

QuakeML

XML concepts for a European
seismological data exchange
infrastructure

Fabian Euchner (ETH Zürich)
Danijel Schorlemmer (USC, Los Angeles)
and the *QuakeML working group*
(ETH, GFZ, USC, USGS, IRIS)

QuakeML – Design Principles

- Intended to cover a broad range of seismological fields
- Support real-time data transfer of seismic events
- Designed as data exchange format, independent of further persistent storage
- Modular approach:
 - Basic event data
 - Resource metadata
 - Inventory
 - ...
- Written from scratch, no adaption of existing (DB-) schema, uses full XML flexibility
- Community-driven development of standards
 - Documents are first discussed in working group, then subjected to **Request for Comments** process
 - Sequence of document maturity levels: Working Draft, Proposed Recommendation, Recommendation

QuakeML – Definition and Standardization

- SED, GFZ and others are in the process of defining a set of QuakeML 1 elements (6 meetings held so far)
- Request for Comments document available May 2007
- Further QuakeML development will be based at SED
- SeisComP 3 and ZMAP++ use QuakeML for internal communication
- Already developed toolchain for automated data structure generation and code generation
- Standard UML 2.0 schema description

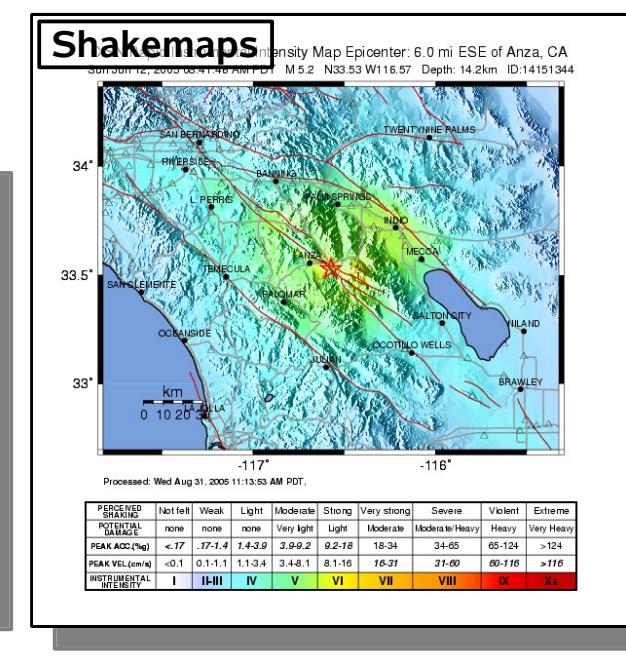
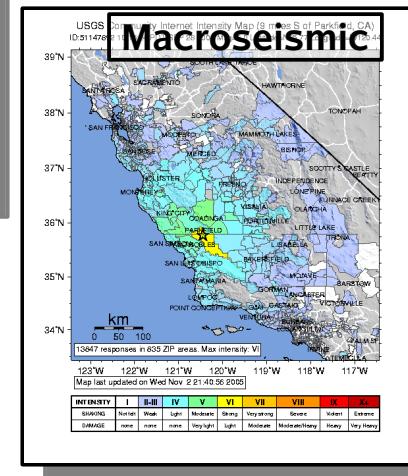
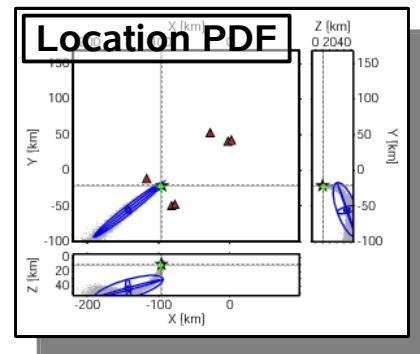
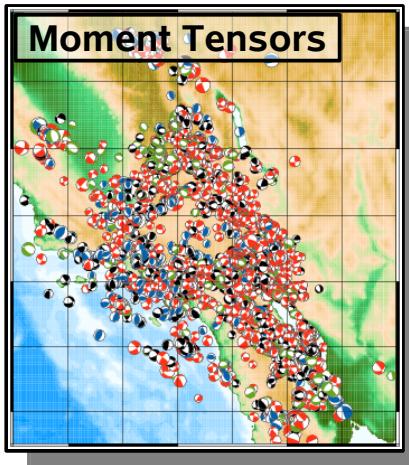
QuakeML – Development Roadmap

