Agile Workflows for Climate Impact Risk Assessment based on the ci:grasp Platform and the jABC Modeling Framework

Samih Al-areqi [a]**, Steffen Kriewald**[b]**, Anna-Lena Lamprecht**[a]**, Dominik Reusser**[b]**, Markus Wrobel**[b] **and Tiziana Margaria**[a]

[a]*Potsdam University, Institute of Computer Science, August-Bebel-Str. 89, 14482 Potsdam, Germany (samih@cs.uni-potsdam.de, lamprecht@cs.uni-potsdam.de, margaria@cs.uni-potsdam.de)*

[b]*Potsdam Institute for Climate Impact Research, Po. Box 60 12 03, 14412 Potsdam, Germany, (kriewald@pik-potsdam.de, reusser@pik-potsdam.de, markus.wrobel@pik-potsdam.de)*

Abstract: Analyzing and assessing potential impacts of climate change are critical and challenging tasks that require the processing of large and heterogeneous datasets. These analyses are particularly demanding because of the multi-scale and multi-objective nature of environmental modeling for climate change impact assessment. The Climate Impacts: Global and Regional Adaptation Support Platform (ci:grasp) is a web-based climate information service for exploring climate change related information in its geographical context. In this paper, we show how the agile workflow development style supported by the jABC process modeling and execution framework permits us to leverage the processes implemented in the ci:grasp platform to a user-accessible level, and thus to enable users to easily define and perform multi-objective workflows tailored to their specific needs. Concretely, we discuss how we decomposed the ci:grasp's original assessment of the impact of sea level rise into basic computational services, which we then used as workflow building blocks. Together with additional functionality from the jABC, we built a flexible library of directly executable workflows for this type of analysis that is especially suitable to increase the productivity of researchers working on the analysis processes.

Keywords: scientific workflows; agile methods; model-driven development; climate information; climate impact risk assessment

1 INTRODUCTION

Analyzing and assessing potential impacts of climate change are critical and challenging tasks that require the processing of large and heterogeneous datasets. These analyses are particularly demanding because of the multi-scale and multi-objective nature of environmental modeling for climate change impact assessment [Lissner et al., 2014; IPCC, 2007]. Today there are numerous researchers working in climate impact analysis and related areas. Harmonization efforts of scenarios and input data for impact modeling allow for the first time to compare results and assess major sources of uncertainty [Warszawski et al., 2014]. There is a large number of emission scenarios, climate models, impact models and fundamental modeling assumptions available. For example, for sea-level rise (SLR), climate change is assessed with respect to the potential loss of agricultural production, calories available and effect for food security [Pradhan et al., 2014], but also with

respect to properties of rural and urban damage functions [Boettle et al., 2013]. To this end, heterogeneous data (such as, e.g., elevation, land-use, population density or yield data) has to be used, which comes in different formats and at different scales, requiring adequate integration and aggregation. Summarizing, combining modeling approaches, various data sources, multiple objectives and basic assumptions for climate impact assessment is a complex challenge.

The emergence of new software engineering approaches and tools such as service-oriented, model-driven methodologies and workflow technologies offers some solutions for the complexity challenges mentioned above. Recently agile methods in the spirit of [Beck et al., 2001] have become increasingly popular in software development. Their core principle is to open the software development work to customers and users, in order to improve the productivity, quality and stakeholder collaboration and satisfactions. Therefore, it values "individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to change over following a plan" [Beck et al., 2001]. The eXtreme Model-Driven Design (XMDD) paradigm [Kubczak et al., 2009; Margaria and Steffen, 2012] is an extreme version of model-driven development that supports a very agile and co-operative development of service-oriented systems by turning system development into user-centric orchestration of intuitive service functionality [Margaria and Steffen, 2009]. The jABC process modeling and execution framework [Steffen et al., 2007] is the current reference implementation of the XMDD paradigm.

