QuakeML
ZMAP++
CSEP

Danijel Schorlemmer (USC, Los Angeles)
the QuakeML working group
(ETH, GFZ, USC, USGS, IRIS)
the ZMAP++ working group
the CSEP working group
Challenges

Challenges in software development in statistical seismology

- Large variety of data (earthquake catalogs have different contents)
- Partly impractical formats (not designed for machine processing)
- Difficult to identify future usage of tools
- Different environments:
  - real-time & automated processing
  - research tools
UNIX Philosophy

“This is the Unix philosophy:

1. Write programs that do one thing and do it well.

2. Write programs to work together.

3. Write programs to handle text streams, because that is a universal interface.”

Doug McIlroy
UNIX Philosophy

“This is the Unix philosophy:

1. Write programs that do one thing and do it well.
2. Write programs to work together.
3. Write programs to handle text streams, because that is a universal interface.”

Doug McIlroy

Do not try to design the “one fits all” solution
Consequences of the UNIX Philosophy

- Modular toolkits to support reassembling of programs or processing chains for specific tasks
- Interoperability on the level of data not codes
- Interface through open standards (recommendations)
- Open-source to warrant:
  - Maximum level of platform independence
  - No distribution problems
  - Independence of software vendors (no lock-in)
The Projects

**QuakeML**
XML-based format family for seismological data
(Participants from ETH, GFZ, SCEC, USGS, IRIS)

**ZMAP++**
Toolkit for statistical analyses of seismicity

**CSEP**
Composite Testing Center software

“This is the Unix philosophy:

1. Write programs that do one thing and do it well.
2. Write programs to work together.
3. Write programs to handle text streams, because that is a universal interface.”
XML

Why XML (eXtensible Markup Language)?

Open and free standards
W3C: XML, XPath, XSLT, XLink, XPointer, XML Schema, ...

Open source & multi-platform tools
Xerces, Xalan (Apache), libxml2, libxslt (Gnome), ...

Platform independent
ASCII-files with defined encodings

Flexible & extensible
XML-Namespace
QuakeML

Earthquake catalog
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 UML class diagram
Version 0.61
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

- unambiguous, persistent identifiers of networked resources
- rich standardized metadata description (Dublin Core)
- resource discovery by standardized query gateways

Example: SED Location Program

```
smi:ch.ethz.sed/Software/NLLoc/4.10
```

Analogy

```
doi:10.1051/0004-6361:20064840
http://dx.doi.org/10.1051/0004-6361:20064840
```
A registry is a resource that stores and provides metadata about other resources

- provides a Web Service which resolves identifiers

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

- if 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>]]`
Publishing of data & metadata
Retrieving metadata
Metadata & Data Exchange

Retrieving QuakeML data

```
<quakeml:xseed>
  <blockette>
    ...
  </blockette>
  ...
</quakeml:xseed>
```

```
quakeml:ch.ethz.sed/Seismogram/DAVOS?
t1=200601011230
```

```
quakeml:ch.ethz.sed/Seismogram/DAVOS...
```
QuakeML – From Data Model to XML and API

- 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

Developed at GFZ
Modularity

- XML Parser
- C++ Classes
- C++ Code
XML Piping

Commandline program

XML Parser

C++ Classes

C++ Code

Pipe in

Pipe out
Scripting Abilities

Python

Seismic network

Data import

Catalog

Data preparation

Catalog

QuakeML

QuakeML
Scripting Abilities

Python

Seismic network → Data import → Catalog → Data preparation → Catalog

Google Earth

XSLT

KML → SVG

Map
GUI program
Modules

- Data import (e.g. ANSS catalog)
- Data preparation:
  - Cutting in space
  - Cutting in time
  - Cutting in magnitude
- Declustering
- Dequarrying
- Completeness estimates:
  - Maximum Curvature
  - EMR
  - Probabilistic magnitude of completeness
- Computation of seismicity parameters:
  - a,b-values
  - Omori-sequence parameters
- Mapping of seismicity parameters
- ...

QuakeML
CSEP

Codegenerating toolchain for:

- QuakeML (with extensions)
- Forecast format
- Result formats
CSEP – Natural Laboratories

Data Preparation
- Seismic network
- Data import
- Catalog
- Data preparation
- Catalog

Forecast Generation
- Gear
- Forecast

Test
- QuakeML
- Forecast
- Test
- Result
QuakeML – In International Projects

SeisComP 3
Earthquake processing program and tsunami warning system for the Indian Ocean

ZMAP++
Statistical seismicity analysis tool

EMSC/ORFEUS
Data center

CSEP – SCEC
Earthquake prediction evaluation and testing center

SAFER WP5
Time-dependent hazard for aftershock sequences & swarms

NERIES JRA2
Time-dependent hazard on nationwide/European scale

CODE

DATA
Summary

- Open research toolkits
- Automated toolchains
- Extensible framework

- Open and changeable specifications
- Fully open source (General public license)
- Support C++ & Python
- Codegenerator can be extended to support other languages (e.g., Java)

MatLab users:
- IPython for interactive scripting
- NumPy/SciPy for scientific programming
- MatPlotLib for plotting using the MatLab-syntax
Contact the QuakeML Group

- Visit our website
  
  http://www.quakeml.ethz.ch

- 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