QuakeML development roadmap

Basic event description
(incl. moment tensors)
Metadata infrastructure
Inventory

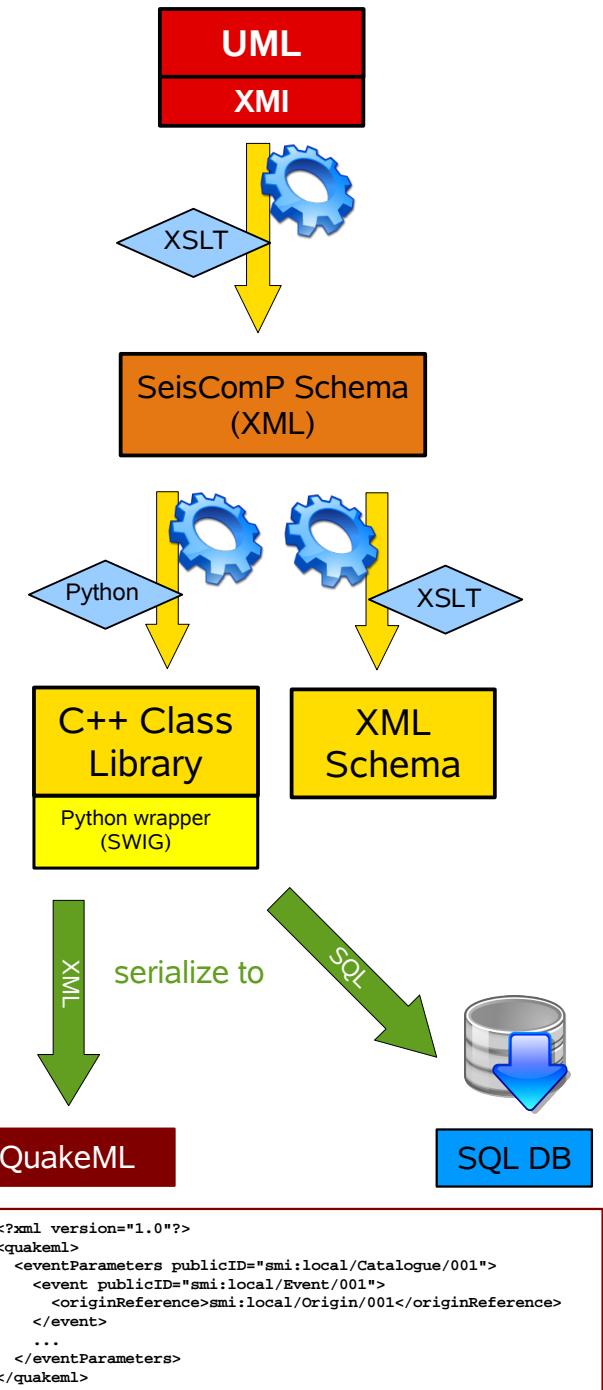
RFC starts May 2007
RFC starts Summer 2007
RFC starts ?

Further development: location PDF (Nlloc), macroseismic information, slip distribution, shakemaps, ...



