Using Open Standards for Retrieval of Multi-Dimensional Raster and Point Cloud Data (system demonstration)

Dimitar Misev, Vaibhav Bajpai, Peter Baumann

School of Engineering and Science
Jacobs University Bremen, Germany
Campus Ring 1, 28759 Bremen
{d.misev,v.bajpai,p.baumann}@jacobs-university.de

Abstract. The OGC Web Coverage Service (WCS) standard defines open interfaces for accessing and processing of raster data, more generally: coverages. Recently revised to WCS 2.0, the standard focusses to make coverages interchangeable across all OGC-based services and is based on Geography Markup Language (GML) 3.2.1, with a small, backwards compatible addition to achieve informational completeness. WCS 2.0 offers several advantages over previous versions, such as: support for general n-D raster data and non-raster coverage types; crisp, modular, and easy to understand; flexible and adaptive; harmonized with GML and Sensor Web Enablement (SWE); improved testability; and allows for efficient and scalable implementations. In this contribution, we demonstrate a fully-fledged, scalable, open-source reference implementation of the WCS 2.0 standard.

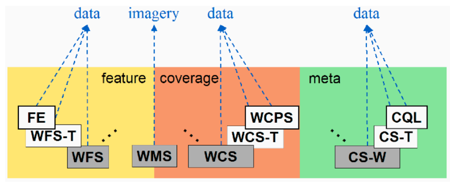
1 Introduction

A coverage, in OGC (Open Geospatial Consortium) and ISO (International Organization for Standardization) nomenclature, is defined as a space-time varying phenomenon [1]. Raster data are common representatives of coverages, but irregular grids, point clouds, and meshes also likewise constitute coverages.

The OGC Web Coverage Service (WCS) specification defines open interfaces for accessing and server-side processing of coverages [14]. Figure 1 shows how WCS embeds itself into the corresponding triad of access services consisting of Web Feature Service (WFS), WCS, and Catalog Service (CSW). The Web Coverage Processing Service (WCPS) allows raster coverage filtering and processing [8], OGC Common Query Language (CQL) offers SQL-style metadata retrieval and WCS-T, a WCS extension like WCPS, defines an open interface for manipulating coverages [26]

WCS 2.0 released in August, 2010 is a completely overhauled version with no longer a single standard but a modular, structured suite of specifications fitting into an extensible overall concept. Technically, it is based on Geography Markup Language (GML) [18] and Sensor Web Enablement (SWE) Common [19] and, hence, achieves a high degree of interoperability with related standards [7]. WCS 2.0 allows exchange of coverages seamlessly between OGC services, thereby establishing a model powerful enough to unify coverage handling across all of OGC. WCS 2.0 now supports more coverages beyond raster like curvilinear grids and more general meshes. The conformance tests are now easier to derive on the implementation under test.

Fig. 1. Some Basic OGC standards related.



Extending the current WCS reference implementation [2] to conform to WCS 2.0 is the focus of this contribution. To this end, the remainder of this contribution is organized as follows. In section 3, we present the WCS 2.0 standard in detail. In section 2 we present the architecture of the reference implementation. Performance Evaluations and Benchmarks are discussed in Section 4. Section 5 describes the related work, with demonstration in Section 6, and Section 7 gives a conclusion.

2 WCS 2.0 Standards Suite

In this section we give a brief overview on the WCS 2.0 suite; the official website is [20]. The coverage model is laid out in the GML Application Schema for Coverages (ASC) specification [3]. It is based on GML 3.2.1 [18], which in turn relies on OGC Abstract Topic 6 [15] which is identical to ISO 19123 [1]. The root of the GML coverage definition is the abstract class `AbstractCoverage` which is a subtype of GML `AbstractFeature`. The main components of `AbstractCoverage` are domainSet, rangeSet and rangeType. In the `domainSet`, the CRS is specified, plus the coverage`s extent. The `rangeSet` contains the concrete range values and `rangeType`, contains a complete specification of the range values, including data type, units of measure, and nil values. To achieve harmonization with the OGC Sensor Web Enablement (SWE) standards family, the DataRecord definition of SWE Common 2.0 [19] has been adopted. Figure 2 shows the extended `AbstractCoverage` structure.

