2nd GENERATION OF RSL'S SPECTRUM DATABASE "SPECCHIO"

A. Hüni*, J. Nieke, J. Schopfer, M. Kneubühler and K. I. Itten

Remote Sensing Laboratories, University of Zurich, Winterthurerstrasse 190, 8057 Zurich, Switzerland. - (ahueni, nieke, jschopf, kneub, itten)@geo.unizh.ch

KEY WORDS: Databases, Data Structures, Metadata, Spectral, Software

ABSTRACT:

The organised storage of spectral data described by according metadata is important for long term use and data sharing with other scientists. The recently redesigned SPECCHIO system acts as a repository for spectral field campaign and reference signatures. An analysis of metadata space has resulted in a non-redundant relational data model and efficient graphical user interfaces with underlying processing mechanisms minimizing the required user interaction during data capture. Data retrieval is based on imposing restrictions on metadata space dimensions and the resulting dataset can be visualised on screen or exported to files. The system is based on a relational database server with a Java application providing the user interface. This architecture facilitates the operation of the system in a heterogeneous computing environment.

1. INTRODUCTION

Ground based hyperspectral signatures are collected for (a) calibration and validation of airborne or spaceborne imagery and its data products, (b) feasibility studies for airborne/spaceborne missions, (c) basic investigation of the relationship between physical or biochemical properties and the electromagnetic reflectance of objects and (d) definition of directional dependence of the reflectance of objects on the illumination and viewing geometry. There are no standardisations of the acquisition process of ground spectral signatures. As a result sharing spectral signature datasets with other scientists is complicated due to differences in data collection techniques and sampling environment conditions (Pfitzner et al., 2006).

Spectral ground sampling campaigns result in significant amounts of data both in number of sampled wavelengths and collected spectra. For efficient research such data need to be documented by metadata and stored in an organized way. This serves three purposes: (a) to ensure the usability of collected data in long-term, (b) to provide other scientists a means of assessing the suitability of a third party dataset for their own research and (c) to enable the retrieval of spectral data based on metadata queries. A relational database seems a natural choice of technology in this respect. However, only two implementations of such systems are currently known: (a) SPECCHIO (Bojinski et al., 2003) and (b) SpectraProc DB (Hueni & Tuohy, 2006).

Experience with the first version of SPECCHIO (Bojinski et al., 2003) has shown that the success of such a system is highly dependant on its utilization by the users. Many researchers were deterred from entering their data into the spectral database due to suboptimal data capturing system interfaces. It has become clear that in order to be successful a spectral database system must (a) provide added value to the user and (b) minimize the manual data input as much as possible by automated metadata generation.

SPECCHIO is used at RSL to (a) store spectral and metadata in a central repository which is accessible to all members of the laboratory, (b) serve as a spectral data source for various calibration/validation and simulation tasks and (c) provide parameters for APEX level 2/3 processing (Schlaepfer & Nieke, 2007 (in preparation)). Due to shortcomings of the first SPECCHIO version in terms of user friendliness and inconsistencies in the data model a redesign was undertaken with the goal to provide a system for non-redundant, centralised and efficient entry, storage and retrieval of spectral data and associated metadata.

2. METHODS

2.1 Spectral, Feature and Metadata Space

A sampled object has a spectral attribute and arbitrary non-spectral attributes, the so called metadata.

The spectral attribute can be interpreted in two principal ways: (a) as a spectral signature in spectral space or (b) as a point in an n-dimensional feature space (Landgrebe, 1997). The sampling process by the spectroradiometer leads to a discretization of the continuous spectral response of the object. Every sampling channel yields a quantitative value which forms a component of the output vector. Spectral signatures represent the spectral response of the sampled object. An object in spectral space is visualized by plotting the spectral response vector against the channel wavelengths. The curve is then again interpreted as a continuous function.

The feature space concept explicitly utilizes the discrete data as produced by the sampling process. Every object is represented by a point in an n-dimensional space, the so called feature space and its position is given by the spectral vector. The dimensionality of the feature space is given by the number of channels of the sensor. The transformation of continuous spectral space into discrete feature space is defined by the spectral response functions of the sensor elements.