QuakeML – From Data Model to XML

- QuakeML data model available as UML class diagram (and XMI)
- We will maintain and distribute a software library for QuakeML:
 - Use XMI for automated code generation
 - Library native in C++, can be used from other object-oriented languages (Python, Java) using wrappers
 - Objects can be serialized as XML (QuakeML) or SQL

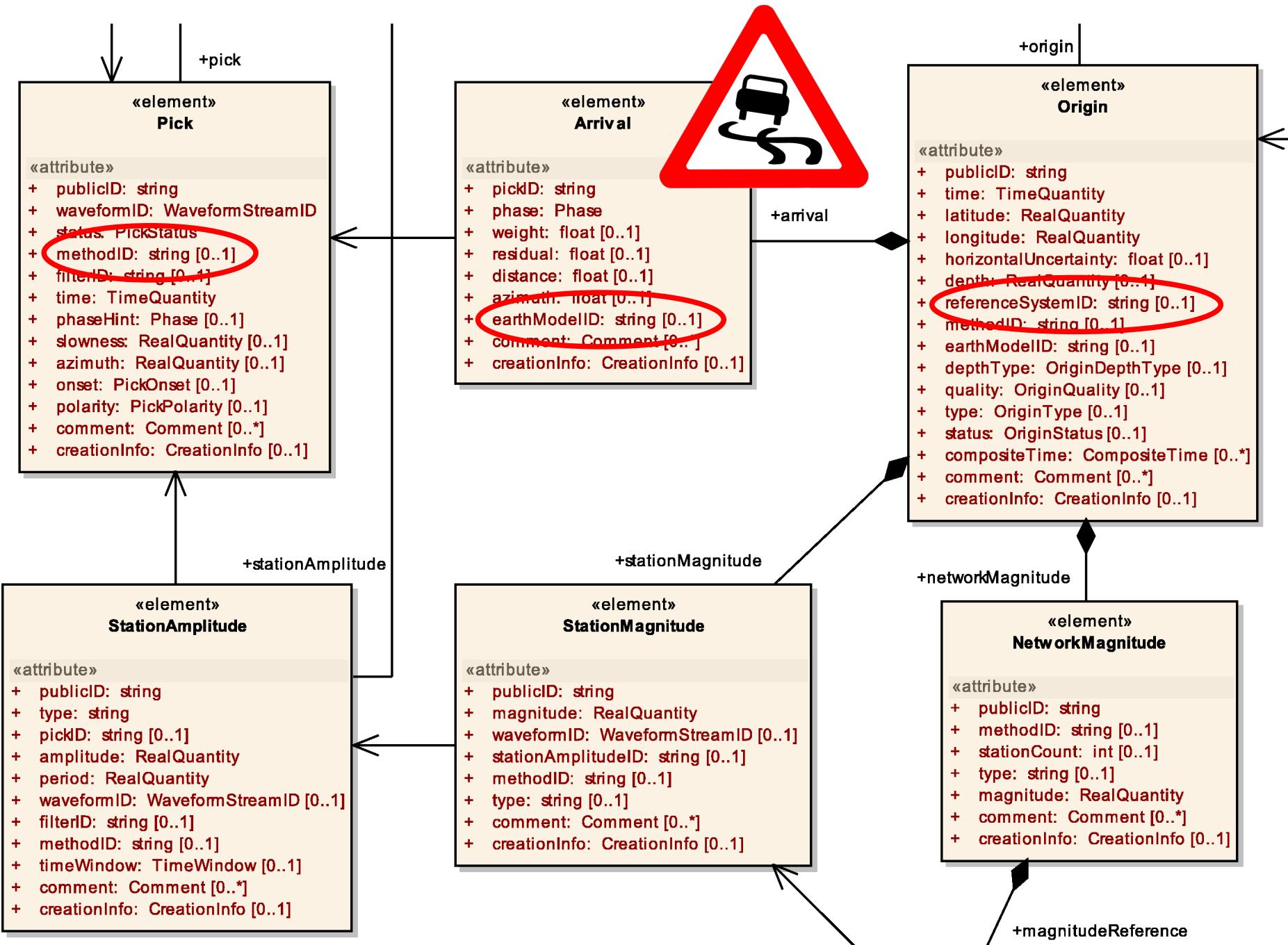


QuakeML – Code Generator

- creates C++ class library, XML Schema, and SQL DB schema from [UML/XMI](#) data model
- creates [Python](#) wrappers using [SWIG](#) toolkit (wrappers for other OO languages can be added)
- written in Python, uses [Cheetah](#) template engine
- open source (GPL)
- originally developed by GFZ as a part of [SeisComP](#), many dependencies on SeisComP components
- a stand-alone version will be provided with the QuakeML standard documents and Schema
- QuakeML code generator/class library enable users to work with QuakeML without having to write own code
- code generator toolchain allows to create own, site-specific [extensions](#) of QuakeML

Metadata/Data Problem

<QuakeML>



Data Exchange Infrastructure & Resource Identifiers

Requirements for seismological information exchange in a global network

- unambiguous, persistent **identifiers** of networked resources
- rich standardized metadata description (Dublin Core)
- resource discovery by standardized query gateways
- information retrieval using standardized services and protocols

QuakeML supports unique, location-independent **identifiers** for resources in a seismological network (in URI format)

`smi:<authority-id>/<resource-key>[#<local-id>]`

URI schema **smi** stands for **seismological meta-information**

Example: SED autopicker

`smi:ch.ethz.sed/Software/Picker/Autopicker/6.0.5`

Analogy: `doi:10.1051/0004-6361:20064840`

`http://dx.doi.org/10.1051/0004-6361:20064840`

Data Exchange Infrastructure & Resource Identifiers

A **registry** is a resource that stores and provides metadata about other resources

- provides a **Web Service** which resolves **identifiers** into a **RDF metadata description**
- vocabulary will be based on Dublin Core, with extensions for seismological resources

Metadata will cover the resource's

- identity (name and description)
- curation (who maintains it?)
- content (what kind of information?)
- data quality
- service (how to retrieve the resource?)



