

# SOAP-based web services in GIS/RDBMS environment

Gerhard Smiatek\*

*Institute for Meteorology and Climate Research (IMK-IFU), Forschungszentrum Karlsruhe, 82467 Garmisch-Partenkirchen, Germany*

Received 10 September 2002; received in revised form 16 January 2004; accepted 20 April 2004

## Abstract

Implementation of Simple Object Access Protocol (SOAP)-based Web services within a collection of classes GSfM (GIS Support for MCCM) is presented. GSfM facilitates the provision of various geographical and tabular input data into the MCCM (Multiscale Chemistry Climate Model). The classes have been written in object-oriented Perl. They allow to write simple Perl scripts and to perform data provision while making intensive use of the GIS and RDBMS running on remote hosts. Appropriate GIS/RDBMS functions are used to create coverages representing the modeling area, to adjust the map projection and to intersect the available land use data as well as to retrieve all required tabular data from the relational database. Due to the ability of GSfM to use two SOAP-based Web services providing access to GIS and RDBMS on foreign hosts, the Perl scripts can run on almost any platform connected to the corporate network. The article describes the setup of SOAP server and client applications based on the SOAP::Lite library. Also an application example in provision data for modeling volatile organic compounds from biogenic sources (BVOC) is presented.

© 2004 Elsevier Ltd. All rights reserved.

**Keywords:** Web service; SOAP; SOAP::Lite; GIS; RBMS; Perl; BVOC

## Software availability

Name of software: GIS Support for MCCM (GSfM)  
Developer: Gerhard Smiatek  
Contact address: Institute for Meteorology and Climate Research (IMK-IFU), Forschungszentrum Karlsruhe, Kreuzeckbahnstrasse 19, 82467 Garmisch-Partenkirchen, Germany. Tel.: +49-8821-183292; fax: +49-8821-183243  
E-mail: [gerhard.smiatek@imk.fzk.de](mailto:gerhard.smiatek@imk.fzk.de)  
Hardware required: any with running Perl (Version > 5.6)  
Software required: ArcInfo GIS (> 7.1), Oracle (> 8) RDBMS, Perl DBI:Oracle module, Perl SOAP::Lite module  
Program language: Perl  
Program size: 200 K  
Availability and cost: free (by contacting the author)

## 1. Introduction

One of the major tasks of a corporate Geographical Information System (GIS) department is to make databases and geographic information available to internal and perhaps also external customers. In practice, however, the customers often wish to do more than to download some digital data. The acceptance of digital geographical data can only be increased if the customers can obtain data tailored to their specific needs. Often they also want to have access to some GIS functions at their specific platform and operating system without knowing much about a GIS or a Relational Database Management System (RDBMS). Here Web services exposing GIS/RDBMS functions are the solution. The Simple Object Access Protocol (SOAP) is a basic tool for Web services. It provides means to invoke objects on remote hosts and make methods calls upon those objects.

This article describes the implementation of Web services based on the SOAP-Lite Perl library. They give access to GIS and RDBMS functions within collection of classes forming the GIS Support for MCCM (GSfM).

\* Tel.: +49-8821-183282; fax: +49-8821-183243.

E-mail address: [gerhard.smiatek@imk.fzk.de](mailto:gerhard.smiatek@imk.fzk.de) (G. Smiatek).

GSfM facilitates the provision of user tailored land use, topography, anthropogenic and biogenic emission data for input in the 3D Multiscale Chemistry Climate Model (MCCM). MCCM is used to model air quality in various parts of the world. As an example of use the data provision for modeling biogenic volatile organic compounds (BVOC) emissions within the MCCM is discussed.

## 2. Data provision for modeling BVOC emissions

### 2.1. Requested data

On hot summer days VOC emissions from biogenic sources play a very important role in air chemistry. They influence the oxidation capacity of the troposphere and change the radiation budget through production of OH radicals and aerosols (Fuentes et al., 2001). They are therefore an important part of the photochemical cycle responsible for ozone (O<sub>3</sub>) formation and must be taken into account in any air quality model. Richter et al. (1999) and Stewart et al. (2003) describe in detail the semi-empirical approach for modeling BVOC within MCCM that is based on the algorithm presented by Guenther (1997). With this algorithm the emission  $E$  of a chemical compound  $i$  by the plant species  $k$  can be described as:

$$E_{ik} = A_k * EF * B_k * C(T, PAR) \quad (1)$$

where  $A_k$  is the area of the emitting plant species  $k$  in km<sup>2</sup>,  $EF$  the average emission potential (emission factor) in µg/gh,  $B_k$  the foliar biomass density in g/m<sup>2</sup> and  $C$  a unitless environmental correction factor. It corrects for the effects of actual temperature ( $T$ ) and actual solar radiation, given as the photosynthetically active radiation ( $PAR$ ), on the emission potential which is calibrated for 30 °C and full sunlight.

