AUTOMATIZATION OF A PROCESS OF AN INUNDATION AREA COMPUTATION

Pavel Veselý [(a)], Štěpán Kuchař [(b)], Jan Martinovič [(c)], Vít Vondrák [(d)]

[(a)]) VŠB - Technical University of Ostrava
Faculty of Electrical Engineering and Computer Science
17. listopadu 15/2172, 708 33 Ostrava, Czech Republic

[(b) (c) (d)] VŠB - Technical University of Ostrava
IT4Innovations
17. listopadu 15/2172, 708 33 Ostrava, Czech Republic

[(a)] pavel.vesely@vsb.cz, [(b)] stepan.kuchar@vsb.cz, [(c)] jan.martinovic@vsb.cz, [(d)] vit.vondrak@vsb.cz

ABSTRACT

A FLOREON+ (FLOods REcognition on the Net) system for floods prediction and inundation areas computation has been developed to help with an operative disaster management and decision support. This article presents automated process of inundation areas computation based on open standards, unified process description and standardized internal component communication. It describes a case study and experimental use of business process description language such as BPEL in geoinformatic environment.

The system has been designed based on a service-oriented architecture (SOA). It contains a set of independent web services, orchestrated by business process interpreter application. External access to the process is clearly standardized implementing OGC WMS/WPS specifications, as well as communication between web services.

1. INTRODUCTION

FLOREON+ (FLOods REcognition on the Net) system (Vondrák, Martinovič, Kožusznik, Štolfa, Kozubek, Kubíček, Vondrák and Unucka 2008; Martinovič, Štolfa, Kožusnik, Unucka and Vodnrák 2008; Unucka, Martinovič, Vondrák, and Rapant 2009) for floods prediction and inundation areas computation has been developed to help with an operative disaster management and decision support. Main goal of our work is to develop versatile methodology for automation of FLOREON+ processes. It has to include system design, usage of language for business process description and execution, communication standards and selection of basic technology platform (Enterprise Service Bus or a similar framework for the process execution). Independency of system's components must be preserved to achieve maximum flexibility of the design and make the components reusable for different processes.

2. RELATED WORK

One of the first works, that shows GIS as several independent cooperating services, is stated in (Alameh, 2003). Implementation of SOA in GIS is described by several projects. They are mostly created as an implementation of INSPIRE architecture, such as in (Friis-Christensen, Bernard, Kanellopoulos, Nogueras-Iso, Peedell, Schade and Thorne, C. 2006). It shows SOA in a prototype of GIS for forest fire assessment. The prototype demonstrates solution of real problems using rapid development of a distributed application, which is facilitated by Spatial Data Infrastructure (SDI) as a basis for distributed service oriented geoprocessing.

An article by Andreja Jonoski (Jonoski, 2012) describes current trends in the field of hydroinformatics. Standardization of communication between web applications is a basic challenge. Therefore several common used formats exist for web services communications and spatial data exchange, especially Open Geospatial Consortium (OGC) formats (WMS, WFS, GML) and its derivates, such as WaterML. The next topic of the current trends is Spatial Data Infrastructure (SDI) data sources. The most important SDIs are National Spatial Data Infrastructure (NSDI) in USA, INSPIRE in EU or United Nation's UNSDI. Using standards enable sharing datasources and making distributed calculations. The article provides an example of a web based application DIANE-CM. It benefits from cooperation of independent web services and spatial data sources for flood risk management.

Using geospatial services in environmental systems describes (Granell, Díaz, Gould, 2010) using SOA applied to alpine runoff models. The application includes usage of geospatial services facilitating discovery, access, processing and visualization of geospatial data in a distributed manner.

Communication between geospatial services must be standardized and unified. Project MEDSI (Rocha, Cestnik and Oliveira 2005) shows example of implementation of OGC standards in a crisis management GIS. The project verified implementation of a prototype based on OGC standards. The implementation was deployed on several software platforms, including open source tools UMN MapServer, GeoServer, as well as commercial Geomedia Web Server and ESRI ArcIMS. The prototype was completed by catalogue services in accordance with a standard Catalogue Service for Web (CSV). It allows searching datasources by its metainformation.

