Accuracy 2010 Symposium, July 20-23, Leicester, UK

Handling and communicating uncertainty in chained geospatial web services

Richard Jones*, **Lucy Bastin**, **Dan Cornford**, and **Matthew Williams**
Knowledge Engineering Group
Aston University
Birmingham, United Kingdom
*jonesrm1@aston.ac.uk

Abstract—**Recent developments in service-oriented and distributed computing have created exciting opportunities for the integration of models in service chains to create the Model Web. This offers the potential for orchestrating web data and processing services, in complex chains; a flexible approach which exploits the increased access to products and tools, and the scalability offered by the Web. However, the uncertainty inherent in data and models must be quantified and communicated in an interoperable way, in order for its effects to be effectively assessed as errors propagate through complex automated model chains. We describe a proposed set of tools for handling, characterizing and communicating uncertainty in this context, and show how they can be used to 'uncertainty-enable' Web Services in a model chain. An example implementation is presented, which combines environmental and publicly-contributed data to produce estimates of sea-level air pressure, with estimates of uncertainty which incorporate the effects of model approximation as well as the uncertainty inherent in the observational and derived data.**

Keywords: uncertainty propagation, probability, web services

I. INTRODUCTION

Recent developments in service-oriented and distributed computing have created exciting opportunities for the exposure of computer models as Web Services. These models may be simple atomic operations such as a buffering process, or they may represent very concrete real-world systems. In the latter case, they are likely to require as inputs at least a few of the spatio-temporal datasets with which GIS researchers are familiar. These data sources bring with them a wealth of assumptions, uncertainties and errors, ranging from the limited reliability of amateur monitoring networks to specific calibration biases (known or inferred) on a satellite or dedicated in-situ sensor.

The handling and representation of such uncertainties is a well-studied and documented area (e.g., Burrough,1992; Hunter and Goodchild, 1993; Duckham et al., 2001; Couclelis, 2003, Heuvelink, 2007) and there are techniques and standards for recording uncertainties (albeit in simple form) as metadata which can be used in an interoperable context (for example, ISO/FDIS 19115 (2003). However, due to the specific and limited scope of existing metadata standards, GIS software that allows metadata to be recorded in compliance with those standards still requires users who want to practically work with uncertainty information to use various workarounds. For example, many off-the-shelf GIS software packages offer tools for generating metadata using ISO 19115 templates. However, the same packages do not characteristically offer tools for easy *use of* that metadata: for example options to generate randomised simulations of a spatial process based on numerical uncertainty metadata. In general, users wishing to deal with spatially-varying uncertainties (rather than, for example, a single root-mean-squared error value) must collate, register and combine the data layers themselves. This illustrates the value of an approach where the uncertainty of a dataset is treated as a component of the data, rather than separate metadata.

Data quality, as discussed above, is often recorded and quantified, though less frequently used for propagating uncertainties in a routine way. However, the uncertainties and assumptions of models themselves are far less commonly elicited and quantified, and there are currently no uniform standards by which a model published as a Web Service can publish its associated uncertainties in a way that can be automatically interpreted and re-used. This has serious implications for the integration of Web-based models in service chains and other linked configurations, to create the 'Model Web' The Model Web approach offers many advantages over tightly coupled and integrated systems, including flexibility, increased access to products and tools, and the scalability successfully demonstrated by the Web. For these reasons, the Model Web is likely to become an important arena for analysis and modelling for policy and planning purposes, as widely-dispersed datasets are located and combined in increasingly complex ways, to gain insights into real world processes and phenomena. Model Web outputs are likely to be increasingly used for practical and critical decision-making, in policymaking, resource prioritisation or emergency response.