The correction factor  $C$  is calculated by a MCCM routine with predicted meteorology data. MCCM is a three dimensional meteorology model based on the Penn State/ NCAR meteorology model (MM5) with coupled air chemistry (Grell et al., 2000). In addition to a weather forecast it is able to simulate the transport of air pollutants. Taking into account the anthropogenic and biogenic emissions it can predict air quality. As MM5, MCCM runs in its horizontal dimension in a domain (mother domain)–subdomain (nest) system (Guo and Chen, 1994). The principle for a mother domain and one refined subdomain is depicted in Fig. 1. The mesh refinement (nesting) scheme provides an efficient way of increasing the resolution over subregions in a simulation. All scalar variables (i.e. land use, elevation, temperature, etc.) are defined at the center of the each mesh cell. The domain/subdomain mesh consists of  $n_i \times n_j$  cells, each of size  $dx$ . In mid-latitude

areas the mostly employed map projection is Lambert conformal conic with the central meridian going through the center point of the domain system ( $x_{lonc}$ ,  $phic$ ) and the standard parallels at 60° and 30°. Thus, all input data, such as land use, DEM, soil data, or anthropogenic emissions must be interpolated to the chosen domain and subdomains. It should be emphasized that any change in the modeling area setup (parameters  $x_{lonc}$ ,  $phic$ ,  $i_{max}$ ,  $j_{max}$ ,  $dx$ ) leads to different parameters of the Lambert projection or different shape of the domain/subdomain coverage and to a new data request.

According to the procedure described in Eq. (1) and the MCCM geometry the required input can precisely be described. It must contain the following information:

1. Area of the emitting land use category or plant species in each domain/subdomain cell for the chosen land use or plant species nomenclature associated with an available data set
2. Land use or plant species-specific emission factors for the different chemical compounds
3. Foliar biomass density data
4. Table relating the land use categories to more general vegetation types

### 2.2. Data processing

The provision of appropriate land use data and the tabular information on biomass and emission factors is facilitated by a collection of classes forming a GIS support for MCCM (GSfM) interface written in object-oriented Perl. A class defines variables and the methods common to all objects of a certain kind. In Perl classes are declared as packages. A package provides a separate name space within a program. The package name can be used to identify data and subroutines defined in that package. In order to act as classes packages must provide methods to generate and manipulate objects. Perl methods are just subroutines associated with a class. They exist to operate on objects that are instances of that class (Conway, 1999). A Perl library is a collection of reusable Perl code that is imported into a Perl script using the *require* operator. However, recently libraries refer more to Perl packages containing reusable Perl code and often some precompiled C code. They are introduced into a Perl script using the *use* operator. Such Perl Library is SOAP:Lite. A special case in Perl are packages stored in separate files with the extension *.pm*. They are called modules. All classes of GSfM are modules. They are classes as they provide methods to create and manipulate objects they define.

The GSfM data base is described in detail by Smiatek (2001). In the present article only the updated GIS part is described. The principle of data processing is depicted in Fig. 2. The procedure starts with the creation of