Metadata are essentially descriptive data about a resource. In the case of spectral data the resource is the spectral response of an object and the metadata contains further information about the object and the sampling environment at the time of data capture. Metadata spaces are n-dimensional spaces defined by descriptive dimensions. The space is most efficiently described

* Corresponding author.

by orthogonal vectors, i.e. the dimensions are independent of each other (Wason & Wiley, 2000).
In the example of SPECCHIO the metadata vector contains four types of variables: (a) quantitative, (b) categorical (qualitative), (c) alphanumeric string and (d) pictorial.
Quantitative variables are gained from measurements of quantitative features of the sampled object or the surrounding environment, e.g. spatial position, ambient temperature or capturing time.
Categorical variable values are assigned to objects on the basis of a priori knowledge. Examples are: landcover type, species, arbitrary sampling site number or sampling location name.
Alphanumeric strings are used to hold textual descriptions. They do not contain information in a structured way but can help the user in understanding the data. In the context of metadata space alphanumeric string variables neither form clusters nor do they group data in any organised way. Strings dimensions are searchable via full text search or can be crawled and indexed previous to queries.
Pictorial variables can hold supplementary information about the sampled object or its environment in form of images, e.g. photos of sky (hemispherical), sampling setup or target. Pictorial variables have the potential of yielding quantitative or qualitative data if subjected to image analysis or image indexing techniques. This is however not further investigated at this point.
Typically, quantitative variables show a high degree of variability while categorical variables concentrate the data into the available classes. This is illustrated in Figure 1 showing two 2D subspaces of the metadata space.

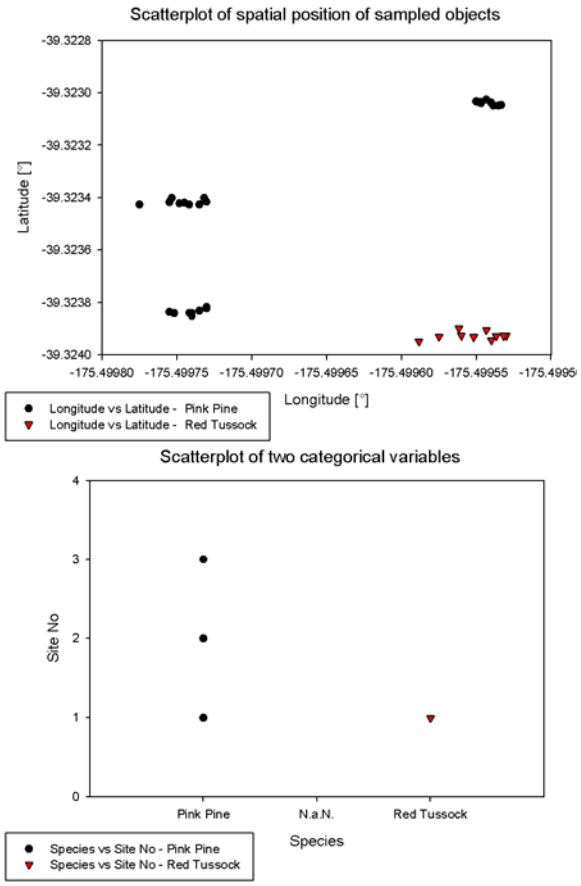

Figure 1: Scatterplots of quantitative (top) and categorical variables (bottom)

The upper plot shows the variability of the spatial position of the samples and the lower plot the grouping function of the categorical variables 'species' and 'site number'. The site number is arbitrarily defined and refers to the spatial region where an object was sampled. For the Pink Pine species there exist three sample sites. In the spatial subspace the sampling positions form clusters that correspond to these sample sites. In the categorical subspace all data points fall into the positions defined by species and site number categories. Two points are worth noting:
(a) Quantitative variables contain individual values per sample. The acquisition of quantitative metadata has the potential of automation by the use of electronic instruments, e.g. GPS or automated weather stations. Setting quantitative variables of a group of samples to the same value (e.g. an average value) is possible if the introduced error is acceptable.
(b) Categorical variables group the data and one value will apply to many samples. Thus the values must not be entered individually per sample but can be set for whole sample groups. Categorical variables lend themselves to data structuring due to the grouping function in categorical subspaces.

Table 1 lists the metadata variables and their data type on spectrum level according to the SPECCHIO data model. The data types are abbreviated as follows: C (Categorical), Q (Quantitative), S (String) and P (Pictorial). The 'Autom.' column lists the possibility of automatic retrieval or calculation: SF (Spectral File) and CA (Calculation).
Table 2 lists further metadata that is relevant at campaign level.