Several concrete subtypes are derived from `AbstractCoverage`, including gridded coverages, point clouds, and other types of varying topological dimension. These allow representing, in addition to multi-dimensional rasters, a large class of practically relevant data structures, such as point clouds, curvilinear grids, Triangulated Irregular Networks (TINs), and general meshes. Figure 3 shows the coverage datatype hierarchy.

All coverage service protocols share two initialization request types. The client will issue a `GetCapabilities` request if information about a service is not known to obtain data and service information like specification version supported, protocols supported, and coverages offered. Further details about selected coverages can be obtained via a `DescribeCoverage` request; the response contains all relevant metadata about the coverages inquired, such as coverage domain extent and range type and CRSs in which the coverage can be addressed. WCS itself defines the `GetCoverage` request for subsetting, scaling, reprojecting, and encoding a coverage [12]. WCS-T adds the TransactionRequest which serves to add a new coverage, update an existing coverage partially or completely

Fig. 2. Extended `AbstractCoverage` Structure

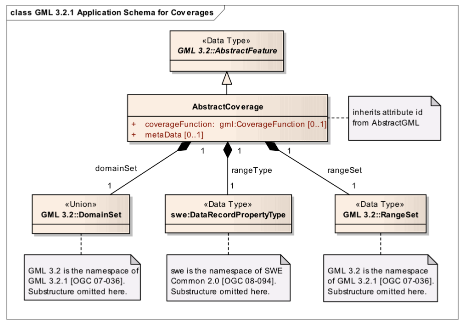
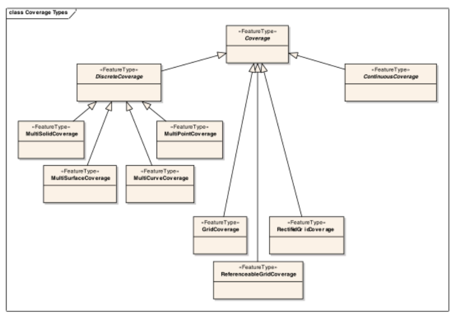


Fig. 3. Coverage Types

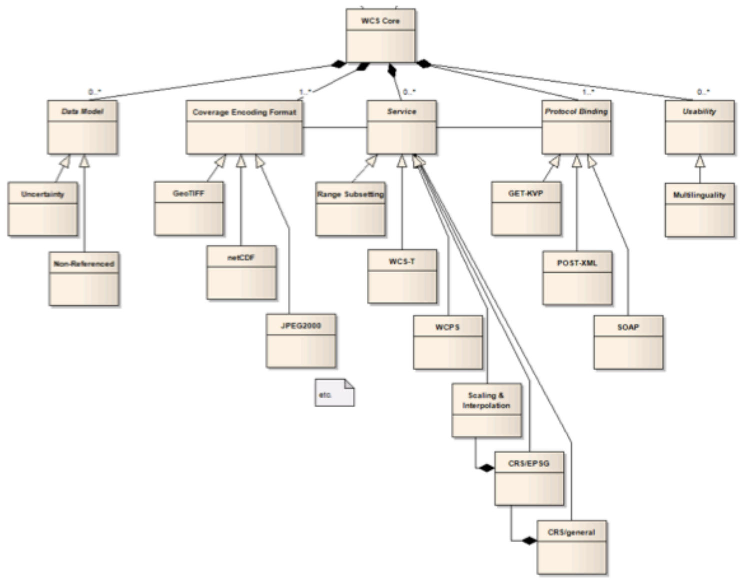


with new data, or to delete a coverage from a WCS repository [26]. WCPS, finally, adds a `ProcessCoverages` request [7].

