



User and Community Driven Framework Development

HUMBOLDT Open Day
Lisboa, September 28th, 2010
Thorsten Reitz, Fraunhofer IGD

Overview

- ▣ The HUMBOLDT Community and the Community Website
- ▣ Collaboration with the Community
 - ▣ Example: 1spatial – development of HALE plugins
- ▣ Lessons Learned
 - ▣ The HUMBOLDT Approach revisited
 - ▣ Establishing a working community

Who is the Community?

- ▣ You!
 - ▣ The users and developers in the HUMBOLDT scenarios
 - ▣ GENESIS: Sharing of Methodology
 - ▣ GS-Soil: INSPIRE harmonisation and transformation, tools reuse
 - ▣ ESDIN: HALE and CST dissemination, feedback
 - ▣ Geoland 2: establish collaboration agreement
 - ▣ Plan4all: NDA for document exchange for WP2 results in preparation
 - ▣ NatureSDI+: HUMBOLDT training planned
 - ▣ eSDI-Net+: Networking, joint meetings
 - ▣ GIGAS: Contributions to the Technology Watch activities
 - ▣ INSPIRE Data Specification Drafting Teams
 - ▣ Snowflake, 1spatial, FME, ...

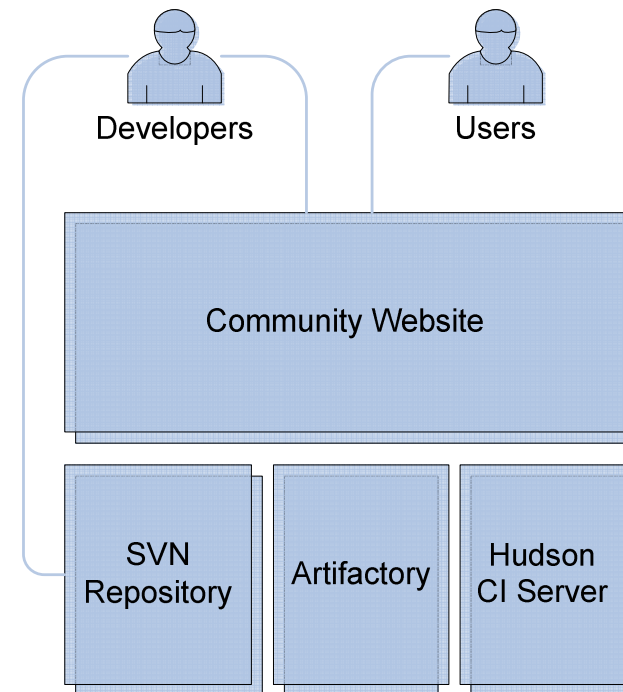
Sub-Communities?