Table 1: Metadata on spectrum level

Metadata variable	**Type**	**Autom.**
Auto number	C	SF
User comment	S	SF
Capturing date and time	Q	SF
Spectral file name	S	SF
Number of spectra averaged internally by the instrument	Q	SF
Sensor	C	SF
File format	C	SF
Instrument	C	SF
Instrument calibration number	C	SF
Foreoptic	C	SF
Illumination source	C	
Sampling environment	C	
Measurement type (single, directional, temporal)	C	
Measurement unit (Reflectance, DN, radiance, absorbance)	C	SF
Target homogeneity	C	
Spatial position (latitude, longitude, altitude)	Q	SF
Landcover (based on CORINE land cover (European Commission DG XI, 1993))	C	
Cloud cover (in octas)	C	
Ambient temperature	Q	
Air pressure	Q	
Relative humidity	Q	
Wind speed (Qualitative description)	C	

Wind direction (categories in 45 degree steps)	C	
Sensor zenith angle	Q	CA (Goniom.)
Sensor azimuth angle	Q	CA (Goniom.)
Sensor distance	Q	
Illumination zenith angle	Q	CA (Sun pos.)
Illumination azimuth angle	Q	CA (Sun pos.)
Illumination distance	Q	
Spectrum names	C	
Target type	C	
Goniometer model	C	
Pictures	P	
Data structuring information	C	Gleaned from folder structures

Table 2: Metadata on campaign level

Metadata variable	**Type**
Investigator	C
File path to spectral data on file system	S
Campaign comments/description	S

2.2 Structuring of Spectral Campaign Data

Spectral sampling campaigns yield spectral signatures of objects. Physically, spectroradiometers produce files containing digitized spectral signatures of the sampled objects with usually one file being created per reading. The sheer number of files resulting from sampling campaigns requires an organised method of storage.

Structuring of data is achieved by sorting them according to discriminating metadata attributes. Categorical metadata variables have a grouping feature that is ideally suited for data structuring. Qualitative attributes could also be used for data structuring given that they are transformed into categorical variables first by some classification. In the process of spectral sample data structuring the values of all utilized metadata variables will be implicitly recorded in some form in the resulting structure.

The example introduced in the previous section consists of samples belonging to two different species (Pink Pine and Red Tussock). For the Pink Pine three sample sites exist while Red Tussock has been sampled at only one site. A first level of structuring could use the species variable resulting in a one dimensional hierarchy. By adding the sampling site number as further structuring criterion a two dimensional hierarchy is defined.

These hierarchies can be directly implemented with an according hierarchical folder structure that is subsequently used to store the spectral signatures (cf. Figure 2).

Three points are worth noting:

(a) Data structures implicitly contain metadata information. Metadata can be gleaned from directory structures by the creating agent when the resource is created (Gill et al., 1998). This information can be redundant or complementary in relation to the metadata attributes stored in the database.

(b) The number of dimensions used to structure data can range from zero (flat data structure) to N where N is equal to the number of available categorical metadata variables. Obviously, by the use of a sampling protocol, files can still be tied with their metadata even when using a flat data structure. However, using a folder structure based on metadata parameters to store field data in first place facilitates the data handling at later stages.

(c) Hierarchies are useful to store and retrieve information. Their limitation is the fixed path one has to follow to retrieve data (Wason & Wiley, 2000). The ordering of the levels usually depends on some logical model the cataloguer has of the data.

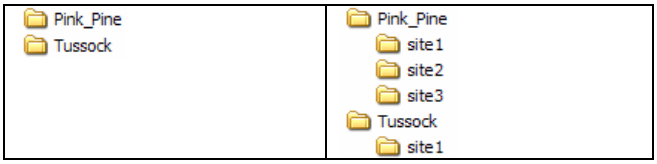

Figure 2: One dimensional hierarchy (left) and two dimensional hierarchy (right)

Data structures always reflect the way the user thinks about the data. Ultimately the structure will depend on the scientific problem to be solved. The data structure is therefore important to the understanding of the data and thus should be stored in the database along with other metadata information. Data structures also facilitate the handling of data and should be utilised in graphical user interface.

2.3 Metadata Generation

The success of a system using metadata relies on the input of such data in first place. If metadata entry involves too much effort users can be deterred from entering their data into the system, thus rendering it useless. The automation must therefore be a prime focus of metadata input.

