

RHEINISCHE FRIEDRICH-WILHELMS-UNIVERSITÄT BONN

MASTERSTUDIENGANG GEOINFORMATIONSSYSTEM

TOWARDS A WEB COVERAGE SERVICE FOR EFFICIENT MULTIDIMENSIONAL INFORMATION RETRIEVAL

“ONE-STOP” EXPLORATIVE GEOANALYSIS SERVICING

Anup Deshmukh

Mumbai, India

Supervisor:

Prof. Dr. Klaus Greve

Geographisches Institut

Universität Bonn, Germany

October 2007

THE FOLLOWING MASTER THESIS WAS PREPARED ON MY OWN WITHOUT ANY ADDITIONAL HELP OTHER THAN THE MENTIONED SOURCES AND TOOLS. ALL USED SOURCES OF LITERATURE ARE LISTED AT THE END OF THE THESIS.

BONN, FRIDAY, OCTOBER 26, 2007

ANUP DESHMUKH

ACKNOWLEDGEMENT

Wernher Von Braun once said ***BASIC RESEARCH IS WHAT I AM DOING WHEN I DON'T KNOW WHAT I AM DOING.*** This has certainly been the case on many occasions when I didn't know what I was doing and what I had to do. I would like to take this opportunity to extend my sincere gratitude to individuals who showed me the path and made me see light at the end of the tunnel.

First of all, there are of course people who were directly involved in the research project. Dr. Angelika Voss – she has been inspirational. Her never ending enthusiasm for the work, reviewing new pieces I wrote almost overnight, and her insights have left a lasting impression. Vera Hernandez – this thesis is her brainchild. Thank you for all the support and advice. Without your initiative and belief in the work this thesis would not have been possible. To the people working at the Spatial Decision Support department at IAIS Fraunhofer – Thank You. It has been a pleasure.

My family has been a bastion of great support; Annegret and Dr. Dilip Gadkari – no matter where I am, I will always remember you. Shikha Dalmia, your understanding and unconditional love has helped me through some dark phases. Mom, Dad – I dedicate this thesis to you. You made it possible.

Prof. Dr. Klaus Greve – this acknowledgment cannot be complete without appreciating your support over the past two years. You have been phenomenal.

Bonn, Friday, October 26, 2007

Anup Deshmukh

ABSTRACT

Historically, Business Intelligence (BI) and Geographic Information Services (GIS) have followed separate development and implementation paths. Customer requests for a complete operational picture and the ability to be more proactive has led to the demand for a synergistic power that can be exploited by integrating BI and GIS. An integrated geographic business intelligence solution (GBIS), a term coined by (ESRI, 2005), enables users to both visualize and manage spatial information and empower decision makers, at different levels, with the location-based intelligence they need to assess, plan and deliver services, present information and deal with ad hoc business queries. The integrated solution improves decision-making and responsiveness while extending the reach of GIS to address a wider range of business solutions. The investigations for a GBIS solution led to the introduction of a new sub-category of spatial decision-making solutions: Spatial Online Analytical Processing (SOLAP) or Spatial OLAP. This study contributes to the development of the SOLAP domain by presenting an interoperable web-based open and extensible prototype solution with the analysis capabilities available in the two technologies. The prototypical solution is an integration based on the Web Coverage Service (WCS¹), as defined by the Open Geospatial Consortium (OGC²), and an OLAP (OnLine Analytical Processing) server. The author has extended an existing WCS implementation by supporting additional coverage types, as defined by the Geography Mark-up Language (GML) specification, and the ability to serve multidimensional data retrieved from an OLAP server. The distinctive feature of this solution being the proficiency to explore the two domains based on a single querying mechanism. The results of the augmented solution, investigated based on scenarios conceptualized by using the Deutsche Presse Agentur (dpa) dataset, have been positive and offer a solid base for further research work in this multidisciplinary domain.

Keywords: BI; DPA; GIS; GIS Web Services; GML; OGC; OLAP; SOLAP; WCS

¹ Is an OGC standard web service exchanging geospatial data (coverage)

² Is a non-profit, international, voluntary consensus standards organization that is leading the development of standards for geospatial and location based services. It defines a palette of open geospatial web interfaces.

ZUSAMMENFASSUNG

Historisch sind Business Intelligence (BI) und Geografische Informationssysteme (GIS) getrennten Entwicklungs- und Implementierungspfaden gefolgt. Kundenanfragen nach einem kompletten betrieblichen Bild, und der Bedarf nach mehr pro-aktivität hat zu einer synergistischen Macht geführt, die mit der Integration von BI und GIS ausgenutzt werden kann. Eine integrierte Geographic Business Intelligence Solution (GBIS), ein von (ESRI, 2005) geprägter Begriff, ermöglicht Benutzern, räumliche Information zu visualisieren und verwalten. Entscheidungsträger können mit standortbezogener Intelligenz auf verschiedenen Ebenen bewerten, planen und Dienste leisten, Information präsentieren und ad hoc Geschäftsfragen beantworten. Eine integrierte Lösung verbessert die Entscheidungsfindung und Ansprechbarkeit, indem sie die Eignung von GIS auf eine breitere Reihe von Geschäftslösungen ausdehnt. Die Suche nach einer GBIS-Lösung führte zur Einführung einer neuen Unterkategorie von raumbezogene Entscheidungsfähige Lösungen: Spatial Online Analytical Processing (SOLAP) oder Spatial OLAP. Diese Studie trägt zur Entwicklung der SOLAP Thematik bei, indem sie eine interoperable, web-basierte, offene und erweiterbare prototypische Lösung mit den Analyse-Fähigkeiten der beiden Technologien präsentiert. Der Prototyp beruht auf dem Web Coverage Service (WCS) nach Definition des Open Geospatial Konsortium (OGC), kombiniert mit einem OLAP Server. Der Autor hat eine vorhandene WCS Implementierung erweitert, um noch nicht vorhandene Coverage-Typen und die Fähigkeit mehrdimensionale Daten von einem OLAP Server anzufordern und zu verarbeiten. Die Besonderheit dieser Lösung besteht darin, beide Aspekte auf der Grundlage einer einzigen Abfragemechanismus abzufragen. Szenarien, konzipiert für Datensätze der Deutsche Presse Agentur (dpa) bilden die Grundlage zur Evaluation der erweiterten Lösung. Die Ergebnisse sind positiv und bieten eine solide Basis für die weitere Forschungsarbeit in diesem mehrdisziplinaren Gebiet.

Keywords: BI; DPA; GIS; GIS Web Services; GML; OGC; OLAP; SOLAP; WCS

LIST OF ABBREVIATIONS

BI – Business Intelligence
CEN – European Committee for Standardization
DPA – Deutsche Presse Agentur
DSS – Decision Support System
GBIS – Geographic Business Intelligence Solution
GIS – Geographic Information System
GML – Geographic Markup Language
ISO – International Organization for Standardization
IT – Information Technology
KVP – Key-Value-Pair
MDB – Multidimensional database
OGC – Open Geospatial Consortium
OLAP – OnLine Analytical Processing
OLTP – OnLine Transactional Processing
OWS – OGC Web Services
RDBMS – Relational database Management System
SDI – Spatial Data Infrastructure
SOLAP – Spatial OLAP
WFS – Web Feature Service
WMS – Web Map Service
WCS – Web Coverage Service
XLink – XML Linking Language
XML – Extended Markup Language
XMLA – XML for Analysis
W3C – World Wide Web Consortium
WS-I – Web Services Interoperability

INDEX

ACKNOWLEDGEMENT	II
ABSTRACT	III
ZUSAMMENFASSUNG	IV
LIST OF ABBREVIATIONS	V
1. INTRODUCTION	1
1.1 MOTIVATION	3
1.2 RESEARCH PROBLEM	7
1.3 RESEARCH APPROACH	7
2. APPLICATION SCENARIOS	10
2.1 SCENARIO 1 – <i>THE BUSINESS MAN</i>	10
2.2 SCENARIO 2 – <i>KNOWLEDGE IS THE KEY</i>	11
2.3 SCENARIO 3 – <i>DEUTSCHE PRESSE AGENTUR</i>	12
2.4 SUMMARY	13
3. GENERAL CONCEPTS	15
3.1 WEB SERVICES – AN INTRODUCTION	15
3.1.1 EXTENSIBLE MARKUP LANGUAGE	17
3.1.2 GEOGRAPHY MARKUP LANGUAGE	18
3.1.3 SIMPLE OBJECT ACCESS PROTOCOL	20
3.1.4 UNIVERSAL DESCRIPTION, DISCOVERY AND INTEGRATION	21
3.1.5 WEB SERVICES DESCRIPTION LANGUAGE	22

3.2	ONLINE ANALYTICAL PROCESSING – AN OVERVIEW	23
3.2.1	OLAP CONCEPTS	25
3.2.2	OLAP OPERATIONS AND ARCHITECTURE	31
3.2.3	OLAP QUERYING CAPABILITIES	35
3.3	OGC WEB SERVICES – INTEROPERABLE GEOPROCESSING	43
3.3.1	WEB MAP SERVICE.....	46
3.3.2	WEB FEATURE SERVICE	47
3.3.3	WEB COVERAGE SERVICE.....	48
3.3.4	SUMMARY	57
3.4	RESEARCH APPROACH – REVISITED.....	60
4.	RELATED WORK	64
4.1	INTRODUCTION	64
4.2	SPATIAL OLAP SOLUTIONS – AN OVERVIEW.....	65
4.3	SUMMARY	68
5.	CASE STUDY: A STRATEGY FOR WEB COVERAGE SERVICE AND ONLINE ANALYTICAL PROCESSING INTEROPERABILITY	71
5.1	INTRODUCTION	71
5.2	PLATFORM AND SOFTWARE REQUIREMENTS	72
5.2.1	GIS DOMAIN	72
5.2.2	OLAP DOMAIN.....	75
5.3	THE DATA BASIS	77
5.3.1	DEUTSCHE PRESSE AGENTUR DATASET	77
5.3.2	GERMAN CITIES AND STATE GEOGRAPHIC DATA.....	83
5.4	MEDIATION PROCESS – CONCEPTUAL MAPPING.....	88
5.4.1	MAPPING A COVERAGE TO A CUBE.....	88
5.4.2	SERVICE CONFIGURATION	91

5.4.3	MAPPING AND WRAPPING – GML TO XMLA.....	97
5.5	PROOF OF CONCEPT.....	111
5.6	SUMMARY.....	122
6.	CONCLUSION AND FUTURE WORK.....	125
6.1	CONCLUSION.....	125
6.2	RECOMMENDATIONS AND FUTURE WORK.....	126
7.	REFERENCES.....	129
8.	APPENDIX A – GLOSSARY.....	136
9.	APPENDIX B – SAMPLE DEEGREE CONFIGURATION FILES.....	143
10.	APPENDIX C – EXAMPLE REQUESTS, RESPONSES.....	153

INDEX OF FIGURES

Figure 1: BI and GIS foundation for a SOLAP. After: (Bédard, 2005/02/10).....	4
Figure 2: The three-tier architecture model of a web service. After: (BrainBell, 2006)	16
Figure 3: Web services architecture (roles and operations). After: (Champion, 2002a)	17
Figure 4: Sample XML document. After: (SysOnyx, 2002)	18
Figure 5: Simple example of a GML-encoded feature. After: (Brentjens, 2004)	19
Figure 6: HTML and GML System analogy. After: (Lake, 2006)	20
Figure 7: Sample SOAP request with header and body. After: (Twardoch, 2003).....	21
Figure 8: Schematic view of the WSDL document syntax. After: (Twardoch, 2003).....	22
Figure 9: Interaction of the web technologies used in web services. After: (Twardoch, 2003).....	23
Figure 10: Categories of Business Intelligence. After: (BI, 2005)	24
Figure 11: OLAP Cube structure example. After: (Databeacon, 2004)	26
Figure 12: Logical Multidimensional model. After: (Oracle, 2003).....	27
Figure 13: Dimension examples: Left - thematic dimension, Right - Spatial dimension. After: (Rodolphe, 2004)	28
Figure 14: OLAP Star schema. After: (OracleDBAExpert, 2003).....	29
Figure 15: Snowflake schema. After: (OracleDBAExpert, 2003)	29
Figure 16: Aggregation Hierarchies.	31
Figure 17: OLAP data exploration. After: (Pederson, et al., 2001).....	32
Figure 18: OLAP System architecture. After: (Baltzer, 2006)	33
Figure 19: OLAP storage options. After: (White, 2003)	36
Figure 20: OLAP Storage option for the prototypical solution	37
Figure 21: MDX syntax	39
Figure 22: XMLA concept and architecture. After: (Zhaohui, et al., 2005)	40
Figure 23: DISCOVER request syntax	40
Figure 24: XMLA sample EXECUTE request.....	42
Figure 25: The Open Web Service (OWS) Framework. After: (OpenGIS, 2003)	44
Figure 26: Summary - OGC data services. After: (Wojnarowska, et al., 2002)	45
Figure 27: Standard interoperable web mapping architecture. After: (Ding, et al., 2002)	46
Figure 28: WMS protocol diagram	47
Figure 29: WFS protocol diagram.....	48
Figure 30: Conceptual model of a coverage. After: (211/WG, 2004).....	50
Figure 31: Conceptual (Graphic) Coverage model. After: (Baumann, 2006)	50

Figure 32: Feature subtypes. Based on: (211/WG, 2004)	51
Figure 33: List (partial) of supported Coverage types. Based on: (211/WG, 2004).....	51
Figure 34: Coverage UML (partial) Schema. Based on: (211/WG, 2004)	52
Figure 35: Spatiotemporal Domain UML schema diagram. Based on: (211/WG, 2004)	53
Figure 36: RangeSet UML schema diagram	54
Figure 37: WCS protocol diagram.	55
Figure 38: Conceptual model for the integrated solution.....	62
Figure 40: Overview of the SOLAP solutions domain.....	65
Figure 41: GMLA WS prototype. After: (Silva, et al., 2006).....	67
Figure 42: IPTC Subject codes. Based on: IPTC/TEC 9701	79
Figure 43: Example of a report from the dpa dataset.....	80
Figure 44: Dimension Report Subject and levels.....	81
Figure 45: Dimension Report Location and levels.....	81
Figure 46: Dimension Report Time and levels	82
Figure 47: Measure(s) Priority and count.....	82
Figure 48: DPA OLAP cube model in star-schema.....	83
Figure 49: Multipoint coverage - XML schema. After: (211/WG, 2004).....	84
Figure 50: MultiSurface coverage - XML schema. After:(211/WG, 2004)	85
Figure 51: Germany cities dataset representing only those cities in germany those are available in the dpa report set.....	85
Figure 52: Germany States dataset representing all the german states	86
Figure 53: deegree Coverage class diagram	87
Figure 54: RangeSubset UML diagram. Based on: (Whiteside, et al., 2006).....	89
Figure 55: Cube UML structure diagram.....	90
Figure 56: Coverage to Cube Mapping	90
Figure 57: WCS service architecture and modules. After: (lat-lon, 2006)	91
Figure 58: deegree WCS configuration document.....	92
Figure 59: CoverageDescription OLAP extension.....	93
Figure 60: Coverage to Cube Mapping	94
Figure 61: deegree Datastore class diagram.....	95
Figure 62: deegree OLAP API package diagram.....	96
Figure 63: deegree request handling	98
Figure 64: DescribeCoverage request parameters. After: (Whiteside, et al., 2006).....	99
Figure 65: DescribeCoverage response UML class diagram. After: (Whiteside, et al., 2006)	100
Figure 66: Mapping example - DescribeCoverage to Discover.....	100

Figure 67: DescribeCoverage to Discover mapping	101
Figure 68: DescribeCoverage xml request example	101
Figure 69: DescribeCoverage response (partial) document	103
Figure 70: GetCoverage request parameters. After: (Whiteside, et al., 2006).....	104
Figure 71: Mapping GetCoverage to Execute.....	105
Figure 72: GetCoverage (partial) request.....	106
Figure 73: Mapped Execute request	108
Figure 74: GetCoverage (partial) response - domain set	109
Figure 75: GetCoverage (partial) response - range set.....	110
Figure 76: Scenario 1 - Proof of Concept.....	113
Figure 77: Scenario 1 - GetCoverage (domain set) response document.....	114
Figure 78: Scenario 1 - GetCoverage (partial range set) response document.....	115
Figure 79: Scenario 1 - GetCoverage (partial range set) response document.....	116
Figure 80: Scenario visualization using the CommonGIS tool	117
Figure 81: Scenario 2 - Proof of Concept.....	118
Figure 82: Scenario 2 - GetCoverage (partial range set) response document.....	119
Figure 83: Scenario 2 - GetCoverage (partial range set) response document.....	120
Figure 84: Scenario visualization using the CommonGIS tool	121

INDEX OF TABLES

Table 1: Summarized differences between OLTP and OLAP. After: (Navathe, 2004)	25
Table 2: Overview OLAP query interfaces	38
Table 3: XMLA DISCOVER schema rowsets	41
Table 4: XMLA supported properties	42
Table 5: Components of WCS operations.....	56
Table 6: OGC Web Services summary	58
Table 7: Research problem characteristics summary	61
Table 8: Related Works summary	69
Table 9: GIS Framework evaluation.....	74
Table 10: OLAP solution evaluation	76
Table 11: 'Mapping' WCS and XMLA request schemas.....	97
Table 12: Prototype solution evaluation	123

1. INTRODUCTION

Many organizations face the challenge of managing and presenting the sheer quantity of data being captured on a monthly, weekly, daily and hourly level. The introduction of business intelligence (BI) applications and technologies has helped organizations gather, provide access to, analyze, and present data and information easily to the decision makers. The applications utilize both relational and multidimensional technologies to form the overall BI infrastructure. From a historical perspective BI is a popularized umbrella term introduced by Howard Dresner of the Gartner Group in 1989 to describe a set of concepts and methods to improve business decision making by using fact-based support systems. BI is a broad category of applications and technologies for gathering, storing, analyzing, and providing access to data to help enterprise users make better business decisions. BI solutions include the activities of decision support systems, query and reporting, online analytical processing (OLAP), statistical analysis, forecasting and data mining. Microsoft defines BI as: *THE PROCESS OF EXTRACTING DATA FROM A DATABASE AND THEN ANALYZING THAT DATA FOR INFORMATION THAT YOU CAN USE TO MAKE INFORMED BUSINESS DECISIONS AND TAKE ACTION*. However, data is not always used to its full potential and part of its *richness*, the spatial component, is simply left out. It has been estimated that about 80% of the data stored in corporate databases integrates spatial information that can be characterized by position, shape, orientation or size (Frankin, April 1992). It is obvious that this meaningful data is worth being integrated in the decision making process to provide a complete operational picture.

To gain better advantage of the spatial dimension in decision making the appropriate tools must be used. Geographic Information Systems (GIS) are the obvious potential candidate for such a task. (Worboys, 1995) provide this typical definition of a conventional GIS: *A GIS IS A COMPUTER-BASED INFORMATION SYSTEM THAT ENABLES CAPTURE, MODELING, MANIPULATION, RETRIEVAL, AND PRESENTATION OF GEOGRAPHICALLY REFERENCED DATA*. GIS provides functionalities like

- 1) spatial data **acquisition** and **input**,
- 2) spatial data **storage** and **management**,
- 3) spatial data **presentation** and **output**, and
- 4) spatial data **manipulation** and **analysis**

Spatial analysis identifies the subset of techniques that are applicable when, as a minimum, data can be referenced on a two-dimensional frame and relate to terrestrial activities. The results of spatial analysis will change if location or extent of the frame changes, or if objects are

repositioned within it. Spatial analysis typically include, for example in a vector context, operations such as map overlay (combining of two or more map layers according to predefined rules), simple buffering (identifying regions on the map within a specified distance of one or more features, such as towns, roads or rivers) and similar basic operations. For raster-based GIS, widely used in the environmental sciences and remote sensing, this typically involves a range of actions applied to grid cells of one or more maps (or images) often involving filtering and/or algebraic operations (map algebra). Descriptive statistics, such as cell counts, mean value, variance, maxima, minima, cumulative values, frequencies and a number of other measures and distance computations are also often included in the generic term spatial analysis.

Since GIS was developed for the spatial domain it lacks the ready availability of analysis tools to help in decision-support beyond the domain. It is recognized that existing GISs *per se* are not adequate for decision-support applications when used alone and that alternative solutions must be used. (Bédard, 2002). Although a wide palette of analysis functionalities are available, this initial set should be enlarged to support a large variety of statistical techniques (descriptive, exploratory, and explanatory) that have been designed specifically for spatial and spatio-temporal data to take full advantage of the data.

BI tools on the other hand, though well-suited for knowledge discovery, are not adapted for the analysis of spatial data (Caron, 1998). In fact, business intelligence treats spatial data like any other data and spatial analysis is limited to predefined nominal locations (e.g. names of countries, states, regions, cities). Support for spatio-temporal analyses is limited (no spatial visualization, practically no spatial analysis, no map-based exploration of data, etc.) The union of spatial and non-spatial technologies, GIS and BI, is an interesting option to overcome the shortcomings of the two domains.

Historically, BI and GIS have followed separate development and implementation paths. Customer requests for a complete operational picture and the ability to be more proactive has led to the demand for a synergistic power that can be exploited by integrating business intelligence and geographic information systems. An integrated geographic business intelligence solution (GBIS), a term coined by (ESRI, 2005), enables users to both visualize and manage spatial information and empower decision makers, at different levels, with the location-based intelligence they need to assess, plan and deliver services, present information and deal with ad

hoc business queries. The integrated solution improves decision-making and responsiveness while extending the reach of GIS to address a wider range of business solutions.

This study contributes to the development of the geographic business intelligence research area by presenting an interoperable web-based open and extensible prototype solution with the analysis capabilities available in the two technologies.

1.1 MOTIVATION

Since BI and GIS are designed to accommodate and serve different purposes they are separate and distinct. The problem of integrating these two environments is multi-faceted. It includes consideration for technological as well as strategic issues. Traditionally BI and GIS applications are closed, isolated and incompatible with each other. Their integration to create GBIS solutions is a nightmare, due to poor documentation, obscure semantics of data, diversity of datasets, heterogeneity of existing systems in terms of data modeling concepts, data encoding techniques, storage structures, access functionality, etc (Bimonte, et al.).

Much of the research investigating the problem of integrating analytic and geographic processing has been carried out by the Information Technology (IT) community (Shekhar S., 2000). GBIS solutions allow an amalgamation of spatial solutions with the different categories of BI solutions. The three main categories being:

- 1) **information and knowledge discovery,**
- 2) **decision support and intelligent systems,** and
- 3) **visualization**

In this context we restrict ourselves to the information and knowledge discovery category of BI solutions. The concept of information and knowledge discovery is very broad and can take different forms. They are applications and subsystems that help people make decisions based on data that is culled from a wide range of sources. Information and knowledge discovery is an agglomeration of many parts (see Chapter 3) with OnLine Analytical Processing or OLAP being a prominent component. The wide acceptance of the new solution because of the advantages OLAP brings (see Chapter 3) has led to the concentration on OLAP solutions for decision support. OLAP has been first defined as: ... *THE NAME GIVEN TO THE DYNAMIC ENTERPRISE ANALYSIS REQUIRED: TO CREATE, MANIPULATE, ANIMATE AND SYNTHESIZE INFORMATION FROM EXEGETICAL, CONTEMPLATIVE AND FORMULAIC DATA ANALYSIS MODELS. THIS INCLUDES THE ABILITY TO DISCERN NEW*

OR UNANTICIPATED RELATIONSHIPS BETWEEN VARIABLES, THE ABILITY TO IDENTIFY THE PARAMETERS NECESSARY TO HANDLE LARGE AMOUNTS OF DATA, TO CREATE AN UNLIMITED NUMBER OF DIMENSIONS AND EXPRESSIONS. (Codd, et al., 1993) OLAP solutions were introduced to solve some limitations of the traditional transactional systems (i.e. OLTP – OnLine Transaction Processing – such as Relational Database Management System – RDBMS), to support aggregated information, rapid comparisons in space, time and other dimensions, trends and knowledge discovery, quick response to unforeseen queries and other complex operations needed during tactic and strategic decision-making processes.

The investigations for a GBIS solution led to the introduction of a new sub-category of spatial decision-making solutions: Spatial Online Analytical Processing (SOLAP) or Spatial OLAP (Rivest, 2001). SOLAP relies on the multidimensional paradigm and on an enriched interactive data exploration processing, thus filling the *analysis gap* between spatial data and geographic knowledge discovery. In spite of its short history, SOLAP already has reached a first level of maturity with its own concepts, technologies and applications. The multidimensional paradigm makes SOLAP an interesting option to be studied in detail for the scope of this study. SOLAP can be defined as (Bédard, et al., October, 2004): *A VISUAL PLATFORM BUILT ESPECIALLY TO SUPPORT RAPID AND EASY SPATIO-TEMPORAL ANALYSIS AND EXPLORATION OF DATA FOLLOWING A MULTIDIMENSIONAL APPROACH COMPRISED OF AGGREGATION LEVELS AVAILABLE IN CARTOGRAPHIC DISPLAYS AS WELL AS IN TABULAR AND DIAGRAM DISPLAYS*. These solutions add a spatial component to the traditional OLAP tool.

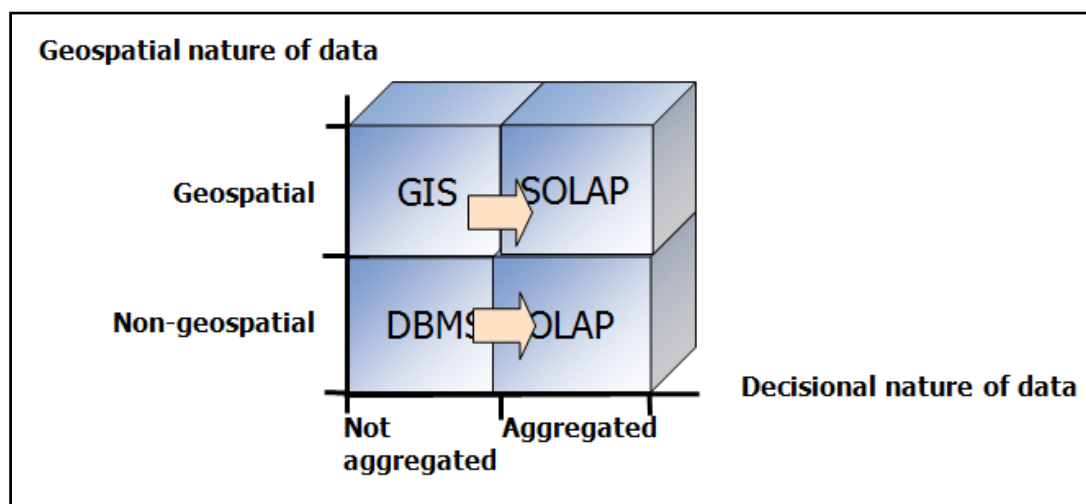


FIGURE 1: BI AND GIS FOUNDATION FOR A SOLAP. AFTER: (BÉDARD, 2005/02/10)

Figure 1 illustrates the two important components, geospatial and non-geospatial, of a SOLAP solution. The geospatial and the non-geospatial components can be divided into the sub-divisions – aggregated and not-aggregated. The SOLAP solution can be illustrated as a solution supporting aggregated geospatial data in a decision making process.

The solutions are based on coupling OLAP functionalities, used to provide multidimensional support, and GIS functionalities, used to store and visualize spatial information (Kouba Z., 2000), (Tchounikine A., 2005). Depending on the functionalities that are prioritized, the solution is termed as (LGS Group, 2000):

- 1) **GIS-centric** – the dominant tool – GIS offers its full functionality, but gets minimum capabilities from the OLAP tool;
- 2) **OLAP-centric** – the dominant tool – OLAP offers its complete functionality, and GIS offers minimum capabilities;
- 3) **Hybrid** – tightly coupled functionality, both the GIS and OLAP domain functionalities are equally represented.

These solutions, some OLAP-dominant and others GIS-dominant, offer a more or less elaborated subset of the desirable functionalities. Although much research has been done on this topic reflected by the continued success and maturing of the field, much needs to be done across many different areas of SOLAP solutions. In particular, the following challenges have been recognized by (Bimonte, et al.) that need to be addressed:

1. The stringent definition of a SOLAP solution supporting spatial data in a multidimensional model is also known as a tightly coupled hybrid solution. The introduction of spatial data in a **multidimensional model raises major problems from the implementation and theoretical point of view**. SOLAP implies a real rethinking of OLAP concepts, for example, storing and modeling the spatial dimension, and extending the spatial algebra. (Bédard, et al., 2001) offers a slightly tempered version of the definition for the non-expert, where SOLAP is defined as: *A NEW TYPE OF USER INTERFACE FOR MULTI-SCALE GIS APPLICATIONS AND WEB MAPPING*. This definition makes it possible to define a loosely coupled hybrid SOLAP solution where the GIS is used as a visual tool for OLAP operations. Loosely coupled hybrid solutions are the most widely available solutions in the market today.
2. **The solutions available are proprietary and inflexible in nature**, catering to specific needs and rarely providing an extensible interface resulting in high development cost. The

proprietary nature of an application is a deterrent to the interoperability³ between applications. There is a need for standardized and open solutions to move away from the current status quo in which the solutions are tightly coupled to their internal data models and structures.

3. **The solutions do not take full advantage of global communications network, such as the Internet**, to perpetuate the broader and free exchange of information and knowledge discovery results. This is due to the tight coupling of the data structure and application logic of the solution itself (a legacy of proprietary solutions).
4. **A GIS-centric SOLAP solution exposes the lack of powerful analytic capabilities to deal with problems.** [(Burrough, 1990);(Jannsen, et al., 1990); (Carver, 1991)] state the following ones:
 - In most GIS solutions spatial analytical functionalities lie mainly in the ability to perform deterministic overlay and buffer operations which are of limited use when multiple and conflicting criteria are concerned.
 - Current GIS solutions do not permit the assessment and comparison of different scenarios. They identify only solutions satisfying all criteria simultaneously.
 - Analytic functionalities found in most GIS solutions are oriented towards management and visualization of data but not effective analysis of them.
5. Closely related to the previous point, **an OLAP-centric solution offers limited GIS functionality to view the spatial distribution and correlations of phenomena**, and limited or no spatial operators to navigate through aggregated spatial data, the analysis would be counter-productive and incomplete even leading to false conclusions in some cases.
6. **The querying capabilities of both GIS and OLAP domains are not adequate to explore spatial multidimensional data.** SOLAP solutions require new spatial multidimensional exploration query languages adding to their complexity.

The conceptual idea of this research is to offer an adequate integrated platform prototype with the endeavor to overcome the aforementioned challenges availing the solutions offering SOLAP analysis capabilities.

³ The capability to communicate, execute programs, or transfer data among various functional units in a manner that requires the user to have little or no knowledge of the unique characteristics of those units.

1.2 RESEARCH PROBLEM

The research motivation forms the backbone of the research problem addressed. To summarize, the specific challenges identified above and encompassed in the research problem are:

- A. The current available solutions are predominantly proprietary in nature: closed, monolithic applications impeding interoperability and not easily extensible.
- B. Desktop-based solutions prevent easy and flexible transfer of data between organizations across various disciplines and industries. Also, the internal data format is a restricting factor for information exchange.
- C. OLAP- and GIS-centric solutions offer only a subset of the possible functionalities and do not fully harness the analytic powers that are there at their disposal.
- D. The introduction of a new spatial multidimensional query language only adds to the complexity of the solution and is a deterrent to a rapid, collaborative development of a solution.
- E. The introduction of spatial data in a multidimensional data model raises conceptual problems (Bimonte, et al.). These issues are established in Chapter 3.

Thus, the research scope of this thesis can be considered as: *AN INTEROPERABLE WEB-BASED HYBRID SOLAP SOLUTION SUPPORTING SPATIAL MULTIDIMENSIONAL INFORMATION EXPLORATION BY ABSTRACTING THE COMPLEXITY OF SEPARATELY QUERYING THE DATA* as a possible solution to overcome the aforementioned challenges.

1.3 RESEARCH APPROACH

The research approach has been divided into three sections. The first two sections deal with the theoretical aspect whereas the third section is based on a use-case scenario to create a prototype to test the integration between the two technologies, OLAP and GIS.

First, based on application scenarios, the author has formulated the requirements an integrated solution must satisfy. The scenarios outline the use and importance of an integrated business and geographic domain application.

Second, the clear trend to a broader use of location information throughout the organization has led to a perpetuation of SOLAP solutions. The author critically investigates the current commercial, academic and open solutions available and highlights the niche the current research aims to occupy.

The **third** part of the thesis concentrates on the development of a SOLAP prototype. The research focuses on finding a solution to overcome the above stated challenges. The author proposes the following approach to handle the abovementioned challenges:

- A. Introducing an interoperable solution compliant with (geospatial) standards and specifications.
- B. A web service solution complying with web interfacing standards and general software engineering requirements offering a scalable and extensible interface.
- C. A tight hybrid integration of a GIS and an OLAP tool. The integration must comply with the industry standards and target specific requirements.
- D. The solution must offer a single querying interface to abstract the querying complexity associated with exploring two different data formats with different interfaces.
- E. Pending the resolving of the issues involved the integration of spatial data in a multidimensional data model must be avoided. (Miquel, et al., 2002) highlights the problems facing such an undertaking.

The aforementioned salient features are a direct reference to the challenges discussed in Section 1.2.

The prototypical solution is an integration based on the Web Coverage Service (WCS⁴), as defined by the Open Geospatial Consortium (OGC⁵), and an OLAP server⁶. The use of OGC web services as the communication interface offers an open and interoperable platform independent implementation for spatial data exploration.

As far as known, at this point of time no OLAP integrated WCS has been realized. Therefore the developed application in the case study can be used to get some first practical experience with the integration application domain and to study the strengths and weaknesses of such a solution.

⁴ Is an OGC standard web service exchanging geospatial data (coverage)

⁵ Is a non-profit, international, voluntary consensus standards organization that is leading the development of standards for geospatial and location based services. It defines a palette of open geospatial web interfaces.

⁶ MS SQL Server

DESCRIPTION OF THE REMAINING CHAPTERS

Our starting point is in **application scenarios**, introducing the scenarios where the integrated solution is a much needed tool (Chapter 2)

Chapter 3, **general concepts** starts with introducing the web service concepts, continues on to describe in detail the important concepts of online analytical processing before introducing the OpenGIS web services with special emphasis on the Web Coverage Service.

In Chapter 4, **related works** are introduced, including a short summarization of the strengths and weaknesses of the individual solutions.

A **case-study** based on the Deutsche Presse Agentur dataset for the period between 2000 and 2005 is presented to outline and discuss the architecture and methodology of the integration process. (Chapter 5)

Conclusions and future work summarizes the study and presents an outlook to the direction in which the technology would be heading (Chapter 6).

2. APPLICATION SCENARIOS

A ubiquitous scenario in many businesses and organizations today is to rely upon reporting from business intelligence systems for summarized historical data to manage critical operations. For example, distance or drive time to the nearest outlet is a key consideration when planning whom to target for retail networks, which makes geo-mapping indicators an essential ingredient in customer and prospect databases in these sectors. Questions like: 1) Where should the next store be located; 2) Where to dispatch emergency crews and which is the fastest route, as well as questions predictive in nature: 1) What if a hurricane occurs in this location; 2) What if a competitor builds a store here, can be answered with the help of a GBIS solution. The integration of GIS and BI systems has many advantages and challenges, and is increasingly finding application in the real world.

The following fictitious, but realistic scenario summaries illustrate this. The scenarios are solution-neutral. The first scenario portrays a use-case for the retail market. An investment sector use-case characterizes the advantages and disadvantages in the second scenario. The third scenario handles the Deutsche Presse Agentur (dpa) news stories. The dpa scenario will be dealt with in greater detail in Chapter 5.

2.1 SCENARIO 1 – *THE BUSINESS MAN*

BACKGROUND OF THE SCENARIO

With the advent of globalization and the multinational setup of the large business organizations, it is of interest to make available the knowledge gained to a larger audience. In today's volatile economy, effective and successful business models are those that are capable of eliminating time and geographic barriers to reach international markets from ones' desktop. An understanding of global expectations, regionalism, nationalism, laws, work hours, and language differences are crucial in order to penetrate and compete in global markets, product brands, and operations. While accepting the challenge of the 21st century it is required to establish an appropriate and cost effective technological framework that is capable of integrating and managing such business intelligence.

SCENARIO SUMMARY

In a hypothetical multinational business, a manager is searching for the highest grossing store in a market area for a specific product. Having found the store, the manager then queries to find: What are the unique geo-demographic characteristics of the units' consumers who have recently purchased the product? This reveals a detailed geo-demographic breakdown of customers by such characteristics as ethnicity, home ownership, and the presence of children as well as students. After the identification is made, the manager may want to profile these purchases and compare their purchase profile to the total customer populations residing within the store's own primary marketing area. Using this information the manager may decide to replicate or adjust merchandising tactics employed at the store level. Aggregated with other store and customer data, customer product profiles may be created to craft market wide advertising and promotion plans. Broad population characteristics show up on a map and provide the marketers of the business with a clear focus for campaigns. The coupling of socio-economic statistics with the GIS environment offers the manager added decision making prowess. Site location analysis, the ability to deploy new marketing strategies and understanding the potential of local and international customer base are some of the advantages the manager gains from the integrated environment.

2.2 SCENARIO 2 – *KNOWLEDGE IS THE KEY*

BACKGROUND OF THE SCENARIO

The competitive edge goes to the investor who has the largest pool of accurate information from which to make the decisions and knows how to apply it. Investors need to monitor every detail of each and every aspect of the field and all related fields which they have invested in or actively seek to invest in. The investment process requires the inputs from many specialists, such as those in the finance, development and environmental areas, as part of the methodical and prudent decision-making process, in addition to the normal due diligence and financial projections.

SCENARIO SUMMARY

A private investor is on the lookout for an opportunity to invest his hard earned money. After considerable deliberations he decides to invest in land. Realizing the intricacies involved the investor decides to approach an investment banking firm. The firm has a proven track record and a high turnover ratio in land investments. The firms' experience taught it that an extended feasibility study required prior to the acquisition of land would entail the five major areas of:

1. Demographic analysis; Future population size, employment patterns, age distribution, wealth indices, economic vitality, cultural, educational and political environment;
2. Growth constraints – What are the political and public pressures?
3. Market diversification – How diverse is the local economic base?
4. Sector growth – Which sectors are growing and how will they affect demand?
5. Non-economic factors – What is the desirability of the proposed project?

Land has value based on when, where and for what purpose it is used, and an assessment and determination of those issues is a very important goal of the investment process, which attempts to remove uncertainty and lower the risk. Keeping this in mind, a comprehensive land research library of aerial photographs, maps and files, powered by a GIS and a BI tool, was created allowing the firm to create an information base to fuel the acquisition process. The coupling of the GIS and BI was a logical step taken by the firm keeping in mind the inter-dependencies of the data. The efficiency, speed and ease of retrieval achieved by this union ratified the decision.

2.3 SCENARIO 3 – *DEUTSCHE PRESSE AGENTUR*

BACKGROUND OF THE SCENARIO

The Deutsche Presse Agentur is an independent news agency offering international coverage of current affairs ranging from politics to economics, scientific and cultural to general events. Dpa covers 250 – 300 news reports, features, interviews and analyses per day, covering all classical news categories. Ignoring the location component in a news data source would be a loss considering the sheer quantity of data, captured, at monthly, weekly, daily and hourly level, available. Another important characteristic of the news dataset is its categorization of data. The data can be categorized according to the news subject or genre. For example a news item can be categorized under politics and/or under economy. Also, multiple sub-categorizations allow for a detailed breakdown of the news reports. Many online news reports have already taken the advantage to link the locations of their stories to Google Maps to further enhance their readability.

SCENARIO SUMMARY

On Monday the 31st of July 2005, Jennifer Apfelschorle – 8 years old at the time was kidnapped from the yard near her home in Bonn, Germany. The ensuing media frenzy and outpouring of concern makes this an interesting case for trend and pattern watching. The editor of a local

newspaper wanting to run the story in his newspaper wants to underline his research on criminal occurrences in Germany with statistical information. He visually explores the data along the geographical dimension – Bonn, the time dimension – Year 2005, and the subject dimension – criminal activities. The resulting visualization on his desktop encourages an intuitive and productive exploration of such a large geospatial data resource. A generalized exploration of all the criminal activities in Germany for the time period 2003 to 2005, or a more specialized, where the editor wants a detailed look at the number criminal cases in Bonn and Köln for the month of May, may result in potentially interesting patterns that needs to be studied and may lead to consolidated findings. The use-case discussed in Chapter 5 deals with the DPA data set for the period between 2000 and 2005 in greater detail.

2.4 SUMMARY

The scenarios identify the fundamental and desirable characteristics of a SOLAP solution. The solution is to be designed to improve accessibility to statistics and other existing information, and to support the discovery of new knowledge. The inherent advantages the solution should deliver are listed below:

A) UBIQUITOUS INFORMATION ACCESS

The same functionality should be available everywhere. Access to information is typically via the World Wide Web. For knowledge gained to be of greater use it is imperative that it is made available to a wider audience.

B) INTEROPERABLE EXCHANGE OF INFORMATION

Integrated interoperable application domain allows technologies to co-exist, communicating seamlessly with each other. Applications are not developed again, but reused by interoperating with each other thereby saving cost and leading to a value increase for the user.

C) CUTBACK OF ANALYSIS TIME

The raw data is organized in such a way that relevant data is clustered together for easier and faster access. Integrated environments minimize the time that a decision maker has to wait for the information to be made available. Also, the nature of graphical information is such that it allows faster grasping capacity thus increasing the speed of trend and pattern analysis.

D) EFFECTIVE COMMUNICATION OF RESULTS

The ability of the map to condense a wealth of information into a visual image and the advantages that brings is readily apparent. Adding a map for analysis to BI brings a competitive advantage by not only allowing decisions to be based on raw data (e.g. spatial information geometries) and more kinds of multidimensional data but also by communicating the resulting information in a way that is readily understood. Seamless transfer of information will guarantee improvements in performance of the organization.

E) EFFECTIVE EXPLORATION OF DATA

The advantage of an OLAP solution allowing different views at different levels helps in the decision making process. OLAP systems support hypothesis-driven exploration of data cubes through operations such as drill-down, roll-up, and selection. Using these operations, an analyst navigates unaided through a huge search space looking at large number of values. Ad hoc querying capabilities allows a single query to explore the data sets bringing a higher level of consistency to the result set.

To conclude, the characteristics identified by the scenarios are in part reflection of the strategies mentioned in Section 1.3. The existing solutions (see Chapter 4) are to be evaluated on the basis of their competence in satisfying the pooled criteria from the aforementioned and from the salient criteria in Section 1.3. The pioneer prototype solution aspired would entail the design of an interoperable, vendor neutral, standards compliant web service with ad hoc multidimensional- and spatial data exploration capabilities. The following chapter introduces the basic technologies necessary to attain the goal.

3. GENERAL CONCEPTS

This chapter discusses the current and evolving standards for OGC web services and the important OLAP concept. The importance of this chapter is highlighted by the necessity to understand the intricacies of the workings of the application domains to fully comprehend the possibilities for interoperability. Before a description of these two important application domains is given, perspective is provided by introducing the general concepts of web services and the underlying technologies.

3.1 WEB SERVICES – AN INTRODUCTION

Web services are modular, self-contained, component applications that can be discovered and used by other applications and other web services. The Web Services Architecture Working Group, a part of the World Wide Web Consortium's Web Services Activity, defines a web service as follows: *A WEB SERVICE IS A SOFTWARE SYSTEM IDENTIFIED BY AN URI, WHOSE PUBLIC INTERFACES AND BINDINGS ARE DEFINED AND DESCRIBED USING XML. ITS DEFINITION CAN BE DISCOVERED BY OTHER SOFTWARE SYSTEMS. THESE SYSTEMS MAY THEN INTERACT WITH THE WEB SERVICE IN A MANNER PRESCRIBED BY ITS DEFINITION, USING XML BASED MESSAGES CONVEYED BY INTERNET PROTOCOLS.* Under this model for internet computing, various web applications can be stitched together: applications from different vendors, of various vintages, written in different computer languages, and running on disparate platforms can easily communicate and cooperate.

Open standards and the focus on communication and collaboration among applications have created an environment (e.g. internet) where web services are becoming the platform for application integration. Typically applications are constructed using multiple web services from various sources that work together regardless of where they reside or how they are implemented. One of the primary advantages of web services is that they allow programs written on different platforms to communicate with each other in a standards-based way.

Web services differ from the proprietary connectors and interfaces traditionally used to integrate disparate applications. The loosely coupled nature of web services is more forgiving than the interface requirements of traditionally integrated systems. They are self-describing and self-announcing, meaning that all specifications related to the use and behavior of a service are part of

the service itself. Additionally, web services allow an application to invoke a remote process or application as if it were part of the invoking application. Figure 2 illustrates the basic three tier architecture of a web service. The web service architecture usually consists of a client requesting data, a server managing and serving the data, and a database holding the data.

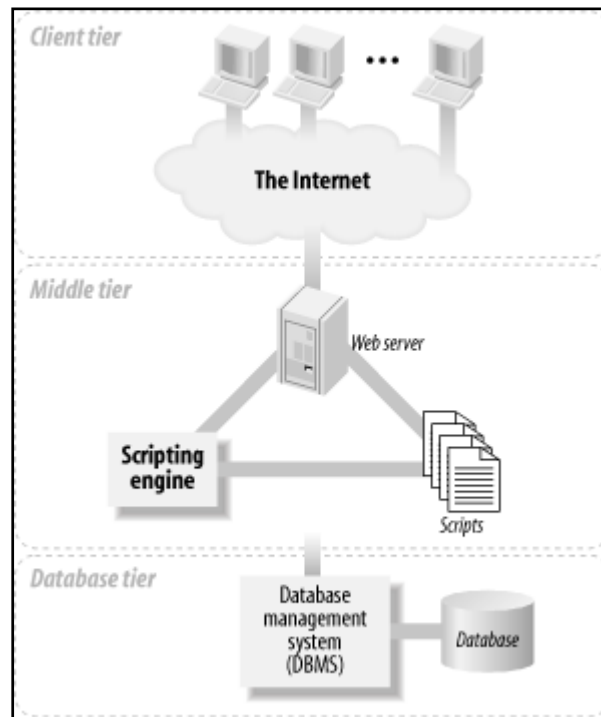


FIGURE 2: THE THREE-TIER ARCHITECTURE MODEL OF A WEB SERVICE. AFTER: (BRAINBELL, 2006)

(Wolter, 2001) explains that among many definitions of web services, several items are in common. First, web services expose their functionality to users through a standard web protocol, typically the Simple Object Access Protocol (SOAP). In addition, for users to be able to build different applications, web services provide a description of their interface, usually expressed in the Web Services Description Language (WSDL, pronounced *wizdel*). Finally, web services use the Universal Discovery Description and Integration (UDDI) mechanism for registration so that potential users can find them easily.

Figure 3 illustrates the web services architecture depicting the roles and operations involved. As shown here, software agents engaged in web service operations act in three so-called roles: as service providers, service requestors and service registries (discovery agencies). The Find/Bind/Publish process, a core concept of web services, is illustrated in the figure with the appropriate technologies used.

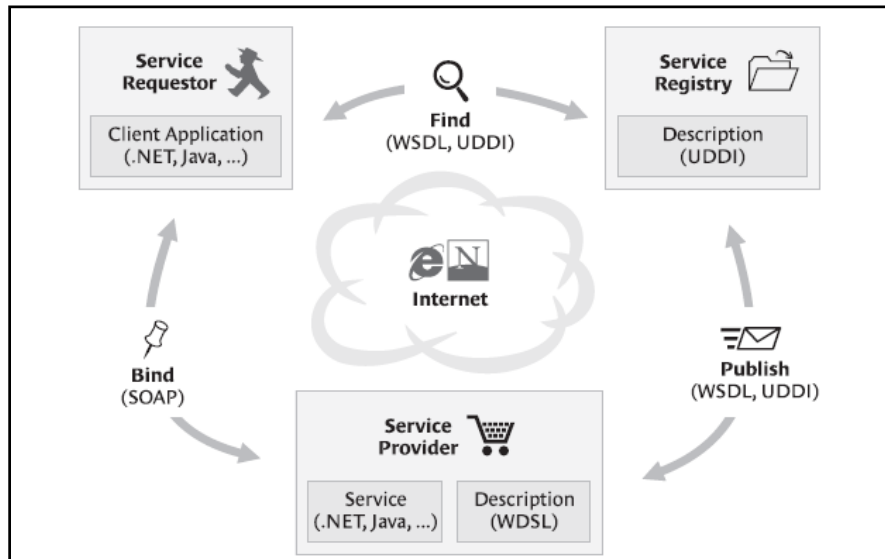


FIGURE 3: WEB SERVICES ARCHITECTURE (ROLES AND OPERATIONS). AFTER: (CHAMPION, 2002A)

After having outlined the basic web service architecture model, the following section introduces the underlying technologies: XML, SOAP, UDDI, and WSDL. XML is the foundation of all technologies used in web services, so that occasionally web services are referred to as “*XML Web Services*”. The following section introduces an overview of the XML technology followed by the abovementioned special-purpose technologies.

