



Data, Information and Process Integration
with Semantic Web Services

DIP

Data, Information and Process Integration with Semantic Web Services

FP6 - 507483

Deliverable

WP 9: Case Study eGovernment

D9.15

SWS Enhanced GIS Prototype (WSMX) Final Version

Bernhard Schreder

Mary Rowlatt

Sandra Stincic

Alexander Wahler

January 22th, 2007



EXECUTIVE SUMMARY

This document represents the documentation for the final version of the SWS enhanced GIS prototype, developed with the WSMX implementation of the DIP architecture.

This prototype contributes to the following golden bullet of DIP: Real Use Case Implementation of SWS in the e-Government sector.

This prototype further develops a Semantic Web GIS in which data sources and services are made available through SWS, described by ontologies, allowing interoperability as well as reasoning to create a comprehensive response adapted to user goals.

The scenario of the prototype is based on the needs for Essex County Council (ECC) emergency planning department to access geospatial data for management and to share it with other partners in case of an emergency situation. We have focused this prototype on an SWS-based Emergency Management System (EMS), at the same time the framework is evolving in a generic manner to support Spatial Data Integration with Semantic Web Services, called the eMerges platform.

Previous versions of the EMS scenario have already been realized using the two different implementations for a Semantic Execution Environment (SEE) developed in DIP: IRS-III and WSMX. This final prototype using the WSMX version of the DIP architecture demonstrates the same scenario as these previous releases, but features a completely new front-end web application which integrates the functionality a SEE provides by means of the DIP API and includes completely new data sources, like data from the Meteorological Office, as well. Again the functionality and data provided by this version is similar to the IRS-III version, proving the interoperability of the two Semantic Execution Environments.

The deliverable provides the fact sheet how WSMX has been used to create the use case scenario. In addition, it will be of interest to the end-user community and other data suppliers (e.g.: Essex County Council and other public authorities' emergency planners).

Document Information

IST Project Number	FP6 – 507483	Acronym	DIP
Full title	Data, Information, and Process Integration with Semantic Web Services		
Project URL	http://dip.semanticweb.org		
Document URL			
EU Project officer	Werner Janusch		

Deliverable	Number	9.15	Title	SWS Enhanced GIS Prototype (WSMX) Final Version
Work package	Number	9	Title	Case Study eGovernment

Date of delivery	Contractual	M 36	Actual	30-Jan-07
Status	version 1.0		final	<input checked="" type="checkbox"/>
Nature	Prototype <input checked="" type="checkbox"/> Report <input type="checkbox"/> Dissemination <input type="checkbox"/> Ontology <input type="checkbox"/>			
Dissemination Level	Public <input type="checkbox"/> Consortium <input checked="" type="checkbox"/>			

Authors (Partner)	Bernhard Schreder (Hanival), Mary Rowlatt (ECC), Sandra Stincic (BT), Alexander Wahler (Hanival)			
Responsible Author	Bernhard Schreder		Email	bernhard.schreder@hanival.net
	Partner	Hanival	Phone	0043(0)1 3195843/13

Abstract (for dissemination)	Public organisations access geospatial data for management as well as for communication purposes. The approach of using traditional Geographical Information Systems (GIS) to access spatial-related data is not always satisfactory as users have to cope with distributed heterogeneous data sources to find appropriate resources for particular situations. This prototype develops a Semantic Web 'GIS' in which data sources and services are made available through SWS, described by ontologies, allowing interoperability as well as reasoning to create a comprehensive response adapted to user goals. We focus on an Emergency Management System as a practical example to show the full potential of this technology.
Keywords	Prototype, emergency planning, GIS, WSMX, Spatial-related data (SRD), Emergency Management System (EMS), eMerges








Version Log			
Issue Date	Rev No.	Author	Change
22-01-07	001	Bernhard Schreder	First draft of prototype factsheet
30-01-07	002	Bernhard Schreder	Final version, including comments from reviewers






