasks, and if he has to organize more than one project to come up with as few critical resource usages as possible. In addition he as to control the overall progress of the projects at hand. Workflows can assist a *project manager* in his job as he can use to control and observe the project progress. His interests lies in getting

- an overview of resource usage,
- an overview of the projects progress,
- well defined and easy to control projects, and in
- means to document the projects for the client.

Role 7: the *client* A client is usually interested in getting a good product, in time, for his money. If a dedicated workflow management is used for executing the project he can get

- status information about the project,
- information about general steps being executed in the overall process (e.g., V&V was included),
- and last but not least such workflows can support the fulfillment of his requirements.

Role 8: the *system administrator* The system administrator's job is to provide a working infrastructure able to execute workflows created by the *workflow designer*. His interests are to be able

- to intervene with workflow execution in case of administrative tasks or system errors, and
- to monitor running workflows and their usage of system resources.

Features of a workflow system and their match to roles

In Table 1 requirements for scientific workflows as proposed by (Ludäscher et al. 2006, Gil et al. 2007) are shown and described according to the M&S workflow and the roles we identified above.

Table 1: Requirements to workflow system in M&S

Requirement	**Role**	**Intention**
Seamless access to resources and services	Workflow Designer / Administrator	supports the transparent use of resources and services
	Modeling expert	seamless access to model repositories, and data sources
	Simulation expert	seamless access to computing services
	V&V expert	seamless access to data sources, computing sources, benchmark models
	Methodology researcher	seamless access to computing sources, benchmark models, method repository
Service composition & reuse and workflow design	Workflow Designer	helps to support the designing parts of the workflow life cycle (build, refine, share)
	Modeling expert	reuse of modeling workflows for similar situations, reuse of models, composition of models
	Simulation expert	reuse of earlier simulation studies
Smart semantic links	Workflow Designer	assists the workflow designer by identifying or creating compatible links between activities
Scalability	Workflow Designer / Administrator	support for arbitrary sized workflows and data
	Methodology researcher	support for different hardware or hardware combinations
	Modeling expert	support for large models and data
	Simulation expert / V&V expert	support for large data, models and intensive computational tasks
User-interaction	Workflow Designer	support for designing workflows incorporating user decisions
	Methodology researcher / Modeling expert / Simulation expert / V&V expert	helps to interact with the system and to steer the workflow
Detached execution	Project Manager	allows for long running tasks and workflow execution on dedicated machines
	Methodology researcher / Simulation expert / V&V expert	detached experiment execution
	Modeling expert	detached validation execution
Reliability and fault-tolerance	Methodology researcher / Modeling expert / Simulation expert / V&V expert	avoid faulty software modules, restrict for validated and verified models and algorithms, recover after error
	Client / Project Manager	error recovery and proper execution strategies
Data provenance	Client / Project Manager	explicit data documentation throughout entire workflow, results in repeatability
	Methodology researcher	gains insight in execution provisional and end results for debugging or automated/smart reruns and repeatability
	Modeling expert	ensures model quality
	Simulation expert	ensures simulation credibility and allows debugging or automated/smart reruns and repeatability
	V&V expert	ensures V&V quality
Smart reruns	Client / Project Manager / Methodology researcher / Modeling expert / Simulation expert / V&V expert	enables rerunning workflows only in parts due to faulty software modules, repeated experiment, on different configurations, changed workflow, changed workflow task, task parameters etc.
Workflow monitoring & control	Administrator / Project Manager	support for Progress information, Stopping, Resuming, Pausing, Scheduling workflows etc.
Workflow Roles	Project Manager / Workflow Designer	defines responsibilities throughout the workflow, workflow system and provides expert support where needed
	Modeling expert	ensures V&V steps are executed by experts
	Methodology researcher / Simulation expert	ensures that V&V and modeling steps are executed by experts

