Improving Spatiotemporal Raster Management in GeoServer

Ing. Simone Giannecchini, GeoSolutions
Ing. Andrea Aime, GeoSolutions
Outline

- GeoServer Quick Intro
- Work Plan
- Current Status
- Next Steps
- Conclusions
GeoSolutions

- Founded in Italy in 2006
- Expertise
  - Image Processing, GeoSpatial Data Fusion
  - Java, Java Enterprise, C++, Python
  - JPEG2000, JPIP, Advanced 2D visualization
- Supporting/Developing FOSS4G projects
  - GeoTools, GeoServer, MapStore
  - GeoBatch, GeoNetwork
- Clients
  - FAO (CIOK, FIGIS, NRL, FORESTRY, ESTG), JRC, ARPAT, Nato CMRE, DLR, EUMETSAT, WFP
  - ITT-VIS, E-GEOS, GEOSMART, OpenGeo, SINERGIS, City of Prato, City of Florence, County of Florence, County of Prato, CSI-Piemonte, NWGEO, IGEA, AMBRERO, LIBEROLOGICO, Astrium UK, EOMAP, City of Wien, County of Bozen, City of Bozen, etc...
- [http://www.geo-solutions.it](http://www.geo-solutions.it)
GeoServer Quick Intro
GeoServer

- GeoSpatial enterprise gateway
  - Java Enterprise
  - Management and Dissemination of raster and vector data

- Standards compliant
  - OGC WCS 1.0, 1.1.1 (RI), 2.0 in the pipeline
  - OGC WFS 1.0, 1.1 (RI), 2.0
  - OGC WMS 1.1.1, 1.3
  - OGC WPS 1.0.0

- Google Earth/Maps support
  - KML, GeoSearch, etc..
Formats and Protocols

**Shapefile**
- PostGIS
- Oracle
- H2
- DB2
- SQL Server
- MySql
- Spatialite
- GeoCouch

**PostGIS**
- WFS
- WCS
- WMS

**ArcSDE**
- WFS

**GeoTIFF**
- WMS
- ArcGrid
- GTopo30
- Xm+world
- Mosaic
- MrSID
- JPEG 2000
- ECW, Pyramid, Oracle GeoRaster, PostGis Raster

**DBMS**
- SQL Server
- MySql
- Spatialite
- GeoCouch

**Servers**
- WFS
- WMS
- WPS
- WCS
- GWC

**Supported Formats**
- PNG, GIF
- JPEG
- TIFF, GeoTIFF
- SVG, PDF
- KML/KMZ
- Shapefile
- GML2
- GML3
- GeoRSS
- GeoJSON
- CSV/XLS
- Raw vector data
- Raw raster data

**GeoServer**
- Google
- Styled maps
- KML superoverlays
- Google maps tiles
- OGC tiles
- OSGEO tiles
Administration GUI

GeoServer

Welcome

Workspace

Basic Store Info

Data directory: release/data/taz_shapes/

Name | Last modified | Size
--- | --- | ---
tasmania_cities.shp | 23-ago-2010 10:21 | 164
 tasmania_roads.shp | 23-ago-2010 10:21 | 8,3K
tasmania_state_boundaries.shp | 23-ago-2010 10:21 | 6,6K
tasmania_water_bodies.shp | 23-ago-2010 10:21 | 9K
RESTful Configuration

- Programmatic configuration of layers via REST calls
  - Workspaces, Data stores / coverage stores
  - Layers and Styles, Service configurations
  - Freemarker templates (incoming)
- Exposing internal configuration to remote clients
  - Ajax - JavaScript friendly
- Various client libraries available in different languages (Java, Python, Ruby, ...).
- Example, geoserver-manager:
  https://github.com/geosolutions-it/geoserver-manager
Security: Authentication

- Pluggable user sources, available out of the box:
  - LDAP, DBMS
- Pluggable authentication mechanisms, available out of the box:
  - BASIC/DIGEST HTTP, CAS
- Possible to integrate with other mechanisms and in-house solutions
- Available since 2.2.0, before only basic HTTP auth + simple text file for users
WMS

- Dissemination of Maps
  - Fusing raster and vector data seamlessly
  - Rule/scale driven rendering
- WMS 1.1.1 and 1.3 support
- SLD
  - Basic support for SLD 1.1 and SE 1.1
  - Full support for SLD 1.0
- CSS extension for compact styling
- Many rendering extensions available
WMS: TIME and ELEVATION

TIME = 20100512T0000000Z
ELEVATION = 0.0

FeatureType Editor
WMS: Rendering Transformations

- On-the-fly data transformations
- Calling spatial analysis processes from SLD docs
- Optimized for performance
- Examples: on the fly contour lines, heat maps, point clustering, point interpolation, GCP based image rectification

