Proceedings of the 2017 Winter Simulation Conference
W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, eds.

PROVENANCE IN MODELING AND SIMULATION STUDIES – BRIDGING GAPS

Andreas Ruscheinski
Adelinde Uhrmacher

Institute of Computer Science
University of Rostock
Albert-Einstein-Str. 22
D-18059 Rostock, GERMANY

ABSTRACT

Simulation studies are intricate processes that require interweaving model refinement and executing diverse experiments. Simulation models and data are the result of complex and interactive model and data generating processes. Information about these processes are required to assess the quality of simulation products. Capturing provenance, i.e., information about how a product has been generated, is a major concern both for assessing and reproducing scientific experiments. For parts of a simulation study, support for capturing and managing provenance is available. However, still gaps exist, e.g., how simulation models have been generated and to look therefore beyond individual simulation experiments and even simulation studies. To bridge those gaps it will be central to exploit, refine, and combine diverse methods effectively, as we demonstrate on a concrete case study and its provenance model.

1 INTRODUCTION

Simulation studies typically focus on exploring what-if scenarios or predicting a system's behavior. Recently also increasingly it is asked, where do the simulation model or simulation data come from? This type of question is the subject of provenance research. Provenance refers to gathering information about how a product has been generated, i.e., who created a product when and why, and when was the product modified by whom (Simmhan et al. 2005). As stated by the W3C Provenance Working Group, provenance provides "information about entities, activities, and people involved in producing a piece of data or thing, which can be used to form assessments about its quality, reliability, or trustworthiness"(Groth and Moreau 2013). In experimental sciences, provenance is essential for ensuring reproducibility of real experiments and of *in-silico* experiments (Cheney et al. 2012). Thereby, reproducibility, in a broader meaning, refers to repeatability, replicability, and reproducibility, in a more narrow meaning. In the following we will interpret the terms similarly as in (Feitelson 2015):

- repeatability: the same results can be generated by applying the same methods in the same experimental setting by the same people.
- replicability: the product can be achieved by others with the same methods.
- reproducibility: the same results can be achieved by others with independent means.

To apply provenance to products of modeling and simulation studies requires to identify the central processes and products of modeling and simulation and to put those into relation. Therefore, we will first present a case study. Secondly we will discuss provenance and reproducibility referring to different modeling and simulation products. We will develop a provenance model for the case study, focusing on simulation models, simulation data, and, in addition, experiment specification as central products. Finally, we will

978-1-5386-3428-8/17/$31.00 ©2017 IEEE

illustrate how different methods can be applied to support different parts of the provenance model and can help bridging gaps to realize a more comprehensive provenance of modeling and simulation products.

2 THE CASE STUDY

The case study that shall illuminate central processes and products of modeling and simulation studies stems from the area of computational biology.

Studying the β-catenin Wnt-signalling pathway, we have observed an increase of nuclear β-catenin within our cellular system (neural progenitor cells) at 1 hour and after 8 hours in the wet-lab, which we cannot easily explain (Mazemondet et al. 2011). Thus, we want to develop a simulation model to reveal the underlying mechanisms. Based on an existing model from literature (Lee et al. 2003) we start reducing the model and adapting the parameters to the new cell type. We execute parameter scans to do a quick face validation and to check whether we come anywhere near the observed behavior. We start extending the simulation model, adding new model components, and check the sensitivity of the model's behavior referring to new quantities introduced into the model. After that again simulation experiments in terms of parameter scans are executed. Finally, one of the extensions does the trick, a fine-tuning of model parameters shows an acceptable compliance between simulation results and real-world observations which leads to the conclusion that at least for the second rise of concentration some membrane mediated processes (autocrine or paracrine mechanisms) sign responsible (Mazemondet et al. 2012).

In a later simulation study, a further extension of the simulation model shall allow to study those membrane mediated processes more closely. Most parameters can be taken from literature, the previous model, or current Wet-Lab experiments, however, some need to be estimated. The extended model is able to reproduce the observed data as the original model did before. The new model is successfully cross validated with a model from literature (Lee et al. 2003) after taking a scaling factor into account. The model and produced trajectories are also validated by predicting data from other wet-lab experiments documented in literature (Hannoush 2008). After perturbing the model similarly as has been done in those wet-lab experiments, the model nicely mimics not only qualitatively but also quantitatively the wet-lab data. However, after performing another perturbation, i.e., preventing the membrane induced dynamics to take effect, it becomes evident that indeed the first rise of the key players is due to mechanisms not yet covered in the model. This requires a further substantial extension of the model. As recent wet-lab studies indicated that ROS might be responsible for the early increase of β-catenin (Rharass et al. 2014), a ROS sub-model is developed, calibrated and integrated. Now the simulation shows the same behavior as observed in the wet-lab experiments (Haack et al. 2015).

Thus, different models, simulation experiments, data, wet-lab experiments, and even simulation studies motivated by different questions have contributed to the final simulation model and the simulation results presented in (Haack et al. 2015). In both simulation studies (Mazemondet et al. 2012, Haack et al. 2015) different versions of a simulation model as well as diverse data sets have been generated, by successively running experiments and refining the simulation model (Rybacki et al. 2014, Peng et al. 2017). Data producing as well as simulation model building processes are closely related within simulation studies (and possibly, as in the above example, reaching out beyond a single simulation study). However, how does the different nature of building a simulation model or generating data by executing a computational experiment, reflect on provenance and reproducing, replicating, or repeating simulation products and, first of all, what are the central products of modeling and simulation studies?

