
Web Services In Emergency Management

A View on Web Service Infrastructures in EM

Dr. Ulrich Raape

Fraunhofer Institute for Factory Operation and Automation (IFF)
Magdeburg, Germany

©Prof. Dr.-Ing. habil. Michael Schenk
Fraunhofer IFF
Magdeburg, 2004

 **Fraunhofer**
Institut
Fabrikbetrieb
und -automatisierung


The International
Emergency
Management
Society

Content

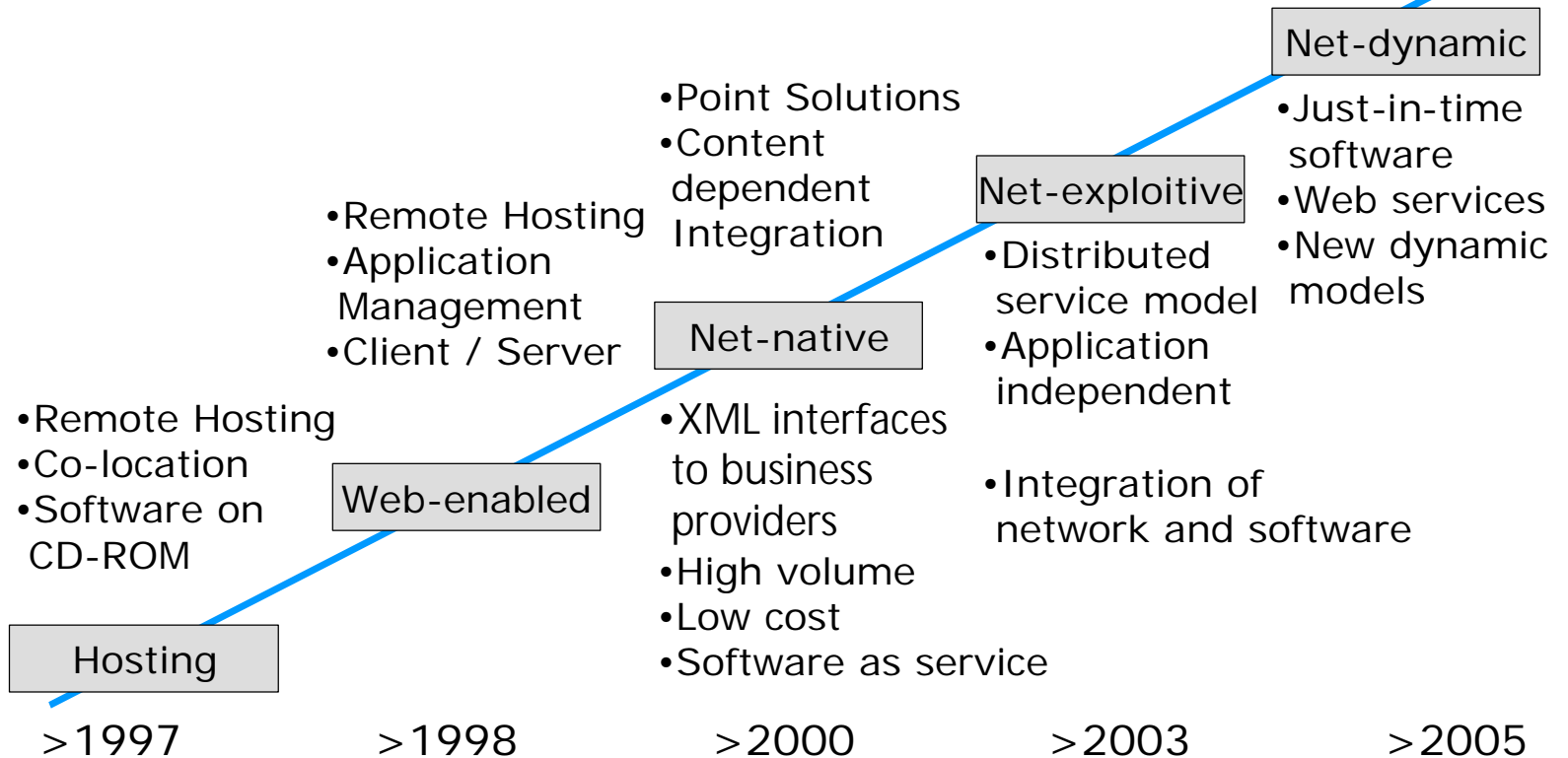
- Motivation
- Information: EM Requirements
- IT Architectures for EM
- Web Service Architecture for EM
- Web Services
- TIEMS Web Service Testbed
- Outlook