3.1.1 EXTENSIBLE MARKUP LANGUAGE

(Chappel, 2002) calls XML the: *universally accepted modern lingua franca for describing data*. (Walsh, 1998) gives a simpler phrasing: *XML is a markup language for structured documentation*. The XML specification defines XML as: *EXTENSIBLE MARKUP LANGUAGE, ABBREVIATED XML, DESCRIBES A CLASS OF DATA OBJECTS CALLED XML DOCUMENTS AND PARTIALLY DESCRIBES THE BEHAVIOR OF COMPUTER PROGRAMS WHICH PROCESS THEM. XML IS AN APPLICATION PROFILE OR RESTRICTED FORM OF SGML, THE STANDARD GENERALIZED MARKUP LANGUAGE*.

XML documents are electronic documents consisting of storage units called entities, which are composed of character data and so-called markup. The character data represents the content of the document, while the markup describes the layout and logical structure of the document and consists of declarations, elements (enclosed in opening and closing tags), components, character references, and processing instructions. As a form of semi-structured data XML can be used as a merge point for both structured and un-structured data. This offers enormous potential to the

enterprise that wants to aggregate two forms of data in one design. This straightforward language enables different systems to *talk* to each other. Being a widely accepted industry standard protocol allows for easy handling and exchanging information thereby supporting system integration.

```
<Addresses>
  <Address>
    <StreetAddress>512 S. State St</StreetAddress>
    <City>Provo</City>
    <State>UT</State>
    <Zip>84601</Zip>
  </Address>
  <Address>
    <StreetAddress>72 S. Main St.</StreetAddress>
    <StreetAddress>Apt #2</StreetAddress>
    <City>Jordansville</City>
    <State>MW</State>
    <Zip>02300</Zip>
  </Address>
  <Address>
    <StreetAddress>1321 E. Lester</StreetAddress>
    <City>Tucson</City>
```

FIGURE 4: SAMPLE XML DOCUMENT. AFTER: (SysONYX, 2002)

3.1.2 GEOGRAPHY MARKUP LANGUAGE

GML is an XML encoding for the modeling, transport and storage of geographic information including both the spatial and non-spatial properties of geographic features. The GML specification (Cox, et al., 2003) defines the XML schema syntax, mechanisms, and conventions that:

- Provide an open, vendor-neutral framework for the definition of geospatial application schemas and objects;
- Allow profiles that support proper subsets of GML framework descriptive capabilities;
- Support the description of geospatial application schemas for specialized domains and support the information communities;
- Enable the creation and maintenance of linked geographic application schemas and datasets;
- Support the storage and transport of application schemas and data sets;

- Increase the ability of organizations to share geographic application schemas and the information they describe.

GML is concerned with the representation of geographic data content independent of any particular visualization of that data. Like any XML encoding, GML represents geographic information in the form of text. Examples are topographic data, cadastral data, or road transport data. The abstract model of geography developed by the OGC is used as base to create GML. The abstract model describes the world in terms of geographic entities called *features*. Essentially a feature is a list of *properties* and *geometries*. Properties have the usual *name*, *type*, *value* description. Geometries are composed of building blocks such as *points*, *lines*, *curves*, *surfaces* and *polygons*. An essential component of a geographic system is the means of referencing the geographic features to the earth's surface or to some structure related to the earth's surface. The current version of GML incorporates an earth based spatial reference system which is extensible and which incorporates the main projection and geo-centric reference frames in use today. Figure 5 illustrates a real world object, a road, as a GML feature. The geometry of this feature is a simple line consisting of coordinates which are referenced to a distinct spatial reference system.

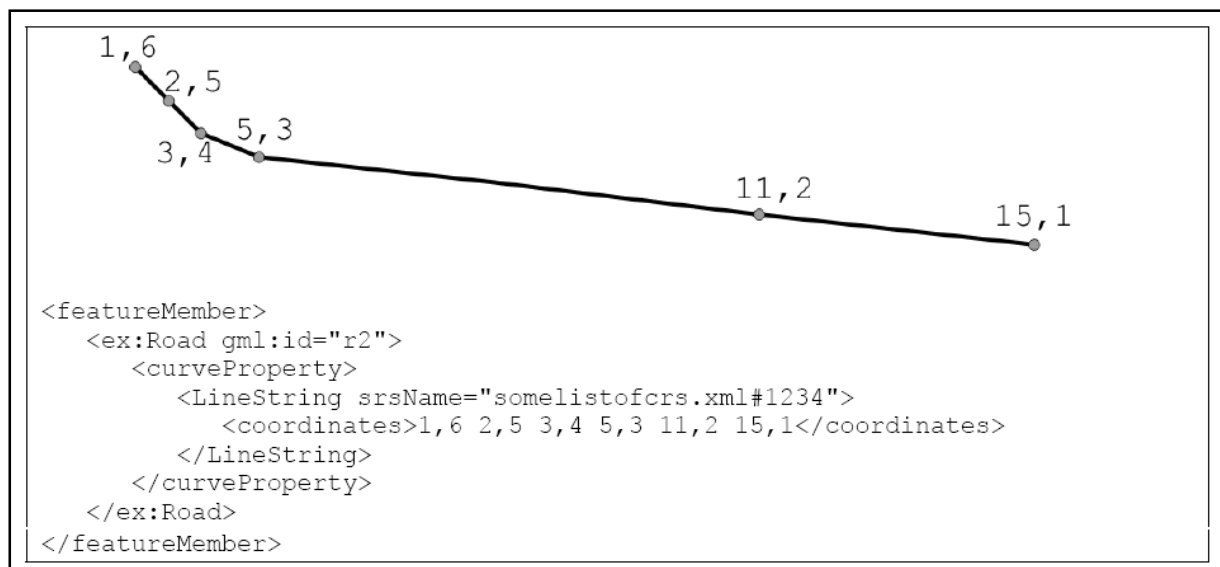


FIGURE 5: SIMPLE EXAMPLE OF A GML-ENCODED FEATURE. AFTER: (BRENTJENS, 2004)

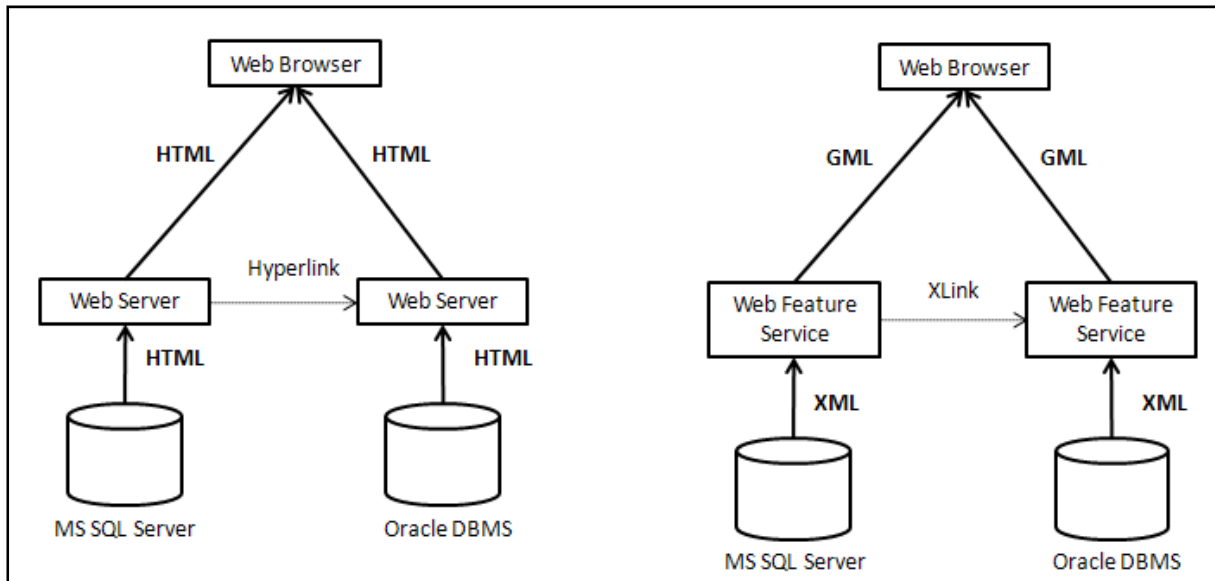


FIGURE 6: HTML AND GML SYSTEM ANALOGY. AFTER: (LAKE, 2006)

Figure 6 exemplifies web service systems using HTML and GML. The basic difference between the two services is the communication language used; GML replaces HTML allowing the web browser access to geographic information stored in the database.

3.1.3 SIMPLE OBJECT ACCESS PROTOCOL

SOAP is a messaging standard used by internet applications to communicate. SOAP allows process control to be passed between two networked applications. It is not a language or a component; it is a communications protocol that defines encoding rules to allow disparate systems to interact. Although the SOAP standard was first introduced by Microsoft, it is now supported by all major vendors and allows web services to be developed in virtually any programming language. Since SOAP messages can be automatically generated and processed in software tools or libraries written in various programming languages and running on different operation systems, this approach allows communication between client applications and web services implemented in different languages and running on different platforms.

SOAP uses existing protocols such as HTTP⁷ or SMTP⁸ to provide a standard way of packaging SOAP documents, also called *messages*. A SOAP message is a XML document composed of exactly one root *Element* that must contain the body of the message and may additionally contain

⁷ HTTP – Hypertext Transfer Protocol - provides a standard for Web browsers and servers to communicate.

⁸ Simple Mail Transfer Protocol - is the de facto standard for e-mail transmissions across the internet.

any header information used to describe the message (meta-information). Figure 7 illustrates a SOAP message example. The tag `<soap:Envelope>` represents the root element consisting of two body elements `<soap:Header>` and `<soap:Body>`.

```
POST http://brinkster.com/twardoch/wsSquare/wsSquare.asmx HTTP/1.1
Host: eu.webmatrixhosting.net
Content-Type: text/xml; charset=utf-8 Content-Length: 299
SOAPAction: "http://www.twardoch.com/ws-namespace/Square"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <Sender>
      John Smith
    </Sender>
  </soap:Header>
  <soap:Body>
    <Square xmlns="http://www.twardoch.com/ws-namespace">
      <x>25</x>
    </Square>
  </soap:Body>
</soap:Envelope>
```

FIGURE 7: SAMPLE SOAP REQUEST WITH HEADER AND BODY. AFTER: (TWARDUCH, 2003)

3.1.4 UNIVERSAL DESCRIPTION, DISCOVERY AND INTEGRATION

UDDI is the *yellow pages* of web services – a directory used to allow an application to announce its availability and services to other applications. Web services registered in an UDDI directory may be discovered by web applications hunting for specific services, goods, or partners. As (Curbera, 2002) points out: the core of UDDI is the UDDI Business Registry (UBR). It is a central registry operated by several companies as a set of self-replicating databases that allows users to query the information they need and give the service providers the ability to update the stored data. The UDDI specification defines the structure and operation of a registry. In particular, the scope and encoding of the information about each service is defined, as well as an Application Programming Interface (API) that allows accessing (Inquiry API) and modifying (Publishing API) the registry. The UDDI specification defines SOAP over HTTP as the correct method for accessing the business registry. The registry stores information about web services using XML and acts as a web service itself.

3.1.5 WEB SERVICES DESCRIPTION LANGUAGE

WSDL is an XML format used to describe the capabilities and interoperability required of a web service. WSDL provides an abstract definition of the communication details between two applications. As the specification states, one of the aims of WSDL is information hiding: *WSDL ENABLES ONE TO SEPARATE THE DESCRIPTION OF THE ABSTRACT FUNCTIONALITY OFFERED BY A SERVICE FROM CONCRETE DETAILS OF A SERVICE DESCRIPTION SUCH AS 'HOW' AND 'WHERE' THAT FUNCTIONALITY IS OFFERED.* (Curbera, 2002) explains that a WSDL service description provides two parts of information: an abstract service interface description as well as some specific protocol-dependent details. Figure 8 gives a schematic view of the interdependencies within a WSDL document. The abstract description of the interface that can be used to exchange messages with the service consists of three main components: the vocabulary, the message, and the interaction. In Figure 8 the vocabulary is represented by the Types – *simpleType* and *complexType*, the message by *content*, and the interaction is an *operation* action.

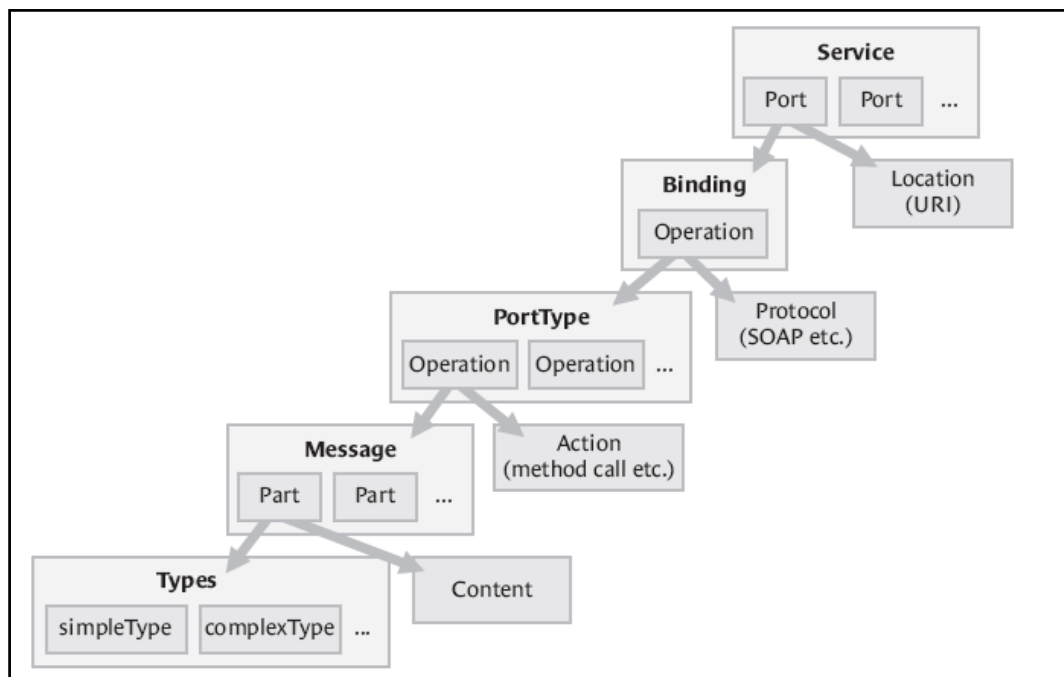


FIGURE 8: SCHEMATIC VIEW OF THE WSDL DOCUMENT SYNTAX. AFTER: (TWARDUCH, 2003)

In conclusion, Figure 9 illustrates how these technologies can be used as a web service. In this illustration, a tool/application is accessing an UDDI registry to discover the available services. The registry could be public or a private enterprise registry. The tool browses the registry by sending XML messages using a SOAP protocol. Within the registry is the description of the

available services, where an entry written in WSDL describes the services and interfaces. The web service is usually built from components, such as Enterprise Java Bean (EJB) components wrapped in XML/SOAP, or it could be a legacy application with an XML/SOAP wrapper. The legacy case will probably be the most prevalent web service in the near future.

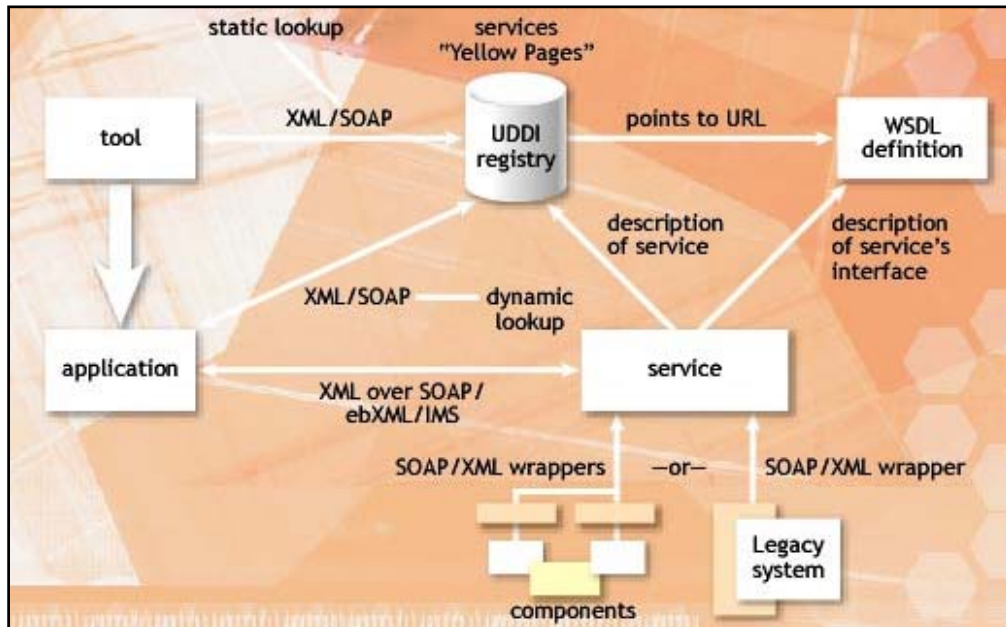


FIGURE 9: INTERACTION OF THE WEB TECHNOLOGIES USED IN WEB SERVICES. AFTER: (TWARDCH, 2003)

3.2 ONLINE ANALYTICAL PROCESSING – AN OVERVIEW

Business Intelligence is a broad category of applications and techniques for gathering, storing, analyzing, and providing access to data to help enterprise users make better business and strategy decisions. The need for multidimensional analysis support leads to the focus of this study on information and knowledge discovery with emphasis on the OLAP solution. Figure 10 gives an overview of the different categories of tools associated with business analytics.

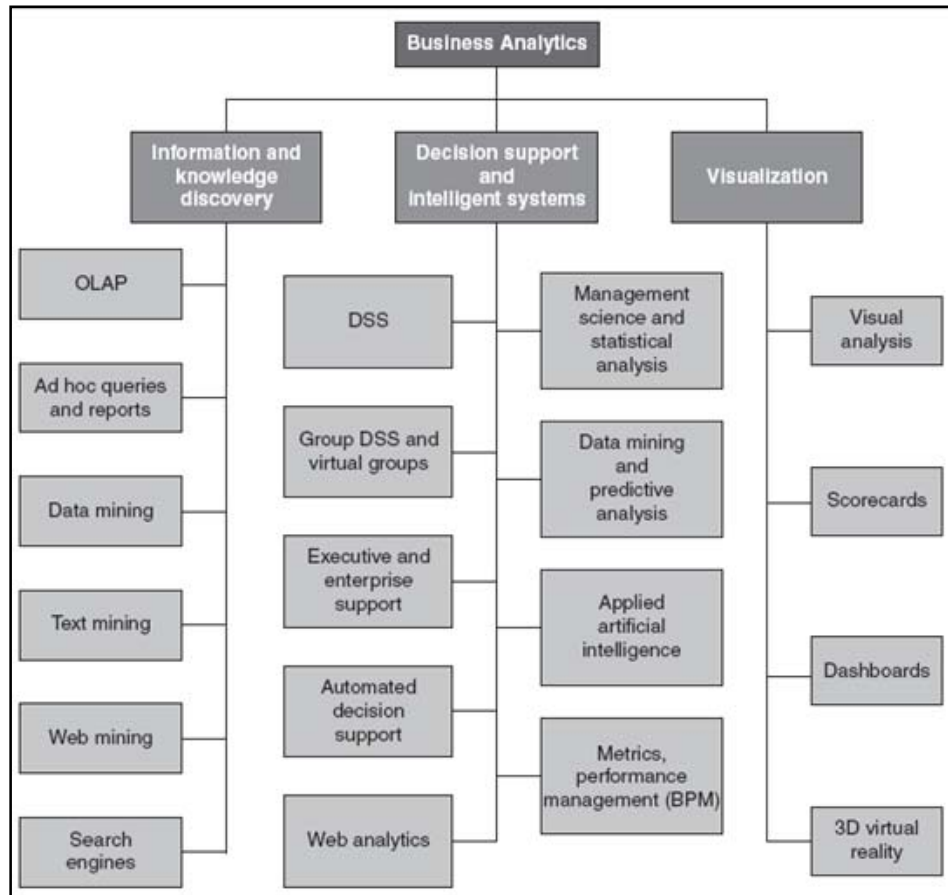


FIGURE 10: CATEGORIES OF BUSINESS INTELLIGENCE. AFTER: (BI, 2005)

Multidimensional database technology is a key factor in the interactive analysis of large amounts of data for decision making purposes. In contrast to previous technologies, these databases view data as multidimensional cubes that are particularly well suited for data analysis. The multidimensional data model is composed of logical cubes, measures, dimensions, hierarchies, levels and attributes. The multidimensional data model is an integral part of OLAP. OLAP tools use database technology to support analysis and mining of long data horizons, and to provide decision makers with a platform from which to generate decision making information. With OLAP analysis it is possible to see trends that are not evident with other types of reporting. Table 1 summarizes the differences between OLTP⁹ and OLAP to provide a quick overview to the advantages of an OLAP tool.

⁹ Online transaction processing - refers to a class of systems that facilitate and manage transaction-oriented applications, typically for data entry and retrieval transaction processing.

	OLTP	OLAP
USER	Clerk, IT Professional	Knowledge worker
FUNCTION	Day to day operations	Decision Support
DATA	Current, isolated	Historical, Consolidated
VIEW	Detailed, Flat relational	Summarized, Multidimensional
# RECORDS ACCESSED	Tens	Millions
# USERS	Thousands	Hundreds
DB SIZE	100 MB-GB	100 GB-TB

TABLE 1: SUMMARIZED DIFFERENCES BETWEEN OLTP AND OLAP. AFTER: (NAVATHE, 2004)

The historical summarized large dataset makes OLAP an exceptional decision support tool. The inability to update the cube on a daily basis, due to time constraints, and the smaller number of user access can be overlooked due to the nature of decision support.

This following section deals with the general OLAP concepts followed by the OLAP architecture and querying mechanism. The purpose of this section is to discuss the technology and thereby show how the technology plays an integral part in supporting spatio-temporal exploration of data.

3.2.1 OLAP CONCEPTS

The term OnLine Analytical Processing, or OLAP, was coined in the early 1990s by E.F. CODD, the pioneer of relational systems, in order to clearly indicate that something different was needed for analytical processes. OLAP has been defined for the first time as (...) *THE NAME GIVEN TO THE DYNAMIC ENTERPRISE ANALYSIS REQUIRED TO CREATE, MANIPULATE, ANIMATE AND SYNTHESIZE INFORMATION FROM EXEGETICAL, CONTEMPLATIVE AND FORMULAIC DATA ANALYSIS MODELS. THIS INCLUDES THE ABILITY TO DISCERN NEW OR UNANTICIPATED RELATIONSHIPS BETWEEN VARIABLES, THE ABILITY TO IDENTIFY THE PARAMETERS NECESSARY TO HANDLE LARGE AMOUNTS OF DATA, TO CREATE AN UNLIMITED NUMBER OF DIMENSIONS, AND TO SPECIFY CROSS-DIMENSIONAL CONDITIONS AND EXPRESSIONS* (Codd, et al., 1993). OLAP technology is based on the multidimensional database approach, which introduces concepts that differ from the concepts found in the transactional database approach. The key multidimensional concepts include: dimensions, members, measures, facts and

most importantly data cubes (Berson, et al., 1997) ; (AltaPlana, 2004); (Pendse, 2000); (Thomsen, 2002)).

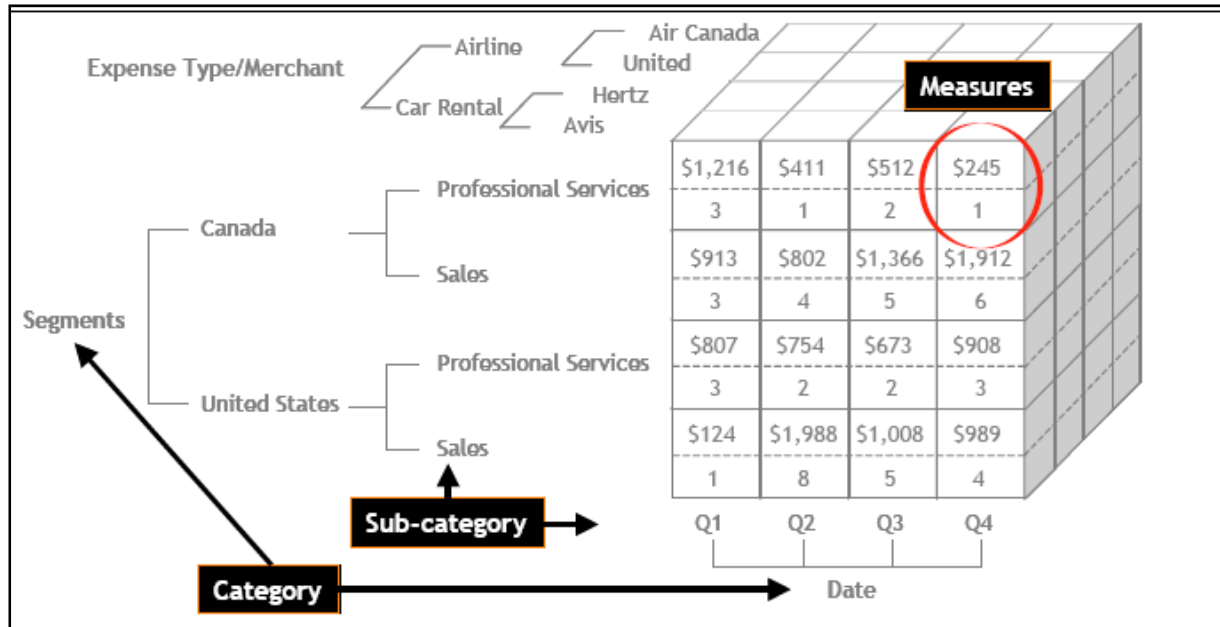


FIGURE 11: OLAP CUBE STRUCTURE EXAMPLE. AFTER: (DATABEACON; 2004)

Figure 11 illustrates a cube example with dimensions such as *Date*, *Segments* and *Expense Type/Merchant*. Each dimension being an aggregation of different hierarchies and levels.

DATA CUBE

A data cube (also called hypercube when more than 3 dimensions are used) is composed of a set of measures aggregated according to a set of dimensions (Thomsen, et al., 1999). Inside a data cube, possible aggregations of measures on the possible combinations of dimension members (the facts) can be pre-computed (to a certain level) and stored to increase query performance. Several data cubes can be built from the same sources of data for different interactive exploration and analysis needs, as they usually are read-only datasets. Figure 12 illustrates the logical multidimensional cube model - a cube consisting of measures and dimensions.

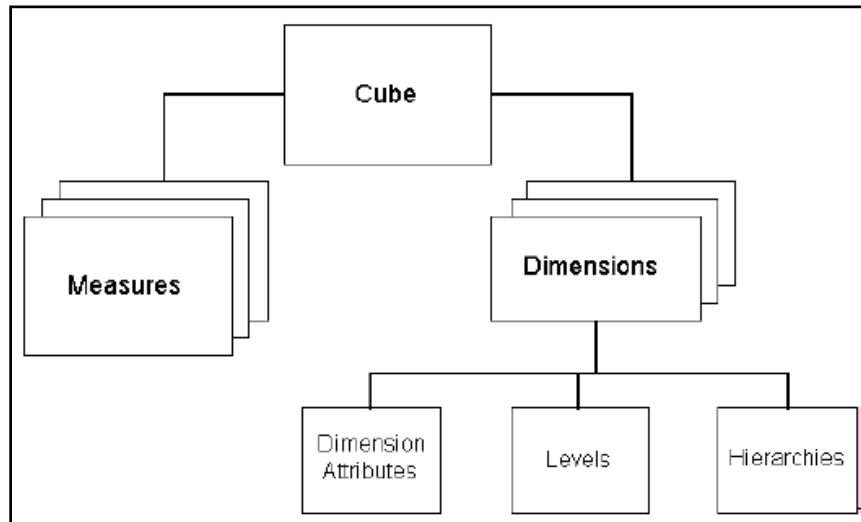


FIGURE 12: LOGICAL MULTIDIMENSIONAL MODEL. AFTER: (ORACLE, 2003)

DIMENSIONS

The dimensions represent the themes of interest for a user, or the analysis axes of an n -dimensional thematic space (ex. *news item subject*, *news item location*, *news item date/time* and *administrative zones* in a news report context). Within the multidimensional database context, the dimensions are seen as the independent variables included in the analysis. Dimensions are organized hierarchically into levels of granularity, or levels of details. Dimensions can be of different types: temporal, spatial (non-cartographic in the case of a conventional (non-spatial) OLAP tool) and descriptive (or thematic) (Bédard, et al., 1997). The concept of dimension in the OLAP context is different from the concept of dimension in a spatial reference system (i.e. longitude, latitude, height axes); nevertheless, dimensions can be spatial such as those using location names solely (ex. names of country-province-area-city). Figure 13 depicts dimension examples. The thematic dimension *race* can be identified by the individual race types, for example, Anglo-Saxon, Indian etc. Another possibility of a dimension is the spatial dimension. The figure also depicts breakdown of the dimension into the individual layers/hierarchies that make it up with the last level being the individual objects, for example, n1, n2, n3 for the level lake.

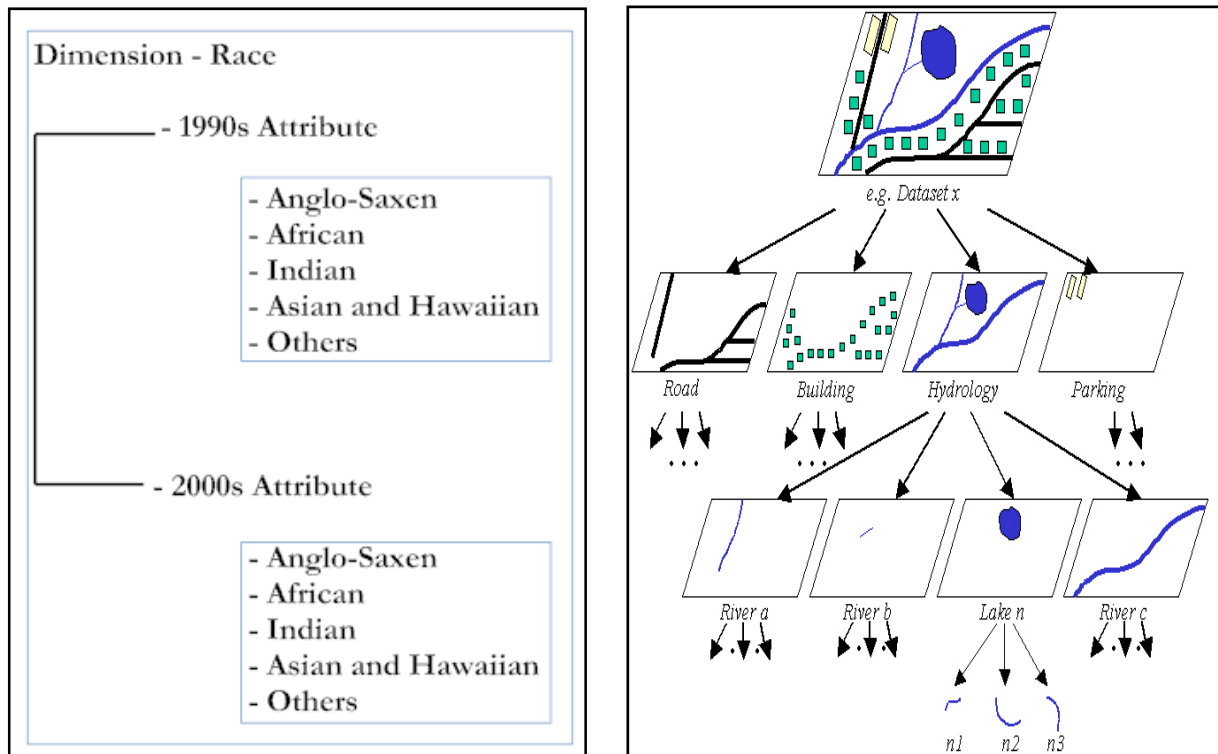


FIGURE 13: DIMENSION EXAMPLES: LEFT - THEMATIC DIMENSION, RIGHT - SPATIAL DIMENSION. AFTER: (RODOLPHE, 2004)

Like its OLTP counterpart the OLAP is also described by a logical design. A logical design is a conceptual, abstract design involving arranging data into a series of logical relationships called entities and attributes. An entity represents a chunk of information. In relational databases, an entity often maps to a table. An attribute is a component of an entity and helps define the uniqueness of the entity. In relational databases, an attribute maps to a column. While entity-relationship diagramming has traditionally been associated with highly normalized models such as OLTP applications, the technique is still useful in dimensional modeling. OLAP is usually modeled using either a star or a snowflake schema¹⁰. A star schema consists of one central fact table and several denormalized¹¹ dimensional tables

¹⁰ Logical design

¹¹ Denormalization is the process of attempting to optimize the performance of a database by adding redundant data

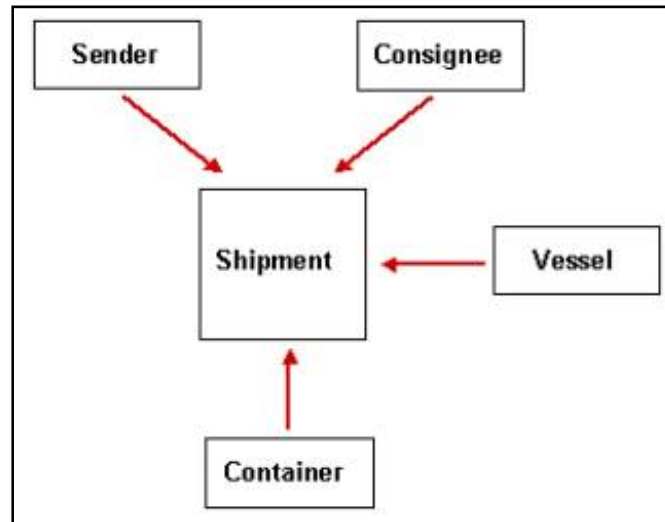


FIGURE 14: OLAP STAR SCHEMA. AFTER: (ORACLEDBAEXPERT, 2003)

A snowflake schema is a normalized star schema such that dimension tables are normalized. The schema illustrated in Figure 15 would represent a star schema when the *type* dimension would be included in the *container* dimension.

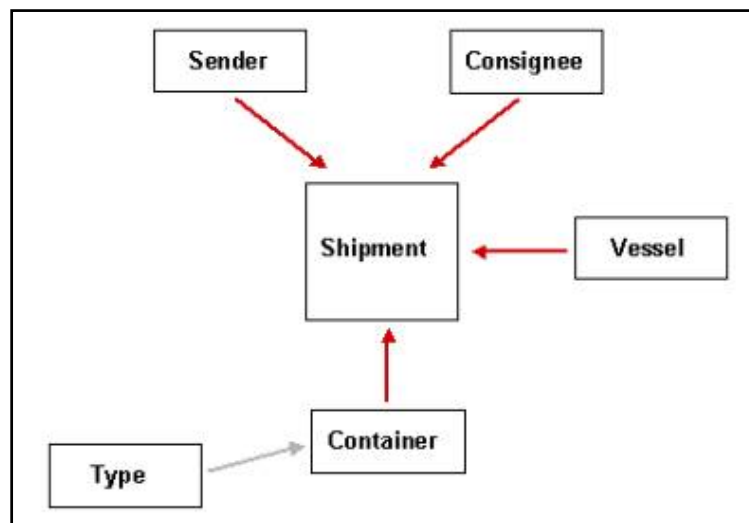


FIGURE 15: SNOWFLAKE SCHEMA. AFTER: (ORACLEDBAEXPERT, 2003)

MEASURES

The measures (for example: count, sum, standardized rate) are the numerical attributes analyzed against the different dimensions. A measure can then be considered as the dependent variable while dimension members are the independent variables. Each of the measures depends on a set of dimensions, which provide the context for the measure. The dimensions together are assumed

to uniquely determine the measure (Chaudhari, et al., 1997). Measures can be based on complex formulas and can contribute to the creation of sophisticated mathematical models for use in the development of scenarios. Each potential combination of dimension members, with the resulting measures' value for a particular aggregation level, represents a fact.

HIERARCHIES AND LEVELS

A **hierarchy** is a way to organize data at different levels of aggregation. In viewing data, analysts use dimension hierarchies to recognize trends at one level, drill down to lower levels to identify reasons for these trends, and roll up to higher levels to see what affect these trends have on a larger sector of the business. Hierarchies are usually presented in the logical level¹² using a star schema, as shown in Figure 14, or using a normalized structure called the snowflake schema (see Figure 15).

Each **level** represents a position in the hierarchy. Each level above the base (or most detailed) level contains aggregate values for the levels below it. The members at different levels have a one-to-many parent-child relation. The level aspect is very important because it determines the volumes of stored data and also the kind of queries the user can formulate. Hierarchies and levels have a many-to-many relationship. A hierarchy typically contains several levels, and a single level can be included in more than one hierarchy.

Data information can be organized at different levels of detail as illustrated in Figure 16. The three dimensions – product, time and location are aggregated based on their hierarchies and levels. The dimension *product* is identified by its hierarchy *product type* and the level *product*. The dimension *time* on the other hand is identified over two hierarchies. One hierarchy level is *month* and the second is *season*. As can be seen in the illustration a dimension can have multiple hierarchies of different lengths. The hierarchy *month* is further broken down into *day* which in turn is an aggregation of the levels – *hour*, *minute* and *second*, and/or *AM/PM*. Similar to the dimension *time* the dimension *location* is an aggregation of multiple hierarchies – *continent* and *country*. The hierarchy *country* is an aggregation of *region* and *state*. The two different hierarchies are based on the same level *city*.

¹² Logical level or logical layer of an information system provides high level functions such as file management, file storage and other basic important functionalities.

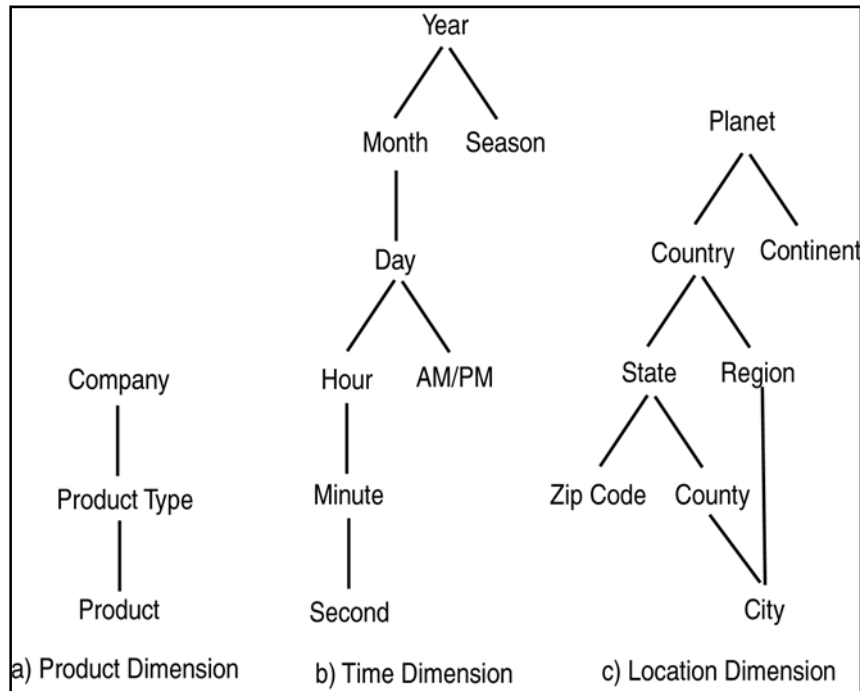


FIGURE 16: AGGREGATION HIERARCHIES.

3.2.2 OLAP OPERATIONS AND ARCHITECTURE

The OLAP client allows the end user to visualize the data using different types of diagrams (for example: bar charts, pie charts) and tables. It also allows the user to explore and analyze the data using different operators such as

1. **drill-down** – visualize a more detailed level within a dimension,
2. **roll-up** – or **drill-up**: visualize a more general level within a dimension,
3. **drill-across** – visualize another member or another dimension at the same level of detail, swap or
4. **pivot** – interchange visible dimensions or visible and background dimensions in order to modify the content of axes used in the diagrams or tables, and
5. **slice and dice** – reduce the dimensionality of the data, *i.e.*, take a projection of the data on a subset of dimensions for selected members of the other dimensions (Chaudhari, et al., 1997).

Figure 17 illustrates the OLAP operations roll-up, drill-down, slice and dice.

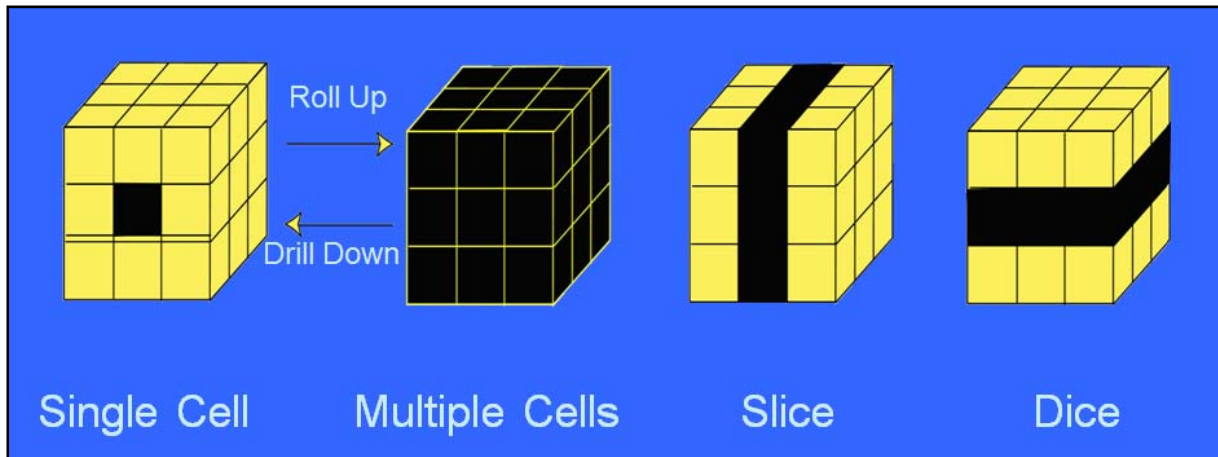


FIGURE 17: OLAP DATA EXPLORATION. AFTER: (PEDERSON, ET AL., 2001)

An OLAP system is built especially to navigate within multidimensional cubes, *i.e.* to go from one dimension to another in an interactive manner and to obtain fast responses. The underlying physical model supporting OLAP is centered on two major views. Depending on the technology (relational, multidimensional, etc.) used to implement the OLAP database it is possible to distinguish four OLAP approaches:

- Relational OLAP (ROLAP)** – relies on manipulating the data stored in the relational database to give the appearance of traditional OLAP's slicing and dicing functionality;
- Multidimensional OLAP (MOLAP)** – data is stored in a multidimensional cube. The storage is not in the relational database, but in proprietary formats; or
- Hybrid OLAP (HOLAP)** – This is a combination of the relational and multidimensional approaches (Pendse, 2000)

Traditional relational database system vendors propose the ROLAP architecture, whereas, others support the MOLAP architecture. The advantage of MOLAP is that it provides a direct multidimensional view of the data whereas the ROLAP architecture is just a multidimensional interface to the relational data. The ROLAP system has two advantages – 1) it can be easily integrated into other existing relational database systems, and 2) relational data can be stored more efficiently than multidimensional data. In ROLAP architecture, data is organized in a star or snowflake schema. The measures of interest for OLAP are stored in the fact table (e.g. dollar amount and units in the table SALES). For each dimension of the multidimensional model there exists a dimension table (e.g. Geography, Product, Time, and Account) with all the levels of aggregation and the extra properties of these levels.

SOLAP – SPATIAL OLAP

(Bédard, et al., 1997) presents a new solution combining the strengths of GIS with the strengths of OLAP coined as SOLAP or Spatial OLAP. The architecture of a SOLAP system is composed of a multidimensional structured spatio-temporal database, a SOLAP server and a SOLAP client (Rivest, et al., 2003). The necessity for the introduction of a SOLAP tool was the inability of the available tools to explore and analyze the spatio-temporal data available. Spatio-temporal exploration and analysis is a subset of exploratory data analysis (Glymour, et al., 1997). It combines two complementary processes, exploration and analysis, in order to identify hypotheses (e.g. potential patterns, associations and unusual occurrences) and determine their prevalence, their strength and predictability (Marchand, et al., 2001).

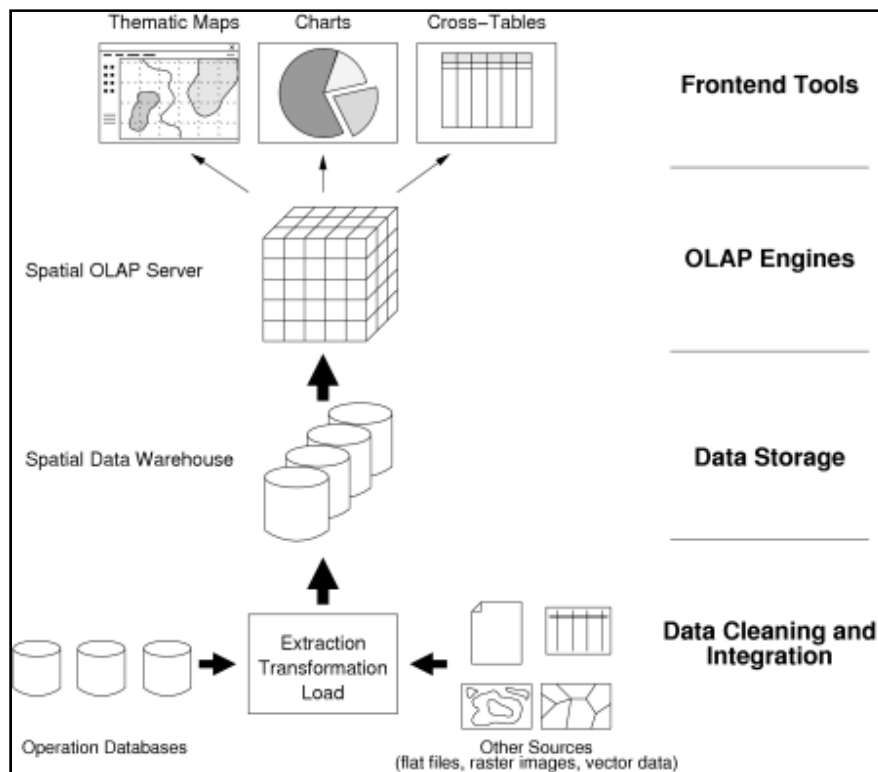


FIGURE 18: OLAP SYSTEM ARCHITECTURE. AFTER: (BALTZER, 2006)

Figure 18 illustrates the SOLAP tool as three-tier architecture. The division between the client, server and data is based on the web service architecture discussed in Section 3.1. A SOLAP solution offers a database supporting both multidimensional as well as spatial data. The server layer offering data access facilities with the client layer responsible for the visual representation of data. The GIS component can also access external data sources to display other relevant

information. The database level stores the geographical as well as operational data. The SOLAP server handles the spatio-temporal multidimensional database and the numerical and spatial calculations necessary to compute the measure values associated with possible combinations of dimension members. The SOLAP client can be defined as a category of software that allows navigation (i.e. to go from one fact to another) within spatial databases and that offers many levels of information granularity, many themes, many epochs, and many display modes, synchronized or not: maps, tables and diagrams (Bédard, et al., 2005)

In general, a SOLAP tool offers much functionality. These functionalities can be divided in the following three categories (Rivest, 2001):

A) VISUALIZATION OF CARTOGRAPHIC AND NON-CARTOGRAPHIC DATA

A SOLAP solution comprises of an OLAP client that permits the representation of multidimensional data through a seamlessly integrated cartographic display. Other conventional types of display include a tabular display and statistical charts, or diagram display. The flexibility to manage the display is another important aspect. The tabular, diagram and spatial displays must be accessible separately and/or simultaneously. The representation of one or more measures simultaneously is an added advantage the SOLAP solution should support. Another desirable characteristic is the capacity to display context data, for example, background maps, for the sole purpose of helping the user to locate the displayed information.

B) EXPLORATION OF DATA

OLAP solutions offer data exploration operations such as drill-down and roll-up and these operations should be directly available in all display types. Other OLAP operations like drill-across and pivot operations should also be made available and apply simultaneously to all displays. Exploration of several hierarchy levels with different granularity must also be supported. The availability of spatio-temporal analysis functions like metric and topological operations is an added benefit. The manipulation of the temporal dimension with a timeline in order to easily exploit the dimension should be included in the SOLAP tool.

C) SUPPORT FOR SPATIAL AND MULTIDIMENSIONAL DATA STRUCTURES

For mapping, spatial exploration, and spatial analysis of data, SOLAP tools should support all types of geometric primitives. The standardized primitives, as defined by ISO [ISO/TC 211 2001], are of particular importance. The support functionality to handle and permit various

representations of analyzed data is a must for a SOLAP server. The ability to manage and interact with different data sources to exploit data stored is an important facet of a SOLAP solution.