3 PROVENANCE, REPRODUCIBILITY, REPLICABILITY, AND REPEATABILITY

(Balci 2012) distinguishes a set of different products within the modeling and simulation life cycle: the formulated problem, the requirement specification, the conceptual model, architecture specification, design specification, executable sub-models, simulation model, simulation results, presented results, and the certified simulation model, implicitly assuming a model generation by composing sub-models. To make our point that trust into simulation studies relies on provenance of different products which might ask for different means and cannot be evaluated in isolation, out of this list our focus will be on two main products of simulation studies, i.e., simulation model and simulation data.

3.1 Provenance of the simulation model

The main product of simulation studies is not only the data produced but the simulation model itself. Major efforts are therefore dedicated to making models accessible and facilitating their reuse. Thereby, suitable annotations play a central role. According to the ODD protocol (Overview, Design concepts, and Details) which is widely adopted in agent-based modeling and simulation, models should be annotated with the purpose of the model, input data, state variables and scales, process overview, design concepts, and initialization (Grimm et al. 2010). Whereas part of this information becomes obsolete, if a formal, executable modeling approach is exploited, other parts of ODD reveal context information of a simulation model. Similar information is gathered in SBML (Systems Biology Markup Language) to facilitate the reuse of models (Hucka et al. 2003). As SBML is designed as an interchange format between different simulation system, it mixes unstructured text, e.g., about the source, the owner, limitations, or purpose of the model, with controlled vocabularies including ontologies e.g., to denote state variables and their initialization. This additional information improves generally the understanding of the model, but also entails information directly related to a model's provenance by providing information about the author and the context of generating the model. For the later also simulation experiments that the model is annotated with are of interest, as they can convey specific, required behavioral properties of the model (Peng et al. 2016), or allow reconstructing, how specific parameter values have been found, e.g., by exploiting specific parameter scans.

Also conceptual models can provide valuable information to assess a simulation model's quality, reliability, or trustworthiness. The term conceptual model is broadly and not coherently used in modeling and simulation. It might subsume rather different concepts that are related to a simulation model and its context, e.g., variables, their scales, relationships between variables, qualitative models, assumptions, invariants and requirements, so all information leading up to the simulation model (Fujimoto et al. 2017). With the formality of the conceptual model the potential for computational support increases. In (Mustafiz et al. 2012) a formalism transformation graph relates formal models at higher abstractions (e.g. Petri Nets models) to more detailed simulation models. This approach allows refining models from qualitative ones to the executable level systematically and semi-automatically.

Information about the simulation model are essential for assessing the quality of the model, improving our trust into the model, and thus reusing it. However, the information does not primarily serve and typically also falls short for repeating, replicating, or reproducing the simulation model. This is partly due to the model building process being highly interactive where phases of model design, refinement, and experimenting with the model are intertwined (Rybacki et al. 2014).

3.2 Provenance of the simulation data

Another product of simulation studies are the data produced. Here clearly the interest lies in reproducing, or in most cases rather repeating or replicating the simulation experiment with the expectation to achieve the same simulation data. Therefore, models are annotated with simulation experiments. As observed in the previous section, annotating models with calibration or validation experiments, e.g., parameter scans, simulation-based optimization, reveals part of the model's building process. However, the annotation with simulation experiments allows also to replicate simulation data shown in publications, which increases the trust into these data. At the same time it increases the trust into the model, as the model is able to produce the documented results in a publication (Waltemath et al. 2011). For replicating the data shown in published figures, standards such as SED-ML (Waltemath et al. 2011) focus on the model, the initialization of the model, the simulator and its configuration (incl. type and version of the execution algorithm, and stop condition), and data to be observed. This allows to repeat the *in-silico* experiments within one group, to replicate simulation data between groups, and to finally challenge the results by exploiting different means, e.g., different execution algorithms, and thus to reproduce the simulation data.

The success of repeating, replicating and reproducing simulation data is typically measured by comparing the results. If deterministic simulations are repeated, we expect exactly the same simulation results. In case of replicating simulation data authors and the computing infrastructure will vary. The later might introduce slight differences among simulation results. If simulation data are reproduced using different methods, the differences are likely to increase. To assess differences, suitable similarity measures need to be defined. This is particularly the case for stochastic simulations. An in-detail discussion of different degrees of *same-ness* of simulation data and the data producing processes when repeating, replicating or reproducing simulation data can be found in (Dalle 2012).

3.3 Provenance of modeling and simulation products

Provenance in modeling and simulation is aimed at increasing our trust in its products. However, this trust is not necessarily used for repeating, replicating, or reproducing the products, at least not when it comes to the simulation model. Whereas we are interested in understanding the context, which a model has been developed in, information about provenance is rather seldom used for reproducing the model in a model generating process. Similarly, this observation likely applies to other simulation products listed in (Balci 2012), e.g., conceptual model, or requirements specification. Provenance information is frequently exploited for and even aimed at repeating, replicating, or reproducing *computational aspects* and thus easier to automate, e.g., the generation of simulation data, of simulation studies. Although provenance of products that rely heavily on human activity might be used for replication or reproduction, due to the effort induced, this seems to happen rather rarely (Uhrmacher et al. 2016). Thus, we expect the concrete means in terms of methods to collect and apply provenance for the diverse artifacts within and across simulation studies to vary.

