Workshop Data Modelling [en]
Thorsten Reitz, wetransform
INSPIRE and Beyond 2018
What is this Data Modelling session about?

- Why do we create data models at all?
- What processes can we use to create models?
- What tools can we use to design models?
- How can we re-use models effectively?
  - How to create Extensions?
  - How to create Profiles?

- Prerequisites:
  - Basic knowledge of object-oriented modelling
  - Basic knowledge of INSPIRE Data Specifications

- Tools:
  - Enterprise Architect
  - hale connect
The Harmonisation Process

Conceptual Harmonisation
Textual description
UML Model (EAP, XMI)
No Data

Data Harmonisation
XSD or Proprietary Model
Schema/Format converted Data

Service Harmonisation
OpenAPI or Proprietary Model
Final Format/API Data
Why do we create Data Models (in the Age of AI/ML)?

- Common Semantics & Understanding
- Structured data is computer-tractable with deterministic outcomes
- Requires less computing power + energy
- New interfaces (REST, WFS 3.0) and schemaless data don’t address:
  - Data Validation
  - Compile-time issue identification
Data Modelling in INSPIRE

- Key component of interoperability
  - Ready-to-use data with clearly defined semantics

- Model-driven approach
  - Platform-Independent Model
  - Focus on semantics and associations
What are the problems in data modelling?

- Over- and underspecification
- Flexibility vs. Rigidity
- Quality hard to assess
  - If you can’t measure it, how can you improve it?
- Long Feedback cycles
- Little implementation feedback during standards design
What are state of the art methods to create data standards?

- Example 1: WFS 3.0
- Example 2: i3S Community Standard

Source: [https://github.com/Esri/i3s-spec/](https://github.com/Esri/i3s-spec/)

Source: [https://github.com/opengeospatial/WFS_FES](https://github.com/opengeospatial/WFS_FES)
Takeaways from the Examples

- Don’t just collect lists of use cases, directly implement them, then abstract the solutions identified
- Clearly show that the proposed model works for its intended use
- Rapid feedback from multiple organisations and domains
- Rapid, continuous feedback cycles, iterations
- *Formal processes kick in at the end*
How do we create data models today?

- Agile Process over tools
  - How to include stakeholders?
  - How to solicit their ongoing input?
  - How to onboard all participants and to enable their input?
- Implementation over concepts
  - No analysis paralysis
  - Try things out
- Data over intuition
  - Measure quality of the data model and act on results
- Re-use and modularisation
  - Profiles
  - Extensions, Mix-Ins
Modelling
Using UML – in an Agile Environment?

- Encompasses more than 20 static and dynamic aspects of a system and its behaviour
- Object-Oriented foundation
- Graphical Design Language
- Commonly (mis-)understood semantics (e.g. Associations)
- Large Models
- Code Generation
- Model Transformation

- Advanced concepts hard to understand (e.g. Inheritance vs.)
- Isolated from implementation
- Sometimes hard to read
Model-Driven Architecture

Selected for INSPIRE because…:

- Consistent Approach
- Cross-Theme Associations
- Effective Re-Use
- Detection of Inconsistencies
- Automated generation of documentation
- Automated generation of schemas
How do we create common data models today?

- Agile Process over tools
  - How to include stakeholders?
  - How to solicit their ongoing input?
  - How to onboard all participants and to enable their input?
- Implementation over concepts
  - No analysis paralysis
  - Try things out
- Data over intuition
  - Measure quality of the data model and act on results
- Re-use and modularisation
  - Profiles
  - Extensions, Mix-Ins
Extensions & Mix-Ins
How can we use INSPIRE to support specific policy and business processes effectively?
- Observation: Extensions provide the opportunity to build new, efficient business processes on top on INSPIRE.

What are the steps to create an interoperable INSPIRE Extension?
- Observation: High uncertainty about the methodology towards an interoperable INSPIRE Extension
INSPIRE Extensions - Methodology

1. Organise a thematic working group
2. Collect Use Cases
3. Identify user requirements and Spatial Object types
4. Perform as-is analysis and gap analysis
5. Design a draft model
6. Test the model and consult stakeholders
7. Iterate over the model and feedback cycle
After release, maintain the model
INSPIRE Extensions - Tools
What is a profile?

- Specific rules for the usage of a data model
- Usually more restrictive than the base model
- Multiple Profiles can be applied to a data set

- Type Obligation
- Property Obligation and Cardinality
- Property Type Constraint
- Value Constraint
- Conditional value constraints
- Conditional existence constraints

- Tags/Tagged Values

- Exported as a...
  - Testsuite (e.g. Schematron, BaseX/ETF)
  - XMI, XSD, ....
Enterprise Architect as Profile Creation Tool

- Add (OCL) Constraints to Model
- No formalism for tags, OCL constraints hard to edit
- Property Obligation and Cardinality
  - Requires change to model
- Property Type Constraint
- Value Constraint
- Conditional value constraints
- Conditional existence constraints
- Tags/Tagged Values
- Exported as a...
  - XMI
  - Java Test Suite
- Open Source Project developed by interactive instruments on top of Shapechange
- Input: UML model (XMI/EA)
- Type Obligation
- Property Obligation and Cardinality
- Tags/Tagged Values
- Limited to GML Application Schemas (not ISO schemas)
- Exported as a...
  - XMI (UML Model)
hale connect profile implementation

- Part of the hale connect SaaS
- Works with any model
- Type Obligation
- Property Obligation and Cardinality
- Property Type Constraint
- Value Constraint
- Conditional value constraints
- Conditional existence constraints
- Tags/Tagged Values
- Exported as a...
  - BaseX/ETF Test
  - XML Schema
#INSPIREMadeEasy

+49 6151 155 408
info@wetransform.to
www.wetransform.to

www.linkedin.com/company/wetransform-gmbh
@wetransformto