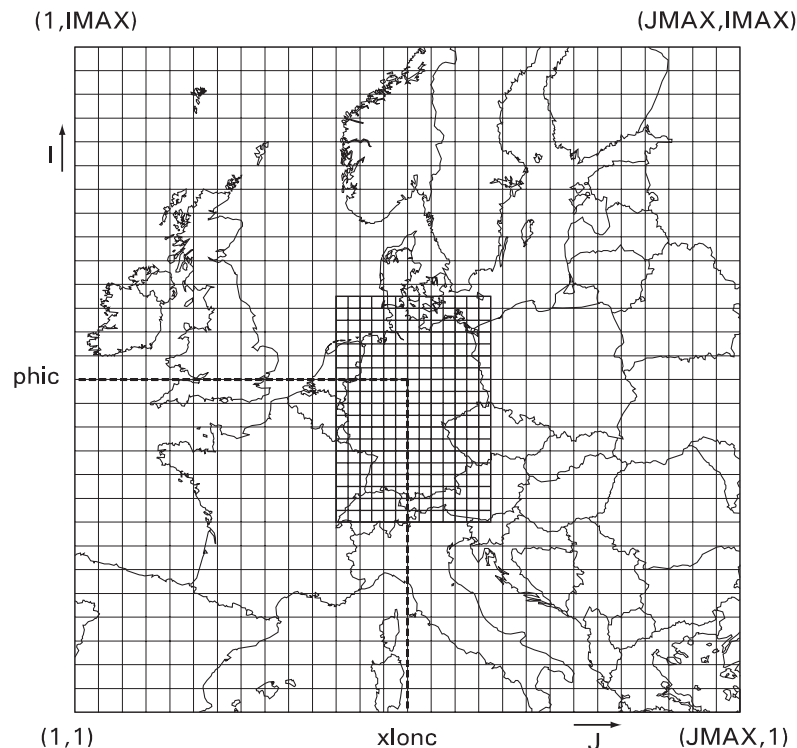


Fig. 1. Example for a domain–subdomain modeling system of MM5/MCCM. The mesh size of the coarse domain is  $100 \times 100$  km and the mesh size of the subdomain covering the area of Germany is  $50 \times 50$  km.

a mesh coverage representing the modelling area. Then tabular data is retrieved from the database and the land cover coverages are projected and intersected with the mesh coverage. The resulting polygon attribute table (PAT) contains all necessary information required to calculate the land cover or plant area within a grid cell of the modelling area.

Fig. 3 shows the class diagram of GSfM in the UML (Unified Modeling Language) notation. One of the strengths of the interface is its ability to replicate the MCCM modeling domain-subdomain system provided by the *Domain* and *Subdomain* classes (Fig. 3C). Thus, no additional interpolations on the model preprocessing level are required. Classes *Bplantspecies* and *Blanduse*

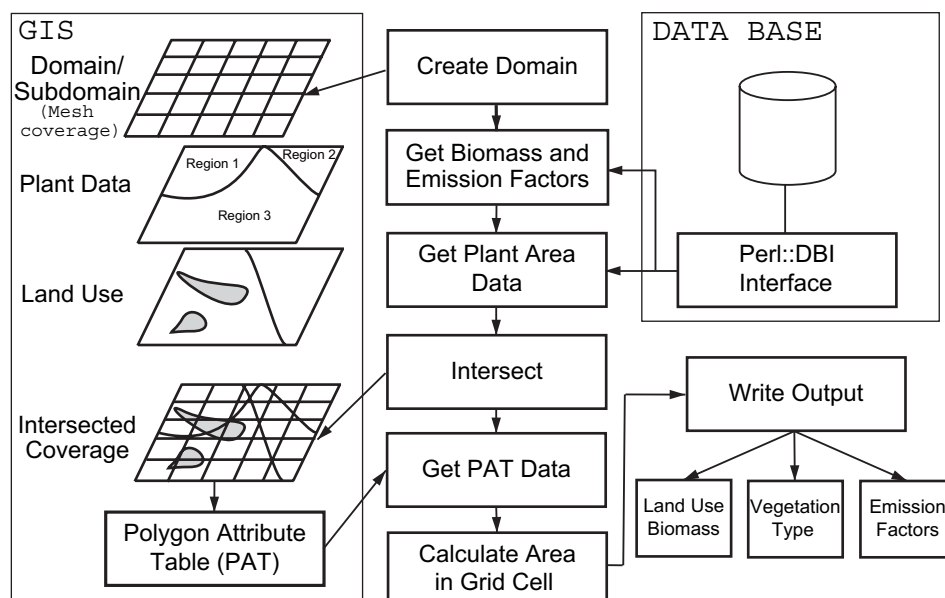


Fig. 2. Principle of the data processing in provision of data in modelling BVOC.

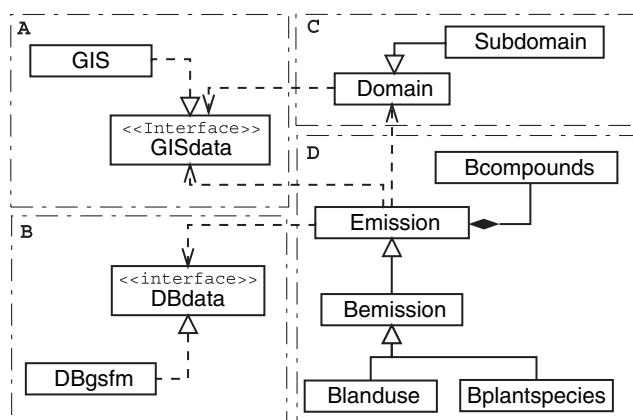


Fig. 3. Class diagram of the BVOC data provision. It is grouped into four parts: GIS access (A), data base access (B), creation of the MCCM model environment (C) and calculation of the land use data (D).

are responsible for the calculation of the area of emitting plant species or land use category (Fig. 3D). Both require one *Domain* or *Subdomain* object. They inherit the abstract classes *Bemission* and *Emission*. The class *Emission* implements methods common to both biogenic VOC emissions and also to anthropogenic emission inventories. Additional parts of GSfM, not considered here, can perform spatial and temporal disaggregation of anthropogenic emissions. The class *Bemission* extends it to features common to plant species and land use related BVOC inventories. It also aggregates the class *Bcompounds*, which implements the names of various chemical VOC compounds and compound groups.