4 A PROVENANCE MODEL FOR OUR CASE STUDY

To illuminate what kind of information can be provided by provenance, we will build a provenance model for the simulation model and simulation data presented in (Haack et al. 2015). Therefore, we will exploit the open provenance model. The open provenance model (Moreau et al. 2011) consists of 1) artifacts, a digital representation within a computer system, in our case simulation model, simulation data, and simulation experiments 2) processes, a series of actions performed on or caused by artifacts and resulting in new ones, and 3) agents, contextual entities enabling, facilitating, controlling, or affecting a process. Five dependencies are distinguished: 1) a process might use an artifact (in a specific role), 2) an artifact might be generated by a process (in a specific role), 3) one process might be triggered by another process, and 4) one artifact might be derived from another artifact, 5) finally a process might be controlled by an agent. Roles allow distinguishing how different products are used or produced by a process, and how different agents are controlling a process. Thus, roles can be used to enrich the provenance model with crucial information.

4.1 Artifacts and relations

A provenance model for our case study (see Section 2) is depicted in Figure 1. A description of the identifiers referring to artifacts and processes can be found in Table 1.

The provenance model allows to trace via the relation *wasDerivedFrom* the final model M3' back via M3 and M2 to M1. Models and simulation data are generated by processes. The model generating process is a rather intricate process, requiring human interaction and running experiments for calibration and validation. The above provenance model treats the entire process of developing a simulation model as black-box. Its internal complexity is indicated by P2, P4, and P6 (the model generating processes) using

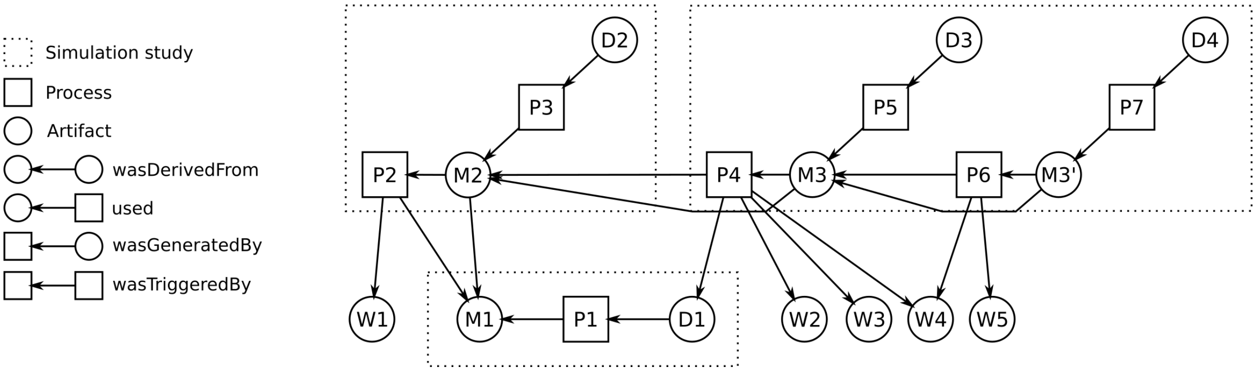

Figure 1: Provenance model of the case study: the prefix P denotes processes, that generate data or models, the prefix D data produced by simulation experiments and the prefix W data from real world (here Wet-lab) experiments, and the prefix M the simulation models generated. Three simulation studies and their results contributed to the final simulation model. Please note that the graph is read from right to left to support the retrospective provenance: what contributed to our simulation artifact.

Table 1: Description of the identifiers from the example provenance model in Figure 1.

Id.	Description	Reference
W1	Westernblot data, e.g., nuclear β-catenin	(Mazemondet et al. 2011)
M1	ODE canonical Wnt model (based on Xenopus)	(Lee et al. 2003)
P1	Simulation data producing process	(Lee et al. 2003)
D1	Simulation data, e.g., concentration of β-catenin	(Lee et al. 2003)
M2	Stochastic Wnt model incl. autocrine mechanisms	(Mazemondet et al. 2012)
P2	Model building process (incl. model building and calibration and validation experiments)	(Mazemondet et al. 2012)
D2	Simulation data, e.g. nuclear β-catenin	(Mazemondet et al. 2012)
P3	Simulation data producing process	(Mazemondet et al. 2012)
W2	Westernblot data, on β-catenin related to Wnt	(Hannoush 2008)
W3	LRP6 initial value	(Bafico et al. 2001)
W4	Westernblot, microscopy data, e.g., on nuclear β-batenin	(Haack et al. 2015)
M3	Stochastic Wnt Model, incl. membrane dynamics	(Haack et al. 2015)
P4	Model building process	(Haack et al. 2015)
D3	Simulation data, e.g. nuclear β-catenin	(Haack et al. 2015)
P5	Simulation data producing process	(Mazemondet et al. 2012)
W5	Westernblot, microscopy data, e.g., on ROS	(Rharass et al. 2014)
M3'	Stochastic Wnt Model, incl. membrane dynamics and ROS model	(Haack et al. 2015)
P6	Model building process	(Haack et al. 2015)
D4	Simulation data, e.g. nuclear β-catenin	(Haack et al. 2015)
P7	Simulation data producing process	(Haack et al. 2015)

several, diverse artifacts from other simulation studies as well as wet-lab experiments for different purposes. The provenance model explicitly describes which data have been used for calibration and which data for validation and how a model builds on previous models. Therefore, defining suitable roles on relations is essential, see Table 2. The artifacts that are used by the model generating processes refer to data that are used as input parameters, data for calibration, for validation, and for cross validation, and to simulation models that are extended or composed in the process of generating a new model.