1. User Community

1. End Users of Spatial Information
2. Decision and Policy Makers

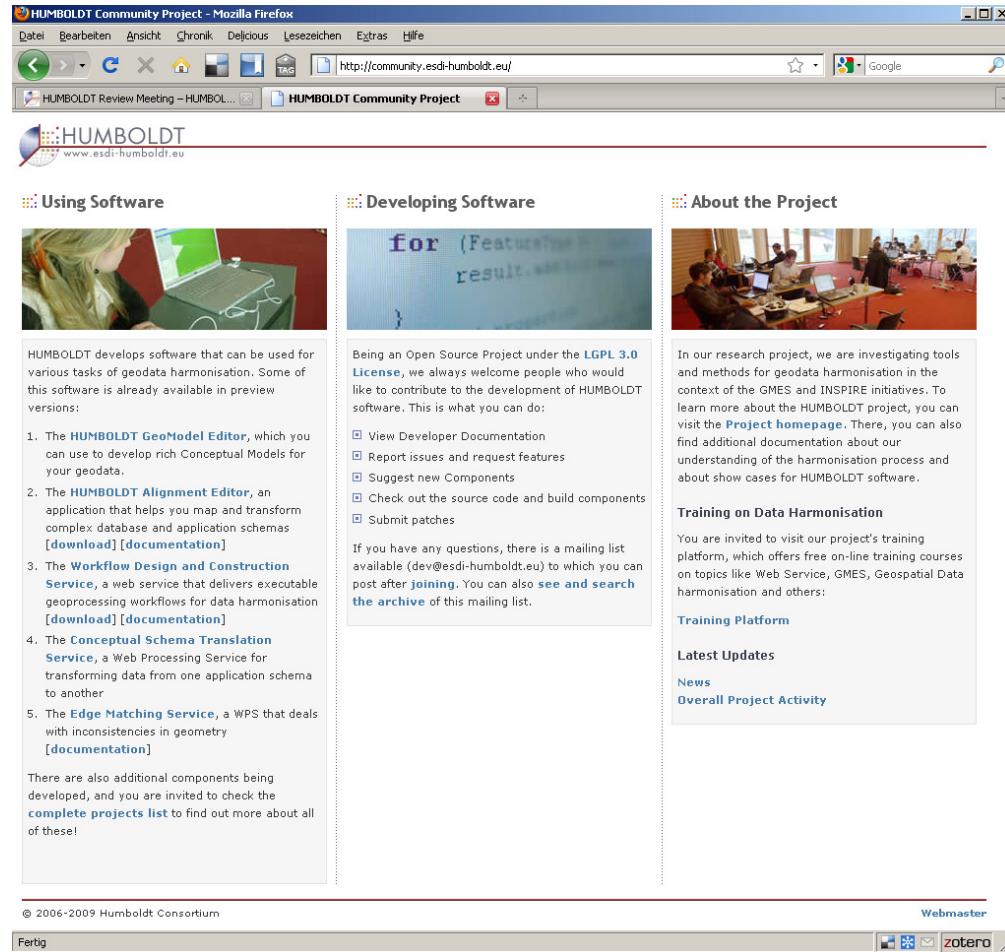
2. Developer Community

1. End Users of Data Harmonisation Tools
(Data Integrators and Custodians)
2. Developers



Goals of a Developer Community

- ▣ Download software
- ▣ Access documentation
- ▣ Capture requirements
- ▣ Report bugs
- ▣ Progress tracing
- ▣ Discuss issues, build a knowledge base of solutions



Using Software

HUMBOLDT develops software that can be used for various tasks of geodata harmonisation. Some of this software is already available in preview versions:

1. The **HUMBOLDT GeoModel Editor**, which you can use to develop rich Conceptual Models for your geodata.
2. The **HUMBOLDT Alignment Editor**, an application that helps you map and transform complex database and application schemas [[download](#)] [[documentation](#)]
3. The **Workflow Design and Construction Service**, a web service that delivers executable geoprocessing workflows for data harmonisation [[download](#)] [[documentation](#)]
4. The **Conceptual Schema Translation Service**, a Web Processing Service for transforming data from one application schema to another
5. The **Edge Matching Service**, a WPS that deals with inconsistencies in geometry [[documentation](#)]

There are also additional components being developed, and you are invited to check the [complete projects list](#) to find out more about all of these!

© 2006-2009 Humboldt Consortium

Developing Software

Being an Open Source Project under the **LGPL 3.0 License**, we always welcome people who would like to contribute to the development of HUMBOLDT software. This is what you can do:

- ▣ View Developer Documentation
- ▣ Report issues and request features
- ▣ Suggest new Components
- ▣ Check out the source code and build components
- ▣ Submit patches

If you have any questions, there is a mailing list available (dev@esdi-humboldt.eu) to which you can post after [joining](#). You can also [see and search the archive](#) of this mailing list.

About the Project

In our research project, we are investigating tools and methods for geodata harmonisation in the context of the GMES and INSPIRE initiatives. To learn more about the HUMBOLDT project, you can visit the [Project homepage](#). There, you can also find additional documentation about our understanding of the harmonisation process and about show cases for HUMBOLDT software.

Training on Data Harmonisation

You are invited to visit our project's training platform, which offers free on-line training courses on topics like Web Service, GMES, Geospatial Data harmonisation and others:

[Training Platform](#)