Trees are the dominant species responsible for more than 80% of the BVOC emissions (Simpson et al., 1999). Therefore, in modeling BVOC land use data at the plant species level, especially forest trees should be used. Unfortunately, such data are not yet available at high geometrical resolution and for large geographic areas. The available statistical data are mostly related to administrative units or forest districts. Thus, the plant species-specific area data have to be disaggregated into grid cells. Class *Bplantspecies* provides appropriate methods to retrieve the plant area data from the database and to perform the disaggregation into the domain/subdomain grid cells. Class *Blanduse* implements the processing of land use data at general categories, such as deciduous, coniferous or mixed forest. This data is used in model applications in areas where plant species area data are not available yet.

To perform their duties, the classes make intensive use of the built-in GIS and RDBMS capabilities. Here, the abstract interface classes *GISdata* and *DBdata* define appropriate methods. *GISdata* (Fig. 3A) defines methods for projecting the domain center coordinates and the land use coverages into the Lambert projection, creating mesh coverages representing the domain and

subdomain, as well as intersecting and retrieving the resulting polygon attribute tables. *DBdata* (Fig. 3B) defines methods allowing to connect to the relational database as well as to select and retrieve plant species area data, emission factors and the biomass values. Those abstract classes are inherited by classes that provide the specific implementation of the defined methods for the available GIS and RDBMS. It should be noted that the database implementation *DBGsfm* uses a Perl DBI module. Appropriate modules for any RDBMS and Perl tutorials are available at the CPAN (Comprehensive Perl Archive Network at <http://www.cpan.org>).

Fig. 4 shows a simple Perl script that uses the described classes to create input data for a subdomain. In lines 2–5 the classes are introduced using the *use* operator. The Perl module *strict* restricts unsafe constructs, e.g. name conflicts. It forces the programmers to declare and assign their local variables with the *my* prefix. Next an object *\$domain* of the *Domain* type is created using the *new* constructor. Its data are MM5 style domain parameters, the name of the coverage to create, the name of the area, the geographic coordinates of the center point, the number of rows and number of columns and the cell size in kilometers. The method *createDomain* creates an appropriate mesh coverage. In line 8 a new object of the *subDomain* type is created. It starts in the 13th row and the 17th column of the *\$domain* parent mesh and has 73 rows and 77 columns with a mesh size of 15×15 km. Because land use data will be processed, a new *Blanduse* object is created and the *domain* method executed indicating that data for the area covered by the *\$nest1* object should be calculated. In the next step we have to project the land use cover to Lambert, intersect it with the mesh cover and unload the polygon attribute table (method *intersect*). Next the area of each land use category in each mesh cell (line 13) is

```
1  #!/usr/bin/perl -e
2  use strict; use Subdomain;
3  use Blanduse; use Domain; use Bcompounds;
4  use DBGsfm; use DBdata;
5  use GISdata; use GIS;
6  my $domain = new Domain('MM5','domain',
7                          'gerafo',11,48,45,55,60000);
8  $domain->createDomain();
9  my $nest1 = new Subdomain($domain,'nest1',
10                           13,17,73,77,15000);
11 $nest1->createDomain();
12 my $bvoc = Blanduse->new();
13 $bvoc->domain($nest1);
14 my $patData = $bvoc->intersect();
15 my $lu = $bvoc->makeAreaLuse($patData);
16 my $ef = $bvoc->getEmissionFactors();
17 my $hp = $bvoc->getHelpFileData();
18 $bvoc->writeAreaFile($lu);
19 $bvoc->writeEffFile($ef);
20 $bvoc->writeHelpFile($hp);
```

Fig. 4. Perl script that creates data for input in the MCCM model.