Table 2: Defining roles for the artifact *used* by process relations (see Figure 1).

Relation	Role	Description
W1-used-P2	calibration	fitted to β-catenin data from wet-lab experiments
M1-used-P2	adaptation	adaptation of parameters, and aggregation of reactions
D1-used-P4	validation	cross validation with the outcomes of the simulation model M1
W2-used-P4	validation	independent Wnt wet-lab experiments for validating the model
W3-used-P4	input values	LRP6 concentration
W4-used-P4	falsification	β-catenin data from wet-lab after distortion of membrane dynamics
M2-used-P4	extension	the model was refined, membrane dynamics were added
W4-used-P6	validation	β-catenin data from Wet-lab after distortion of membrane dynamics
M3-used-P6	composition	added ROS / DVL model component
W5-used-P6	calibration	fitted to ROS data from wet-lab experiments
M3-used-P5	experimentation	used to experiment and produce data, similarly relation between M1 and P1, M2 and P3, M3' and P7

Please note that this provenance graph only lists a fraction of the actual data artifacts used within and outside the simulation studies. In addition, the level of detail is biased by our focus on the simulation study presented in (Haack et al. 2015). In addition, our provenance model does not consider agents and relating agents to specific processes. The reason for this is that in our case study only one person signed responsible for executing all steps at least in the two studies that lead to the models M2, M3, and M3' respectively. However the roles can describe how the agent was affecting the process, e.g, in the *changed simulation model*-role the agent participated in the development of the simulation model or in the *verified & validated simulation model*-role the agent participated in the verification and validation of the simulation model. Another omission refers to a modeling and simulation product that is crucial for provenance of simulation models and simulation data, i.e., simulation experiments.

4.2 Experiments as first class products and thus subject of provenance

Simulation experiments, their careful design, configuration, and conduction are central in developing a model and for generating simulation data. Making experiments explicit and specifying them unambiguously facilitates high quality and reproducible modeling and simulation research (Waltemath et al. 2011, Ewald and Uhrmacher 2014, Teran-Somohano et al. 2015). Simulation experiments are an invaluable means to add to the provenance of simulation products, like simulation data and simulation models (see also Section 3), and as such are increasingly seen as a product of simulation studies in their own rights. Thus, they become themselves, subject of provenance. In (Peng et al. 2017) simulation experiments originally executed with model M3 (E1) were semi-automatically reused, updated (E1'), and executed with model M3' to support the validation of M3'. The approach traced the experiment specifications back to earlier experiment specifications and executions which were updated partly interactively and partly automatically (Figure 2). The provenance model makes artifacts and relations how artifacts have been generated explicit. From the provenance model constraints and even necessities to act can be automatically derived, e.g., that if doubts about an artifact are raised processes that rely on these artifacts need to be redone.

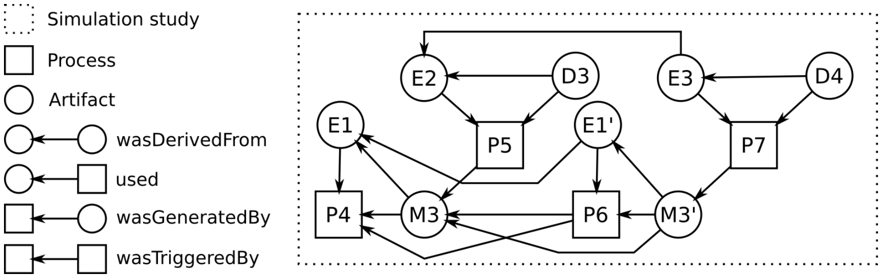

Figure 2: Provenance model including experiments for the third simulation study.

5 METHODS AND TECHNIQUES

As provenance is concerned with questions like who created these artifacts, when were the artifacts created or modified and by whom, and what was the process used to create them, we take a closer look at meta-data and techniques that can be exploited to collect, document, interpret, and retrieve the required information.

5.1 Version control systems

A version control system (VCS) allows keeping track of document changes. Each change of a document creates a new version of it. Based on the document history, it is possible to determine what has been changed. Most VCS systems provide further information like the creation-time, the time of the last change and authors of these changes. Each version can be annotated with a comment which gives further information about and reasons for the changes. All documents are stored in files and managed in a repository.

If individual models are stored in files, then the above methods can easily be applied to track changes and authors of changes within models. Applying VCS to models is of particular interest for more complex simulation studies and those in which more than one person are involved in the development of a model. Similarly, this applies to the specification of simulation experiments. Please notice, version control appears of less relevance to the simulation data as simulation data are not changed, but rather newly generated.