Current OLAP tools automatically aggregate numerical measures along hierarchies using different types of functions, for example distributive, algebraic and holistic. Likewise, aggregation functions for spatial data have also been defined. For example, spatial distributive aggregates include convex hull, geometric union, and geometric intersection. To include these in the OLAP concept would imply a real rethinking of OLAP concepts to support SOLAP tools. The introduction of spatial data in multidimensional model raises problems from implementation and theoretical point of view (Bimonte, et al.). They cite the following issues to be resolved: 1) modeling the spatial dimension, 2) requirement for an extended SOLAP algebra, and 3) the wide scope of visualization issues involved.

To conclude, although SOLAP tools offer spatio-temporal exploration possibilities, there are some major challenges that need to be overcome.

3.2.3 OLAP QUERYING CAPABILITIES

The OLAP architecture has been a controversial topic. Experts have argued about the pros and cons of various approaches to storing and processing dimensional data and which approach is the best to use for any given application. Of the many approaches to OLAP, four main options dominate (see Figure 19).

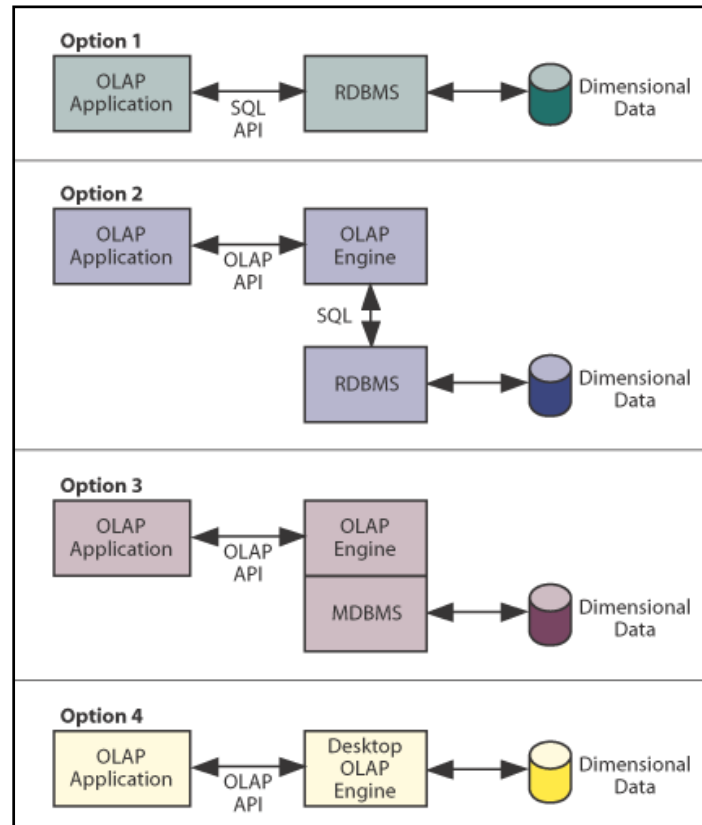


FIGURE 19: OLAP STORAGE OPTIONS. AFTER: (WHITE, 2003)

OPTION 1

Stores the data in a relational database (usually using a star or snowflake schema table design) and accesses it via SQL statements. The benefit of this option is that it avoids the need of a specialized multidimensional database product. However, this approach has often been criticized for poor performance and for the limited power of SQL as a language for OLAP.

OPTION 2 – ROLAP

Use an OLAP engine to retrieve data from an RDBMS and perform more complex processing on the retrieved data.

OPTION 3 – MOLAP

Store the data in a multidimensional database (in the form of a multidimensional array or *cube*) and access it using OLAP visual tools or language statements. The database may be managed by either a client- or server-based multidimensional DBMS (MDBMS). This approach has been popular because the MDBMS can be optimized for OLAP (i.e., the MDBMS incorporates an OLAP engine), which provides good performance, especially when large amounts of memory are available for array processing.

OPTION 4 – DOLAP¹³

Store small amounts of data in files on a user's desktop computer and access it via an OLAP engine. Often the data for manipulation is extracted from an RDBMS or MDBMS. To date, this option has been very popular, but the growth in the use of web-based thin clients means that companies are now looking for solutions that move client OLAP processing to web-based servers.

OPTION 5 – WOLAP

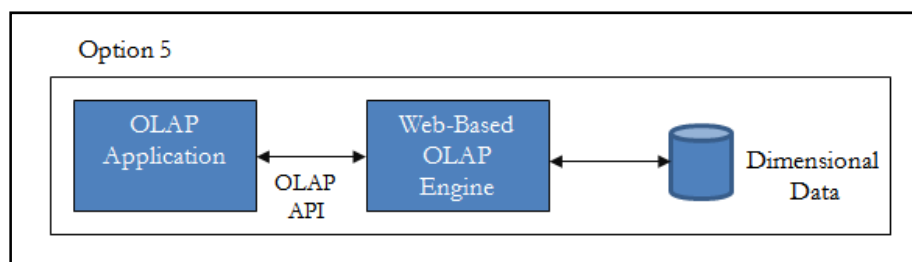


FIGURE 20: OLAP STORAGE OPTION FOR THE PROTOTYPICAL SOLUTION

Use a web-based OLAP or WOLAP engine to retrieve dimensional data from either a relational or multidimensional OLAP database. The author bases the proposed solution on the WOLAP architecture.

Unlike relational databases – which had SQL as the standard query language and wide-spread API¹⁴'s such as ODBC¹⁵, JDBC¹⁶, and OLEDB¹⁷ – there was no such unification in the OLAP domain for a long time. The first real standard API was OLE DB for OLAP (ODBO) specification from Microsoft and introduced the Multidimensional Expressions (MDX) query language. OLAP BAPI (Business Application and Programming Interface) is another interface introduced to query the multidimensional data. In 2001 Microsoft and Hyperion announced the XML for Analysis specification which was endorsed by the OLAP domain since it was based on standardized internet protocols. Although the interfaces are different, the underlying command language used to retrieve data, MDX, is the same between all the interfaces. The differences between the interfaces are entirely based on the programming model. Table 2 summarizes the differences.

¹³ Desktop OLAP – single-tier, desktop-based application

¹⁴ Application Programming Interface

¹⁵ Open Database Connectivity

¹⁶ Java Database Connectivity

¹⁷ Object Linking and Embedding, Database

INTERFACE	QUERY LANGUAGE	CALL INTERFACE	PLATFORM
ODBO	MDX	COM (Component Object Model) Protocol	Microsoft Windows platforms
OLAP BAPI	MDX	RFC (Remote Function Call) protocol	SAP support platforms ¹⁸
XMLA	MDX	XML over HTTP/SOAP	All platforms

TABLE 2: OVERVIEW OLAP QUERY INTERFACES

OLE DB FOR OLAP (ODBO)

OLE DB for OLAP (ODBO) is a Microsoft published specification and an industry standard for multidimensional data processing. It is the most widely supported multidimensional Application Programming Interface (API) to date. Platform-specific to Microsoft Windows, ODBO was specifically designed for OLAP systems by Microsoft as an extension to Object Linking and Embedding Database (OLE DB). ODBO uses Microsoft's Component Object Model (COM), and as such, it is somewhat complex and is limited to the Windows platform. Essentially, ODBO extends the ability of OLE DB to access multidimensional (OLAP) data stores, allowing users to perform sophisticated data analysis through fast, consistent, interactive access to a variety of data, both relational and multi-dimensional.

Because of it being limited to the windows platform it is not suitable for web service applications, which usually are based across many platforms and therefore the technology must be platform independent.

MULTIDIMENSIONAL EXPRESSIONS (MDX)

OLAP Services supports MDX functions as a full language implementation for creating and querying cube data. MDX is a declarative query language that can be used to extract information from a database. The output of a MDX query has the form of a cube itself. The query captures

¹⁸ See <http://service.sap.com/platforms> or <http://service.sap.com/pam> for a list of all supported platforms

both the output specification and the dimensionality of the resulting cube. MDX is superficially similar in many ways to SQL syntax that is typically used with relational databases. However, MDX is not an extension of the SQL language and is different from SQL in many ways.

```
SELECT axis specification ON COLUMNS,
axis specification ON ROWS
FROM cube_name
WHERE slicer_specification
```

FIGURE 21: MDX SYNTAX

Figure 21 illustrates the MDX syntax specification. In MDX, the select statement is used to specify a dataset containing a subset of multidimensional data. A basic MDX query consists of 4 parts –the number of axes (e.g. ON COLUMNS, ON ROWS) to be returned, the members for each dimension to include will be specified on each axis by using the *axis specification*, the name of the cube that sets the context of the MDX query and the members from a slicer dimension (*slicer specification*) on which data is sliced for members from the axis dimensions.

Even though MDX is not an open standard, but rather a Microsoft owned specification, it has been adopted by the OLAP community as a *de facto* industry standard.

OLAP BUSINESS APPLICATION PROGRAMMING INTERFACE (BAPI)

OLAP BAPIs are SAP¹⁹ defined objects that are implemented as RFC²⁰-enabled function modules. The methods defined are similar to ODOBO and XMLA. The interdependence with SAP platforms makes this unsuitable for a web service application.

XML FOR ANALYSIS (XMLA)

XMLA is a web service designed for connecting to multidimensional data sources. Similar to ODOBO, XMLA defines a XML-based API and interaction model between a *consumer* (application) and a *provider* (web service). The *openness* of XMLA makes it available across all

¹⁹ SAP is the world's largest business software company

²⁰ A remote function call (RFC) is a procedure for data interchange between a client and server.

platforms thus, the ideal querying interface for the proposed integrated application. It has been created as an initiative of the Microsoft Corporation and Hyperion Solutions Corporation to provide an open access for multidimensional databases. This standardized access enables a non-proprietary communication between client applications and OLAP data servers through the Internet. The XMLA implementation is based on the web services architecture and the description of its service is defined in terms of a WSDL document.

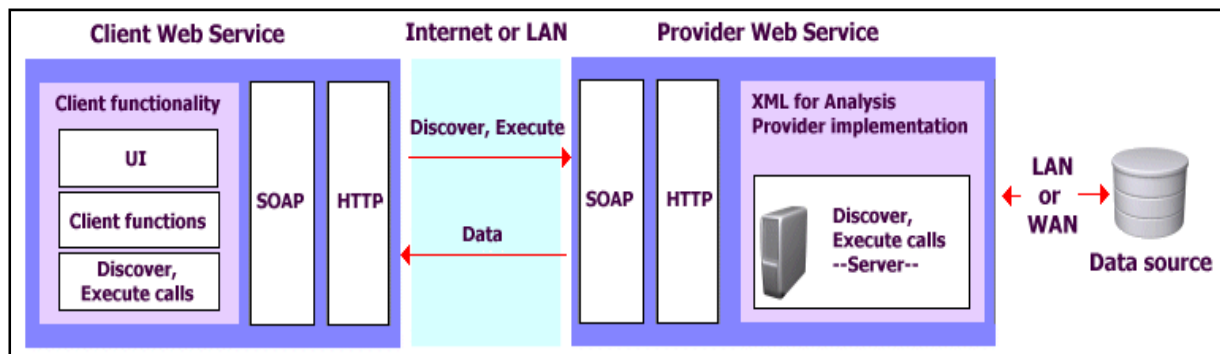


FIGURE 22: XMLA CONCEPT AND ARCHITECTURE. AFTER: (ZHAOHUI, ET AL., 2005)

XMLA supports two operations:

1. **Discover** – retrieves information, such as the list of available databases or details about a specific object, from an OLAP server.

```
<Discover>
  <RequestType>...</RequestType>
  <Restrictions>...</Restrictions>
  <Properties>...</Properties>
</Discover>
```

FIGURE 23: DISCOVER REQUEST SYNTAX

A DISCOVER request comprises of:

- 1.1. **RequestType** – Determines the type of schema rowset from which the DISCOVER method returns data.

The following Table 3 gives an overview ²¹ of the important schema rowsets supported by a XMLA provider.

²¹ For a full reference refer to the Microsoft TechNet website at (<http://technet.microsoft.com/en-us/library/ms126079.aspx>)

SCHEMA ROWSET	DESCRIPTION
MDSHEMA_CUBES Rowset	Describes the structure of the cubes within a database
MDSHEMA_DIMENSIONS Rowset	Describes the shared and private dimensions within a database
MDSHEMA_HIERARCHIES Rowset	Describes each hierarchy that is contained in a particular dimension
MDSHEMA_LEVELS Rowset	Describes each level within a particular hierarchy
MDSHEMA_MEASURES Rowset	Describes each measure within a cube
MDSHEMA_MEMBERS Rowset	Describes the members within a database

TABLE 3: XMLA DISCOVER SCHEMA ROWSETS

1.2. **Restrictions** – Contains columns and data used to restrict the information retrieved by the DISCOVER method. Each restriction column is defined by a separate XML element. The value of the restriction column is the data contained by the XML element, and the name of the restriction column corresponds to the name of the XML element.

1.3. **Properties** – represents the XMLA properties used to control aspects of the DISCOVER and EXECUTE methods, such as defining the information required to connect to the data source, specifying the return format of the result set, or specifying the locale in which the data should be formatted.

The following Table 4 gives an overview²² of the XMLA properties supported by an analysis server.

²² For a full reference refer to the Microsoft TechNet website at(<http://technet.microsoft.com/en-us/library/ms186627.aspx>)

PROPERTY NAME	DESCRIPTION
CATALOG	Identifier to be used when establishing a session with an Analysis Server. Identifies the database to connect to.
CUBE	Identifies the cube that sets the context of the command.
DATA SOURCE INFO	Contains information, such as the instance name required to connect to the data source
FORMAT	Determines the type of result set that is returned from the DISCOVER and EXECUTE methods

TABLE 4: XMLA SUPPORTED PROPERTIES

2. **Execute** – request involving data transfer, such as retrieving or updating data on the server. The operation has two parameters: *Command* – command to be executed (e.g. MDX, DMX or SQL) and *Properties* –XMLA list of properties (see Table 4).

```

<soap:Envelope>
  <soap:Body>
    <Execute xmlns="urn:schemas-microsoft-com:xml-analysis">
      <Command>
        <Statement>SELECT Measures.MEMBERS ON COLUMNS FROM Sales</Statement>
      </Command>
      <Properties>
        <PropertyList>
          <DataSourceInfo/>
          <Catalog>FoodMart</Catalog>
          <Format>Multidimensional</Format>
          <AxisFormat>TupleFormat</AxisFormat>
        </PropertyList>
      </Properties>
    </Execute>
  </soap:Body>
</soap:Envelope>

```

FIGURE 24: XMLA SAMPLE EXECUTE REQUEST

These operations enable the access to both data and metadata stored in a multidimensional OLAP Server. As these methods are invoked through SOAP, the input and output are XML documents.

In conclusion, the abovementioned discussed the concepts, architecture and querying mechanism of an OLAP server. The knowledge gained from this section forms the base of the prototype solution presented in Chapter 5. OLAP offers a wide palette of architectural and querying possibilities. The author has decided to make use of the WOLAP architecture and the XMLA

web service as the querying service. This decision is influenced by the characteristics discussed in Section 1.3 and Chapter 2.

3.3 OGC WEB SERVICES – INTEROPERABLE GEOPROCESSING

The Open GIS Consortium, Inc. (OGC), a not-for-profit trade association dedicated to promoting new technical and commercial approaches for interoperable geoprocessing, was founded in 1994 in response to widespread recognition of the problem of non-interoperability and its many negative ramifications for industry, government, and academia (Bühler, et al., 1998). (Bühler, et al., 1998) define interoperability as: *THE ABILITY FOR A SYSTEM OR COMPONENTS OF A SYSTEM TO PROVIDE INFORMATION PORTABILITY AND INTER-APPLICATION, COOPERATIVE PROCESS CONTROL. INTEROPERABILITY, IN THE CONTEXT OF THE OPENGIS SPECIFICATION, IS SOFTWARE COMPONENTS OPERATING RECIPROCALLY (WORKING WITH EACH OTHER) TO OVERCOME TEDIOUS BATCH CONVERSION TASKS, IMPORT/EXPORT OBSTACLES, AND DISTRIBUTED RESOURCE ACCESS BARRIERS IMPOSED BY HETEROGENEOUS ENVIRONMENTS AND HETEROGENEOUS DATA.* OGC envisions the full integration of geospatial data and geoprocessing resources into mainstream computing and the widespread use of interoperable, commercial geoprocessing software throughout the global information infrastructure (Bühler, et al., 1998). OGC is not a *de jure* standards organization. It is an industry consortium whose consensus process involves technology users and providers and whose objective is market enablement through interoperability between commercial geoprocessing software products. OGC adds implementation detail to the standards of *de jure* standards organizations, and extends *de jure* standards whenever the marketplace needs stricter guidelines to experience greater growth.

Because there are so many incompatible standards in the geo-information technology domain, geospatial information and geoprocessing are not part of most information systems, and sharing geodata between geoprocessing systems and between user communities requires considerable time and expertise. OGC addresses the profusion of data format and data transfer standards by creating open, common interfaces between software system components, and letting those systems use any data format internally. (Bühler, et al., 1998) define an open system as: *A SYSTEM THAT IMPLEMENTS OPEN INTERFACE SPECIFICATIONS AND STANDARDS THAT PROMOTE APPLICATION PORTABILITY, SCALABILITY, INTEROPERABILITY, DIVERSITY, MANAGEABILITY, EXTENSIBILITY, COMPATIBILITY WITH LEGACY COMPONENTS, AND USER PORTABILITY.* These OGC interfaces provide

access to both information and functionality. OGC web services can be grouped into three basic categories (Alameh, 2002):

- a) **Data services** (such as the OGC Web Mapping, Web Coverage and Web Feature Services) offer customized data to users. These services are tightly coupled with specific data sets.
- b) **Processing services** are not associated with specific datasets. Instead they provide operations for processing or transforming data in a manner determined by user-specified parameters (Alameh, 2001). Such services can provide generic processing functions such as projection/coordinate conversion, rasterization/vectorization, map overlay, imagery manipulation, or feature detection and imagery classification.
- c) **Registry/catalogue services** are used to classify, register, describe, search, maintain and access information about web services. Types of registries are differentiated by their role such as registries for cataloguing data types, online data instances, service types and online service instances. The contents of registry/catalogue services are metadata describing other services.

Figure 25 illustrates the OWS Framework designed to provide a framework for coordinated development of new and extended services. As also, enable interoperable services through standard interfaces and encoding etc. The framework is based on the Publish/Find/Bind schema discussed in Section 3.1. Application services, registry services, encodings, data services, portrayal services, and processing services make up the framework.

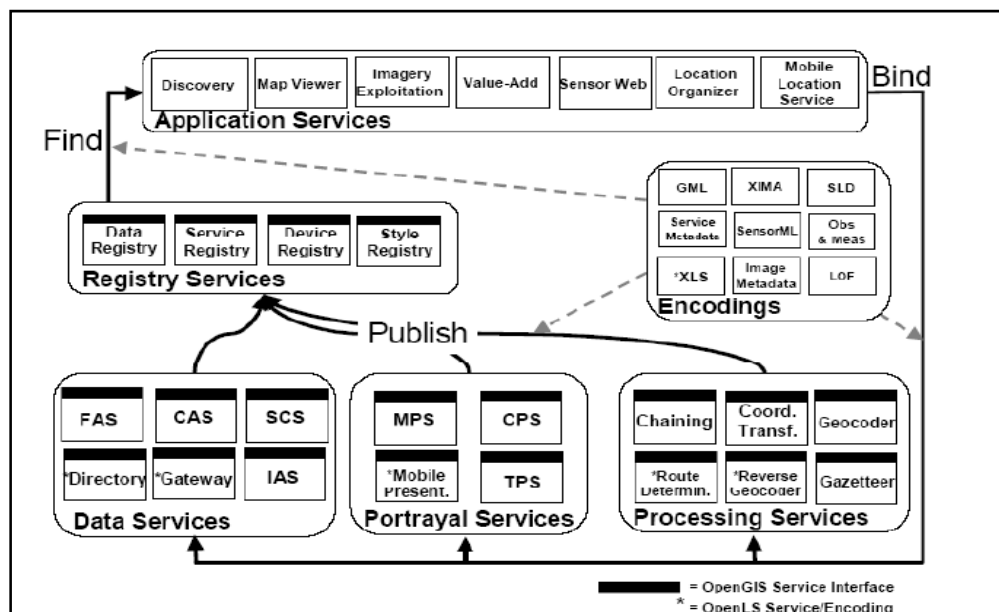


FIGURE 25: THE OPEN WEB SERVICE (OWS) FRAMEWORK. AFTER: (OPENGIS, 2003)

In order for a sustainable and extensible GIS web service architecture to exist, the basic services should be accessed via standardized interfaces. The International Organization for Standardization defines standards as: *DOCUMENTED AGREEMENTS CONTAINING TECHNICAL SPECIFICATIONS OR OTHER PRECISE CRITERIA TO BE USED CONSISTENTLY AS RULES, GUIDELINES, OR DEFINITIONS OF CHARACTERISTICS, TO ENSURE THAT MATERIALS, PRODUCTS, PROCEDURES, AND SERVICES ARE FIT FOR THEIR PURPOSE.*

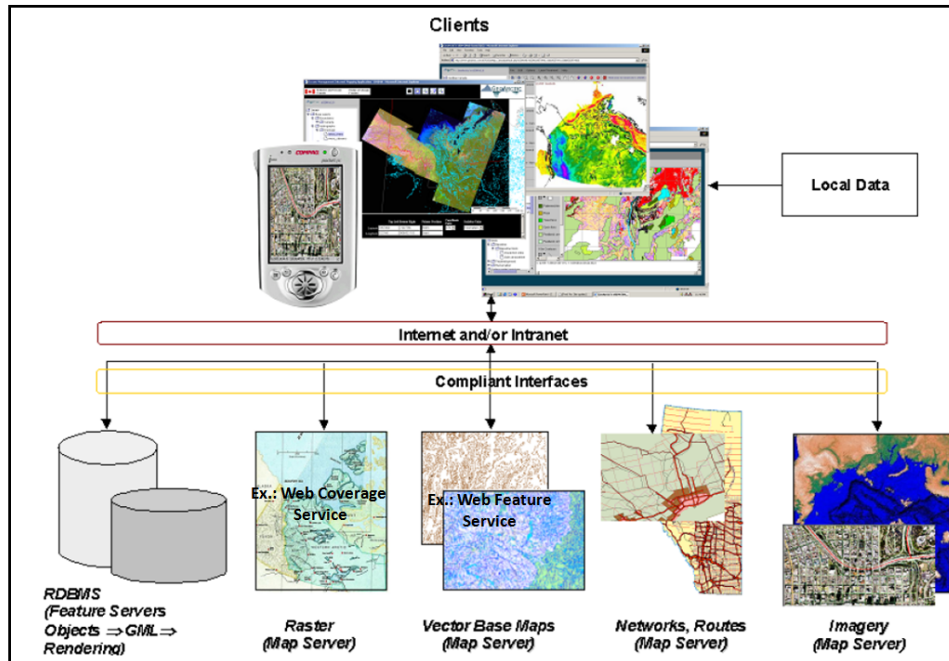


FIGURE 26: SUMMARY - OGC data services. AFTER: (WOJNAROWSKA, ET AL., 2002)

Figure 26 illustrates the deployment of the OGC specified services. The various data services provide raw GIS data through compliant interfaces for a compatible client to be displayed to the user.

This service-based model is rapidly materializing as a result of the advancements in general web service technologies and the focused efforts of the OGC in the areas of service categorization and interoperability of service interfaces [(Alameh, 2002); (Bühler, et al., 1998)]. In the following section the three major data services, the Web Map Service, the Web Feature Service and the Web Coverage Service are dealt with in detail. The integrated solution concentrates on data integration and portrayal, hence the focus on the aforementioned technologies.

3.3.1 WEB MAP SERVICE

A Web Map Service produces maps of georeferenced data. The WMS specification (Beaujardière, 2002) defines a map as: *A VISUAL REPRESENTATION OF GEODATA*; A map is not the data itself. The maps are generally rendered in a pictorial format such as PNG, GIF, or JPEG, or occasionally as vector-based graphical elements in Scalable Vector Graphics (SVG) or in Web Computer Graphics Metafile (WebCGM) formats. The specification defines three WMS operations:

- a) **GetCapabilities** – returns service-level metadata, which is a description of the service’s information content and acceptable request parameters in XML;
- b) **GetMap** – returns a map image whose geospatial and dimensional parameters are well-defined;
- c) **GetFeatureInfo** – returns information about the particular features shown on a map.

A basic WMS classifies its georeferenced information holdings into *Layers* or *Themes* and offers a finite number of predefined *Styles* in which to display those layers. The behavior of a Web Map Service can be extended to allow user-defined symbolization of feature data instead of named layers and styles. Furthermore, individual map layers can be requested from different servers. The WMS specification thus enables the creation of a network of distributed map servers from which clients can build customized maps. Figure 27 illustrates the interoperable web mapping structure where each map server is accessed by the client through a common interface. In a distributed OGC WMS, a WMS server can also run as a WMS client that requests capabilities and maps from other WMS servers.

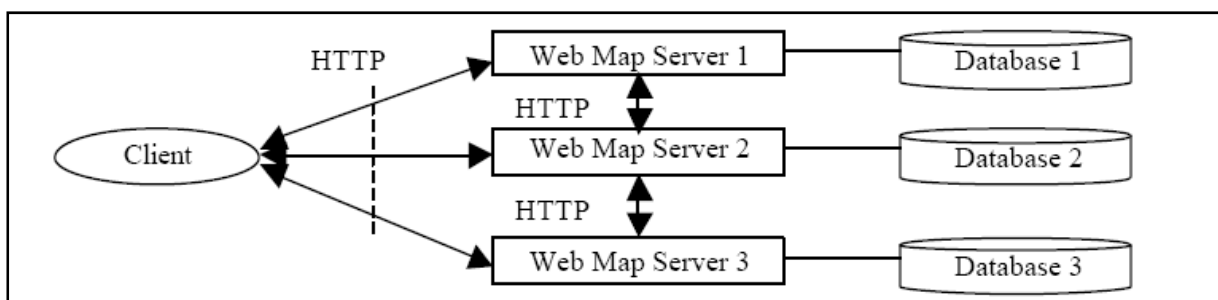


FIGURE 27: STANDARD INTEROPERABLE WEB MAPPING ARCHITECTURE. AFTER: (DING, ET AL., 2002)

A *Cascading Map Server* is a WMS that behaves like a client of other Web Map Services and behaves like a WMS to other clients. In terms of the client-server architecture, a Cascading Map Server is a mediator for Web Map Services.

Figure 28 illustrates the sequence of requests and responses between a WMS client and server application.

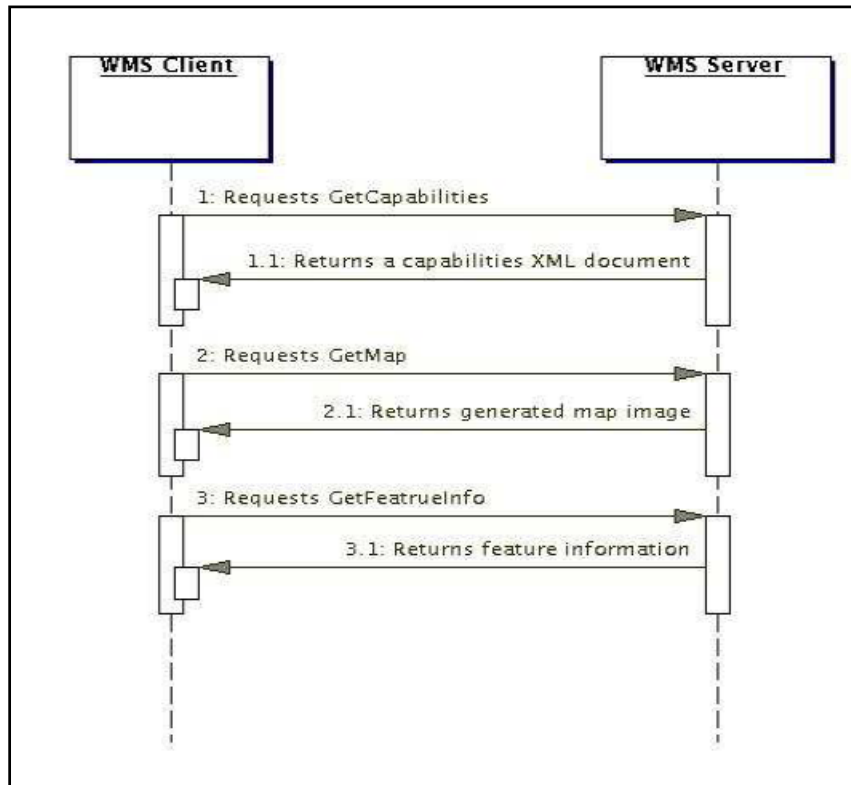


FIGURE 28: WMS PROTOCOL DIAGRAM

3.3.2 WEB FEATURE SERVICE

The OGC WMS allows a client to overlay map images for display served from multiple Web Map Services on the internet. In a similar fashion, the OGC Web Feature Service allows a client to retrieve geospatial (vector-) data encoded in GML from multiple Web Feature Services.

Besides the operations to retrieve data, the WFS describes operations to edit (insert, update, delete) data on the level of a feature. The ability to edit features makes the WFS the only open, interoperable standard with which data can be edited over the internet. The WFS specification defines the following four operations:

- a) **GetCapabilities** – returns the capabilities document that contains a description of all the operations that the server supports and a list of all feature types²³ that it can service.
- b) **GetFeature** – retrieves the feature(s) requested by the client application.

²³ Feature type – a class of features. Feature instances are grouped into classes with common characteristics

- c) **DescribeFeatureType** – the operation returns a description of the feature class the client has currently requested for.
- d) **Transaction operations** – the transactional operations – **InsertFeature**, **UpdateFeature**, **DeleteFeature** to edit the data.

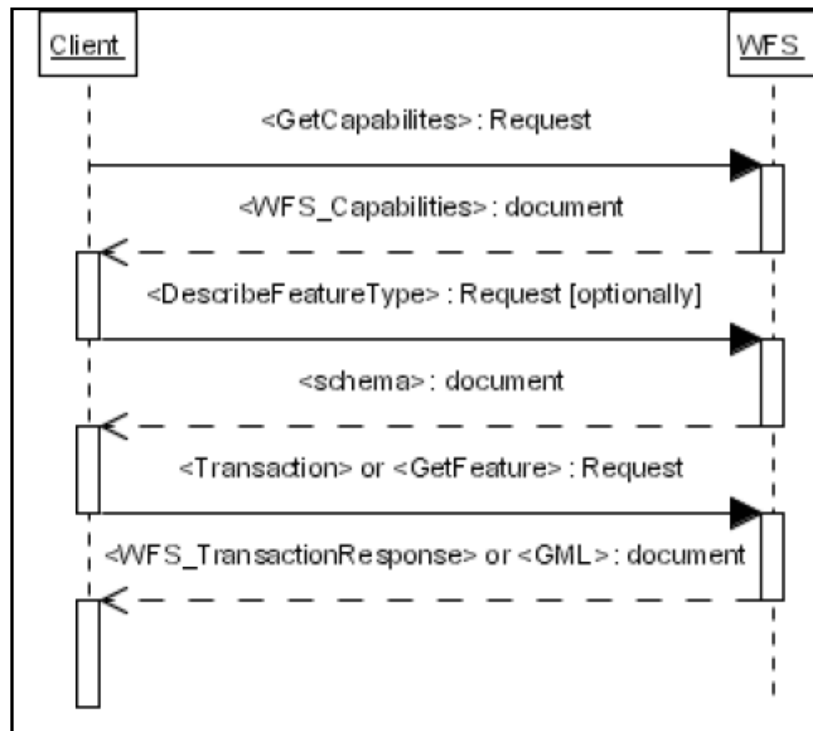


FIGURE 29: WFS PROTOCOL DIAGRAM.

Figure 29 illustrates transfer of requests and responses between the client and the Web Feature Service. Each valid request sent by the client is responded with a valid response document as specified by the OGC. The standardization of querying and responses forms the backbone of interoperable operations between different web GIS solutions.

3.3.3 WEB COVERAGE SERVICE

An OGC Web Coverage Service (WCS) extends the Web Map Server (WMS) interface to support electronic interchange of geospatial data as *coverages* – that is, digital geospatial information representing space-varying phenomena. It provides access to potentially detailed and rich sets of geospatial information, in forms that are useful for client-side rendering, multi-valued coverages, and input into scientific models and other clients.

The WCS may be compared to the WMS and the WFS; like them it allows clients to choose portions of a server's information holdings based on spatial constraints and other criteria. Unlike WMS which filters and portrays spatial data to return static maps (rendered as geo-registered pictures by the server), the Web Coverage Service provides available data together with their detailed descriptions. It allows complex queries against these data and returns data with its original semantics (instead of pictures) which can be interpreted, extrapolated, etc. and not just portrayed. Unlike WFS which returns discrete geospatial features, the Web Coverage Service returns representations of space-varying phenomena that relate a spatial-temporal domain to a (possibly multidimensional) range of properties. The specification defines three operations:

- a) **GetCapabilities** – An OGC specified GetCapabilities operation returns a complete service description.
- b) **DescribeCoverage** – lets clients request a full description of one or more coverages served by a particular WCS server. The server responds with an XML document that fully describes the identified coverages;
- c) **GetCoverage** – returns a coverage bundled in a well-known coverage format.

WCS DATA MODEL –THE COVERAGE

The WCS is the standard for basic retrieval from large scale, multidimensional data (e.g. raster data) better known as coverages. The International Standards Organization (ISO) defines coverage as: *COVERAGES SUPPORT MAPPING FROM A SPATIOTEMPORAL DOMAIN TO ATTRIBUTE VALUES WHERE ATTRIBUTE TYPES ARE COMMON TO ALL GEOGRAPHIC POSITIONS WITHIN THE SPATIOTEMPORAL DOMAIN. A SPATIOTEMPORAL DOMAIN CONSISTS OF A COLLECTION OF DIRECT POSITIONS IN A COORDINATE SPACE. EXAMPLES OF COVERAGES INCLUDE RASTER, TRIANGULATED IRREGULAR NETWORKS, POINT COVERAGES, AND POLYGON COVERAGES. COVERAGES ARE THE PREVAILING DATA STRUCTURES IN A NUMBER OF APPLICATION AREAS, SUCH AS REMOTE SENSING, METEOROLOGY, AND BATHYMETRIC ELEVATION, SOIL, AND VEGETATION MAPPING.* Coverages are often represented as a map, which is a term from mathematics used to describe a projection of values from a range function onto a sample space or domain set. This abstract notion of a coverage as a function is also supported in (Baumann, 1999). Figure 30 illustrates the mapping between the spatio-temporal domain and the range domain on the basis of a function f .

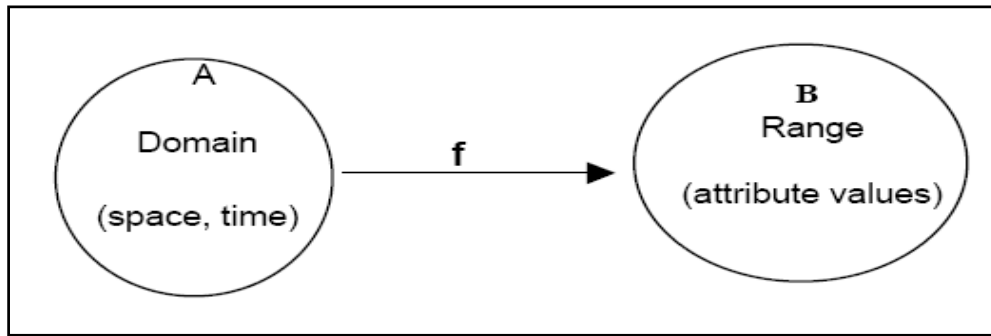


FIGURE 30: CONCEPTUAL MODEL OF A COVERAGE. AFTER: (211/WG, 2004)

Coverage is seen as 2D, 3D, or 4D matrix of values. The axes have a location ‘X’, ‘Y’, and ‘Z’ or time ‘T’ semantics. Following the functional viewpoint, the axes collectively are called coverage domain, the coverage values form the coverage range.

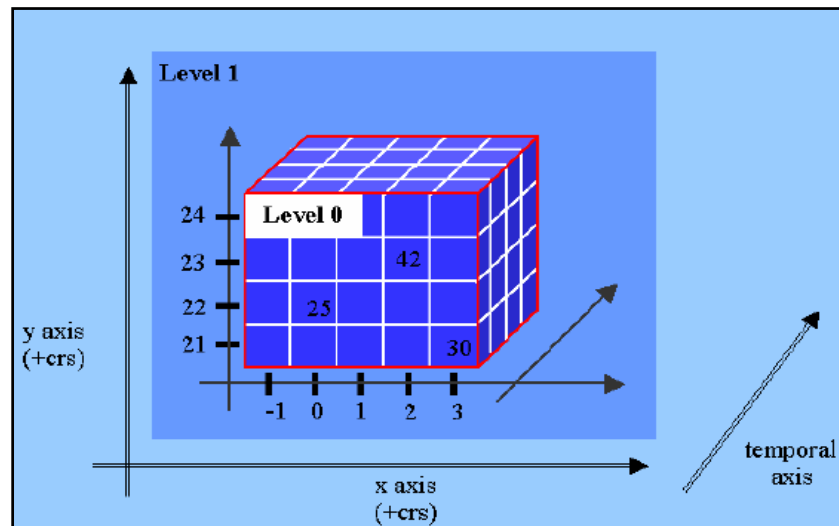


FIGURE 31: CONCEPTUAL (GRAPHIC) COVERAGE MODEL. AFTER: (BAUMANN, 2006)

Figure 31 depicts a conceptual coverage model. Coverage can be viewed as a cube with the x-, y-, and temporal axis as dimensions.

As far as the general geo-information framework is concerned, coverage is a specialized feature type. Figure 32 illustrates the specialized types of the feature.

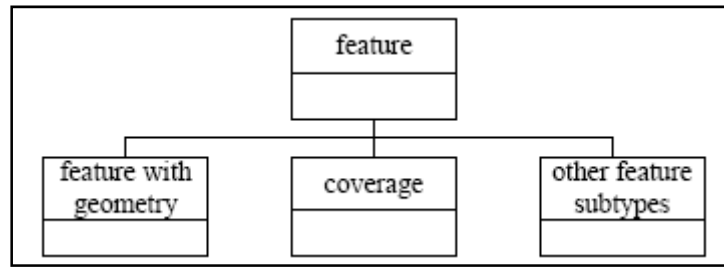


FIGURE 32: FEATURE SUBTYPES. BASED ON: (211/WG, 2004)

Coverage can be designed to represent a single feature or a set of features. A requirement to model coverages is that the same quantities are measured at different locations. The distribution of the locations does not need to be regularly gridded nor do they need to represent continuous values across the domain. Because the same attributes are located at different spatio-temporal locations, one may view the collections of features as a single conceptual entity.

The following figure shows a subset of the coverage types introduced by ISO 19123.

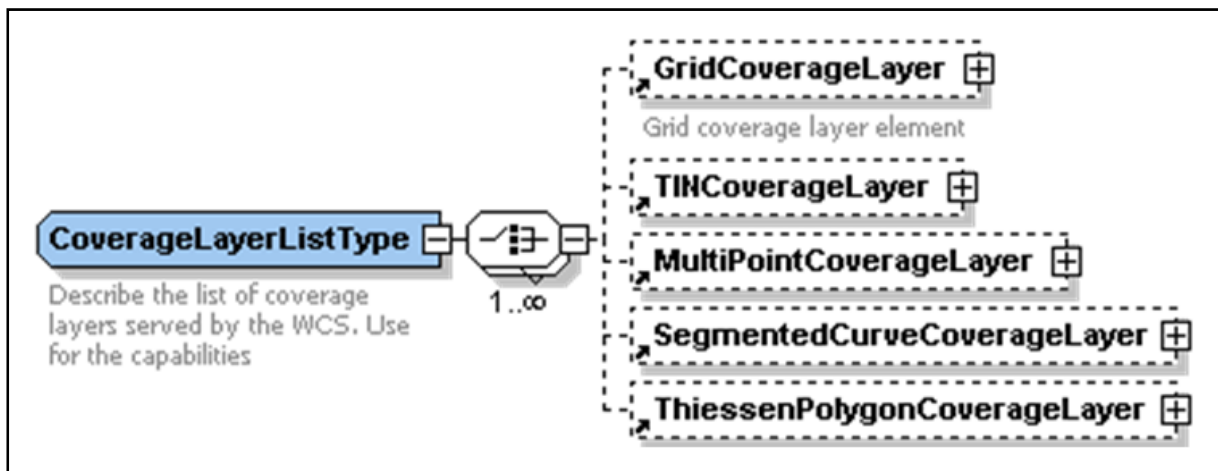


FIGURE 33: LIST (PARTIAL) OF SUPPORTED COVERAGE TYPES. BASED ON: (211/WG, 2004)

MultiPointCoverageLayer type is characterized by a finite domain consisting of points. Generally, the domain is a set of irregularly distributed points; the principal use of multipoint coverages is to provide a basis for continuous coverage functions. The domain of GridCoverageLayer is a set of GridPoints that are associated with records of feature attribute values through a GridValuesMatrix element.

Coverages are defined in the GML specification 3.1.1²⁴. Figure 34 illustrates the Coverage description as defined in GML 3.1.1. A valid coverage description has a *CoverageOfferingType* which in turn is represented by a *CoverageOfferingBriefType* and the elements *domainSet*, *rangeSet* as well as a list of supported coordinate reference systems (*supportedCRSs*); output formats (*supportedFormats*) and interpolation functions (*supportedInterpolations*). Further details and explanations are handled in Chapter 5.

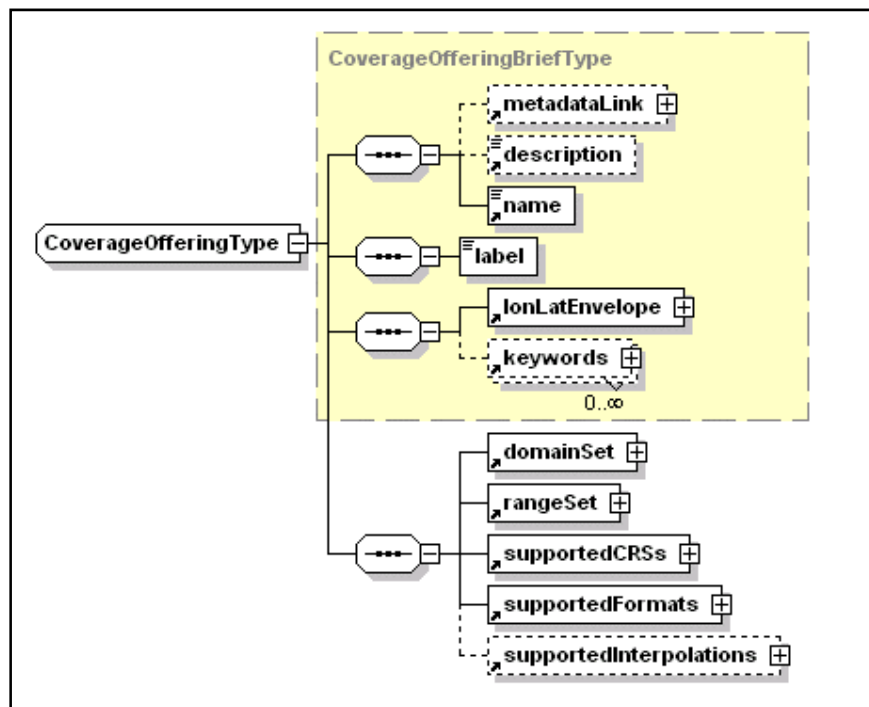


FIGURE 34: COVERAGE UML (PARTIAL) SCHEMA. BASED ON: (211/WG, 2004)

SPATIOTEMPORAL DOMAIN OF A COVERAGE: DOMAINSET

A spatiotemporal domain is a set of geometric objects described in terms of direct positions. The direct positions are associated with a spatial or temporal coordinate reference system. Commonly used spatiotemporal domains include point sets, grids, collections of closed rectangles, and other collections of geometric objects.

²⁴ Geography Markup Language (version 3.1.1) as described in an Encoding Specification by the OGC. Available at: http://portal.opengeospatial.org/files/?artifact_id=4700.

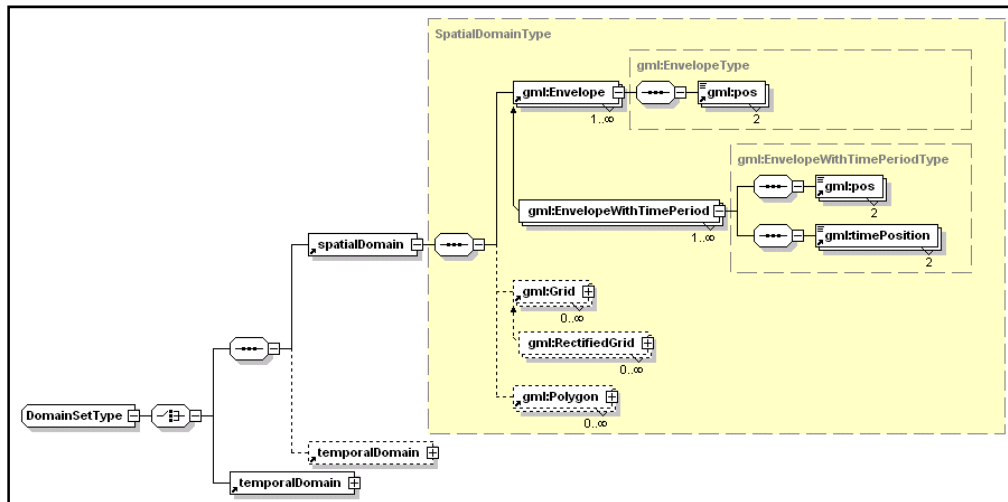


FIGURE 35: SPATIOTEMPORAL DOMAIN UML SCHEMA DIAGRAM. BASED ON: (211/WG, 2004)

THE RANGE OF A COVERAGE: RANGESET

The range of a coverage is a set of feature attribute values. Coverages often model many associated functions sharing the same spatiotemporal domain. Therefore, the value set is represented as a collection of records with a common schema. A coverage's range consists of a list of fields; each field either can be atomic (such as temperature), or have a composite value. A composite field consists of an n-dimensional array which can be addressed along the axes (in WCS lingua: keys chosen from key lists).

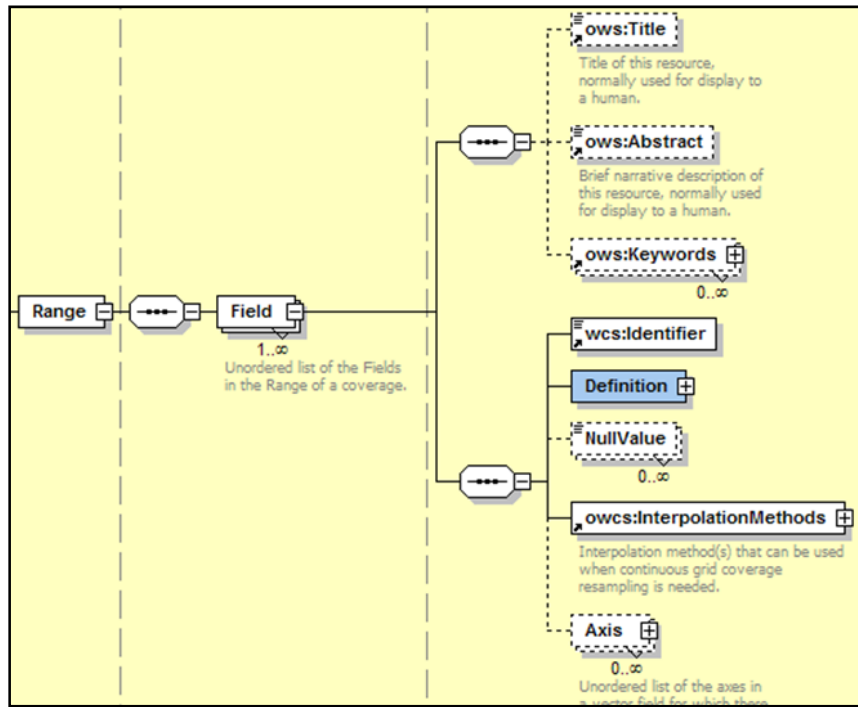


FIGURE 36: RANGESET UML SCHEMA DIAGRAM

The data model coverage has been identified as the key concept for bridging the gap between the OLAP and the GIS data model. The mapping between the two can be conceptualized based on the fact that the coverage data model is similar to the OLAP cube model in that the coverage can be seen as a cube having at least three dimensions (spatial, temporal and attribute dimension).

DATA EXPLORATION CAPABILITIES

The WCS supports the networked interchange of multidimensional and multi-temporal geospatial data as coverages through the three mandatory operations (aka request), GetCapabilities, DescribeCoverage, and GetCoverage. The request may be encoded as key-value pairs (KVP), or as an XML document using HTTP GET or POST methods (Evans, 2005). A simplified protocol diagram (see Figure 37) illustrates the messages that might be passed back and forth between client applications and the WCS server via a specified interface in order to process a typical request.