Many spectroradiometer output files contain a range of parameters that can be directly stored in metadata parameter fields. By conversion to other file formats, e.g. ENVI SLB, some of this information can be lost and subsequently require manual data capture. It is therefore important to support and use native spectroradiometer file formats whenever possible.

While some metadata is available from the input files other data may be generated computationally, e.g. (a) sun geometry and (b) goniometer viewing geometry for the FIGOS and LAGOS goniometers (Schopfer et al., 2007 (in print)).

The importance of automated metadata generation has been pointed out above. Table 1 lists the metadata variables that can be automated. Out of 34 variables, 12 can be read from the input file (for the example of an ASD binary file with a GPS device connected to the field laptop), 1 can be gleaned from the directory structure (data structuring information), 4 can be calculated (illumination geometry and sensor geometry for outdoor sampling and goniometer experiments respectively) and 17 need to be captured manually. Of these remaining 17, three are quantitative variables related to the sampling environment and could be automated using an electronic weather station, 14 are categorical variables and one is pictorial. Categorical variable values apply to groups of spectra in many cases and their data entry can thus be carried out via group operations minimising the needed user interaction.

2.4 Database Model

An optimal data model stores the data in a set of small, stable tables. Complex user views are reduced to such models by the process of database normalisation (McFadden & Hoffer, 1988).

The engineering of the database model had three prime goals: (a) removal of data redundancy, (b) minimizing needed data entry by the user and (c) providing high repeatability for categorical parameters. These issues are actually coupled: a non redundant system will implicitly reduce the required data input and store categorical variables in separate tables. The content of categorical parameters should be one entry out of a well defined set of possible values. Repeatability is the ability to have the

same resource described in the same way on two or more occasions (Wason & Wiley, 2000). For categorical variables the repeatability is therefore increased by providing these values in a separate table, thus defining the possible set and restricting the data access to read only for normal users. Data capturing effort is minimized by reducing the action to the selection of one value out of the set.
An example is the cloud cover in octas: the nine possible classes are by definition the only values the cloud parameter can assume. A spectrum will refer to the cloud cover by a foreign key, thus restricting the possible set of values to the predefined ones.
Consider the example of environmental conditions of a sampling site. The parameters involved are: cloud cover, wind speed, wind direction, humidity, air pressure and temperature. If spectra of a sampling site are collected in a short time frame it is unlikely that the environmental variables will change significantly. They thus apply to all collected spectra. Given the normalized data model, these variables are entered only once and all involved spectra are referencing this single entry by foreign keys.

2.5 System Architecture

The core of the SPECCHIO system is a MySQL database (MySQL AB, 2005) hosted on a database server (cf. Figure 3). The SPECCHIO application was implemented as a Java 2 (Sun Microsystems Inc., 2006) application and is therefore operating system independent which is of importance in a heterogeneous computing environment. The application thus runs on any machine with a Java Virtual Machine (VM) installation and connects to the database via TCP/IP on standard port 3306.
The application can also be run remotely from a terminal on a server by the use of the X11 protocol.
The spatial aspect of the data sets offers the possibility for direct linkage with a GIS system. For the example of ArcGIS (ESRI, 2006) a database connection is established via ODBC (Open Database Connectivity).

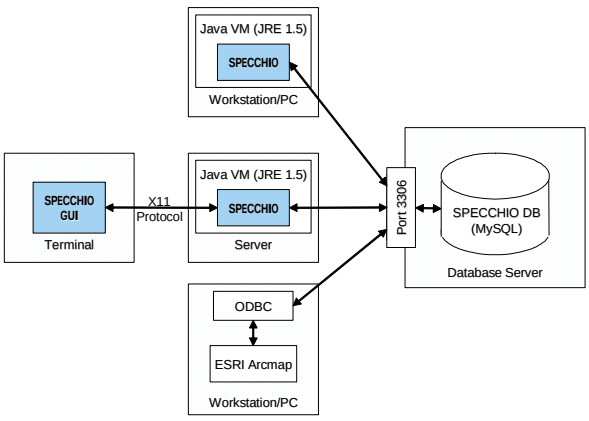

Figure 3: SPECCHIO system architecture

3. RESULTS

The resulting system is documented hereafter by describing the workflow from the loading of a new campaign till the data retrieval.

