INSPIRE KEN Follow-Up Webinar on Coverages & WCS

Peter Baumann
Jacobs University | rasdaman GmbH
baumann@rasdaman.com
Overview

- Part 1: Coverages / WCS Recap
- Part 2: The New Coverage Implementation Schema
- Part 3: Q&A
Part 1: Coverages / WCS Recap
Collecting Coverages

sensor feeds
[OGC SWE]

coverage server

simulation data
Serving Coverages

sensor feeds
[OGC SWE]

coverage server

simulation data
Service Orchestration

**SWE, SOS**: upstream sensor data capturing

**W*S**: downstream download, processing, visualization

WMS
WCS
WCPS
WPS

simulation data
(Part of) The OGC Standards Quilt

- **WMS**: "portrays spatial data → pictures"
- **WCS**: "provides data + descriptions; data with original semantics, may be interpreted, extrapolated, etc."
  
  [09-110r3]
Coverage Definition

class GML 3.2.1 Application Schema for Coverages

«FeatureType»
GML::Feature

«FeatureType»
Coverage

«Union»
GML::DomainSet

«Union»
GML::RangeSet

«type»
SWE Common::DataRecord

contains hook for metadata

from SWE Common

ISO 19123 is abstract → many different implementations possible → not per se interoperable

OGC coverage std is concrete and interoperable

[OGC 09-146r2]
Coverage & CRS
Coverage Encoding

- Pure GML: complete coverage, in GML
- Special Format: other suitable file format (ex: MIME type “image/tiff”)
- Multipart-Mixed: multipart MIME, type “multipart/mixed”
Sample Mixed Encoding: TIFF

- Multipart/related MIME
- Part 1: GML
- Part 2: eg, TIFF
- Consistency in metadata required
  - Otherwise bug

```xml
<?xml version="1.0" ...?>
<gmlcov:RectifiedGridCoverage ...
-<gml:domainSet> ...
  - <gml:rangeSet>
    - <gml:File>
      - <gml:rangeParameters xlink:href="grey.tif"
        xlink:role="http://www.opengis.net/spec/GMLCOV_geotiff-coverages/1.0"
        xlink:arcrole="fileReference"/>
      - <gml:fileReference>grey.tif</gml:fileReference>
      - <gml:fileStructure/>
      - <gml:mimeType>image/tiff</gml:mimeType>
    - <gml:File>
  - <gml:rangeSet>
    - <gmlcov:rangeType> ...
-<gmlcov:RectifiedGridCoverage>
```

Content-Type: image/tiff
Content-Description: coverage data
Content-Transfer-Encoding: binary
Content-ID: grey.tif
Content-Disposition: INLINE

...binary TIFF data...

--wcs--
WCS Suite Big Picture

Class WCS Suite Logical View

Abstract Topic 6

- CML
- SWE Common

Core

Coverage Implementation Schema

Format Encoding

- GeoTIFF
- netCDF
- JPEG2000
- GML/JP2
- JPIP
- CRIB2
- etc.

Data Model

- Quality
- Hierarchies

WCS-T

- Processing
- Range Subsetting
- Scaling
- CRS
- Interpolation

Service

Protocol Binding

- GET-KVP
- POST-XML
- SOAP
- REST
- JSON

Usability

- Multilinguality

Extensions

Application Profiles

data

service

EO-WCS

MetOcean-WCS
WCS Core GetCoverage: Trim & Slice

- Download a coverage [subset], values guaranteed unchanged
  - Delivery in "Native Format"

- Ex: "download coverage c001"
  
  http://www.acme.com/wcs?SERVICE=WCS & VERSION=2.0 & REQUEST=GetCoverage & COVERAGEID=c001

- Ex: "coverage c001, lat/long cutout, time slice t=2009-11-06T23:20:52"
  
  http://www.acme.com/wcs?SERVICE=WCS & VERSION=2.0 & REQUEST=GetCoverage & COVERAGEID=c001 & SUBSET=Long(100,120) & SUBSET=Lat(50,60) & SUBSET=time("2009-11-06T23:20:52")

- Ex: "coverage c001, lat/long slice, timeseries"
  
  http://www.acme.com/wcs?SERVICE=WCS & VERSION=2.0 & REQUEST=GetCoverage & COVERAGEID=c001 & SUBSET=Long(100) & SUBSET=Lat(60)
WCS Core GetCoverage: Format Encoding

- Download a coverage [subset], values guaranteed unchanged
  - ...if format allows!