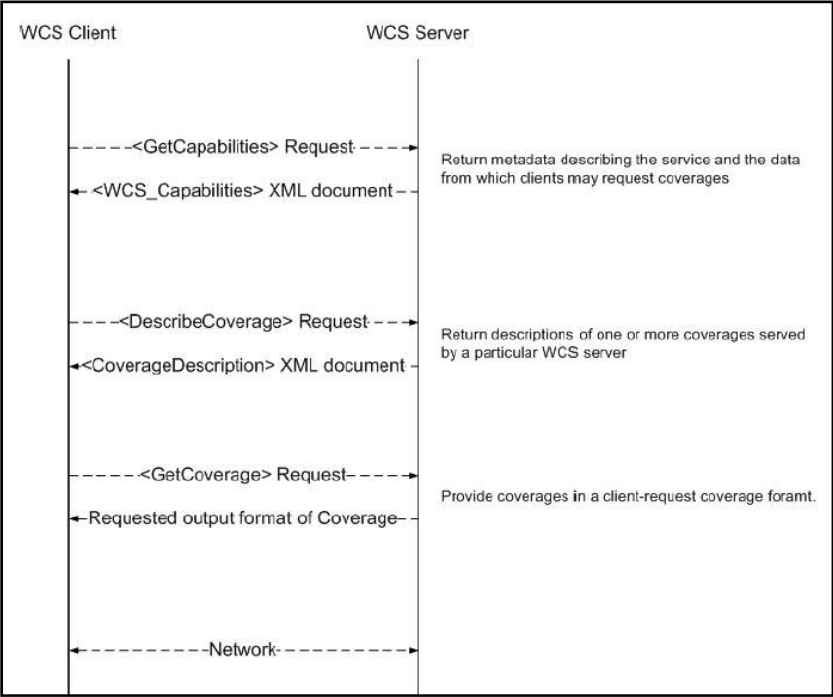


FIGURE 37: WCS PROTOCOL DIAGRAM.

The basic components of a request are listed in the Table 5. The ability of the query to explore the spatio-temporal data model is a very important aspect for the WCS being selected as the GIS interface for the SOLAP solution.

	GETCAPABILITIES	DESCRIBECOVERAGE	GETCOVERAGE
URI REQUEST PARAMETERS	REQUEST =GetCapabilities SERVICE =WCS	http://server_url/path/script? REQUEST =DescribeCoverage SERVICE =WCS VERSION =1.0.0	http://server_url/path/script? REQUEST =GetCoverage SERVICE =WCS, VERSION =1.0.0 COVERAGE =name CRS =crs_identifier, BBOX =minx, miny, maxx, maxy, minz, maxz TIME =time1, time2, ... min/max/res WIDTH =w, HEIGHT =h, DEPTH =d FORMAT =format
XML REQUEST COMPONENTS	SERVICE	SERVICE VERSION	SERVICE VERSION <sourceCoverage> <DomainSubset> <RangeSubset> <InterpolationMethod> <output>
WCS RESPONSE	<WCS_Capabilities> <Service> <Capabiltiy> <ContentMetadata> ... <WCS_Capabilities>	<CoverageDescription> <CoverageOffering> <domainSet> <rangeSet> ... </CoverageDescription>	Coverage <i>view</i> extracted as specified by the request parameters.

TABLE 5: COMPONENTS OF WCS OPERATIONS

3.3.4 SUMMARY

The standardized service interfaces from the OGC, including WMS, WFS and WCS have commonalities in their design. All are interfaces handling geographical data over a distributed environment. However, each of them serves a different purpose. Table 6 compares the most important operations/requests.

OPERATION/REQUEST	WMS	WFS	WCS
DESCRIBE CAPABILITIES	Yes	Yes	Yes
RETRIEVAL OF GRAPHICS	Yes	No	No
RETRIEVAL OF UNRENDERED / RAW GEODATA	No	Yes	Yes
INFORMATION ON A SPECIFIC FEATURE	Yes	Yes	Yes
DATA MODELS	Static Map images (PNG, BMP, SVG, etc)	Discrete geospatial Features (GML format): object- centric, heterogeneous collection of properties, “summary-view”, relational view	Space-varying phenomena Coverages: Property-centric, variation of homogeneous property, patterns and anomalies, multidimensional view
EXPLORATION OF DATA IS POSSIBLE BY	Bounding Box, Time, Elevation	Bounding Box, Time, Elevation, values of attributes	Spatial and Non-Spatial filters ²⁵
EDITING CAPABILITIES	No	Yes	No

TABLE 6: OGC WEB SERVICES SUMMARY

²⁵ Filter Encoding: defines an XML encoding for filter expressions. A filter expression constrains property values to create a subset of a group of objects. The goal, typically, is to operate on just those objects by, for example, rendering them in a different color or saving them to another format.

Both the WFS and WCS are strong contenders for the integrated GIS platform. The WMS is not considered due to the fact that it is a *pure* visualization service with no GIS analysis functionalities. The requesting client does not get raw data but a readymade map which it merely displays. The strengths of the WFS lies in its capability to not only query features based on their geometry properties but also, with the help of the Filter Encoding Specification, query feature characteristics. In WFS, the requesting client gets the raw data for the geographic feature it had requested. This means that geographic coordinate data such as line, point and polygon features are returned. It is the prerogative of the requesting machine to use the returned information in whatever manner it wants. The information is returned in the form of GML. Another advantage of the WFS specification is the ability to edit feature information using the transactional operations as defined by the specification. Although the WFS offers a multitude of optimal functionalities as seen above the major deficiency is the querying capability. A basic WFS query is not suited for multidimensional exploration. The specification only allows filter expressions to be added to the query which leads to a greater level of complexity. The predicate or filter language must be defined in XML and be derived for the Common Query Language (CQL) as defined in the OpenGIS Catalogue Interface Implementation Specification.

The author proposes to use the Web Coverage Service as the GIS platform for the integration. Similar to the WFS the WCS allow access to raw geographic data which can be used by the client for further analysis or visualization. The querying capabilities of the WCS include exploration along the spatial, temporal and attribute property subset. Thus a WCS query inherently supports multidimensional exploration. The definition of the WCS data model, the coverage, supports a multidimensional data representation. The coverage can be conceptually visualized as a cube structure thus drawing similarities with the already present cube structure of the OLAP solution. The similarities in the data model and the multidimensional data exploration capacity of the WCS make it the suitable GIS supplement in the prototype solution.

3.4 RESEARCH APPROACH – REVISITED

As mentioned in Chapter 1 the research approach has been divided into three parts. In this section the author revisits the research approach and introduces the criteria on which the existing solutions will be evaluated and presents an overview to the prototype implementation.

The research problem as stated in Section 1.2: *AN INTEROPERABLE WEB-BASED HYBRID SOLAP SOLUTION SUPPORTING SPATIAL MULTIDIMENSIONAL INFORMATION EXPLORATION BY ABSTRACTING THE COMPLEXITY OF SEPARATELY QUERYING THE DATA* shows the way for the OGC specified data service to be selected. Table 7 summarizes the consolidated information gained from Section 1.3 and from the scenarios in Chapter 2 and offers an explanation for the same. This table acts as a reference for the author to test the existing solutions and the proposed prototype solution.

CHARACTERISTICS	EXPLANATION	RELEVANT ARCHITECTURE AND COMPONENTS
INTEROPERABLE	The application should be able to access other applications/tools and legacy systems through interoperable interfaces, such as web services, at different levels, such as metadata, data, and analysis services	Service Chaining (Alameh, 2003)
COMPLIANT WITH OGC STANDARDS AND SPECIFICATIONS	The integrated application should support geospatial specifications, such as OGC Web Service, and standards such as ISO/TC 211 standards, so that relevant geospatial applications can be accessed in consistent, predictable and interchangeable manner.	OGC Web Services
COMPLIANT WITH OLAP STANDARDS AND SPECIFICATIONS	The OLAP domain supports de facto industry standards that should also be supported.	OLAP de facto standards (XMLA)
COMPLETE GIS ANALYSIS FUNCTIONALITIES	The GIS analysis functionalities such as buffering, proximity or neighborhood analysis	OGC Web Services (e.g. WFS, Filter Encoding)
COMPLETE OLAP ANALYSIS FUNCTIONALITIES	Functionalities like Drill-Down, Drill-Up, Pivot etc.	OLAP server, MDX
VENDOR NEUTRAL	Legacy systems were developed based on different vendor solutions. An interoperable application should comply with community standards and be independent of vendor specific requirements.	Service discovery and service chaining
SCALABLE AND EXTENSIBLE	The application should comply with the general software engineering requirements on component reuse and expansion. Here the use of the OGC specifications would help make the application scalable and extensible.	Publish-Find-Bind
DISTRIBUTED COMPUTING (EX. WEB SERVICES)	The application should comply with web interfacing standards, such as HTTP for communication, XML for encoding requests and responses.	OGC Web Services, XMLA
QUERYING CAPABILITIES	The application should offer a single querying interface to abstract the querying complexity associated with exploring two different data types with two different application interfaces.	OGC Web Service (e.g. WCS)

TABLE 7: RESEARCH PROBLEM CHARACTERISTICS SUMMARY

In an integrated interface that intends to be extensible and interoperable it is very important to be able to clearly separate the functionalities that are the exclusive responsibility of the integrities. In the case of the proposed solution the WCS acts as the frontend interface intercepting the incoming client request and processing it to serve spatial data and relies on the OLAP solution to retrieve the multidimensional data. The WCS overlooks the query and response management by responding to the client request with an augmented spatio- multidimensional result set.

The architectural similarities between the two technologies, OLAP and WCS, and the use of standardized protocols allowing the interoperable exchange of information has been used as the starting point for the integration. The 3-tier architecture²⁶ (see Section 3.1), predominant in web services, divides the interface into three distinct sections. First, the client/visual tier. Second, the server/middleware tier and third, the data tier. The proposed solution concentrates on the augmentation of the server level to serve a multitude of data from the data tier.

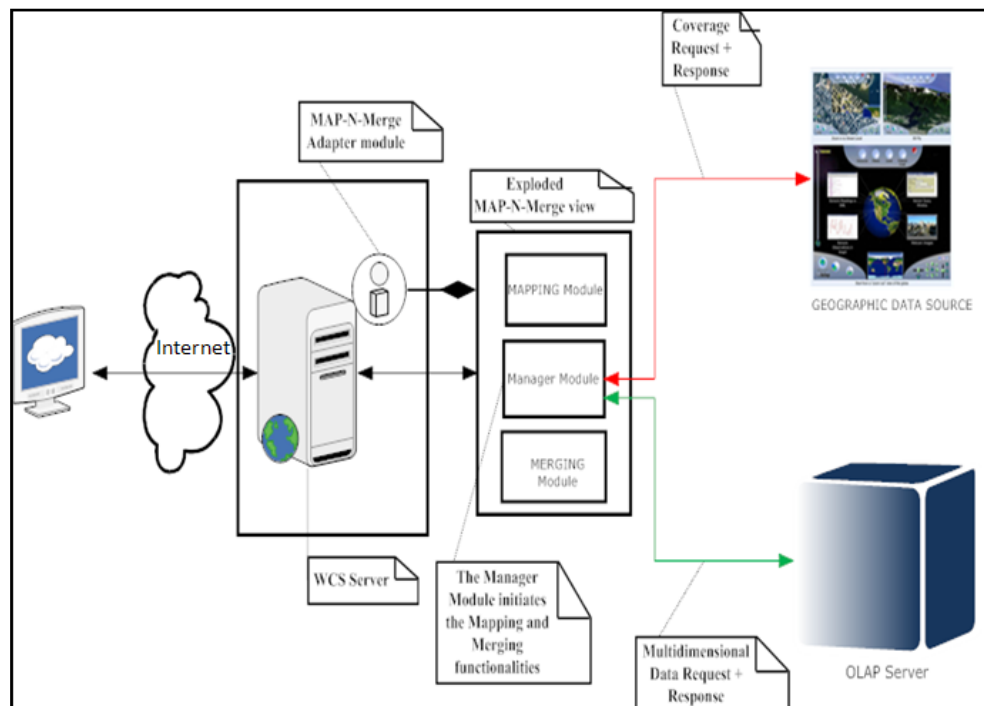


FIGURE 38: CONCEPTUAL MODEL FOR THE INTEGRATED SOLUTION

Figure 38 shows the overall architecture of the proposed solution. To process the request and responses generated the proposed solution defines an aggregator module (Chapter 5). The module is designed to be a part of the WCS implementation and will be incorporated in an existing WCS

²⁶ is a client-server architecture in which an application is executed by more than one distinct software agent

implementation²⁷, conceptualized as a data source handler. The aggregator solution is a two step approach. In the first step, the incoming WCS request is mapped to an OLAP request, DescribeCoverage to Discover and GetCoverage to Execute. The functionality is divided between the two domains with OLAP retrieving the multidimensional data while the WCS being responsible for spatial data retrieval. Once the data sets, representing the raw geographical data from the GIS data source and the multidimensional information from the OLAP data source, are retrieved, the aggregator proceeds to merge the two into an appropriate format to be passed on to the client.

²⁷ The existing WCS implementation decided upon is the deegree WCS. Deegree is a free Software project offering substantial building blocks for building a Spatial Data Infrastructure by implementing the standards of the Open GIS Consortium

4. RELATED WORK

This chapter discusses current and evolving solutions in the SOLAP domain. On the basis of the characteristics listed in Table 7 (see Section 3.4), the solutions will be summarily evaluated to offer some perspective of the niche the proposed solution aims to occupy.

4.1 INTRODUCTION

A substantial amount of research has been done in regards to the SOLAP domain. In spite of all this research activity, commercial solutions efficiently coupling OLAP and GIS appeared on the market only very recently. These solutions, some OLAP-centric, some GIS-centric, some hybrid, present only a subset of the desirable functionalities (see Section 3.2.2) of a Spatial OLAP technology. Some are still limited to static map visualization of OLAP query results. Others require the storage of each potential individual map view on the server thus, affecting the update effectiveness of spatial data. These solutions present limitations with regards to interactive data manipulation and exploration through cartographic views. However, the main bottleneck still remains the building of spatial data cubes, especially when data come from different sources. The substantial technical challenges in developing and deploying SOLAP solutions has been discussed in detail in the previous chapters. Solutions supporting some SOLAP characteristics (see Chapter 2) have appeared in the market recently, either from key proprietary companies such as ERSI²⁸, MapInfo²⁹, Oracle Corporation, and Kheops Technologies, or from innovative companies such as the Fraunhofer Institut for Intelligent Analysis Information Systems (IAIS)³⁰ with its software CommonGIS.

Figure 39 illustrates the SOLAP domain with categorization along the type of solution architecture. Examples of SOLAP desktop implementations are complimented with examples of server as well as SOLAP database solutions.

²⁸ world leader in GIS modeling and mapping software and technology

²⁹ GIS Software developer

³⁰ IAIS focuses on research and development on innovative systems for data analysis and information extraction, in software and in hardware.

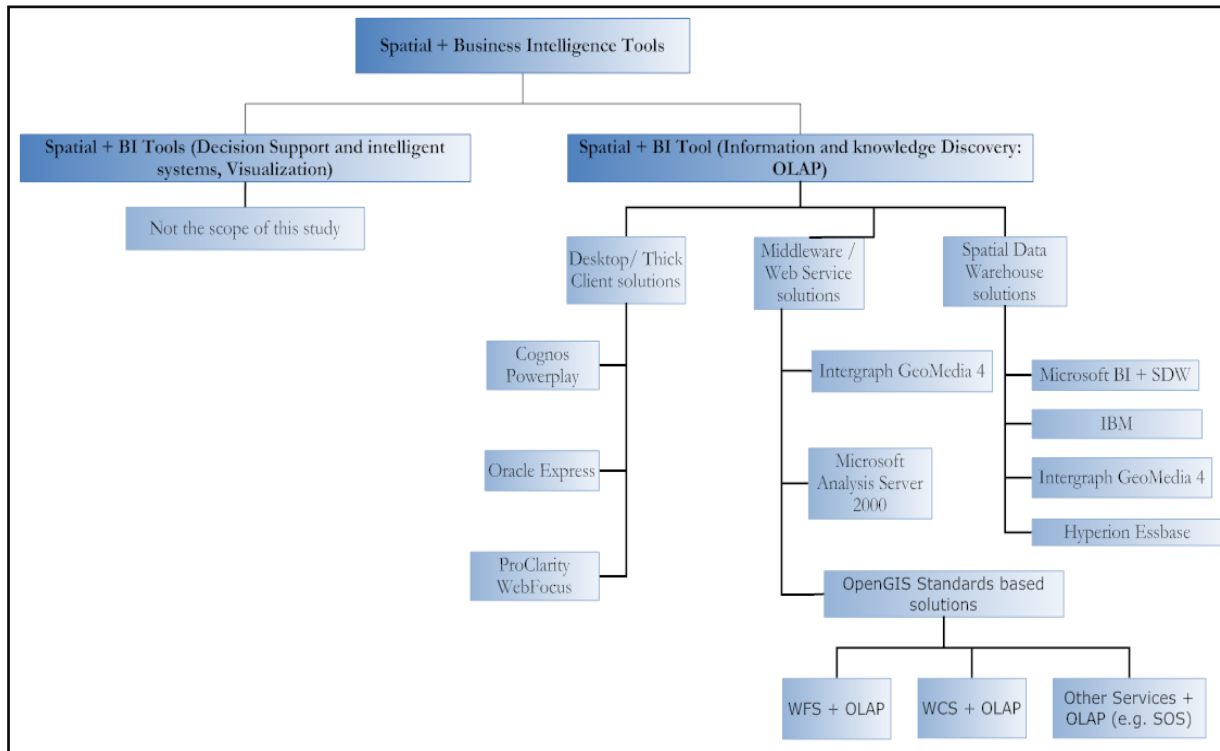


FIGURE 39: OVERVIEW OF THE SOLAP SOLUTIONS DOMAIN

4.2 SPATIAL OLAP SOLUTIONS – AN OVERVIEW

Several research projects aiming at combining analytical and spatial databases were carried out since the mid-nineties. In this section, the author reviews works that closely relate to the proposed solution.

The first SOLAP software commercially available, JMap Spatial OLAP, was developed at the Laval University under the guidance of (Bédard, et al., 2005). It was the first web-enabled technology to completely integrate geospatial dimensions within a business intelligence decision support environment. Examples of commercial SOLAP applications include the Geographic Business Warehouse (GBW) combining ESRI's GIS with SAP's business warehouse BW. It was designed for users with low or no GIS background and provides them with a very easy to learn interface and a sophisticated BI application associated to geographic software using MDX for the OLAP system. A whole array of companies ranging from Hyperion, with their multidimensional database Essbase, to IBM, offering an integrated SOLAP solution incorporating the Hyperion Essbase OLAP server and the IBM DB2 Spatial Extender, present Spatial OLAP functionalities.

Oracle offers not only a relational database system but also in its last version the Oracle OLAP package, which makes available several components for the creation of analytical systems including an ETL³¹ tool, DW³² and OLAP building functionalities, data mining functionalities, data mining services and tools for building charts and reports. Oracle DBMS supports the spatial extension named Oracle Spatial. The combination of the functionalities, Oracle OLAP and Oracle Spatial, offers a complete Spatial OLAP solution to Oracle users.

Another commercial software package that provides BI and GIS support is the Microsoft Business Intelligence and Data Warehousing tool. The SQL Server solution brings several components for the development of decision support environments including an ETL tool, a relational database, a component for analytical and data mining processing and tools for report management. With regards to the spatial processing, the SQL Server does not offer any native functionality. However, geographical processing can be made available by using, for example, the MapInfo Spatial Ware, which is based on SQL and the OGC specifications.

Many research projects have built bridges between OLAP and GIS to facilitate the development of hybrid systems similar to the most recent commercial releases, such as:

1. **GOAL** (Kouba, et al., 2000), a project that aims to accomplish the integration between DW, OLAP and GIS.
2. **SIGOLAP** (Ferreira, et al., 2001) provides the integration between GIS and OLAP by a three tier architecture, which is based on a mediation approach.
3. **CommonGIS**, where the authors (Hernández, et al., 2005) following an exploratory spatial data analysis approach present the software extended to process hierarchical multidimensional data from OLAP warehouses.
4. The authors (Bimonte, et al.) present the web based SOLAP prototype **GeWOlap**. GeWOlap is an extensible GIS-OLAP integrated solution supporting geographical dimensions and measures, and based on a formal multidimensional model named **GeoCube** (Bimonte, et al., 2006). GeWOlap is based on a three tier architecture consisting of Object-Relational DBMS (Oracle) supporting spatial data, an OLAP server (Mondrian), and a front-end client coupling an OLAP and a GIS web client.
5. The **GMLA Web Services** (Silva, et al., 2006) enable the integration of the analytic (*i.e.* XMLA) and geographic services (*i.e.* WFS). This integration provides users with the

³¹ Extraction, Transform, Loading

³² Data warehouse

capability of executing their geographic multidimensional requests thus enabling them to query and analyze geographic and/or multidimensional data stored in a Geographic Data Warehouse (GDW). The GMLA WS is both client and server. It can be seen as a client because it may access two other services (*i.e.* XMLA and WFS), and as a server, because a WSDL interface is provided to allow other applications to be developed. The solution introduces a new language for data exchange, the Geography Markup Language for Analysis (GMLA). The GMLA imports, extends, and integrates the original XML Schemas. With GMLA, data that had been previously described by two distinct schemas (GML and XMLA), can now be described and semantically integrated by just one schema. This resulting schema is then used by the GMLA Web Service to describe and validate the geographic-multidimensional data.

Figure 40 illustrates the GMLA WS prototype solution. The illustration is divided into two sections with the system architecture depicted on the left and the service example on the right. The system is based on three-tier architecture with the GMLA WS server controlling the flow of information from the client to the database. The example depicts a query requesting for food items in a small section of USA

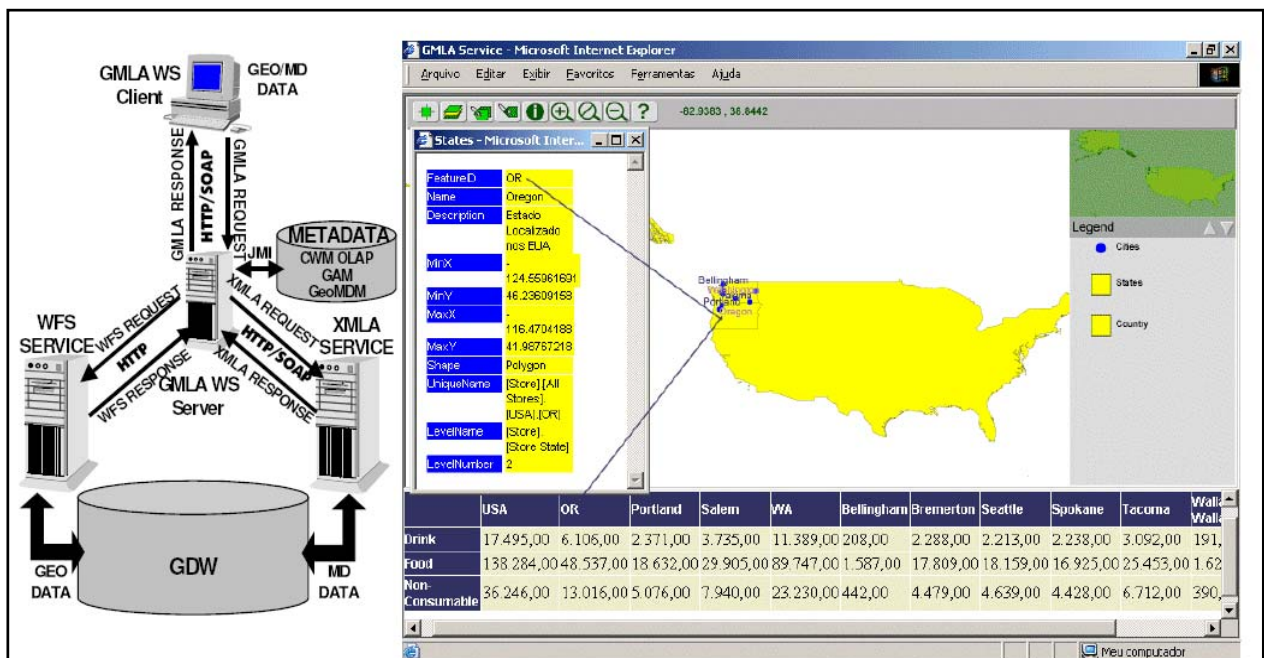


FIGURE 40: GMLA WS PROTOTYPE. AFTER: (SILVA, ET AL., 2006)

4.3 SUMMARY

With regard to the works discussed above, the author concludes that the solutions such as JMap, GBW, SIGOLAP, CommonGIS and GOAL, although offering an approach to integrate GIS and OLAP functionalities, rely on proprietary technologies thus preventing the solutions from being multi-platform based approaches and from providing extensibility and data interchange capabilities. The solution GeWOlap concentrates on introducing the geographical dimension into the cube model to overcome the challenges as mentioned by (Bimonte, et al.). The three tier architecture allows for interaction between different components but the absent OGC services restrict the extensibility of this solution. GMLA WS is based on open standards and supports interoperability. The hybrid integration allows both GIS and OLAP functionalities to be available to the user. The introduction of a GMLA client rounds up the solution providing visualization support for the spatial multidimensional data. Though apparently an appropriate solution for an integrated approach, the introduction of an exclusive architecture the Geographic On-line Analytical Processing Architecture (GOLAPA) and the conception of a new language GMLA dissuades the solution from being an universally acceptable one.

Table 8 evaluates the solutions on the basis of the characteristics identified by the author in Table 7 (see Section 3.4). With the help of the table the author proposes to identify the niche the prototype, presented in Chapter 5, will occupy.

CHARACTERISTICS	SOLAP SOLUTIONS					
	JMAP	SIGOLAP	COMMONGIS	GOAL	GEWOLAP	GMLA WS
INTEROPERABLE SOLUTION	No	Yes	No	No	No	Yes
BASED ON OGC (COMPLAINT) STANDARDS AND SPECIFICATIONS	No	No	No	No	No	Yes
COMPLIANT WITH OLAP STANDARDS AND SPECIFICATIONS	Yes	Yes	Yes	Yes	Yes	Yes
GIS ANALYSIS FUNCTIONALITIES (SPATIAL OPERATIONS)	Yes	Yes	Yes	No	Yes	Yes
OLAP ANALYSIS FUNCTIONALITIES	Yes	Yes	Yes	Yes	Yes	Yes
USED TECHNOLOGIES (VENDOR NEUTRAL)	Based on proprietary solutions	Based on proprietary solutions, restricted to the Windows OS	Based on proprietary solutions	Based on proprietary solutions, restricted to Windows OS	Based on proprietary solutions	Vendor and platform neutral
SCALABLE AND EXTENSIBLE	No	No	No	No	No	Yes
WEB BASED SOLUTION	Yes	Yes	No	No	Yes	Yes
QUERYING CAPABILITIES	Graphical User Interface with internal querying mechanism using MDX	Internal querying mechanism. MDX and Autodesk Mapguide querying functionalities	Internal querying mechanism. MDX and Java OLAP implementation	Internal querying mechanism. MDX and ArcView querying functionalities.	Graphical User Interface with internal querying mechanism using MDX	Introduction of a new querying language the GMLA

TABLE 8: RELATED WORKS SUMMARY

In conclusion the deficiencies of the solutions, although fulfilling the goal of integrating GIS and OLAP functionalities, showing a lack of standards, user requirement for technical expertise as well as a detailed knowledge of the underlying architecture, difficulty in interfacing to back-end external systems or real-time devices, and a lack of sophisticated GIS and analytical capabilities can be seen as a motivation for the work given in this study. Here a special mention of the solution GMLA WS is in order. GMLA WS is an amalgamation of the OGC Web Feature Service and an OLAP server. In this sense, the solution presented has almost achieved the objective of integrating OpenGIS and OLAP functionalities. The increased complexity introduced by the creation of a new query language, the GMLA, and the architecture frame, the GOLAPA architecture, is seen as a hindrance. Thus the author presents a new integration approach based on the OGC specified Web Coverage Service.

5. CASE STUDY: A STRATEGY FOR WEB COVERAGE SERVICE AND ONLINE ANALYTICAL PROCESSING INTEROPERABILITY

This chapter is divided into two parts. First, case study data in the form of the geographical Germany data and the Deutsche Presse Agentur data set with the extraction of relevant information is described. This is followed by the portrayal of the WCS and OLAP interoperable solution: a system prototype for spatial and multidimensional data exploration.

Throughout the implementation and testing phase the author used a 2.8 GHz Pentium 4 equipped with 1Gb RAM, 120Gb hard disk space, MS SQL Server 2000 and MS Analysis Server 2000. For the extension of the WCS the following products were used:

1. Eclipse SDK 3.3 (Software Development Kit)
2. Altova XMLSpy (XML Editor)
3. Apache Tomcat 5.5 (Web Server)
4. JDK 1.5 (Java Development Kit) and
5. deegree Web Coverage Service: official reference implementation of the OGC WCS 1.0.0

5.1 INTRODUCTION

The goal of this research is to introduce a solution allowing the exploration of multidimensional and spatio-temporal data based on an open and standardized GIS technology. With this goal in mind and with an aim of improving the process of decision making through a simpler and efficient querying interface, the author proposes a decisional information system by augmenting the OGC specified Web Coverage Service with an OLAP server.

The strategy developed in this section consists of the combination and extension of two pre-existent standardized works. The idea is to achieve interoperability by exchanging information, both spatio-temporal and multidimensional, via standard exchange formats e.g. XML. The open flexible framework and collaborative culture of open source project such as the OGC Web Coverage Service allows for rapid, collaborative development using the accumulated expertise in this domain thus reducing the need to develop new functionalities.

As already mentioned, the work has been divided into two phases. In the first phase, relevant data from the Deutsche Presse Agentur for the years between 2000 and 2005 (inclusive) were incorporated into the MS SQL Server data server and an OLAP cube was compiled. Section 0 deals with the data set in detail, and in conclusion the OLAP cube compiled is presented. In the second phase, the deegree Web Coverage Service, an OGC reference implementation, was extended to support MultiPoint and MultiSurface Coverages, as specified in GML 3.1, and in the final step the mediation between the two domains, OLAP and WCS, was implemented. Section 5.4 discusses the implementation in detail.

5.2 PLATFORM AND SOFTWARE REQUIREMENTS

A considerable amount of services and solutions are available in the market today for the two, OLAP and WCS, technologies. The following criteria to test the suitability of the framework with the following key-topics as a base were used to decide on the framework(s).

1. Implemented the **industry specifications** and **standards**? – Is it a **reference implementation**?
2. **Platforms** – on what platforms can the framework run?
3. **Data source connections** – what data sources may be used with the framework
4. Available **documentation** and instructions – how much information/instructions are available for the framework?
5. **Deployment and Configuration** – how easy/difficult is it to deploy/install and configure the framework
6. **Price Tag** – is the framework a commercial product or open source project?
7. **Development status** – is the framework being continuously developed or not?

Answering these questions will give a good basis to substantiate the framework chosen.

5.2.1 GIS DOMAIN

Since many frameworks implementing OGC specifications exist, a first selection was based on whether or not the framework was a reference OGC implementation³³. Setting this as the most

³³ A reference implementation is a fully functional implementation of a specification in reference to which other implementations can be evaluated. The OGC provides open source reference implementations to ensure maximum transparency of its specifications for both vendors and customers. OGC reference implementations are provided as-is,

important criterion for the product selected restricts the choice of frameworks. The OGC web service specifications are implemented and supported by many different organizations and products. As per the (OGC) Compliance and Interoperability Testing Initiative (CITE) the products deegree, GeoServer and GeoNetwork provide reference implementations for the OGC specifications. Of these three, deegree offers reference implementations for the Web Map Service 1.3.0, Web Map Service 1.1.1 and the Web Coverage Service 1.0.0. GeoServer and GeoNetwork³⁴ offer reference implementations for the Web Feature Service 1.1, Web Feature Service 1.0.0 and the Catalogue Services/Web 2.0.1 respectively.

In summary, Table 9 evaluates the open source solutions against the criteria stated above. Since the scope of the thesis is restricted to the WCS, this is the only framework which has been evaluated.

with no implied or explicit warranty from the OGC or implementation creators. The OGC does certify that these implementations are compliant with their respective specifications.

³⁴ GeoNetwork open source is a standard based Free and Open Source catalog application to manage spatially referenced resources through the web. It provides powerful metadata editing and search functions as well as an embedded interactive web map viewer.

SOLUTIONS	OGC REFERENCE IMPLEMENTATION	PLATFORMS	DATA SOURCE CONNECTIONS	DEPLOYMENT AND CONFIGURATION	DOCUMENTATION	DEVELOPMENT STATUS
DEEGREE	Yes (WCS 1.0.0)	All	Oracle, ESRI, PostGIS	Difficult	Insufficient	No Activity
GEO SERVER	Yes (WCS 1.1.0)	All	Oracle, ESRI, PostGIS	Easy	Substantial	Currently being extended to support Multidimensional Coverages

TABLE 9: GIS FRAMEWORK EVALUATION

For the WCS in the demo system, the choice stands between deegree and the GeoServer project. From the comparison, the conclusions are that both the frameworks should work fine, since they both implement the OGC specification and both can be connected to the commonly used databases. The deegree WCS is chosen over the GeoServer project since the author was looking for a stable base without a changing interface. The GeoServer project WCS implementation is in the beta phase of development and the author recommends the GeoServer project after the planned improvements have been integrated into the main project stream.

5.2.2 OLAP DOMAIN

In the mid 2000, the Open Source OLAP market began to establish itself, with several companies springing up with offerings. Although the open source BI market is burgeoning the products on offer often do not adhere to the industry norms or specifications making them unsuitable for the integration. This may be the direct reflection of not having a central organization body like the OGC coordinating and directing the development efforts of the community towards achieving a central goal. Of the open source products available only the Pentaho Analysis Services: Mondrian Project has managed to keep abreast with the developments in the OLAP domain and offer a comprehensive solution. Pentaho has established itself as a leader within the burgeoning open source Business Intelligence (BI) market with its Open BI Suite and Reporting Server products. Both offerings display an impressive range of choice BI functionality built on open source developed source code. BI applications such as Hyperion Essbase and the Microsoft Analysis Server have been the forerunners on the commercial front. The commercial systems have an edge over the open source community due to the time and money invested in the products. MSSQL Server 2000 Analysis Services has helped define industry standards for OLAP solutions including the XMLA for Analysis standard for data access. Microsoft was also involved in the introduction of the query language MDX and helping it achieve industry wide recognition.

The author restricts his research to the two solutions, MSSQL Server 2000 Analysis Service and Pentaho Mondrian, and in Table 10 evaluates the solutions against the criteria stated above. Since the scope of the thesis is restricted to the data retrieval, this is the only aspect of the solution which has been evaluated. Aspects such as reporting and visualizing have not been taken into consideration.

SOLUTIONS	INDUSTRY STANDARDS	PLATFORMS	DATA SOURCE CONNECTIONS	DOCUMENTATION	DEPLOYMENT AND CONFIGURATION	PRICE TAG	DEVELOPMENT STATUS
PENTAHO MONDRIAN	Yes	All	MSSQL, PostgreSQL, Oracle	Substantial	Difficult	Open Source	Is currently under development. Recent version release on August 31, 2007
ANALYSIS SERVICE 2000	Yes	Windows	MSSQL	Substantial	Easy	Commercial	Fully functional release

TABLE 10: OLAP SOLUTION EVALUATION

For the demo system the MSSQL Analysis Service 2000 has been selected as the working copy. This was influenced by the availability of a working installed service and because of the difficulties faced with the Pentaho Mondrian installation. Pentaho Mondrian now offers a new graphical cube designer tool but at the time of the implementation the tool was in a beta phase and hence the decision. For future work the author recommends the Pentaho Mondrian solution not least due to its comprehensive open source OLAP support backed with its functionality to be platform-independent and supporting the commonly used data sources.

5.3 THE DATA BASIS

This section gives some background information on the case study data set. First, the Deutsche Presse Agentur data is introduced, followed by a brief discussion of the spatial data model used.

5.3.1 DEUTSCHE PRESSE AGENTUR DATASET

The data used in this research is a subset of the data collected from the Deutsche Presse Agentur (DPA) Network. The DPA is one of the world's leading international news agencies supplying news on a global basis and entirely from its own sources. DPA is the only German news agency supplying a news service covering the complete spectrum of international and national affairs with full coverage of political, economic, scientific, cultural and general affairs and sports. The agency also has 12 German regional services. DPA provides a comprehensive coverage of news reports in both audio and visual format. Text news reports are augmented with photos, audio, graphics and multimedia images. The dpa-text services provide a constant stream of information and are used as a working basis by daily newspapers, radio and television stations as well as national news agencies.

The dataset used in this research has been provided by the Knowledge Discovery department at the IAIS Fraunhofer (Sankt Augustin) institute and covers the period from January 01, 2000 to December 31, 2005. The dataset is standardized according to the International and Press Telecommunications Council (IPTC³⁵) specification (IPTC/TEC 9701). The IPTC not only provides news exchange formats to the news industry but also creates and maintains sets of topics (NewsCodes) to be assigned as metadata values to news objects. The IPTC groups NewsCodes to help easily identify the specific purpose they are used for. The four main categories are:

1. **Descriptive NewsCodes** – This is a group of taxonomies to describe the content of news items properly
2. **Administrative NewsCodes** – This is a group of taxonomies for the administrative properties of news items.
3. **Transmission NewsCodes** – This is a group of taxonomies with controlled values for the transmission process, and

³⁵ Based in Windsor, United Kingdom, is a consortium of the world's major news agencies and news industry vendors. It develops and maintains technical standards for improved news exchange that are used by virtually every major news organization in the world

4. **Exchange Format NewsCodes** – This is a group of taxonomies with values to support specific functionalities of the different IPTC news exchange format standards.

For the scope of the current research the Descriptive NewsCodes are of importance. It is broken down into:

1. **Genre** – describing the nature, journalistic, or intellectual characteristic of a news object, not specifically its content
2. **Scene** – describes the scene of what is covered by the content
3. **Subject Code** – The subject code is a three level hierarchy for describing content by a well defined set of terms. Topics of level *Subject* provide a description of the editorial content of News at a high level; a *SubjectMatter* provides a description at a more precise level and finally a *SubjectDetail* at a rather specific level. Currently about 1300 terms are available – and several of them could be assigned to a single news object enabling a very narrow description of the content
4. **Subject Qualifiers** – provide a narrower attribute-like context for e.g. a sports-related subject code, like sex of acting people, indoor/outdoor sports etc.

Figure 41 illustrates the Subject Codes hierarchy. The first item in the list symbolizes the general subject name with each hierarchical step representing detailed information. For e.g., Crime, law and justice is the main subject name with police and later law enforcement representing the subject matter and subject detail respectively. The natural hierarchy and importance of this meta-information qualifier makes it an ideal candidate for a dimension in the planned cube structure. An example query formulated in this regard could be: How many reports with the subject name ‘Corporate Crime’ were reported in Bonn for the period between 01 July, 2002 and 01 July 2003.

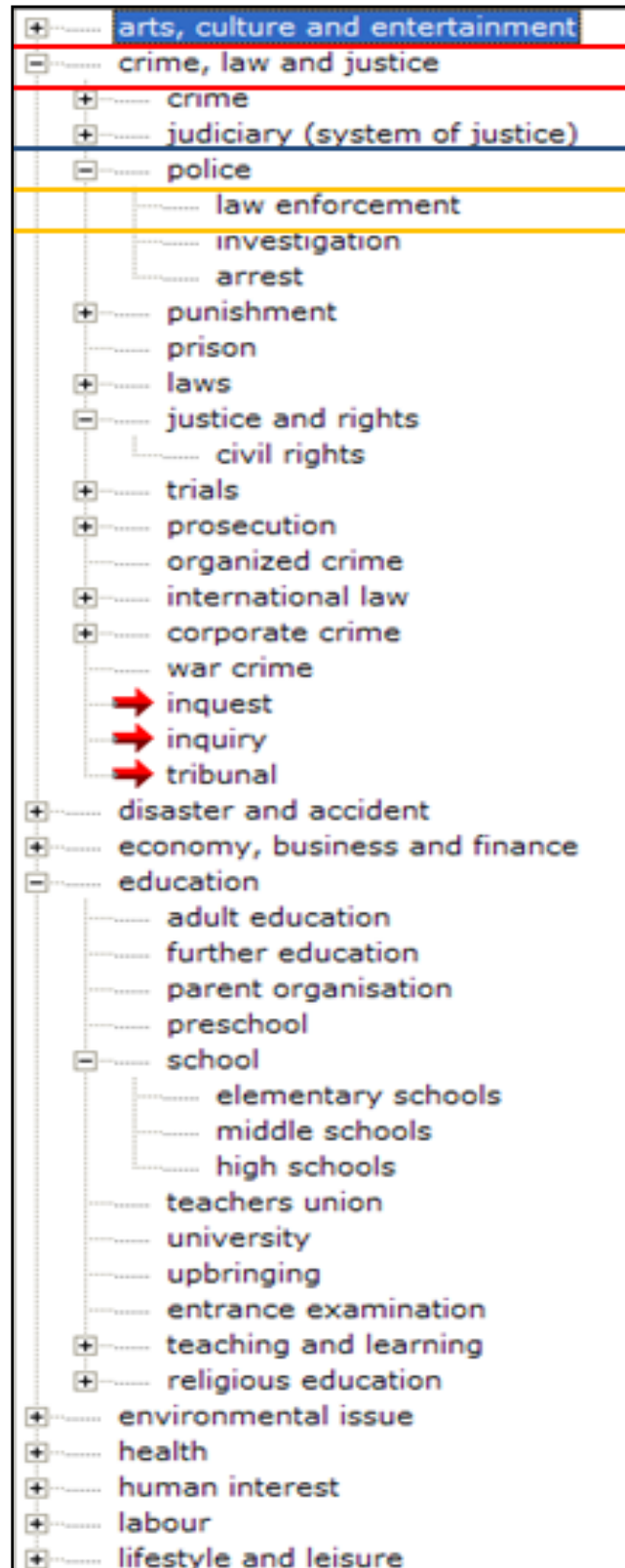


FIGURE 41: IPTC SUBJECT CODES. BASED ON: IPTC/TEC 9701

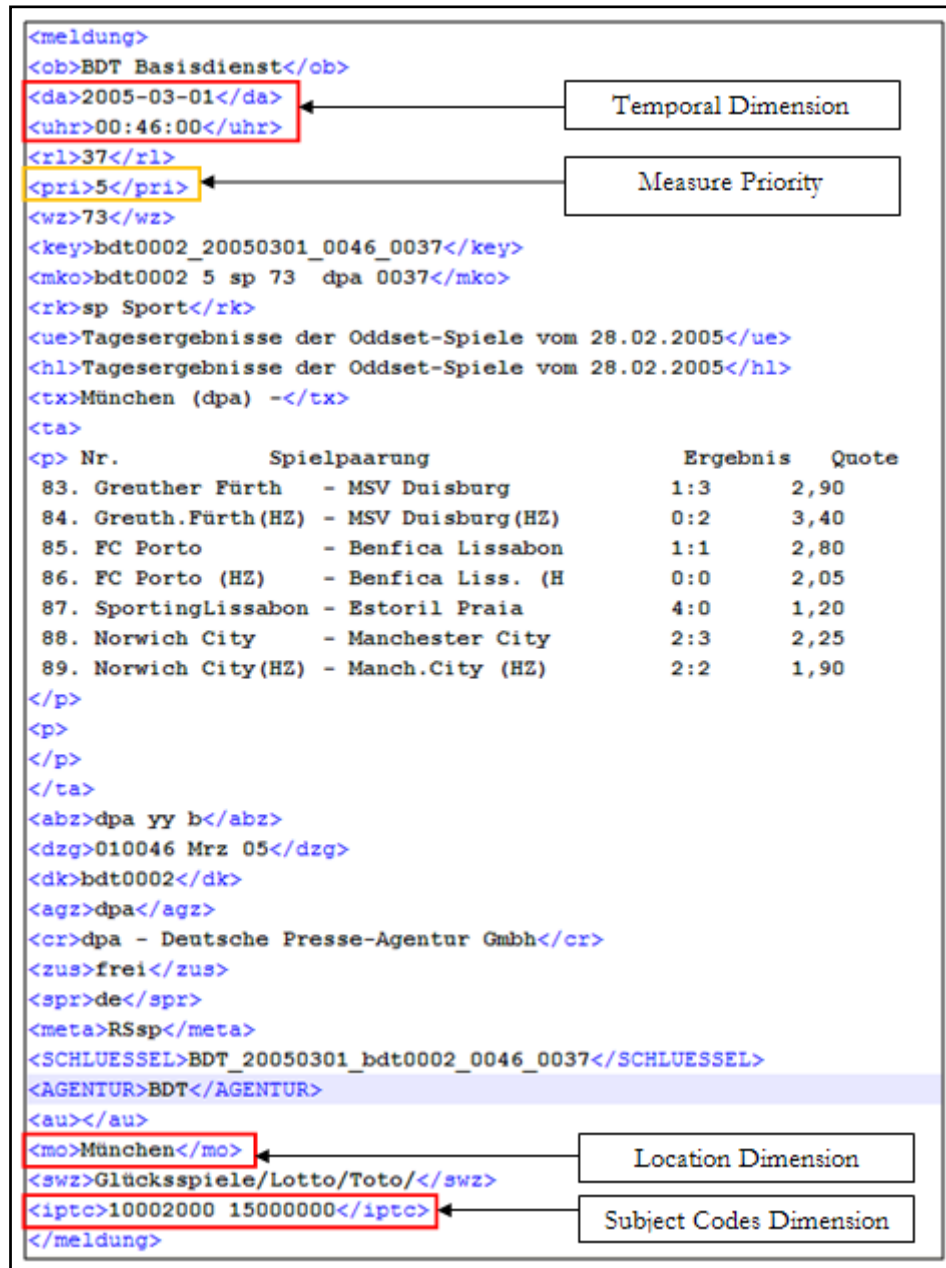


FIGURE 42: EXAMPLE OF A REPORT FROM THE DPA DATASET

Figure 42 illustrates an example report instance from the dataset. The individual nodes or fields of the report will not be explained in detail here. A cube consists of Measures, Dimensions and Levels. The research dataset puts forward minimum three dimensions. The Subject Code has already been established as a dimension. The spatial component of the news report is very important meta-information and is represented by the node `<mo>` or *Meldung (Report) Ort (Location)* characterizing the name of the cities for the current report. The data integration and interpretation³⁶ phase maps

³⁶ The data integration and interpretation phase is restricted to mapping only the German cities to their respective states. A generic worldwide mapping of the city to state mapping system has not been done.

the cities to the respective states. Another dimensional property the report model furnishes is the date and time of the report. The nodes <da> or *Date* and <uhr> or *Time* characterize this property. An OLAP Measure is defined precisely like an OLAP dimension, i.e., it is a concept hierarchy. The node <pri> or *Priority* identifies a Measure in this context. The Priority level symbolizes the urgency of the news report. Another interesting Measure is the number of reports of a particular IPTC subject. Though this Measure is not available in the dataset, OLAP offers the calculated Measure³⁷ functionality to identify important facts interesting for the end-user.

To summarize, the cube consists of the following dimensions with the hierarchy of each dimension established as Levels:

REPORT SUBJECT

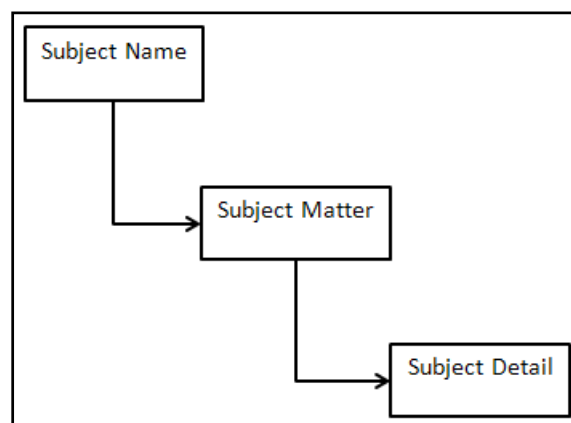


FIGURE 43: DIMENSION REPORT SUBJECT AND LEVELS

REPORT LOCATION

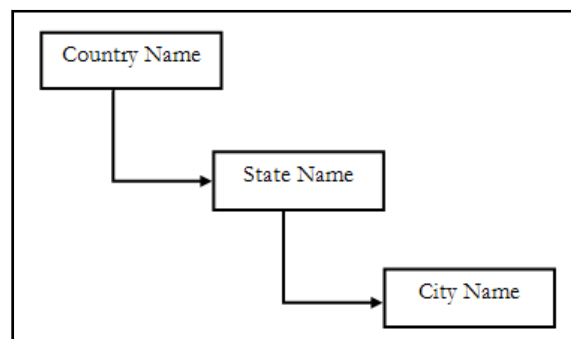


FIGURE 44: DIMENSION REPORT LOCATION AND LEVELS

³⁷ Calculated members are members of a dimension or a measure group that is defined based on a combination of cube data, arithmetic operators, numbers, and functions.