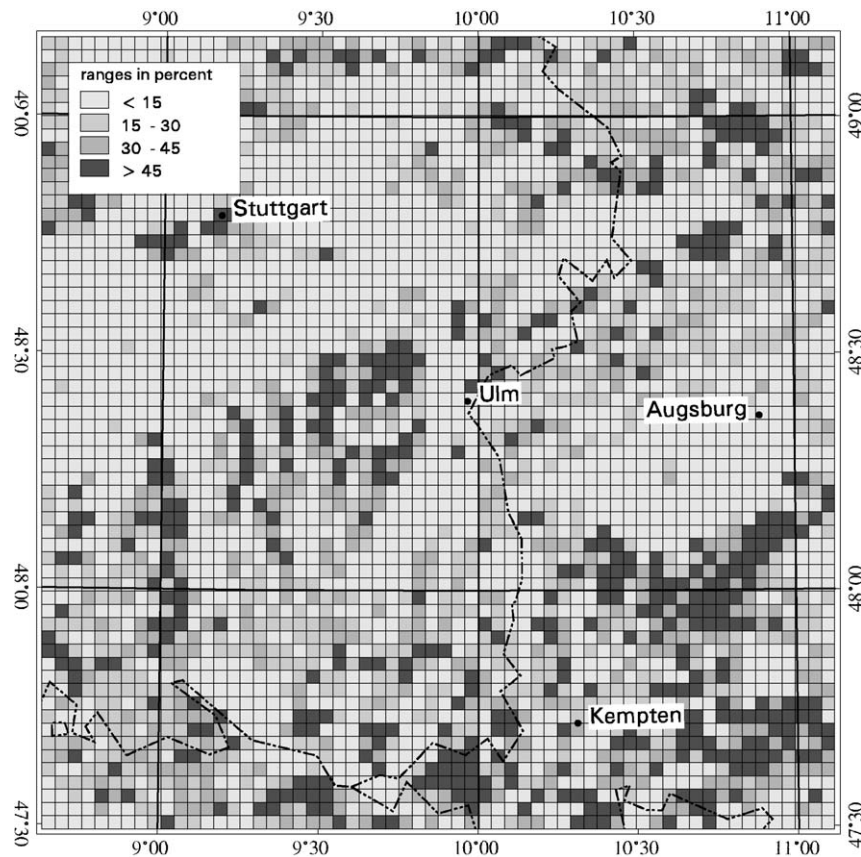


Fig. 5. Percentage of coniferous forest area calculated from CORINE land cover data (StaBuA, 1996). Model domain with 77 columns and 73 rows covering and the cell size  $3 \times 3$  km located over South Germany.

calculated and the emission factors (line 14) and an additional table relating the land use categories to more general vegetation types are retrieved from the relational database. This information is required to perform a VOC speciation within the MCCM model run. Finally three files are written to the local directory ready for use by the MCCM.

GSfM works reliably and it is easy to use. Fig. 5 shows an application example. For simplicity only the relative area of coniferous forest in each mesh cell is depicted.

There are, however, some considerable limitations:

- GSfM can only be operated on a host with direct access to the GIS and thus it is limited to the operation system of the host.
- Any changes in the Domain/Subdomain design require an additional consultation of the GIS department as new data have to be created.
- The access to the RDBMS over a network can only be provided if appropriate client database software is available on the host. (This is not only additional administrative work, in many cases the client software might simply be not available.)
- The data requester cannot extend the interface, e.g. in order to change the data format.

SOAP-based Web services help to overcome those limitations. They allow the creation of an environment in which GSfM can be used on any platform by accessing the GIS and RDBMS on remote hosts.

### 2.3. Web service

SOAP is a well-documented wire protocol specification for invoking methods on servers, services, components and objects using XML (Extensible Markup Language) and HTTP (Hyper Text Transport Protocol) as the method invocation mechanism. Its specification was published in the year 2000 (Box et al., 2000). SOAP administrates an XML vocabulary that is used for representing method parameters, return values, and exceptions. It provides a very practical way to communicate over the Internet and is easy to implement on any platform. SOAP implementations exist for Java, C/C++/Perl or Python. Here we will consider the SOAP::Lite Perl library. This library handles both client and server functions for SOAP communications. The word 'Lite' in the name refers to the simple interface it provides, not to its capabilities. A short SOAP::Lite tutorial is given by Kulchenko (2001).