Following the OGC core/extension model for modular specifications [17], WCS is split into a Core and an open-ended set of extensions. The Core defines properties that any WCS implementation must offer in order to be conformant with WCS overall. Extensions define further functionality which an implementation can add. Figure 4 shows the structuring of extensions using UML.

Data Model Extensions add further information such as error estimates, uncertainty information, and other quality relevant metadata to coverages. *Service Model Extensions*, on the other hand, extend functionality on the coverages offered. WCS-T allows modifying a server`s offering by adding, updating, or deleting coverages [26]. The Processing extension [6] ties in the OGC Web Coverage Processing Service (WCPS) Standard [5] which defines a declarative query language on multi-dimensional raster data in the tradition of SQL. The Scaling & Interpolation extension adds scaling, resampling, and interpolation to the `GetCoverage` request. The family of *Coverage Format Encoding Extensions* support coverage delivery in the various well-known image and scientific data formats such as `GeoTIFF`, `NetCDF`, and `JPEG2000`. The fourth extension category con-

Fig. 4. Structuring the WCS Extension Univers[4]



sists of the *Protocol Bindings* that supports `GET/KVP` [9], `XML/POST` [10], and `SOAP` [11]. Finally *Usability Extensions* add multilingual support.

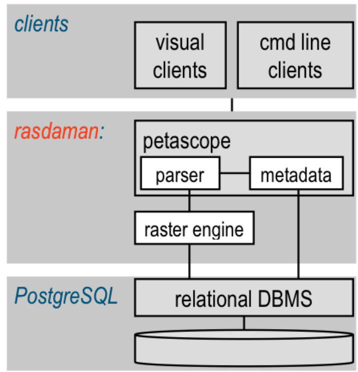
3 PetaScope Architecture

The reference implementation, see Figure 5 is based on the open-source raster server `rasdaman` (raster data manager). The official website is [24]. The array database system stores multi-dimensional MDD raster data in either files or inside a relational database, such as `PostgreSQL`. For adequate storage and retrieval of the raster data, full array support across all the database layers is provided.

Internally, manifold optimizations on logical and physical level help to achieve performance and scalability. On the logical level, such arrays can have any dimensionality, and an arbitrary number of elements per dimension (fixed or variable). The elements present in the array (cells) can be of any type, simple or derived. The rasdaman query language, `rasql`, extends standard `SQL` to multi-dimensional raster arrays in a declarative, evaluation-safe manner. On the physical level multidimensional indexing, clustering, tiling and compression of the MDD arrays is supported. In particular, tiling is very important since, due to performance and memory requirements, using Binary Large Objects (BLOBs) is not an option. To achieve best performance the rasdaman server is implemented in `C++` with APIs in `C++` and `Java`.

The PetaScope layer of rasdaman adds geo semantics by providing OGC based interfaces for WCS, WCPS, WCS-T, and WPS. This Java servlet package receives requests,

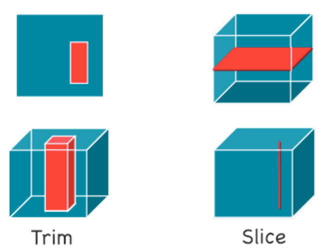
Fig. 5. WCS 2.0 Reference Implementation System Architecture



translates them into `rasql` for processing by rasdaman, and sends the results generated back to the client.

In this contribution, we extend the petascope layer of rasdaman to conform to WCS 2.0 as specified under the OGC specification with the features mentioned in Section 2. The implementation adheres to the *conformance classes* that contain a testing recipe for each requirement structured in the WCS 2.0 draft. The reference implementation in addition to the raster coverages can now also handle point clouds and efficiently perform subsetting (trimming/slicing) operations (see Figure 6) on them. Using automated scripts the point clouds stored in standard `UOS` file format are pushed into the metadatabase layer of petascope and are then available for retrieval using WCS requests.