Table 1: Requirements to workflow system in M&S

Requirement	**Role**	**Intention**
	Client	ensures that experts executed certain steps
Collaborations & Delegation	Workflow Designer / Project Manager	provides connections to different services, teams, tools or experts
	Modeling expert	delegates V&V tasks or uses modeling knowledge by other teams
	Methodology researcher / Simulation expert	delegates V&V and modeling tasks
	V&V expert	working with special tools, teams or services to provide V&V techniques
	Client	outsource tasks or incorporate additional experts and knowledge
System Stability	Project Manager	Compensates environmental changes at system level to ensure, e.g., repeatability
	Methodology researcher / Modeling expert / Simulation expert / V&V expert	repeating experiments, evaluations or calculations even on changed environment
	Client	higher credibility of results
Data Security	Administrator	controls access to parts of the workflow system, who can edit, run, share etc. a workflow
	Methodology researcher / Modeling expert / Simulation expert	getting non manipulated results in between steps and control access to results
	V&V expert / Client / Project Manager	reliable non altered results and prevent unauthorized access

5 SKETCHING REALIZATION REQUIREMENTS FOR JAMES II

The following section sketches the first steps on integrating workflows in the M&S system JAMES II.

Experiments are executed in the experimentation layer of JAMES II. Experiments once defined can run in a mostly automated way and are therefore predestined to be the first item to be replaced or enhanced using workflows. This doesn't mean that the overall realization will be limited to that very part of JAMES II but will evolve over time and find its way into more parts of JAMES II to be able to eventually fully support presented use cases as well as even more complex ones.

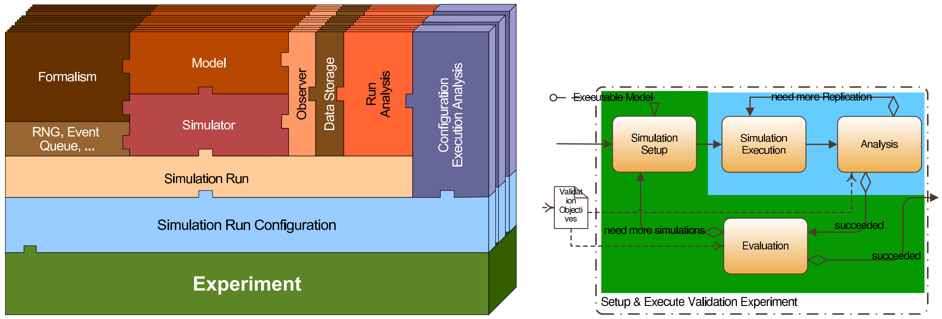
Currently the experimental layer is hard coded. Its flexibility stems from a general view on experiments as source of parameter combinations and its extensive usage of the plug-in system which allows to exchange parts of the functionality. Thereby any type of experiment, using sequential or parallel execution strategies, using different data sinks, automatically generated combinations of algorithms to compute the jobs at hand can be executed. Figure 5(a) shows the items of and their interaction within the experimentation layer of JAMES II.

Drawbacks of the current experimentation layer are:

- the experimentation layer starts with an already defined model
- the experimentation layer is flexible in a way that some of its parts are interchangeable (simulation algorithm, etc.), it follows the strategy pattern (Gamma et al. 1995) which basically provides an experiment skeleton which is filled with, e.g., modules or plug-ins that fit but the skeleton itself resp. the order and amount of plug in places and their execution strategy is fixed
- incorporating changes or adding functionality is rather tedious and error prone
- complete and sufficient documentation is hard to achieve
- role incorporation not intended

Parts that define experiment execution behaviour are the *Experiment* s and the *Simulation Run Configuration* . For the V&V experiment shown in Figure 2 step *Simulation Execution* and *Analysis* are encapsulated by the *Simulation Run Configuration* , whereas *Evaluation* and *Simulation Setup* are encapsulated by an *Experiment* (see Figure 5(b)). To support the shown V&V experiment an *Experiment* and a *Simulation Run Configuration* have to be programmed. Having a programmed experiment and a programmed configuration makes it hard or even impossible to track execution results and provisional data during experiment execution if it is not intended by the implementation.