REPORT TIME

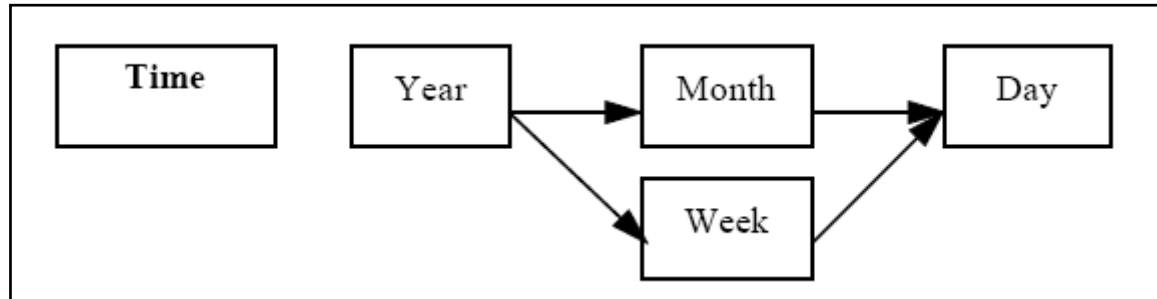


FIGURE 45: DIMENSION REPORT TIME AND LEVELS

MEASURE PRIORITY AND COUNT (NUMBER OF REPORTS OF A CERTAIN SUBJECT)

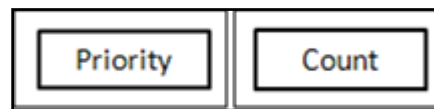


FIGURE 46: MEASURE(S) PRIORITY AND COUNT

In the Figure 47, IPTC Subject, Report Location and Time are non-geometrical dimensions. Report Cities and Report States contain geometrical information and are stored separately to be accessed over the Report Location table. The Measure Priority is stored in the fact table Report and the Measure Count is calculated on the basis of the IPTC Report Codes.

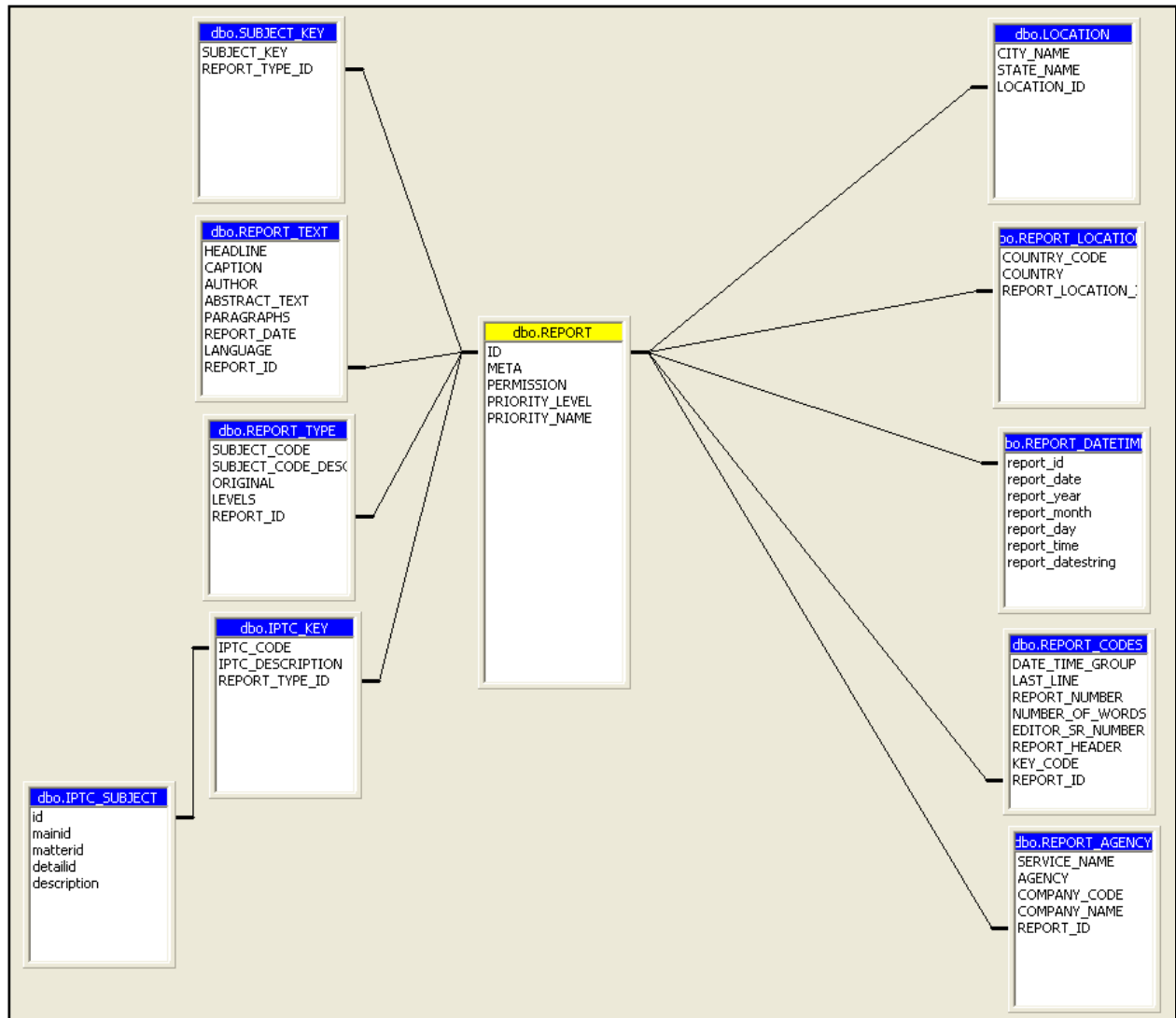


FIGURE 47: DPA OLAP CUBE MODEL IN STAR-SCHEMA

5.3.2 GERMAN CITIES AND STATE GEOGRAPHIC DATA

The degree reference implementation chosen as a base for this solution supports, as a WCS 1.0 reference implementation, only grid and image data types. However, the OGC Coverage Abstract Specification provides a broader definition of a coverage. *A COVERAGE CAN BE DESIGNED TO REPRESENT A SINGLE FEATURE, OR A SET OF FEATURES. FOR EXAMPLE, A COVERAGE MAY HAVE A SPATIAL DOMAIN CONTAINING A SINGLE PARCEL, OR AN ENTIRE PLATT. IN THE LATER CASE, THE COVERAGE MAY BE TREATED AS MODELING A SINGLE FEATURE (THE PLATT), OR AS A COLLECTION OF FEATURES (THE COLLECTION OF PARCELS).* Therefore a coverage can also include a collection of points or a collection of surfaces. The current WCS specification 1.1 specifies: *UNLIKE WFS[OGC 02-058], WHICH RETURNS DISCRETE GEOSPATIAL*

FEATURES, THE WEB COVERAGE SERVICE RETURNS COVERAGES REPRESENTING SPACE-VARYING PHENOMENA THAT RELATE TO A SPATIO-TEMPORAL DOMAIN TO A POSSIBLY (MULTIDIMENSIONAL) RANGE OF PROPERTIES, thereby acknowledging the support for the definition above. This justifies the decision to implement the MultiPoint coverage specification to serve the German cities geographical information using the WCS. Figure 48 illustrates the MultiPoint coverage XML schema as specified in GML 3.1.

```
<complexType name="MultiPointCoverageType">
  <complexContent>
    <restriction base="gml:AbstractCoverageType">
      <sequence>
        <group ref="gml:StandardObjectProperties"/>
        <element ref="gml:boundedBy"/>
        <element ref="gml:multiPointDomain"/>
        <element ref="gml:rangeSet"/>
        <element ref="gml:coverageFunction" minOccurs="0"/>
      </sequence>
    </restriction>
  </complexContent>
</complexType>
```

FIGURE 48: MULTIPOINT COVERAGE - XML SCHEMA. AFTER: (211/WG, 2004)

The MultiPoint coverage schema brings forward an interesting aspect. As per the specification the coverage function representing the link between the domain set and the range set need not be defined. This aspect allows the author the luxury of mapping each geometry element in the spatial domain to each element in the range domain, similar to a Feature.

In a MultiSurface coverage the domain is partitioned into a collection of surface elements comprising of GML specified MultiSurface geometries. The coverage function then maps each surface element in the collection to a value in the range set.

```

<complexType name="MultiSurfaceCoverageType">
  <complexContent>
    <restriction base="gml:AbstractCoverageType">
      <sequence>
        <group ref="gml:StandardObjectProperties"/>
        <element ref="gml:boundedBy"/>
        <element ref="gml:multiSurfaceDomain"/>
        <element ref="gml:rangeSet"/>
        <element ref="gml:coverageFunction" minOccurs="0"/>
      </sequence>
    </restriction>
  </complexContent>
</complexType>

```

FIGURE 49: MULTISURFACE COVERAGE - XML SCHEMA. AFTER:(211/WG, 2004)

The spatial data for the cities and states of Germany are available as ESRI shape files³⁸. The cities information is stored as a point dataset. Point datasets represent the most elementary and most common type of data. The city dataset contains information as regards to the name of the individual cities and their suburb as well as the ZIP/postal code information.

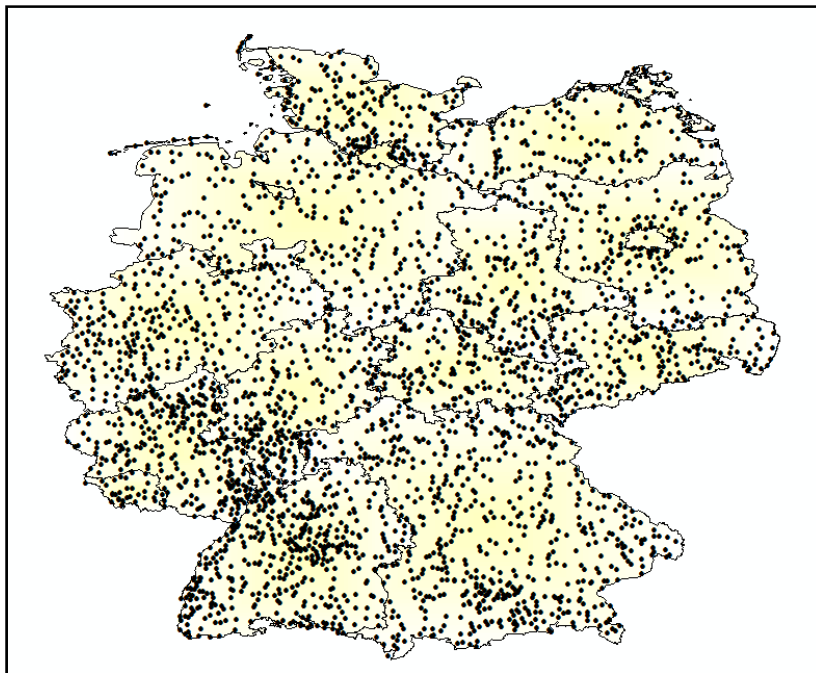


FIGURE 50: GERMANY CITIES DATASET REPRESENTING ONLY THOSE CITIES IN GERMANY THOSE ARE AVAILABLE IN THE DPA REPORT SET.

³⁸ The ESRI (the world leader in GIS modeling and mapping software and technology) Shapefile is a popular geospatial vector data format for GIS software. It is developed and regulated by ESRI as a (mostly) open specification for data interoperability among ESRI and other software products.

The Germany states ESRI shape file contains geometrical information of the individual states. Along with the name of the individual state the geographical information allows to identify the cities lying within the state boundaries thus propagating a two level dimensional hierarchy.



FIGURE 51: GERMANY STATES DATASET REPRESENTING ALL THE GERMAN STATES.

The polygon shapefile is to be represented using a MultiSurface coverage in the augmented degree WCS.

As already mentioned, the degree WCS interface was extended to support MultiPoint and MultiSurface coverages. Figure 52 gives an overview of the class diagram with the internal dependencies.

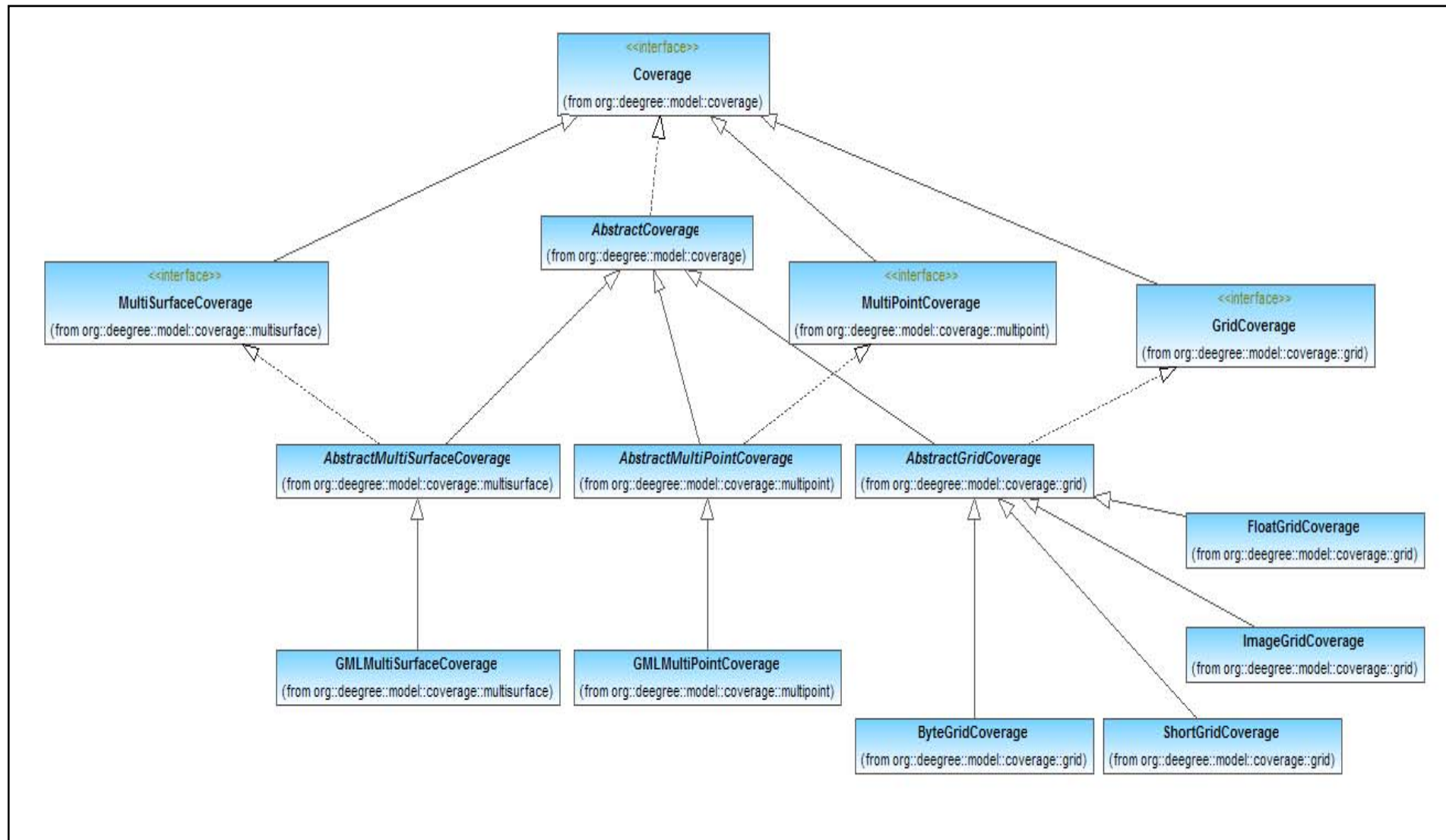


FIGURE 52: DEGREE COVERAGE CLASS DIAGRAM

The concept of a coverage reader and writer, already present in deegree, was then extended to support the multipoint and multisurface coverages.

5.4 MEDIATION PROCESS – CONCEPTUAL MAPPING

In order to demonstrate the concept an open and extensible solution relying on existing technologies to support integrated access of distributed heterogeneous (OLAP and GIS) data sources over the Internet is developed. This section first discusses the conceptual mediation process between the two domains and then on the basis of the prototype the author illustrates the proof of concept.

In the case of the integrated solution, the WCS provide the geographical data while the OLAP cube puts forward the attribute data. The interoperability and interaction between the two domains on the basis of a single querying mechanism can only be achieved by identifying the common elements between the two schemas and formulating a strategy to query the information from the domains. Chapter 3 provided an insight into the architecture and the working mechanisms of both an OLAP server and the WCS. In this section the author highlights the similarities between the two.

5.4.1 MAPPING A COVERAGE TO A CUBE

To facilitate the exchanging and interaction of the data it is important that the common elements of the two domains be identified. As already mentioned, coverage is a representation of a function between a domain subset, identifying the geographical scope, and a range subset, identifying the homogeneous attribute values. An OLAP solution provides attribute information pertaining to a certain subject. Hence we concentrate on the range set element of the WCS. Figure 53 illustrates the RangeSubset³⁹ element being made up of a series of FieldSubsets which in turn comprise of a series of AxisSubsets.

³⁹ According to Web Coverage Specification 1.1.0

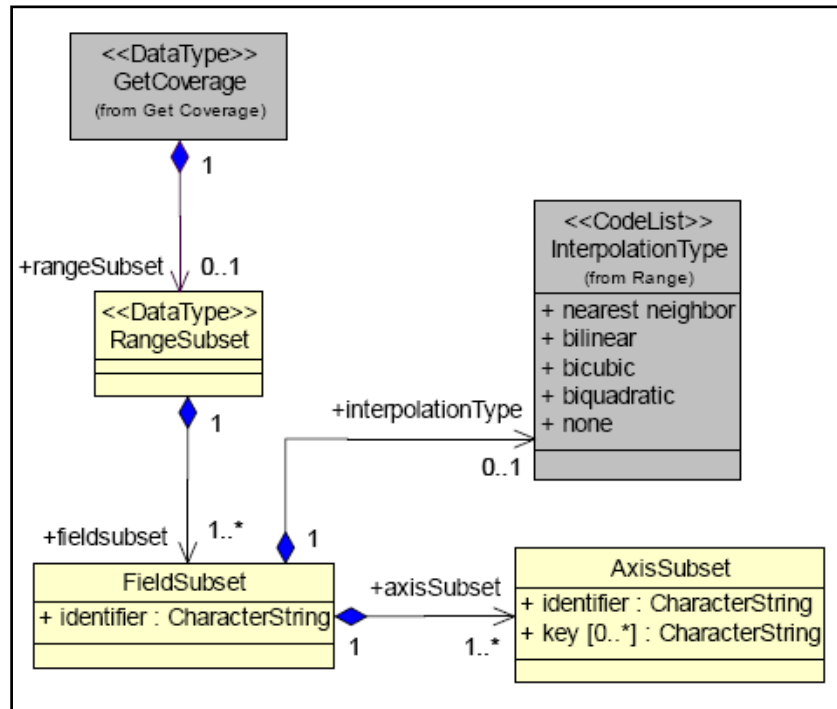


FIGURE 53: RANGESUBSET UML DIAGRAM. BASED ON: (WHITESIDE, ET AL., 2006)

The RangeSubset offers a three level hierarchy that is in sync with the natural OLAP cube hierarchy. The author identifies a RangeSubset to represent an OLAP cube structure. Each FieldSubset would subsequently be identified as a dimension in the cube. A FieldSubset is composed of a series of AxisSubsets. This representation is identified by the author to be similar to a cube dimension composed of a series of hierarchies. Thus the element AxisSubset represents a dimension hierarchy. As can be recognized an AxisSubset in turn is composed of a sequence of keys. This in turn is identified by the author to represent the levels of a cube hierarchy. To help visualize a parallel to the schema above, Figure 54 illustrates the OLAP cube structure with its dependencies as a UML diagram

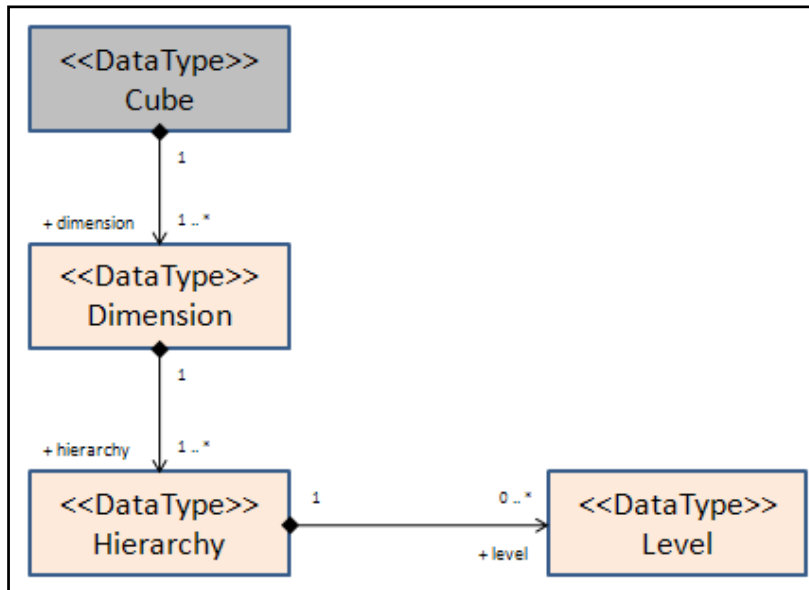


FIGURE 54: CUBE UML STRUCTURE DIAGRAM.

To summarize, the similarities in the hierarchy structure lead to the mapping between the two structural schemas illustrated in Figure 55.

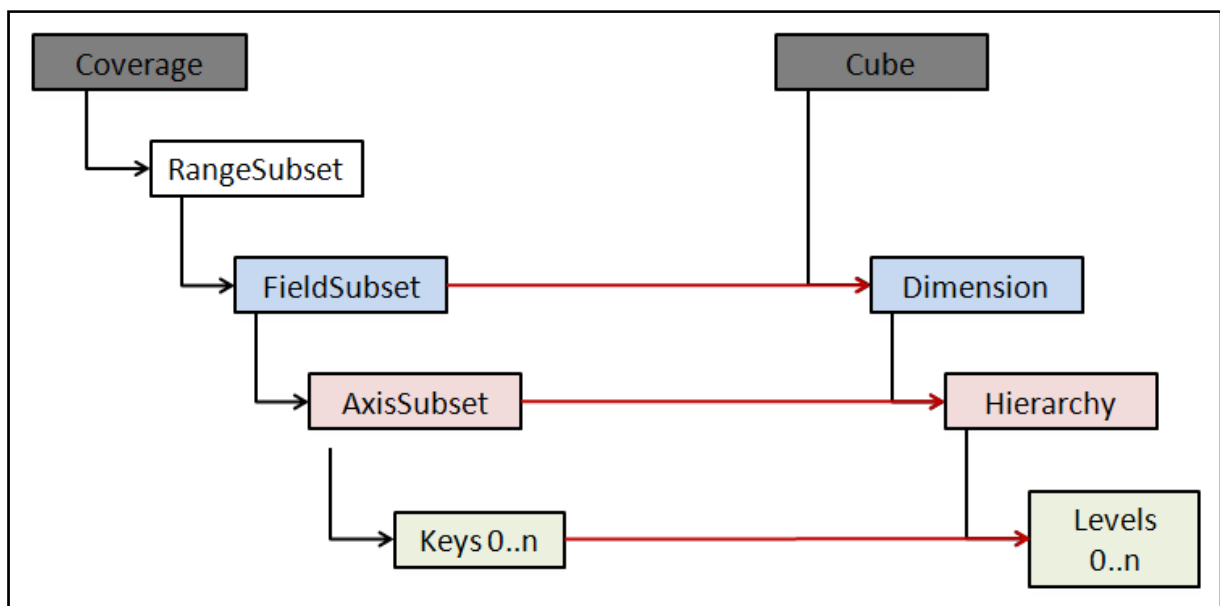


FIGURE 55: COVERAGE TO CUBE MAPPING

5.4.2 SERVICE CONFIGURATION

The demo service architecture allows access to both the geographical data made available by the WCS and the attribute information offered by the OLAP server. The following section introduces the deegree WCS system architecture and configuration followed by the OLAP analysis server architecture. Subsequently the proposed augmented WCS solution is presented.

Figure 56 shows the overall architecture of the modules that make up the deegree WCS. The HTTP interface is realized by a servlet registered with a servlet engine (e.g. Tomcat). The delegation of the incoming request is carried out over the dispatcher and the responsible service is called to life. Depending on the requested coverage type the WCS accesses the responsible data store to return the information to the user.

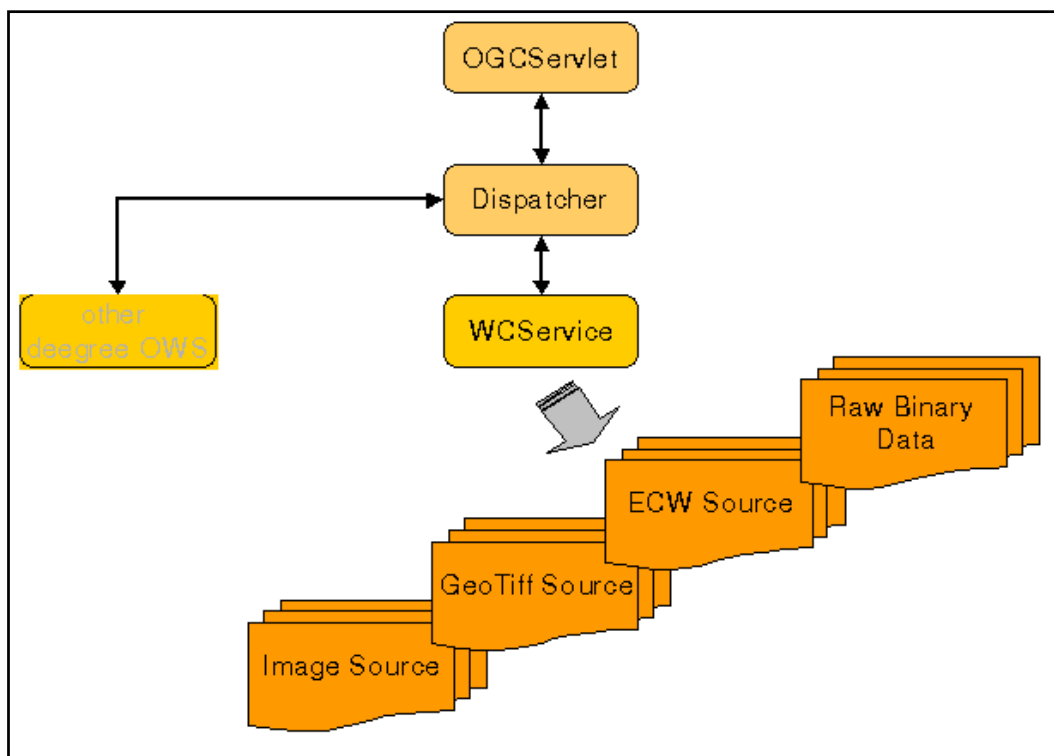


FIGURE 56: WCS SERVICE ARCHITECTURE AND MODULES. AFTER: (LAT-LON, 2006)

The deegree WCS is very flexible concerning the configuration possibilities, the adaption to different data sources, formats, layouts and server environments. This flexibility is realized by adapting the central configuration⁴⁰ document of the deegree WCS. The configuration is prepared using XML

⁴⁰ Based on the WCS capabilities document

files and is based on the OGC WCS 1.0.0 capabilities document and a few custom elements (see Figure 57).

```
<WCS Capabilities xmlns="http://www.opengis.net/wcs" xmlns:gml="http://www.opengis.net/gml" xmlns:xlink="http://www.w3.org/1999/xlink">
  <deegree:deegreeParam>
    <Service>
      <metadataLink xlink:type="simple" xlink:href="http://www.deegree.org" xlink:role="http://www.deegree.org" />
      <description>deegree WCS being OGC WCS 1.0.0 reference implementation</description>
      <name>deegree WCS</name>
      <label>deegree WCS</label>
      <keywords>
        <keyword>
          <name>deegree</name>
        </keyword>
      </keywords>
      <responsibleParty>
        <name>deegree</name>
      </responsibleParty>
      <fees codeSpace="http://www.deegree.org">NONE</fees>
      <accessConstraints codeSpace="http://www.deegree.org">NONE</accessConstraints>
      <accessConstraints codeSpace="http://www.deegree.org">SOME</accessConstraints>
    </Service>
    <Capability>
      <ContentMetadata xlink:type="simple" xlink:href="http://www.deegree.org" xlink:role="http://www.deegree.org" />
      <CoverageOfferingBrief gml:id="ID000003">
        <metadataLink xlink:type="simple" xlink:href="http://www.altova.com" xlink:role="http://www.altova.com" />
        <description>none</description>
        <name>GermanyCities</name>
        <label>Germany Cities</label>
        <lonLatEnvelope srsName="WGS84 (DD)">
          <gml:pos dimension="2">5.87439, 47.3102</gml:pos>
          <gml:pos dimension="2">15.02907, 55.01741</gml:pos>
        </lonLatEnvelope>
        <keywords>
          <keyword>
            <name>Germany</name>
          </keyword>
        </keywords>
        <deegree:Configuration>file:///C:/projects/deegree/resources/wcs/multipoint_coverage_description.xml</deegree:Configuration>
      </CoverageOfferingBrief>
      <CoverageOfferingBrief gml:id="ID000001">
        <metadataLink xlink:type="simple" xlink:href="http://www.altova.com" xlink:role="http://www.altova.com" />
        <description>none</description>
        <name>GermanyStates</name>
        <label>Germany States</label>
        <lonLatEnvelope srsName="WGS84 (DD)">
          <gml:pos dimension="2">5.87439, 47.3102</gml:pos>
          <gml:pos dimension="2">15.02907, 55.01741</gml:pos>
        </lonLatEnvelope>
        <keywords>
          <keyword>
            <name>Germany</name>
          </keyword>
        </keywords>
        <deegree:Configuration>file:///C:/projects/deegree/resources/wcs/multisurface_coverage_description.xml</deegree:Configuration>
      </CoverageOfferingBrief>
    </ContentMetadata>
  </WCS Capabilities>
```

FIGURE 57: DEEGREE WCS CONFIGURATION DOCUMENT

The configuration document offers the user complete control over the WCS behavior. This section introduces only the important aspect⁴¹ of the configuration file the `<ContentMetadata>` tag. This section contains basic information about the coverages offered by the WCS. This information is presented by defining a `<CoverageOfferingBrief>` tag for each coverage. A `<CoverageOfferingBrief>` tag contains basic information about the coverages, such as, name, label, bounding box and a link to the complete coverage description encapsulated in the `<deegree:Configuration>` tag. The coverage description (see OGC WCS 1.0.0 specification) is extended by an additional section `<olap:Extension>`. This extension serves as the data store

⁴¹ For a complete overview please refer to the WCS documentation at (www.deegree.org).

connection linking the coverage to an OLAP cube. Data store must be assigned to coverages in order for them to serve OLAP data. The data store connection mapping allows coverage to be mapped to a cube on the OLAP server.

```

<olap:Extension>
  <olap:Datastore>MSSQL-OLAP</olap:Datastore>
  <olap:OLAPConnection>
    <olap:ServerURL>http://msspade/xmla/msxisapi.dll</olap:ServerURL>
    <olap:RestrictionList>
      <olap:Cube>DPA</olap:Cube>
    </olap:RestrictionList>
    <olap:PropertyList>
      <olap:DataSourceInfo>Local Analysis Server</olap:DataSourceInfo>
      <olap:Catalog>DeutschePresseAgentur</olap:Catalog>
      <olap:Format>Tabular</olap:Format>
    </olap:PropertyList>
  </olap:OLAPConnection>
</olap:Extension>

```

FIGURE 58: COVERAGEDESCRIPTION OLAP EXTENSION

Figure 58 depicts the data store definition in a <CoverageDescription> configuration file. The parameters exemplify the XMLA properties parameters (see Chapter 3). Explanation of all the parameters:

1. **Datastore** – identifies the OLAP server backend
2. **ServerURL** – refers to the OLAP server web address.
3. **Cube** (see XMLA Property Cube) – identifies the cube the coverage is to be linked to
4. **DataSourceInfo** (see XMLA Property DataSourceInfo) – the name the Analysis Server is identified by
5. **Catalog** (see XMLA Property Catalog) – the name of the database the cube is stored under
6. **Format** (see XMLA Property Format) – the output format the server should communicate information in.

To summarize, the deegree WCS configuration has been augmented with a new Extension tag identifying the OLAP server backend and to map coverage to a cube. Figure 59 illustrates the 1:n relation between coverages and the cubes stored in the backend.

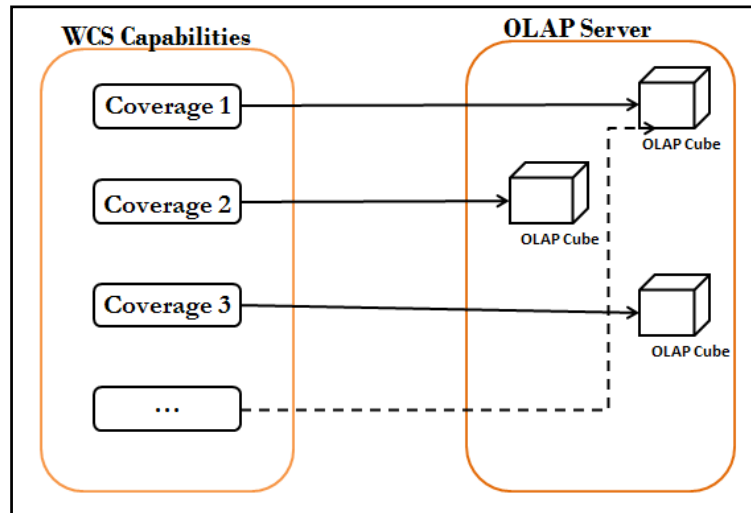


FIGURE 59: COVERAGE TO CUBE MAPPING

Data stores act as facades to concrete data sources to provide an easy and uniform access mechanism. Deegree widely supports the concepts of data stores in its architectural framework. This concept was extended to address a multidimensional database server. Figure 60 illustrates the deegree datastore class diagram.

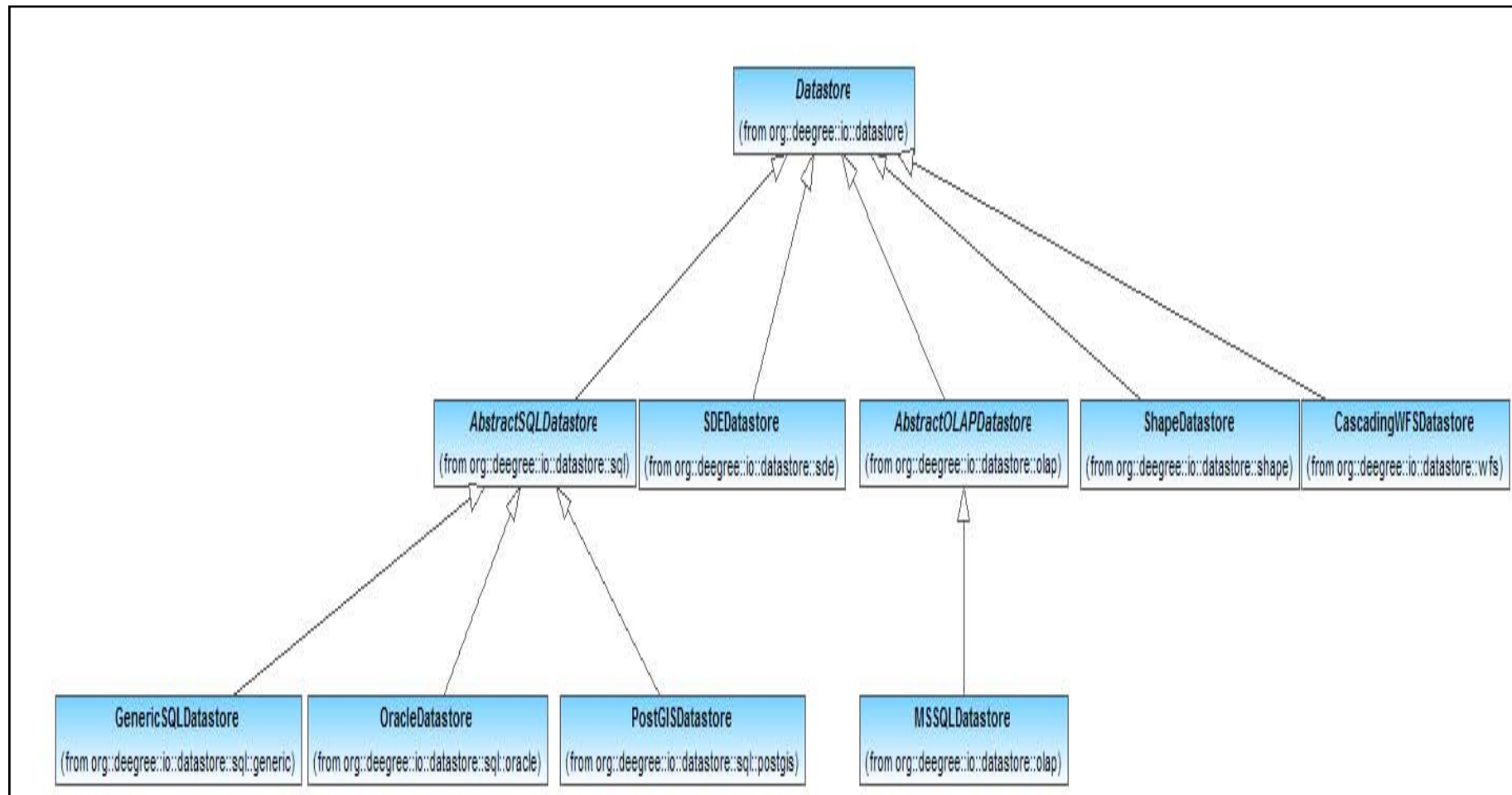


FIGURE 60: DEGREE DATASTORE CLASS DIAGRAM

Web services are more or less independent from data access. The basic idea behind a datastore concept is to have a unique façade for each data source format. This allows an application to easily switch between different data source formats. For example, currently the MSSQLDatastore has been implemented to access OLAP cube data. If the application were to be extended to support the Mondrian OLAP server, this could easily be realized by extending the abstract OLAP datastore and creating a new implementation, thus allowing for a clean and easy modular architecture structure.

The datastore provides an interface, a mechanism to explore the backend data source, in this case the MS SQL Analysis server. To facilitate the exploration of data the XMLA specification 1.1 was implemented (in part) by the author. As presented in Chapter 3, the XMLA specification offers two operations, Discover and Execute, to explore the multidimensional cube. These operations were implemented to formulate XMLA queries. Figure 61 illustrates the package structure for the OLAP querying mechanism.

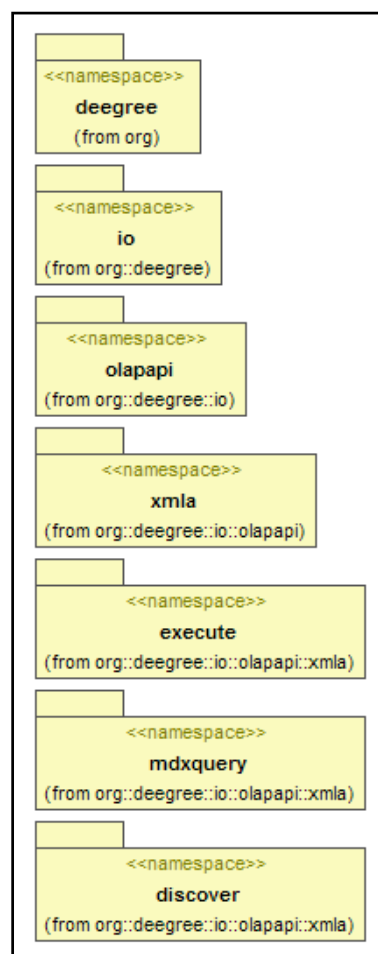


FIGURE 61: DEEGREE OLAP API PACKAGE DIAGRAM

Supplementing the Discover and Execute operations is the MDX query package. This package is responsible for building a valid MDX query from the incoming WCS request. The working of this is explained in the upcoming section.

5.4.3 MAPPING AND WRAPPING – GML TO XMLA

To ensure a working solution the mediation process must be conducted not only between the underlying data format of the two domains but also between the request schemas. Table 11 illustrates the mapping between the request schemas of the two domains.

REQUEST OPERATION	WEB COVERAGE SERVICE	XML FOR ANALYSIS (OLAP SERVER)
METADATA REQUEST	DescribeCoverage	Discover
DATA REQUEST	GetCoverage	Execute <ul style="list-style-type: none"> • MDX • mdXML
COMMUNICATION PLATFORM	HTTP – Get, Post	SOAP / HTTP
REQUEST FORMAT	XML	XML – SOAP
RESPONSE FORMAT	GML3 – XML W3C XML Schema	SOAP – XML W3C XML Schema

TABLE 11: 'MAPPING' WCS AND XMLA REQUEST SCHEMAS

The integration process will be demonstrated on the basis of the two mapped WCS operations: DescribeCoverage and GetCoverage. The goal of the augmented solution is to be a schematic OGC compliant solution and thus the OGC specified request and response documents schemas will not be modified. The author has interpreted⁴² the individual XML elements in both these documents to initiate the demo solution.

Figure 62 illustrates the degree request handling mechanism. The incoming WCS request is intercepted by the WCSService handler to be dispatched to the respective query handlers depending on the type of query received.

⁴² The interpretation of the XML elements in the documents may not be in sync with that of the original authors of the specification but is necessary to have a compliant solution.

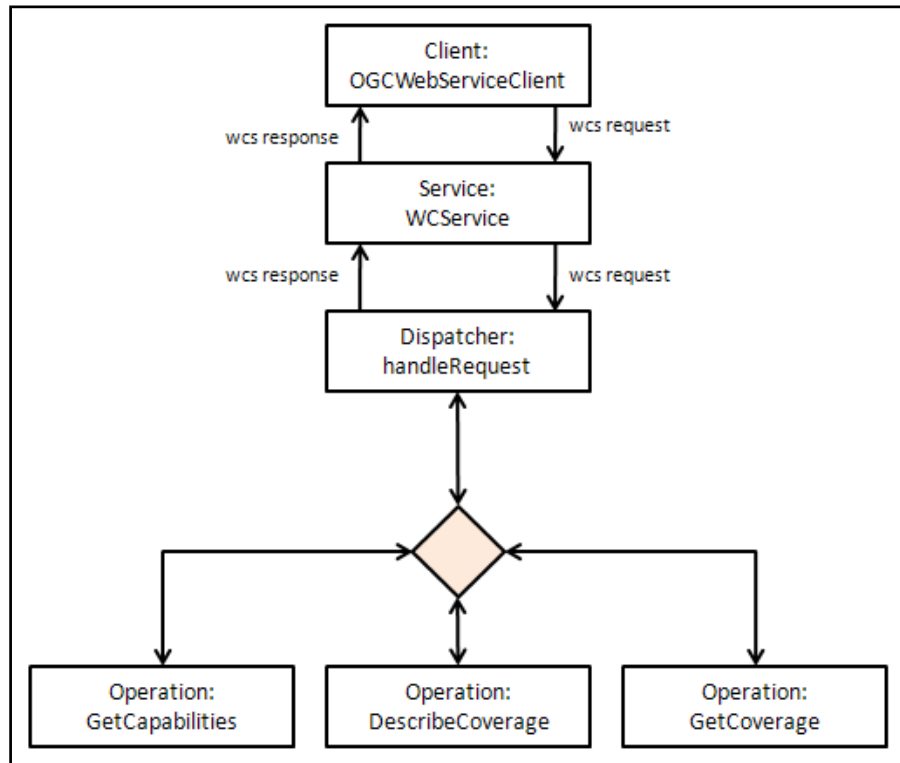


FIGURE 62: DEEGREE REQUEST HANDLING

The respective query handlers i.e. the DescribeCoverage query handler and the GetCoverage query handler are then responsible to first map the incoming request to the XMLA Discover and Execute requests and then wrap the responses to return a single coverage object back to the client.

GetCapabilities

The GetCapabilities operation returns an XML document describing the service and brief descriptions of the data collections from which the clients may request the coverages. The GetCapabilities request and response documents have not been modified in the augmented solution. The information returned gives a brief description of the coverage without the specific details regarding its composition. Clients can use the information returned by the WCS server to formulate simple GetCoverage requests, but in order to make more finely tuned GetCoverage requests, clients usually need to obtain further details using the DescribeCoverage operation.

DescribeCoverage

A DescribeCoverage operation returns an XML document describing one or more coverages served by the WCS. It provides information for a client to assess the data's fitness for use, and to formulate

fine-grained GetCoverage requests. The parameters required for a valid DescribeCoverage request are listed below in Figure 63.

Names ^a	Definition	Data type and value	Multiplicity and use
service service	Service type identifier	Character String type, not empty Value is OWS type abbreviation, namely "WCS"	One (mandatory)
request request	Operation name	Character String type, not empty Value is operation name, namely "DescribeCoverage"	One (mandatory)
version version	Specification version for operation	Character String type, not empty Value is specified by each Implementation Specification and Schemas version	One (mandatory)
identifier Identifier	Unambiguous identifier of a coverage, unique for this WCS	Character String type, not empty ^b	One or more (mandatory) Include for each desired coverage
<p>a See Table 1 for UML and XML naming conventions.</p> <p>b Values are identifiers in the Contents section of the GetCapabilities operation response. When the Capabilities XML document does not have a Contents section, clients must obtain a valid coverage identifier from another source, such as from a catalog service. See Subclause 8.3.4.1.</p>			

FIGURE 63: DESCRIBECOVERAGE REQUEST PARAMETERS. AFTER: (WHITESIDE, ET AL., 2006)

The response to a valid DescribeCoverage operation describes each coverage requested specified by the identifier in the DescribeCoverage request. Figure 64 illustrates the valid DescribeCoverage operation response UML class diagram.

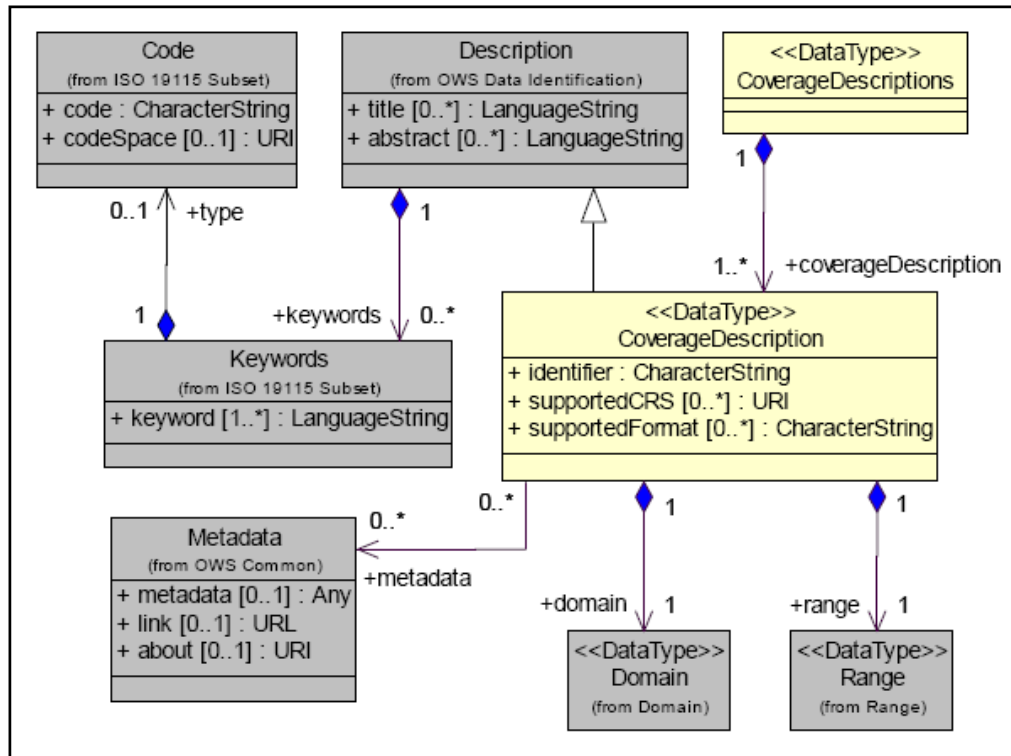


FIGURE 64: DESCRIBECOVERAGE RESPONSE UML CLASS DIAGRAM. AFTER: (WHITESIDE, ET AL., 2006)

As mentioned above the incoming DescribeCoverage operation is mapped to the XMLA Discover request. Figure 65 helps illustrate the request mapping concept the author proposes.