Next work (Castronova, Goodall, Elag, 2012) advances an idea of interoperability between multiple independent spatial web services by presenting a design for a web service, which is build in accordance to the OGC Web Processing Service (WPS) protocol. The WPS is used as an interface between hydromodeler environment based on Open Modelling Interface (OpenMI) standard and client-side workflow.

One of the big challenges in environmental GIS using SDI is integration of multiple geodata sets and takeover geodata from different datasources. (Foerster, Lehto, Sarjakoski, Sarjakoski, Stoter, 2009) presents a process for geodata generalization and schema transformation in a web service's architecture to achieve interoperability between different geodata.

The final step to make our computation process fully automated is an orchestration of web services. Research project Orchestration Services for GeoWeb (Prager, Klímek, Růžička, 2009) describes standards in a field of service's cooperation, such as Business Process Execution Language (BPEL), Ontology Web Language for Services (OWL-S) or XML Process Definition Language (XPDL). It shows practical implementation of BPEL using Oracle BPEL process management and cooperation of geoservices in a workflow on Oracle's Business Process Management platform.

Calling geoservices in a workflow we have to take into account asynchronous communication (Zhao, Peisheng, Di, Liping, Yu, Geong, 2012). Asynchronous workflow can be distinguished at calling asynchronous web services in a workflow or calling a workflow as an asynchronous service. The article (Zhao, Peisheng, Di, Liping, Yu, Geong, 2012) describes several case studies. They have some common features: calling asynchronous services, using OGC standards (WPS, WCS, GML) and using WS-BPEL to describe workflow itself.

Our work shows an application of standards and ESB technologies in a real-world geoinformation system. The work links geoinformation services (in accordance to OGC standards) to a workflow in an open-source ESB platform OpenESB. We concentrate on practical usage of ESB technologies such as BPEL in hydrologic simulations. As opposed to the projects we mentioned before we add ESB and workflow technologies to GIS. Our application shows an importance of the transformation of geodata, when the data are moved between the hydrological models. The transformation to common standards is very important to achieve independence of the web services, which gains an access to the models.

3. METHODOLOGY

The first step is to describe the process of flood prediction using its graphical representation in Business Process Model and Notation (BPMN) (Object Management Group 2008) (see Figure 1). This human-readable form shows an overview of execution flows and data transmission between components of the system.

In order to execute actions within business processes with web services, the process has to be described using some process execution language as Web Services Business Process Execution Language (WSBPEL) (OASIS 2007) or similar language such as Microsoft's Workflow Foundation XAML (Microsoft MSDN 2013). Transformation from BPML to the execution language depends on the used. It is described later in this paper.

Each web service involved in the process must be described by Web Services Description Language (WSDL) (W3C, 2001). This is an XML-based interface description language that is used for describing the functionality offered by a web service. The WSDL document for a web service of the process is used in combination with XML schema description of a Web Processing Service (WPS) standard.

The most important goal is to standardize external interfaces of the process and internal service communication. To ensure independence of each component inside the process and its integration with other processes, common communication standards have to be used. Several specialized standards for exchange of hydrologic data or communication with web services exist (Vitolo, Buytaert, Reuser 2012). Hydrologic data can be stored in a common geoformat GML/KML or its specialized derivates WaterML and UncertML. Standards OGC WPS, WMS or WFS (Open GIS Consortium 1998) can be used to assemble a query to a geoinformation web service, including a spatial query. Web Processing Service (WPS) standard is used to execute a spatial function on a web service. Caller of the process can request a visualized map and set its parameters using the Web Map Service (WMS) standard.

4. SOFTWARE ENVIROMENT

Open Enterprise Service Bus (OpenESB) tool have been selected as a platform for creating workflow and executing BPEL (OpenESB, 2012). This tool has several advantages:

- Open source software with strong community support without any additional costs,
- Comfortable and powerful user interface based on NetBeans IDE tool for rapid development of services and workflows,

- Because the platform is Java-based, the solution can be deployed to different operating systems without changes.
- Solutions created on OpenESB platform are reliable, strong and scalable, appropriate to create crisis management applications.

Final application services were deployed on Oracle GlassFish server. It is a reference implementation of Java EE platform sponsored by Oracle Corporation. The OpenESB is strongly integrated with GlassFish and deployment of the services is completely automated.

5. BUILDING THE PROCESS

The process goes through several tasks during its execution:

- Data transformation.
- Rainfall-Runoff simulation.
- Hydrodynamic simulation.
- Data transfer.