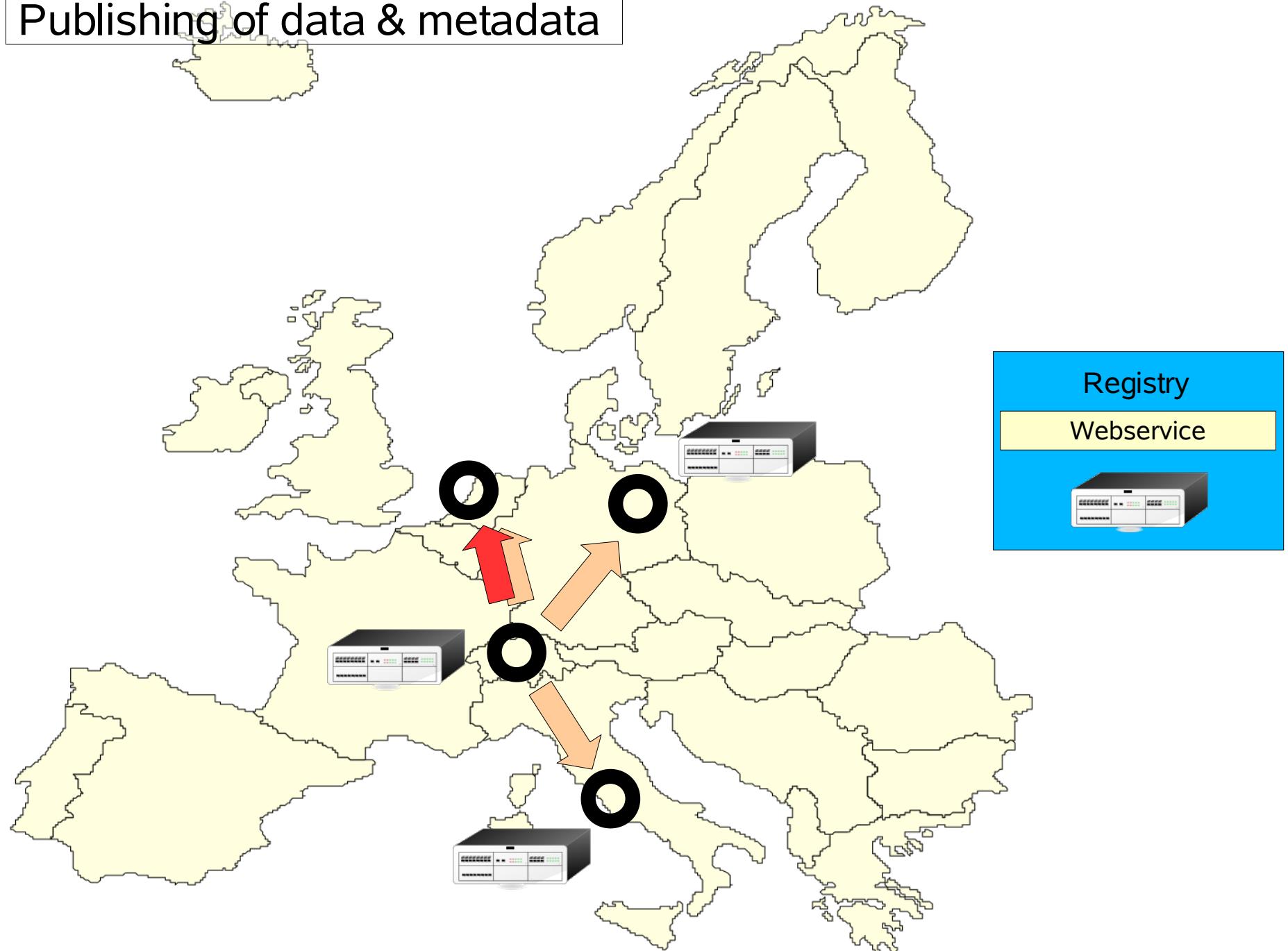
QuakeML Resource Metadata

The **RDF metadata** contain information on how to retrieve the **resource data**, e.g., URLs, pointers to Web Service descriptions (WSDL), etc.

- if the resource data are in **QuakeML format**, they can be retrieved via registries with a short-cut identifier
`quakeml:<authority-id>/<resource-key>[#<local-id>[?<parameters>]]`
- URI schema **quakeml** indicates that resource data are available in QuakeML format
- additional **<parameters>** may be required to retrieve the resource data (e.g., waveform data for a time interval)
- **<parameters>** part is in URL parameter format:
`par1=value1&par2=value2&...`