Models of real systems inevitably either involve some degree of approximation, or include processes that are effectively random. Model inputs are rarely, if ever, free of error, and the assumptions inherent in models themselves introduce further uncertainty. As limited knowledge and intrinsic inaccuracies interact, model outputs (however accurate the inputs) become subject to errors, and decision-makers are faced with the challenge of making critical judgements from these uncertain outputs. The issue of uncertainty propagation is even more pressing when one considers the mechanisms by which service chains are composed, since component nodes can potentially be discovered and consumed entirely automatically. Without a

---

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n° [248488], UncertWeb.

mechanism for quantifying, managing and communicating uncertainty at all stages in a service chain, nothing can be said quantitatively about the value of the information contained in that chain's outputs.

This paper demonstrates the potential for achieving accountable, automatic uncertainty quantification by providing interoperable standards for representing and communicating error and adding support for uncertain inputs and outputs to Model Web components.

II. A SIMPLE PROTOTYPE

To demonstrate the potential for handling and communicating uncertainty in chained geospatial Web Services, a simple uncertainty-enabled Model Web has been created. Web Services allow us to use standard components of the World Wide Web to do complicated analyses. For example, an interpolation algorithm can be exposed as a standard Web Service, making it accessible to anyone connected to the Web with the right credentials. The standard components include the HTTP transfer protocol, which has the GET operation for requesting data from a service, and the POST operation for sending data to a service.

The prototype Model Web consists of a number of uncertainty-enabled Web Services, from which a chain can be constructed to correct a weather station level pressure measurement to sea level (see Fig. 1). Each component of this simple Model Web is able to handle uncertainty which has been quantified and encoded according to an XML standard. This enables propagation of uncertainty through the chain, and produces a realistic estimate of the reliability of the final model outputs. The use of individual interoperable components allows the component services to be reused.

The sequence firstly harvests user-contributed air pressure measurements for the UK from the Weather Underground sensor network. For each of these measurement locations, surrounding elevation values are retrieved from a service noting the supplier's estimate of errors on the elevation data. These values are passed to an interpolation Web service which estimates the elevation (and associated error) for the location in question, taking into account the noted elevation error. Finally, another Web service is used to correct the air pressure measurement to pressure at sea level using the predicted elevation, and uncertainty. The model chain makes use of a variety of representations of uncertain values (e.g., simple statistics such as variance, and samples from a probability density function) to produce a summary of the distribution of possible output values that encapsulates intervening sources of uncertainty.

III. IMPLEMENTING THE PROTOTYPE

A. *Use of Standards*

Each component in the Model Web needs to communicate in a similar, recognised manner in order to enable interoperability. Without a shared communication language, unique client interfaces must be written for each Web service that a user wishes to consume. Adding several services to a workflow becomes extremely complex, since the representations of input and output data will not often match between each process in the sequence. Adhering to common standards enables interoperability (Kiehle et al., 2006), and solves these issues. A number of standards for Web Services have been developed by the OGC, with the Sensor Observation Service (SOS) and Web Processing Service (WPS) being of most relevance in this scenario.

B. *Representing and Communicating Error*

UncertML is a recently developed XML language that can be used to encode probabilistic representations of uncertainty in an interoperable manner (Williams et al., 2009). UncertML supports a range of uncertainty types, including: simple statistics-based characterisations (such as mean and (co)variance); probability distributions; non-parametric representations (such as realisations from a conditional simulation). UncertML is designed to combine strong and weak typing as appropriate, making it possible to create new uncertainty types as required, while maximising interoperability. We have used UncertML and its API (Application Programming Interface) to 'uncertainty-enable' the services used in our model Web. The term 'uncertainty-enabled' relates to the ability to parse and utilise XML-encoded information on data uncertainty.

C. *Workflow Detail*

The workflow (Fig. 1) initially gathers air pressure measurements from Weather Underground. The Weather Underground API uses non-standardised XML-based representations of station observations. In order to allow standardised access to this data, so it can form a part of the Model Web, we harvest this data at set intervals and add the observations to a SOS. Unfortunately, as the Weather Underground data does not quantify uncertainty, it is not possible to include any location or result uncertainty as input to the pressure correction sequence although recent work (Williams et al., 2010) indicates that a statistical combination of this data with sparser but more reliable measurements (e.g. from the UK Meteorological Office) could allow enhanced, 'quality assured' data to be supplied from a similar SOS which in turn made use of a correction WPS.