Figure 1 shows positions of each task in the process. Some of them, such as data transformation, are executed repeatedly in different places and with different input/output arguments.

Before each step, data should be transformed into a format that is used by the processing component/model. Data in the process has spatial information, so data transformation is not changing only the format of the data inside storage, but a conversion of spatial coordinates may be needed, such as conversion between different spatial projection systems. Results are converted to some of the standardized or commonly used formats (GML, Shapefile) or into a geodatabase storage using a geointerface, such as PostGIS.

Rainfall-Runoff simulation component involves four rainfall-runoff models: HEC-HMS (HEC-USACE 2010), HYDROG (HySoft 2010), MikeSHE (DHI 2011a) and our own in-house model called Math1D. All of them can be accessed using one web service. Caller determines the model to perform the simulation by a specialized parameters set in the input of the process. Different models need different data input formats and so the data conversion has to be executed before passing inputs into the selected model.

Results of the Math1D model can be statistically evaluated by modelling the uncertainty of its input parameters. The Monte-Carlo simulation method is used for estimating possible river discharge volumes based on the uncertainty of precipitation and meteorology forecast and provides several confidence intervals that can support the decisions in the operational disaster management (Kuchař, Kocyan, Praks, Litschmannová, Martinovič, Vondrák 2012).

The hydrodynamic modelling phase is executed after the rainfall-runoff results are acquired from the models in the first step. HEC-RAS (HEC-USACE 2010) and MIKE 11 (DHI 2011b) hydrodynamic models are used in the FLOREON+ system. Selection of the computation model is made during the process execution based on the parameter set at the start of the process. This parameter also governs the data format for data conversion before execution of the selected model.

The main idea of data transfer is not to transmit all the data between web services, but send and retrieve only links to the data stored in a place that is accessible by both sides of the web service call. This kind of transitions is fully accepted by the OGC WPS standard.

Final data are converted to OGC GML format and stored in PostgreSQL database through PostGIS interface or visualized according to OGC WMS request and sent back to the caller.

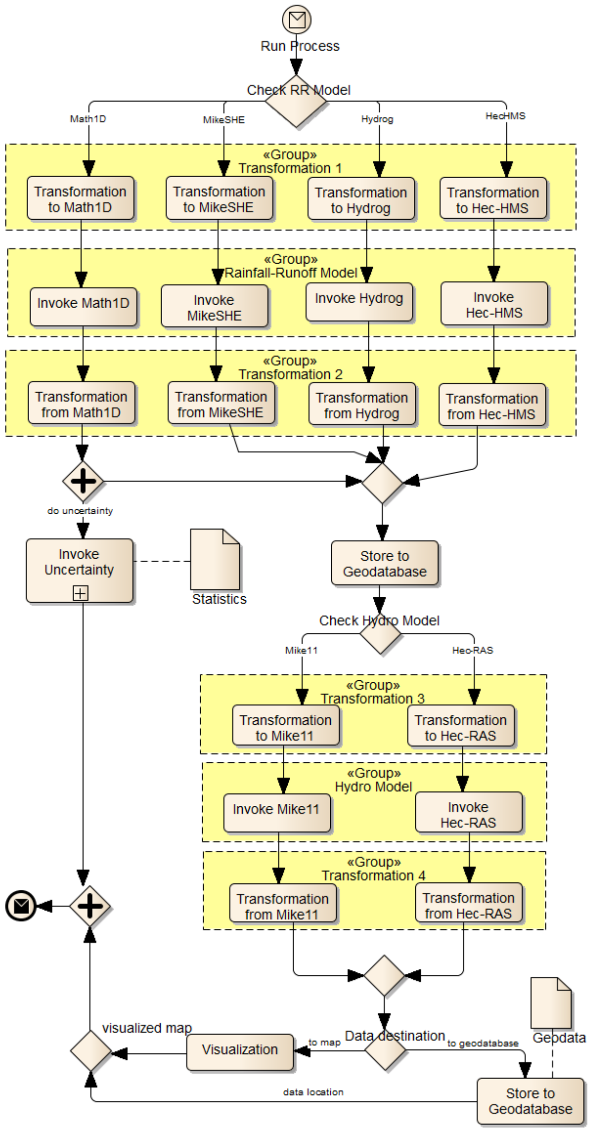

Figure 1: Process description using BPMN notation

6. CREATING WSDL FILES

Because the process will be published as a web service, it needs a Web Services Description Language (WSDL) document. The service will be standardized in accordance with OGC WPS standard. The format of the communication with the service is described in a schema descriptor (XSD) file on a web page http://schemas.opengis.net. This schema is set as a type in definition of the WSDL document. The WSDL itself was created using OpenESB's NetBeans IDE. WSDLs of the services of rainfall-runoff and hydrological models describe input parameters of the models and a format of their responses.