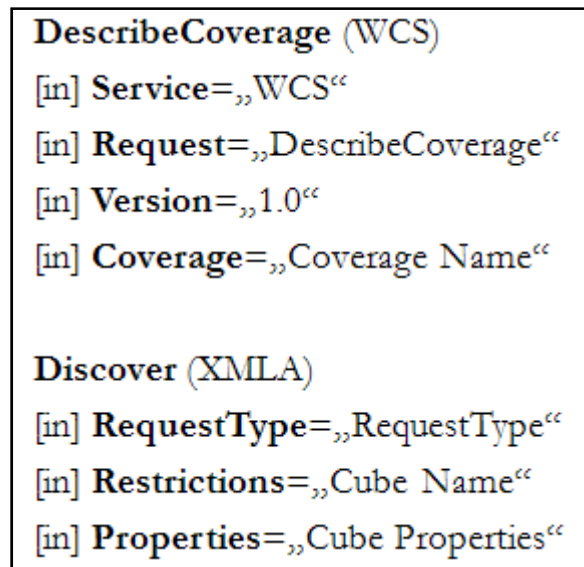


FIGURE 65: MAPPING EXAMPLE - DESCRIBECOVERAGE TO DISCOVER

The XMLA Discover request offers the functionality to retrieve the metadata for a wide palette of schema rowsets (see Chapter 3). From this wide range of operations the operations retrieving

indispensible information have been mapped to the DescribeCoverage operation (see Figure 66). This is necessitated by the fact that there does not exist one operation that would retrieve all the required information pertaining to the dimensions, hierarchies, levels and measures (with all the required meta-information). The sum of the response of these operations is needed to build the association to the coverage.

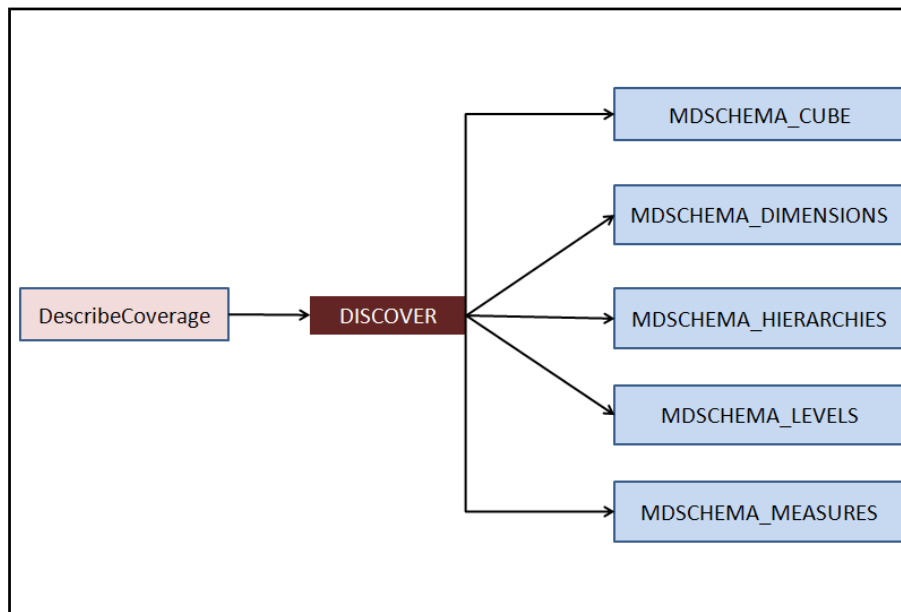


FIGURE 66: DESCRIBE COVERAGE TO DISCOVER MAPPING

Figure 67 illustrates a typical DescribeCoverage request.

```

<xml version="1.0" encoding="UTF-8">
<DescribeCoverage xmlns="http://www.opengis.net/wcs/1.1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wcs/1.1 http://schemas.opengis.net/wcs/1.1.0/DescribeCoverage.xsd"
service="WCS" version="1.1.0">
  <Identifier>GermanyPointData</Identifier>
</DescribeCoverage>
  
```

FIGURE 67: DESCRIBE COVERAGE XML REQUEST EXAMPLE

The DescribeCoverage handler carries out these steps to ensure an integrated response:

1. The incoming request is validated against the WCS configuration document.
2. After having successfully validated the query the handler retrieves the coverage offerings of those coverages requested for in the query.

3. The coverage offerings are retrieved on hand the information⁴³ provided in the WCS configuration file.
4. After having successfully retrieved the coverage offerings the DescribeCoverage handler scrutinizes the coverages to check if they are mapped to an OLAP cube.
5. If the coverages are mapped, the Discover cube Meta requests are created.
6. A specially created OLAP Datastore⁴⁴ is initialized using the parameters defined in the deegree WCS configuration file to help carry out the queries.
7. The coverage offerings are used to wrap the successful returned XMLA response document(s) to create a valid OGC specified response document.
8. The DescribeCoverage handler returns the response document back to the service dispatcher to return to the client.

Figure 68 illustrates a partial DescribeCoverage response document augmented with information from an OLAP cube mapped to a coverage.

⁴³ See deegree WCS documentation

⁴⁴ A deegree datastore is responsible for accessing the underlying database to retrieve the information. Refer to the deegree documentation.

```

<CoverageDescription version="1.0.0" xmlns="http://www.opengis.net/wcs" xmlns:deegree=
  <CoverageOffering>
    <name>GermanyCityPoint</name>
    <label>Germany Cities</label>
    <lonLatEnvelope srsName="WGS84 (DD)">
      <gml:pos dimension="2">5.87439 47.3102</gml:pos>
      <gml:pos dimension="2">15.02907 55.01741</gml:pos>
    </lonLatEnvelope>
    <keywords>[]
    <domainSet>[]
    <rangeSet>
      <RangeSet>
        <name>DeutschePresseAgentur</name>
        <label>DeutschePresseAgentur</label>
        <Field>
          <identifier>[REPORT SUBJECT]</identifier>
          <Axis>
            <identifier>[REPORT CATEGORY]</identifier>
            <AvailableKeys>
              <Key>(Gesamt)</Key>
              <Key>MainId</Key>
              <Key>MatterId</Key>
              <Key>DetailId</Key>
            </AvailableKeys>
          </Axis>
        </Field>
        <Field>
          <identifier>[REPORT LOCATION]</identifier>
          <Axis>
            <identifier>[REPORT LOCATION]</identifier>
            <AvailableKeys>
              <Key>(Gesamt)</Key>
              <Key>State Name</Key>
              <Key>City Name</Key>
            </AvailableKeys>
          </Axis>
        </Field>
      </RangeSet>
    </CoverageOffering>
  </CoverageDescription>

```

FIGURE 68: DESCRIBE COVERAGE RESPONSE (PARTIAL) DOCUMENT

In conclusion, the mapping between the DescribeCoverage and Discover operations is successful. The response document not only provides geographical domain information but also the meta-information describing the cube associated with the coverage. The document is a valid OGC DescribeCoverage response document and hence the goal of maintaining the OGC compatibility has been achieved.

GetCoverage

The GetCoverage operation allows the retrieval of subsets of coverages. A WCS server processes a GetCoverage operation request and returns a response to the client that either contains or references the requested coverage(s).

Figure 69 presents an overview⁴⁵ of the GetCoverage request parameters to identify the coverage subset to be retrieved.

Names ^a	Definition	Data type and value	Multiplicity and use
service service	Service type identifier	CharacterString type, not empty Value is OWS type abbreviation, namely "WCS"	One (mandatory)
request request	Operation name	CharacterString type, not empty Value is operation name, namely GetCoverage	One (mandatory)
version version	Specification version for operation	CharacterString type, not empty Value is specified by each Implementation Specification and Schemas version	One (mandatory)
identifier Identifier	Unambiguous identifier of coverage to draw from	CharacterString type, not empty ^b	One (mandatory)
domainSubset DomainSubset	Defines desired subset of domain set of identified coverage	DomainSubset data structure, see Table 24	One (mandatory)
rangeSubset RangeSubset	Defines desired subset of range(s) of identified coverage	RangeSubset data structure, see Table 27	Zero or one (optional) ^c
output Output	Specifies desired output options	Output data structure, see Table 25	One (mandatory)
<p>^a See Table 1 for UML and XML naming conventions.</p> <p>^b Values are identifiers in Contents section of GetCapabilities operation response. When this information is not in the Contents section of the Capabilities XML document, clients shall obtain a valid coverage identifier from another source, such as a catalog service.</p> <p>^c Providing a RangeSubset specifies a selection of range components and/or selects spatial interpolation methods for selected range components. Omitting a RangeSubset shall result in all range fields being delivered in their entirety, using the default spatial interpolation methods specified for each field.</p>			

FIGURE 69: GETCOVERAGE REQUEST PARAMETERS. AFTER: (WHITESIDE, ET AL., 2006)

The incoming GetCoverage is mapped to the XMLA Execute request. Figure 70 helps illustrate the request mapping concept the author proposes.

⁴⁵ For detailed information refer to the OGC Web Coverage Specification 1.1.0

```
GetCoverage(WCS)  
[in] Service=„WCS“  
[in] Request=„GetCoverage“  
[in] Version=„1.0“  
[in] Coverage=„Coverage Name“  
[in] DomainSubset=„Domain Information“  
[in] RangeSubset=„Range Information“  
[in] Output=„Output Format“  
  
Execute(XMLA)  
[in] Command=„MDX dimension Information“  
[in] Properties=„Cube Properties“
```

FIGURE 70: MAPPING GETCOVERAGE TO EXECUTE

A successful interoperable solution allows an abstraction of the functionalities between the solutions working together. In this case the WCS assumes responsibility to handle the subsetting of the coverage based on the domain parameters in the incoming request. The responsibility to subset based on the range parameters is transferred to the OLAP server mapped to the Execute request. The WCS range parameters requested for are mapped, based on the aforementioned schema, to the cube schema, and with the help of the MDX specification a query is formulated⁴⁶ to retrieve the requested information in the format specified.

To have a better understanding of the working of the integrated solution the workings will be explained on the basis of an example GetCoverage request. Figure 71 illustrates a GetCoverage request example

⁴⁶ The prototype solution is responsible for building the MDX queries from the GetCoverage request parameters

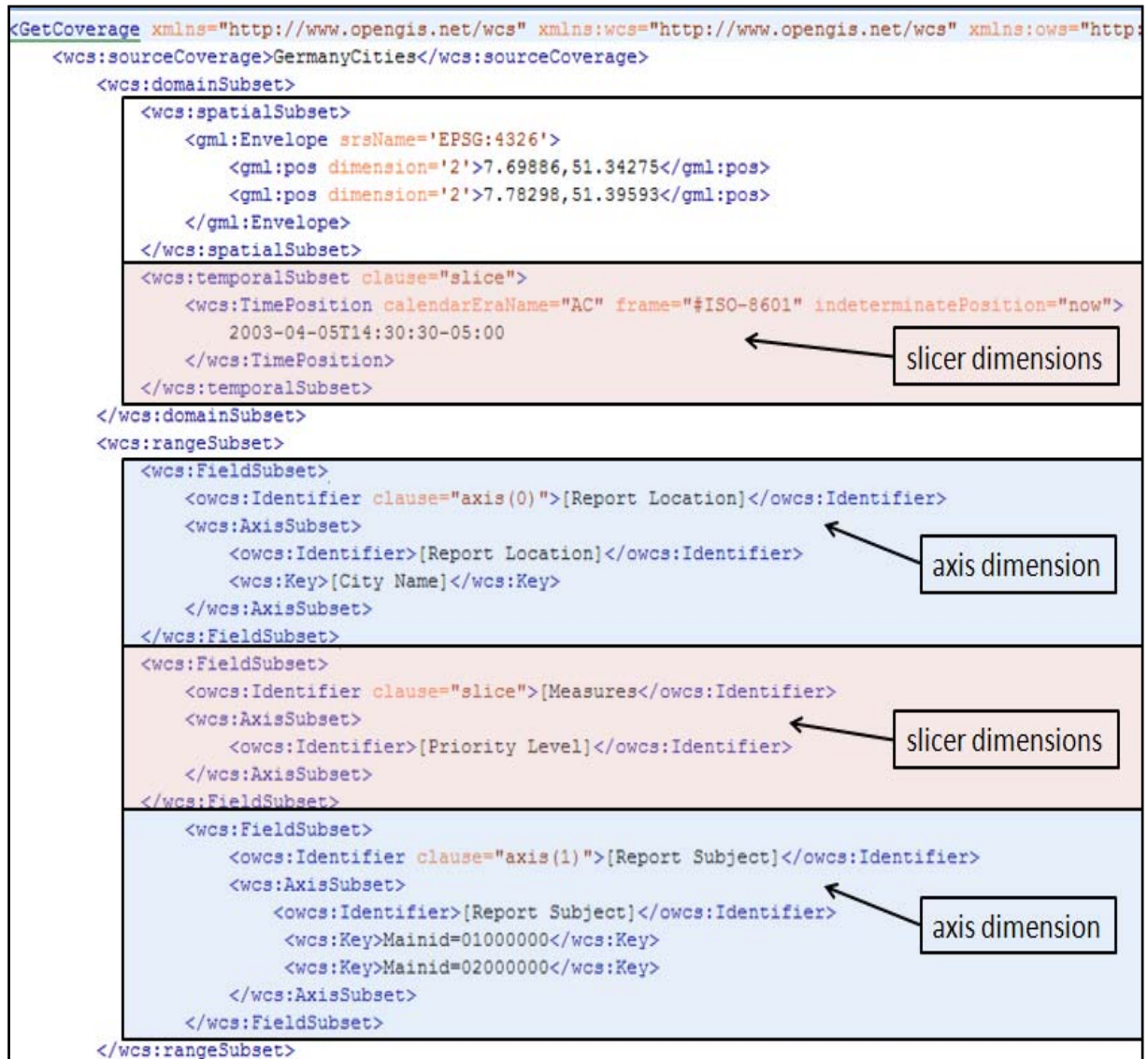


FIGURE 71: GETCoverage (PARTIAL) REQUEST

The following steps illustrate the mediation process:

1. The spatial subset element of the GetCoverage request has two roles to play.
 - a. First, the information is used to retrieve a geographical subset of the GML coverage.
 - b. Second, a GML coverage subset contains a list of discrete features. This list will in turn be used in the Execute request – mapped to the geographical dimension in the cube.
2. Before the mapping is carried out, it is imperative to identify the dimensions requested for as slicer or as axis dimensions. This identification helps build the MDX query to retrieve the attribute information. To serve this purpose the GetCoverage request schema was extended. The temporal subset element and the Field Subset Identifier element were extended with an

optional **clause** attribute. This attribute specifies whether the dimension is to represent a slicer or an axis dimension. If the clause specifies it to be an axis dimension, it is mandatory to specify the axis number⁴⁷ the dimension should be represented on.

3. The temporal subset defined in the request above is identified as a slicer dimension. The temporal subset specified is a time position. This information will be used in the **WHERE** clause of the MDX request to restrict the selection on the axes.
4. The range subset is composed of three field subsets, two of which are axis dimension requests and one a slicer dimension. The axes definitions include the Report Location, specifying the geographical dimension of the cube, and the Report Subject, identifying the report categories. The slicer dimension specifies a Measure dimension. As per the MDX specification the Measure dimension in the where clause specifies that the query shall return the values for the measure specified and not the default measure of the cube.
5. The Report Location axis clause specifies the level name, *Key – City Name*, the current query should explore. To ensure only the members visible in the requested geographical context are returned the author uses the list of discrete features returned by the spatial subset request to identify them.
6. The Report Subset axis clause identifies the level *Mainid* and the two specific members the query should be restricted on.
7. Figure 72 illustrates the corresponding XMLA Execute operation request. The properties list specifying the data connection to be established and the MDX command statement formulating the requested information.

⁴⁷ Refer to the MDX specification.

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <SOAP-ENV:Body>
    <Execute xmlns="urn:schemas-microsoft-com:xml-analysis">
      <Command>
        <Statement>
          SELECT { [Report Location].[City Name].[Bonn], [Report Location].[City Name].[Köln], ... } ON AXIS(0)
              { [Report Subject].[Mainid].[01000000], [Report Subject].[Mainid].[02000000] } ON AXIS(2)
          FROM DPA
          WHERE { [Report Time].[2003].[April].[05], [Measures].[Priority Level].[Members] }
        </Statement>
      </Command>
      <Properties>
        <PropertyList>
          <DataSourceInfo>Microsoft Analysis Server</DataSourceInfo>
          <Catalog>DeutschePresseAgentur</Catalog>
          <Format>Multidimensional</Format>
        </PropertyList>
      </Properties>
    </Execute>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

FIGURE 72: MAPPED EXECUTE REQUEST

Figure 73 illustrates the spatial domain of the valid coverage slice retrieved. The figure illustrates the domain represented by a series of discrete points.

```

<GermanyCities xmlns:gml="http://www.opengis.net/gml">
  <gml:domainSet>
    <gml:MultiPoint>
      <gml:pointMembers>
        <gml:Point>
          <gml:pos>7.7119 51.3858</gml:pos>
        </gml:Point>
        <gml:Point>
          <gml:pos>7.76482 51.3803</gml:pos>
        </gml:Point>
      </gml:pointMembers>
    </gml:MultiPoint>
  </gml:domainSet>
  <gml:rangeSet>
    <gml:DataBlock>
      <gml:rangeParameters>[]
      <gml:doubleOrNullTupleList>2</gml:doubleOrNullTupleList>
    </gml:DataBlock>
  </gml:rangeSet>
</GermanyCities>

```

FIGURE 73: GETCOVERAGE (PARTIAL) RESPONSE - DOMAIN SET

Figure 74 illustrates the range set depicting the attribute information retrieved from the cube. The range set is represented using a Data Block with the range parameters composed of a series of axes. The values returned are then appended as a tuple list.

```

<gml:rangeSet>
  <gml:DataBlock>
    <gml:rangeParameters>
      <gml:CompositeValue>
        <gml:valueComponents>
          <Axes>
            <Axis name="Axis0">
              <Tuples>
                <Tuple>
                  <Member Hierarchy="Report Location">
                    <UName>[Report Location].[Alle Report Location].[Nordrhein-Westfalen].[Bonn]
                    </UName>
                    <Caption>Iserlohn</Caption>
                    <LName>[Report Location].[City Name]</LName>
                    <LNum>2</LNum>
                    <DisplayInfo>0</DisplayInfo>
                  </Member>
                </Tuple>
                <Tuple>[]
              </Tuples>
            </Axis>
            <Axis name="Axis1">[]
            <Axis name="SlicerAxis">
              <Tuples>
                <Member Hierarchy="Measures">
                  <UName>[Measures].[Priority Level]</UName>
                  <Caption>Priority Level</Caption>
                  <LName>[Measures].[MeasuresLevel]</LName>
                  <LNum>0</LNum>
                  <DisplayInfo>0</DisplayInfo>
                </Member>
              </Tuples>
            </Axis>
          </gml:valueComponents>
        </gml:CompositeValue>
      </gml:rangeParameters>
    <gml:doubleOrNullTupleList>2</gml:doubleOrNullTupleList>
  </gml:DataBlock>
</gml:rangeSet>
</GermanyCities>

```

FIGURE 74: GETCOVERAGE (PARTIAL) RESPONSE - RANGE SET

In conclusion, although the mapping between the GetCoverage and Execute operations is successful it has been achieved only by extending the existing schema documents as defined by OGC. Also, the response coverage document does not adhere completely to the GML specification. These shortcomings although important are not serious and are common when bringing together domains as different as GIS and OLAP.

Weaknesses and Constraints

As in the case of any multidisciplinary solution there are irregularities and constraints that crop up and need to be addressed. Following are the weaknesses and constraints of the prototype solution:

Weaknesses

1. The GetCoverage request schema has been extended to include a new attribute in the query.
2. The GetCoverage response schema has been modified to return OLAP cube information. The response type Data Block⁴⁸ does not accept the parameters supplied. A viable solution to this problem would be to store the attribute information in a file and allow the coverage to reference this file using a valid URL. The GML specification provides the means of encoding the range set as a binary file provided the binary file can be located remotely from the referencing GML instance.

Constraints

1. Due to temporal constraints the KVP GetCoverage request has not been implemented.
2. The temporal domain is not evaluated to return geographical information but used exclusively for attribute information
3. MDX offers a multitude of functionalities but the prototype solution being restricted by the WCS GetCoverage specification is capable of offering only a small subset of these.

5.5 PROOF OF CONCEPT

In Chapter 2 the author presented the Deutsche Presse Agentur scenario. In this section the author illustrates the effective functionality of the prototype solution.

Recap: *On Monday the 31st of July 2005, Jennifer Apfelschorle – 8 years old at the time was kidnapped from the yard near her home in Bonn, Germany. The ensuing media frenzy and outpouring of concern makes this an interesting case for trend and pattern watching. The editor of a local newspaper wanting to run the story in his newspaper wants to underline his research on crime occurrences in Germany with statistical information. He visually explores the data along the geographical dimension – Bonn, the time dimension – Year 2005, and the subject dimension – crime. The resulting visualization on his desktop encourages an intuitive and production exploration of such a large geospatial data resource. A generalized exploration of all criminal activities in Germany for the time period 2003 to 2005, or a more*

⁴⁸ GML 3.1 specification

specialized, where the editor wants a detailed look in at all the criminal activities in Bonn and Köln only for the month of May, may result in potentially interesting patterns that need to be studied and may lead to consolidated findings.

The scenario can be formulated into queries such as:

1. Query 1: All the criminal cases in Bonn and surroundings for the year 2005.
2. Query 2: All criminal cases in Germany for the time period 2003 to 2005, and
3. Query 3: All criminal cases in Bonn and Köln for the month of May in the year 2005.

In the following section Query 1 and Query 3 are illustrated in detail Query 2 has been left out due to the size of the response document but for the sake of completeness will be included in Appendix C – Example Requests, Responses.

QUERY 1

The query consists of four parts:

1. The geographical aspect (Report Location) – Bonn and surroundings represented as a spatial bounding box.
2. Temporal time period between 01. January 2005 and 31. December 2005.
3. The attribute aspect – Report subject category – crime
4. The measure queried for – Count (number of reports)

Figure 75 illustrates the GetCoverage request the augmented WCS would encounter.

```

<GetCoverage xmlns="http://www.opengis.net/wcs" xmlns:wcs="http://www.opengis.net/wcs" xmlns:ows="http://www.opengis.net/ows">
  <wcs:sourceCoverage>GermanyCities</wcs:sourceCoverage>
  <wcs:domainSubset>
    <wcs:spatialSubset>
      <gml:Envelope srsName='EPSG:4326'>
        <gml:pos dimension='2'>6.95204,50.65149</gml:pos>
        <gml:pos dimension='2'>7.24603,50.82822</gml:pos>
      </gml:Envelope>
      <gml:Grid dimension='2'>[]</gml:Grid>
    </wcs:spatialSubset>
    <wcs:temporalSubset clause="slice">
      <wcs:TimePeriod>
        <wcs:BeginPosition calendarEraName="AC" frame="#ISO-8601" indeterminatePosition="now">
          2005-01-01T00:00:00-05:00
        </wcs:BeginPosition>
        <wcs:EndPosition calendarEraName="AC" frame="#ISO-8601" indeterminatePosition="now">
          2005-12-14T00:00:00-05:00
        </wcs:EndPosition>
        <wcs:TimeResolution>P1D</wcs:TimeResolution>
      </wcs:TimePeriod>
    </wcs:temporalSubset>
  </wcs:domainSubset>
  <wcs:rangeSubset>
    <wcs:FieldSubset>
      <owcs:Identifier clause="axis(0)">[Report Location]</owcs:Identifier>
      <wcs:AxisSubset>
        <owcs:Identifier>[Report Location]</owcs:Identifier>
        <wcs:Key>[City Name]</wcs:Key>
      </wcs:AxisSubset>
    </wcs:FieldSubset>
    <wcs:FieldSubset>
      <owcs:Identifier clause="axis(1)">[Report Subject]</owcs:Identifier>
      <wcs:AxisSubset>
        <owcs:Identifier>[Report Subject]</owcs:Identifier>
        <wcs:Key>Matterid=02001000</wcs:Key>
      </wcs:AxisSubset>
    </wcs:FieldSubset>
    <wcs:FieldSubset>
      <owcs:Identifier clause="slice">[Measures]</owcs:Identifier>
      <wcs:AxisSubset>
        <owcs:Identifier>[Count]</owcs:Identifier>
      </wcs:AxisSubset>
    </wcs:FieldSubset>
  </wcs:rangeSubset>
</GetCoverage>

```

FIGURE 75: SCENARIO 1 - PROOF OF CONCEPT

Figure 76 depicts a valid (partial) GetCoverage response document. The illustration depicts the cities represented by their individual coordinates in the coverage domain set. The request returned seven cities found within the spatial boundaries of the bounding box.

```

<GermanyCities xmlns:gml="http://www.opengis.net/gml">
  <gml:domainSet>
    <gml:MultiPoint id="Wesseling">
      <gml:pointMembers>
        <gml:Point><gml:pos>6.97313 50.82616</gml:pos></gml:Point>
      </gml:pointMembers>
    </gml:MultiPoint>
    <gml:MultiPoint id="Bonn">
      <gml:pointMembers>
        <gml:Point><gml:pos>7.09602 50.73737</gml:pos></gml:Point>
      </gml:pointMembers>
    </gml:MultiPoint>
    <gml:MultiPoint id="Bornheim">
      <gml:pointMembers>
        <gml:Point><gml:pos>6.98684 50.77066</gml:pos></gml:Point>
      </gml:pointMembers>
    </gml:MultiPoint>
    <gml:MultiPoint id="Königswinter">
      <gml:pointMembers>
        <gml:Point><gml:pos>7.1928 50.67958</gml:pos></gml:Point>
      </gml:pointMembers>
    </gml:MultiPoint>
    <gml:MultiPoint id="Siegburg">
      <gml:pointMembers>
        <gml:Point><gml:pos>7.2043 50.79396</gml:pos></gml:Point>
      </gml:pointMembers>
    </gml:MultiPoint>
    <gml:MultiPoint id="Sankt Augustin">
      <gml:pointMembers>
        <gml:Point><gml:pos>7.18271 50.76756</gml:pos></gml:Point>
      </gml:pointMembers>
    </gml:MultiPoint>
    <gml:MultiPoint id="Troisdorf">
      <gml:pointMembers>
        <gml:Point><gml:pos>7.14822 50.81376</gml:pos></gml:Point>
      </gml:pointMembers>
    </gml:MultiPoint>
  </gml:domainSet>
  <gml:rangeSet>

```

FIGURE 76: SCENARIO 1 - GETCOVERAGE (DOMAIN SET) RESPONSE DOCUMENT

Figure 77 depicts the partial range subset returned by the request operation. The range subset is represented by a series of axis information's requested. The first axis represents the report location comprising of a list of three cities – Bonn, Siegburg and Troisdorf which have reported crime cases. The second axis depicts the information regarding the report subject requested for – criminal cases. The order of these dimensions depends on the query formulated.


```

<GermanyCities xmlns:gml="http://www.opengis.net/gml">
  <gml:domainSet>[]
  <gml:rangeSet>
    <gml:DataBlock>
      <gml:rangeParameters>
        <gml:CompositeValue>
          <gml:valueComponents>
            <Axes>
              <Axis name="Axis0">
                <Tuples>
                  <Tuple>
                    <Member Hierarchy="Report Location">
                      <UName>[Report Location].[Alle Report Location].[NordrheinWestfalen].[Bonn]</UName>
                      <Caption>Bonn</Caption>
                      <LName>[Report Location].[City Name]</LName>
                      <LNum>2</LNum>
                      <DisplayInfo>131072</DisplayInfo>
                    </Member>
                    <Member Hierarchy="Report Location">
                      <UName>[Report Location].[Alle Report Location].[NordrheinWestfalen].[Siegburg]</UName>
                      <Caption>Siegburg</Caption>
                      <LName>[Report Location].[City Name]</LName>
                      <LNum>2</LNum>
                      <DisplayInfo>131072</DisplayInfo>
                    </Member>
                    <Member Hierarchy="Report Location">
                      <UName>[Report Location].[Alle Report Location].[NordrheinWestfalen].[Troisdorf]</UName>
                      <Caption>Troisdorf</Caption>
                      <LName>[Report Location].[City Name]</LName>
                      <LNum>2</LNum>
                      <DisplayInfo>131072</DisplayInfo>
                    </Member>
                  </Tuple>
                </Tuples>
              </Axis>
              <Axis name="Axis1">
                <Tuples>
                  <Tuple>
                    <Member Hierarchy="Report Subject">
                      <UName>[Report Subject].[Alle Report Subject].[02000000].[02001000]</UName>
                      <Caption>02001000</Caption>
                      <LName>[Report Subject].[Matterid]</LName>
                      <LNum>2</LNum>
                      <DisplayInfo>1000</DisplayInfo>
                    </Member>
                  </Tuple>
                </Tuples>
              </Axis>
            </gml:valueComponents>
          </gml:CompositeValue>
        </gml:rangeParameters>
      </gml:DataBlock>
    </gml:rangeSet>
  </gml:domainSet>
</GermanyCities>

```

FIGURE 77: SCENARIO 1 - GETCOVERAGE (PARTIAL RANGE SET) RESPONSE DOCUMENT

Figure 78 illustrates the slicer dimensions – the measure – count, and the dimension – time. Also the raw data i.e. the number of reported criminal cases in the three cities identified in the axis dimension – report location. The order of the values corresponds to the order of the names of the cities in the axis dimension.

```

<GermanyCities xmlns:gml="http://www.opengis.net/gml">
  <gml:domainSet>[]
  <gml:rangeSet>
    <gml:DataBlock>
      <gml:rangeParameters>
        <gml:CompositeValue>
          <gml:valueComponents>
            <Axes>
              <Axis name="Axis0">[]
              <Axis name="Axis1">[]
              <Axis name="SlicerAxis">
                <Tuples>
                  <Tuple>
                    <Member Hierarchy="Measures">
                      <UName>[Measures].[Count]</UName>
                      <Caption>Count</Caption>
                      <LName>[Measures].[MeasuresLevel]</LName>
                      <LNum>0</LNum>
                      <DisplayInfo>0</DisplayInfo>
                    </Member>
                    <Member Hierarchy="Report Time">
                      <UName>[Report Time].[Alle Report Time]</UName>
                      <Caption>Alle Report Time</Caption>
                      <LName>[Report Time].[ (Gesamt) ]</LName>
                      <LNum>0</LNum>
                      <DisplayInfo>4</DisplayInfo>
                    </Member>
                  </Tuple>
                </Tuples>
              </Axis>
            </Axes>
          </gml:valueComponents>
        </gml:CompositeValue>
      </gml:rangeParameters>
      <gml:doubleOrNullTupleList>41 3 3</gml:doubleOrNullTupleList>
    </gml:DataBlock>
  </gml:rangeSet>
</GermanyCities>

```

FIGURE 78: SCENARIO 1 - GETCOVERAGE (PARTIAL RANGE SET) RESPONSE DOCUMENT

Figure 79 illustrates the query visualized using the CommonGIS tool. The tool offers an interactive interface to the data allowing the user to select measures, and dimensions. The dimensions time and category are nominated using a slider mechanism whereas the geographical dimension is nominated using the bounding box of the visible area (top left corner of the figure).

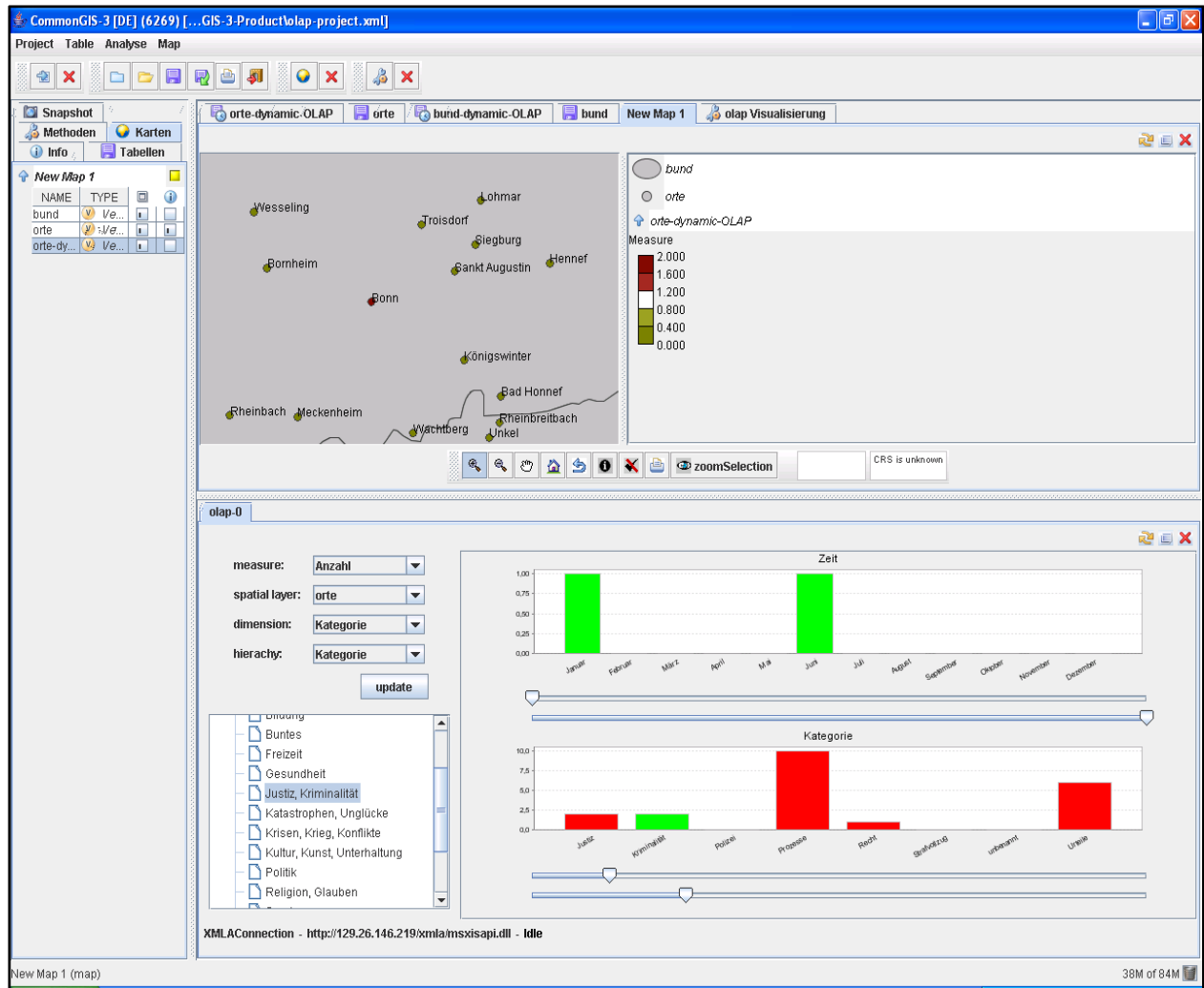


FIGURE 79: SCENARIO VISUALIZATION USING THE COMMONGIS TOOL

In conclusion, the editor of the newspaper can clearly note the high number of criminal cases – 41 – reported in Bonn for the period between January 2005 and December 2005, whereas the neighboring cities – Siegburg and Troisdorf have a relatively small number of reported cases.

QUERY 3

The query consists of four parts:

1. The geographical aspect (report location) – A Bounding Box as a spatial bounding box big enough to include both Bonn and Köln
2. Temporal time period between 01. May 2005 and 31. May 2005.
3. Report subject category – crime
4. Attribute information Report Location specifying the name of the cities the response is to be restricted to. In this case Bonn and Köln.

Figure 80 illustrates the GetCoverage request for the second query example.

```
<GetCoverage xmlns="http://www.opengis.net/wcs" xmlns:wcs="http://www.opengis.net/wcs" xmlns:ows="http://www.opengis.net/ows"
  <wcs:sourceCoverage>GermanyCities</wcs:sourceCoverage>
  <wcs:domainSubset>
    <wcs:spatialSubset>
      <gml:Envelope srsName='EPSG:4326'>
        <gml:pos dimension='2'>6.5571,50.1005</gml:pos>
        <gml:pos dimension='2'>7.7080,51.1586</gml:pos>
      </gml:Envelope>
    </wcs:spatialSubset>
    <wcs:temporalSubset clause="slice">
      <wcs:TimePeriod>
        <wcs:BeginPosition calendarEraName="AC" frame="#ISO-8601" indeterminatePosition="now">
          2005-05-01T00:00:00-05:00
        </wcs:BeginPosition>
        <wcs:EndPosition calendarEraName="AC" frame="#ISO-8601" indeterminatePosition="now">
          2005-05-31T00:00:00-05:00
        </wcs:EndPosition>
        <wcs:TimeResolution>P1D</wcs:TimeResolution>
      </wcs:TimePeriod>
    </wcs:temporalSubset>
  </wcs:domainSubset>
  <wcs:rangeSubset>
    <wcs:FieldSubset>
      <owcs:Identifier clause="axis(0)">[Report Location]</owcs:Identifier>
      <wcs:AxisSubset>
        <owcs:Identifier>[Report Location]</owcs:Identifier>
        <wcs:Key>[City Name]=Bonn</wcs:Key>
        <wcs:Key>[City Name]=Köln</wcs:Key>
      </wcs:AxisSubset>
    </wcs:FieldSubset>
    <wcs:FieldSubset>
      <owcs:Identifier clause="axis(1)">[Report Subject]</owcs:Identifier>
      <wcs:AxisSubset>
        <owcs:Identifier>[Report Subject]</owcs:Identifier>
        <wcs:Key>Detailid=02001007</wcs:Key>
      </wcs:AxisSubset>
    </wcs:FieldSubset>
  </wcs:rangeSubset>
```

FIGURE 80: SCENARIO 2 - PROOF OF CONCEPT

Figure 81 illustrates a partial⁴⁹ GetCoverage response document. The range subset depicts the axis information retrieved with the report location characterized by the two cities – Bonn and Köln. The second axis depicts the report subject information requested for – criminal activities.

⁴⁹ The complete response document can be accessed under the Appendix B – Sample degree Configuration files

```

<GermanyCities xmlns:gml="http://www.opengis.net/gml">
  <gml:domainSet>[]
    <gml:rangeSet>
      <gml:DataBlock>
        <gml:rangeParameters>
          <gml:CompositeValue>
            <gml:valueComponents>
              <Axes>
                <Axis name="Axis0">
                  <Tuples>
                    <Tuple>
                      <Member Hierarchy="Report Location">
                        <UName>[Report Location].[Alle Report Location].[Nordrhein-Westfalen].[Bonn]</UName>
                        <Caption>Bonn</Caption>
                        <LName>[Report Location].[City Name]</LName>
                        <LNum>2</LNum>
                        <DisplayInfo>131072</DisplayInfo></Member>
                      <Member Hierarchy="Report Location">
                        <UName>[Report Location].[Alle Report Location].[Nordrhein-Westfalen].[Köln]</UName>
                        <Caption>Köln</Caption>
                        <LName>[Report Location].[City Name]</LName>
                        <LNum>2</LNum>
                        <DisplayInfo>131072</DisplayInfo>
                      </Member>
                    </Tuple>
                  </Tuples>
                </Axis>
                <Axis name="Axis1">
                  <Tuples>
                    <Tuple>
                      <Member Hierarchy="Report Subject">
                        <UName>[Report Subject].[Alle Report Subject].[02000000].[02001000]</UName>
                        <Caption>02001000</Caption>
                        <LName>[Report Subject].[Matterid]</LName>
                        <LNum>2</LNum>
                        <DisplayInfo>1000</DisplayInfo>
                      </Member>
                    </Tuple>
                  </Tuples>
                </Axis>
                <Axis name="SlicerAxis">[]

```

FIGURE 81: SCENARIO 2 - GETCOVERAGE (PARTIAL RANGE SET) RESPONSE DOCUMENT

Figure 82 illustrates the slicer dimensions – the measure – count, and the dimension – time. Also the raw data i.e. the number of reported criminal cases in the two cities identified in the axis dimension – report location. The order of the values corresponds to the order of the names of the cities in the axis dimension.

```

<GermanyCities xmlns:gml="http://www.opengis.net/gml">
  <gml:domainSet>
    <gml:rangeSet>
      <gml:DataBlock>
        <gml:rangeParameters>
          <gml:CompositeValue>
            <gml:valueComponents>
              <Axes>
                <Axis name="Axis0">
                <Axis name="Axis1">
                <Axis name="SlicerAxis">
                  <Tuples>
                    <Tuple>
                      <Member Hierarchy="Measures">
                        <UName>[Measures].[Count]</UName>
                        <Caption>Count</Caption>
                        <LName>[Measures].[MeasuresLevel]</LName>
                        <LNum>0</LNum>
                        <DisplayInfo>0</DisplayInfo>
                      </Member>
                      <Member Hierarchy="Report Time">
                        <UName>[Report Time].[Alle Report Time]</UName>
                        <Caption>Alle Report Time</Caption>
                        <LName>[Report Time].[ (Gesamt) ]</LName>
                        <LNum>0</LNum>
                        <DisplayInfo>4</DisplayInfo>
                      </Member>
                    </Tuple>
                  </Tuples>
                </Axis>
              </Axes>
            </gml:valueComponents>
          </gml:CompositeValue>
        </gml:rangeParameters>
        <gml:doubleOrNullTupleList>41 317</gml:doubleOrNullTupleList>
      </gml:DataBlock>
    </gml:rangeSet>
  </gml:domainSet>
</GermanyCities>

```

FIGURE 82: SCENARIO 2 - GETCOVERAGE (PARTIAL RANGE SET) RESPONSE DOCUMENT

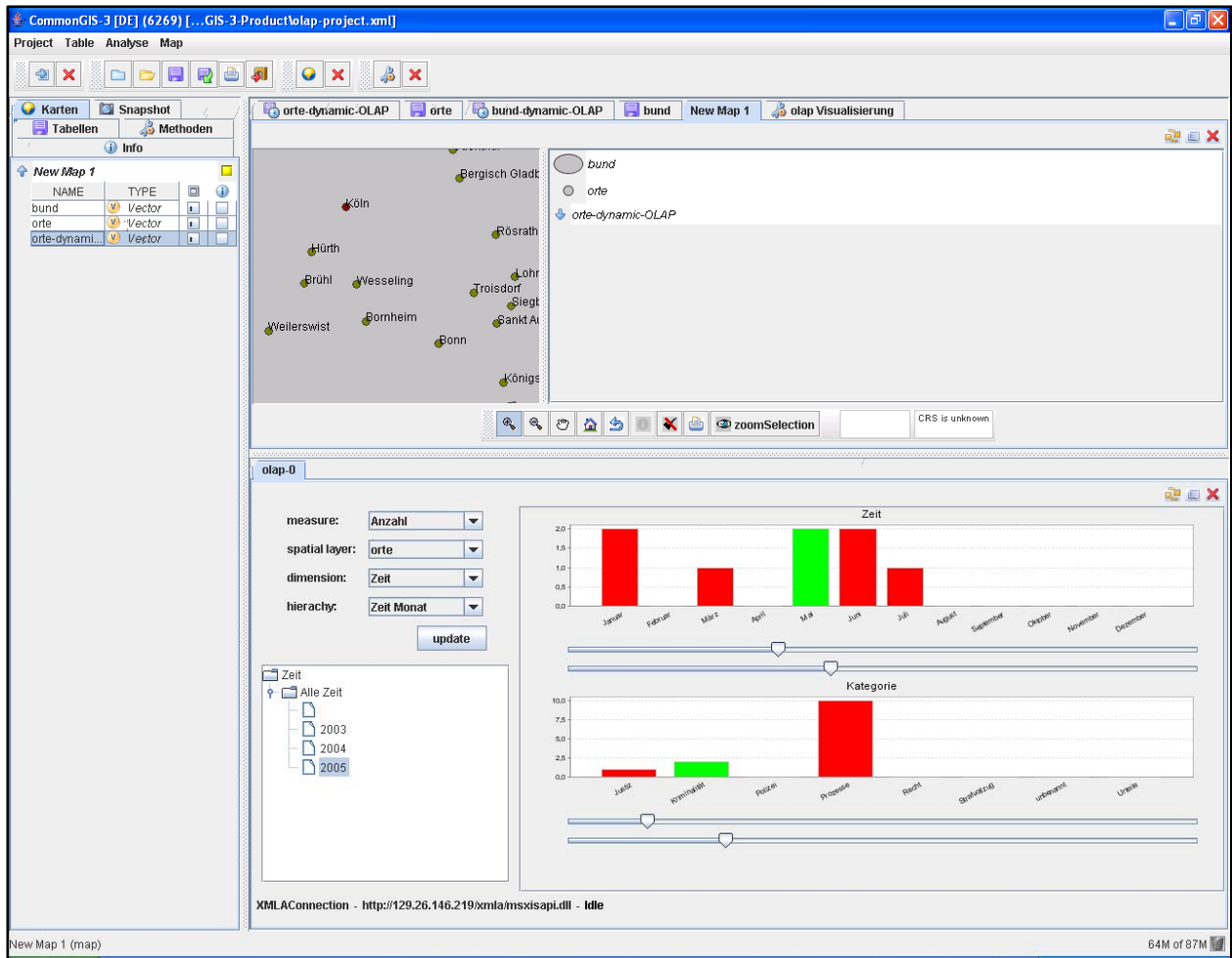


FIGURE 83: SCENARIO VISUALIZATION USING THE COMMONGIS TOOL

Figure 83 illustrates the query visualized using the CommonGIS tool. Both cities, Bonn and Köln, have a number of reported crime cases for the month of May, 2005, where Köln is represented with a high number of cases – 317. The editor of the newspaper can use this information as a base to compare the number of reported criminal cases in the month of May over a period of years. For Bonn, the number of reported cases is 41 which is the same number of cases as reported in scenario 1. This offers a very interesting piece of information – all the reported crime cases for the year 2005 took place only in the month of May. This information can be used by the police to maybe increase their surveillance during the month of May at trouble spots in the city.

The scenarios etched out at the start of the research work helped in setting a guideline for the functionalities the solution should be suitable for. The prototype solution is successful in carrying out the tasks and is a viable base on which further research can be carried out.

5.6 SUMMARY

This chapter on the basis of a case study introduced a prototype solution to fulfill the objectives introduced in Chapter 1. The following table, first introduced in Chapter 4, evaluates the prototype solution.

CHARACTERISTICS	PROTOTYPE SOLUTION – AUGMENTED WCS	PROPOSED SOLUTION
INTEROPERABLE	Yes	
COMPLIANT WITH OGC STANDARDS AND SPECIFICATIONS	Partly – the GetCoverage request and response documents have been modified. The request document has been modified to support an extra attribute element, whereas the response document does not conform fully to the WCS specification	The GetCoverage request being appended with an extra attribute element will not have an effect on the solution being interoperable or functioning with existing clients. Although the GetCoverage response document has been modified the changes are minimal.
COMPLIANT WITH OLAP STANDARDS AND SPECIFICATIONS	Yes	
COMPLETE GIS ANALYSIS FUNCTIONALITIES	Yes	
COMPLETE OLAP ANALYSIS FUNCTIONALITIES	Few – functionalities like drill-down, drill-up, pivot have not been implemented.	These functionalities could be integrated in the GetCoverage request by extending the request document to support extra attribute information similar to the clause attribute extension.
VENDOR NEUTRAL	No – Although the GIS domain is represented by an open source solution the software used for the OLAP domain is a commercial application.	Replacing the MS Analysis Services OLAP server with an open source solution would help overcome this aspect. This is easy to achieve since the solution is modular in nature.
SCALABLE AND EXTENSIBLE	Yes	
DISTRIBUTED COMPUTING	Yes	
QUERYING CAPABILITIES	Single querying interface	

TABLE 12: PROTOTYPE SOLUTION EVALUATION

In summary, the solution offers a promising foundation for further research. There are a few shortcomings which must be overcome for the tool to be a complete viable option but in context of an integrated solution involving two domains, GIS and OLAP, so distinct from each other, they are manageable.

6. CONCLUSION AND FUTURE WORK

This research project started with the objective of creating *middleware* to allow a client to gain access to geographic and multidimensional data. The intention was to provide the client with easy access to the data using an OGC specified WCS interface. This chapter summarizes the research work done. Discussions on future directions follow the summary of the research.

6.1 CONCLUSION

The integration of a GIS and OLAP solution as a single tool provides a wider context for decision support. The capability of the solution to visualize analytical aggregated data provides an innovative approach to provide geospatial and multidimensional data to the user. In this research study the author first highlighted the need for spatial OLAP solutions in order to provide multidimensional information to geospatial users. The concepts of OGC web services and the different OLAP approaches available were introduced. Different works published by both commercial and academic institutes classified the several categories of SOLAP solutions available in the market today. Finally, based on the consolidated findings a prototype solution has been presented to test the concept and highlight the benefits of such an approach to allow users to access quality information using a single interface and existing technologies.

Spatial OLAP solutions are widely available in the market today. This work is one of the few attempts to provide a solution using a standardized open source GIS technology. The solution was implemented using a commercial OLAP server, the MS SQL Analysis Server 2000, and the OGC specified Web Coverage Service. Such a solution can support users in assessing various heterogeneous information sources in the spatial context. The structured multidimensional data with a spatial context at different levels of detail can help answer a series of questions as illustrated in the scenarios in Chapter 2. Furthermore, the cartographic visualization of this data opens up a whole new dimension of decision support opportunities.