GeoSolutions
GeoWebCache Integration

- Direct calls to GeoServer rendering engine
- Support for layers modified through WFT-T
- Support for various tile protocols
  - GMap, Gearth
  - OpenLayers, VEarth, Bing
- Speed-up factor 10/100
- Disk quota support
GWC integration

- Custom gridset definition
- Per layer caching configuration
KML/KMZ

TEMPORAL SERIES

KML EXTRUDE

KML SUPEROVERLAY

GeoSolutions
WFS

- Dissemination and filtering of vector data
- WFS 1.0, 1.1 and 2.0 (since 2.2.0)
- Transaction and paging available in all versions
- Simplified filtering via CQL

**Formats:**

- GML 2, 3.1 and 3.2
- CSV, Excel spreadsheet, GeoRSS, GeoJSON
- Shapefile (zipped)
- Any other format supported by ogr2ogr (configurable)
- More output formats can be generated by XSLT
Complex Feature*

- Application/community schemas
- Complex Features
  - Attributes as sub-features
  - Attributes as list of features
  - Tree-like structure
- Mixing in a single tree heterogeneous data sources
WCS

- Raster data dissemination
  - Raw raster data useful for analysis, no maps!
  - Support for TIME and ELEVATION (via ImageMosaic plugin)
- WCS 1.0 and 1.1.1
- Output formats
  - GeoTiff, ArcGrid
  - GDAL based formats under discussion
- Extensions
  - ELEVATION as band management
- WCS 2.0.X Implementation Ongoing
WPS

- WPS 1.0
- Official Extension
- Raster and Vector data support
- High performance processes (raster/vector statistics, raster/vector format conversions and more), asynchronous call support

Integrated WPS

- Direct access to data sources
- Automatic publishing of results as new layers
- Embedding processes into SLD styles (rendering transformation, since 2.2.0)
Database configuration backend (2.3)

- Pluggable configuration backends
- In-memory implementation + XML storage (current one)
- Database based implementation (as a community module)
- Pluggable, add your own (any takers for a NoSQL elastic implementation?)
Improved clustering for GWC in 1.4.x:
- Metastore removed
- Disk quota can work off a central DBMS
- Distributed locks, avoid concurrent computation of same tile at the same time

Active/active clustering of GWC now possible
Wow, catalogue services in GeoServer!

Catalog Service for the Web 2.0.2

Pluggable record backend

Pluggable record type support

Not a replacement for a full-fledged GeoNetwork (not at the moment, at least)
WCS 2.0 implementation with extensions:
- Range subsetting
- Scaling and interpolation
- CRS (reprojection)
- GeoTiff & NetCDF encoding

Earth Observation profile support
- Temporal series
- Exposing mosaic structure
- EO metadata describing sensors

NetCDF support as both input and output

Sponsors
- DLR (German spatial agency)
- EUMETSAT (European operational satellite agency for monitoring weather, climate and the environment)
Spatiotemporal Raster
Work Plan
Work Plan – The great plan

- Goals
  - Better support EO and MetOc data in GeoServer
  - Implement WCS 2.0.1 in GeoServer
  - Implement WCS-EO in GeoServer
  - Implement WMS-EO in GeoServer
  - Support serving NetCDF Data from GeoServer
  - Improve ability to manage Multidimensional data via REST in GeoServer

- Activities
  - WCS 2.0.1
  - WMS-EO
  - WCS-EO
  - Ingestion with GeoBatch
  - Work on multidimensional input formats
  - PostgisRaster support
Work Plan – WMS-EO

- How this fits in GeoServer’s world
- GSIP 84

Wizard to configure EO layer groups
Extending LayerGroup concept
Support same style on both raster and vector data
Support custom dimensions
Alter map on the fly to support band combination

GeoSolutions
WMS-EO Quirks

- Root layer must respond with a specific layer rather than the composition of the children
  - Kind of a *default* visualization for a certain Dataset
  - E.g. when I send a GetMap for the root layer I could get back the *browse image* default layer
- Same styles for both Raster as well as Vector data
  - E.g. *yellow* can be use for both flags as well as outlines
- Peculiar behavior for Band layer (raw data)
  - E.g. multiple bands at different wavelengths
  - Can request either 1 (grayscale image) or 3 (RGB image)
  - Different combinations are prohibited
- Peculiar Behavior for GetFeatureInfo
Work Plan – WCS 2.0

- **Build the basics**
  - Core service
  - KVP binding
  - XML binding

- **Implement the GetCoverage extensions**
  - CRS
  - Scaling
  - Interpolation
  - Range subsetting
  - GeoTiff
  - GML
  - NetCDF