7. A USE OF THE WPS STANDARD

The WPS standard has been used, because it has two main advantages:

- It is supported by OGC, the leading international industry consortium in a field of geo-web industry standardization.
- It has universal design with ability to carry information with different types and structures.

In a BPEL process description we use "assign" BPEL element to map WPS XML nodes and attributes to model service's properties. The WPS contains "Input" nodes with identifier and data. A BPEL "predicate" node that consists XPath functions were created in BPEL process. A predicate applies a condition to a node that can have multiple values. Values of selected nodes are assigned to web service's parameters. Also some transformations from string type to number were done during assignment.

WPS request can contain an envelope with coordinates of an area of interest. This envelope is used to draw a final map with floodlakes.

8. PROCESS DESCRIPTION IN BPEL

Creating BPEL document on the basis of BPMN diagram depends on OpenESB tools for designing of the process. This is how BPMN objects correspond with OpenESB notation:

- BPMN start and end event are equal to receive and reply activity,
- BPMN activities are mostly created as invoke web service activity,
- BPMN connections are created as assign activity,
- BPMN gateways are equal to their appropriate structured activities.

We created BPEL document in NetBeans IDE. Than we added actions into the process, started from receive activity, web services invocation and assign activities, up to reply action. Partner links where created from receive and reply action using main service WSDL. It represents request/response of the web service of the process. Next partner links where created from each invocation activity to appropriate web services with service's WSDL file. Assign activity where set to connect input and output parameters of the web services and/or the receive/reply actions. We created simplified version of the process as a sample of the OpenESB BPEL diagram and for testing purposes. This simplified process shows Figure 2. A partner link to a WSDL of the process is on the left side of the figure. Next partner links on the right side shows connection to WSDL of the model services.

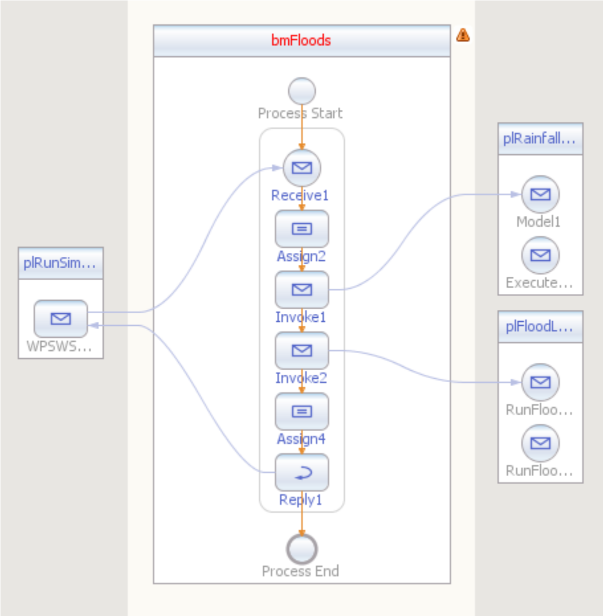

Figure 2: Simplified BPEL process

9. CREATING SERVICE ASSEMBLY

The service assembly (calling Composite Application in OpenESB speak) is a group of service units gathered together to create single application. The service assembly includes metadata for "wiring" the service units together (associating service providers and consumers), as well as wiring service units to external services. This provides a simple mechanism for performing composite application assembly using services (Java Community Process, 2013). We used NetBeans IDE provided with OpenESB to create composite application. Figure 3 shows diagram of connections from input request ports (HTTP and SOAP) to the BPEL process and from the process to external services of rainfall-runoff and hydrological models.