For each pressure measurement, the following correction sequence is executed to convert the station level measurement to sea level. Each step in this sequence is exposed as a process on a WPS. The first step retrieves elevation samples from the SRTM. For a station location in the UK, given as a GML point, a grid measuring 9x9 cells is created around the location and the elevation and associated uncertainty at each cell is returned. In work by Rodriguez et al. (2005), a comparison of the radar elevations with kinematic GPS transects determined that height estimates for Eurasia had a standard deviation of 3.7 metres, which is returned by the process as the elevation uncertainty for each grid location. The set of 81 elevations is returned as an O&M observation collection with embedded UncertML Distributions. In this particular example, all elevations have the same estimated uncertainty and correlations are ignored, but the UncertML schema allows a different estimate and characterisation of uncertainty for every data point in a collection. In UncertML all data values are inherently uncertain, (i.e., they are random variables, rather than values with defined quality metadata).

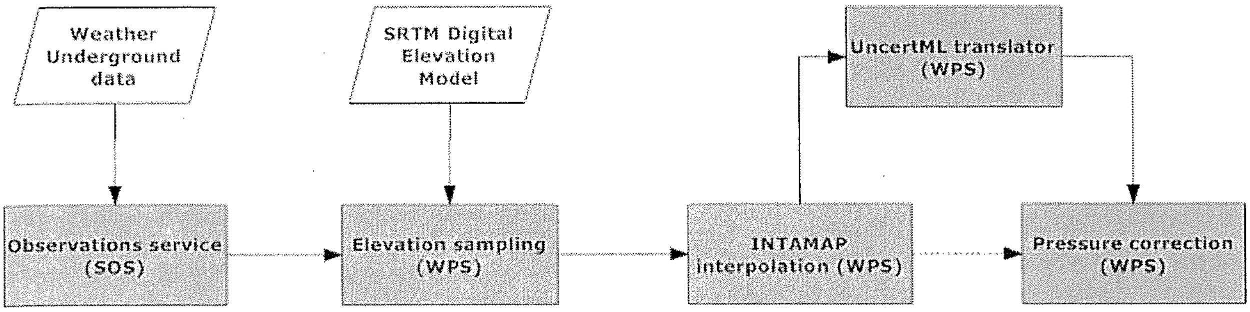

Figure 1. An overview of the pressure correction workflow

The next process interpolates these samples, using the INTAMAP automatic interpolation service (INTAMAP, 2009) to obtain a predicted elevation at the station location. INTAMAP exposes a range of interpolation algorithms via a WPS, including interpolation methods that enable the propagation of uncertainty from the individual samples to the interpolated output (Ingram et al., 2009). A prediction of the elevation at the station location is returned, complete with an uncertainty estimate, again represented as an UncertML Distribution.

Finally, the predicted elevation can be used with the station level pressure measurement to correct to sea level. As previously mentioned, the output from the INTAMAP automatic interpolation service is an UncertML Distribution, but the pressure correction process only accepts a set of realisations as an input. Therefore, an intermediate translation Web service was created to generate a set of realisations from a distribution. With appropriate semantic information attached to service descriptions this sort of data translation could be automated in future versions.

The correction algorithm has additional sources of uncertainty. Based on expert elicitation, the standard temperature is assumed to have a Gaussian distribution with a mean of 288.15 and a standard deviation of 8°C. A Gaussian distribution is also assumed for the temperature lapse rate, with a mean of 0.00649 and standard deviation of 0.00175°Cm^{-1}. For each of the elevation realisations, the pressure correction algorithm is performed with a realisation from the standard temperature and lapse rate distributions. A set of UncertML realisations is returned by the service.