Reviewer			
	Rob Davies	Email	
	Partner MDR	Phone	
		Email	
	Partner	Phone	

Project Consortium Information

Partner	Acronym	Contact
National University of Ireland Galway	NUIG 	Dr. Sigurd Harand Digital Enterprise Research Institute (DERI) National University of Ireland, Galway Galway Ireland Email: sigurd.harand@deri.org Tel: +353 91 495112
Fundacion De La Innovacion.Bankinter	Bankinter 	Monica Martinez Montes Fundacion de la Innovacion. BankInter Paseo Castellana, 29 28046 Madrid, Spain Email: mmtnez@bankinter.es Tel: 916234238
British Telecommunications Plc.	BT 	Dr John Davies BT Exact (Orion Floor 5 pp12) Adastral Park Martlesham Ipswich IP5 3RE, United Kingdom Email: john.nj.davies@bt.com Tel: +44 1473 609583
Swiss Federal Institute of Technology, Lausanne	EPFL 	Prof. Karl Aberer Distributed Information Systems Laboratory École Polytechnique Fédérale de Lausanne Bât. PSE-A 1015 Lausanne, Switzerland Email : Karl.Aberer@epfl.ch Tel: +41 21 693 4679
Essex County Council	Essex 	Mary Rowlett, Essex County Council PO Box 11, County Hall, Duke Street Chelmsford, Essex, CM1 1LX United Kingdom. Email: maryr@essexcc.gov.uk Tel: +44 (0)1245 436524
Forschungszentrum Informatik	FZI 	Andreas Abecker Forschungszentrum Informatik Haid-und-Neu Strasse 10-14 76131 Karlsruhe Germany Email: abecker@fzi.de Tel: +49 721 9654 0
Institut für Informatik, Leopold-Franzens Universität Innsbruck	UIBK 	Prof. Dieter Fensel Institute of computer science University of Innsbruck Technikerstr. 25 A-6020 Innsbruck, Austria Email: dieter.fensel@deri.org Tel: +43 512 5076485

Partner	Acronym	Contact
ILOG SA	<p>ILOG</p>  <p>Changing the rules of business</p>	<p>Christian de Sainte Marie 9 Rue de Verdun, 94253 Gentilly, France Email: csma@ilog.fr Tel: +33 1 49082981</p>
inubit AG	<p>Inubit</p> 	<p>Torsten Schmale inubit AG Lützowstraße 105-106 D-10785 Berlin Germany Email: ts@inubit.com Tel: +49 30726112 0</p>
Intelligent Software Components, S.A.	<p>iSOCO</p> 	<p>Dr. V. Richard Benjamins, Director R&D Intelligent Software Components, S.A. Pedro de Valdivia 10 28006 Madrid, Spain Email: rbenjamins@isoco.com Tel. +34 913 349 797</p>
MDR Partners	<p>MDR</p> 	<p>Rob Davies MDR Partners 8 St. Andrew Street Hertford, Herts. United Kingdom, SG14 1JA, Email: rob.davies@mdrpartners.com +44 (0)208 8763121</p>
Hanival Internet Services GmbH	<p>HANIVAL</p> 	<p>Alexander Wahler Hanival Internet Services GmbH Kirchengasse 13/1a A-1070 Wien Email: wahler@niwa.at Tel:+43(0)1 3195843-11 </p>
The Open University	<p>OU</p> 	<p>Dr. John Domingue Knowledge Media Institute The Open University, Walton Hall Milton Keynes, MK7 6AA United Kingdom Email: j.b.domingue@open.ac.uk Tel.: +44 1908 655014</p>
SAP AG	<p>SAP</p> 	<p>Dr. Elmar Dörner SAP Research, CEC Karlsruhe SAP AG Vincenz-Priessnitz-Str. 1 76131 Karlsruhe, Germany Email: elmar.dorner@sap.com Tel: +49 721 6902 31</p>