The aim of this paper is to show how climate impact analysis processes implemented in the ci:grasp platform[1] are leveraged to a user-accessible level by the agile workflow development style supported by the jABC, enabling users to easily define and perform multi-objective workflows tailored to their specific needs. Section 2 gives an overview about the ci:grasp platform and the SRTM package. Section 3 gives a brief introduction to the jABC framework. Section 4 discusses service integration and agile workflow design by means of an example scenario that addresses the assessment of climate impacts due to sea-level rise. Finally, Section 5 discusses conclusions and future work.

2 THE CI:GRASP PLATFORM AND THE SRTM PACKAGE

The ci:grasp project is a web-based climate information service. It aims to support decision makers to better understand impacts from climate change, to prioritize adaptation needs, and to plan and implement appropriate adaptation measures [Wrobel et al., 2013; Wrobel and Reusser, 2014]. It models climate information and related knowledge in the form of text, maps, and graphs. Main drivers for climate change included on ci:grasp are sea level rise (SLR), change in temperature and precipitation and increased drought risk. These drivers, as well as more frequent extreme events expected under a changing climate, provide a high risk for human lives and cause massive economic damages. Figure 1 gives a short overview of the information presented on ci:grasp.

Here, we will use SLR related impacts as one example to demonstrate the power of agile workflow development as a way to easily define and perform multi-objective workflows in a tailored mannor. For SLR, the first step is identifying vulnerable areas in order to locate potential need for prevention measures. For the efficient identification of potentially flooded areas, a package [Kriewald, 2013] for the data analysis language R [R Core Team, 2014] provides processing of SRTM data [Jarvis et al., 2008] as a basis for results presented on ci:grasp. Ci:grasp then allows users to adjust parameters and explore results for coastal regions over the world in an interactive viewer.
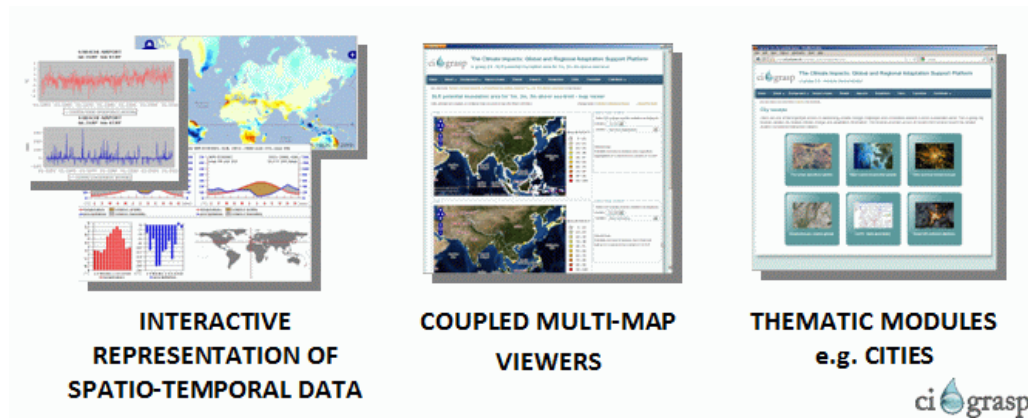
---

[1]http://www.cigrasp.org

Figure 1. Overview of the information available on ci:grasp

3 THE jABC MODELING FRAMEWORK

The jABC [Steffen et al., 2007] is a multi-purpose and domain-independent framework for model-driven application development. It inherits the power of eXtreme Model-Driven Design (XMDD) [Kubczak et al., 2009; Margaria and Steffen, 2012] to enable end users easily to use and compose services into agile workflows. It provides them with a comprehensive and intuitive graphical user interface that facilitates workflow development. jABC users easily develop workflow applications by composing reusable building-blocks (called Service-Independent Building Blocks, or SIBs) into hierarchical (flow-) graph structures (called Service Logic Graphs, or SLGs) that are executable models of the application.

The workflow development process is furthermore supported by an extensible set of plugins providing additional functionalities, so that the SLGs can be analyzed, verified, executed, and compiled directly in the jABC. This way of handling the collaborative design of complex software systems has proven to be effective and adequate for the co-operation of non-programmers and technical people. Recently the framework has also been extended by functionality for semantics-based semi-automatic service composition [Lamprecht et al., 2010; Naujokat et al., 2012], which has been shown to be beneficial especially for dealing with variant-rich scientific workflows [Lamprecht, 2013].