How easily the changes between model versions can be interpreted depends on the description format of the simulation models. If model and simulators are not clearly distinguished, assessing the changes becomes more difficult. Also the metaphor of model equals document appears less applicable. However, at the moment we do have a clear separation between model and simulator and in addition a specification of syntax and semantics, the identification of changes is not restrained to the usual textual differences, but can reveal semantic information about changes. To those belong whether a comment has been changed, whether variables have been added, whether parameter values have been updated or a behavior rule has been deleted (Waltemath et al. 2013). The more restrictive the format is in which the models are stored the easier it is to automatically identify and classify the changes, and thus to provide valuable information to the user. However, assessing automatically the impact of a change remains still an open challenge.

Regarding our open provenance model, a version control system can be used to retrieve information about the *wasDerivedFrom* relation between model versions (here M3 and M3') within the simulation study. Also different models might be related by *wasDerivedFrom*, e.g. M3 and M2. Being designed in different simulation studies (and thus for a different purpose or based on different data), this qualifies for models M3 and M2 being considered different models not merely versions of one model.

Also, referring to simulation experiments, a declarative and constrained specification of simulation experiments facilitates to retrieve simulation experiments and to automatically evaluate the changes between different simulation experiments, e.g., between E1 and E1' (Figure 2) (Peng et al. 2017).

Version control systems, such as Git, support the automatic generation of provenance models (De Nies et al. 2013) and their application to modeling and simulation studies the automatic tracking of the *wasDerivedFrom* relation within this particular study. Version control systems appear already widely used in modeling and simulation, e.g., be this as part of project management as supported in Matlab (Mathworks

2017) or be this as a feature of model repositories, e.g., the Physiome Model Repository 2 (Yu et al. 2011). However, they do not reach beyond individual simulation studies and are constrained to one particular relation of the provenance model, i.e., *wasDerivedFrom*, and are not applicable to all products of the modeling and simulation study equally, e.g., to simulation data.

5.2 Scientific workflows, scripts and domain specific languages

Workflows, which have originally been developed to support the automation of repetitive business tasks (WfMC 1996, p. 8), can also capture complex data analysis processes. These scientific workflows can be viewed as executable specifications of data driven processes and therefore can be placed closely to scripting approaches, like those in Perl and Python, or domain specific languages. As scientific workflows, scripts and domain-specific languages can be used to automate and organize computational processes, they only provide support for those parts of our provenance model that rely on computation. This refers mainly to how simulation data are produced but also to those parts of the model generating process that relies on simulation experiments for calibration and validation.

The main difference between these approaches is the way how the computational processes are described and the services the approaches are shipped with. Workflows provide well-defined languages for specifying and executing complex computational data processing tasks from simpler ones and systematically capture provenance information in an execution environment for the derived data products (Ludäscher et al. 2009, p. 10-11). By using scripts, the user needs to implement the management of used and generated data and the execution procedure. To record provenance data, the execution of the script needs to be logged and analyzed. The domain specific languages for experimentation are closing the gap between simple scripting and workflow systems referring to features offered and accessibility of the language. They provide a syntax for a precise description of experiments using domain-specific terminology (Ewald and Uhrmacher 2014).

Since these approaches typically focus on computerized processes, the encoded provenance data relates to these processes. Referring to our provenance model the *wasGeneratedBy* relations between simulation data and processes can easily be covered. The description of the process, independently whether it has been done as scripting, in a domain-specific language, or as scientific workflow, refers to the artifact simulation experiment in the provenance model. Additionally, the *used* relation between the processes, simulation model and the data is typically encoded in these descriptions (which themselves form the simulation experiment artifacts E1, . . . , E3).

Whereas most workflow systems assume workflows to follow repetitive patterns, some take a closer look at the changes between workflows. Approaches like Model-as-you-go (Sonntag and Karastoyanova 2013), rooted in adaptive workflow systems such as ADEPT flex (Reichert and Dadam 1998), add flexibility to the execution of experimental workflows. Other workflow approaches turn their attention to the question of how workflows change. VisTrails (Scheidegger et al. 2008) was designed to support exploratory tasks in which workflows are iteratively refined. We find workflow specifications being treated as first class objects. Thereby the evolution of workflows, or in our case simulation experiments, can be observed. Thus, workflows become the subject of provenance, in addition to providing provenance of the generated data.

Workflow systems ship with an automatic documentation and annotation of the achieved products based on the data or model generating processes. Thus, crucial provenance information is collected. However, additional provenance information is needed which typically rely on explicit annotation. Some workflow systems provide an infrastructure to support annotations and integrate ontologies (Wolstencroft et al. 2013).