Replacing the *Experiment* and the *Simulation Run Configuration* with workflows allows for



(a) Current experimentation layer of JAMES II

(b) Color Mapped validation experiment (from Figure 2) steps to layer parts (green: *Experiment* ; blue: *Simulation Run Configuration*)

Figure 5: Experimentation Layer of JAMES II and mapping of workflow steps to it

- flexible definition of any kind of experiment execution schemes,
- easy extension, changing or testing of experiment schemes,
- auto documentation,
- data provenance,
- progress information,
- repeatability combined with automated/smart reruns, and
- role based execution and resource management.

Considering the presented requirements and presented example use cases the question arises whether such workflows should be realized based on an already existing workflow management system or whether such a system should be realized in JAMES II. The same question arises when considering role resp. user management in combination with workflows.

As we have the design rule also stated in (Leye et al. 2008) that the core of JAMES II has to be completely independent from any third party tool or library (to make the core highly reusable) and both the workflow integration and role management would be core components since experimentation layer and maybe other core components will rely on it, we will create at least a simple workflow and user management system for JAMES II on our own. But as the spirit of JAMES II dictates interchangeability it will be possible to combine this with or to even replace the functionality by any external solution such as Project Trident, Kepler or Taverna. The myExperiment project for instance could also be used as external solution for storing and sharing workflows.

6 CONCLUSIONS

Well defined processes to achieve diverse M&S products are important. Workflows can be used to define those processes and to ensure adherence of step orders within those processes.

Workflows give the benefits of being able to track product creation stepwise, make the process of creation repeatable, documented, monitorable and automatable, gain insight of what comes out of and goes into different steps, give an intuitive way of describing processes as well as ensure a specific quality.

Systems such as Trident Project, Kepler and Taverna support arbitrary scientific oriented workflows they are more like visual data processing tools aiming at automating recurring data processing steps. Workflows for M&S processes can have a fair amount of user involved tasks and can be more flow oriented than data oriented. Additionally a workflow system for the M&S domain should fulfill the presented requirements to leverage the benefits of workflows.

An important requirement is the support of roles. Using roles results in clearly defined responsibilities so they are an essential part when it comes to produce reliable and high quality products in M&S. Through roles it is possible to have domain experts incorporated and enforced within M&S processes as suggested by, e.g., Balci (Balci 1990)

We strongly believe that the application of workflows and its benefits, especially process documentation, is one of the means to increase the overall quality of M&S results and is a prerequisite to overcome the crisis of credibility (Kurkowski 2006). As shown for JAMES II, workflow support in M&S software is a desirable and possible feature and should be evaluated for other M&S systems as well.

ACKNOWLEDGMENT

This research is supported by the DFG (German Research Foundation).

REFERENCES

Altintas, I., C. Berkley, E. Jaeger, M. Jones, B. Ludäscher, and S. Mock. 2004. Kepler: An extensible system for design and execution of scientific workflows. In *Proceedings of the 16th International Conference on Scientific and Statistical Database Management (SSDBM'04)*.

Balci, O. 1990. Guidelines for successful simluation studies (tutorial session). In *Proceedings of the 1990 Winter Simulation Conference*, ed. O. Balci, 25–32. Piscataway, NJ, USA: IEEE Press.

Balci, O. 2004. Quality assessment, verification, and validation of modeling and simulation applications. In *Proceedings of the 2004 Winter Simulation Conference*, 122–129: Winter Simulation Conference.

Barga, R., J. Jackson, N. Araujo, D. Guo, N. Gautam, K. Grochow, and E. Lazowska. 2008. Trident: Scientific workflow workbench for oceanography. *Services, IEEE Congress on* 0:465–466.

Barga, R., J. Jackson, N. Araujo, D. Guo, N. Gautam, and Y. Simmhan. 2008. The trident scientific workflow workbench. In *ESCIENCE '08: Proceedings of the 2008 Fourth IEEE International Conference on eScience*, 317–318. Washington, DC, USA: IEEE Computer Society.

