Sharing Geoprocessing Workflows with Business Process Model and Notation (BPMN)

Xeni Kechagioglou
University of Cagliari
Via Marengo 2
09123 Cagliari (CA), Italy
xenikec@gmail.com

Rob Lemmens
University of Twente, ITC
Hengelosestraat 99
7514 AE Enschede, The Netherlands
r.l.g.lemmens@utwente.nl

Vasilios Retsios
University of Twente, ITC
Hengelosestraat 99
7514 AE Enschede, The Netherlands
v.retsios@utwente.nl

ABSTRACT
Graphical geoprocessing workflows are often built visually on interactive canvases of GIS software. Such workflows cannot be shared among different software, due to structural and semantical differences. This study experiments with a workflow created for ILWIS software and transforms it into a BPMN process model, exploiting XML serialisations of the two workflows. Ultimately, it aims at contributing to interoperability of geoprocessing workflows, through an extended approach serving as a frame around workflow conversion.

CCS Concepts
• Applied computing→Extensible Markup Language (XML) • Software and its engineering→Extensible Markup Language.

Keywords
Geoinformatics; Geoprocessing; Workflow; Interoperability; Business Process Model and Notation (BPMN); eXtensible Stylesheet Language Transformations (XSLT); ILWIS.

1. INTRODUCTION
A workflow is taken to be a representation of a stepwise process, with its logic and tasks designed to be understood and carried out by processing machines. Workflows that fully or partially entail geoprocessing operations can be considered geoprocessing workflows, in the same way that the terms have been used in the engineering report for the Open Geospatial Consortium Web Service testbed 6 [6]. Although definition does not exclude interaction and integration with other types of activities in a single workflow, this work considers processes comprising exclusively geoprocessing tasks.

Building workflows in a visual, graphical way is a particularly useful feature for process planners, the latter spanning from inexperienced users seeking simple, repetitive procedures to professionals designing complex flows of geodata processing. Such graphical icons on the software's interactive canvas usually come with rather intuitive semantics, which renders workflow design quite a straightforward operation and further facilitates exchange of knowledge among users of the same software. Standardisation of such graphical notations boosts interoperability between software packages that choose to use them, allowing for workflow transfer and correct process enacting from different platforms.

Geographical Information System (GIS) software often comes with a graphical geoprocessing workflow builder. Yet, graphical notations vary among software in terms of both constructs and semantics, a fact that blocks cross-platform transfer of workflow diagrams. The problem is equally valid whether one speaks of workflows as abstract templates or as concrete procedures with all their parameters resolved for a certain case at hand. If, however, a standardized notation suitable for geoprocessing flows were to be employed, the problematic situation could potentially be eased.

This study, aiming at contributing to the broader frame of geoprocessing workflow interoperability, experiments with graphical notation conversion from a GIS software specific to a standardised one, namely from ILWIS specific to Business Process Model and Notation (BPMN), taking advantage of their eXtensible Markup Language (XML) schemata for workflows. The objective is to explore meaningful feasibility of such a task and pave, in this way, the path to a more extensive investigation of the subject.

2. APPROACH
There are two aspects to consider when adapting a geoprocessing workflow from one software to another. The first one is conversion of the notation specific to a GIS software to an intermediate, standardised notation and back. This aims at mapping notation constructs and semantics, as well as flow logic, in a way that preserves their meaning and ultimately allows for correct execution of the process. It therefore appears imperative that the intermediate notation is rich in constructs and semantics and expressive enough in flow logic, to cover complex process modelling requirements adequately.

The second aspect to consider is detection of geo-operators, i.e., low-level tasks synthesizing a geoprocessing workflow, which are similar or even correspond exactly to those of the software used to create the original workflow. This feature presumes description of geo-operators in a way that links their attributes, for example their required inputs and outputs, their preconditions, their functionality, or the mathematical models they apply.

Our approach therefore starts with the creation of a workflow in the original GIS software and moves on to convert it to a process in a standardised notation. It then finds corresponding geo-operators in the target GIS software, using Semantic Web technologies that include a geo-operations ontology, Linked Data

ICGDA 2019, March 15–17, 2019, Prague, Czech Republic

ACM ISBN 978-1-4503-6245-0/19/03…$15.00
DOI: https://doi.org/10.1145/3318236.3318239

and SPARQL queries [9]. Finally, it adjusts the intermediate workflow model to the new geo-operators and translates the standardised notation to the notation of the target software. If the target software has APIs for its functions, then the solution may well be completed one step ahead of the final translation, provided that the intermediate notation can connect to the APIs and execute the process directly from its own management platform.