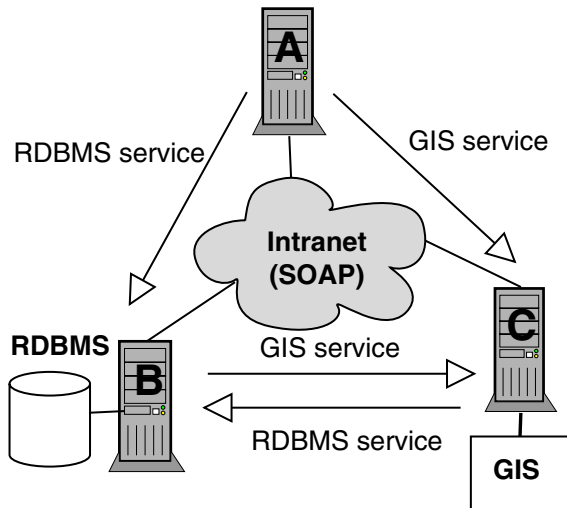


Fig. 6. SOAP service structure. The host B provides a Web service giving access to the data base and the host C provides a Web service giving access to the GIS. Host A can use both services over the network.

Consider a host structure consisting of three platforms as shown in Fig. 6. Host C is a Unix system and hosts the GIS software. Host B is a Linux system and it hosts the RDBMS software and the database. Host A is a MS Windows NT and has neither GIS nor RDBMS client software. Hence the host C will complain about the not available Perl::DBI interface to the RDBMS. The host B will miss the GIS and the host A will miss both RDBMS and GIS. The presented interface will not run on any of those hosts. Two SOAP services will solve the problem. The first SOAP service is implemented on the HOST C. It provides access to the GIS via the *GIS* class. The second service resides on the host B and provides access to the database via the *DBgsfm* class. Now the same Perl code can be executed on UNIX, Linux or MS-NT. If executed on A the system will invoke all RDBMS related methods on the remote host B and all GIS related methods on host C.

Fig. 7 shows how straightforward it is to define a SOAP client and invoke a remote method, here the *createDomain* method of the *GIS* class. The operator *use* introduces the *SOAP::Lite* and the *Domain* modules. Then an object *\$domain* of the *Domain* type is created as previously described. The client variable *\$client* initializes the SOAP service. The *URI* method creates the namespace with the *GIS* module. URIs are unique identifiers. Due to their namespace SOAP classes of the same name but from different hosts can clearly be distinguished. The *PROXY* method contains the URL (Uniform Resource Locator) to the SOAP service. In the example only the name and port of the host publishing the Web service are used. *\$gis* is an object of the *GIS* type. Now any program that requires a method of the *GIS* class will invoke it on the host *myserver* regardless of the host on which it is executed.

```

1 #!/usr/bin/perl -w
2
3 use SOAP::Lite;
4 use Domain;
5 my $domain = new Domain('MM5', 'domain',
6                           'ger', 11, 48, 45, 55, 60000);
7 my $client = SOAP::Lite
8   $client->uri('GIS');
9   $client->proxy('http://myserver:8081');
10  $client->transport->timeout(500);
11 my $gis = $client->call(new =>)
12             ->result;
13 my $resp = $client->
14             createDomain($gis, $domain);
  
```

Fig. 7. Client that accesses the *GIS* class.

It is also simple to write the server code (see Fig. 8). Here, the server is created as a stand-alone SOAP server that runs on its own port. However, it is also possible to implement the service on an existing Apache server and to run the SOAP service like a Common Gateway Interface (CGI) program. After introducing the SOAP and GIS modules using the *use* operator a new object of the SOAP class is initialized. It uses the HTTP protocol and listens at the port 8081. The list of classes that are exposed to the SOAP service is given to the *dispatch\_to()* method. In order to maintain control over classes published by the SOAP service one should provide the class name as an argument. However, it is also possible to dispatch dynamically all classes residing in one or more directories. The method *handle* invokes the service. In a similar way the service providing access to the RDBMS is constructed. The server resides on the host B and dispatches to the *DBgsfm* class.

### 3. Results and discussion

#### 3.1. Access to GIS

The *GSfm* class collection provides tools to write simple Perl scripts able to create input data tailored to the needs of the MCCM model at any platform with a working Perl environment. The *GIS* and *DBgsfm* classes check environmental variables pointing to the available GIS and RDBMS. If GIS or RDBMS is not available on the specific platform appropriate Web services are used. If any change in the model outline is

```

1 #!/usr/bin/perl -w
2 use SOAP::Transport::HTTP;
3 use GIS;
4 $daemon = SOAP::Transport::HTTP::Daemon
5   -> new (LocalPort => 8081)
6   -> dispatch_to('GIS')
7   ;
8 $daemon->handle;
  
```

Fig. 8. Server definition dispatching to the *GIS* class.

required the user can adjust his Perl script and start the data compilation again.