Fig. 6. Subsetting on Coverages: Trimming and Slicing



Protocol Binding Extensions describe client/server communication protocols (including request parameter encodings) which a WCS implementation may offer and use. Protocol Bindings for WCS 2.0 requests using `KVP` (key-value pair) [9] and XML syntax based on XML schema definitions with both `HTTP POST` [10] and `SOAP` [11] communication are now supported.

Coverages can be delivered by GetCoverage requests in different data formats. In addition and independently from any data format, coverages can be delivered as either pure GML documents, or as a format-encoded data file only, or as a combination where GML is used to represent the metadata and the data file holds the range values. To this end, GML and GeoTIFF format encoding extensions are now supported in the implementation.

4 Performance Evaluation

Fig. 7. Performance Analysis of Trim/Slice Operation on Point Clouds

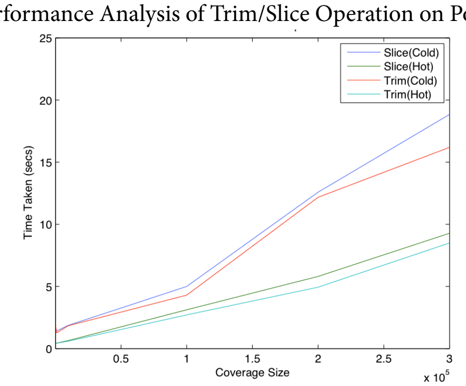


Figure 7 shows the results of a performance evaluation. A WCS 2.0 GetCoverage query under test performs trimming and slicing operations (see Figure 6) respectively on a point cloud coverage of varying sizes. COLD displays results in the case when the database and application servers were just started and query was run for the first time; HOT is time elapsed when the query ran the next several times and the caches were available with the coverage data.

5 Related Work

MATLAB [23] and Mathematica [25] are popular computing environments that support importing scientific data but they target desktop environments. Further, working with data volumes which exceed main memory capacity by several orders of magnitude is not efficient. MapServer [22] is an open source geo-server for building spatially-enabled internet applications, however it currently can handle only 2D data and relies on file-based storage. Our implementation on the other hand, provides efficient retrieval of raster data using a flexible query language, supports retrieval of non-raster data like point clouds, is scalable and supports up to n-Dimensions.

NASA is developing, with our collaboration, an on-board satellite interface which allows to task WCPS queries in an ad-hoc fashion[13]. Based on an own implementation independent from PetaScope, NASA`s plan is to significantly enhance quality and intelligence of service for the next generation of earth observation satellites.

6 Demonstration

Demonstration data will cover raster coverages of spatio-temporal dimensions, ranging from 1-D timeseries over 2-D remote sensing and seafloor imagery to 3-D geophysics and 4-D climate model data, alongwith non-raster coverages such as point clouds of varying sizes. The operations involve either retrieval of complete coverage or a subset thereof using either one of trim/slice operations. The WCS requests can be sent using either a key-value pair, (`KVP`) or XML based `POST/SOAP` communication protocol. In addition to the prefabricated demo cases, a sandbox allows to visually experiment with the queries. All this functionality is additionally permanently made available on the Earth-Look [21]

These queries challenge the engine in different ways. Subsetting speed is mainly driven by disk access (and the fit of the tiling structure) and to some lesser extent by agglomerating the result from the tiles read, data format encoding, and shipping to the client. The operations mentioned above are mostly CPU-bound. Depending on the Internet connection, either a local notebook server or Jacobs University server will be accessed. Visitors can simultaneously exercise prefabricated and ad-hoc queries via Earth-Look [21].