Partner	Acronym	Contact
Sirma AI Ltd.	<p>Sirma</p>  <p>Ontotext Knowledge and Language Engineering Lab of Sirma</p>	<p>Atanas Kiryakov, Ontotext Lab, - Sirma AI EAD Office Express IT Centre, 3rd Floor 135 Tzarigradsko Chausse Sofia 1784, Bulgaria Email: atanas.kiryakov@sirma.bg Tel.: +359 2 9768 303</p>
Unicorn Solution Ltd.	<p>Unicorn</p> 	<p>Jeff Eisenberg Unicorn Solutions Ltd, Malcha Technology Park 1 Jerusalem 96951 Israel Email: Jeff.Eisenberg@unicorn.com Tel.: +972 2 6491111</p>
Vrije Universiteit Brussel	<p>VUB</p>  <p>Vrije Universiteit Brussel</p>	<p>Pieter De Leenheer Starlab- VUB Vrije Universiteit Brussel Pleinlaan 2, G-10 1050 Brussel ,Belgium Email: Pieter.De.Leenheer@vub.ac.be Tel.: +32 (0) 2 629 3749</p>

LIST OF KEY WORDS/ABBREVIATIONS

ECC	Essex County Council
EMS	Emergency Management System
GIS	Geographical Information System
GUI	Graphical User Interface
SEE	Semantic Execution Environment
SOA	Service Oriented Architecture
SWS	Semantic Web Services
TUI	Text User Interface
WSMO	Web Service Modeling Ontology
WSMX	Web Service Execution Environment

TABLE OF CONTENTS

EXECUTIVE SUMMARY	I
LIST OF KEY WORDS/ABBREVIATIONS	VII
TABLE OF CONTENTS	VIII
1 INTRODUCTION	1
2 ARCHITECTURE PROTOTYPE FACT SHEET	1
2.1 Available Documentation	1
2.2 Demonstration Information	2
2.3 Description of purpose, scope and functionality	3
2.4 Installation Guidelines	3
2.5 Type of API	4
2.6 Detailed license information.....	4
2.7 Roadmap of future plans.....	4
3 CONCLUSION	4
REFERENCES	5
APPENDIX 1: SAMPLE WSML DESCRIPTION	6
APPENDIX 2: SAMPLE WEB SERVICE USED FOR THE PROTOTYPE	7

LIST OF FIGURES

Figure 1: SWS Enhanced GIS Prototype.....	2
---	---

LIST OF TABLES

Table 1: WSMX components used for the prototype	3
---	---

1 INTRODUCTION

This document represents the accompanying documentation for the prototype deliverable “SWS Enhanced GIS Prototype (WSMX) Final Version”, which is the final, updated version of the GIS prototype using the WSMX implementation of the DIP architecture. The document itself is a revised version of the previous WSMX prototype fact sheet [9].

As a prototype deliverable, this document is structured as follows: Following this introduction, Section 2 features the Architecture Prototype Fact Sheet, which contains all the relevant information concerning the prototype, i.e. installation guidelines and links to additional documentation relevant for the prototype implementation. The conclusion summarises the functionalities and status of the final prototype. In Appendix I some sample WSMX descriptions of the Semantic Web Services, Goals and Ontologies as used in this version are presented.

As this version of the GIS prototype uses the same Front-end GUI as the IRS version described in [1], the document refers to this deliverable instead of repeating information described therein. The functionalities of this version of the prototype includes the possibility of collecting data about differently classified locations (e.g. rest centres, supermarkets, inns, schools etc.). as well as data about snowfall, as provided by the Meteorological Office. Further information on these services is provided in Section 2.3.

2 ARCHITECTURE PROTOTYPE FACT SHEET

The following fact sheet contains all relevant information for this prototype deliverable, and consists of information about the available documentation, installation guidelines as well as API and licence information.

2.1 Available Documentation

Figure 1 below depicts an updated overview of the WSMX based prototype for the SWS Enhanced GIS Prototype. The components shown in the figure supersede the same components shown in the overall architecture presented in Figure 1 of [2].