4 EXAMPLE: SLR WORKFLOWS

This section deals with the combination of ci:grasp (or more specifically the SRTM tools) and the jABC to achieve an intuitive and flexible handling of the underlying computational processes. In Section 4.1 we explain how we turned the original srtmtools package into a collection of services that can be used as workflow building blocks in the jABC. In Section 4.2 we then sketch some workflow variations of the sea-level rise (SLR) impact assessment that can easily be adapted by users in order to follow their specific objectives.

4.1 Servification of the SLR climate impact assessment tools

The service orientation paradigm postulates that any kind of computational resource should be seen and handled as a service – that is, a well-defined unit of functionality with a well-defined interface – to provide a high level of abstraction and reusability (cf.,

e.g., [Margaria, 2007]). Of course, this postulate also holds for the complex processes and applications that have been developed for climate impact assessment. With the term *servification* we refer to the process of turning arbitrary software components into proper services that are adequate, for example, for (re-) use in workflow management systems.

For the servification of the SLR impacts assessment tools described in Section 2, the various existing R scripts, which have been used to produce the data available on the ci:grasp platform, have been decomposed into separate and independent functions and equipped with with well-defined inputs and outputs in order to provide proper services adequate for the envisaged application. Technically, these services are currently simply provided in the form of autonomously running R scripts, so that they can easily be encapsulated into SIBs and be used as workflow building blocks within the jABC. Concretely, 16 services have been created until now, which can be categorized in four groups:

- Data loading (of elevation data, land use data, caloric data and population data)

- Computation (of flooded areas, yield loss, caloric energy loss and land loss classes)

- Resampling (of land use data with flooded areas, population data with flooded areas, and yield data with land use data)

- Output generation (creation of PNG, PDF, TXT, GeoTiff/ASCII output files and result visualization in an interactive map)

In the jABC, the SIBs accessing these specifically developed impact assessment services can then be combined with already existing SIBs, which provide additional functionality for, e.g., data input/output, for evaluating conditions and for basic user interaction. Based on this large library of services, the SLR impact assessment was reconstructed in a flexible workflow-based way. This is discussed in the next section.

4.2 Agile workflow design

The example of SLR impact assessment that we detail in the following addresses effects on livelihood and society based on different datasets. The agile workflow development style allows to flexibly consider multiple objectives and different user's preferences regarding inputs and outputs.

Figure 2 (center) shows a simple workflow for assessing the impact of sea-level rise on the agricultural yield loss for a region to be selected by the user. From left to right (the SIB with the underlined name denotes the starting point), it performs (1) selection of the working directory for input and output data; (2) definition of the investigated area by coordinates of name; (3) downloading the digital elevation model of the selected area; (4) entering of the magnitude of sea level rise; (5) computation of the flooded area; (6) computation of the yield loss due to the flooding; and (7) generation of an output file with results in PDF format. Some of the SIBs are marked by a green circle, which indicates that the functionality represented by this building block is actually more complex and defined by a separate (sub-) model. For example, SIB (3) encapsulates a (sub-) model for the selection and loading of elevation data (shown at the top of the figure). As another example (shown at the bottom of the figure), the SIB (6) is a composite service that allows for the computation of several types of yield loss for different climate scenarios. Note that it again makes use of other (sub-) models. This hierarchical modeling style allows to organize workflow applications in different levels of abstraction, from coarse-granular and more conceptual views at the higher levels, down to fine-granular and more technical views at the lower levels.
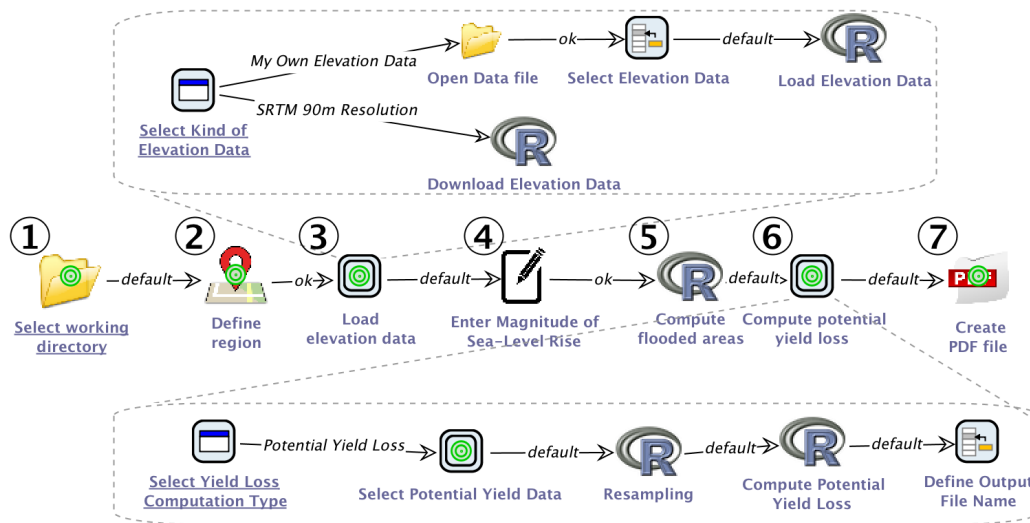
Figure 2. Workflow for SLR impact assessment regarding yield loss.