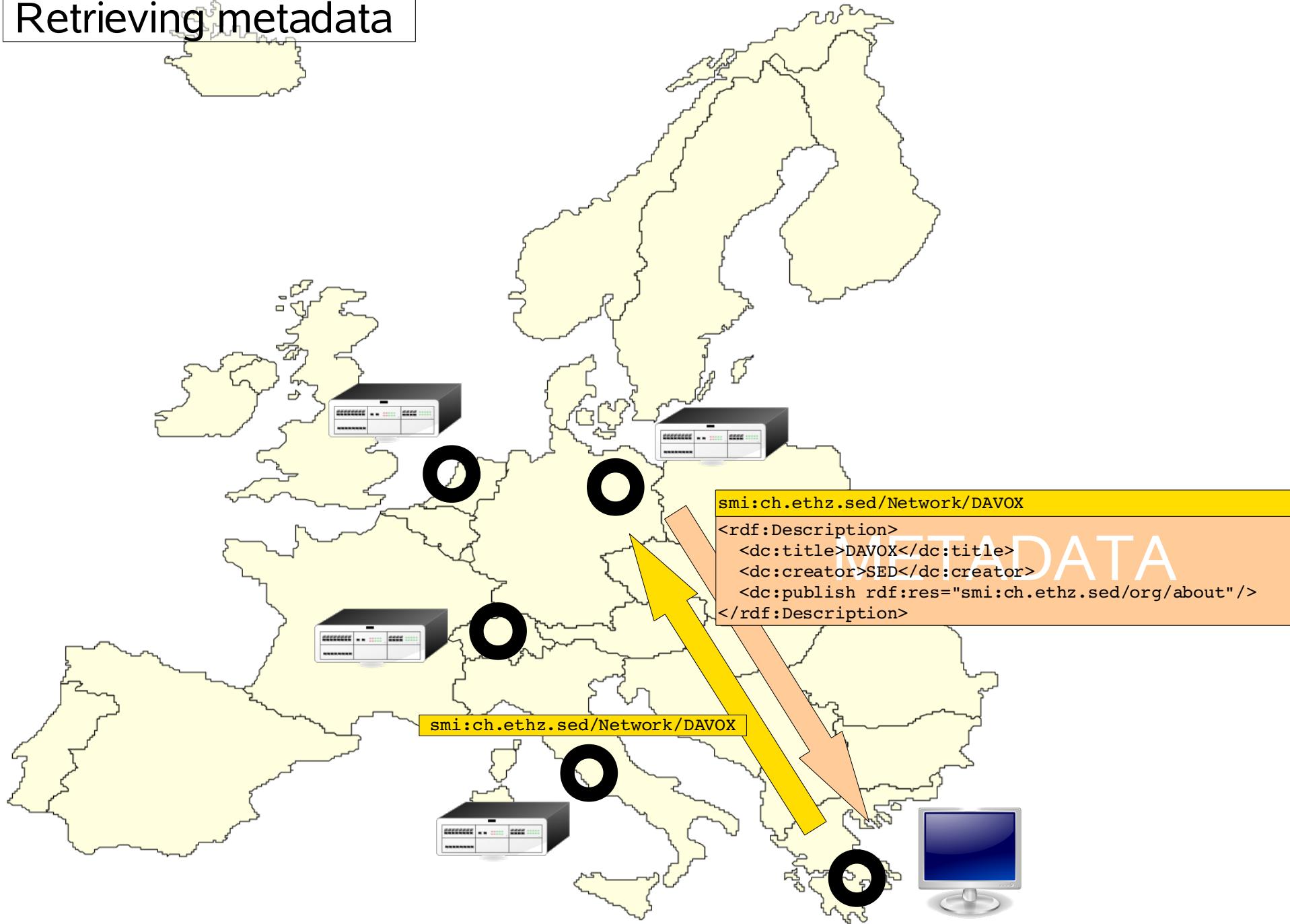
Metadata & Data Exchange

Publishing of data & metadata



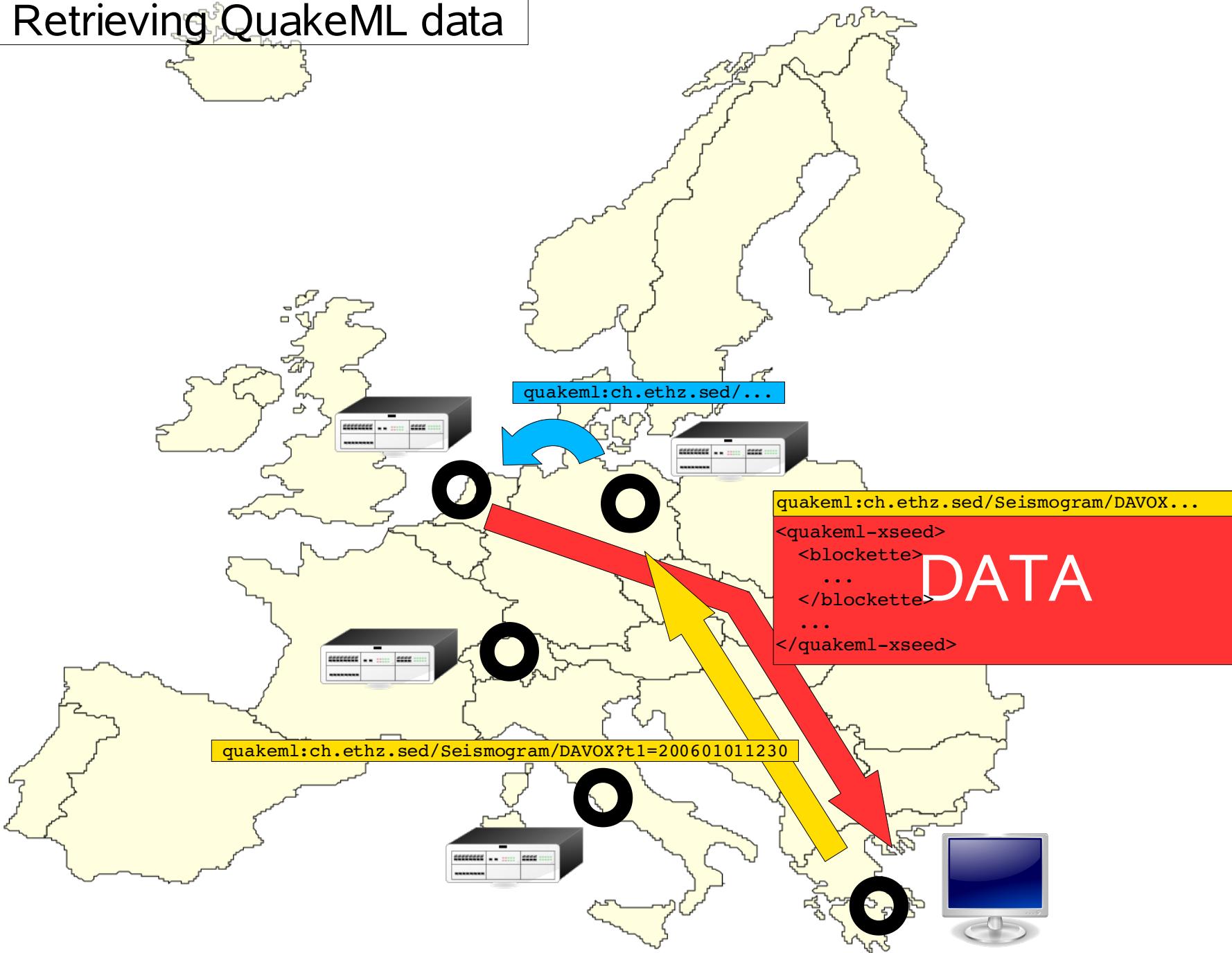
Metadata & Data Exchange

Retrieving metadata



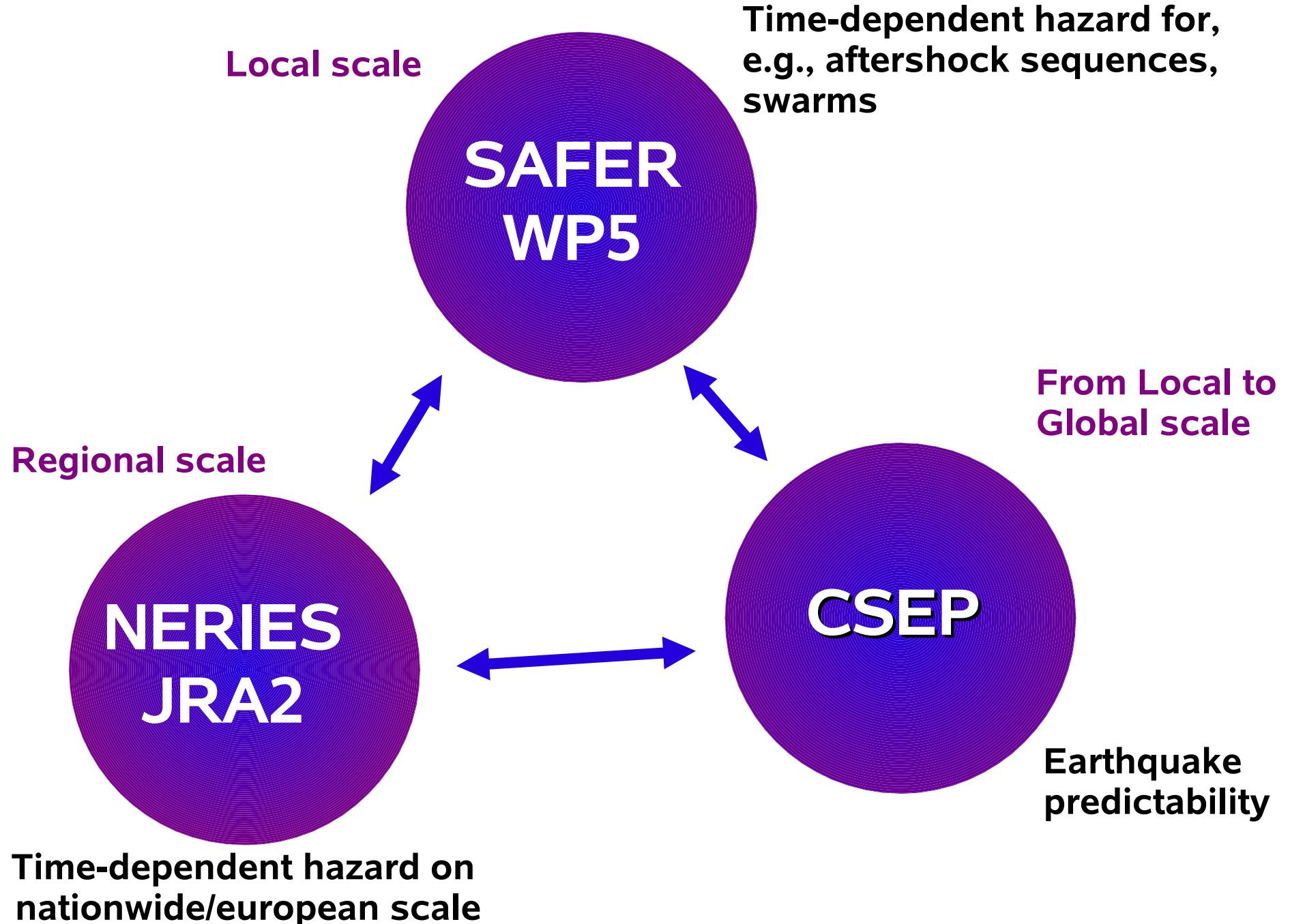
Metadata & Data Exchange

Retrieving QuakeML data



QuakeML – In International Projects

<QuakeML>



QuakeML Development Team

QuakeML core development team (quakeml@sed.ethz.ch)

ETH Zürich

- ★ Fabian Euchner
- ★ Philipp Kästli
- ★ Silvio Maraini
- ★ Stefan Wiemer
- ★ Jochen Wössner

GFZ Potsdam

- ★ Jan Becker
- ★ Winfried Hanka
- ★ Andres Heinloo
- ★ Joachim Saul
- ★ Bernd Weber

USC, Los Angeles

- ★ Danijel Schorlemmer

USGS/NEIC, Denver

- ★ Ray Buland

IRIS DMC, Seattle

- ★ Linus Kamb

Contact Us

- Check our website

<http://www.quakeml.ethz.ch>

<http://www.quakeml.org> (upcoming)

- Write us

quakeml@sed.ethz.ch

- for participating in the request for comments process:
Send us your email-address and you will receive the
RFC documents
- for questions
- for suggestions