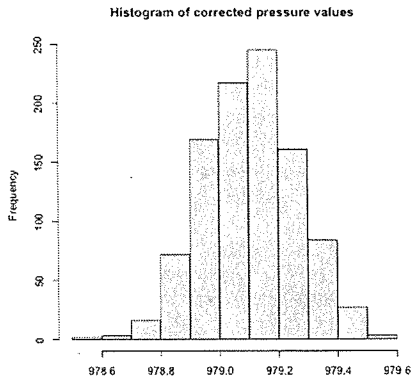

Figure 2. Histogram of the workflow output

D. Workflow Orchestration

To create workflows involving web services, OASIS (Organization for the Advancement of Structured Information Standards) define the BPEL (Business Process Execution Language) standard. Using this language it is possible to specify peer-to-peer interaction between a number of services, of both a synchronous and asynchronous nature (OASIS, 2007). Interactions for a workflow are specified in an XML-based script and deployed remotely on an engine, which is responsible for the orchestration of the services involved. The chain is then exposed and can be consumed in a 'stateless' manner. This simply means that the component services store no information about the specific client.

BPEL is built on Web Service standards, and has a reliance on two related technologies: WSDL (Web Service Description Language) and to a lesser extent, SOAP (Simple Object Access Protocol). A service can only be included in a BPEL workflow if it is described with a WSDL document. SOAP is the preferred format for messages, but other mechanisms can be used. While the current SOS (OGC, 2007a) and WPS (OGC, 2007b) specifications do not contain enough concrete detail for service providers to integrate SOAP / WSDL, it is possible to support the inclusion of existing SOS and WPS instances in BPEL workflows through the use of generic WSDL documents. These documents simply describe the HTTP POST interfaces of the operations of interest, i.e. GetObservation and Execute for a SOS and WPS respectively. The GetObservation operation allows a user to request observations based on given criteria, such as bounding box or time span, and the Execute operation performs a selected process on data.

Albeit is possible to use existing SOS and WPS instances in the BPEL chain without modifying any services, there are some issues with this method. Using SOS and WPS instances based on the current OGC specifications, only HTTP POST and GET transport is available. Any error (or 'exception') which occurs part-way through a chain's execution (for example, a failure of an individual model to converge on a satisfactory solution, or a memory problem due to huge data inputs) needs to be identified at source in order to correct the problem or at least evaluate its impact on the subsequent analyses. While SOAP transport has standard fault elements, allowing BPEL to catch a fault returned by a service, HTTP POST does not. As the generic WSDL documents for a WPS only specify that the input message must be of type Execute and the output message will be of type ExecuteResponse, the consumer is unaware of the exact input and output parameters for each process. Therefore, these parameters must be obtained through a GetCapabilities request and added to the base Execute document in the BPEL script. This makes the chain creation using OGC Web Services and BPEL complicated, as the user must be knowledgeable on the WPS specification. These matters could be resolved with concise WSDL documents or an appropriate client for BPEL workflow creation.

Additional issues arise for the user when using services that have input and outputs described with complex schemas, such as GML and O&M. These schemas define a number of geographic and observation-based primitives. Due to the large number of elements in these schemas, the service consumer is often unaware of the exact elements that are required, for example a GML Point with latitude/longitude coordinates, or an O&M Observation with a floating point result. It is rare that a service will be able to understand every element in a complex schema. While the precise data requirements could be specified in a textual description, there is currently no mechanism supporting machine readable descriptions in the WPS specification, something which would facilitate the automatic composition of chains and allow the generation of more intuitive client software and interfaces.

The WPS specification allows data to be passed as a reference, therefore reducing data transfer between services and the orchestration engine. However, some data may require additional selection within the BPEL process as only a subset of the output data from one service may be required as input to the next service, making reference passing unfeasible and therefore requiring data to be transferred between the orchestration engine and services.

Other workflow systems, such as Taverna and Kepler, are also freely and publicly available. These systems have several advantages over BPEL – for example, a more compact set of data flow modelling primitives, and tools to support the complete workflow lifecycle such as discovery and results analysis (Tan et al., 2009). While it is possible to share the saved workflow descriptions, they generally run locally and require that the recipient wishing to execute the workflow installs a copy of the client. This presents a barrier to workflow reuse. BPEL, on the other hand, exposes chains as standard WSDL documents, allowing them be consumed either by using a generic client, or by automatically generating client code which can be used in applications.