The study focuses on the conformance between the two distinct domains with an emphasis on evaluation with the help of a prototype solution. Although the prototype solution presented in this research has some limitations, they are nevertheless typical of multidisciplinary research. For the solution on offer is not a hybrid Spatial OLAP solution but rather a GIS-centric solution. While the

MDX query language is a potent language to query multidimensional data structures, the GetCoverage request in its current form is not sufficient to make best use of a multidimensional dataset. So the question: **Is the WCS interface the correct choice for the interoperability?** It can be answered on the basis of the following aspects: In practical terms of data sharing mechanism, the WCS interface is suitable to share spatial as well as multidimensional data. Clients using the OGC WCS operations (i.e. GetCapabilities, DescribeCoverage, and GetCoverage) can access both types of datasets. However, the querying of the multidimensional data was possible only after extending the GetCoverage request schema with the additional clause parameter. In its current form the WCS is not equipped to query data from cubes using the full functionality, such as pivot, drill-down, drill-up, the OLAP server has to offer. Even so, the extension requirement to support such functionalities is minimal. The developments in the open source domain, especially the GeoServer project, extending the WCS to support multidimensional coverages raises hopes that the request schema of the WCS will be modified in the near future to support querying over n-dimensions. In summary, interoperability is an ambiguous term due to different perceptions and demands of individuals and communities. The best approach for interoperability is therefore users' requirements dependent. Keeping this in mind the interoperability offered by the WCS is limited, but sufficient.

6.2 RECOMMENDATIONS AND FUTURE WORK

The proposed prototype presents convincing arguments for performing spatial multidimensional data exploration using the WCS. Nonetheless, the author foresees further improvements and extensions of the solution. These are some of the issues that should be considered in the future:

1. Complete existing WCS implementation. The WCS implementations available in the open source domain leave a lot to be desired in respect of the functionalities implemented as characterized in the specification.
2. Current WCS implementations only support grid and image coverages. This is a short coming which must be addressed in the near future. There are a few projects, for example THREDDS⁵⁰ and DEWS⁵¹, which are currently extending the WCS specification to implement Point coverages. It is imperative for deegree and GeoServer to consider introducing support for further GML coverage types.
3. The deegree WCS is specification 1.0 reference implementation. There seems to be no activity by deegree to *upgrade* their WCS implementation to WCS specification 1.1.

⁵⁰ Thematic Realtime Environmental Distributed Data Services

⁵¹ <http://www.resc.rdg.ac.uk/twiki/bin/view/Resc/DeWS>

GeoServer on the other hand is already a reference implementation for the current specification and is looking towards extending it, instead of deegree, to support multidimensional coverages. It might be of interest to consider it once the architectural changes are completed.

4. Investigating ways to introduce OLAP analysis functionalities in the GetCoverage request would enable the current solution to be a very viable hybrid solution. An example would be extending the request operation with additional attribute values as done in the envisaged prototype (see Section 5.4).
5. OGC is contemplating adoption of a technology called Web Processing Service (WPS). WPS is a generic web service in that the specific processes provided by a WPS implementation are defined by the owner of that implementation. A WPS can be configured to offer any sort of GIS functionality. It takes a defined set of inputs, applies a calculation, and produces a set of outputs. WPS is designed to work with spatially referenced data but can also be used to work with other kinds of data. WPS is still in its infancy and is not an OGC standard. But it offers interesting possibilities especially keeping its generic and open ended nature in mind.
6. (Baumann, 2006) introduces the Web Coverage Processing Service (WCPS). WCPS supports the WCS functionalities and extends it with an expression language to form requests of arbitrary complexity allowing e.g., multi-valued coverage results. The expression language being developed inherently supporting not only geographic data but also geophysical, engineering, etc. Currently, the WCPS is an OGC Best Practice paper. If collaboration with this group is possible, it would be advisable to introduce the OLAP query language, MDX, to this mix. Thus overcoming the shortcomings of this research work.
7. Although open source solutions have matured in recent years their use and deployment is still a time consuming task. The temporal restrictions of a master thesis did not permit using an open source OLAP solution. This is definitely an aspect to look into. At this stage Mondrian is the best open source OLAP solution available in the market and the integration should be carried out with this solution.

The present research contributes to a class of efficient spatial as well as multidimensional data exploration solutions. This sets a good starting point for a new range of research projects focusing on the exploration and analysis of these data sets. There is a strong potential for coupling this approach to other services available, such as the Web Processing Service, offering better data mining algorithms and other fast aggregated views of the spatial multidimensional data. While contemporary

solutions offer some of the functionalities of the two domains, none permit the usage of the entire palette of OLAP and spatial analysis operators. Here again, there is a strong opportunity to contribute to this already very promising field.

7. REFERENCES

- 211/WG, ISO/TC. 2004.** *Geography Markup Language (GML)*. s.l. : OGC GML RWG, 2004.
- Alameh, N. 2003.** *Chaining Geographic Information Web Services*. s.l. : IEEE Internet Computing, 2003.
- . **2001.** *Scalable and Extensible Infrastructures for Distributing Interoperable Geographic Information Services on the Internet*. USA : PhD Dissertation, 2001.
- . **2002.** Service Chaining of Interoperable Geographic Information Web Services. *Massachusetts Insitute of Technology*. [Online] 2002. [Cited: August 07, 2007.]
<http://web.mit.edu/nadinesa/www/paper2.pdf>.
- AltaPlana. 2004.** OLAP and OLAP Server Definitions. [Online] 2004. [Cited: August 02, 2007.]
<http://altaplana.com/olap/glossary.html>.
- Baltzer, Oliver. 2006.** cgmLab Portal. [Online] 7 10, 2006. [Cited: August 15, 2007.]
<http://cgmlab.cs.dal.ca/Members/obaltzer/SOLAP/>.
- Baumann, Peter. 2006.** *OGC Best Practices: Web Coverage Processing Service (WCPS)*. s.l. : OGC 06-035r1, 2006.
- . **1999.** *Towards a Standard for Interoperable Earth System Raster Services*. Munich, Germany : Shaping the Change, XXIII FIG Congress, 1999.
- Beaujardière, de La, J. (editor). 2002.** *Web Map Service Implementation Specification, version 1.1.1*. USA : OpenGIS Consortium Inc., 2002. OGC 01-068r3.
- Bédard, Y and Fortin, M. October, 2004.** *Développement d'un système de découverte des connaissances spatio-temporelles issues d'un chantier de fouilles archéologiques*. Canadian Institute of Geomatics, Montreal, Canada : Paper presented at Géomatique, October, 2004.
- Bédard, Y. 2005/02/10.** *Mariage de la géomatique et de l'informatique décisionnelle: le programme de recherche de la nouvelle Chaire industrielle CRSNG en bases de données géospatiales décisionnelles*. [Powerpoint] Canda : Université Laval, 2005/02/10.
- Bédard, Y, et al. 1997.** *Étude de l'état actuel et des besoins de R&D relativement aux architectures et technologies des data warehouses appliquées aux données spatiales: Research Report for National Defence Canada Centre for Research in Geomatics*. Laval University : s.n., 1997.
- Bédard, Y, Merrett, T and Han, J. 2001.** *Fundamentals of spatial data warehousing for geographic knowledge discovery*. s.l. : Geographic Data Mining and Knowledge Discovery, 2001.
- Bédard, Y, Proulx, M and Rivest, S. 2005.** *Enrichissement du OLAP pour l'analyse géographique: exemples de réalisation et différentes possibilités technologiques*. Paris : Bentayeb, F., Boussaid, O., Darmont, J., Rabaseda, S. (Eds.), Entrepôts de Données et Analyse en ligne, 2005.

- Bédard, Y., and Bernier E. 2002.** *Supporting multiple representation with spatial databases views management and the concept of VUEL*. Ottawa Canada : In Proceedings of Workshop on Multi-Scale Representations of Spatial Data, 2002.
- Berson, A and Smith, S. 1997.** *Data Warehousing, Data Mining, and OLAP*. s.l. : McGraw-Hill, 1997.
- BI. 2005.** *Business Analytics and Data Visualization*. 2005.
- Bimonte, S, Tchounikine, A and Miquel, M. 2006.** *GeoCube, a Multidimensional Model and Navigation Operators Handling Complex Measures: a Spatial OLAP Application*. Izmir, Turkey : In Proceedings of the 4th Biennial International Conferences Advances in Information Systems, 2006.
- Bimonte, Sandro, Tchounikine, Anne and Miquel, Maryvonne.** *Spatial OLAP: Open Issues and a Web Based Prototype*. Villeurbanne Cedex, France : Laboratoire d'InfoRmatique en Images et Systèmes d'information.
- Bishr, Y, Radwan, M and Molenaar, M. 1998.** *Aspects of interoperability: a GII perspective*. Netherlands : International Institute for Aerospace Survey and Earth Sciences (ITC), 1998.
- BrainBell. 2006.** Php - Php Mysql - Three Tier Architectures Tutorial. [Online] BrainBell, 2006. [Cited: August 15, 2007.]
http://www.brainbell.com/tutors/php/php_mysql/Three_Tier_Architectures.html.
- Brentjens, T J. 2004.** *OpenGIS Web Feature Services for editing cadastral data*. Apeldoorn, The Netherlands : s.n., 2004.
- Bühler, K and McKee, L. 1998.** *The OpenGIS Guide: Introduction to Interoperable Geoprocessing and the OpenGIS Specification*. Waltham, USA : OpenGIS Consortium Inc, 1998.
- Burrough, P H. 1990.** *Methods of spatial analysis in GIS*. s.l. : International Journal of Geographical Information Systems, 1990.
- Caron, P Y. 1998.** *Étude du potentiel OLAP pour supporter l'analyse spatio-temporelle*. Laval University, Canada : MSc Thesis, 1998.
- Carver, S. 1991.** *Integrating multicriteria evaluation with GIS*. s.l. : International Journal of Geographical Information Systems, 1991.
- Champion, M. 2002a.** *Web Services Architecture*. Cambridge : a, 2002a.
- Chappel, D. 2002.** *Understanding .NET A Tutorial and Analysis*. Boston : s.n., 2002.
- Chaudhari, S and Dayal, U. 1997.** *An Overview of Data Warehousing and OLAP Technology*. s.l. : SIGMOD Record 26(1), 1997.
- . 1996. *Decision Support, Data Warehouse, and OLAP*. s.l. : Tutorials of the Twenty-Second international Conf. On Very Large Data Base, 1996.
- Chrisman, N. 1996.** *Exploring Geographic Information Systems*. s.l. : John Wiley and Sons, 1996.

- Codd, E F, Codd, S B and Salley, C T. 1993.** *Providing OLAP (On-Line Analytical Processing) to User-Analysts: An IT Mandate*. s.l. : Hyperion white papers, 1993.
- Cox, S, et al. 2003.** *OpenGIS Geography Markup Language (GML) Implementation Specification, version 3*. USA : OpenGIS Consortium Inc, 2003. Reference number: OGC 02-023r4.
- Curbera, F. 2002.** *Unraveling the Web Services Web; An Introduction to SOAP, WSDL and UDDI*. Washington : In IEEE Internet Computing 2, 2002.
- Databeacon; 2004.** *Why OLAP - An overview of On-Line Analytical Processing*. 2004.
- deegree - Building Spatial Data Infrastructures based on Free Software. [Online] [Cited: September 08, 2007.] <http://www.deegree.org/>.
- Ding, Hao, Pascoe, Richard and Churcher, Neville. 2002.** *Implementing OGC Web Map Service Client Applications Using JSP, JSTL, and XMLC*. University of Canterbury, New Zealand : s.n., 2002.
- ESRI - OLAP for ArcGIS Overview. [Online] [Cited: September 08, 2007.] <http://www.esri.com/software/arcgis/extensions/olap/about/overview.html>.
- ESRI, Information Builders. 2005.** *Geographic Information Systems and Business Intelligence: Perfect Together*. 2005.
- Evans, J D. 2005.** *Web Coverage Service (WCS), Version 1.0.0*. s.l. : Open Geospatial Consortium Inc., 2005.
- Ferreira, A, Campos, M and Tanaka, A. 2001.** *An Architecture for Spatial and Dimensional Analysis Integration*. s.l. : In Proceedings of SCI 2001, v.XIV CSE, Part II, 2001.
- Fidalgo, R, et al. 2004.** *Geodnframe: A framework for guiding the design of geographical dimensional schemas*. s.l. : In Data Warehousing and Knowledge Discovery (DaWaK), 2004.
- Fidalgo, R, et al. 2004.** *Providing multidimensional and geographical integration based on a GDW and metamodels*. s.l. : In Brazilian Symposium on Databases, 2004.
- Frankin, C. April 1992.** *An Introduction to Geographic Information Systems: Linking Maps to Databases*. s.l. : Database, April 1992.
- GeoServer. [Online] [Cited: September 08, 2007.] <http://geoserver.org/>.
- Gill, H and Rao, P. 1996.** *The official Guide to Data Warehousing*. s.l. : QUE Corporation, 1996.
- Glymour, C, et al. 1997.** *Statistical themes and lesson for data mining: Data mining and knowledge discovery*. 1997.
- Goodchild, M. 1992.** *Geographical Data Modeling*. s.l. : Computers & Geosciences, 1992.
- Hernández, V, et al. 2005.** *Sustainable decision support by the use of multilevel and multi-criteria spatial analysis on the Nicaragua Development Gateway*. Egypt : In Proc. 8th International Conference on the Global Spatial Data Infrastructure, 2005.

Hyperion - The Business Performance Management Software Leader. [Online] [Cited: September 08, 2007.] <http://www.hyperion.com/>.

IBM Corporation. [Online] [Cited: September 08, 2007.] <http://www.ibm.com>.

International Organization for Standardization. [Online] [Cited: September 08, 2007.] http://www.iso.org/iso/iso_catalogue.htm.

Jannsen, R and Rietveld, P. 1990. *Multicriteria analysis and geographical information systems: An application to agricultural land-use in Netherlands*. Dordrecht : Geographical Information Systems for Urban and Regional Planning, 1990.

Kouba Z., Matousek K., and Miksovsky P. 2000. *On Data Warehouse and GIS integration*. Greenwich, UK : In Proceedings of the 11th Int. Conf. and Workshop on Database and Expert System Applications, 2000.

Kouba, Z, Matousek, K and Miksovsky, P. 2000. *On Data Warehouse and GIS integration*. s.l. : In Proceedings of DEXA 2000, LNCS 1873, 2000.

Lake, Ron. 2006. *GML and Building the GeoWeb*. s.l. : Galdos Systems Inc., 2006.

lat-lon. 2006. *degree - Web Coverage Service*. Bonn, Germany : s.n., 2006.

LGS Group, Inc. 2000. *Analysis of Health Surveillance Business Intelligence Tools and Applications*. 2000.

Longley, P. 1999. *Geographical Information Systems: Principles, Techniques, Applications and Management*. s.l. : John Wiley and Sons, 1999.

MacEachren, A M and Kraak, M J. 2001. *Research challenges in geovisualization, Cartography and Geographic Information Science*. 2001.

MapInfo Corporation. [Online] [Cited: September 08, 2007.] <http://www.mapinfo.com/location/integration>.

Marchand, P, Bédard, Y and Edwards, G. 2001. *A hyper cube-based method for spatio-temporal exploration and analysis*. s.l. : Geoinfomatica, 2001.

Matias, Rosa and Moura-Pires, Joao. 2005. *Spatial On-Line Analytical Processing (SOLAP): A Tool to Analyse the Emission of Pollutants in Industrial Installations*. Portugal : s.n., 2005.

McKee, Lance, Kuhn, Wayland and Kuhn, Werner. 1997. *The OpenGIS Consortium's Purpose and Technical Approach*. Münster, Germany : Photogrammetric Week '97, 1997.

Microsoft SQL Server. [Online] [Cited: September 08, 2007.] <http://www.microsoft.com/sql/default.msp>.

Miquel, Maryvonne, et al. 2002. *Modeling Multidimensional Spatio-Temporal data warehouses in a context of evolving specifications*. Ottawa : Symposium on Geospatial Theory, Processing and Applications, 2002.

Navathe, Shamkant. 2004. *Database Technology and Challenge of Emerging Applications*. Atlanta : s.n., 2004.

- OGC.** Reference Implementations | Compliance and Interoperability Testing Initiative (CITE). [Online] OGC - Open Geospatial Consortium Inc. [Cited: September 15, 2007.] <http://cite.opengeospatial.org/reference>.
- Open Geospatial Consortium. [Online] [Cited: September 08, 2007.] <http://www.opengeospatial.org/>.
- OpenGIS. 2003.** *OpenGIS Web Services Architecture, Discussion Paper*. 2003.
- Oracle. 2003.** *Application Developer's Guide*. 2003.
- Oracle Corporation. [Online] [Cited: September 08, 2007.] <http://www.oracle.com/index.html>.
- OracleDBAExpert. 2003.** The very basics of data warehouse design. [Online] October 12, 2003. [Cited: September 28, 2007.] <http://www.oracledbaexpert.com/oracle/papers/TheVeryBasicsOfDataWarehouseDesign.htm>.
- Page, OGC Web Services Initiative 1 (OWS1) Baseline Documents. 2002.** OGC. [Online] 2002. [Cited: August 07, 2007.] <http://ip.opengis.org/ows1/docIndex.html>.
- Pederson, Torben and Jensen, Christian. 2001.** *Multidimensional Database Technology*. Aalborg University : s.n., 2001.
- Pendse, N. 2000.** Glossary, The OLAP Report. [Online] 2000. [Cited: August 02, 2007.] <http://www.olapreport.com/glossary.htm>.
- Pentaho Analysis Services: Mondrian Project. [Online] [Cited: September 08, 2007.] <http://mondrian.pentaho.org/>.
- PostgreSQL - The world's most advanced open source database. [Online] [Cited: September 08, 2007.] <http://www.postgresql.org/>.
- Raden, Niel. 1997.** Hired Brains - Articles. [Online] 1997. [Cited: September 15, 2007.] <http://www.hiredbrains.com/artic1.html>.
- 1997.** Red Brick Systems, Inc. [Online] 1997. [Cited: August 02, 2007.] <http://www-306.ibm.com/software/data/informix/redbrick/>.
- Rivest, S, et al. 2003.** *SOLAP: A new type of user interface to support spatio-temporal multidimensional data exploration and analysis*. Quebec, Canada : Proceedings of the ISPRS Joint Workshop on Spatial, Temporal and Multi-Dimensional Data Modelling and Analysis, 2003.
- Rivest, S, et al. 2005.** *SOLAP: Merging Business Intelligence with Geospatial Technology for Interactive Spatio-Temporal Exploration and Analysis of data*. s.l. : Journal of International Society for Photogrammetry and Remote Sensing, 2005.
- Rivest, S., Bédard Y., and Marchand P. 2001.** *Toward Better Support for Spatial Decision Making: Defining the Characteristics of Spatial On-Line Analytical Processing (SOLAP)*. s.l. : Geomatica, 2001.

- Rodolphe, Devillers. 2004.** *Conception d'un système multidimensionnel d'information sur la qualité des données géospatiales*. University Laval, Canada : s.n., 2004.
- Scotch, M and Parmanto, B. 2005.** *SOVAT: Spatial OLAP Visualization and Analysis Tool*. Waikoloa : In Proceedings of HICSS-38, IEEE, 2005.
- Shekhar S., et al. 2000.** *Map Cube_ A Visualization Tool for Spatial Data Warehouse*. 2000.
- Silva, J, Times, V and Salgado, A. 2006.** *An open source and web based framework for geographic and multidimensional processing*. s.l. : In Proceedings of Symposium on Applied Computing, ACM press, 2006.
- Stefanovic, N. 1997.** *Design and implementatin of online analytical processing (OLAP) of spatial data*. Simon Fraser University, Canada : s.n., 1997.
- Sybase - Database Management, Mobile Solutions and Messaging, Analytics, Data Integration.
[Online] [Cited: September 08, 2007.] <http://www.sybase.com/>.
- SysOnyx. 2002.** XML Schema Editor - a white paper. [Online] SysOnyx, 2002. [Cited: September 28, 2007.] <http://www.sysonyx.com/xml-schema-editor/Introduction.asp>.
- Tchounikine A., et al. 2005.** *Reveu des Nouvelles Technologies de l'Information (RNTI)*. s.l. : Editions Cepadué, numéro spécial, 2005.
- Teradata Data Warehousing. [Online] [Cited: September 08, 2007.] <http://www.teradata.com/t/>.
- The OLAP Report. [Online] [Cited: September 08, 2007.] <http://www.olapreport.com/>.
- Thomsen, E. 2002.** *OLAP Solutions: Building Multidimensional Information Systems, 2nd Edition*. s.l. : John Wiley and Sons, 2002.
- Thomsen, E, Spofford, G and Chase, D. 1999.** *Microsoft OLAP Solutions*. New York : John Wiley & Sons, Inc, 1999.
- Twardoch, Adam. 2003.** *Web Services Technologies: XML, SOAP and UDDI*. [Dokument] Frankfurt (Oder) : European University Viadrina Frankfurt (Oder), 2003.
- . 2003. *Web Services Technologies: XML, SOAP and UDDI*. Frankurt (Oder) : s.n., 2003.
- W3C - World Wide Web Consortium. [Online] [Cited: September 08, 2007.] <http://www.w3.org/>.
- Walsh, Norman. 1998.** A Technical Introduction to XML. [Online] February 18, 1998. [Cited: August 02, 2007.] <http://nwalsh.com/docs/articles/xml/>.
- White, Colin. 2003.** Intelligent Business Strategies: OLAP in the Database. [Online] June 2003. [Cited: August 18, 2007.] http://www.dmreview.com/article_sub.cfm?articleId=6807.
- Whiteside, A and Evans, J. 2006.** *Web Coverage Service (WCS) Implementation Specification, Version 1.1.0*. s.l. : Open Geospatial Consortium Inc., 2006. 06-083r8.
- Wojnarowska, Marta and Ady, Bridget. 2002.** *Interoperable Solutions in Web-based Mapping*. [Dokument] Calgary, Alberta : GeoArtic International Services Ltd., 2002.

Wolter, R. 2001. XML Web Services Basics. [Online] December 2001. [Cited: August 02, 2007.] <http://msdn2.microsoft.com/en-us/library/ms996507.aspx>.

Worboys, M. 1995. *GIS: A Computing Perspective*. London, UK : Taylor and Francis, 1995.

XMLA, XML for Analysis, ODBO, OLE DB for OLAP, XMLA provider, bridge, server. [Online] [Cited: September 08, 2007.] <http://www.xmlforanalysis.com/>.

Zhaohui, Tang and Sigmond. 2005. *Building Data Mining Solutions with OLE DB for DM and XML for Analysis*. 2005.

8. APPENDIX A – GLOSSARY

The terminology has changed as much as technology in this field. Many of the terms are interchangeable. Here are a few common terms.

Application Server

Server software that manages one or more pieces of software in a way that makes the managed software available over a network, usually to a Web service

Analysis Server

The server component of Analysis Services that is specifically designed to create and maintain multidimensional data structures and provide multidimensional data in response to client queries

Axis

A set of tuples. Each tuple is a vector of members. A set of axes defines the coordinates of a multidimensional data set.

Business Intelligence (BI)

Application and process of analyzing business data. This is generally synonymous with *decision support systems* (DSS). Solutions that incorporate Web site data as well are sometimes called e-Business Intelligence solutions

Calculated Member

A member of a dimension whose value is calculated at run time using an expression. Calculated member values may be derived from other members' values. A calculated member is any member that is not an input member. For example, a calculated member Profit can be determined by subtracting the value of the member Costs from the value of the member Sales.

Cell

A cell is a single point in a cube. Cubes have cells for all possible combinations of points from all of the cube's dimensions.

Coordinate

One of a sequence of n numbers designating the position of a point in n -dimensional space

Coordinate conversion

Change of coordinates, based on a one-to-one relationship, from one coordinate reference system to another based on the same datum.

Coordinate reference system

Coordinate system which is related to the real world by a datum.

Coordinate system

Set of mathematical rules for specifying how coordinates are to be assigned to points.

Cube

A multidimensional structure that forms the basis for OLAP applications. Despite the name, most OLAP cubes have more than three dimensions.

Data Model

A data model is an abstract representation of the data used by an organization, such that a meaningful interpretation of the data may be made by the model's readers. The data model may be at a conceptual, external or internal level.

Database Management System (DBMS)

Used to store, process and manage data in a systematic way. May use a variety of underlying storage methods, including relational, multidimensional, network and hierarchical.

Data warehouse

A data warehouse is a non-volatile store of cleansed, possible summarized data used for analysis purposes. Another term for this is the *relational data store*.

Data warehousing

The process of collecting, cleansing, sifting, and storing data from a variety of operational sources, and delivering that information to end users for reporting and analysis. While a data

warehouse is a repository, data warehousing is an end-to-end process, reaching from the source systems to the end user.

Datum

Parameter or set of parameters that define the position of the origin, the scale, and the orientation of a coordinate reference system

Dimension

A collection of data of the same type, allowing us to structure the multidimensional database. A dimension is sometimes referred to as an axis. In a measure, each cell of data is associated with one single position in each dimension. Time, Location and Product are the classic dimensions.

Decision Support System (DSS)

It's a data interrogation and presentation system, specifically used for decision-making.

Drill Up/Drill Down

A technique for navigating through levels of data ranging from the most summarized (up) to the most detailed (down). For example, when viewing the details of sales data by year, a user can drill down to display sales data by quarter, and further to display data by month.

Drill Through

To retrieve the detailed data from which the data in a cube cell was summarized.

Execute Information System (EIS)

EIS, or Decision Support System

Fact

A row in a fact table in a data warehouse. A fact contains values that define a data event such as a sales transaction.

Fact Table

A central table in a data warehouse schema that contains numerical measures and keys relating facts to dimension tables. Fact tables contain data that describes specific events within a business, such as bank transactions or product sales.

Feature

Abstraction of real world phenomena

Grid

1. A network of uniformly spaced points or lines on the CRT for location points.
2. A set of regularly spaced sample points.
3. In cartography, an exact set of reference lines over the earth's surface.

Hierarchy

The positions of a dimension organized according to a series of cascading one to many relationships. This way of organizing data is comparable to a logical tree, where each member has only one parent but a variable number of children.

HOLAP

Hybrid OLAP. A storage mode that uses a combination of multidimensional data structures and relational database tables to store multidimensional data. Analysis Services stores aggregations for a HOLAP partition in a multidimensional structure and stores facts in a relational database.

Interoperability

Is the ability of a system or a product to work with other systems or products without special effort on the part of the customer

Metadata

Metadata is data about data. It typically includes information such as accuracy, quality, extent, custodianship, and collection methodology. Meta-data is typically stored in data models, dictionaries, schemas and other representations.

Multidimensional expression language (MDX)

The multidimensional language equivalent to SQL. The language used to define multidimensional data selections and calculations in Microsoft's OLE DB.

Measure

In a cube, a set of values that are based on a column in the cube's fact table and are usually numeric. Measures are the central values that are aggregated and analyzed.

Member

An item in a dimension representing one or more occurrences of data. A member can be either unique or non-unique. For example, 1997 and 1998 represent unique members in the year level of a time dimension, whereas January represents non-unique members in the month level because there can be more than one January in the time dimension if it contains data for more than one year.

MOLAP

Multidimensional OLAP. A storage mode that uses a proprietary multidimensional structure to store a partition's facts and aggregations or a dimension. The data of a partition is completely contained within the multidimensional structure. Multidimensional database

OLAP

On-Line Analytical Processing, or OLAP, is a term coined in 1933 by Dr. E.F. Codd. He defines OLAP applications as applications that deliver fast analysis of shared, multidimensional information.

OLTP

Online Transactional Processing. A data processing system designed to record all of the business transactions of an organization as they occur. An OLTP system is characterized by many concurrent users actively adding and modifying data.

Private Dimension

A dimension created for and used by a specific cube. Unlike shared dimensions, private dimensions are available only to the cube in which they are created.

Process

In a cube, the series of operations that rebuilds the cube's structure, loads data into a multidimensional structure, calculates summaries, and saves the precalculated aggregations. As a verb, to populate a cube with data and aggregations. It is one of three processing options for a cube.

ROLAP

Relational OLAP. A storage mode that uses tables in a relational database to store multidimensional structures.

Schema

In Analysis Services, a schema is a description of multidimensional objects such as cubes and dimensions.

SELECT

In Analysis Services, the MDX statement used to query cubes and return record sets of multidimensional data.

Shared Dimension

A dimension created within a database that can be used by any cube in the database.

Slice

A subset of the data in a cube, specified by limiting one or more dimensions by members of the dimension. For example, facts for a particular year constitute a slice of multiyear data.

Slicer

A dimension used to filter multidimensional data. This dimension appears in the WHERE clause of a MDX statement.

Snowflake Schema

An extension of a star schema such that one or more dimensions are defined by multiple tables. In a snowflake schema, only primary dimension tables are joined to the fact table. Additional dimension tables are joined to primary dimension tables.

Star Schema

Arrangement of data in a relational database. In the middle is the fact table, whose columns constitute the multidimensional measures. The branches of the star, which radiate from the fact table, correspond to the dimensions. The conceptual data model represents this star schema.

Time Dimension

A dimension that breaks time down into levels such as Year, Quarter, Month, and Day. In Analysis Services, a special type of dimension created from a date/time column.

Tuples

An ordered collection of members from different dimensions. For example, is a tuple formed by members of two dimensions: Geography and Time. A single member is a degenerated case of a tuple and can be used as an expression without the parentheses.

Vector

A data structure for representing point and line data by means of 2- or 3-dimensional geometric (Cartesian x,y or x,y,z) coordinates with exacting topological requirements. In connection with GIS and computer graphics, "vector" can refer to a set of line segments joined end-to-end to make a curved path in space. Vector objects also have a set of attributes (such as ID number, color, drawing style) attached to them.

XML

eXtensible Markup Language, the emerging standard for defining, representing and dynamically sharing information across the Internet. Already widely used in OLAP products.

XML for Analysis

Microsoft's new multi-platform version of OLE DB for OLAP that allows MDX queries to be handled using XML documents. Not yet in widespread use, but supported by Hyperion Essbase 7 and will be the native query API Microsoft for SQL Server 2005.

9. APPENDIX B – SAMPLE DEEGREE CONFIGURATION FILES

WEB COVERAGE SERVICE – WEB.XML

```
<?XML VERSION="1.0"?>
<!DOCTYPE WEB-APP PUBLIC "-//SUN MICROSYSTEMS, INC.//DTD WEB APPLICATION 2.3//EN"
"HTTP://JAVA.SUN.COM/DTD/WEB-APP_2_3.DTD">
<WEB-APP>
  <DISPLAY-NAME>DEEGREE WCS</DISPLAY-NAME>
  <DESCRIPTION>DEEGREE 2.0 WEB COVERAGE SERVICE</DESCRIPTION>
  <SERVLET>
    <SERVLET-NAME>OWSERVICE</SERVLET-NAME>
    <SERVLET-CLASS>
      ORG.DEEGREE.ENTERPRISE.SERVLET.OGCSERVLETCONTROLLER
    </SERVLET-CLASS>
    <INIT-PARAM>
      <PARAM-NAME>SERVICES</PARAM-NAME>
      <PARAM-VALUE>WCS</PARAM-VALUE>
      <DESCRIPTION>
        LIST OF SUPPORTED SERVICES, E.G.: WFS,WMS (COMMA SEPARATED)
      </DESCRIPTION>
    </INIT-PARAM>
    <INIT-PARAM>
      <PARAM-NAME>WCS.HANDLER</PARAM-NAME>
      <PARAM-VALUE>ORG.DEEGREE.ENTERPRISE.SERVLET.WCSHANDLER</PARAM-VALUE>
    </INIT-PARAM>
    <INIT-PARAM>
      <PARAM-NAME>WCS.CONFIG</PARAM-NAME>
      <PARAM-VALUE>WEB-INF/CONF/WCS/WCS_CONFIGURATION.XML</PARAM-VALUE>
    </INIT-PARAM>
    <LOAD-ON-STARTUP>1</LOAD-ON-STARTUP>
  </SERVLET>
  <SERVLET-MAPPING>
    <SERVLET-NAME>OWSERVICE</SERVLET-NAME>
    <URL-PATTERN>/OGCWEBSERVICE</URL-PATTERN>
  </SERVLET-MAPPING>
  <WELCOME-FILE-LIST>
    <WELCOME-FILE>INDEX.JSP</WELCOME-FILE>
```

```
</WELCOME-FILE-LIST>
<ERROR-PAGE>
  <ERROR-CODE>500</ERROR-CODE>
  <LOCATION>/ERROR.JSP</LOCATION>
</ERROR-PAGE>
<ERROR-PAGE>
  <EXCEPTION-TYPE>
    ORG.DEEGREE.OGCWEBSERVICES.OGCWebServiceException
  </EXCEPTION-TYPE>
  <LOCATION>/ERROR.JSP</LOCATION>
</ERROR-PAGE>
</WEB-APP>
```

WCS – CONFIGURATION FILE

```
<?XML VERSION="1.0" ENCODING="UTF-8"?>
<WCS_CAPABILITIES XMLNS="HTTP://WWW.OPENGIS.NET/WCS"
  XMLNS:GML="HTTP://WWW.OPENGIS.NET/GML"
  XMLNS:XLINK="HTTP://WWW.W3.ORG/1999/XLINK"
  XMLNS:DEEGREE="HTTP://WWW.DEEGREE.ORG/WCS"
  VERSION="1.0.0" UPDATESEQUENCE="1.0.0">
  <DEEGREE:DEEGREEPARAM>
    <DEEGREE:DEFAULTONLINERESOURCE
      XMLNS:XLINK="HTTP://WWW.W3.ORG/1999/XLINK"
      XLINK:TYPE="SIMPLE" XLINK:HREF="HTTP://127.0.0.1:8080/DEEGREEWCS" />
    <DEEGREE:CACHESIZE>250</DEEGREE:CACHESIZE>
    <DEEGREE:REQUESTTIMELIMIT>10</DEEGREE:REQUESTTIMELIMIT>
    <DEEGREE:DATADIRECTORYLIST>
      <DEEGREE:DATADIRECTORY>C:/DATA</DEEGREE:DATADIRECTORY>
    </DEEGREE:DATADIRECTORYLIST>
  </DEEGREE:DEEGREEPARAM>
  <SERVICE>
    <METADATALINK XLINK:TYPE="SIMPLE"
      XLINK:HREF="HTTP://WWW.DEEGREE.ORG" XLINK:ROLE=http://www.deegree.org
      XLINK:ARCROLE="HTTP://WWW.DEEGREE.ORG" XLINK:TITLE="DEEGREE WCS METADATA"
      XLINK:SHOW="NEW" XLINK:ACTUATE="ONLOAD"
      GML:REMOTESHEMA="HTTP://WWW.DEEGREE.ORG" ABOUT="HTTP://WWW.DEEGREE.ORG"
      METADATATYPE="TC211" />
    <DESCRIPTION>
```

DEEGREE WCS BEING OGC WCS 1.0.0 REFERENCE IMPLEMENTATION

```
</DESCRIPTION>
<NAME>DEEGREE WCS</NAME>
<LABEL>DEEGREE WCS</LABEL>
<KEYWORDS>
  <KEYWORD>DEEGREE</KEYWORD>
  <KEYWORD>WCS</KEYWORD>
  <TYPE CODESPACE="HTTP://WWW.DEEGREE.ORG">DEEGREE INTERNAL</TYPE>
</KEYWORDS>
<KEYWORDS>
  <KEYWORD>REFERENCE IMPLEMENATION</KEYWORD>
  <KEYWORD>WCS</KEYWORD>
  <TYPE CODESPACE="HTTP://WWW.DEEGREE.ORG">OGC</TYPE>
</KEYWORDS>
<RESPONSIBLEPARTY>
  <INDIVIDUALNAME>ANUP DESHMUKH</INDIVIDUALNAME>
  <ORGANISATIONNAME>IAIS FRAUNHOFER</ORGANISATIONNAME>
  <POSITIONNAME>STUDENT</POSITIONNAME>
  <CONTACTINFO>
    <PHONE>
      <VOICE>01799567509</VOICE>
      <FACSIMILE>01799567509</FACSIMILE>
    </PHONE>
    <ADDRESS>
      <DELIVERYPOINT>SCHLOSS BIRLINGHOVEN</DELIVERYPOINT>
      <CITY>SANKT AUGUSTIN</CITY>
      <ADMINISTRATIVEAREA>NRW</ADMINISTRATIVEAREA>
      <POSTALCODE>53000</POSTALCODE>
      <COUNTRY>GERMANY</COUNTRY>
      <ELECTRONICMAILADDRESS>anup.deshmukh@iais@fraunhofer.de
    </ELECTRONICMAILADDRESS>
    </ADDRESS>
    <ONLINERESOURCE XLINK:TYPE="SIMPLE" XLINK:HREF="HTTP://WWW.LAT-LON.DE"
      XLINK:ROLE="HTTP://WWW.LAT-LON.DE" XLINK:ARCROLE="HTTP://WWW.LAT-LON.DE"
      XLINK:TITLE="LAT/LON HOMEPAGE" XLINK:SHOW="NEW" XLINK:ACTUATE="ONLOAD" />
    </CONTACTINFO>
  </RESPONSIBLEPARTY>
  <FEES CODESPACE="HTTP://WWW.DEEGREE.ORG">NONE</FEES>
  <ACCESSCONSTRAINTS CODESPACE="HTTP://WWW.DEEGREE.ORG">NONE</ACCESSCONSTRAINTS>
```

```
<ACCESSCONSTRAINTS CODESPACE="HTTP://WWW.DEEGREE.ORG">SOME</ACCESSCONSTRAINTS>
</SERVICE>
<CAPABILITY>
  <REQUEST>
    <GETCAPABILITIES>
      <DCPTYPE>
        <HTTP>
          <GET>
            <ONLINERESOURCE XLINK:TYPE="SIMPLE"
              XLINK:HREF=http://127.0.0.1:8080/deegreewcs?
              XLINK:ROLE="HTTP://WWW.DEEGREE.ORG"
              XLINK:ARCROLE="HTTP://WWW.DEEGREE.ORG" XLINK:TITLE="STRING"
              XLINK:SHOW="NEW" XLINK:ACTUATE="ONLOAD" />
            </GET>
          <POST>
            <ONLINERESOURCE XLINK:TYPE="SIMPLE"
              XLINK:HREF=http://127.0.0.1:8080/deegreewcs?
              XLINK:ROLE="HTTP://WWW.DEEGREE.ORG"
              XLINK:ARCROLE="HTTP://WWW.DEEGREE.ORG" XLINK:TITLE="STRING"
              XLINK:SHOW="NEW" XLINK:ACTUATE="ONLOAD" />
          </POST>
        </HTTP>
      </DCPTYPE>
    </GETCAPABILITIES>
  <DESCRIBECOVERAGE>
    <DCPTYPE>
      <HTTP>
        <GET>
          <ONLINERESOURCE XLINK:TYPE="SIMPLE"
            XLINK:HREF=http://127.0.0.1:8080/deegreewcs?
            XLINK:ROLE="HTTP://WWW.DEEGREE.ORG"
            XLINK:ARCROLE="HTTP://WWW.DEEGREE.ORG" XLINK:TITLE="STRING"
            XLINK:SHOW="NEW" XLINK:ACTUATE="ONLOAD" />
          </GET>
        <POST>
          <ONLINERESOURCE XLINK:TYPE="SIMPLE"
            XLINK:HREF=http://127.0.0.1:8080/deegreewcs?
            XLINK:ROLE="HTTP://WWW.DEEGREE.ORG"
```



```
        XLINK:ARCRole="HTTP://WWW.DEEGREE.ORG" XLINK:TITLE="STRING"
        XLINK:SHOW="NEW" XLINK:ACTUATE="ONLOAD" />
    </POST>
</HTTP>
</DCPType>
</DESCRIBECOVERAGE>
<GETCOVERAGE>
    <DCPType>
        <HTTP>
            <GET>
                <ONLINERESOURCE XLINK:TYPE="SIMPLE"
                    XLINK:HREF=http://127.0.0.1:8080/deegreewcs?
                    XLINK:ROLE="HTTP://WWW.DEEGREE.ORG"
                    XLINK:ARCRole="HTTP://WWW.DEEGREE.ORG" XLINK:TITLE="STRING"
                    XLINK:SHOW="NEW" XLINK:ACTUATE="ONLOAD" />                </GET>
            <POST>
                <ONLINERESOURCE XLINK:TYPE="SIMPLE"
                    XLINK:HREF=http://127.0.0.1:8080/deegreewcs?
                    XLINK:ROLE="HTTP://WWW.DEEGREE.ORG"
                    XLINK:ARCRole="HTTP://WWW.DEEGREE.ORG" XLINK:TITLE="STRING"
                    XLINK:SHOW="NEW" XLINK:ACTUATE="ONLOAD" />
            </POST>
        </HTTP>
    </DCPType>
</GETCOVERAGE>
</REQUEST>
<EXCEPTION>
    <FORMAT>APPLICATION/VND.OGC.SE_XML</FORMAT>
    <FORMAT>APPLICATION/DEEGREE_XML</FORMAT>
</EXCEPTION>
<VENDORSPECIFICCAPABILITIES />
</CAPABILITY>
<CONTENTMETADATA XLINK:TYPE="SIMPLE" XLINK:HREF="HTTP://WWW.DEEGREE.ORG"
    XLINK:ROLE="HTTP://WWW.DEEGREE.ORG" XLINK:ARCRole="HTTP://WWW.DEEGREE.ORG/WCS"
    XLINK:TITLE="STRING" XLINK:SHOW="NEW" XLINK:ACTUATE="ONLOAD"
    GML:REMOTESHEMA="HTTP://WWW.DEEGREE.ORG" VERSION="1.0.0"
    UPDATESEQUENCE="1.0.0">
    <COVERAGEOFFERINGBRIEF GML:ID="ID000003">
        <METADATALINK XLINK:TYPE="SIMPLE" XLINK:HREF="HTTP://WWW.ALTOVA.COM"
```

```
    XLINK:ROLE="HTTP://WWW.ALTOVA.COM" XLINK:ARCROLE="HTTP://WWW.ALTOVA.COM"
    XLINK:TITLE="STRING" XLINK:SHOW="NEW" XLINK:ACTUATE="ONLOAD"
    GML:REMOTE_SCHEMA="HTTP://WWW.ALTOVA.COM" ABOUT="HTTP://WWW.ALTOVA.COM"
    METADATA_TYPE="TC211" />
<DESCRIPTION>NONE</DESCRIPTION>
<NAME>GERMANYCITIES</NAME>
<LABEL>GERMANY CITIES</LABEL>
<LONLATENVELOPE SRSNAME="WGS84(DD)">
    <GML:POS DIMENSION="2">5.87439,47.3102</GML:POS>
    <GML:POS DIMENSION="2">15.02907,55.01741</GML:POS>
</LONLATENVELOPE>
<KEYWORDS>
    <KEYWORD>GERMANY</KEYWORD>
    <KEYWORD>CITIES</KEYWORD>
    <KEYWORD>TEST DATA</KEYWORD>
    <KEYWORD>OLAP CUBEDATASET</KEYWORD>
</KEYWORDS>
<DEGREE:CONFIGURATION>
    ../MULTIPOINT_COVERAGE_DESCRIPTION.XML
</DEGREE:CONFIGURATION>
</COVERAGEOFFERINGBRIEF>
<COVERAGEOFFERINGBRIEF GML:ID="ID000001">
    <METADATALINK XLINK:TYPE="SIMPLE" XLINK:HREF="HTTP://WWW.ALTOVA.COM"
        XLINK:ROLE="HTTP://WWW.ALTOVA.COM" XLINK:ARCROLE="HTTP://WWW.ALTOVA.COM"
        XLINK:TITLE="STRING" XLINK:SHOW="NEW" XLINK:ACTUATE="ONLOAD"
        GML:REMOTE_SCHEMA="HTTP://WWW.ALTOVA.COM" ABOUT="HTTP://WWW.ALTOVA.COM"
        METADATA_TYPE="TC211" />
    <DESCRIPTION>NONE</DESCRIPTION>
    <NAME>GERMANYSTATES</NAME>
    <LABEL>GERMANY STATES</LABEL>
    <LONLATENVELOPE SRSNAME="WGS84(DD)">
        <GML:POS DIMENSION="2">5.87439,47.3102</GML:POS>
        <GML:POS DIMENSION="2">15.02907,55.01741</GML:POS>
    </LONLATENVELOPE>
    <KEYWORDS>
        <KEYWORD>GERMANY</KEYWORD>
        <KEYWORD>STATES</KEYWORD>
        <KEYWORD>TEST DATA</KEYWORD>
        <KEYWORD>OLAP CUBEDATASET</KEYWORD>
```

```
</KEYWORDS>
  <DEEGREE:CONFIGURATION>
    ../MULTISURFACE_COVERAGE_DESCRIPTION.XML
  </DEEGREE:CONFIGURATION>
</COVERAGEOFFERINGBRIEF>
</CONTENTMETADATA>
</WCS_CAPABILITIES>
```

MULTIPOINT COVERAGE DESCRIPTION – GERMANY CITIES

```
<?XML VERSION="1.0" ENCODING="UTF-8"?>
<COVERAGEDESCRIPTION XMLNS="HTTP://WWW.OPENGIS.NET/WCS"
  XMLNS:GML=http://www.opengis.net/gml
  XMLNS:DEEGREE="HTTP://WWW.DEEGREE.ORG/WCS"
  XMLNS:OLAP="HTTP://WWW.FRAUNHOFER.DE/OLAP"
  XMLNS:XLINK="HTTP://WWW.W3.ORG/1999/XLINK" VERSION="1.0.0"
  UPDATESEQUENCE="STRING">
  <COVERAGEOFFERING>
    <DESCRIPTION>NONE</DESCRIPTION>
    <NAME>GERMANYCITIES</NAME>
    <LABEL>GERMANY CITIES</LABEL>
    <LONLATENVELOPE SRSNAME="WGS84(DD)">
      <GML:POS DIMENSION="2">5.87439,47.3102</GML:POS>
      <GML:POS DIMENSION="2">15.02907,55.01741</GML:POS>
    </LONLATENVELOPE>
    <KEYWORDS>
      <KEYWORD>GERMANY</KEYWORD>
      <KEYWORD>CITIES</KEYWORD>
      <KEYWORD>TEST DATA</KEYWORD>
    </KEYWORDS>
    <DOMAINSET>
      <SPATIALDOMAIN ID="APP:NAME">
        <GML:ENVELOPE SRSNAME="HTTP://WWW.OPENGIS.NET/GML/SRS/EPSG.XML#4326">
          <GML:POS DIMENSION="2">5.87439,47.3102</GML:POS>
          <GML:POS DIMENSION="2">15.02907,55.01741</GML:POS>
        </GML:ENVELOPE>
      </SPATIALDOMAIN>
    </DOMAINSET>
    <RANGESET>
```

```
<NAME>DEFAULT</NAME>
<LABEL>DEFAULT</LABEL>
</RANGESET>
<SUPPORTEDCRSS>
  <REQUESTRESPONSECRSS
    CODESPACE="HTTP://WWW.EPSG.ORG">EPSG:4326</REQUESTRESPONSECRSS>
  <NATIVECRSS CODESPACE="HTTP://WWW.EPSG.ORG">EPSG:4326</NATIVECRSS>
</SUPPORTEDCRSS>
<SUPPORTEDFORMATS NATIVEFORMAT="XML">
  <FORMATS>TEXT/XML</FORMATS>
</SUPPORTEDFORMATS>
<SUPPORTEDINTERPOLATIONS DEFAULT="NEAREST NEIGHBOR">
  <INTERPOLATIONMETHOD>NEAREST NEIGHBOR</INTERPOLATIONMETHOD>
</SUPPORTEDINTERPOLATIONS>
<DEEGREE:EXTENSION TYPE="MULTIPOINTINDEXED">
  <DEEGREE:RESOLUTION MAX="99999999" MIN="0">
    <DEEGREE:RANGE>
      <DEEGREE:NAME>DEFAULT</DEEGREE:NAME>
    </DEEGREE:RANGE>
    <DEEGREE:SHAPE TILEPROPERTY="FILENAME" DIRECTORYPROPERTY="FOLDER"
      SRSNAME="EPSG:4326">ORTE/CITIES</DEEGREE:SHAPE>
    </DEEGREE:RESOLUTION>
  </DEEGREE:EXTENSION>
<OLAP:EXTENSION>
  <OLAP:DATASTORE>MSSQL-OLAP</OLAP:DATASTORE>
  <OLAP:OLAPCONNECTION>
    <OLAP:SERVERURL>HTTP://MSSPADE/XMLA/MSXISAPI.DLL</OLAP:SERVERURL>
    <OLAP:RESTRICTIONLIST>
      <OLAP:CUBE>DPA</OLAP:CUBE>
    </OLAP:RESTRICTIONLIST>
    <OLAP:PROPERTYLIST>
      <OLAP:DATASOURCEINFO>LOCAL ANALYSIS SERVER</OLAP:DATASOURCEINFO>
      <OLAP:CATALOG>DEUTSCHEPRESSEAGENTUR</OLAP:CATALOG>
      <OLAP:FORMAT>TABULAR</OLAP:FORMAT>
    </OLAP:PROPERTYLIST>
  </OLAP:OLAPCONNECTION>
</OLAP:EXTENSION>
</COVERAGEOFFERING>
</COVERAGEDescription>
```

MULTISURFACE COVERAGE DESCRIPTION – GERMANY STATES

```
<?XML VERSION="1.0" ENCODING="UTF-8"?>
<COVERAGEDESCRIPTION XMLNS="HTTP://WWW.OPENGIS.NET/WCS"
  XMLNS:GML=http://www.opengis.net/gml
  XMLNS:DEEGREE="HTTP://WWW.DEEGREE.ORG/WCS"
  XMLNS:OLAP="HTTP://WWW.FRAUNHOFER.DE/OLAP"
  XMLNS:XLINK="HTTP://WWW.W3.ORG/1999/XLINK" VERSION="1.0.0"
  UPDATESEQUENCE="STRING">
<COVERAGEOFFERING>
  <DESCRIPTION>NONE</DESCRIPTION>
  <NAME>GERMANYSTATES</NAME>
  <LABEL>GERMANY STATES</LABEL>
  <LONLATENVELOPE SRSNAME="WGS84(DD)">
    <GML:POS DIMENSION="2">5.87439,47.3102</GML:POS>
    <GML:POS DIMENSION="2">15.02907,55.01741</GML:POS>
  </LONLATENVELOPE>
  <KEYWORDS>
    <KEYWORD>GERMANY</KEYWORD>
    <KEYWORD>STATES</KEYWORD>
    <KEYWORD>TEST DATA</KEYWORD>
  </KEYWORDS>
  <DOMAINSET>
    <SPATIALDOMAIN ID="APP:NAME">
      <GML:ENVELOPE SRSNAME="HTTP://WWW.OPENGIS.NET/GML/SRS/EPSSG.XML#4326">
        <GML:POS DIMENSION="2">5.87439,47.3102</GML:POS>
        <GML:POS DIMENSION="2">15.02907,55.01741</GML:POS>
      </GML:ENVELOPE>
    </SPATIALDOMAIN>
  </DOMAINSET>
  <RANGESSET>
    <NAME>DEFAULT</NAME>
    <LABEL>DEFAULT</LABEL>
  </RANGESSET>
  <SUPPORTEDCRSS>
    <REQUESTRESPONSECRSS
      CODESPACE="HTTP://WWW.EPSSG.ORG">EPSSG:4326</REQUESTRESPONSECRSS>
    <NATIVECRSS CODESPACE="HTTP://WWW.EPSSG.ORG">EPSSG:4326</NATIVECRSS>
  </SUPPORTEDCRSS>
```

```
<SUPPORTEDFORMATS NATIVEFORMAT="XML">
  <FORMATS>TEXT/XML</FORMATS>
</SUPPORTEDFORMATS>
<SUPPORTEDINTERPOLATIONS DEFAULT="NEAREST NEIGHBOR">
  <INTERPOLATIONMETHOD>NEAREST NEIGHBOR</INTERPOLATIONMETHOD>
</SUPPORTEDINTERPOLATIONS>
<DEEGREE:EXTENSION TYPE="MULTISURFACEINDEXED">
  <DEEGREE:RESOLUTION MAX="99999999" MIN="0">
    <DEEGREE:RANGE>
      <DEEGREE:NAME>DEFAULT</DEEGREE:NAME>
    </DEEGREE:RANGE>
    <DEEGREE:SHAPE TILEPROPERTY="FILENAME" DIRECTORYPROPERTY="FOLDER"
      SRSNAME="EPSG:4326">BUNDESLAND/D_LAND05_REGION</DEEGREE:SHAPE>
    </DEEGREE:RESOLUTION>
  </DEEGREE:EXTENSION>
<OLAP:EXTENSION>
  <OLAP:DATASTORE>MSSQL-OLAP</OLAP:DATASTORE>
  <OLAP:OLAPCONNECTION>
    <OLAP:SERVERURL>HTTP://MSSPADE/XMLA/MSXISAPI.DLL</OLAP:SERVERURL>
    <OLAP:RESTRICTIONLIST>
    <OLAP:CUBE>DPA</OLAP:CUBE>
    </OLAP:RESTRICTIONLIST>
    <OLAP:PROPERTYLIST>
      <OLAP:DATASOURCEINFO>LOCAL ANALYSIS SERVER</OLAP:DATASOURCEINFO>
      <OLAP:CATALOG>DEUTSCHEPRESSEAGENTUR</OLAP:CATALOG>
      <OLAP:FORMAT>TABULAR</OLAP:FORMAT>
    </OLAP:PROPERTYLIST>
  </OLAP:OLAPCONNECTION>
</OLAP:EXTENSION>
</COVERAGEOFFERING>
</COVERAGEDescription>
```

10. APPENDIX C – EXAMPLE REQUESTS, RESPONSES

This appendix provides examples of WCS operation requests and responses used in the prototype scenario testing phase mentioned in Chapter 5.