Latest Updates

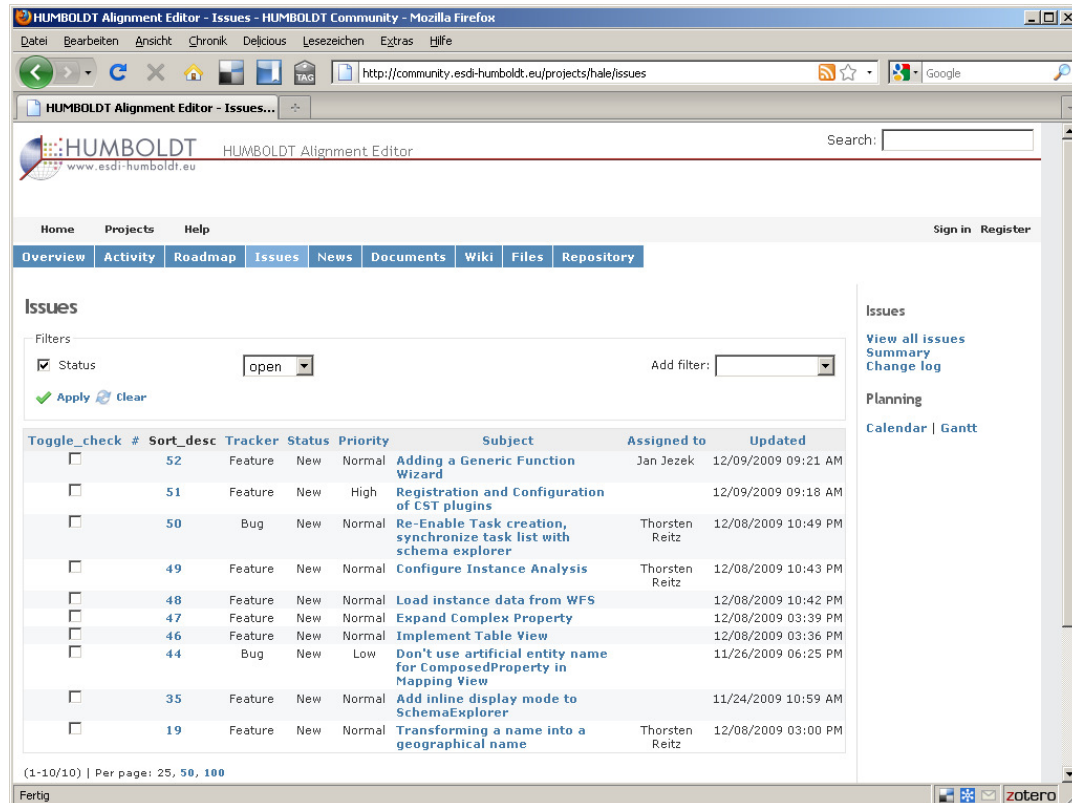
[News](#)
[Overall Project Activity](#)

Webmaster

Fertig

Handling Community Input

- ▣ Processing of user requirements in the technical team
 - ▣ Primary Inputs: Community Forum (preferred!), Community Bugtracker, E-Mail, specification
 - ▣ Feedback from reporter if required
 - ▣ Assignment to developer and milestone
- ▣ Resolution of ticket
- ▣ Closing ticket if resolution is confirmed by reporter



HUMBOLDT Alignment Editor - Issues - HUMBOLDT Community - Mozilla Firefox

http://community.esdi-humboldt.eu/projects/hale/issues

HUMBOLDT Alignment Editor

Home Projects Help Sign in Register

Overview Activity Roadmap Issues News Documents Wiki Files Repository

Issues

Filters

Status Add filter:

Apply

Toggle_check	#	Sort_desc	Tracker	Status	Priority	Subject	Assigned to	Updated
<input type="checkbox"/>	52		Feature	New	Normal	Adding a Generic Function Wizard	Jan Jezek	12/09/2009 09:21 AM
<input type="checkbox"/>	51		Feature	New	High	Registration and Configuration of CST plugins		12/09/2009 09:18 AM
<input type="checkbox"/>	50		Bug	New	Normal	Re-Enable Task creation, synchronize task list with schema explorer	Thorsten Reitz	12/08/2009 10:49 PM
<input type="checkbox"/>	49		Feature	New	Normal	Configure Instance Analysis	Thorsten Reitz	12/08/2009 10:43 PM
<input type="checkbox"/>	48		Feature	New	Normal	Load instance data from WFS		12/08/2009 10:42 PM
<input type="checkbox"/>	47		Feature	New	Normal	Expand Complex Property		12/08/2009 03:39 PM
<input type="checkbox"/>	46		Feature	New	Normal	Implement Table View		12/08/2009 03:36 PM
<input type="checkbox"/>	44		Bug	New	Low	Don't use artificial entity name for ComposedProperty in Mapping View		11/26/2009 06:25 PM
<input type="checkbox"/>	35		Feature	New	Normal	Add inline display mode to SchemaExplorer		11/24/2009 10:59 AM
<input type="checkbox"/>	19		Feature	New	Normal	Transforming a name into a geographical name	Thorsten Reitz	12/08/2009 03:00 PM