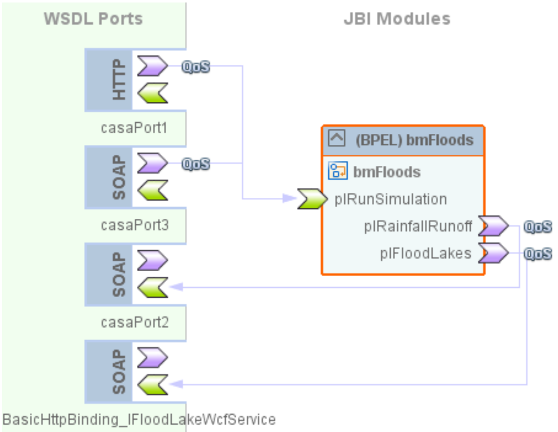

Figure 3: Diagram of composite application

10. EXPERIMENTAL RUN OF THE APPLICATION

We run application several times using a WPS request with basic parameters passed to the process. All parameters where passed using common WPS node "Input". Parameter's names where set in a subnode "Identifier" and values in a subnode "DataLiteral". First parameter "modelId" commands the process to use specific model. In our first test case it was HEC-HMS model for rainfall-runoff and hydrodynamics. Second test case we do the same with Math1D model. Parameter "basinSchemaId" sets preset schematization of the river, out tests used river Olše. "startTime" and "endTime" parameters set time period for data computation. Parameters from input request were transformed to rainfall-runoff and hydrodynamic web service's form during an execution of BPEL commands.

Table 1 shows computing times of rainfall-runoff service for different time periods. It shows that the computing time is very weakly dependent on the selected time period. Computing times in rainfall-runoff are much shorter than computing in hydrodynamic services. They are shown in table 2. Selected time period in a hydrodymanic significantly rises computing time up to several hours.

Table 1: Rainfall-runoff services run times

Rainfall-runoff services		
Time period	HEC-HMS	Math1D
48 hours	29.4 s	20 s
72 hours	29.7 s	21 s
96 hours	30.2 s	23.4 s

Table 2: Hydrodynamic services run times

Hydrodynamic services		
Time period	HEC-HMS	Math1D
6 hours	1597 s	1684.1 s
12 hours	2338.5 s	2651.1 s
24 hours	4325.2 s	4654 s

The whole process will be executed automatically in a selected time interval or casually by user's demands. Data sources of the models are refreshed by more precise values in a six hours interval. The same interval was set for automatic process execution. This execution can be made for a long time period 24 hour or even 48 hour. The time periods for casual execution of the process which will be run by user's request shouldn't be longer than 6 hours.

11. CONCLUSION

This paper describes automation of a computation of flood lakes. The process is built on enterprise service bus platform using BPEL to achieve flexibility and reliability. User or caller of the process can choose computational rainfall-runoff model and hydrodynamic model just by setting correct input parameters. Thereafter each step in the process is done automatically. The process is posted as a web service in accordance with OGC WPS standard. It allows integration of the service to complex geographic information systems in a further work. Each web service created by this work is reusable and can be accessed from other processes, which will be created in next work.

12. FUTURE WORK

We want to create deeply standardized and universal system. Each web service, which is applied in the system, can be in accordance with OGC standards. Data transformation wasn't fully tested and it can't be presented in the paper. In next work we will finish transformation services to make each part of the system fully independent.

Input data, which are used inside the models for computation, will be in a future automatically extracted from different data sources. This extraction can start the process of inundation areas computation or it can be a part of the process.

ACKNOWLEDGMENTS

This work was supported by the European Regional Development Fund in the IT4Innovations Centre of Excellence project (CZ.1.05/1.1.00/02.0070).

REFERENCES

Alameh, N., 2003, Chaining geographic information Web services, *IEEE Internet Computing*, vol. 7 (5): 22-29.

Castronova A. M., Goodall J. L., Elag M. M., 2012. Models as web services using the Open Geospatial Consortium (OGC) Web Processing Service (WPS) standard, *Environmental Modelling & Software*, pp. 72-83, Volume 41, March 2013.

DHI, 2011. *MIKE SHE - integrated catchment modelling*. Available from: http://www.dhisoftware.com/Products/WaterResources/MIKESHE.aspx [10th May 2013].

DHI, 2011. *MIKE 11 - river modelling unlimited*. Available from: http://www.dhisoftware.com/Products/WaterResources/MIKE11.aspx [10th May 2013].