SCENARIO 1: ALL THE CRIMINAL CASES IN BONN AND SURROUNDINGS FOR THE YEAR 2005.

REQUEST DOCUMENT

```
<GETCOVERAGE XMLNS="HTTP://WWW.OPENGIS.NET/WCS"
  XMLNS:WCS=http://www.opengis.net/wcs XMLNS:OWS="HTTP://WWW.OPENGIS.NET/OWS"
  XMLNS:OWCS="HTTP://WWW.OPENGIS.NET/WCS/1.1/OWS"
  XMLNS:GML="HTTP://WWW.OPENGIS.NET/GML" SERVICE="WCS" VERSION="1.0.0">
<WCS:SOURCECOVERAGE>GERMANYCITIES</WCS:SOURCECOVERAGE>
<WCS:DOMAINSUBSET>
  <WCS:SPATIALSUBSET>
    <GML:ENVELOPE SRSNAME='EPSG:4326'>
      <GML:POS DIMENSION='2'>6.95204,50.65149</GML:POS>
      <GML:POS DIMENSION='2'>7.24603,50.82822</GML:POS>
    </GML:ENVELOPE>
  </WCS:SPATIALSUBSET>
  <WCS:TEMPORALSUBSET CLAUSE="SLICE">
    <WCS:TIMEPERIOD>
      <WCS:BEGINPOSITION CALENDARERANAME="AC" FRAME="#ISO-8601"
        INDETERMINATEPOSITION="NOW">
        2005-01-01T00:00:00-05:00
      </WCS:BEGINPOSITION>
      <WCS:ENDPOSITION CALENDARERANAME="AC" FRAME="#ISO-8601"
        INDETERMINATEPOSITION="NOW">
        2005-12-14T00:00:00-05:00
      </WCS:ENDPOSITION>
      <WCS:TIMERESOLUTION>P1D</WCS:TIMERESOLUTION>
    </WCS:TIMEPERIOD>
  </WCS:TEMPORALSUBSET>
</WCS:DOMAINSUBSET>
<WCS:RANGESUBSET>
  <WCS:FIELDSUBSET>
```

```
<OWCS:IDENTIFIER CLAUSE="AXIS(0)">[REPORT LOCATION]</OWCS:IDENTIFIER>
<WCS:AXISSUBSET>
  <OWCS:IDENTIFIER>[REPORT LOCATION]</OWCS:IDENTIFIER>
  <WCS:KEY>[CITY NAME]</WCS:KEY>
</WCS:AXISSUBSET>
</WCS:FIELDSUBSET>
<WCS:FIELDSUBSET>
  <OWCS:IDENTIFIER CLAUSE="AXIS(1)">[REPORT SUBJECT]</OWCS:IDENTIFIER>
  <WCS:AXISSUBSET>
    <OWCS:IDENTIFIER>[REPORT SUBJECT]</OWCS:IDENTIFIER>
    <WCS:KEY>DETAILID=02001000</WCS:KEY>
  </WCS:AXISSUBSET>
</WCS:FIELDSUBSET>
</WCS:RANGESUBSET>
<WCS:OUTPUT>
  <WCS:CRS>EPSG:4326</WCS:CRS>
  <WCS:FORMAT>TEXT/XML</WCS:FORMAT>
</WCS:OUTPUT>
</GETCOVERAGE>
```

MAPPED XMLA REQUEST DOCUMENT

```
<?XML VERSION="1.0" ENCODING="UTF-8"?>
<SOAP-ENV:ENVELOPE
  xmlns:SOAP-ENV="HTTP://SCHEMAS.XMLSOAP.ORG/SOAP/ENVELOPE/"
  SOAP-ENV:ENCODINGSTYLE="HTTP://SCHEMAS.XMLSOAP.ORG/SOAP/ENCODING/">
  <SOAP-ENV:BODY>
    <EXECUTE xmlns="URN:SCHEMAS-MICROSOFT-COM:XML-ANALYSIS"
      SOAP-ENV:ENCODINGSTYLE="HTTP://SCHEMAS.XMLSOAP.ORG/SOAP/ENCODING/">
      <COMMAND>
        <STATEMENT>
          <SOAP-ENV:ENVELOPE
            SOAP-ENV:ENCODINGSTYLE="HTTP://SCHEMAS.XMLSOAP.ORG/SOAP/ENCODING/">
            <SOAP-ENV:BODY>
              <EXECUTE
                SOAP-ENV:ENCODINGSTYLE="HTTP://SCHEMAS.XMLSOAP.ORG/SOAP/ENCODING/">
                <COMMAND>WITH MEMBER [REPORT TIME].[TIME RANGE] AS 'COUNT( [REPORT
TIME].[2005].[JANUAR].[1] : [REPORT TIME].[2005].[DEZEMBER].[14])'
```



```
SELECT NON EMPTY { [REPORT LOCATION].[CITY NAME].[WESSELING], [REPORT LOCATION].[CITY
NAME].[BONN], [REPORT LOCATION].[CITY NAME].[BORNHEIM], [REPORT LOCATION].[CITY
NAME].[KÖNIGSWINTER], [REPORT LOCATION].[CITY NAME].[SIEGBURG], [REPORT LOCATION].[CITY
NAME].[SANKT AUGUSTIN], [REPORT LOCATION].[CITY NAME].[TROISDORF] } ON AXIS(0),
NON EMPTY { [REPORT SUBJECT].[MATTERID].[02001000] } ON AXIS(1) FROM DPA
WHERE ( [MEASURES].[COUNT] ) </COMMAND>

    </EXECUTE>
</SOAP-ENV:BODY>
</SOAP-ENV:ENVELOPE>
</STATEMENT>
</COMMAND>
<PROPERTIES>
    <PROPERTYLIST>
        <CATALOG>DEUTSCHEPRESSEAGENTUR</CATALOG>
        <DATASOURCEINFO>LOCAL ANALYSIS SERVER</DATASOURCEINFO>
        <FORMAT>MULTIDIMENSIONAL</FORMAT>
    </PROPERTYLIST>
</PROPERTIES>
</EXECUTE>
</SOAP-ENV:BODY>
</SOAP-ENV:ENVELOPE>
```

RESPONSE DOCUMENT

```
<GERMANYCITIES XMLNS:GML="HTTP://WWW.OPENGIS.NET/GML">
    <GML:DOMAINSET>
        <GML:MULTIPOINT ID="WESSELING">
            <GML:POINTMEMBERS>
                <GML:POINT>
                    <GML:POS>6.97313 50.82616</GML:POS>
                </GML:POINT>
            </GML:POINTMEMBERS>
        </GML:MULTIPOINT>
        <GML:MULTIPOINT ID="BONN">
            <GML:POINTMEMBERS>
                <GML:POINT>
                    <GML:POS>7.09602 50.73737</GML:POS>
                </GML:POINT>
            </GML:POINTMEMBERS>
```

```
</GML:MultiPoint>
<GML:MultiPoint ID="BORNHEIM">
  <GML:PointMembers>
    <GML:Point>
      <GML:pos>6.98684 50.77066</GML:pos>
    </GML:Point>
  </GML:PointMembers>
</GML:MultiPoint>
<GML:MultiPoint ID="KÖNIGSWINTER">
  <GML:PointMembers>
    <GML:Point>
      <GML:pos>7.1928 50.67958</GML:pos>
    </GML:Point>
  </GML:PointMembers>
</GML:MultiPoint>
<GML:MultiPoint ID="SIEGBURG">
  <GML:PointMembers>
    <GML:Point>
      <GML:pos>7.2043 50.79396</GML:pos>
    </GML:Point>
  </GML:PointMembers>
</GML:MultiPoint>
<GML:MultiPoint ID="SANKT AUGUSTIN">
  <GML:PointMembers>
    <GML:Point>
      <GML:pos>7.18271 50.76756</GML:pos>
    </GML:Point>
  </GML:PointMembers>
</GML:MultiPoint>
<GML:MultiPoint ID="TROISDORF">
  <GML:PointMembers>
    <GML:Point>
      <GML:pos>7.14822 50.81376</GML:pos>
    </GML:Point>
  </GML:PointMembers>
</GML:MultiPoint>
</GML:DOMAINSET>
<GML:RANGESet>
  <GML:DATABLOCK>
```

```
<GML:RANGEPARAMETERS>
  <GML:COMPOSITEVALUE>
    <GML:VALUECOMPONENTS>
      <AXES>
        <AXIS NAME="AXIS0">
          <TUPLES>
            <TUPLE>
              <MEMBER HIERARCHY="REPORT LOCATION">
                <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[NORDRHEIN-
WESTFALEN].[BONN]</UNAME>
                <CAPTION>BONN</CAPTION>
                <LNAME>[REPORT LOCATION].[CITY NAME]</LNAME>
                <LNUM>2</LNUM>
                <DISPLAYINFO>131072</DISPLAYINFO>
              </MEMBER>
              <MEMBER HIERARCHY="REPORT LOCATION">
                <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[NORDRHEIN-
WESTFALEN].[SIEGBURG]</UNAME>
                <CAPTION>SIEGBURG</CAPTION>
                <LNAME>[REPORT LOCATION].[CITY NAME]</LNAME>
                <LNUM>2</LNUM>
                <DISPLAYINFO>131072</DISPLAYINFO>
              </MEMBER>
              <MEMBER HIERARCHY="REPORT LOCATION">
                <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[NORDRHEIN-
WESTFALEN].[TROIJDORF]</UNAME>
                <CAPTION>TROIJDORF</CAPTION>
                <LNAME>[REPORT LOCATION].[CITY NAME]</LNAME>
                <LNUM>2</LNUM>
                <DISPLAYINFO>131072</DISPLAYINFO>
              </MEMBER>
            </TUPLE>
          </TUPLES>
        </AXIS>
        <AXIS NAME="AXIS1">
          <TUPLES>
            <TUPLE>
              <MEMBER HIERARCHY="REPORT SUBJECT">
```

```
<UNAME>[REPORT SUBJECT].[ALLE REPORT  
SUBJECT].[02000000].[02001000]</UNAME>  
<CAPTION>02001000</CAPTION>  
<LNAME>[REPORT SUBJECT].[MATTERID]</LNAME>  
<LNUM>2</LNUM>  
<DISPLAYINFO>10</DISPLAYINFO>  
</MEMBER>  
</TUPLE>  
</TUPLES>  
</AXIS>  
<AXIS NAME="SLICERAXIS">  
<TUPLES>  
<TUPLE>  
<MEMBER HIERARCHY="MEASURES">  
<UNAME>[MEASURES].[COUNT]</UNAME>  
<CAPTION>COUNT</CAPTION>  
<LNAME>[MEASURES].[MEASURESLEVEL]</LNAME>  
<LNUM>0</LNUM>  
<DISPLAYINFO>0</DISPLAYINFO>  
</MEMBER>  
<MEMBER HIERARCHY="REPORT TIME">  
<UNAME>[REPORT TIME].[ALLE REPORT TIME]</UNAME>  
<CAPTION>ALLE REPORT TIME</CAPTION>  
<LNAME>[REPORT TIME].[GESAMT]</LNAME>  
<LNUM>0</LNUM>  
<DISPLAYINFO>4</DISPLAYINFO>  
</MEMBER>  
</TUPLE>  
</TUPLES>  
</AXIS>  
</AXES>  
</GML:VALUECOMPONENTS>  
</GML:COMPOSITEVALUE>  
</GML:RANGEPARAMETERS>  
<GML:DOUBLEORNULLTUPLELIST>41 3 3</GML:DOUBLEORNULLTUPLELIST>  
</GML:DATABLOCK>  
</GML:RANGESET>  
</GERMANYCITIES>
```

SCENARIO 2: ALL CRIMINAL CASES IN GERMANY FOR THE TIME PERIOD 2003 TO 2005**REQUEST DOCUMENT**

```
<GETCOVERAGE XMLNS="HTTP://WWW.OPENGIS.NET/WCS"
  XMLNS:WCS=http://www.opengis.net/wcs XMLNS:OWS="HTTP://WWW.OPENGIS.NET/OWS"
  XMLNS:OWCS="HTTP://WWW.OPENGIS.NET/WCS/1.1/OWS"
  XMLNS:GML="HTTP://WWW.OPENGIS.NET/GML" SERVICE="WCS" VERSION="1.0.0">
  <WCS:SOURCECOVERAGE>GERMANYSTATES</WCS:SOURCECOVERAGE>
  <WCS:DOMAINSUBSET>
    <WCS:SPATIALSUBSET>
      <GML:ENVELOPE SRSNAME='EPSG:4326'>
        <GML:POS DIMENSION='2'>4,47</GML:POS>
        <GML:POS DIMENSION='2'>8.52</GML:POS>
      </GML:ENVELOPE>
    </WCS:SPATIALSUBSET>
    <WCS:TEMPORALSUBSET CLAUSE="SLICE">
      <WCS:TIMEPERIOD>
        <WCS:BEGINPOSITION CALENDARERANAME="AC" FRAME="#ISO-8601"
          INDETERMINATEPOSITION="NOW">
          2003-01-01T00:00:00-05:00
        </WCS:BEGINPOSITION>
        <WCS:ENDPOSITION CALENDARERANAME="AC" FRAME="#ISO-8601"
          INDETERMINATEPOSITION="NOW">
          2005-12-14T00:00:00-05:00
        </WCS:ENDPOSITION>
        <WCS:TIMERESOLUTION>P1D</WCS:TIMERESOLUTION>
      </WCS:TIMEPERIOD>
    </WCS:TEMPORALSUBSET>
  </WCS:DOMAINSUBSET>
  <WCS:RANGESUBSET>
    <WCS:FIELDSUBSET>
      <OWCS:IDENTIFIER CLAUSE="AXIS(0)">[REPORT LOCATION]</OWCS:IDENTIFIER>
      <WCS:AXISSUBSET>
        <OWCS:IDENTIFIER>[REPORT LOCATION]</OWCS:IDENTIFIER>
        <WCS:KEY>[STATE NAME]</WCS:KEY>
      </WCS:AXISSUBSET>
    </WCS:FIELDSUBSET>
  <WCS:FIELDSUBSET>
```

```
<OWCS:IDENTIFIER CLAUSE="AXIS(1)">[REPORT SUBJECT]</OWCS:IDENTIFIER>
<WCS:AXISSUBSET>
  <OWCS:IDENTIFIER>[REPORT SUBJECT]</OWCS:IDENTIFIER>
  <WCS:KEY>DETAILID=02001000</WCS:KEY>
</WCS:AXISSUBSET>
</WCS:FIELDSUBSET>
</WCS:RANGESUBSET>
<WCS:OUTPUT>
  <WCS:CRS>EPSG:4326</WCS:CRS>
  <WCS:FORMAT>TEXT/XML</WCS:FORMAT>
</WCS:OUTPUT>
</GETCOVERAGE>
```

MAPPED XMLA REQUEST DOCUMENT

```
<?XML VERSION="1.0" ENCODING="UTF-8"?>
<SOAP-ENV:ENVELOPE
  xmlns:SOAP-ENV="HTTP://SCHEMAS.XMLSOAP.ORG/SOAP/ENVELOPE/"
  SOAP-ENV:ENCODINGSTYLE="HTTP://SCHEMAS.XMLSOAP.ORG/SOAP/ENCODING/">
  <SOAP-ENV:BODY>
    <EXECUTE xmlns="URN:SCHEMAS-MICROSOFT-COM:XML-ANALYSIS"
      SOAP-ENV:ENCODINGSTYLE="HTTP://SCHEMAS.XMLSOAP.ORG/SOAP/ENCODING/">
      <COMMAND>
        <STATEMENT>
          <SOAP-ENV:ENVELOPE
            SOAP-ENV:ENCODINGSTYLE="HTTP://SCHEMAS.XMLSOAP.ORG/SOAP/ENCODING/">
            <SOAP-ENV:BODY>
              <EXECUTE
                SOAP-ENV:ENCODINGSTYLE="HTTP://SCHEMAS.XMLSOAP.ORG/SOAP/ENCODING/">
                <COMMAND>WITH MEMBER [REPORT TIME].[TIME RANGE] AS 'COUNT( [REPORT
TIME].[2005].[MAI].[1] : [REPORT TIME].[2005].[MAI].[11])'
SELECT NON EMPTY { [REPORT LOCATION].[CITY NAME].[BONN], [REPORT LOCATION].[CITY
NAME].[KÖLN] } ON AXIS(0),
NON EMPTY { [REPORT SUBJECT].[MATTERID].[02001000] } ON AXIS(1) FROM DPA
WHERE ( [MEASURES].[COUNT] ) </COMMAND>
              </EXECUTE>
            </SOAP-ENV:BODY>
          </SOAP-ENV:ENVELOPE>
        </STATEMENT>
```

```
</COMMAND>
<PROPERTIES>
  <PROPERTYLIST>
    <CATALOG>DEUTSCHEPRESSEAGENTUR</CATALOG>
    <DATASOURCEINFO>LOCAL ANALYSIS SERVER</DATASOURCEINFO>
    <FORMAT>MULTIDIMENSIONAL</FORMAT>
  </PROPERTYLIST>
</PROPERTIES>
</EXECUTE>
</SOAP-ENV:BODY>
</SOAP-ENV:ENVELOPE>
```

RESPONSE DOCUMENT

The spatial information has not been included in this response document. The geometry information retrieved for all surfaces in Germany is too large to include here.

```
<GERMANYSTATES XMLNS:GML="HTTP://WWW.OPENGIS.NET/GML">
  <GML:DOMAINSET>
    <GML:MULTISURFACE ID="SCHLESWIG-HOLSTEIN">
      <GML:SURFACEMEMBERS>
        <GML:SURFACE>
          <GML:PATCHES>
            <GML:POLYGONPATCH>
              <GML:EXTERIOR>
                <GML:LINEARRING>
                  ...
                </GML:LINEARRING>
              </GML:EXTERIOR>
            </GML:POLYGONPATCH>
          </GML:PATCHES>
        </GML:SURFACE>
        ...
      </GML:SURFACEMEMBERS>
    </GML:MULTISURFACE>
  </GML:DOMAINSET>
  <GML:RANGESSET>
    <GML:DATABLOCK>
      <GML:RANGEPARAMETERS>
```

```
<GML:COMPOSITEVALUE>
  <GML:VALUECOMPONENTS>
    <AXES>
      <AXIS NAME="AXIS0">
        <TUPLES>
          <TUPLE>
            <MEMBER HIERARCHY="REPORT LOCATION">
              <UNAME>
                [REPORT LOCATION].[ALLE REPORT LOCATION].[SCHLESWIG-HOLSTEIN]
              </UNAME>
              <CAPTION>SCHLESWIG-HOLSTEIN</CAPTION>
              <LNAME>[REPORT LOCATION].[STATE NAME]</LNAME>
              <LNUM>1</LNUM>
              <DISPLAYINFO>132072</DISPLAYINFO>
            </MEMBER>
            <MEMBER HIERARCHY="REPORT LOCATION">
              <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[HAMBURG]</UNAME>
              <CAPTION>HAMBURG</CAPTION>
              <LNAME>[REPORT LOCATION].[STATE NAME]</LNAME>
              <LNUM>1</LNUM>
              <DISPLAYINFO>132072</DISPLAYINFO>
            </MEMBER>
            <MEMBER HIERARCHY="REPORT LOCATION">
              <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[NIEDERSACHSEN]
              </UNAME>
              <CAPTION>NIEDERSACHSEN</CAPTION>
              <LNAME>[REPORT LOCATION].[STATE NAME]</LNAME>
              <LNUM>1</LNUM>
              <DISPLAYINFO>132072</DISPLAYINFO>
            </MEMBER>
            <MEMBER HIERARCHY="REPORT LOCATION">
              <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[BREMEN]</UNAME>
              <CAPTION>BREMEN</CAPTION>
              <LNAME>[REPORT LOCATION].[STATE NAME]</LNAME>
              <LNUM>1</LNUM>
              <DISPLAYINFO>132072</DISPLAYINFO>
            </MEMBER>
            <MEMBER HIERARCHY="REPORT LOCATION">
              <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[NORDRHEIN-WESTFALEN]
```



```
</UNAME>
<CAPTION>NORDRHEIN-WESTFALEN</CAPTION>
<LNAME>[REPORT LOCATION].[STATE NAME]</LNAME>
<LNUM>1</LNUM>
<DISPLAYINFO>131341</DISPLAYINFO>
</MEMBER>
<MEMBER HIERARCHY="REPORT LOCATION">
  <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[HESSEN]</UNAME>
  <CAPTION>HESSEN</CAPTION>
  <LNAME>[REPORT LOCATION].[STATE NAME]</LNAME>
  <LNUM>1</LNUM>
  <DISPLAYINFO>131285</DISPLAYINFO>
</MEMBER>
<MEMBER HIERARCHY="REPORT LOCATION">
  <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[RHEINLAND-PFALZ]
</UNAME>
  <CAPTION>RHEINLAND-PFALZ</CAPTION>
  <LNAME>[REPORT LOCATION].[STATE NAME]</LNAME>
  <LNUM>1</LNUM>
  <DISPLAYINFO>132072</DISPLAYINFO>
</MEMBER>
<MEMBER HIERARCHY="REPORT LOCATION">
  <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[BADEN-WÜRTTEMBERG]
</UNAME>
  <CAPTION>BADEN-WÜRTTEMBERG</CAPTION>
  <LNAME>[REPORT LOCATION].[STATE NAME]</LNAME>
  <LNUM>1</LNUM>
  <DISPLAYINFO>132072</DISPLAYINFO>
</MEMBER>
<MEMBER HIERARCHY="REPORT LOCATION">
  <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[BAYERN]</UNAME>
  <CAPTION>BAYERN</CAPTION>
  <LNAME>[REPORT LOCATION].[STATE NAME]</LNAME>
  <LNUM>1</LNUM>
  <DISPLAYINFO>132072</DISPLAYINFO>
</MEMBER>
<MEMBER HIERARCHY="REPORT LOCATION">
  <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[SAARLAND]</UNAME>
  <CAPTION>SAARLAND</CAPTION>
```

```
<LNAME>[REPORT LOCATION].[STATE NAME]</LNAME>
<LNUM>1</LNUM>
<DISPLAYINFO>132072</DISPLAYINFO>
</MEMBER>
<MEMBER HIERARCHY="REPORT LOCATION">
  <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[BERLIN]</UNAME>
  <CAPTION>BERLIN</CAPTION>
  <LNAME>[REPORT LOCATION].[STATE NAME]</LNAME>
  <LNUM>1</LNUM>
  <DISPLAYINFO>132072</DISPLAYINFO>
</MEMBER>
<MEMBER HIERARCHY="REPORT LOCATION">
  <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[BRANDENBURG]</UNAME>
  <CAPTION>BRANDENBURG</CAPTION>
  <LNAME>[REPORT LOCATION].[STATE NAME]</LNAME>
  <LNUM>1</LNUM>
  <DISPLAYINFO>132072</DISPLAYINFO>
</MEMBER>
<MEMBER HIERARCHY="REPORT LOCATION">
  <UNAME>
    [REPORT LOCATION].[ALLE REPORT LOCATION].[MECKLENBURG-VORPOMMERN]
  </UNAME>
  <CAPTION>MECKLENBURG-VORPOMMERN</CAPTION>
  <LNAME>[REPORT LOCATION].[STATE NAME]</LNAME>
  <LNUM>1</LNUM>
  <DISPLAYINFO>132072</DISPLAYINFO>
</MEMBER>
<MEMBER HIERARCHY="REPORT LOCATION">
  <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[SACHSEN]</UNAME>
  <CAPTION>SACHSEN</CAPTION>
  <LNAME>[REPORT LOCATION].[STATE NAME]</LNAME>
  <LNUM>1</LNUM>
  <DISPLAYINFO>132072</DISPLAYINFO>
</MEMBER>
<MEMBER HIERARCHY="REPORT LOCATION">
  <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[SACHSEN-ANHALT]
  </UNAME>
  <CAPTION>SACHSEN-ANHALT</CAPTION>
  <LNAME>[REPORT LOCATION].[STATE NAME]</LNAME>
```

```
<LNUM>1</LNUM>
<DISPLAYINFO>132072</DISPLAYINFO>
</MEMBER>
<MEMBER HIERARCHY="REPORT LOCATION">
  <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[THÜRINGEN]</UNAME>
  <CAPTION>THÜRINGEN</CAPTION>
  <LNAME>[REPORT LOCATION].[STATE NAME]</LNAME>
  <LNUM>1</LNUM>
  <DISPLAYINFO>132072</DISPLAYINFO>
</MEMBER>
</TUPLE>
</TUPLES>
</AXIS>
<AXIS NAME="AXIS1">
  <TUPLES>
    <TUPLE>
      <MEMBER HIERARCHY="REPORT SUBJECT">
        <UNAME>[REPORT SUBJECT].[ALLE REPORT SUBJECT].[02000000].[02001000]
        </UNAME>
        <CAPTION>02001000</CAPTION>
        <LNAME>[REPORT SUBJECT].[MATTERID]</LNAME>
        <LNUM>2</LNUM>
        <DISPLAYINFO>1000</DISPLAYINFO>
      </MEMBER>
    </TUPLE>
  </TUPLES>
</AXIS>
<AXIS NAME="SLICERAXIS">
  <TUPLES>
    <TUPLE>
      <MEMBER HIERARCHY="MEASURES">
        <UNAME>[MEASURES].[COUNT]</UNAME>
        <CAPTION>COUNT</CAPTION>
        <LNAME>[MEASURES].[MEASURESLEVEL]</LNAME>
        <LNUM>0</LNUM>
        <DISPLAYINFO>0</DISPLAYINFO>
      </MEMBER>
      <MEMBER HIERARCHY="REPORT TIME">
        <UNAME>[REPORT TIME].[ALLE REPORT TIME]</UNAME>
```

```
<CAPTION>ALLE REPORT TIME</CAPTION>
<LNAME>[REPORT TIME].[({GESAMT})]</LNAME>
<LNUM>0</LNUM>
<DISPLAYINFO>4</DISPLAYINFO>
</MEMBER>
</TUPLE>
</TUPLES>
</AXIS>
</AXES>
</GML:VALUECOMPONENTS>
</GML:COMPOSITEVALUE>
</GML:RANGEPARAMETERS>
<GML:DOUBLEORNULLTUPLELIST>358 829 945 94 1598 530 465 792 1649 195 867 1281 148
654 305 265
</GML:DOUBLEORNULLTUPLELIST>
</GML:DATABLOCK>
</GML:RANGESSET>
</GERMANYSTATES>
```

SCENARIO 3: ALL CRIMINAL CASES IN BONN AND KÖLN FOR THE MONTH OF MAY IN THE YEAR 2005.

REQUEST DOCUMENT

```
<GETCOVERAGE XMLNS="HTTP://WWW.OPENGIS.NET/WCS"
  XMLNS:WCS=http://www.opengis.net/wcs XMLNS:OWS="HTTP://WWW.OPENGIS.NET/OWS"
  XMLNS:OWCS="HTTP://WWW.OPENGIS.NET/WCS/1.1/OWS"
  XMLNS:GML="HTTP://WWW.OPENGIS.NET/GML" SERVICE="WCS" VERSION="1.0.0">
<WCS:SOURCECOVERAGE>GERMANYCITIES</WCS:SOURCECOVERAGE>
<WCS:DOMAINSUBSET>
  <WCS:SPATIALSUBSET>
    <GML:ENVELOPE SRSNAME='EPSG:4326'>
      <GML:POS DIMENSION='2'>6.5571,50.1005</GML:POS>
      <GML:POS DIMENSION='2'>7.7080,51.1586</GML:POS>
    </GML:ENVELOPE>
  </WCS:SPATIALSUBSET>
  <WCS:TEMPORALSUBSET CLAUSE="SLICE">
    <WCS:TIMEPERIOD>
      <WCS:BEGINPOSITION CALENDARERANAME="AC" FRAME="#ISO-8601">
```

```
INDETERMINATEPOSITION="NOW">
  2005-05-01T00:00:00-05:00
</WCS:BEGINPOSITION>
<WCS:ENDPOSITION CALENDARERANAME="AC" FRAME="#ISO-8601"
INDETERMINATEPOSITION="NOW">
  2005-05-11T00:00:00-05:00
</WCS:ENDPOSITION>
<WCS:TIMERESOLUTION>P1D</WCS:TIMERESOLUTION>
</WCS:TIMEPERIOD>
</WCS:TEMPORALSUBSET>
</WCS:DOMAINSUBSET>
<WCS:RANGESUBSET>
  <WCS:FIELDSUBSET>
    <OWCS:IDENTIFIER CLAUSE="AXIS(0)">[REPORT LOCATION]</OWCS:IDENTIFIER>
    <WCS:AXISSUBSET>
      <OWCS:IDENTIFIER>[REPORT LOCATION]</OWCS:IDENTIFIER>
      <WCS:KEY>[CITY NAME]=BONN</WCS:KEY>
      <WCS:KEY>[CITY NAME]=KÖLN</WCS:KEY>
    </WCS:AXISSUBSET>
  </WCS:FIELDSUBSET>
  <WCS:FIELDSUBSET>
    <OWCS:IDENTIFIER CLAUSE="AXIS(1)">[REPORT SUBJECT]</OWCS:IDENTIFIER>
    <WCS:AXISSUBSET>
      <OWCS:IDENTIFIER>[REPORT SUBJECT]</OWCS:IDENTIFIER>
      <WCS:KEY>DETAILID=02001000</WCS:KEY>
    </WCS:AXISSUBSET>
  </WCS:FIELDSUBSET>
</WCS:RANGESUBSET>
<WCS:OUTPUT>
  <WCS:CRS>EPSG:4326</WCS:CRS>
  <WCS:FORMAT>TEXT/XML</WCS:FORMAT>
</WCS:OUTPUT>
</GETCOVERAGE>
```

MAPPED XMLA REQUEST DOCUMENT

```
<?XML VERSION="1.0" ENCODING="UTF-8"?>
<SOAP-ENV:ENVELOPE
  XMLNS:SOAP-ENV="HTTP://SCHEMAS.XMLSOAP.ORG/SOAP/ENVELOPE/"
```

```
        SOAP-ENV:ENCODINGSTYLE="HTTP://SCHEMAS.XMLSOAP.ORG/SOAP/ENCODING/">
<SOAP-ENV:Body>
  <EXECUTE XMLNS="URN:SCHEMAS-MICROSOFT-COM:XML-ANALYSIS"
    SOAP-ENV:ENCODINGSTYLE="HTTP://SCHEMAS.XMLSOAP.ORG/SOAP/ENCODING/">
    <COMMAND>
      <STATEMENT>
        <SOAP-ENV:ENVELOPE
          SOAP-ENV:ENCODINGSTYLE="HTTP://SCHEMAS.XMLSOAP.ORG/SOAP/ENCODING/">
          <SOAP-ENV:Body>
            <EXECUTE
              SOAP-ENV:ENCODINGSTYLE="HTTP://SCHEMAS.XMLSOAP.ORG/SOAP/ENCODING/">
              <COMMAND>WITH MEMBER [REPORT TIME].[TIME RANGE] AS 'COUNT( [REPORT
TIME].[2003].[JANUAR].[1] : [REPORT TIME].[2005].[DEZEMBER].[14])'
SELECT NON EMPTY { [REPORT LOCATION].[STATE NAME].[SCHLESWIG-HOLSTEIN], [REPORT
LOCATION].[STATE NAME].[HAMBURG], [REPORT LOCATION].[STATE NAME].[NIEDERSACHSEN],
[REPORT LOCATION].[STATE NAME].[BREMEN], [REPORT LOCATION].[STATE NAME].[NORDRHEIN-
WESTFALEN], [REPORT LOCATION].[STATE NAME].[HESSEN], [REPORT LOCATION].[STATE
NAME].[RHEINLAND-PFALZ], [REPORT LOCATION].[STATE NAME].[BADEN-WÜRTTEMBERG], [REPORT
LOCATION].[STATE NAME].[BAYERN], [REPORT LOCATION].[STATE NAME].[SAARLAND], [REPORT
LOCATION].[STATE NAME].[BERLIN], [REPORT LOCATION].[STATE NAME].[BRANDENBURG], [REPORT
LOCATION].[STATE NAME].[MECKLENBURG-VORPOMMERN], [REPORT LOCATION].[STATE
NAME].[SACHSEN], [REPORT LOCATION].[STATE NAME].[SACHSEN-ANHALT], [REPORT
LOCATION].[STATE NAME].[THÜRINGEN] } ON AXIS(0),
NON EMPTY { [REPORT SUBJECT].[MATTERID].[02001000] } ON AXIS(1) FROM DPA
WHERE ( [MEASURES].[COUNT] ) </COMMAND>
            </EXECUTE>
          </SOAP-ENV:Body>
        </SOAP-ENV:ENVELOPE>
      </STATEMENT>
    </COMMAND>
  <PROPERTIES>
    <PROPERTYLIST>
      <CATALOG>DEUTSCHEPRESSEAGENTUR</CATALOG>
      <DATASOURCEINFO>LOCAL ANALYSIS SERVER</DATASOURCEINFO>
      <FORMAT>MULTIDIMENSIONAL</FORMAT>
    </PROPERTYLIST>
  </PROPERTIES>
</EXECUTE>
</SOAP-ENV:Body>
```

</SOAP-ENV:ENVELOPE>

RESPONSE DOCUMENT

```
<GERMANYCITIES XMLNS:GML="HTTP://WWW.OPENGIS.NET/GML">
  <GML:DOMAINSET>
    <GML:MULTIPOINT ID="LANGENFELD">
      <GML:POINTMEMBERS>
        <GML:POINT>
          <GML:POS>6.95913 51.11272</GML:POS>
        </GML:POINT>
      </GML:POINTMEMBERS>
    </GML:MULTIPOINT>
    <GML:MULTIPOINT ID="MONHEIM">
      <GML:POINTMEMBERS>
        <GML:POINT>
          <GML:POS>6.89445 51.09093</GML:POS>
        </GML:POINT>
      </GML:POINTMEMBERS>
    </GML:MULTIPOINT>
    <GML:MULTIPOINT ID="GREVENBROICH">
      <GML:POINTMEMBERS>
        <GML:POINT>
          <GML:POS>6.5674 51.09263</GML:POS>
        </GML:POINT>
      </GML:POINTMEMBERS>
    </GML:MULTIPOINT>
    <GML:MULTIPOINT ID="DORMAGEN">
      <GML:POINTMEMBERS>
        <GML:POINT>
          <GML:POS>6.81527 51.10382</GML:POS>
        </GML:POINT>
      </GML:POINTMEMBERS>
    </GML:MULTIPOINT>
    <GML:MULTIPOINT ID="LEICHLINGEN">
      <GML:POINTMEMBERS>
        <GML:POINT>
          <GML:POS>7.00963 51.10192</GML:POS>
        </GML:POINT>
      </GML:POINTMEMBERS>
    </GML:MULTIPOINT>
  </GML:DOMAINSET>
</GERMANYCITIES>
```

```
</GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="WERMELSKIRCHEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.2138 51.14123</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BERGHEIM">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.63388 50.96364</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="KERPEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.71737 50.88795</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="FRECHEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.81016 50.91135</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="PULHEIM">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.80456 50.99844</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BR?HL">
  <GML:POINTMEMBERS>
```



```
<GML:POINT>
  <GML:POS>6.89495 50.82746</GML:POS>
</GML:POINT>
</GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="H?RTH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.90565 50.87435</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="ERFTSTADT">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.76567 50.80076</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="WESSELING">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.97313 50.82616</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="KÖLN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.95673 50.94184</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="LEVERKUSEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.94584 51.05943</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
```

```
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BURSCHEID">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.11602 51.08483</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BERGISCH GLADBACH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.12532 50.95994</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="OVERATH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.2858 50.93184</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="RÖSRATH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.1815 50.9003</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="ODENTHAL">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.11691 51.02723</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="NÜMBRECHT">
  <GML:POINTMEMBERS>
    <GML:POINT>
```

```
<GML:POS>7.54147 50.90295</GML:POS>
</GML:POINT>
</GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="GUMMERSBACH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.55904 51.02453</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="WIEHL">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.54995 50.94984</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="WIPPERFÜRTH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.39467 51.11773</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BERGNEUSTADT">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.65324 51.02214</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="ENGELSKIRCHEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.40778 50.98754</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
```

```
<GML:MULTIPOINT ID="NÜRVENICH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.64608 50.80736</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BONN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.09602 50.73737</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BORNHEIM">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.98684 50.77066</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="MECKENHEIM">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.01864 50.62368</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="WACHTBERG">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.13942 50.60768</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="RHEINBACH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.94784 50.62518</GML:POS>
```

```
</GML:POINT>
</GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="REMAGEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.2296 50.56839</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BAD NEUENAHN-AHRWEILER">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.11362 50.54519</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="SINZIG">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.24911 50.54549</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="GRAFSCHAFT">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.06321 50.56869</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="ALTENAHN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.99194 50.518</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BERG">
```

```
<GML:POINTMEMBERS>
  <GML:POINT>
    <GML:POS>6.94534 50.55318</GML:POS>
  </GML:POINT>
</GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="DERNAU">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.04202 50.53599</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="MAYSCHO?">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.01854 50.52479</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="ADENAU">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.93164 50.382</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="NÜRBURG">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.95714 50.34561</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="REIFFERSCHIED">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.88706 50.4046</GML:POS>
    </GML:POINT>
```

```
</GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="LINZ">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.28349 50.56849</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BAD HÖNNINGEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.31369 50.51589</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="UNKEL">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.2187 50.60348</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BAD HONNEF">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.2313 50.64418</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="RHEINBREITBACH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.2294 50.61808</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="KÖNIGSWINTER">
  <GML:POINTMEMBERS>
```

```
<GML:POINT>
  <GML:POS>7.1928 50.67958</GML:POS>
</GML:POINT>
</GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="SIEGBURG">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.2043 50.79396</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="SANKT AUGUSTIN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.18271 50.76756</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="HENNEF">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.28229 50.77546</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="LOHMAR">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.21003 50.83737</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="MUCH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.40628 50.90565</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
```



```
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="TROISDORF">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.14822 50.81376</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="EUSKIRCHEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.78497 50.66088</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="MECHERNICH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.64559 50.59069</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BAD MÜNSTEREIFEL">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.76027 50.55698</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="WEILERSWIST">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.84146 50.75636</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="NETTERSHEIM">
  <GML:POINTMEMBERS>
    <GML:POINT>
```

```
<GML:POS>6.6281 50.49049</GML:POS>
</GML:POINT>
</GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="DAUN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.83036 50.19584</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="DREIS-BR?CK">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.80856 50.27462</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="GEROLSTEIN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.66528 50.22052</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="NEROTH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.74778 50.19603</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="J?NKERATH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.57209 50.34451</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
```

```
<GML:MULTIPOINT ID="KOBLENZ">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.60196 50.3534</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="LAHNSTEIN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.60532 50.30038</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BOPPARD">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.58595 50.22913</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="VALLENDAR">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.61186 50.39931</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="H?HR-GRENZHAUSEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.66863 50.43149</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="M?LHEIM-K?RLICH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.48987 50.38521</GML:POS>
```

```
</GML:POINT>
</GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BASSENHEIM">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.45556 50.35781</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="KALTENENGERS">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.54235 50.41651</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="NAUORT">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.62256 50.4651</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="TREIS-KARDEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.30188 50.17314</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="ISENBURG">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.58985 50.4773</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="KLEINMAISCHEID">
```

```
<GML:POINTMEMBERS>
  <GML:POINT>
    <GML:POS>7.60575 50.51079</GML:POS>
  </GML:POINT>
</GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="KARBACH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.61605 50.15843</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="HALSENBACH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.55115 50.17473</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="N?RTERSCHAUSEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.48237 50.22502</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BICKENBACH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.52605 50.12583</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="DERNBACH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.59425 50.54449</GML:POS>
    </GML:POINT>
```

```
</GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="LINKENBACH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.55176 50.57129</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="ST. GOAR">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.70445 50.15134</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="KOBERN-GONDORF">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.45168 50.30651</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="HATZENPORT">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.41207 50.22863</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="WOLKEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.46767 50.33371</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="WINNINGEN">
  <GML:POINTMEMBERS>
```

```
<GML:POINT>
  <GML:POS>7.51326 50.31182</GML:POS>
</GML:POINT>
</GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BRAUBACH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.65064 50.27122</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="KAMP-BORNHOFEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.61966 50.22773</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="NEUWIED">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.47716 50.43</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="WEI?ENTHURM">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.45678 50.40731</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="ANHAUSEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.55115 50.4975</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
```

```
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="WALDBREITBACH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.42107 50.54979</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="LEUTESDORF">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.37918 50.4557</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="ANDERNACH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.40107 50.4317</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="KRUF" >
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.33798 50.38411</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="OBERZISSEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.2004 50.4522</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="WEHR">
  <GML:POINTMEMBERS>
    <GML:POINT>
```



```
<GML:POS>7.2129 50.4209</GML:POS>
</GML:POINT>
</GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="MAYEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.22911 50.32582</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="ACHT">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.07142 50.36921</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="ANSCHAU">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.09263 50.29581</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BAAR">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.03582 50.34521</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="MENDIG">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.27239 50.3714</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
```

```
<GML:MULTIPOINT ID="POLCH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.31569 50.30012</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="KAISERSESCH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.13443 50.23082</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="ULMEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>6.97844 50.20753</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="COCHEM">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.16991 50.14594</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="BEILSTEIN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.23991 50.10923</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="B?CHEL">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.07933 50.17233</GML:POS>
```

```
</GML:POINT>
</GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="RODENBACH">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.64864 50.63898</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
<GML:MULTIPOINT ID="MEINERZHAGEN">
  <GML:POINTMEMBERS>
    <GML:POINT>
      <GML:POS>7.63655 51.10372</GML:POS>
    </GML:POINT>
  </GML:POINTMEMBERS>
</GML:MULTIPOINT>
</GML:DOMAINSET>
<GML:RANGESSET>
  <GML:DATABLOCK>
    <GML:RANGEPARAMETERS>
      <GML:COMPOSITEVALUE>
        <GML:VALUECOMPONENTS>
          <AXES>
            <AXIS NAME="AXIS0">
              <TUPLES>
                <TUPLE>
                  <MEMBER HIERARCHY="REPORT LOCATION">
                    <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[NORDRHEIN-
WESTFALEN].[BONN]</UNAME>
                    <CAPTION>BONN</CAPTION>
                    <LNAME>[REPORT LOCATION].[CITY NAME]</LNAME>
                    <LNUM>2</LNUM>
                    <DISPLAYINFO>131072</DISPLAYINFO>
                  </MEMBER>
                  <MEMBER HIERARCHY="REPORT LOCATION">
                    <UNAME>[REPORT LOCATION].[ALLE REPORT LOCATION].[NORDRHEIN-
WESTFALEN].[K?LN]</UNAME>
                    <CAPTION>K?LN</CAPTION>
```

```
<LNAME>[REPORT LOCATION].[CITY NAME]</LNAME>
<LNUM>2</LNUM>
<DISPLAYINFO>131072</DISPLAYINFO>
</MEMBER>
</TUPLE>
</TUPLES>
</AXIS>
<AXIS NAME="AXIS1">
  <TUPLES>
    <TUPLE>
      <MEMBER HIERARCHY="REPORT SUBJECT">
        <UNAME>[REPORT SUBJECT].[ALLE REPORT
SUBJECT].[02000000].[02001000]</UNAME>
        <CAPTION>02001000</CAPTION>
        <LNAME>[REPORT SUBJECT].[MATTERID]</LNAME>
        <LNUM>2</LNUM>
        <DISPLAYINFO>10</DISPLAYINFO>
      </MEMBER>
    </TUPLE>
  </TUPLES>
</AXIS>
<AXIS NAME="SLICERAXIS">
  <TUPLES>
    <TUPLE>
      <MEMBER HIERARCHY="MEASURES">
        <UNAME>[MEASURES].[COUNT]</UNAME>
        <CAPTION>COUNT</CAPTION>
        <LNAME>[MEASURES].[MEASURESLEVEL]</LNAME>
        <LNUM>0</LNUM>
        <DISPLAYINFO>0</DISPLAYINFO>
      </MEMBER>
      <MEMBER HIERARCHY="REPORT TIME">
        <UNAME>[REPORT TIME].[ALLE REPORT TIME]</UNAME>
        <CAPTION>ALLE REPORT TIME</CAPTION>
        <LNAME>[REPORT TIME].[GESAMT]</LNAME>
        <LNUM>0</LNUM>
        <DISPLAYINFO>4</DISPLAYINFO>
      </MEMBER>
    </TUPLE>
```

```
</TUPLES>
</AXIS>
</AXES>
</GML:VALUECOMPONENTS>
</GML:COMPOSITEVALUE>
</GML:RANGEPARAMETERS>
<GML:DOUBLEORNULLTUPLELIST>41 317</GML:DOUBLEORNULLTUPLELIST>
</GML:DATABLOCK>
</GML:RANGESSET>
</GERMANYCITIES>
```