- Ex: “coverage c001, in GeoTIFF”

  http://www.acme.com/wcs ? SERVICE=WCS & VERSION=2.0
  & REQUEST=GetCoverage & COVERAGEID=c001
  & FORMAT=“image/tiff“

- MIME types preferred, but there may be conventions
  - Ex: GDAL

- formats may define add‘l specific parameters
  - Ex: JPEG quality factor
WCS Extension – Processing [OGC 13-057]

- WCS wrapper for OGC Web Coverage Processing Service (WCPS)
  - high-level spatio-temporal geo raster query language

- "From MODIS scenes M1, M2, M3: difference between red & nir, as TIFF"
  - ...but only those where nir exceeds 127 somewhere

```plaintext
for $c$ in (M1, M2, M3) where some($c$.nir > 127) return encode($c$.red - $c$.nir, "image/tiff")
```

(tiff\textsubscript{A}, tiff\textsubscript{C})
Visualization-as-a-Query

for $s$ in (SatImage), $d$ in (DEM)
where $s$/metadata/@region = "Glasgow"
return
   encode(
       struct {
           red: (char) $s$.b7[x0:x1,x0:x1],
           green: (char) $s$.b5[x0:x1,x0:x1],
           blue: (char) $s$.b0[x0:x1,x0:x1],
           alpha: (char) scale( $d$, 20 )
       },
       "image/png"
   )
WCS Extension – CRS [OGC 11-053]

- retrieval (& bbox) in CRSs different from Native CRS
  - Extension to GetCoverage request
  - Capabilities document lists supported CRSs

- Recall: coverage = 1 datacube, with 1 CRS in domainSet
  - Possibly compound: horizontal, height, time, non-referenced, ...

- Ex: http://www.acme.com/wcs ? SERVICE=WCS & VERSION=2.0
  & REQUEST=GetCoverage & COVERAGEID=c001
  & SUBSETTINGCRS=http://www.opengis.net/def/crs/EPSG/0/4326
  & OUTPUTCRS=http://www.opengis.net/def/crs/EPSG/0/4326

- This needs
  - Compound CRSs → CRS NTS
  - New CRSs: vertical, time, index, proxies, ... → Time + Index CRS NTS [OGC 13-102r2]
    - Ex: underspecific ImageCRS → Index1D, Index2D, ...
Inset: CRS Name Types

- **WGS84, RESTful:**
  - [http://www.opengis.net/def/crs/EPSG/0/4326](http://www.opengis.net/def/crs/EPSG/0/4326)

- **WGS84, KVP:**

- **Parametrized (”AUTO“) CRSs:**
  - [http://www.opengis.net/def/crs?authority=OGC&version=1.3 & code=AUTO42003 & UoM=m & CenterLongitude=-100 & CenterLatitude=45](http://www.opengis.net/def/crs?authority=OGC&version=1.3 & code=AUTO42003 & UoM=m & CenterLongitude=-100 & CenterLatitude=45)

- **Ad-hoc combination of CRSs:**

- **Proprietary CRS definition:**
  - [http://www.acme.com/def/this-is-EPSG-4326](http://www.acme.com/def/this-is-EPSG-4326)

- **Inline CRS definition:**
  - `srsName="#crsdef"`
WCS CRS: Capabilities Retrieval

- With WCS-CRS: Capabilities doc contains list of supported CRSs

- Ex:

  ```xml
  <wcs:crsSupported>
  http://www.opengis.net/def/crs/EPSG/0/4326
  </wcs:crsSupported>
  <wcs:crsSupported>
  http://www.opengis.net/def/crs/EPSCG
  </wcs:crsSupported>
  <wcs:crsSupported>
  http://www.opengis.net/def/crs?authority=OGC&version=1.3&code=AUTO42003&UoM=m&CenterLongitude=-100&CenterLatitude=45
  </wcs:crsSupported>
  <wcs:crsSupported>
  http://www.acme.com/def/this-is-EPSCG-4326
  </wcs:crsSupported>
  <wcs:crsSupported>
  http://www.opengis.net/def/crs/OGC/0/AnsiDate
  </wcs:crsSupported>
  ```
WCS Adoption