Motivation

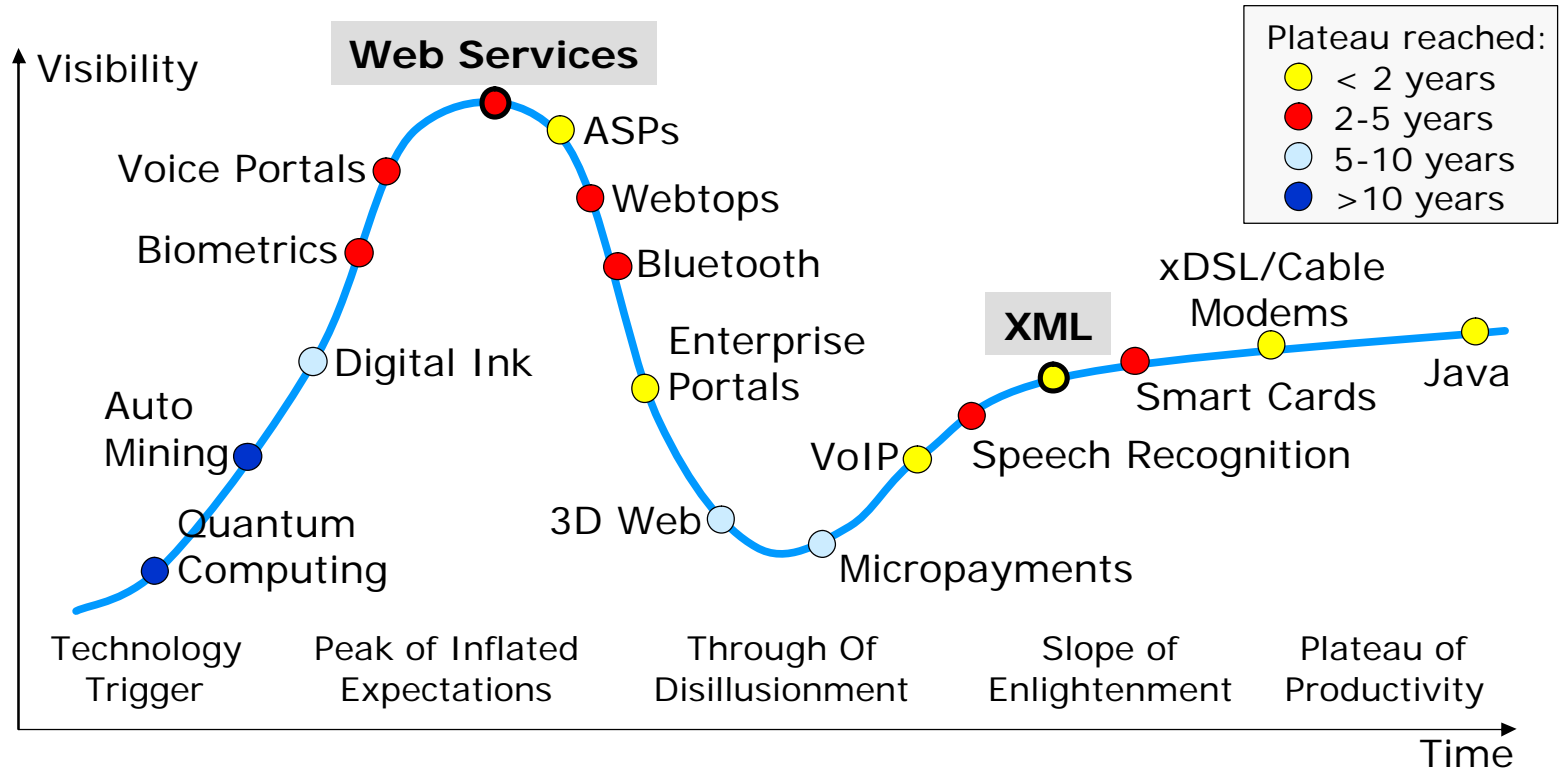
Why Web Services in Emergency Management?

- Web Services are an increasingly successful method of sharing business (domain) logic, data and processes
 - ↳ Web Services allow applications to interface, internals are hidden
 - ↳ Web Service Infrastructures have unique features and drawbacks
 - ↳ Same applies to „Services“ (without „Web“, e.g. CORBA-based)
- Emergency Management needs flexible IT solutions
 - ↳ Access to distributed data, ad-hoc configuration, scenario support, ...
- Could EM Applications benefit from the Web Service Approach (also called „Service Oriented Architecture“ SOA)?