- **Add the output format extensions**
  - GeoTIFF
  - GML Grid

- **Ensure CITE tests compliancy**
Work Plan – WCS 2.0

- Processing Chain

1. Crop
2. RangeSubset
3. Scale & Interpolate
4. Reproject & Interpolate
5. Encode

Core

Extension

Extension

Extension

Extension

GeoSolutions
Work Plan – WCS-EO

- **Build on top of a working WCS 2.0 with full extensions**
  - WCS 2.0
  - CRS extension
  - Range subsetting extension
  - Interpolation extension
  - Scaling extension
  - GeoTiff extension
  - NetCDF extension

- **Add support for the WCS-EO extras**
  - Listing coverage datasets in the capabilities documents (based on image mosaic contents, which will have to be marked as “exposed” so that we show their inner structure for EO)
  - Support describe coverage dataset
  - Support returning results for an entire dataset in GetCoverage
Work Plan – WCS-EO

- Add support for downloading the original file in case of no subsetting/reprojection/scaling/format change
  - Add support to GeoTools readers to signal they are returning us an original file
  - Use that information to download the original file directly
- Add support for WCS EO metadata in readers
  - Associate each file with EO metadata
  - Include such information in DescribeCoverage/DescribeEODataset
- Ensure CITE tests compliancy
Work Plan – Additional Format Support

- **NetCDF support**
  - Improve existing NetCDF/CF input format, support CF convention and make sure the samples provided by DLR/EUMETSAT can be read
  - Expose NetCDF internal data as a set of 2D slices
  - Write new NetCDF/CF output format for GeoServer
Work Plan – Additional Format Support

- **ImageMosaic NetCDF integration**
  - Allow the ImageMosaic to handle NetCDF files
  - Expose NetCDF internal structure (times, elevations)
  - Make ImageMosaic handle slices of the NetCDF file as granules
Work Plan – Additional Format Support

- **ImageMosaic improvements**
  - Mark returned coverage as single granule when a single, whole file has been used to generate the output
  - Expose the inner structure to support DataSet time and stitched mosaic notions
  - Allow manipulation of the index to support REST config extensions
Work Plan – REST Interface Improvements

- Add REST support to expose an image mosaic internal structure
  - Dimensions
  - Granules

- Dimensions: list, edit, create, remove
  - /workspaces/<ws>/coveragestores/<cs>/coverages/<mosaic>/dimensions
  - /workspaces/<ws>/coveragestores/<cs>/coverages/<mosaic>/dimensions/<dimension>[.format]
  - TODO paging and query of dimension domain to overcome WMS Dimensions limitations

- Granules: list, edit, create, remove
  - /workspaces/<ws>/coveragestores/<cs>/coverages/<mosaic>/index
  - /workspaces/<ws>/coveragestores/<cs>/coverages/<mosaic>/index/pageN
  - /workspaces/<ws>/coveragestores/<cs>/coverages/<mosaic>/index/pageN/granuleM
Spatiotemporal Raster
Current Status
Current Status - Overview

- **WMS EO**
  - Layer group extension complete
  - GeoServer support for static custom dimensions added
  - Custom dimension support added to image mosaic, GeoServer support for dynamic dimensions complete

- **WCS 2.0**
  - Basic implementation completed
  - GetCoverage extensions completed: crs, interpolation, scaling, range subsetting
  - GeoTiff request parameters and compliant output format
  - GML grid output format

- **WCS-EO**
  - Basic extensions are in already

- **NetCDF reader support**
  - Initial work on low level readers

- **Enhancement to ImageMosaic**
  - Support for Custom dimensions
Current Status – WMS-EO

- Created GeoServer improvement proposal
  - GSIP 84
- Discussed the proposal on the mailing list:
  - See here and here
- LayerGroup Enhancements
  - Added Type property
    - **Single**: the layer group as today, a single exposed layer with a name
    - **Named tree**: the layer group retains a Name in the layer tree, but also exposes its nested layers in the capabilities document
    - **Container tree**: the layer group is exposed in the tree, but does not have a Name element, showing structure but making it impossible to get all the layers at once
    - **Earth Observation tree**: created to manage the earth observation requirements
  - **Added Root Layer and Root Layer Style**, used only when Type EO is selected
  - **Added methods** `renderingLayers()` and `renderingStyles()`
Updated GeoServer web administration interface to edit new LayerGroup properties.
Current Status – WMS-EO

- Added Support for custom dimensions to GeoServer
- “Dynamic dimensions”, dimensions whose existence the reader gets to know about at runtime
Current Status – WMS-EO