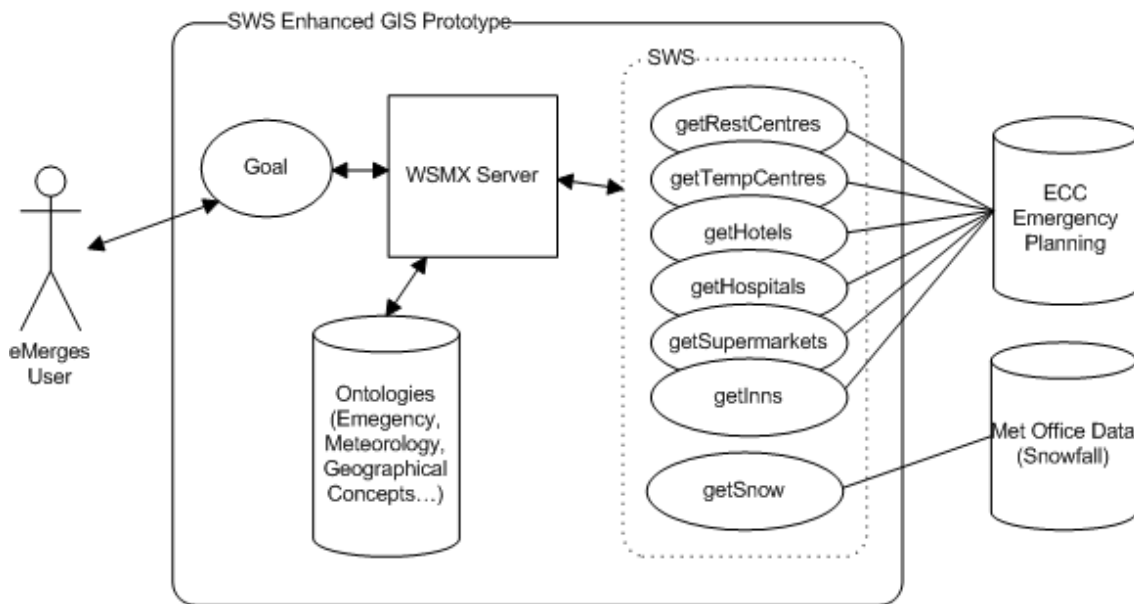


Figure 1: SWS Enhanced GIS Prototype

This version of the SWS Enhanced GIS Prototype utilises the ECC Emergency Planning Services, to collect and aggregate information about possible rest centres, hospitals, inns, schools and supermarkets in the vicinity. An additional data source for this version of the prototype was the Meteorological Office, providing data about the snowfall in a defined area.

The separate parts of the prototype application are documented as follows:

- The Front-end GUI – the eMerges Web Application – has been described in detail in [1].
- The WSMX prototype implementation for the DIP architecture has been described in various WP6 deliverables. The version used for the SWS Enhanced GIS Prototype is the latest available version available in December 2006. Refer to [3], [6] and [8] for a detailed description of the execution semantics and components used for the prototype.
- The ECC Emergency Planning Web Services' WSDL descriptions are available at <http://ngwr.labs.bt.com:8080/axis/services>.

2.2 Demonstration Information

The prototype can be demonstrated in the same way as the IRS-based version. The Screencast tutorial available at <http://irs-test.open.ac.uk/sgis-dev/> provides a detailed walkthrough of using the eMerges Web Application to access the prototype functionalities.

Using the new version of the Front-end GUI, the user can achieve the goal of finding possible accommodations in a given radius by accessing the WSMX server, in addition to requesting data about current snowfall in a give area. A WSML goal is created by an intermediate adapter component, then sent to the WSMX server. The server instance invokes one of several deployed Semantic Web Services, which provide the capability to fulfil the goal sent to WSMX. Additionally the Front-end interacts with WSMX by

way of the DIP API, using the internal repository provided by WSMX to fetch ontologies and goal descriptions to be used in service requests.