3.1 Campaign Data Loading

The definition of a new sampling campaign requires only the definition of a campaign name and the specification of the top directory, i.e. the root of the hierarchical structure.
Once a campaign is defined its data is loaded by a single mouse click. The hierarchy of the campaign is automatically parsed. The folder structure is stored in entries in the hierarchy table and spectral files are read and data filled into the appropriate tables.
In the case of new data being added to an existing campaign, the loading process can be started again and will only load the new information into the database.
Supported file formats are: ASD binary (Analytical Spectral Devices Inc.), GER signature (Spectra Vista Co., 2005), ENVI Spectral Library (Research Systems Inc., 2005) and MFR OUT (Yankee Environmental Systems Inc., 2000).

3.2 Metadata Editing

Once data of a campaign have been loaded, their metadata can be edited by using the Metadata Editor (cf. Figure 4). The structure of the campaign is visualized by a tree structure (lower left in Figure 4). Selection of the data to be edited happens via this tree. Three tabs (right side in Figure 4) hold the metadata fields of the campaign, the hierarchy and the spectrum. The content of the fields reflects the current selection in the tree. Multiple updates are possible by selecting multiple hierarchies and/or spectra. A metadata conflict detection is executed for multiple selections and only non-conflicting metadata parameters can be updated, e.g. if every spectrum in a selection already refers to a different spatial position editing will be disabled for the position.
Categorical variable values are selected from combo boxes that are pre-filled from the database. Quantitative variables are entered into fields restricted to numerical values.
Each of the three tabs has associated reset and update buttons which will restore the previous values or commit the changes to the database respectively.
Further functionality includes: highlighting of mandatory fields according to the chosen metadata quality level, indication of missing metadata in the selection tree and overriding of the conflict detection.

3.3 Data Queries and Output

Data is queried by using the Query Builder. Two operational modes are supported: (a) direct selection of records by using a tree structure (browsing) and (b) specification of query conditions (metadata space constraints).
Restriction in several metadata dimensions is achieved by a logical AND of the constraints per dimension.
Figure 5 shows a spectrum report window with a spectral plot on the left side and metadata attributes listed on the right side. Data can be written as CSV (Comma Separated Values) and ENVI Spectral Library files.