SOAP::Lite also provides a full range of error and exception handling methods that might be required to handle errors occurring during access of a SOAP service. Their application is described by Kulchenko (2001). Here, we shall consider some issues related to the GIS/RDBMS. The GIS can only handle persistent objects stored on disk. Thus, appropriate files, coverages, projected and intersected coverages must be created. Two problems might result:

Firstly, depending on the hardware, the projection of a land use coverage from i.e. geographic to Lambert and intersection with a *Domain* or *Subdomain* coverage will take some time. High resolution data might even require hours of computation. The system will issue a time-out error after some time of network inactivity while waiting for the GIS task to be completed. To prevent time-out errors the time period required for GIS and RDBMS activity must be extended using the *timeout()* method. This problem is no simple to solve as the time required to complete the GIS related tasks also depends on the current load of the host where the server is implemented.

Secondly, in order to create coverages on disk and intersect them, GIS-scripts or appropriate commands must be executed at the operating system level. This might introduce some serious security risks to the entire system. Therefore, only execution of scripts written by appropriate methods provided by the *GIS* class is allowed. Thus, the user cannot write his own GIS and RDBMS scripts and execute them using the presented Web services.

SOAP is a project similar to CORBA (Common Object Request Broker Architecture)(OMG, 2002) or DCOM, the distributed extension to the COM (Component Object Model)(Box, 1999). The disadvantage of those technologies is that applications built on these protocols are often vendor dependent. SOAP is not related to any vendor allowing independent implementations. SOAP implementations for different languages and operating systems exists in both open source and proprietary areas. The major advantage of the SOAP::Lite library for Perl is its simple application. SOAP simplifies significantly XML-based communications (e.g. Kokkonen et al., 2003) and distributed grid-services (e.g. Mineter et al., 2003).

There are several land use data sets available for remote processing using the GSfM. For the area of Europe a default composite data set compiled from the CORINE (Coordination of Information on the Environment)(EU, 1992) project, PELCOM data base (Mücher et al., 2000) and global land cover characteristics (GLCC) (U.S. Geological Survey, 2002) is available. The land use categories of the CORINE and PELCOM data were transferred USGS nomenclature used in the MM5-Model. For regional applications in central

Europe CORINE data with 250×250 m resolution and 44 different land use categories can be used.

### 3.2. Data base access

The GSfM Web service simplifies significantly the selection of tabular data from the relational data base. In the Perl script there is only one line (line 14 in Fig. 4) that returns the reference to a table containing the emission factors for the land use categories of the data set used. Without knowledge of the data base structure and SQL (Structured Query Language) as well as without a data base account the user gets the latest version of the data. In the background, however, SQL code with complex constructs, such as *union* involving several different tables, is executed. Depending on the type of the requesting object emission factors for isoprene,  $\alpha$ -pinene, other monoterpenes and other VOC are retrieved if the object is of type *Blanduse*. For objects of type *Bplantspecies*, emission factors for compounds and compound groups forming the RADM2 (Regional Acid Deposition Model)(Stockwell et al., 1990). Chemical mechanism used in the MCCM will be provided. In addition, for *Blanduse* objects, a table relating the land use categories to more general vegetation types, such as forests, agriculture or natural vegetation, is retrieved from the database. This information is used in MCCM to split the general BVOC emissions described as other monoterpenes and other VOC into the required RADM2 categories. Unlike GIS service the RDBMS service communicates via DBI::module directly with the data database. Therefore SQL scripts are neither written onto disk nor executed.

### 3.3. Extension of the GSfM capabilities

The GSfM user has also the possibility to enhance the system. In that case some knowledge of Perl is required. He can create a new class which inherits the *Blanduse* or *Bplantspecies* and define his own methods or overload any of the available methods. Simple changes to the output data format can directly be accommodated in the executing Perl script. As the reference to the array of data (\$lu, \$ef and \$hp, see Fig. 4) is available one can skip the GSfM *write...File* methods, re-arrange the data and write the result to disk.

## 4. Conclusions

SOAP-based Web services are excellent means to create platform neutral applications that facilitate a fast provision of data tailored to specific needs of the users. In addition they allow an easy implementation of extensions to GSfM while taking advantage of the powerful GIS and RDBMS and the data. For internal

deployment of Web services on private corporate networks making geographic information and databases developed in the GIS/RDBMS department widely available, SOAP is a very reasonable approach. Due to the implemented Web services the presented GIS/RDBMS interface runs on almost any host that is connected to the corporate network and the MCCM users are able to create the required input for BVOC modeling at their specific platform.