- Updated GetCapabilities and GetMap request handlers for WMS 1.1.1 and WMS 1.3.0

```
<Layer queryable="1">
    <Name>sf:EO LayerGroup sample</Name>
    <Title>An Example</Title>
    <Abstract>Layer-Group type layer: sf:EO LayerGroup sample</Abstract>
    <CRS>EPSG:26713</CRS>
    <EX_GeographicBoundingBox>...</EX_GeographicBoundingBox>
    <BoundingBox CRS="EPSG:26713" minx="589434.4971235897" miny="4913947.342298816" maxx="600614.2067249198" maxy="5101614.1808442892" />
</Layer>

<Layer queryable="1">
    <Name>sf:streams</Name>
    <Title>Spearfish streams</Title>
    <Abstract>...</Abstract>
    <KeywordList>...</KeywordList>
    <CRS>EPSG:26713</CRS>
    <CRS>EPSG:84</CRS>
    <EX_GeographicBoundingBox>...</EX_GeographicBoundingBox>
    <BoundingBox CRS="EPSG:84" minx="-103.87789019829768" miny="44.372335260095554" maxx="-98.74951600834961" maxy="49.05322286694339" />
    <Style>...</Style>
</Layer>

<Layer queryable="1">
    <Name>sf:restricted</Name>
    <Title>Spearfish restricted areas</Title>
    <Abstract>...</Abstract>
    <KeywordList>...</KeywordList>
```

GeoSolutions
Current Status – WCS 2.0

- GeoTools
  - EMF models
  - XML Parsing

- Geoserver
  - Service architecture: changes in input params
    - OWS2 allows repeated keys in KVP
  - Created WCS 2.0.1 service
    - Community module
Current Status – WCS 2.0

- **Geoserver**
  - Implemented operations
    - **GetCapabilities** (KVP GET + XML POST)
    - **DescribeCoverage** (KVP GET + XML POST)
    - **GetCoverage** (KVP GET + XML POST)
  - Implemented Handling exceptions according to OWS2.0
    - Some differences from v1:
      - Referenced schemas
      - Introduced support to HTTP response codes
  - **So far we did not focus on support for Time and Elevation**
Current Status – WCS 2.0

- CRS Extensions is in
- Interpolation Extension is in
- Range Subsetting Extension is in
- Scaling Extension is in
- GeoTiff Extension is in
- GML Output is in
Current Status – CITE Tests

- After some fixing on CITE test definition for WCS 2.0, 101 tests are passing and 14 are failing.

- The base repository used for tests is [https://github.com/geoserver/geoserver-cite-tools](https://github.com/geoserver/geoserver-cite-tools)

- Patch for the CITE tests is at [https://github.com/geoserver/geoserver-cite-tools/blob/master/20121213_tests_wcs_200_tags_r5_ctl.patch](https://github.com/geoserver/geoserver-cite-tools/blob/master/20121213_tests_wcs_200_tags_r5_ctl.patch)


- Tickets already opened at [http://cite.opengeospatial.org/issues](http://cite.opengeospatial.org/issues)

Current Status – NetCDF as Input

- Multidimensional Datasets management
- I/O, metadata management and ND APIs
- Low level - 2D Slices Readers (ImageIO-Ext):
  - ImageIO readers, format specific Image Readers (NetCDF) (ImageIO Direct readers)
- High level - Coverage API (GeoTools):
  - CoverageStore, CoverageAccess, CoverageSource, Driver (GeoTools ND plugins)
  - SpatioTemporal ImageReaders (GeoTools smart readers)
    - Build spatioTemporal metadata
    - Expose SliceDescriptors describing 2D Slices
      - (temporal extent, vertical extent, imageIndex, envelope)
**Current Status – NetCDF as Input**

- **Spatiotemporal Metadata**
  - [http://demo.geo-solutions.it/share/ImageIOMetadataDoc.pdf](http://demo.geo-solutions.it/share/ImageIOMetadataDoc.pdf)

- **Collecting domains information**
  - CoordinateReferenceSystems
  - RectifiedGrid
  - Boundings
  - Bands

- **Simplifications to be done**
  - spatioTemporal metadata
  - sampleDimensions (Bands)
  - ND dimensions to 2D slice mapping
Current Status – NetCDF as Input

- **ImageIO-Ext NetCDF low level access libraries/plugins**
  - [https://github.com/geosolutions-it/imageio-ext.git](https://github.com/geosolutions-it/imageio-ext.git)
  - [https://github.com/geosolutions-it/imageio-ext/tree/1.1.x/library/netcdf-core](https://github.com/geosolutions-it/imageio-ext/tree/1.1.x/library/netcdf-core)
  - [https://github.com/geosolutions-it/imageio-ext/tree/1.1.x/plugin/unidata-netcdf/netcdf](https://github.com/geosolutions-it/imageio-ext/tree/1.1.x/plugin/unidata-netcdf/netcdf)
  - 2D slices of ND datasets accessed through direct imageIndex-kes
  - Index to slice mapping: COARDS convention: t,z,y,x
  - global and variable’s attributes are exposed (needed to build metadata)
  - Currently based on 4.0.41 (consider switching to 4.3)

- **GeoTools multidimensional rasters and NetCDF Driver**
  - [https://github.com/simboss/geotools/tree/master-multidimraster](https://github.com/simboss/geotools/tree/master-multidimraster)
  - /modules/unsupported/coverage-experiment
  - /modules/unsupported/coverage-experiment/netcdf