Table 1. Objectives to assess SLR impacts.

SIB	Description
compute rural and urban GDP at risk	focuses on potential economic damage in coastal communities
compute population at risk of migration	focuses on the number of people that would be affected
compute potential yield loss	compute potential production value affected in USD
compute potential land loss (ha)	determine the area that will be potentially inundated
compute potential production affected ($)	focuses on the economic value of the agricultural loss
compute potential caloric energy loss	focuses on the potential number of peoples annual diets lost

Figure 3 shows an extended workflow for SLR impact assessment with preconfigured variation points, which make it easy for the user to build variants of the basic workflow described above: he/she just needs to change the execution path to including another of the options from the variation points, simply by dragging the connecting branches to other SIBs. Variation point 1 allows for the easy modification of the way the considered region is defined. Instead of directly entering the coordinates, it is also possible to enter the name of a place or an address that is then used as the center of the region, and to select a region interactively from a map. Variation Point 2 comprises a collection of composite SIBs performing different computations, according to different objectives to assess SLR impacts, as summarized in Table 1. As has been sketched in Figure 2, all these computations are in fact realized by separate workflow models. Variation Point 3 contains SIBs that are responsible for creating different output formats. Users can easily select one or more formats for the output of the final results, choosing from (a) static maps in different formats (jpeg,pdf,png,ps), (b) an interactive map using the Google Maps API[2], (c) a text-file containing some summary and statistic information and