This paper deals with the transformation of geoprocessing workflows to a standardised notation, using BPMN as the intermediate notation and ILWIS as the original GIS software. Although the choices presented followingly set the frame for the implementation of the entire approach, the rest of the solution is work-in-progress.

2.1 Business Process Model and Notation

BPMN is a widely used notation for processes with at least some structure, initially specified for business process management, i.e., for standardising communication between business analysts, technical developers, and business people [7]. Its visual language, comprising elements, shapes/connections and markers, is easy to understand, while its richness allows for modelling from atomic tasks to complicated, cross-platform, multi-actor processes, with loops, events, communications and decisions.

A fundamental feature of BPMN is its platform neutrality. It means that one could potentially load a process described in XML to any BPMN editor and have it displayed diagrammatically. Version 2.0 of the notation incorporates BPEL, therefore tools that claim BPEL process execution conformance can further enact the modelled workflow by connecting to desktop applications or web services. This means that resolved geoprocessing workflows are potentially directly executable if the GIS software provides an API. Last, business process management suites working with BPMN may allow for cost allocation and process analysis for monitoring and eventual optimisation.

2.2 Integrated Land and Water Information System - ILWIS

ILWIS (http://52north.org/communities/ilwis) is a GIS and Remote Sensing open source software, used for geoinformation professional and training activities and serving as a platform for research projects [2]. The software, currently in its 4th version, comes with an interactive workflow builder, capable of supporting large and complex geoprocessing workflows [5]. The notation used includes boxes representing data and operations, and arrows indicating the flow between them. These visual workflows can be exported in JavaScript Object Notation (JSON) format, rendering them shareable and deployable through web applications [4].

An additional reason for ILWIS' suitability for this study is that its development team has already embarked on research for converting workflows from more abstract definitions to concrete, case-based ones [1]. This, together with ongoing work on describing and linking geo-operators from different sources, can provide the solution to adapting workflows to different GIS software. Considering also the fact that the software is currently being redesigned to include APIs, ILWIS emerges as an optimal platform to experiment on every step of the approach described above.

3. PREPARATORY PHASE

As mentioned in the introduction, this study takes advantage of the features offered by XML. XML is a text-based data representation format for exchange of data between applications, easy to read by humans and simple to parse by computers. It is therefore a good choice when aiming at increasing the interoperability of applications. JSON, on the other hand, is an equally good choice for interoperability and even easier to parse. The decision to work with XML was ultimately based not on advantages of one format or the other, but merely on the fact that BPMN is XML-defined, which means that all BPMN diagrams can be - and normally are in BPMN 2.0 compliant editors - serialised in XML.

To describe a workflow that will serve as the artifact to experiment with in the following phase (conversion to BPMN), we need information that defines the geo-operators of our GIS software, as well as information that describes their connection in a workflow. ILWIS, in its role as a research platform, does not include a unique schema for definition of both geo-operators and workflows at the moment, as no imperative need had arisen before this study. This does not affect its performance in professional and educational activities, but adds some extra work for schema definition in this preparatory phase.

An XML description of all ILWIS geo-operators was extracted with a Python script. From this output, we automatically derived an XML Schema Definition (XSD) using the Venetian Blind design pattern, which allows for both globally defined, reusable elements and types, as well as for local ones. The XSD was further inspected and corrections were made, particularly with regard to global and local elements.

The path to derive an XSD for the structure of geoprocessing workflows in ILWIS was slightly more complex. Starting with a sample workflow that was verified and implemented in a previous study [5], we translated its JSON file into XML. An automatic one to one translation is usually not completely accurate, because item types in one format do not necessarily have an equivalent in the other, so post-processing was performed to confirm that the derived XML file was meaningful. Subsequently, the same procedure as for the geo-operators above was followed, in order to get an XSD. Complexity of the sample workflow we used was sufficient to cover the basic structure of any simple workflow, but will have to be increased for later experiments to include splits and decisions in the flow.

Table 1. Proposed unique XSD for ILWIS

Elements	Types	Indicators
workflow	workflowType	xsd:sequence
operation	operationType	xsd:sequence
input	inputType	xsd:sequence
output	outputType	xsd:sequence
connection	connectionType	xsd:sequence
fromOperationID	xsd:byte	
toOperationID	xsd:byte	

Finally, a unique schema was proposed by aligning the two XSDs. Its most basic elements, adopted to proceed with the study, are shown in Table 1, along with their types and the XSD indicators where relevant. *Workflow*, *operation*, *input* and *output* define the components of a process, while *connection*, *fromOperationID* and *toOperationID* describe the flow of tasks. Attributes of the elements are not shown, but include an *id*, a *description*, a *resource* (in this study all values would be *ILWIS*), and a *name* for each.