- **SpatioTemporalImageReaders**
  - Wraps ImageIO-Ext direct readers to provide SliceDescriptors and populate spatioTemporal metadata (use Direct reader to access exposed datasets attributes and dimensions)

- **NetCDFSpatioTemporalImageReaders**
  - Parse NetCDF attributes and metadata to setup SpatioTemporal metadata and SliceDescriptors
Current Status – Things to clarify

- Coverage names are defined as NCName in OWS2

- ExceptionReport version when a service/version pair can not be inferred from the request

- Doubts with AxesLabels VS AxisIdentififers for subsetting
- Doubts with AxesLabels VS AxisIdentififers for scaling
- Missing Array-Type Range Type
- Still a little unclear on Time management (potential PICNIC)
- Not sure how to manage non equally-spaced Elevation levels
  - Using the Bands with multivalued domain the past, now hard to do this way
Current Status – Blockers

- WCS Extensions are moving targets
  - did a first implementation, we’ll have to revisit once they settle down
- CITE tests are available for WCS 2.0
  - 2.0.1 version with tests for extensions is in heavy development by OGC right now
- Patches for CITE tests should be ack by OGC
- Some fixes in CITE tests code are needed
  - Some OWS2 requirements are not being incorporated
    - HTTP return codes
    - ExceptionReport schema version
- Missing WCS extensions
  - RangeSubset for other types of Field?
Next Step - Overview

- **WMS EO**
  - Create community module for WMS-EO specific extensions:
    - GetMap for bands merging
    - Complete wizard for WMS EO layer group creation

- **WCS 2.0.1**
  - Improve testing
  - Review extensions implementations once spec settles down
  - Test against CITE tests for WCS 2.0.1 once they are ready

- **WCS EO**
  - Build support for datasets (by marking mosaics as dataset sources)
  - Amend GetCapabilities output
  - Create DescribeEODataset and its output
  - Add support for datasets in GetCoverage, as well as extracting a particular coverage out of them
Next Step - Overview

- REST configuration improvements
  - Add support for programmatically manipulating mosaics
  - Expose these capabilities via REST
- Additional input/output formats
  - Complete work on NetCDF input, expose inner structure as a set of 2D slices
  - Extend mosaic to integrate NetCDF slices as granules
  - Prepare NetCDF output
  - Extend mosaic to support PostGIS rasters
- Ingestion work
- Moving WCS 2.0 community module into standard extension
Next Step – Possible Issues

- **WMS-EO**
  - **Band Coverages:**
    - 1 Dimension – Greyscale rendering
    - 3 Dimensions – RGB rendering
    - Other number of Dimensions - WMS-EO server should throw an error and this behaviour differs from WMS standard
  
  Rendering will require additional processing: WMS service must recognize Band layers and render them accordingly to WMS-EO specification.

- **Root Layer must have a Time Dimension**
  - This Time Dimension will be inherited by all contained Layers.

- **Outline Layers**
  - a Time Dimension parameter will be required for getMap and getFeatureInfo request. It could be a time range
  - getFeatureInfo requests must return product information and acquisition time
Next Step – Possible Issues

- Uncertainty of WCS extensions future modifications and timing
- Uncertainty of WCS 2.0.1 CITE tests completion and availability
- Uncertainty of WCS EO CITE tests completion and availability
- Refactoring to support PostGIS rasters, possible community issues on overlap with JDBC image mosaic
- Window to make large changes in GeoServer core closes January 21 2013, we can continue work on trunk but it will have to go though proposals for backport or delayed to 2.4.x. series (due September 2013)
Next Step – Possible Issues

- **NetCDF input support**
  - Entirely new approach requires deep modification to GeoTools-ImageMosaic/GeoServer
  - Potential performances/memory issues may happen (with respect to using GeoTiff based imageMosaics VS using NetCDF based ImageMosaics)

- **Vertical dimensions specifications**
  - Nightmare in some cases with compound CRS
Conclusions

- WCS 2.0.X specs look way better than 1.0
  - Nice usage of extensions
  - Clear model, Clear way of thinking
  - Still some grey areas
- So far good, steady progress has been made
  - Minor issues in adapting GeoTools and GeoServer to the specs
  - Idiosyncrasies with CITE tests
  - Some delays from the Community
  - Also positive feedback from the Community
- Challenging Tasks ahead of us
  - Multidimensional data
  - WCS-EO
  - Linking WMS-EO to data
  - TIME and ELEVATION
  - Ingestion
The End

Questions?

simone.gianneccchini@geo-solutions.it
Rendering: real world units

```xml
<Rule>
  <LineSymbolizer uom="http://www.opengeospatial.org/se/units/metre">
    <Stroke>
      <CssParameter name="stroke-width">
        <ogc:Literal>5</ogc:Literal>
      </CssParameter>
    </Stroke>
  </LineSymbolizer>
</Rule>
```
WMS: PNG8 with alpha