---

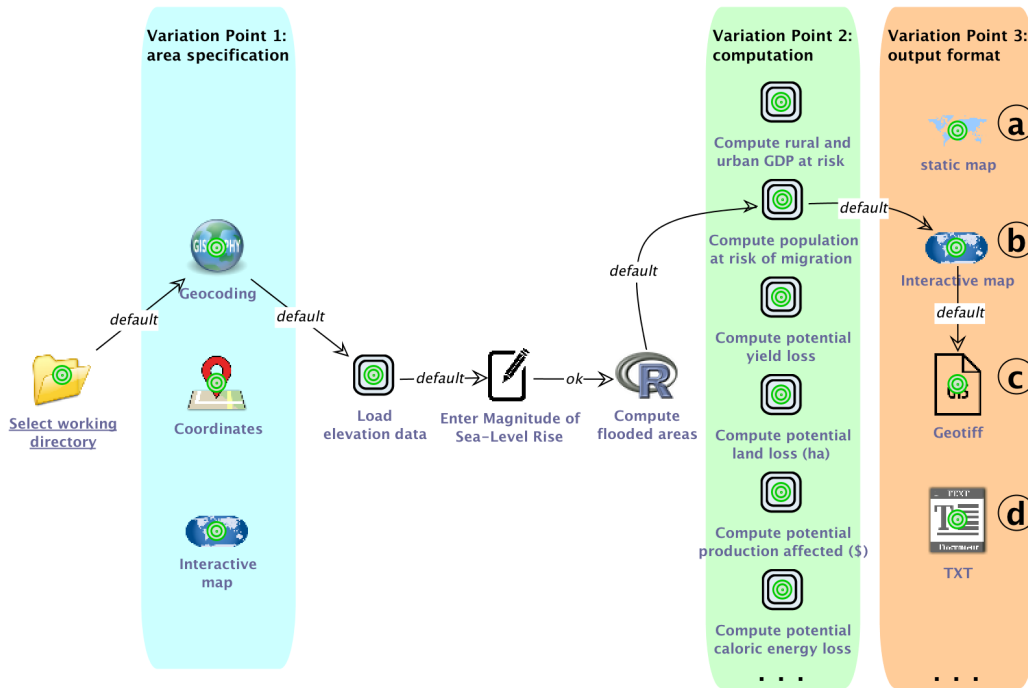[2]https://developers.google.com/maps/

Figure 3. Workflow for SLR impact assessment with preconfigured variation points.

(d) a geo-referenced file (GeoTiff, ASCII) which can be used for further external GIS processing. Through these variation points, adjustment of the assessment is accessible at the user level, even without classical programming skills.

Due to the limited space in this paper, we only demonstrate a few of the many possible workflow variations here. The full project provides a comprehensive agile and flexible library of directly executable workflows, currently comprising around 20 models composed of more than 180 SIBs and spanning three hierarchy levels.With this library workflows for various kinds of SLR analyses can be easily and flexibly created according to the user's objectives and preferences.

5 CONCLUSION

We showed how the agile workflow development style supported by the jABC process modeling and execution framework permits us to leverage the processes implemented in the ci:grasp platform to a user-accessible level, and thus to enable users to easily define and perform multi-objective workflows tailored to their specific needs. This allows us to use the jABC framework to address the complexity challenges of climate impact risk assessment discussed earlier. The described level of agility is achieved essentially by:

- rigorous *service orientation*, turning basic components as well as their compositions into flexibly reusable pieces of functionality,

- *hierarchical modeling* that allows to represent processes on different levels of abstraction, ranging from completely user-accessible top-level workflows that hide all technical details to fine-granular service compositions on the lower levels, and

- an *intuitive graphical interface* that makes it easy also for non-programmers to design and adapt workflows according to their specific preferences and constraints.

For the exemplary scenario of assessing the impact of sea-level rise, we discussed how we decomposed the ci:grasp's original implementation into basic computational services, which we then used as workflow building blocks. Together with additional functionality from the jABC, we built a flexible library of directly executable workflows for this type of analysis. We are going to provide it to the impact assessment community in two ways finally: First, as full jABC workflow projects that are especially suitable for increasing the productivity of researchers working on the analysis processes. Second, as preconfigured versions of individual workflows that are made available via the ci:grasp website, allowing web site users to easily repeat the impact assessment with own locally available data In a next step, other impact assessment models than the one applied in ci:grasp may go through a similar servification process, extending the library with additional and alternative services. Moreover, more flexible inclusion of various, heterogeneous data sources can be achieved with additional SIBs.

Furthermore, it will be beneficial to improve the multi-objective and multi-scale risk assessment by the use of semantics-based workflow design methodology, similar to the work described by [Lamprecht, 2013]. As the successful application of such methodologies crucially depends on adequate domain modeling, a major part of our future work will focus on the application and design of domain-specific ontologies. Once a semantic-based workflow design is available, the integration of the computational part of an impact assessment into a broader adaptation cycle will be possible.

REFERENCES

Beck, K., M. Beedle, A. van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. C. Martin, S. Mellor, K. Schwaber, J. Sutherland, and D. Thomas. Manifesto for Agile Software Development. `http://agilemanifesto.org`, 2001. [Online; last accessed 4-March-2014].

Boettle, M., D. Rybski, and J. P. Kropp. How changing sea level extremes and protection measures alter coastal flood damages. *Water Resour. Res.*, 49(3):1199–1210, 2013.