4. BPMN AS A MEDIATOR

4.1 An Example Workflow to Work With

Using ILWIS' workflow builder and from the schema of Table 1, we created an example of a geo-processing workflow in its simplest possible form, to serve as the basis for exercising a conversion to BPMN.

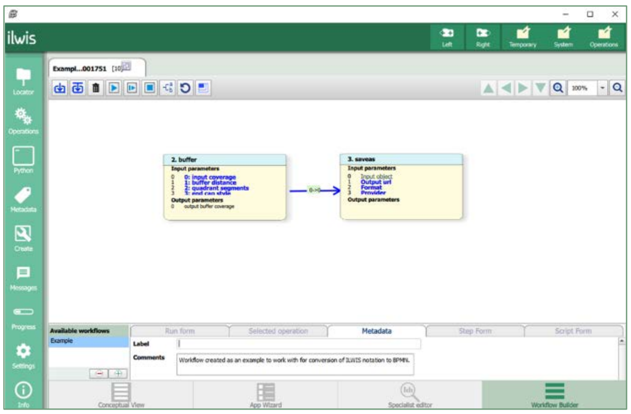

Figure 1. A simple workflow in ILWIS.

```
<workflow id="_1">
    <connections>
        <connection id="_6">
            <fromOperationID>_2</fromOperationID>
            <toOperationID>_3</toOperationID>
        </connection>
    </connections>
    <operations>
        <operation id="_2" name="buffer"/>
        <operation id="_3" name="saveas"/>
    </operations>
</workflow>
```

Figure 2. The example workflow in XML.

The workflow in graphical form is shown in Figure 1, while its XML serialisation is shown in Figure 2. It consists of two operators, each identified by an *id* number and a *name*, and a connection between them, indicating the direction of the flow. Input and output datasets are not considered at this stage.

4.2 Mapping ILWIS Elements to BPMN

BPMN 2.0 specification comes with a set of machine consumable documents, including five XSD files (Figure 3), downloadable from the OMG site (https://www.omg.org/spec/BPMN/2.0/). BPMN20.xsd is the top-level schema, while the process semantics are defined in Semantic.xsd and the definitions for the graphical presentation of the process diagrams are in BPMNDI.xsd. Last, DI.xsd and DC.xsd define diagram object elements like shapes and edges.

dtc/10-05-04	BPMN20.xsd
dtc/10-05-04	BPMNDI.xsd
dtc/10-05-04	DC.xsd
dtc/10-05-04	DI.xsd
dtc/10-05-04	Semantic.xsd

Figure 3. The XSDs for BPMN and their document numbers.

The XSD files, together with the BPMN specification document, were studied and a mapping between ILWIS and BPMN elements was created, as shown in Table 2. This will serve for converting ILWIS workflows into BPMN diagrams.

Table 2. Mapping of the elements in Table 1 to BPMN elements and types.

ILWIS elements	BPMN elements	BPMN types
workflow	process	tProcess
operation	scriptTask	tScriptTask
input	dataInput	tDataInput
output	dataOutput	tDataOutput
connection	sequenceFlow	tSequenceFlow
fromOperationID	incoming	xsd:QName
toOperationID	outgoing	xsd:QName

4.3 Structural Soundness of BPMN Diagrams

Producing BPMN workflow diagrams that are structurally sound cannot merely rely on mapping elements that satisfy another notation, but requires further input. To this end, we consider the three conditions of sound directed graphs as originated from Carl Adam Petri's work [8]:

1. The process model has exactly one start event *i*.
2. The process model has exactly one end event *o*.
3. Each node in the process model is on a path from *i* to *o*.