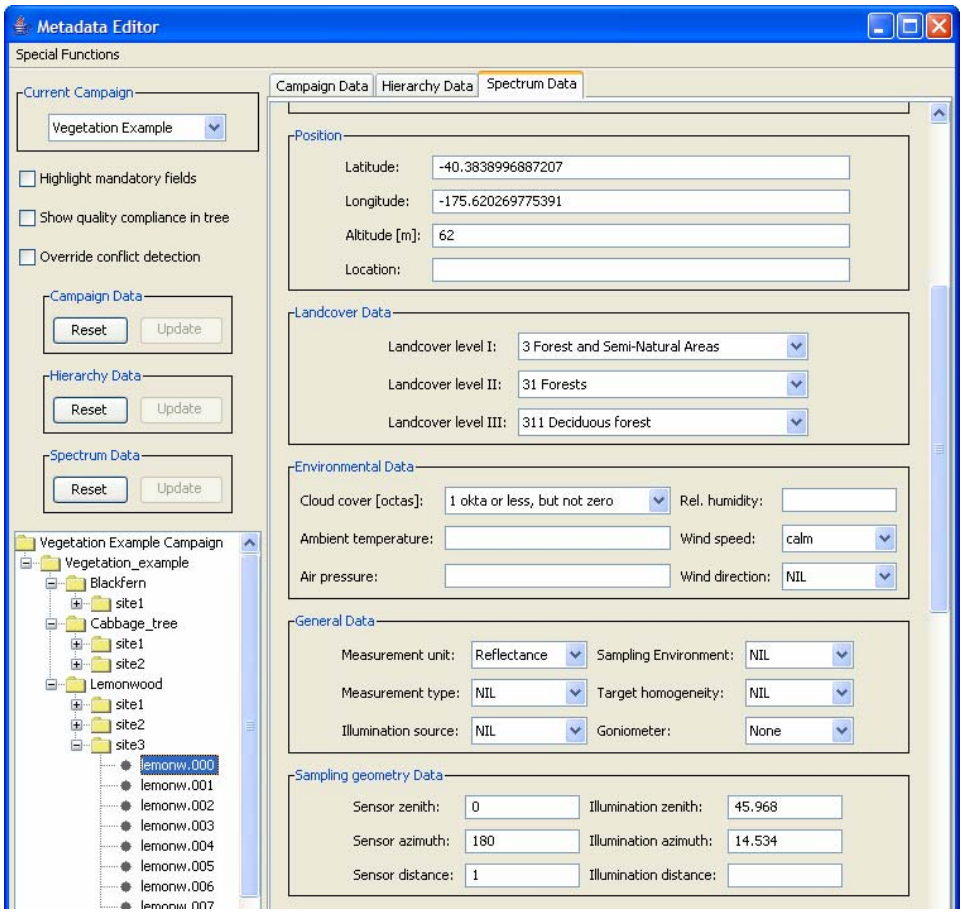

Figure 4: SPECCHIO metadata editor

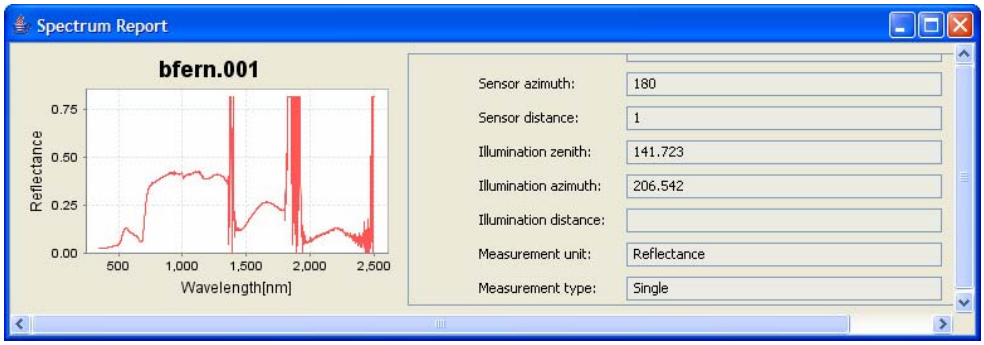

Figure 5: SPECCHIO spectrum report window

4. DISCUSSION

4.1 Spectral Databases

Relational databases are well suited to store spectral data and associated metadata. An issue to consider is the amount of data to be expected. Imaging spectroscopy produces high volumes of data where several hundred gigabytes per campaign may be acquired. SPECCHIO is used to store existing spectral library data and spectral signatures collected during field and laboratory experiments. The number of spectra in commonly available spectral libraries, e.g. USGS spectral library (Clark et al., 1993), is limited by the number of represented materials and the number of spectra per material where the latter is usually one. Field or laboratory campaigns on the other hand can yield several hundred to around thousand spectra per day per sampling instrument. The spectral information, apart from pictorial metadata, requires the most storage space. The maximum effective table size for MySQL databases is dependent on the operating system and varies between 2GB and 16TB (MySQL AB, 2007). With an average value of 3.5 kilobyte per spectrum storage problems are not expected.

4.2 Data Model

The data model used for SPECCHIO includes metadata already suggested by other authors (Bojinski et al., 2003; Pfitzner et al., 2005). Extensions to the model are expected. Some new requirements might be defined during practical use of the system while others have already been identified such as the documentation of white references. Reference panels show degradations over time. A frequent monitoring of the spectral properties of the panels and checking against national standards is highly recommended. The history of white references should be included in spectral databases and thus should be part of the data model. This would provide (a) basic information for the

assessment of spectral data quality and (b) allow for the correction of spectra based on ratios between used panel and national standard.

4.3 Processing Functionality

The processing functionality is currently restricted to metadata operations like sun angle calculation and automated white reference linking.
The main interest of the user is still the spectral data and spectral processing functions are a key factor to the efficient study of hyperspectral data (Hueni & Tuohy, 2006). Possible functions include: waveband filtering (removal of noisy spectral regions), smoothing, sensor convolution, derivatives, feature space transformations (e.g. principal components transformation or MNF), spectral mixing, BRDF (Bidirectional Reflectance Distribution Function) retrieval and statistics. Conceptually different solutions would be possible: (a) adding processing functionality to the SPECCHIO software in form of new Java code, (b) calling external routines, e.g. IDL (Research Systems Inc., 2006) or (c) embedding the SPECCHIO database into a suitable processing package.