Friis-Christensen, A., Bernard, L., Kanellopoulos, I., Nogueras-Iso, J., Peedell, S., Schade, S., & Thorne, C., 2006. Building service oriented applications on top of a spatial data infrastructure– a forest fire assessment example. *AGILE-Shaping the future of Geographic Information Science in Europe*, 2006, Visegrád, Hungary.

Foerster T., Lehto L., Sarjakoski T., Sarjakoski L. T., Stoter L., 2012, Map generalization and schema transformation of geospatial data combined in a Web Service context, *Computers, Environment and Urban Systems*, pp. 79-88, Volume 34, Issue 1, January 2010.

Granell C., Díaz L., Gould M., 2010, Service-oriented applications for environmental models: Reusable geospatial services, *Environmental Modelling & Software*, pp. 182-198, February 2010.

HEC-USACE, 2010. *Hydrologic Engineering Center – US Army Corps of Engineers*. Available from: http://www.hec.usace.army.mil/ [10th May 2013] .

HySoft, 2010. *HySoft.* Available from: http://www.hysoft.cz/ [10th May 2013].

Java Community Process, *JSR-000208 Java Business Integration*, 2013, Available from: http://jcp.org/aboutJava/communityprocess/final/jsr208/index.html [9. June 2013].

Jonoski, A., 2012, *Hydroinformatics and Decision Support: Current Technological Trends and Future Prospects*. BALWOIS 2012.

Kuchař, Š., Kocyan, T., Praks, P., Litschmannová, M., Martinovič, J., Vondrák, V., 2012. Simulation of Uncertainty in Rainfall-Runoff Models and their Statistical Evaluation in the Floreon System. *The 11th International Conference on Modeling and Applied Simulation 2012*, pp.128-133. 19-21 September, Vienna, Austria.

Martinovič, J., Štolfa, S., Kožusnik, J., Unucka, J., and Vodnrák, I., 2008, FLOREON - the system for an emergent flood prediction, *ECEC-FUBUTEC-EUROMEDIA*, Porto, Portugal.

Microsoft, MSDN, 2013. *XAML Overview (WPF).* Available from: http://msdn.microsoft.com/en-us/library/ms752059.aspx [10th May 2013].

Object Management Group, 2008. *Business Process Modeling Notation*, V1.1. Available from: http://www.omg.org/bpmn/Documents/BPMN_1-1_Specification.pdf [10th May 2013].

Open GIS Consortium, 1998. *OGC® Standards and Specifications.* Available from: http://www.opengis.org/public/abstract.html [10th May 2013].

OpenESB, 2012. *The Open Enterprise Service Bus.* Available from: http://www.open-esb.net [5th June 2013].

OASIS, 2007. *Web Services Business Process Execution Language Version 2.0*. Available from: http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html [10th May 2013].

Prager, M., Klímek, F., Růžička, J.. 2009, *GeoWeb Services Orchestration Based on BPEL or BPMN*, GIS Ostrava 2009.

Rocha, A., Cestnik, B., Oliveira, M. A., 2005. *Interoperable geographic information services to support crisis management*. Berlin Heidelberg: Springer.

Unucka, J., Martinovic, J., Vondrak, I., Rapant, P., & Brebbia, C. A., 2009. Overview of the complex and modular system FLOREON+ for hydrologic and environmental modeling. *In Proceedings of the 5th International Conference on River Basin Management*, Malta, 2009. pp. 207-216. WIT Press.

Vitolo, C., Buytaert, W., Reuser, D. E., 2012. Hydrological Models as Web Services: An Implementation using OGC Standards. *HIC- 10th International Conference on Hydroinformatics*, Hamburg, Germany.

Vondrák, I., Martinovič, J., Kožusznik, J., Štolfa, S., Kozubek, T., Kubíček, P., Vondrák, V. , Unucka, J., 2008. A description of a highly modular system for the emergent flood prediction, *Computer Information Systems and Industrial Management Applications, 2008, CISIM '08.* 7th.

W3C, 2001, Web Services Description Language (WSDL) 1.1. Available from: http://www.w3.org/TR/wsdl [4th June 2013].

Zhao, P., Di, L., & Yu, G., 2012. Building asynchronous geospatial processing workflows with web services. *Computers & Geosciences*, 39, 34-41.