Noting that events are not included in ILWIS workflow notation, we added elements to the example XML to serve as placeholders for a start and an end event, together with connective flow elements to adhere to the very first and the very last operators of the workflow (Figure 4).

```
<startIDs>
    <firstID>_11</firstID>
    <secondID>_22</secondID>
</startIDs>
<endIDs>
    <firstID>_99</firstID>
    <secondID>_88</secondID>
</endIDs>
<flowIDs>
    <firstID>_112</firstID>
    <secondID>_223</secondID>
</flowIDs>
```

Figure 4. Elements added to ensure structural soundness.

```
<xsl:for-each select="/workflows/workflow/operations/operation">
    <xsl:text>&#xA;&#x9;&#x9;</xsl:text>
    <scriptTask>
        <xsl:attribute name ="id">
            <xsl:value-of select="@id"/>
        </xsl:attribute>
        <xsl:attribute name ="name">
            <xsl:value-of select="@name"/>
        </xsl:attribute>
        <xsl:choose>
            <xsl:when test="/workflows/workflow/connections/connection/toOperationID=@id">
                <xsl:text>&#xA;&#x9;&#x9;&#x9;</xsl:text>
                <incoming><xsl:value-of select="/workflows/workflow/connections/connection/@id"/></incoming>
            </xsl:when>
            <xsl:otherwise>
                <xsl:text>&#xA;&#x9;&#x9;&#x9;</xsl:text>
                <incoming></incoming>
            </xsl:otherwise>
        </xsl:choose>
        <xsl:choose>
            <xsl:when test="/workflows/workflow/connections/connection/fromOperationID=@id">
            <xsl:text>&#xA;&#x9;&#x9;&#x9;</xsl:text>
                <outgoing><xsl:value-of select="/workflows/workflow/connections/connection/@id"/></outgoing>
            </xsl:when>
            <xsl:otherwise>
            <xsl:text>&#xA;&#x9;&#x9;&#x9;</xsl:text>
                <outgoing></outgoing>
            </xsl:otherwise>
        </xsl:choose>
        <xsl:text>&#xA;&#x9;&#x9;</xsl:text>
    </scriptTask>
</xsl:for-each>
```

Figure 5. Excerpt of the XSLT file.

Unlike the *connection* element of Figure 2, id values in Figure 4 are random, properly because they are placeholders rather than hardcoded to the specific operators of this workflow. Resolution of flow connections to the very first operator for the start event and the very last operator for the end event takes place in the following step, through conversion rules.

4.4 Conversion Rules with XSLT

The eXtensible Stylesheet Language Transformations (XSLT) is a language for transforming XML documents, expressed in the form of a stylesheet that states the rules for conversion from one schema to another [3]. In our case, the stylesheet serves to convert the example ILWIS workflow with its XML elements and its placeholder-elements into XML elements and types for BPMN, both for process semantics and for diagram presentation. Figure 5 above shows an excerpt of the stylesheet that transforms every ILWIS *operation* element to a BPMN *scriptTask* element and assigns the attributes and connections of the first to the second.

4.5 Testing the Conversion

To test the stylesheet, we used the XML-to-XML converter Treebeard (https://sourceforge.net/projects/treebeard/). Using the ILWIS XML workflow as input and providing the XSLT file as conversion rules, Treebeard produced an XML file with BPMN elements. Figure 6 shows an excerpt of the produced file, including the two *scriptTasks* and the single *startEvent*.

The produced file can be loaded to a BPMN editor to further check the result diagrammatically. Theoretically, this could be done in any editor that is BPMN 2.0 compliant, but, in practice, we experienced that each one requires the XML file to follow the editor's style for *id* allocation. Yaoqiang-BPMN-Editor-5.3.12 (https://sourceforge.net/projects/bpmn/) simply requires that *id* numbers be preceded by an underscore, while it also provides for a very user-friendly interaction between the source (the XML encoding) and the diagram tabs.

```
<process id="_1" isClosed="false" isExecutable="true" processType="None">
    <scriptTask id="_2" name="buffer">
        <incoming/>
        <outgoing>_6</outgoing>
    </scriptTask>
    <scriptTask id="_3" name="saveas">
        <incoming>_6</incoming>
        <outgoing/>
    </scriptTask>
    <startEvent isInterrupting="true" id="_11">
        <outgoing>_2</outgoing>
        <outputSet/>
    </startEvent>
```

Figure 6. Excerpt of the produced XML file with BPMN elements.

The XML file produced from converting the ILWIS workflow through the stylesheet is a valid BPMN diagram (Figure 7), with two script tasks corresponding to the two geo-operators, one start and one end event, and flow of process correctly displayed among the constructs.

Normally, with regard to structural soundness, diagrams should be translated into a Petri Net language and have a relevant software perform a behavioural analysis of the modelled workflow to check whether the three conditions are met. However, the example ILWIS workflow prepared and used for this article is simple enough, so the BPMN diagram it produces may be evaluated and found sound merely by inspection.