Gamma, E., R. Helm, R. Johnson, and J. Vlissides. 1995. *Design patterns: elements of reusable object-oriented software*. Addison-Wesley Professional.

Gil, Y., E. Deelman, M. Ellisman, T. Fahringer, G. Fox, D. Gannon, C. Goble, M. Livny, L. Moreau, and J. Myers. 2007, dec.. Examining the challenges of scientific workflows. *Computer* 40 (12): 24 –32.

Himmelspach, J., R. Ewald, and A. M. Uhrmacher. 2008, December. A flexible and scalable experimentation layer for JAMES II. In *Proceedings of the 2008 Winter Simulation Conference*, ed. S. Mason, R. Hill, L. Moench, O. Rose, J. Fowler, and T. Jefferson, 827–835: IEEE Computer Society.

Himmelspach, J., and A. M. Uhrmacher. 2009a. The james ii framework for modeling and simulation. In *HIBI '09: Proceedings of the 2009 International Workshop on High Performance Computational Systems Biology*, 101–102. Washington, DC, USA: IEEE Computer Society.

Himmelspach, J., and A. M. Uhrmacher. 2009b, June. What contributes to the quality of simulation results? In *Proceedings of the 2009 INFORMS Simulation Society Research Workshop*, ed. L. H. Lee, M. E. Kuhl, J. W. Fowler, and S. Robinson, 125–129. University of Warwick, Coventry, U.K.: INFORMS Simulation Society.

Hoops, S., S. Sahle, R. Gauges, C. Lee, J. Pahle, N. Simus, M. Singhal, L. Xu, P. Mendes, and U. Kummer. 2006. Copasi - a complex pathway simulator. *Oxford Journal - Bioinformatics* 22 (24): 3067–3074.

Hull, D., K. Wolstencroft, R. Stevens, C. Goble, M. Pocock, P. Li, and T. Oinn. 2006, July. Taverna: a tool for building and running workflows of services. *Nucleic Acids Research* 34 (Web Server issue): 729–732.

ISO 9001:2008 2008. *Quality management systems – requirements*. ISO, Geneva, Switzerland.

Jeschke, M., and R. Ewald. 2008. Large-scale design space exploration of ssa. In *CMSB '08: Proceedings of the 6th International Conference on Computational Methods in Systems Biology*, 211–230. Berlin, Heidelberg: Springer-Verlag.

Kemper, P. 2009, September. Report generation for simulation traces with traviando. In *International Conference on Dependable Systems & Networks*, 347–352: IEEE Computer Society.

Kurkowski, S. H. 2006, October. *Credible mobile ad hoc network simulation-based studies*. Ph. D. thesis, Colorado School of Mines.

Law, A. M. 2007. *Simulation modeling and analysis*. 4 ed. McGraw-Hill International.

Lee, E. A., and S. Neuendorffer. 2007, Oct. Tutorial: Building ptolemy ii models graphically. Technical Report UCB/EECS-2007-129, EECS Department, University of California, Berkeley.

Leye, S., J. Himmelspach, M. Jeschke, R. Ewald, and A. M. Uhrmacher. 2008. A grid-inspired mechanism for coarse-grained experiment execution. In *DS-RT '08: Proceedings of the 2008 12th IEEE/ACM International Symposium on Distributed Simulation and Real-Time Applications*, 7–16. Washington, DC, USA: IEEE Computer Society.

Leye, S., J. Himmelspach, and A. M. Uhrmacher. 2009, March. A discussion on experimental model validation. In *Proceedings of the 11th International Conference on Computer Modeling and Simulation*, 161–167: IEEE Computer Society.

Leye, S., and A. M. Uhrmacher. 2010, Mar. A flexible and extensible architecture for experimental model validation. In *3rd International ICST Conference on Simulation Tools and Techniques (SIMUTOOLS'2010), Malaga, Spain*.

Liu, J., F. L. Perrone, D. M. Nicol, M. Liljenstam, C. Elliott, and D. Pearson. 2001. Simulation modeling of large-scale ad-hoc sensor networks. In *European Simulation Interoperability Workshop*.