- Large, growing implementation basis; known:
  - rasdaman, GDAL, GeoServer, MapServer, EOxServer, QGIS, OpenLayers, Leaflet, OPeNDAP, GMU, NASA WorldWind, …
  - Pyxis, ERDAS, ESRI ArcGIS, …

- proven in large-scale deployments
  - 130+ TB per single database
  - 1 query → 1,000+ cloud nodes

- Going ISO:
  - OGC CIS 1.1 → ISO 19123-2
  - OGC WCS → ISO WCS
Part 2: The New Coverage Implementation Schema, CIS 1.1
(under adoption in OGC & ISO)
Coverages: Key Features Revisited / Added

- Irregular grids: concise definition
  - Clarification on ReferenceableGrid, generalization of GML 3.3 GridByXXX
  - Sensor model support (SensorML 2.0)
  - Warped CRSs

- Interpolation: discrete vs continuous grids

- Interleaved representation → general partitioning scheme

- JSON prepared

- Spec renaming GMLCOV → CIS (adopted Spring 2015)
Managing Power

- Separate conformance classes for core, gridded and discrete data, partitioning, encoding

```plaintext
coverage-partitioning
  «depends-on» coverage
    «depends-on» grid-regular
    «depends-on» grid-irregular
    «depends-on» grid-transformation
    «depends-on» discrete-pointcloud
    «depends-on» discrete-mesh
  «depends-on» gml-coverage
    «depends-on» multipart-coverage
    «depends-on» other-format-coverage
    «depends-on» json-coverage
```

CIS::AbstractCoverage

Feature

+ coverageFunction :GML::CoverageFunction [0..1]
+ envelope :CIS::EnvelopeByAxis

+ domainSet
  «Data Type»
  CIS::DomainSet

+ rangeSet
  «Data Type»
  CIS::RangeSet

+ rangeType
  «Data Type»
  SWE Common :: DataRecord

+ rangeType
  «Data Type»
  CIS::Extension
  + any :any [0..*]

+ interpolationRestriction
  «Data Type»
  CIS::InterpolationRestriction
  + allowedInterpolation :anyURI [0..*]

+ metadata
  0..1

+ metadata
  0..1

«Data Type»
GML::DataBlock

«Data Type»
GML::File
Grid Types → Axis Types

- Observation: GML Rectified, Referenceable
to non-intuitive, hard to describe, cases missing

- CIS approach: axis types
  - Index axis: not georeferenced, IndexCRS
  - Regular axis: georeferenced, constant spacing
  - Irregular axis: georeferenced, variable spacing
  - Distorted axes: georeferenced, arbitrary grid point locations
  - Algorithmic grids: such as sensor model

- All combinations possible
  - GML 3.3 as special cases, SensorML 2.0 integrated
srsName is a Reference!

- „can we store WKT in srsName?“
  
  No. It is a reference. But it can refer to a WKT, or any other (!) type of definition. Not determined in GMLCOV / CIS.

- „can we reference non-canonical (=OGC) CRss?“
  
  Yes. It can refer to a WKT, or any other (!) type of definition.

```xml
<cis:domainSet>
  <cis:GeneralGrid srsName="http://www.opengis.net/def/crs/EPSG/0/4326" unmlabels="deg_deg" axislables="Lat Long" srsDimension="2">  
    <cis:regularGrid srsName="#crsdef" resolution="5"/>
    <cis:gridLimits>
      <cis:ind i="1" j="3"/>
      <cis:ind i="4" j="3"/>
    </cis:gridLimits>
  </cis:GeneralGrid>
</cis:domainSet>
```

srsName="http://www.acme.com/def/this-is-EPDG-4326"

srsName="#crsdef"
CIS::EnvelopeByAxis

see examples with CIS 1.1 schema

```xml
<cis:GeneralGridCoverage
 xmlns:cis='http://www.opengis.net/cis/1.1'
 xmlns:gml='http://www.opengis.net/gml/3.2'
 xmlns:swe='http://www.opengis.net/swe/2.0'
 xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
 xsi:schemaLocation= 'http://www.opengis.net/cis/1.1 ..//cisAll.xsd'
 gml:id="CIS_002">
 <cis:domainSet>
   <cis:GeneralGrid  srsName="http://www.opengis.net/def/crs/EPSG/0/4326"
   uomLabels="deg deg" axisLabels="Lat Long" srsDimension="2">
     <cis:regularAxis axisLabel="Lat" lowerBound="-90" upperBound="-80" resolution="5"/>
     <cis:regularAxis axisLabel="Long" lowerBound="0" upperBound="10" resolution="5"/>
     <cis:gridLimits srsName="http://www.opengis.net/def/crs/OGC/0/0/1/2D" axisLabels="i j">
       <cis:indexAxis axisLabel="i" lowerBound="0" upperBound="2"/>
       <cis:indexAxis axisLabel="j" lowerBound="0" upperBound="2"/>
     </cis:gridLimits>
   </cis:GeneralGrid>
 </cis:domainSet>
</cis:GeneralGridCoverage>
```
CIS::EnvelopeByAxis