5.3 Ontologies and annotations

An ontology provides a naming and definition of categories, properties, and interrelationships of entities within a particular domain. Ontologies can be used to refer to the application domain of our simulation model, e.g., to what entities in the real world do the variables refer to, or what are the model's assumption. Ontologies can also refer to the modeling and simulation techniques used, and thus help us, e.g., to classify a model as being discrete event and being formalized within a specific formalism (Miller et al. 2004) or which type of simulation algorithm has been used (Waltemath et al. 2011). Thus, if models or simulation experiments are annotated with ontologies they help to relate different models or simulation experiments to each other and querying databases containing models or simulation experiments. The annotation of simulation models provides meta-data on the model, about its assumptions, parameters, constants, and behavior. Regarding the provenance of the simulation model, annotations can be used for referencing used data in the model creation process and relating the final model with other models. For example, in our case study the simulation model M3 can be annotated with information about wet-lab data used as model parameters (W3 - P4) or simulation data used for cross validation with another model (D1 - P4). These relations can be categorized by an according provenance model. Meta-data of simulation experiments typically entails further information about the simulation algorithm, procedures used for data processing, data generated and the used simulation model (Waltemath et al. 2011). Regarding the provenance of the simulation data these annotations describe the relation between the simulation experiments, used simulation model and the data generated. For example in the provenance model (see Figure 2) the relations from the data generating process P5 to the simulation model M3 and the generated data D3 can be encoded in the artifact simulation experiment E1. The more structured and formalized annotations are the easier information can be extracted automatically and be put to use.

5.4 Combining methods

Each of the previous described methods can cover or support different relationships between artifacts and processes within the provenance model.

- The *wasDerivedFrom* relationship between artifacts can be supported by exploiting a version control system to modeling and simulation products. It is applicable to simulation model and simulation experiments, and modeling and simulation products which are updated rather than being generated like simulation data. Whereas pure version control systems provide only a difference between documents, further support for the user is necessary to assess changes. This requires further information about the structure and semantics of the modeling and simulation products.
- How artifacts are *used* within processes is specified in the scripting of a simulation experiment, domain-specific languages and scientific workflows. In addition, annotations provide background on the data or models used. In this context ontologies play an important role.
- The annotation of simulation models and simulation data with simulation experiments informs the *wasGeneratedBy* relationship between artifacts and processes. It allows to repeat, replicate and reproduce simulation data and supports the replication and reproduction of simulation models.
- The *wasTriggeredBy* relation between processes again relies on annotations and leads, if applied to model building processes, typically to new versions of a model that then can be traced by the version control system. Thus, this additional information can be used to distinguish more important from less important changes within models.

The annotation of simulation models and simulation data with simulation experiments can be done automatically if simulation experiments are scripted, specified in a domain-specific language, exploit scientific workflows, or the used tools provide some format to facilitate storage and reuse of simulation experiments. For relating simulation experiments across simulation studies and also for interpreting changes between individual simulation experiments standardized descriptions are required, e.g., (Waltemath et al. 2011). Similarly, this applies to simulation models. Changes within simulation models and simulation experiments can then be tracked by a VCS. To assess and interpret the differences and changes, annotations and ontologies, as well as the provenance model can provide further clues. Thus, the provenance model provides structure and information in relating model and data generating processes across individual experiments and even simulation studies. Recorded provenance data of a simulation study can be used to query and infer information about its products using methods like OPQL(Lim et al. 2011). For example we can query for data which has been used for validating or calibrating a simulation model and infer if the data were later found out to be invalid, which simulation models needs to be newly calibrated and validated. Provenance data can be used to infer consequences of invalid calibration or validation data.

6 CONCLUSION

To provide comprehensive provenance within modeling and simulation, we identified central processes and products within a simulation study and mapped those to processes and artifacts within a provenance model. The provenance model developed for our simulation studies is based on the open provenance model. Exploiting roles in relating artifacts and processes provides additional structure and distinguishes explicitly between data used as inputs, for calibration, or validation within the model developing process. The provenance model provides a bird eyes view and additional structure and information about products and processes and their interrelations within and across simulation studies.

Different methods, which we grouped into version control systems, description of modeling and simulation processes via scripting, domain-specific languages, and workflows, and the annotation of simulation products, can be exploited to support different aspects of the provenance model.

Although developments in workflow management systems, like VisTrails, already combine version control systems and workflows to provide provenance of data generating processes and workflows, more tool support is necessary for managing provenance of modeling and simulation studies. For example, the model building process, that implies intertwining interactive model refinement and execution of simulation experiments, is not well-supported by workflows and provenance yet. Realizing provenance requires significant effort. To be of practical value collecting provenance data should be realized as transparent as possible. Knowledge about the structure, processes, products, agents, and their relations within modeling and simulation studies needs to be formalized to direct the collection of provenance data and to allow a more informed assessment of modeling and simulation products, and thus to close the gap between individual simulation experiments and simulation studies. Further the possibilities and limits of retrieving useful information based on provenance data, besides the consequences of invalidation calibration or validation data, needs to be explored using recorded provenance data from real simulation studies before the benefits of provenance data can be fully determined.

ACKNOWLEDGMENTS

The research was funded by the DFG (UH 66/18).

REFERENCES

Bafico, A., G. Liu, A. Yaniv, A. Gazit, and S. A. Aaronson. 2001. “Novel Mechanism of Wnt Signalling Inhibition Mediated by Dickkopf-1 Interaction with LRP6/Arrow”. *Nature cell biology* 3 (7): 683–686.

Balci, O. 2012. “A Life Cycle for Modeling and Simulation”. *Simulation* 88 (7): 870–883.

Cheney, J., A. Finkelstein, B. Ludäscher, and S. Vansummeren. 2012. “Principles of Provenance (Dagstuhl Seminar 12091)”. Volume 2, 84–113.