2.3 Description of purpose, scope and functionality

The general purpose, scope and functionality for the SWS Enhanced GIS Prototype are described in [1]. This version of the prototype uses WSMX to invoke the Semantic Web Services providing ECC Emergency Planning and weather information. WSMX is an execution environment, which enables discovery, selection, mediation, invocation and interoperation of Semantic Web Services (SWS). WSMX is based on the conceptual model provided by WSMO, being at the same time a reference implementation of it, respectively of the DIP architecture as defined in [6].

The prototype implementation of the SWS Enhanced GIS Prototype requires a subset of the currently available WSMX Components. The components used for the prototype can be seen in Table 1.

Table 1: WSMX components used for the prototype

Component name
Adapter Framework
CommunicationManager
Parser
ResourceManager

These components are used for a modified version of the execution semantics accessed by the “achieveGoal” entry point into WSMX. Execution semantics, or operational semantics, is the formal definition of the operational behaviour of a system. Functionality provided by the WSMX system as a whole can be described in terms of its entry points - the standardized interfaces of the system enabling communication with any external entities requesting services from the system. Entry points and corresponding execution semantics have been explained in detail in [6], respectively in [8]. In additions to those standard entry points, this version of the WSMX based prototype utilizes the DIP API to exchange data with the repository provided by WSMX.

Different web services are utilised by the prototype, each of which is additionally described by a WSML service description. A sample description in WSML is given in Appendix I, while Appendix II contains the information about the new weather service, providing snow data.

2.4 Installation Guidelines

The eMerges Web Application itself, running at <http://hanival.net/sgis-dev/>, can only be accessed online. The current release of WSMX (version 0.3) is available in the form of

different packages, either from the SourceForge project page¹, or from the website <http://www.wsmx.org>, which hosts nightly builds of the WSMX Core and Components from the WSMX CVS.

Detailed Installation Guidelines for WSMX can be found in [3].

2.5 Type of API

The API for component interfaces (their sources, binaries and documentation) is available for download at: <http://sourceforge.net/projects/wsmx>. All implementations of the DIP architecture, such as WSMX or IRS, conform to the Semantic Execution Environment (SEE) Integration API. Third party component providers should download the newest version of the SEE Integration API, in order for their components to be compatible with the DIP architecture.

Additional APIs used for the architecture prototype include the WSMO API [7].

2.6 Detailed license information

WSMX uses the GNU Lesser General Public Licence².

The third party software components and libraries included in the current WSMX release are using diverse licences, which were first presented in Appendix 2 of [4].

More detailed information about licensing of DIP components is provided in [5].

2.7 Roadmap of future plans

This is the final version of the prototype available in the scope of the DIP project. Still, development is continuing on two different courses: The eMerges framework will be further expanded and will be developed into a generic framework for spatial data integration through SWS. The ongoing work on eMerges can be followed at <http://irs-test.open.ac.uk/sgis-dev/>.

Additionally, the development of the Semantic Execution Environments is ongoing. The DIP architecture and its implementations WSMX and IRS-III are going to form the technological basis for a couple of new research projects, such as the SUPER project³.

3 CONCLUSION

This deliverables provides the fact sheet for the final version of the SWS Enhanced GIS Prototype based on the WSMX implementation prototype for the DIP architecture. For demonstration purposes the online version of the prototype accessible at <http://hanival.net/sgis-dev/> can be used. A Screencast tutorial, demonstrating the use of the prototype is available at <http://irs-test.open.ac.uk/sgis-dev/>. The tutorial provides more detailed information on the usage of the eMerges prototype.