see examples with CIS 1.1 schema

```
<cis:GeneralGridCoverage xmlns:cis="http://www.opengis.net/cis/1.1"
  xmlns:gml="http://www.opengis.net/gml/3.2"
  xmlns:swe="http://www.opengis.net/swe/2.0"
  xsi:schemaLocation="http://www.opengis.net/cis/1.1 ..//cisAll.xsd"
  gml:id="CIS_001">
  <cis:domainSet>
                   &l2=http://www.opengis.net/def/crs/OGC/0/AnsiDate"
                   uomLabels="deg deg m d" axisLabels="Lat Long h date" srsDimension="4">
      <cis:regularAxis axisLabel="Lat" lowerBound="-90" upperBound="-80" resolution="5"/>
      <cis:regularAxis axisLabel="Long" lowerBound="0" upperBound="180" resolution="5"/>
      <cis:irregularAxis axisLabel="h" lowerBound="0" upperBound="100" directPositions="0,100"/>
      <cis:irregularAxis axisLabel="date" lowerBound="2015-12-01" upperBound="2015-12-02" directPositions="2015-12-01, 2015-12-02"/>
      <cis:gridLimits srsName="http://www.opengis.net/def/crs/OGC/0/Index4D" axisLabels="i j k l"/>
    </cis:GeneralGrid>
  </cis:domainSet>
</cis:GeneralGridCoverage>
```
CIS::AbstractGridCoverage

CIS::AbstractGridCoverage

«Feature Type»
CIS::AbstractGridCoverage

+ domainSet

CIS::GeneralGridCoverage

«Feature Type»
CIS::GeneralGridCoverage

+ domainSet

CIS::Grid

«Data Type»
CIS::Grid

+ dimension :positiveInteger

CIS::GridCoverage

«Feature Type»
CIS::GridCoverage

+ envelope :GML::Envelope

CIS::RectifiedGridCoverage

«Feature Type»
CIS::RectifiedGridCoverage

+ envelope :GML::Envelope

GML::Grid

«Data Type»
GML::Grid

GML::RectifiedGrid

«Data Type»
GML::RectifiedGrid
CIS::C’ByPartitioning

[Diagram of class and feature relationships involving `CIS::CoverageByPartitioning`, `CIS::CoverageByDomainAndRange`, `CIS::AbstractCoverage`, `Data Type SWE Common: DataRecord`, and `CIS::Extension`, `CIS::InterpolationRestriction` classes with attributes and relationships such as `coverage`, `rangeType`, `metadata`, `interpolationRestriction`, `partition`, `domainSet`, and `rangeSet`.]

© 2016 rasdaman
CIS::C’ByPartitioning

See examples coming with CIS 1.1 schema, currently:
05_2D_index.xml; 10_2D_regular.xml;
20_3D_height.xml; 30_4D_height+time.xml;
40_1D_regular.xml; 45_2D_distorted.xml;
50_3D_partitioned.xml; 55_1D_timeseries-partitioned.xml; 56_3D_timeseries-multipart.xml;
57_1D_timeseries-interleaved.xml; 60_point-cloud.xml; 80_2D_interpolation.xml;
90_sensormodel.xml; z0_gridcoverage.xml;
z1_rectifiedgridcoverage.xml
CIS 1.1: Summary

- CIS 1.1 = compatible extension to GMLCOV 1.0
  - General grids; SensorML 2 integration; nonnumeric coordinates; "time-interleaved", interpolation; splitting into req classes
  - In sync with ISO TC211

- Concepts elaborated in T-11 (10 change requests)

- Spec + ATS ready, on pending >3 weeks
  - Approach: full copy GMLCOV → CIS (not inc, like GML 3.3)