- Support for paletted PNG with alpha transparency
- Best of both worlds: compact but good looking
- Good quality, yet usable in interactive setups
Virtual services

- Expose different OGC services per workspace
- Styles and layer groups per workspace
- Have different administrators per workspace → multi-tenancy
WFS: 2.0 and XSLT

- WFS 2.0
  - GML 3.2
  - Paging (back-ported to other versions)
  - Joins (scalar, temporal, spatial) between feature types
  - Stored queries
- XSLT output format:

  ![Diagram](image_url)
CSW 2.0.2 (2.3)

- Current implementation
  - Demo backend with Dublin Core record support, passes CITE certifications tests
  - ISO + Dublin core backend reporting layers in the GeoServer configuration, in development
  - ebRIM (Earth Observation profile) implementation plus proxy to a in-house, vendor specific catalog (proxy front-end model)

- Currently a community module
  - will be graduated to extension once the ISO backend over the GeoServer own config is completed
Work Plan – The great plan

- Milestones
  - **M1, Dec 2012**
    - WCS 2.0
    - WMS-EO
  - **M2, End of Feb 2013**
    - Additional Formats
    - REST Improvements
  - **M3, End of March 2013**
    - WCS-EO
    - Ingestion
  - **M4, End of June 2013**
    - Support
Work Plan – WMS-EO

- Abstract Model
  - Exposing a meaningful (EO-Wise) complex structure
  - Raw data, products, browse images, footprints
  - Heavy usage of WMS Layer Inheritance (with some quirks)

```
MERISS__RR_2P

1. 1..* OPT
    MERISS__RR_2P_outlines
      <DIMENSION name="time"...>
        - optional
        - queryable
        - queryable (XML con time ISO8601)
        - 10 predefined styles (white, yellow, orange, red, magenta, blue, cyan, green, brown, black)

1..*
    MERISS__RR_2P_bands
      - optional
      - queryable
      - support Dimension in case we have available things like polarization, wavelength and so on

1..*
    MERISS__RR_2P_geopar
      - optional
      - queryable
      - support a LEGENDURL with a colorbar

1..*
    MERISS__RR_2P_flagname
      - optional
      - queryable
      - 10 predefined styles (white, yellow, orange, red, magenta, blue, cyan, green, brown, black)
```
Work Plan – WMS-EO

- Extending the LayerGroup GeoServer concept
  - Show the nested layers in the capabilities document
  - Allow the root of the group to be represented by a separate layer (the overview one)
  - The above is a set of API, GUI and REST config changes, so a GeoServer proposal is needed

- Add the notion of custom dimensions in raster data
  - Needed to support the “eoproduct_bands” layer
  - Modify the GeoServer API to support custom dimensions (was almost ready in this respect)
  - Modify the grid coverage readers API to allow new dimensions to be exposed
  - Allow “dynamic” dimensions to be exposed (dimensions that are configured by the user)
  - Adapt the GUI to allow new dimensions to be configured
Work Plan – WMS EO

- **Support for band merging**
  - Add extension point to GeoServer to change the layer/styles on the fly → requires a proposal
  - Throw exceptions if outside of the supported cases (one band for gray, three bands for RGB)

- **Wizard- base configuration**
  - Configure easily a EO layer group from a mosaic and a set of layers
  - Automate creating footprints
  - Automate creating bands
  - Cherry pick geophysical and mask layers
  - Automatically associate the predefined styles to the mask layers and footprints

- **Tricky Stuff**
  - Allow using the same style name on both mask layers and footprint layers
  - Develop a new filter function to choose which symbolization to use based on the layer nature
Current Status – WMS-EO

- **Additional Domains support on GeoTools**
  - GEOT-4347

- **AdditionalDomainAttributes for indexer.properties**
  - See example [here](#)

- **Reader support getDynamicParameters**
  - Custom mosaic dimensions <-> dynamic read parameter descriptors
  - Currently only Strings are supported

- **WMS Service Enhancements**
  - Get metadataNames and metadataValues from the reader
  - Get dynamicParameters from the reader
  - Combine them to send the request to the coverage reader
Current Status – WCS 2.0