¹ Available at <http://sourceforge.net/projects/wsmx/>

² <http://www.opensource.org/licenses/lgpl-license.php>

³ <http://www.ip-super.org/>

REFERENCES

- [1] Gugliotta, A., et al. (2007): *D9.14: SWS Enhanced GIS prototype (IRSIII) Final Version*, DIP Project, WP9 Case Study eGovernment. <http://dip.semanticweb.org>
- [2] Gutierrez, L., et al. (2006): *D9.11: SWS Enhanced GIS prototype (IRSIII) v1.0*, DIP Project, WP9 Case Study eGovernment. <http://dip.semanticweb.org>
- [3] Haselwanter, T.; Schreder, B.; Wahler, A.; Zaremba, M. (2006): *D6.13: Software Deliverable and Installation Guidelines V3*, DIP Project, WP6 Interoperability and Architecture. <http://dip.semanticweb.org>
- [4] Haselwanter, T.; Schreder, B.; Wahler, A.; Zaremba, M. (2005): *D6.7: Architecture Prototype v1.0*, DIP Project, WP6 Interoperability and Architecture. <http://dip.semanticweb.org>
- [5] De Saint Marie, C. (2005): *D13.2: Analysis of the appropriate open-source licensing schemata, including those used for related WS and B2B standards*, DIP Project, WP6 Interoperability and Architecture. <http://dip.semanticweb.org>
- [6] Zaremba, M. et al. (2006): *D6.11: Semantic Web Services Architecture and Information Model*, DIP Project, WP6 Interoperability and Architecture. <http://dip.semanticweb.org>
- [7] Dimitrov, M. et al. (2006): *D6.14: WMSO API v2.0 WSMO4J + DIP API v3.0*, DIP Project, WP6 Interoperability and Architecture. <http://dip.semanticweb.org>
- [8] Domingue, J.; Norton, B., Shafiq, O., Zaremba, M., Sapkota, B., Moran, M., (2006): *D6.12: Execution Semantics for Semantics-Enabled Systems, Standardisation activity with OASIS*, DIP Project, WP6 Interoperability and Architecture. <http://dip.semanticweb.org>
- [9] Schreder, B., et al. (2006): *D9.13: SWS Enhanced GIS Prototype (WSMX) v 1.0*, DIP Project, WP9 Case Study eGovernment. <http://dip.semanticweb.org>

APPENDIX 1: SAMPLE WSML DESCRIPTION

A sample WSML Web Service description used for this version of the prototype is shown below, specifically for the “getSupermarkets” WS, which returns a list of supermarkets in the vicinity, given coordinates and a radius. Other information for a supermarket includes the opening hours and specific goods provided. The Namespace declarations have been omitted from the WSML fragment.

```

webService getSupermarketsWS
  nonFunctionalProperties
    dc#title hasValue "WSMO for ECC Supermarkets"
    dc#subject hasValue {"Rest Centre", "Emergency", "GIS"}
    dc#description hasValue "WSMO Web service for ECC Supermarkets"
    dc#date hasValue "2006-04-30"
    dc#type hasValue _"http://www.wsmo.org/2004/d2#webservice"
    dc#format hasValue "text/html"
    dc#identifier hasValue
    _"http://www.essexcc.gov.uk/services/getSupermarketsWS"
    dc#source hasValue
    _"http://www.essexcc.gov.uk/services/getSupermarketsWS"
    dc#language hasValue "en-US"
    dc#relation hasValue {
      _"http://www.essexcc.gov.uk/emergency",
    }
    _"http://dip.bat.bt.co.uk:8081/axis/services/eccSupermarkets?wsdl"
    _"http://owner" hasValue _"http://eccRestCentres"
    wsmo#version hasValue "$Revision: 1.30 $"
    wsmo#endpointDescription hasValue
    _"http://dip.bat.bt.co.uk:8081/axis/services/eccSupermarkets"
  endNonFunctionalProperties

  capability getSupermarketsWSCapability
    nonFunctionalProperties
      dc#description hasValue "This capability provides a description
of the getSupermarkets service"
    endNonFunctionalProperties

  sharedVariables ?geodata

  precondition getSupermarketsWSPreconl
    nonFunctionalProperties
      dc#description hasValue "The input has to be a radius and
coordinates"
    endNonFunctionalProperties

  definedBy
    ?geodata memberOf ser1#supermarketsReq
    and ?geodata[ser1#latitude hasValue ?latitude]
    and ?geodata[ser1#longitude hasValue ?longitude]
    and ?geodata[ser1#radiusKM hasValue ?radiusKM] .

  interface getSupermarketsWSInterface
    choreography _"http://www.essexcc.gov.uk/services#choreography"

