

# RELEASE OF DIGGS V1.1

David Burggraf (Galdos)  
Dan Ponti (USGS)  
Chris Bray (Keynetix)  
Loren Turner (Caltrans)



Presentation to the DIGGS Teams  
April 15, 2010

# Agenda



- Looking back, looking ahead (Turner)
- Task 1 findings, Task 2 objectives (Turner)
- Summary of changes in v1.1 (Ponti)
- Impacts of changes for users and developers (Burggraf)
- Schema and examples (Burggraf)
- Where to find DIGGS v1.1 and the examples (Bray)
- Domain-dependent v1.2 work – need your help (Turner)

# Looking Back...



- **March 2009** – Orlando DIGGS workshop
- **May 2009** – Contracts initiated with Galdos and Compusult to conduct review of DIGGS v1.0a.
- **July 2009** – v1.0a review work completed.
- **July 2009** – Governance committee published recommendations.
- **Oct 2009** – Synthesis of issues of v1.0a published.
- **Nov 2009** – New Project Plan presented.
- **Jan 2010** – Contract with Galdos initiated.
- **April 2010** – DIGGS v1.1 released.

# Looking Ahead...



- **May-June 2010** (est.) – Release DIGGS v1.2.
- **July 2010** (est.) – Tools provided to team to start using and reviewing DIGGS.
- **Sept 2010** (est.) – Release DIGGS v2.0.
- **Oct 2010** (est.) – Pilot applications using DIGGS.

# Task 1 Findings, Task 2 Objectives



- Domain-independent schema issues:
  - GML “Object-Property” Patterning
  - Inheritance of Objects
  - Use of GML Profiles
  - Organization of DIGGS Schemas
  - Relative Referencing of Schemas
  - Update to GML 3.2
- Domain-dependent issues:
  - Code Tables
  - Key Fields
  - Table Data
- Tools:
  - Validation Using Web Services
  - Schema Management Tools

# Summary of Changes in v1.1

- Fixed GML Object-Property rule
- Fixed import/includes – no longer need OASIS catalog
- Migrated DIGGS to GML 3.2
- Implemented a GML profile for DIGGS
- Reorganized to 5 namespace schemas with one file per namespace
- Implemented *gml:identifier* and *gml:id*
  - ▣ *gml:identifier* – for globally unique id (use URN); only applies to gml features (locations, projects, samples, layer systems, tests)
  - ▣ *gml:id* – for all features and objects for referencing (database handle); objects are complex types that are “recognized” as properties in gml.
- Removed unnecessary abstract types
- Implemented RIM for codelists in lieu of GML dictionaries.
  - ▣ Allows for single vocabulary with translations.

# Summary of Changes in v1.1

- Tabular data
  - retain existing structure, but implement code lists in v1.2 to restrict column types
  - removed generic table property
  - tables to be included under specific test features only
- Geometry
  - Projects – reference point, linear extent, areal extent
  - Linear referencing – gml method to reference positions in a borehole will be adopted in gml 3.3.
- Metadata
  - AssociatedFiles, Roles, Remarks, Specifications, Equipment, BusinessAssociates, Contracts are cast as gml metadata so that gml aware applications will recognize those objects as metadata.
  - No longer assigning metadata properties at the base level to prevent recursion.
  - All features carry associatedFile, roles, and remarks metadata properties.
  - All objects carry remarks metadata properties.
  - All tests are features and carry specifications and equipment metadata properties as default.
  - More work needed on this for v1.2.

# Impacts of Changes



## Validation of schemas

- ▣ V1.1 Complete.xsd took 1.5 seconds
  - with full schema checking on (all imported and included schemas)
  - Oxygen 10/XercesJ validator
- ▣ V1.0a complete.xsd took 90 seconds
  - with same validator, same settings, same environment

60x validation speed-up

# Impacts of Changes



## Complexity measure

- ▣ Metric: randomly generated instance file sizes
  - Generated <Diggs> root element using Oxygen 10
  - Maximum recursivity set to 3 levels
  - 5 instance files generated and averaged
- ▣ V1.0a
  - Average file size = 10397.65 MB
- ▣ V1.0a with GML Profile
  - Average file size = 355.74 MB (~30x decrease)
- ▣ V1.1
  - Average file size = 14.2 KB (~750,000x decrease)

# Impacts of Changes

## Generated Instance File Settings

Generate Sample XML Files

Schema Options

Namespace	Element
<ANY>	<ANY>

Add Edit Delete

Options

Namespace: <ANY>

Element: <ANY>

Generate optional elements

Generate optional attributes

Values of elements and attributes: Default (ignore restrictions)

Preferred number of repetitions: 1

Maximum recursivity level: 3

"Choice" and "Substitution Group"

Choice strategy: Random

Generate the other options as comments

OK Cancel

# Impacts of Changes

## Generated Instance File Settings

W3C XML Schema

URL:

Namespace:

Root Element:

Output folder:

Filename prefix:  Extension:

Number of instances:

Open first instance in editor

Namespaces

Default Namespace:

Prefix	Namespace
ns1	http://schemas.diggsml.com/1.0a/environmental
ns2	http://schemas.diggsml.com/1.0a/piling
ns3	http://www.witsml.org/schemas/131
ns4	http://schemas.diggsml.com/1.0a/geotechnical
ns5	http://schemas.diggsml.com/1.0a
ns6	http://schemas.diggsml.com/1.0a/monitoring
ns7	http://www.opengis.net/gml
ns8	http://www.w3.org/1999/xlink

OK Cancel

# DIGGS Schema and Examples



- Tour of schemas and instances in XML editor

# Where to find DIGGS v1.1

The screenshot shows the Microsoft Internet Explorer browser window displaying the DIGGS website. The address bar contains the URL <http://www.diggsml.com/>, which is circled in red. A callout bubble points to this URL with the text: "The schema files will be hosted for referencing at: <http://schemas.diggsml.com/1.1/>".

On the left side of the page, there is a "Main Menu" with several links. The "Schemas" link is circled in red, and a callout bubble points to it with the text: "DIGGS v1.1, documentation, and 20 example files can be downloaded here."

The main content area of the page features a search bar, a "View Edit Clone Export" toolbar, and a section titled "Data Interchange for Geotechnical and GeoEnvironmental Specialists (DIGGS)". Below this, there is a paragraph of text describing the organization and its goals. A "Recent Blog Entries (Full Blog Listing)" section is also visible, featuring a blog entry titled "Project Plan for the Development of DIGGS" by LTurner, dated Wed, 02/12/2009 - 08:42. The blog entry includes a link to the project plan document and a brief description of its content.

# Domain-Dependent Work

- DIGGS v1.2 work requires your active involvement in order to be successful.
- Business Rules
  - Salvatore hosted initial meeting to identify needs.
  - Sub-committee needed to develop a “business rules” document in layman terms.
  - Volunteers?
- Codelists
  - RIM approach implemented.
  - Sub-committee needed to review the existing codelists and implement consistently throughout schema.
  - Volunteers?