Ludäscher, B., I. Altintas, C. Berkley, D. Higgins, E. Jaeger, M. Jones, E. A. Lee, J. Tao, and Y. Zhao. 2006. Scientific workflow management and the kepler system: Research articles. *Concurr. Comput. : Pract. Exper.* 18 (10): 1039–1065.

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

Pawlikowski, K. 2003. Do not trust all simulation studies of telecommunication networks. *Information Networking*:899–908.

Pawlikowski, K., H. d. Joshua Jeong, and J. s. Ruth Lee. 2002. On credibility of simulation studies of telecommunication networks. *IEEE Communications Magazine* 40:132–139.

Perrone, L. F., C. J. Kenna, and B. C. Ward. 2008. Enhancing the credibility of wireless network simulations with experiment automation. In *WIMOB '08: Proceedings of the 2008 IEEE International Conference on Wireless & Mobile Computing, Networking & Communication*, 631–637. Washington, DC, USA: IEEE Computer Society.

Rizk, A., G. Batt, F. Fages, and S. Soliman. 2008. On a continuous degree of satisfaction of temporal logic formulae with applications to systems biology. In *CMSB '08: Proceedings of the 6th International Conference on Computational Methods in Systems Biology*, 251–268. Berlin, Heidelberg: Springer-Verlag.

Roure, D. D., and C. Goble. 2009. Software design for empowering scientists. *IEEE Software* 26:88–95.

Roure, D. D., C. Goble, J. Bhagat, D. Cruickshank, A. Goderis, D. Michaelides, and D. Newman. 2008, December. myexperiment: Defining the social virtual research environment. In *4th IEEE International Conference on e-Science*, 182–189: IEEE Press.

Sargent, R. G. 2008. Verification and validation of simulation models. In *Proceedings of the 2008 Winter Simulation Conference*, ed. S. Mason, R. Hill, L. Moench, O. Rose, J. Fowler, and T. Jefferson, 157–169.

Sargent, R. G., R. E. Nance, C. M. Overstreet, S. Robinson, and J. Talbot. 2006. The simulation project life-cycle: models and realities. In *WSC '06: Proceedings of the 38th conference on Winter simulation*, 863–871: Winter Simulation Conference.

UC Berkley EECS Dept. 2010, January. Ptolemy project - ptolemy ii. http://ptolemy.berkeley.edu/ptolemyII/. Zugriff am 7. Januar 2010.

van der Aalst, W., and K. van Hee. 2004, January. *Workflow management: Models, methods, and systems*, Volume 1 of *MIT Press Books*. The MIT Press.

Weidemann, A., S. Richter, M. Stein, S. Sahle, R. Gauges, R. Gabdoulline, N. Semmelrock, B. Besson, I. Rojas, R. Wade, and U. Kummer. 2008. Sycamore - a systems biology computational analysis and modeling research environment. *Oxford Journal - Bioinformatics* 24 (24): 1463–1464.

AUTHOR BIOGRAPHIES

STEFAN RYBACKI holds a diploma in Computer Science from the University of Rostock and pursues a PhD at the Modeling and Simulation Group at the University of Rostock. His main research interests are the pros and cons of workflows in the realm of M&S. His email address is <stefan.rybacki@uni-rostock.de>.

ENRICO SEIB is a master student writing his masters thesis on workflows in M&S.

JAN HIMMELSPACH is a post doc in the Computer Science Department at the University of Rostock. He received his doctorate in Computer Science from the University of Rostock. His research interest is on software engineering for modeling and simulation, credibility of modeling and simulation, and on efficient modeling and simulation solutions. His e-mail address is <jan.himmelspach@uni-rostock.de>

A. M. UHRMACHER is a Professor at the Department of Computer Science at the University of Rostock and head of the Modeling and Simulation Group. Her research interests are in modeling and simulation methodologies and their applications. Her e-mail and web addresses are <lin@informatik.uni-rostock.de> and <www.informatik.uni-rostock.de/~lin>.