```

```

stateSignature
_ "http://www.essexcc.gov.uk/services#stateSignature"
  importsOntology {
    _ "http://www.essexcc.gov.uk/services/service1",
    _ "http://www.essexcc.gov.uk/emergency" }

    in ser1#supermarketsReq withGrounding {
_ "http://dip.bat.bt.co.uk:8081/axis/services/eccSupermarkets?wsdl#wsdl
.interfaceMessageReference(eccSupermarkets/getSupermarketsInRadius/get
SupermarketsInRadiusRequest)" }
    out ser1#restcentresResp withGrounding {
_ "http://dip.bat.bt.co.uk:8081/axis/services/eccSupermarkets?wsdl#wsdl
.interfaceMessageReference(eccSupermarkets/getSupermarketsInRadius/get
SupermarketsInRadiusResponse)" }

  transitionRules
_ "http://www.essexcc.gov.uk/services#rulesGetSupermarkets"

  forall {?x} with (?x memberOf ser1#supermarketsReq) do
    add(_#1 memberOf ser1#restcentresResp )
  endForall

```

Listing 1: Sample WSMML Web Service description

APPENDIX 2: SAMPLE WEB SERVICE USED FOR THE PROTOTYPE

The eccSnow Service

The **eccSnow** Service provides data about snowfall. It can display lists of polygons which have snow level bigger than a given value at a certain time (method *getSnowPolygon*), in the radius around a certain point (method *getSnowPolygonInRadius*) or data inside/outside of the given polygon (method *getSnowPolygonWithinPolygon*).

getSnowPolygon

- input:
 - time – time of measure (e.g. “1500”)
 - snow – amount of snowfall in millimetres (e.g. “4”)

getSnowPolygonInRadius

- input:
 - latitude - latitude of the start point
 - longitude - longitude of the start point
 - radius – radius around the start point (in kilometres)
 - time – time of measure (e.g. “1500”)
 - snow – amount of snowfall in millimetres (e.g. “4”)

getSnowPolygonWithinPolygon

- input:

- crd – list of polygon coordinates in the form of latitude/longitude pairs, e.g. “1.0,49.0, 1.0,54.0, 0.3,54.0, 0.3,49.0, 1.0,49.0”, where the first and last pair need to be the same
- option - determines where the objects are - true for within, false for outside the polygon
- radiusKM - radius in KM around the polygon area (0 if not needed)
- time – time of measure (e.g. “1500”)
- snow – amount of snowfall in millimetres (e.g. “4”)

Output is the same for all three methods (MetResult) – a list of polygons with:

- cellx - numeric, location in met snowgrid
- celly - numeric, location in met snowgrid
- snow - amount of snowfall in that polygon (in millimetres)
- coordinates - set of lat/long pairs

Location: <http://ngwr.labs.bt.com:8080/axis/services/eccSnow>

WSDL description: <http://ngwr.labs.bt.com:8080/axis/services/eccSnow?wsdl>

Example usage:

[http://ngwr.labs.bt.com:8080/axis/services/eccSnow?method=getSnowPolygonInRadius
&latitude=51.8582&longitude=0.5643&radiusKM=20&time=1500&snow=1](http://ngwr.labs.bt.com:8080/axis/services/eccSnow?method=getSnowPolygonInRadius&latitude=51.8582&longitude=0.5643&radiusKM=20&time=1500&snow=1)