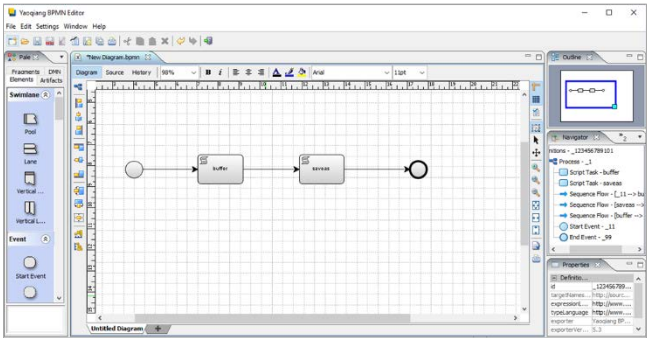

Figure 7. The BPMN diagram of the geoprocessing workflow, as displayed by Yaoqiang-BPMN-Editor.

5. CONCLUSION AND OUTLOOK

Visual design of geoprocessing workflows, although catered for in many GIS software and perceived as a very useful feature for users, does not allow for workflow exchange among software. In this study, we created a frame for an approach, which, starting with the creation of a workflow in one GIS software, may eventually produce a process that can be used by other software, too.

The article presented the first step of this approach, namely conversion of a geoprocessing workflow from software-specific to an intermediate, standardized notation, i.e., from a process implementable solely in ILWIS to one modeled in BPMN. The conversion started with the definition of an XSD schema for ILWIS, followed by mapping of its elements to BPMN. Conversion rules were coded in XSLT and the final result, namely a workflow described in XML, was a sound BPMN diagram that could be displayed visually in a BPMN editor.

Complexity of the example workflow used in this article was very low. The first step into future work is therefore the increase of sample workflow complexity, by including splits and decisions, as well as inputs and outputs. The final result may be tested by enacting the process directly from a BPMN platform that is BPEL process execution compliant. This would connect to ILWIS API to run the geo-operations.

Connecting this study with the rest of the approach aiming at increasing interoperability of GIS platforms, including the selection of corresponding geo-operators from other GIS software based on their descriptions and their substitution in the workflow, would complete the frame this current study forms part of.

6. REFERENCES

[1] De Carvalho Diniz, F. 2016. *Composition of semantically enabled geospatial web services*. Enschede, University of Twente Faculty of Geo-Information and Earth Observation (ITC).

[2] ILWIS Open Source GIS/RS software. 2016. Video introduction. Retrieved from https://vimeo.com/user29453510/review/153355429/1c1a97df84

[3] Kay, M. 2017. XSL Transformations (XSLT) Version 3.0. W3C Recommendation. W3C. Retrieved from https://www.w3.org/TR/2017/REC-xslt-30-20170608/

[4] Lemmens, R., Farifteh, J., Piccinini, C., Retsios, B., Schouwenburg, M. and Bonina, J. 2016. Visual and coded geoprocessing workflows based on ILWIS and Python. *Journal of GeoPython*, 1, 1, 21–24. Retrieved from http://www.geopython.net/pub/Journal_of_GeoPython_1_2016.pdf

[5] Lemmens, R., Toxopeus, B., Boerboom, L., Schouwenburg, M., Retsios, B., Nieuwenhuis, W. and Mannaerts, C. 2018. Implementation of a comprehensive and effective geoprocessing workflow environment. In *Proceedings of FOSS4G 2018, Academic Track, Volume XLII-4/W8* (Dar es Salaam, Tanzania, August 29–31, 2018). ISPRS 2018. International Society for Photogrammetry and Remote Sensing, 123-127.

[6] OGC. 2009. OGC 09-053r5, version 0.3.*0 OWS-6 Geoprocessing workflow architecture engineering report*. OGC. Retrieved from portal.opengeospatial.org/files/?artifact_id=34968

[7] OMG. 2011. formal/2011-01-03 *Business Process Model and Notation (BPMN) Version 2.0*. OMG. Retrieved from https://www.omg.org/spec/BPMN/2.0

[8] Petri, C.A., 1962. *Kommunikation mit automaten*. PhD thesis. Institut fur Instrumentelle Mathematik, Bonn.

[9] Ubels, S. 2018. *Understanding abstract geo-information workflows and converting them to executable workflows using Semantic Web technologies.* Enschede, University of Twente Faculty of Geo-Information and Earth Observation (ITC). Retrieved from https://dspace.library.uu.nl/handle/1874/365897