

# Fuego Core

## Middleware for Mobile Data Communications and Synchronization

Sasu Tarkoma, Ramya Sri Kalyanaraman, Jaakko Kangasharju, Tancred Lindholm, Kimmo Raatikainen

### Project Objectives 2002-2006

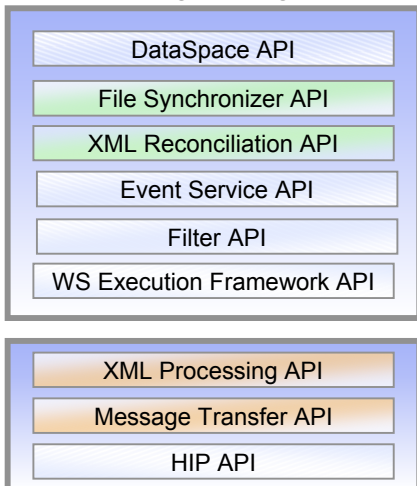
- To specify and implement efficient data communications and synchronization services for mobile computing
- Work areas: efficient XML interchange, event-based computing, XML-aware data synchronization
- Funded by Tekes and industry partners
- Open source software available at [hoslab.cs.helsinki.fi](http://hoslab.cs.helsinki.fi)
- Homepage: [www.hiit.fi/fuego/fc](http://www.hiit.fi/fuego/fc)

**Contact:** [kimmo.raatikainen@hiit.fi](mailto:kimmo.raatikainen@hiit.fi) (Project Leader), [sasu.tarkoma@hiit.fi](mailto:sasu.tarkoma@hiit.fi) (Project Manager)

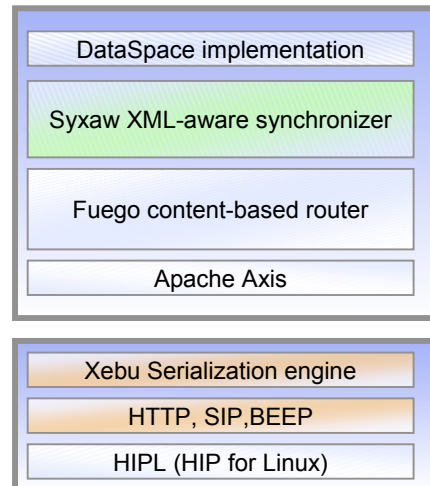
### XML-aware File Synchronization

- Idea: set up shared data spaces by *linking* directory trees on different devices to each other
- Data always available and writable. Optimistic share model
- Framework for reconciling concurrently modified XML data by three-way merging
- Designed for limited bandwidth and high latency
- Simple HTTP GET/POST synchronization protocol
- Very few network roundtrips → GPRS-friendly
- Efficient XML-based model for directories
- Implementation integrates with existing file systems in a nondisruptive manner

### Application Programming Interfaces



### Implemented Services



### Event System

- Scalable distributed event framework for mobile computing
- The Fuego event router consists of two parts: access server functionality with buffering and handover support for mobile clients, and extensible routing core for distributed operation
- New data structures for efficient content-based routing: poset-derived forest and the weakly merging forest
- Rendezvous-based mobility support for fast handovers and subscription topology updates
- Dataspace for filter-based collection tracking

### XML Messaging

- Improving XML-based messaging for mobile devices
- Message service: Asynchronous one-way and two-way messaging with a variety of message exchange patterns
- XML Processing API: Streaming processing of messages with included support for typed data
- Message format: Compact and simple binary representation compatible with XML
- Protocol: Two-way messaging with small overhead and awareness of mobility