IV. SUMMARY AND DISCUSSION

This relatively simple chain illustrates the potential for chaining of geospatial (and, potentially, non-geospatial) web services allowing informed decision-making under real-world conditions. Using interoperable standards, granular functionality can be exposed as reusable Web Services. These services can then be utilised to create complex workflows. The use of 'uncertainty-enabled' services allows the propagation of uncertainty throughout the chain.

In taking the idea to more complex geospatial models, several technical and scientific issues arise. In our contrived chain we were able to ensure that service inputs and outputs were compatible with each other, and in the event that they were not, a translation service was used. However, in reality services will use many data formats, requiring translating data from one format to another and a restricted set of formats for services taking part in the uncertainty-enabled Model Web. The suitability of BPEL will also have to be assessed for more complex models, as outstanding issues such as efficiency, exception handling, and composition complexity will be even more prevalent. Furthermore a mechanism will be required to specify that a service can understand and propagate uncertainty, and describe what inputs and outputs have uncertainty, and are supported. If all services can describe the type of uncertainty they support in a standard way, automatic translation will be possible.

REFERENCES

Burrough, P. (1992). Development of intelligent geographical information systems. *International Journal of Geographical Information Science*. 6, 497–513.

Couclelis, H. (2003). The certainty of uncertainty: gis and the limits of geographic knowledge. *Transactions in GIS*. 7(2), 165–175.

Duckham, M., Mason, K., Stell, J. and Worboys, M. (2001). A formal approach to imperfection in geographic information. *Computers, Environment and Urban Systems*. 25, 89–103.

Heuvelink, G., Brown, J. and van Loon, E. (2007). A probabilistic framework for representing and simulating uncertain environmental variables. *International Journal of Geographical Information Science*. 21(5), 1–11.

Hunter, G. and Goodchild, M. (1993). Managing uncertainty in spatial databases: putting theory into practice. *Journal of the Urban and Regional Information Systems Association*. 5(2), 55–62.

Ingram, B., Cornford, D. and Barillec, R. (2009). Projected sequential gaussian processes: flexible interpolation for large data sets. *StatGIS2009*, Milos, Greece.

INTAMAP (2009). *Interoperability and automated mapping*. http://www.intamap.org.

ISO/FDIS 19115 (2003). *19115:2003 Geographic information – Metadata*. ISO, Geneva, Switzerland.

Kiehle, K., Greve, K., Heier, C. (2006). Standardized geoprocessing - taking spatial data infrastructures one step further. In: *Proceedings of the 9th AGILE Conference on Geographic Information Science, Visegrád, Hungary*.

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.

OGC (2007a). *Sensor observation service*. Available from http://portal.opengeospatial.org/files/?artifact_id=26667.

OGC (2007b). *OPENGIS® web processing service*. Available from http://portal.opengeospatial.org/files/?artifact_id=24151.

Rodriguez, E. et al. (2005). *An assessment of the SRTM topographic products*. Technical Report JPL D-31639, Jet Propulsion Laboratory, Pasadena, California.

Tan W., Missier P., Madduri R., and Foster I. (2009). Building scientific workflow with Taverna and BPEL: A comparative study in caGRID. In: *Proceedings. 4th International workshop on Engineering Service-Oriented applications* (WESOA), pp. 118–129.

Williams, M., Cornford, D., Bastin, L. and Pebesma., E. (2009). *Uncertainty Markup Language (UncertML). discussion paper 08-122r1*. Open Geospatial Consortium. Available from http://portal.opengeospatial.org/files/?artifact_id=33234.

Williams, M., Cornford, D., Bastin, L., Jones, R. and Parker, S. (2010, in review). Automatic processing, quality assurance and serving of real-time weather data. *Computers and Geosciences*.