(1-10/10) | Per page: 25, 50, 100

Fertig

Issues

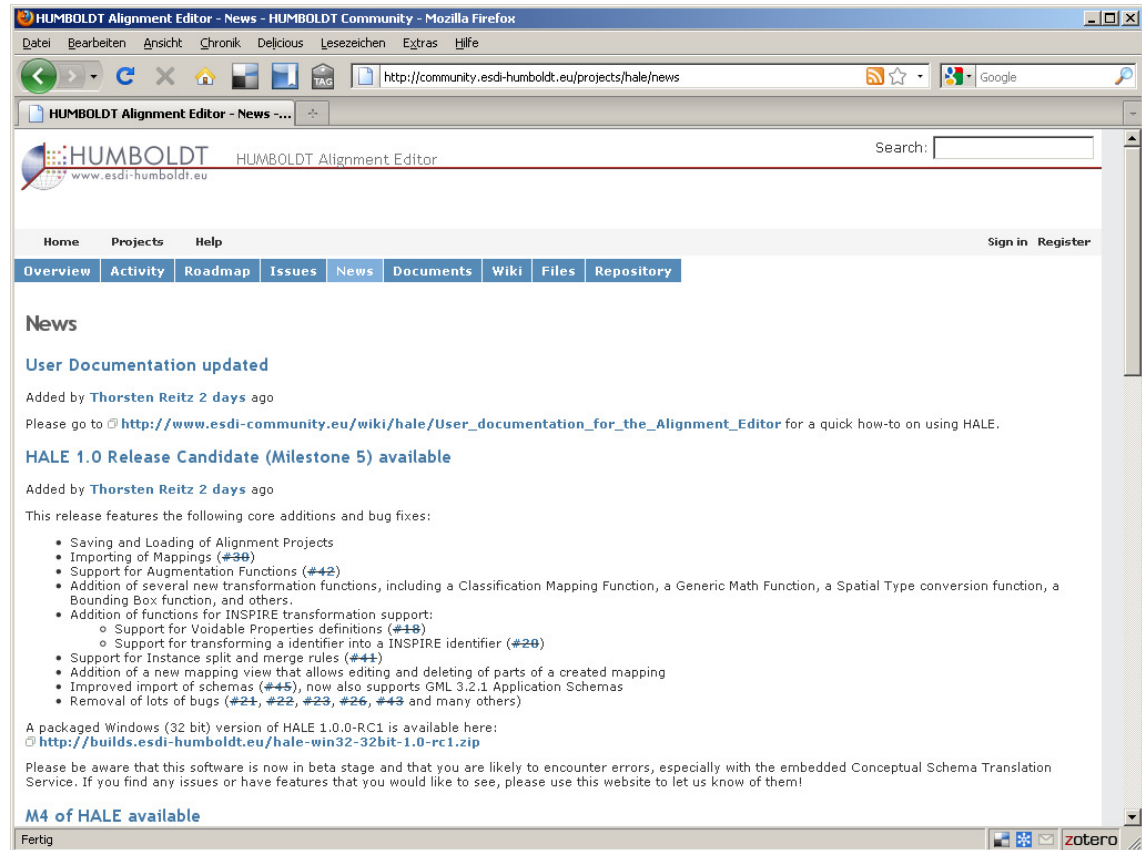
[View all issues](#)
[Summary](#)
[Change log](#)

Planning

[Calendar](#) | [Gantt](#)

Community Impact

- ▣ More than 220 tickets managed through this process (about 30 from external users)
- ▣ Status 24.09.2010:
191 closed, 35 open
- ▣ Average ticket closing time:
8 days (maximum: 221 days, minimum: 1 day)
- ▣ Registered external users:
~35
- ▣ Component Downloads:
~450



The screenshot shows a web browser window displaying the 'HUMBOLDT Alignment Editor - News' page. The browser's address bar shows the URL <http://community.esdi-humboldt.eu/projects/hale/news>. The page content includes a search bar, navigation tabs (Overview, Activity, Roadmap, Issues, News, Documents, Wiki, Files, Repository), and a 'News' section. The news items listed are:

- User Documentation updated**
Added by Thorsten Reitz 2 days ago
Please go to http://www.esdi-community.eu/wiki/hale/User_documentation_for_the_Alignment_Editor for a quick how-to on using HALE.
- HALE 1.0 Release Candidate (Milestone 5) available**
Added by Thorsten Reitz 2 days ago
This release features the following core additions and bug fixes:
 - Saving and Loading of Alignment Projects
 - Importing of Mappings (#30)
 - Support for Augmentation Functions (#42)
 - Addition of several new transformation functions, including a Classification Mapping Function, a Generic Math Function, a Spatial Type conversion function, a Bounding Box function, and others.
 - Addition of functions for INSPIRE transformation support:
 - Support for Voidable Properties definitions (#18)
 - Support for transforming a identifier into a INSPIRE identifier (#20)
 - Support for Instances split and merge rules (#44)
 - Addition of a new mapping view that allows editing and deleting of parts of a created mapping
 - Improved import of schemas (#45), now also supports GML 3.2.1 Application Schemas
 - Removal of lots of bugs (#21, #22, #23, #26, #43 and many others)

A packaged Windows (32 bit) version of HALE 1.0.0-RC1 is available here:
<http://builds.esdi-humboldt.eu/hale-win32-32bit-1.0-rc1.zip>

Please be aware that this software is now in beta stage and that you are likely to encounter errors, especially with the embedded Conceptual Schema Translation Service. If you find any issues or have features that you would like to see, please use this website to let us know of them!

M4 of HALE available

The browser status bar at the bottom shows 'Fertig' and the Zotero extension icon.

Community Contributions

- ▣ Testing and Evaluation of Components
 - ▣ German AAA Data
 - ▣ ISO 19115 Metadata
 - ▣ ESDIN
 - ▣ ...
- ▣ Contribution of bugfixes and platform compatibility fixes
- ▣ Contribution of extensions
 - ▣ 1spatial RIF exporter for HALE



Collaboration with 1spatial

The 1 spatial collaboration

- ▣ INSPIRE has:
 - ▣ Implementing Rules requiring Technical Guidance
 - ▣ Network Services IR for Transformation of
 - ▣ Coordinate Systems, Natural Languages, File Formats, Geometries and Schemas

- ▣ JRC contract awarded in Nov 2009 to develop:

Technical Guidance for INSPIRE
Schema Transformation Network Services

Architectural Goals

- ▣ Be flexible and vendor neutral
 - ▣ Promote the adoption of capable open standards for schema description and model mapping definitions
- ▣ Decouple transformation tasks
 - ▣ Model mapping rule definition
 - ▣ Schema transformation execution
- ▣ Define open interfaces
 - ▣ Meet generic INSPIRE schema transformation needs

Project Achievements

- ▣ Conducted State of the Art Analysis
 - ▣ Evaluated existing standards, tools & projects
 - ▣ Identified HUMBOLDT (HALE & CST)
- ▣ Developed Technical Guidance, recommending
 - ▣ GML for schema descriptions
 - ▣ RIF for model mapping definitions
- ▣ Produced Prototype Demonstrator & Video
 - ▣ Used HALE alongside other open source, freeware and commercial products to prove feasibility of guidance in vendor neutral service environment, interoperable through the use of open standards.