- Support for CRS extension (both XML and KVP):
  - Specify subsetting CRS (for crop in world coordinates)
  - Specify output CRS (for reprojection of output)

```xml
<?xml version="1.0" encoding="UTF-8"?>
<wcs:GetCoverage xmlns:wcs="http://www.opengis.net/wcs/2.0"
    xmlns:wcsCrs="http://www.opengis.net/wcs/service-extension/crs/1.0"
    xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.opengis.net/wcs/2.0 http://schemas.opengis.net/wcs/2.0/wcsAll.
    service="WCS" version="2.0.1">
    <wcs:Extension>
        <wcsCrs:subsettingCrs>http://www.opengis.net/def/crs/EPSG/0/3857</wcsCrs:subsettingCrs>
        <wcsCrs:outputCrs>http://www.opengis.net/def/crs/EPSG/0/3857</wcsCrs:outputCrs>
    </wcs:Extension>
    <wcs:CoverageId>Blue Marble</wcs:CoverageId>
    <wcs:DimensionTrim>
        <wcs:Dimension><wcs:TrimLow>1.6308305401213994E7</wcs:TrimLow><wcs:TrimHigh>1.6475284637403902E7</wcs:TrimHigh></wcs:Dimension>
    </wcs:DimensionTrim>
    <wcs:DimensionTrim>
        <wcs:Dimension><wcs:TrimLow>-5543147.203861462</wcs:TrimLow><wcs:TrimHigh>-5311971.846945147</wcs:TrimHigh></wcs:Dimension>
    </wcs:DimensionTrim>
    <wcs:format>image/tiff</wcs:format>
</wcs:GetCoverage>
```
Current Status – WCS 2.0

- Support for interpolation extension (nearest-neighbor, linear, cubic, both XML and KVP):
  - We don’t supported different interpolations for Lat,Lon
  - We don’t support
    - Lost area
    - Barycentric
    - Quadratic
  - We will strive to support Nearest for Time dimension

```xml
<?xml version="1.0" encoding="UTF-8"?>
<wcs:GetCoverage xmlns:wcs="http://www.opengis.net/wcs/2.0"
xmlns:int="http://www.opengis.net/WCS service-extension interpolation/1.0"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wcs/2.0 http://schemas.opengis.net/wcs/2.0/wcsAll.xsd http://www.opengis.net/wcs/2.0 http://schemas.opengis.net/wcs/2.0/wcsAll.xsd"
service="WCS" version="2.0.1">
  <wcs:Extension>
    <int:Interpolation>
      <int:InterpolationMethod>
        interpolationMethod="http://www.opengis.net/def/interpolation/OGC/1/linear" />
      </int:InterpolationMethod>
    </int:Interpolation>
  </wcs:Extension>
  <wcs:CoverageId>wcs_BlueMarble</wcs:CoverageId>
  <wcs:format>image/tiff</wcs:format>
</wcs:GetCoverage>```
Current Status – WCS 2.0

- Range subsetting support (both XML and KVP):
  - Based on SWE:DataRecord
  - No support yet for bands referring to arrays of values, like Wavelength

```xml
<?xml version="1.0" encoding="UTF-8"?>
<wcs:GetCoverage xmlns:wcs="http://www.opengis.net/wcs/2.0"
 xmlns:gml="http://www.opengis.net/gml/3.2"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns:rsub="http://www.opengis.net/wcs/range-subsetting/1.0"
 service="WCS" version="2.0.1">
 <wcs:CoverageId>wcs__BlueMarble</wcs:CoverageId>
 <wcs:Extension>
   <rsub:rangeSubset>
     <rsub:rangeItem>
       <rsub:rangeComponent>band1</rsub:rangeComponent>
     </rsub:rangeItem>
     <rsub:rangeItem>
       <rsub:rangeComponent>band3</rsub:rangeComponent>
       <rsub:endComponent>band5</rsub:endComponent>
     </rsub:rangeItem>
     <rsub:rangeInterval>
       <rsub:startComponent>band1</rsub:startComponent>
       <rsub:endComponent>band5</rsub:endComponent>
     </rsub:rangeInterval>
   </rsub:rangeSubset>
 </wcs:Extension>
</wcs:GetCoverage>
```
Current Status – WCS 2.0

- Scaling support, uniform factor, different factor by axis, to specified size, to specified extent (both XML and KVP)
Current Status – WCS 2.0

- GeoTiff extension, setting parameters for the generated GeoTiff (both XML and KVP)
  - Band interleaving not supported
  - Floating Point predictor not supported