- Implementations available for critical points:
  - Irregular grids: EarthServer project
  - Sensor model: KEYW
  - Partitioning: rasdaman and other array databases
  - GMLCOV 1.0 parts pre-existing, copied over

- Outlook: adding JSON
Sample Mixed Encoding: TIFF

- Multipart/related MIME
- Part 1: GML
- Part 2: eg, TIFF
- Consistency in metadata required
  - Otherwise bug

<?xml version="1.0" ...>
<gmlcov:RectifiedGridCoverage ...>
<gml:domainSet>...</gml:domainSet>
  <gml:rangeSet>
    <gml:File>
      <gml:rangeParameters xlink:href="grey.tif"
xlink:role="http://www.opengis.net/spec/GMLCOV_geotiff-coverages/1.0"
xlink:arcrole="fileReference"/>
      <gml:fileReference>grey.tif</gml:fileReference>
      <gml:fileStructure/>
      <gml:mimeType>image/tiff</gml:mimeType>
    </gml:File>
  </gml:rangeSet>
</gmlcov:RectifiedGridCoverage>

--wcs
Content-Type: image/tiff
Content-Description: coverage data
Content-Transfer-Encoding: binary
Content-ID: grey.tif
Content-Disposition: INLINE

...binary TIFF data...
--wcs--
OGC Coverage Types

- Spatio-temporal!

- MultiSolid Coverage
- MultiSurface Coverage
- MultiCurve Coverage
- MultiPoint Coverage
- Grid Coverage
- Rectified GridCoverage
- Referenceable GridCoverage

«FeatureType» Abstract Coverage as per GML 3.2.1
Part 3: Q&A
Questions

Q: Harmonize the CRS used in the “srsName” parameter for different GMLCOV file components.


Example uncorrelated with OGC coverages

OGC coverages offer simple, powerful (n-D!) way of handling space/time coordinates

- `boundedBy` is optional, may be different CRS, may be approximate; from GML!
- `domainSet` is mandatory, authoritative: contains Native CRS of coverage
- `domainExtent` is unknown to me (not found in GML 3.2.1)
Questions

- Elevation: Vertical CRS defining elevation values in rangeSet?
  - referenceFrame property from rangeType/swe:Quantity [OGC 08-094r1 SWE Common Data Model]

Use of swe:Quantity not governed by WCS.SWG! Best practice highly desirable. „Me no expert“ – who is willing to join a task force? Mail thread wrap-up?

[Emmanuel Devys et al]
DEM Example

Based on email discussion with Emmanuel Devys, Roger Lott, et al
Questions

- tiling / mosaicking, coverage aggregations within the GMLCOV files

- Build GMLCOV example for Elevation & Orthoimagery
  - standardized way to implement tiling (describing logical structures like e.g. mapsheets, administrative units like regions or districts, etc.)
  - mosaic elements (OI)
  - modelling coverage aggregations

Caveat: implementation detail, WCS interface std does not define implementation.

**COVERAGE ≠ IMAGE**

Do NOT standardize tiling on server!

Why should it be necessary?

Any serious server today can do seamless mosaicked maps

Meaning, purpose, how to serve?
Questions

- partial conceptual redundancies between INSPIRE coverages attributes and GMLCOV components / INSPIRE coverage model extensions
  - Ex: domainExtent vs. gml:boundedBy
    - agreed to minimize INSPIRE extensions as possible

- Issues:
  - Sometimes duplication of information
    - not much danger (consistency can be verified automatically)
    - Redundancy existing, eg, with format encodings
  - WCS2.0 ignores INSPIRE extensions - dangerous
Relevant Links

- **Wikipedia primers:**
  - [Coverages](#)
  - [Web Coverage Service](#)
  - [Web Coverage Processing Service](#)

- **OGC:**

- Coverage service standards online demo: [http://standards.rasdaman.com](http://standards.rasdaman.com)

- The rasdaman Array Database System

- The EarthServer initiative
That‘s all folks!
rasdaman: Agile Array Analytics

- „raster data manager“: SQL + n-D raster objects
  ```sql
  select img.green[x0:x1,y0:y1] > 130
  from LandsatArchive as img
  where avg_cells(img.nir) < 17
  ```

- Scalable parallel “tile streaming” architecture

- In operational use
  - OGC Web Coverage Service Core Reference Implementation