## References

- Box, D., 1999. House of COM. *Microsoft Systems Journal* 14 (9).
- Box, D., Ehnebuske, D., Kakivaya, G., Layman, A., Mendelsohn, N., Nielsen, H., Thatte, S., Winer, D., 2000. Simple Object Access Protocol (SOAP) 1.1. Available from: <http://www.w3.org/TR/SOAP/>.
- Conway, D., 1999. *Object Oriented Perl*. Manning Publications 512 pp.
- EU, 1992. CORINE Land Cover. 22 pp.
- Fuentes, J.D., Hayden, B.P., Garstang, M., Lerdau, M., Fitzjarrald, D., Baldocchi, D.D., Monson, R., Lamb, B., Geron, C., 2001. New Directions: VOCs and biosphere-atmosphere feedbacks. *Atmospheric Environment* 35 (1), 189–191.
- Grell, G.A., Emeis, S., Stockwell, W.R., Schoenemeyer, T., Forkel, R., Michalakes, J., Knoche, R., Seidl, W., 2000. Application of a multiscale, coupled MM5/Chemistry Model to the complex terrain of the VOTALP Valley Campaign. *Atmospheric Environment* 34, 1435–1453.
- Guenther, A., 1997. Seasonal and spatial variations in the natural volatile organic compound emissions. *Ecological Applications* 7 (1), 34–45.
- Guo, Y.-R., Chen, S., 1994. Terrain and Land Use for the Fifth-Generation Penn State/NCAR Mesoscale Modeling System (MM5): Program TERRAIN, NCAR Technical Note NCAR/TN-397+IA, 114 pp.
- Kokkonen, T., Jolma, A., Koivusalo, H., 2003. Interfacing environmental simulation models and databases using XML. *Environmental Modelling & Software* 18, 463–471.
- Kulchenko, P., 2001. SOAP::Lite for Perl. Available from: <http://www.soaplite.com>.
- Mineter, M.J., Jarvis, C.H., Dowers, S., 2003. From stand-alone programs towards grid-aware services and components: a case study in agricultural modelling with interpolated climate data. *Environmental Modelling & Software* 18, 379–391.
- Mücher, C.A., Steinnocher, K., Kressler, F., Heunks, C., 2000. Land cover characterization and change detection for environmental monitoring of pan-Europe. *International Journal of Remote Sensing* 21, 1159–1181.
- OMG, 2002. Common Object Request Broker Architecture: Core Specification. 1166 pp. Available from: <http://www.omg.org>.
- Richter, K., Knoche, R., Schoenemeyer, Th., Smiatek, G., Steinbrecher, R., 1999. Abschätzung biogener Kohlenwasserstoff-Emissionen. *Umweltwissenschaften und Schadstoff-Forschung. Zeitschrift für Umweltchemie und Ökotoxikologie* 10 (6), 319–325.
- Simpson, D., Winiwarter, W., Brjesson, G., Cinderby, S., Ferreira, A., Guenther, A., Hewitt, N., Janson, R., Kahlil, M.A.K., Owen, S., Pierce, T., Puxbaum, H., Shearer, M., Skiba, U., Steinbrecher, R., Tarrason, L., Öquist, M.G., 1999. Inventorying emissions from nature in Europe. *Journal of Geophysical Research* 104, 8113–8152.
- Smiatek, G., 2001. GIS and RDBMS Class System in Support of BVOC Emission Inventories. *Proceedings of the 5th. GLOREAM Symposium, Wengen, Switzerland, September 24–26, 2001*. Available from: [http://people.web.psi.ch/keller\\_j/GLOREAM/WS2001/Final/Table\\_of\\_contents.html](http://people.web.psi.ch/keller_j/GLOREAM/WS2001/Final/Table_of_contents.html).
- StaBuA, 1996. Statistisches Bundesamt. Daten zur Bodenbedeckung. CORINE Land Cover. Datensatz auf CD-ROM, Wiesbaden.
- Stewart, E.H., Hewitt, C.N., Bunce, R.G.H., Steinbrecher, R., Smiatek, G., Schoenemeyer, T., 2003. A highly spatially and temporally resolved inventory for biogenic isoprene and monoterpene emissions — model description and application to Great Britain. *Journal of Geophysical Research* 108 (D20), 4644. doi: 10.1029/2002JD002694.
- Stockwell, W.R., Middleton, P., Chang, J.S., Tang, X., 1990. The second generation regional acid deposition model chemical mechanism for regional air quality modeling. *Journal of Geophysical Research* 100, 22907–22928.
- U.S. Geological Survey, 2002. Global Land Cover Characteristics Data Base. Available from: [http://edcdaac.usgs.gov/glcc/globdoc2\\_0.html](http://edcdaac.usgs.gov/glcc/globdoc2_0.html).