```xml
<?xml version="1.0" encoding="UTF-8"?>
<wcs:GetCoverage
 xmlns:wcs="http://www.opengis.net/wcs/2.0"
 xmlns:wcsgeotiff="http://www.opengis.net/wcs/geotiff/1.0"
 xmlns:gml="http://www.opengis.net/gml/3.2"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="http://www.opengis.net/wcs/2.0 http://schemas.opengis.net/wcs/2.0/core/GetCoverage.html">
  <wcs:Extension>
    <wcsgeotiff:compression>JPEG</wcsgeotiff:compression>
    <wcsgeotiff:jpeg_quality>75</wcsgeotiff:jpeg_quality>
    <wcsgeotiff:predictor>None</wcsgeotiff:predictor>
    <wcsgeotiff:interleave>pixel</wcsgeotiff:interleave>
    <wcsgeotiff:tiling>true</wcsgeotiff:tiling>
    <wcsgeotiff:tileheight>256</wcsgeotiff:tileheight>
    <wcsgeotiff:tilewidth>256</wcsgeotiff:tilewidth>
  </wcs:Extension>
  <wcs:CoverageId>grey</wcs:CoverageId>
  <wcs:Format><image/tiff/></wcs:Format>
  <wcs:mediaType>multipart/related</wcs:mediaType>
</wcs:GetCoverage>
```
Current Status – WCS 2.0

- GML grid as output format
- Grid linear scan from Top-Left
- Single field per Band
  - We might improve names soon
- Notice the choice of axes

Names for the grid
- \(i,j,k\) as per columns, rows ordering
Current Status – NetCDF as Input

- **NetCDF Sample datasets have been analyzed**
  - Polyphemus (t,z,lon,lat grid)
  - Gome2 (lon, lat grid)
  - BlackForest (lon, lat grid) – small dataset
  - Eumetsat ASCAT (complex datasets on swath grids) – geolocated, not georectified

- **Additional NetCDF Samples from MetOc world have been collected**

- **Data which needs to be analyzed and added to the report**
  - OSTM Jason2 – 1D dataset (time)
  - Eumetsat IASI – (pressure level, surface emissivity, wavelength, surface temperature, cloud formation and irregular lon,lat grids) – need interpolation
  - Eumetsat Hyperspectral+sounding IASI – (huge dataset)
# Current Status – CITE Tests

## Test Suite: GCC WCS 2.0.0 Test Suite

<table>
<thead>
<tr>
<th>Test Suite</th>
<th>Test Case Name</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test case 1</td>
<td>main</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 2</td>
<td>get-kvp</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 3</td>
<td>post-xml-main</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 4</td>
<td>post-xml-core-req-1</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 5</td>
<td>get-kvp-core-req-1</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 6</td>
<td>post-xml-core-req-2</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 7</td>
<td>get-kvp-core-req-2</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 8</td>
<td>post-xml-core-req-3</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 9</td>
<td>get-kvp-core-req-3</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 10</td>
<td>post-xml-core-req-4</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 11</td>
<td>get-kvp-core-req-4</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 12</td>
<td>post-xml-core-req-5</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 13</td>
<td>get-kvp-core-req-5</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 14</td>
<td>post-xml-core-req-6</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 15</td>
<td>get-kvp-core-req-6</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 16</td>
<td>post-xml-core-req-7</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 17</td>
<td>get-kvp-core-req-7</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 18</td>
<td>post-xml-core-req-8</td>
<td>Passed</td>
</tr>
<tr>
<td>Test case 19</td>
<td>get-kvp-core-req-8</td>
<td>Passed</td>
</tr>
</tbody>
</table>

## Summary

- **Pass:** 16
- **Warning:** 0
- **Fail:** 0
Current Status – CITE Tests

- We opened 4 more tickets, but they have not been acknowledged yet
Current Status – WCS 2.0

- Unit/integration test coverage: 83% as reported by Cobertura

Coverage Report - All Packages

<table>
<thead>
<tr>
<th>Package</th>
<th># Classes</th>
<th>Line Coverage</th>
<th>Branch Coverage</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Packages</td>
<td>30</td>
<td>83%</td>
<td>61%</td>
<td>3,702</td>
</tr>
<tr>
<td>org.geoserver.wcs2_0</td>
<td>19</td>
<td>83%</td>
<td>67%</td>
<td>5,133</td>
</tr>
<tr>
<td>org.geoserver.wcs2_0.exception</td>
<td>3</td>
<td>64%</td>
<td>33%</td>
<td>2,167</td>
</tr>
<tr>
<td>org.geoserver.wcs2_0.kvp</td>
<td>11</td>
<td>87%</td>
<td>79%</td>
<td>3,389</td>
</tr>
<tr>
<td>org.geoserver.wcs2_0.response</td>
<td>11</td>
<td>85%</td>
<td>46%</td>
<td>3,133</td>
</tr>
<tr>
<td>org.geoserver.wcs2_0.util</td>
<td>4</td>
<td>74%</td>
<td>62%</td>
<td>4,167</td>
</tr>
<tr>
<td>org.geoserver.wcs2_0.xml</td>
<td>2</td>
<td>63%</td>
<td>N/A</td>
<td>1,667</td>
</tr>
</tbody>
</table>

Report generated by Cobertura 1.9.2 on 17/12/12 19.27.
**WPS: asynchronous calls**

- Asynchronous WPS support for long running processes

![Diagram showing the process flow of WPS:]

- **Client**
  - Execute
  - Status poll
  - Status poll
  - Status?
  - Results

- **WPS protocol handler**
  - Internal submit
  - Update status
  - Update status
  - Write results

- **WPS executor**

---

*Image by GeoSolutions*