Dalle, O. 2012. “On Reproducibility and Traceability of Simulations”. In *Proceedings of the 2012 Winter Simulation Conference*, edited by C. Laroque, R. Pasupathy, and J. Himmelspach, 1–12. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

De Nies, T., S. Magliacane, R. Verborgh, S. Coppens, P. Groth, E. Mannens, and R. Van de Walle. 2013. “Git2PROV: Exposing Version Control System Content as W3C PROV”. In *Posters & Demonstrations Track within the 12th International Semantic Web Conference, Proceedings*, 125–128: CEUR-WS.

Ewald, R., and A. M. Uhrmacher. 2014. “SESSL: A Domain-specific Language for Simulation Experiments”. *ACM Transactions on Modeling and Computer Simulation (TOMACS)* 24 (2): 11:1–11:25.

Feitelson, D. G. 2015, January. “From Repeatability to Reproducibility and Corroboration”. *SIGOPS Oper. Syst. Rev.* 49 (1): 3–11.

Fujimoto, R., C. Bock, W. Chen, E. Page, and J. Panchal. (Eds.) 2017. *Research Challenges in Modeling and Simulation for Engineering Complex Systems*. Springer.

Grimm, V., U. Berger, D. L. DeAngelis, J. G. Polhill, J. Giske, and S. F. Railsback. 2010. “The ODD Protocol: A Review and First Update”. *Ecological modelling* 221 (23): 2760–2768.

Groth, P., and L. Moreau. 2013. “PROV-Overview. An Overview of the PROV Family of Documents”.

Haack, F., H. Lemcke, R. Ewald, T. Rharass, and A. M. Uhrmacher. 2015, 03. “Spatio-temporal Model of Endogenous ROS and Raft-Dependent WNT/Beta-Catenin Signaling Driving Cell Fate Commitment in Human Neural Progenitor Cells”. *PLOS Computational Biology* 11 (3).

Hannoush, R. N. 2008, 10. “Kinetics of Wnt-Driven -Catenin Stabilization Revealed by Quantitative and Temporal Imaging”. *PLOS ONE* 3 (10): 1–6.

Hucka, M., A. Finney, H. M. Sauro, H. Bolouri, J. C. Doyle, H. Kitano, A. P. Arkin, B. J. Bornstein, D. Bray, A. Cornish-Bowden et al. 2003. “The Systems Biology Markup Language (SBML): A Medium for Representation and Exchange of Biochemical Network Models”. *Bioinformatics* 19 (4): 524–531.

Lee, E., A. Salic, R. Krüger, R. Heinrich, and M. W. Kirschner. 2003. “The Roles of APC and Axin Derived from Experimental and Theoretical Analysis of the Wnt Pathway”. *PLOS Biology* 1 (1): e10.

Lim, C., S. Lu, A. Chebotko, and F. Fotouhi. 2011. “OPQL: A First OPM-level Query Language for Scientific Workflow Provenance”. In *2011 IEEE International Conference on Services Computing*, 136–143. IEEE.

Ludäscher, B., M. Weske, T. McPhillips, and S. Bowers. 2009. “Scientific Workflows: Business as Usual?”. In *International Conference on Business Process Management*, 31–47. Springer.

Mathworks 2017. “Matlab R2017a Documentation - About Source Control with Projects”. https://de.mathworks.com/help/simulink/ug/about-source-control-with-projects.html. [Online; accessed 27-April-2017].

Mazemondet, O., R. Hubner, J. Frahm, D. Koczan, B. M. Bader, D. G. Weiss, A. M. Uhrmacher, M. J. Frech, A. Rolfs, and J. Luo. 2011. “Quantitative and Kinetic Profile of Wnt/β-catenin Signaling Components during Human Neural Progenitor Cell Differentiation”. *Cellular & Molecular Biology Letters* 16 (4): 515.

Mazemondet, O., M. John, S. Leye, A. Rolfs, and A. M. Uhrmacher. 2012, 08. “Elucidating the Sources of -Catenin Dynamics in Human Neural Progenitor Cells”. *PLOS ONE* 7 (8): 1–12.

Miller, J. A., G. T. Baramidze, A. P. Sheth, and P. A. Fishwick. 2004, April. “Investigating Ontologies for Simulation Modeling”. In *37th Annual Simulation Symposium, 2004. Proceedings.*, 55–63.

Moreau, L., B. Clifford, J. Freire, J. Futrelle, Y. Gil, P. Groth, N. Kwasnikowska, S. Miles, P. Missier, J. Myers, B. Plale, Y. Simmhan, E. Stephan, and J. V. den Bussche. 2011. “The Open Provenance Model Core Specification (v1.1)”. *Future Generation Computer Systems* 27 (6): 743–756.

Mustafiz, S., J. Denil, L. Lúcio, and H. Vangheluwe. 2012. “The FTG+PM Framework for Multi-paradigm Modelling: An Automotive Case Study”. In *Proceedings of the 6th International Workshop on Multi-Paradigm Modeling*, MPM ’12, 13–18: ACM.

Peng, D., T. Warnke, F. Haack, and A. M. Uhrmacher. 2016. “Reusing Simulation Experiment Specifications to Support Developing Models by Successive Extension”. *Simulation Modelling Practice and Theory* 68:33–53.