7 Conclusion

With WCS 2.0, OGC aims at providing a technologically up-to date, extensible service definition allowing for efficient, scalable implementations. Extensive proofreading by scientists and practitioners from many different fields have contributed substantially to get as close as possible to the goals stated. Our reference implementation in this context constitutes a `HTTPServlet` that receives coverage requests in different communication formats and supports retrieval of both raster and point cloud data in varied encoding formats from a flexible, multi-dimensional, multi-purpose geo raster server which is available in open source.

Among our research avenues are further server-side query optimizations, implementation of the Earth Observation and Aviation Services standard currently in development under OWS-8 [16]. Further, we envisage application of the PetaScope technology in domains beyond the earth sciences, such as human brain imaging and gene expression analysis.

References

1. *Geographic Information - Coverage Geometry and Functions.*, Number 19123:2005. ISO, 2005.
2. AAiordăchioaie and PBaumann. Petascope: An open-source implementation of the ogc wcs geo service standards suite. *Scientific and Statistical Database Management*, pages 160--168, 2010.
3. PBaumann, editor. *GML Application Schema for Coverages.* Ogc document 09-146 edition.
4. PBaumann, editor. *WCS 2.0 -- Overview (White Paper).* Ogc document 09-153r1 edition.
5. PBaumann. *Web Coverage processing Service (WCPS) Interface Standard.* Ogc document 08-068r2 edition.
6. PBaumann, editor. *Web Coverage Service (WCS) -- Processing extension 1.0.* Ogc document 08-059r1 edition.
7. PBaumann. Beyond rasters: Introducing the new ogc web coverage service 2.0. *Proceedings of the 18th SIGSPATIAL International ...*, Jan 2010.
8. PBaumann. The ogc web coverage processing service (wcps) standard. *Geoinformatica*, pages 1--33, 2010.
9. PBaumann. *OGC® Web Coverage Service 2.0 Interface Standard - KVP Protocol Binding Extension.* Ogc 09-147r1 edition, September 2010.
10. PBaumann, editor. *OGC® Web Coverage Service 2.0 Interface Standard - XML/POST Protocol Binding Extension.* Ogc 09-148r1 edition, September 2010.
11. PBaumann, editor. *OGC® Web Coverage Service 2.0 Interface Standard - XML/SOAP Protocol Binding Extension.* Ogc 09-149r1 edition, September 2010.
12. PBaumann, editor. *OGC® Web Coverage Service 2.0 Interface Standard: Core.* Ogc 09-110r3 edition, September 2010.
13. Pat Cappelaere. http://www.geobliki.com/2009/08/11/nasa-wcps-for-ground-space-interoperability.
14. JohnD. Evans, editor. *Web Coverage Service (WCS), Version 1.0.0.* Ogc 03-065r6 edition.
15. OGC, editor. *Abstract Specification Topic 6: Schema for cov- erage geometry and functions.*, volume OGC document 07-111.
16. Phase 8 (OWS-8) seen 10-02-2011 OGC WebServices. http://www.opengeospatial.org/projects/initiatives/ows-8.
17. Policy.SWG, editor. *The Specification Model -- a Standard for Modular Specifications.* Ogc document 08-131r3 edition.
18. CPortele, editor. *Geography Markup Language (GML) Encoding Standard 3.2.1.* Ogc document 07-036 edition.
19. ARobin, editor. *SWE Common 2.0.* Ogc document 08-094 edition.
20. OGC. WCS2.0 seen 08-02-2011. http://www.opengeospatial.org/standards/wcs.
21. EarthLook seen 10-02-2011. http://www.earthlook.org.
22. MapServer seen 10-02-2011. http://mapserver.org/.
23. MATLAB seen 10-02-2011. http://www.mathworks.com/products/matlab/.
24. Raster DataManager seen 10-02-2011. http://www.rasdaman.org.
25. WolframMathematica seen 10-02-2011. http://www.wolfram.com/mathematica/.
26. Arliss Whiteside, editor. *Web Coverage Service (WCS) Transaction Operation Extension.* Number 07-068r4. ogc, 2008. edition.