IPCC. Summary for Policymakers. In Parry, M., O. Canziani, J. Palutikof, P. van der Linden, and C. Hanson, editors, *Climate Change 2007: Impacts, Adaptation and Vulnerability. Contribution of Working Group II to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change*, pages 7–22. Cambridge University Press, Cambridge UK, 2007.

Jarvis, A., H. Reuter, A. Nelson, and E. Guevara. *Hole-filled SRTM for the globe Version 4*, 2008. [Online; last accessed 7-March-2014].

Kriewald, S. *srtmtools: SRTM tools*, 2013. R package version 2013-00.0.1.

Kubczak, C., S. Jörges, T. Margaria, and B. Steffen. eXtreme Model-Driven Design with jABC. In *CTIT Proc. of the Tools and Consultancy Track of the Fifth European Conference on Model-Driven Architecture Foundations and Applications (ECMDA-FA)*, volume WP09-12, pages 78–99, 2009.

Lamprecht, A.-L. *User-Level Workflow Design - A Bioinformatics Perspective*, volume 8311 of *Lecture Notes in Computer Science*. Springer, 2013.

Lamprecht, A.-L., S. Naujokat, T. Margaria, and B. Steffen. Synthesis-Based Loose Programming. In *Proc. of the 7th Int. Conf. on the Quality of Information and Communications Technology (QUATIC 2010), Porto, Portugal*, pages 262–267, September 2010.

Lissner, T. K., D. E. Reusser, J. Schewe, and T. Lakes. Linking human well-being and livelihoods with climate change impacts: contextualizing uncertainty in projections of water availability. *HESSD*, page inPrep, 2014.

Margaria, T. Service is in the Eyes of the Beholder. *IEEE Computer*, November 2007.

Margaria, T. and B. Steffen. Agile IT: Thinking in User-Centric Models. In Margaria, T. and B. Steffen, editors, *Leveraging Applications of Formal Methods, Verification and Validation*, volume 17 of *Communications in Computer and Information Science*, pages 490–502. Springer Berlin / Heidelberg, 2009.

Margaria, T. and B. Steffen. Service-Orientation: Conquering Complexity with XMDD. In Hinchey, M. and L. Coyle, editors, *Conquering Complexity*, pages 217–236. Springer London, 2012.

Naujokat, S., A.-L. Lamprecht, and B. Steffen. Loose Programming with PROPHETS. In de Lara, J. and A. Zisman, editors, *Proc. of the 15th Int. Conf. on Fundamental Approaches to Software Engineering (FASE 2012), Tallinn, Estonia*, volume 7212 of *LNCS*, pages 94–98. Springer Heidelberg, 2012.

Pradhan, P., M. Lüdeke, D. E. Reusser, and J. P. Kropp. Food Self-Sufficiency across scales: How local can we go? *Environmental Science and Technology*, page under review, 2014.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2014. [Online; last accessed 7-March-2014].

Steffen, B., T. Margaria, R. Nagel, S. Jörges, and C. Kubczak. Model-Driven Development with the jABC. In Bin, E., A. Ziv, and S. Ur, editors, *Hardware and Software, Verification and Testing*, volume 4383 of *Lecture Notes in Computer Science*, pages 92–108. Springer Berlin / Heidelberg, 2007.

Warszawski, L., K. Frieler, V. Huber, F. Piontek, O. Serdeczny, and J. Schewe. The inter-sectoral impact model intercomparison project (isi-mip): Project framework. *Proceedings of the National Academy of Sciences*, 111(9):3228–3232, 2014.

Wrobel, M., A. Bisaro, D. Reusser, and J. Kropp. Novel approaches for web-based access to climate change adaptation information – mediation adaptation platform and ci:grasp-2. In Hřebíček, J., G. Schimak, M. Kubásek, and A. Rizzoli, editors, *Environmental Software Systems. Fostering Information Sharing*, volume 413 of *IFIP Advances in Information and Communication Technology*, pages 489–499. Springer Berlin Heidelberg, 2013.

Wrobel, M. and D. Reusser. Towards an Interactive Visual Understanding of Climate Change Findings on the Net: Promises and Challenges. In Schneider, B. and T. Nocke, editors, *Image Politics of Climate Change*, pages p187–210. Transcript, London, 2014.