Peng, D., T. Warnke, F. Haack, and A. M. Uhrmacher. 2017. “Reusing Simulation Experiment Specifications in Developing Models by Successive Composition – A Case Study of the Wnt/β-catenin Signaling Pathway”. *Simulation: Transactions of the Society for Modeling and Simulation International.*

Reichert, M., and P. Dadam. 1998. “ADEPT flex—Supporting Dynamic Changes of Workflows without Losing Control”. *Journal of Intelligent Information Systems* 10 (2): 93–129.

Rharass, T., H. Lemcke, M. Lantow, S. A. Kuznetsov, D. G. Weiss, and D. Panáková. 2014. “Ca2+-mediated Mitochondrial Reactive Oxygen Species Metabolism Augments Wnt/β-catenin Pathway Activation to Facilitate Cell Differentiation”. *Journal of Biological Chemistry* 289 (40): 27937–27951.

Rybacki, S., F. Haack, K. Wolf, and A. M. Uhrmacher. 2014. “Developing Simulation Models - from Conceptual to Executable Model and Back - an Artifact-based Workflow Approach”. In *Proceedings of the 7th International ICST Conference on Simulation Tools and Techniques*, SIMUTools ’14, 21–30: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

Scheidegger, C., D. Koop, E. Santos, H. Vo, S. Callahan, J. Freire, and C. Silva. 2008. “Tackling the Provenance Challenge One Layer at a Time”. *Concurrency and Computation: Practice and Experience* 20 (5): 473–483.

Simmhan, Y. L., B. Plale, and D. Gannon. 2005, September. “A Survey of Data Provenance in E-Science”. *SIGMOD Rec.* 34 (3): 31–36.

Sonntag, M., and D. Karastoyanova. 2013. “Model-as-you-go: An Approach for an Advanced Infrastructure for Scientific Workflows”. *Journal of Grid Computing* 11 (3): 553–583.

Teran-Somohano, A., A. E. Smith, J. Ledet, L. Yilmaz, and H. Oğuztüzün. 2015. “A Model-driven Engineering Approach to Simulation Experiment Design and Execution”. In *Proceedings of the 2015 Winter Simulation Conference*, edited by L. Yilmaz, I.-C. Moon, W. K. V. Chan, and T. Roeder, 2632–2643. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Uhrmacher, A. M., S. C. Brailsford, J. Liu, M. Rabe, and A. Tolk. 2016. “Panel - Reproducible Research in Discrete Event Simulation - A Must or Rather a Maybe?”. In *Proceedings of the 2016 Winter Simulation Conference*, edited by P. Frazier, T. Roeder, R. Szechtman, and E. Zhou, 1301–1315. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Waltemath, D., R. Adams, D. A. Beard, F. T. Bergmann, U. S. Bhalla, R. Britten, V. Chelliah, M. T. Cooling, J. Cooper, E. J. Crampin, A. Garny, S. Hoops, M. Hucka, P. Hunter, E. Klipp, C. Laibe, A. K. Miller, I. Moraru, D. Nickerson, P. Nielsen, M. Nikolski, S. Sahle, H. M. Sauro, H. Schmidt, J. L. Snoep, D. Tolle, O. Wolkenhauer, and N. Le Novre. 2011, 04. “Minimum Information about a Simulation Experiment (MIASE)”. *PLOS Computational Biology* 7 (4): 1–4.

Waltemath, D., R. Adams, F. T. Bergmann, M. Hucka, F. Kolpakov, A. K. Miller, I. I. Moraru, D. Nickerson, S. Sahle, J. L. Snoep, and N. Le Novère. 2011. “Reproducible Computational Biology Experiments with SED-ML - The Simulation Experiment Description Markup Language”. *BMC Systems Biology* 5 (1): 198.

Waltemath, D., R. Henkel, R. Hlke, M. Scharm, and O. Wolkenhauer. 2013. “Improving the Reuse of Computational Models Through Version Control”. *Bioinformatics* 29 (6): 742.

WfMC, G. 1996. “Terminology & Glossary”. *Document No WFMC-TC-1011. Workflow Management Coalition. Winchester* 204:1200.

Wolstencroft, K., R. Haines, D. Fellows, A. Williams, D. Withers, S. Owen, S. Soiland-Reyes, I. Dunlop, A. Nenadic, P. Fisher, J. Bhagat, K. Belhajjame, F. Bacall, A. Hardisty, A. Nieva de la Hidalga, M. P. Balcazar Vargas, S. Sufi, and C. Goble. 2013. “The Taverna Workflow Suite: Designing and Executing Workflows of Web Services on the Desktop, Web or in the Cloud”. *Nucleic Acids Research* 41 (W1).

Yu, T., C. M. Lloyd, D. P. Nickerson, M. T. Cooling, A. K. Miller, A. Garny, J. R. Terkildsen, J. Lawson, R. D. Britten, P. J. Hunter, and P. M. F. Nielsen. 2011. “The Physiome Model Repository 2”. *Bioinformatics* 27 (5): 743.

AUTHOR BIOGRAPHIES

ANDREAS RUSCHEINSKI is a Ph.D. student in the modeling and simulation group at the University of Rostock. His email address is andreas.ruscheinski@uni-rostock.de.

ADELINDE M. UHRMACHER is professor at the Institute of Computer Science, University of Rostock and head of the modeling and simulation group. Her email address is adelinde.uhrmacher@uni-rostock.de.