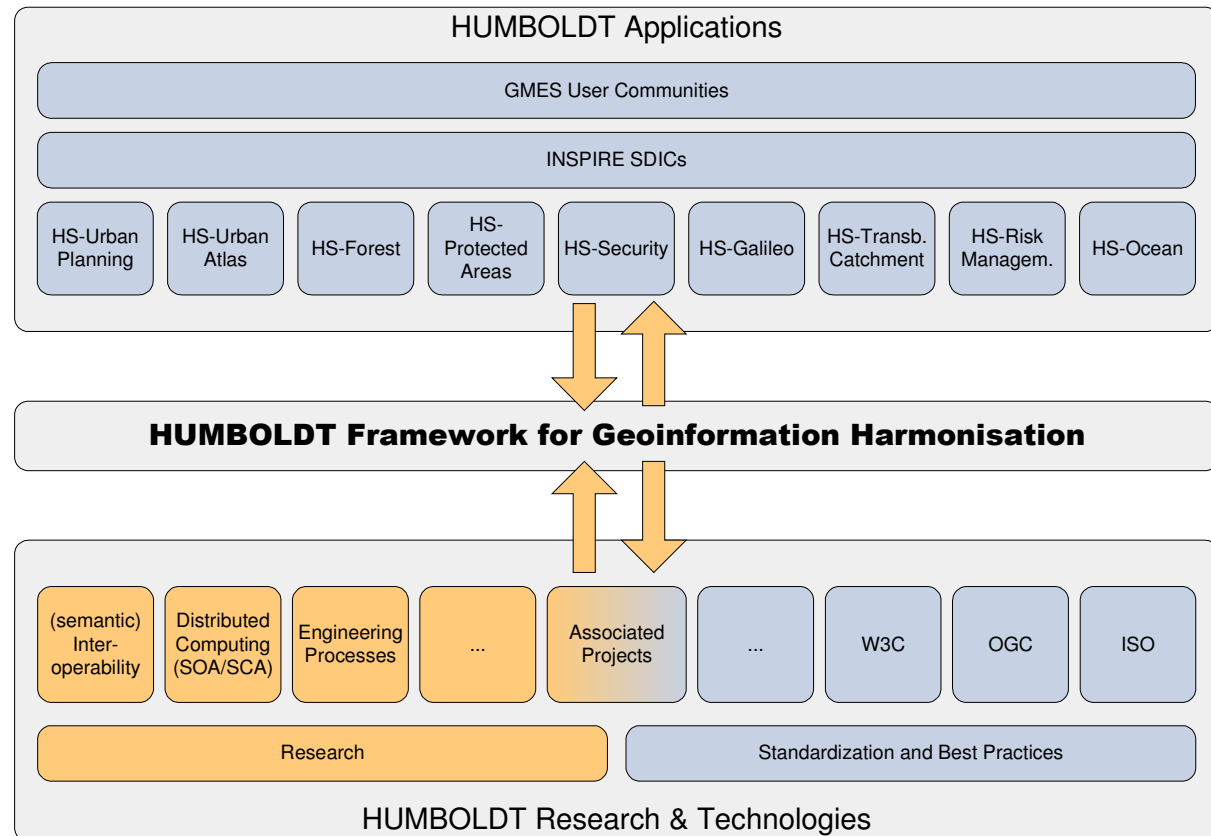
Lessons learned from working with the community

Lessons Learned – The Approach

- ▣ The HUMBOLDT Approach revisited:
 - ▣ R&T Drivers
 - ▣ Application Drivers

- ▣ Technical drivers were stronger throughout the first half of the project

- ▣ Application drivers often felt detached from development work



Lessons Learned – Timing and Work Plan

- ▣ Iterations in development and specifications
 - ▣ Originally, an agile development approach was foreseen, but could not be adopted to the different partner cultures and skills – *this risk was not identified early enough!*
 - ▣ Fallback to “big” Framework versions with intervals of six to nine months led to detachment from end user requirements and working too much on “invisible” parts
 - ▣ In the end, component-centered agile release plans were adopted, which worked reasonably well

- ▣ Establishment of the Developer Community should have happened earlier
 - ▣ At least for project-internal purposes, community-like structures could have been established after six to twelve months

- ▣ Decoupling of development reduces communication effort somewhat

Lessons Learned – Communication

- ▣ Developer-2-User Communication:
 - ▣ Rapid Prototyping (Specific, small sub-scenarios as prototypes for the demonstrators (even smaller than the Fast Track Scenario))
 - ▣ Focus on implementation of a single use case with well-known harmonisation issue
 - ▣ Require clear identification of critical harmonisation issues

- ▣ Developer-2-Developer Communication:
 - ▣ Different development cultures in each organisation
 - ▣ Highly distributed team, often with isolated single developers per organisation
 - ▣ Insufficient co-ordination so that all developers are available in the same time slots (“sprints”)
 - ▣ Winter of Code Workshops

Lessons Learned – Skills and Tasks

- ▣ Harmonisation projects require a set of skills that those involved possess:
- ▣ Required Skills on the User side:
 - ▣ Basic Data Modeling (e.g. UML) and analysis,
 - ▣ (Geospecific) standards,
 - ▣ Writing/Editing User stories, use cases and requirements,
 - ▣ Ideally users with a cross-discipline background
- ▣ Required Skills on the Developer side:
 - ▣ Project infrastructure (especially build and CI systems),
 - ▣ (Geospecific) standards and concepts,
 - ▣ Libraries
- ▣ Required Skills on the project management side:
 - ▣ Ensure continuous communication between users and developers,
 - ▣ Conflict management

Lessons Learned – Trust

- ▣ Earning User Trust
 - ▣ Ensure Transparency in specification processes and even more in development work
 - ▣ Start with high-visibility items that are relatively cheap to solve, not with “back office” parts that take long to implement

- ▣ Earning Developer Trust
 - ▣ Clearer commitment of users to evaluation and testing
 - ▣ Willingness to try to understand specifications and documentation