5. CONCLUSIONS

The second generation of the SPECCHIO system has introduced significant improvements in non-redundant data storage, automated loading of sampling campaigns, documentation of hierarchical campaign structures, extraction of metadata from spectroradiometer files, high repeatability of data entry for categorical variables due to the definition of the possible value sets in the database, efficient metadata entry by group updates and intuitive, flexible and fast data query and output.
The combination of a central database server with the implementation of the user interface and underlying processing functionality in Java allow the utilisation of the system in a heterogeneous computing environment.
Future work includes the validation and possible extension of the data model, the inclusion of spectral processing functionality or embedding of the system in other tools or processing chains and evaluation of methods for the assessment of data quality.
For further information please refer to:
http://www.geo.unizh.ch/rsl/research/SpectroLab/projects/specchio_index.shtml (SPECCHIO Website).

REFERENCES

Analytical Spectral Devices Inc. Technical Guide. http://www.asdi.com/tg_rev4_web.pdf (accessed 20. Mar., 2007)

Bojinski, S., Schaepman, M., Schlaepfer, D. & Itten, K., 2003. SPECCHIO: a spectrum database for remote sensing applications. *Computers & Geosciences*, 29, 27-38.

Clark, R. N., Swayze, G. A., Gallagher, A. J., King, T. V. V. & Calvin, W. M., 1993. The U. S. Geological Survey, Digital Spectral Library: Version 1: 0.2 to 3.0 microns, *U.S. Geological Survey Open File Report*, Vol. 93, pp. 1340.

ESRI. 2006. ArcGIS (V. 9.2). Redlands, CA

European Commission DG XI. 1993. *CORINE land cover*: European Commission Directorate-General Environment, Nuclear Safety and Civil Protection, Office for Official Publications of the European Communities, Luxembourg.

Gill, T., Gilliland-Swetland, A. & Baca, M., 1998. *Introduction to Metadata: Pathways to Digital Information*, Getty Research Institute.

Hueni, A. & Tuohy, M., 2006. Spectroradiometer Data Structuring, Pre-Processing and Analysis - An IT Based Approach. *Journal of Spatial Science*, 51(2), 93-102.

Landgrebe, D., 1997. On Information Extraction Principles for Hyperspectral Data. West Lafayette, Purdue University.

McFadden, F. R. & Hoffer, J. A., 1988. *Database Management*. Redwood City, The Benjamin/Cummings Publishing Co.

MySQL AB. 2005. MySQL, http://www.mysql.com

MySQL AB. 2007. MySQL 5.0 Reference Manual. http://dev.mysql.com/doc/refman/5.0/en/index.html (accessed 16. Feb., 2007)

Pfitzner, K., Bartolo, R. E., Ryan, B. & Bollhöfer, A. 2005. *Issues to consider when designing a spectral library database.* SSC 2005 Spatial Intelligence, Innovation and Praxis: The national biennial Conference of the Spatial Sciences Institute, Melbourne.

Pfitzner, K., Bollhöfer, A. & Carr, G., 2006. A standard design for collecting vegetation reference spectra: Implementation and implications for data sharing. *Journal of Spatial Science*, 51(2), 79-92.

Research Systems Inc., 2005. ENVI (V. 4.2). Boulder, CO.

Research Systems Inc., 2006. IDL (V. 6.3). Boulder, CO.

Schlaepfer, D. & Nieke, J. 2007 (in preparation), 23-25 April. *Optimizing the Workflow for APEX Level2/3 Processing.* EARSeL Workshop on Imaging Spectroscopy, Bruges, Belgium.

Schopfer, J., Dangel, S., Kneubühler, M. & Itten, K. 2007 (in print). *Dual Field-of-View Goniometer System FIGOS.* ISPMSRS, Davos, Switzerland.

SPECCHIO Website. http://www.geo.unizh.ch/rsl/research/SpectroLab/projects/specchio_index.shtml

Spectra Vista Co., 2005. *GER 3700 User Manual* (3.2 ed.). New York.

Sun Microsystems Inc., 2006. JavaTM 2 Platform Standard Edition (V. 5.0). Santa Clara, CA

Wason, T. D. & Wiley, D., 2000. Structured Metadata Spaces. *Journal of Internet Cataloging*, 3(2/3), 263-277.

Yankee Environmental Systems Inc., 2000. *MFR-7 Rotating Shadowband Radiometer* (2.10 ed.). Turner Falls, MA.