Evolution of Application Service Providing (ASP)



Technologies under Development

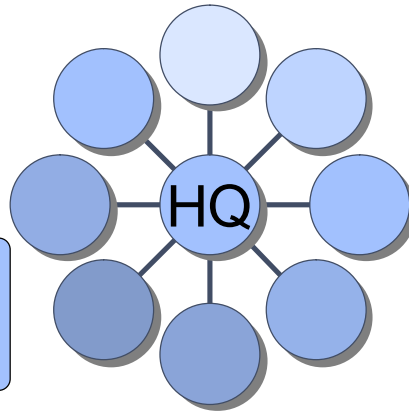
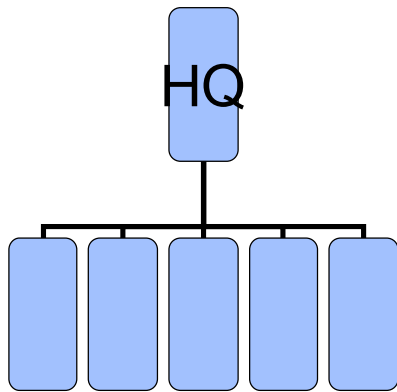


Information: EM Requirements

- Information should be
 - ↳ Accessible: fast, reliable access
 - ↳ Up-to-date: not outdated
 - ↳ Accurate: as precise as possible
- Information Systems should be
 - ↳ Flexible: Applicable to ad-hoc situations
 - ↳ High Resilience/ Fault Tolerance/ Availability/ Survivability
 - ↳ Applicable to more than 1, if not all 4 Phases of EM
- How to comply with as much requirements as possible?
 - ↳ Difficult to comply with one requirement without neglecting the others
 - ↳ New ideas and approaches necessary to „fill the gaps“

IT Architectures for EM

Different Views / Philosophies on how Systems should look like to support Emergency Management tasks



Headquarter Style, centric,
Command&Control Philosophy

Mixed
Styles

Infrastructure Style, distributed,
Cooperation (ERC) Philosophy

How could a Web Service Infrastructure for Emergency Management look like?

- The Infrastructure consists of a (numerous) set of Web Services with well defined content and structure
 - ↪ Services can provide static information or dynamic content (e.g. realtime measurements, simulations)
 - ↪ Services are registered in a Catalog / Registry
 - ↪ Service Trading: find a service by asking a Registry, bind the service by connecting to it, communicate with it (publish/find/bind/execute concept)
 - ↪ Services can be „connected“ to form Service Chains
- Example
 - ↪ Local evacuation in Seoul (toxic release)
 - ↪ (will be described during presentation)

Services to be provided within the Infrastructure

- Non-spatial
 - ↳ General non-spatial information (all types of databases)
 - ↳ Domain-specific non-spatial information
- Spatial
 - ↳ Basic geographic information
 - ↳ Topology, Terrain, Land use, Borders, Population, ...
 - ↳ Infrastructures (Rivers, Streets, Supplies, ...)
- Dynamic
 - ↳ Weather (current and forecast)
 - ↳ Simulations (modular and interoperable)



Web Services and related Standards

- Web Service
 - ↳ Standardized way of integrating web-based applications over the Internet
 - ↳ Web Services are not tied to any one operating system or programming language.
- Related Standards
 - ↳ Extensible Markup Language (XML): used to tag the data
 - ↳ Simple Object Access Protocol (SOAP): used to transfer the data
 - ↳ Web Services Description Language (WSDL): used to describe the services
 - ↳ Universal Description, Discovery and Integration (UDDI): used for listing available services
 - ↳ Electronic Business XML (ebXML)

Spatial Web Services: Overview

- OpenGIS Consortium Specifications
 - ↪ WMS: Web Map Service
 - ↪ WFS: Web Feature Service
 - ↪ WCS: Web Coverage Service
 - ↪ WCAS: Web Catalog Service
 - ↪ WFS-G: Gazetteer Service
 - ↪ WTS: Web Terrain Service
 - ↪ WCTS: Web Coordinate Transformation Service
 - ↪ SWE: Sensor Web Enablement
 - Sensor Planning Service (SPS)
 - Web Notification Service (WNS)
 - Sensor Collection Service (SCS)

Dynamic Web Services

- Dynamic Web Services

- ↪ The „missing link“ that complement static web services
- ↪ Approach to integrate Distributed Simulation Architectures to allow distributed „Simulation Service“ to interact and co-operate
- ↪ „Distributed spAtio-temporaL Interoperability“ („DALI“) Approach to integrate OGC specifications and the HLA standard for distributed simulation
- ↪ Concepts and Technologies currently developed in R&D projects

- Synergies

- ↪ Services could be combined for 4-dimensional scenarios
- ↪ Scenario management

TIEMS Web Service Testbed

deegree

- Testbed in preparation
 - ↪ Spatial, Non-Spatial and Dynamic Web Services
 - ↪ Integration and Administration Services
- Aims and Key Facts
 - ↪ Showcase of application potential, Proof-of-concept of new services
 - ↪ Demonstration of Spatial/Dynamic Web Service Integration
 - ↪ Example for low-cost approach
 - ↪ Open Source Solution (e.g. deegree Framework)
 - ↪ Public Domain Data (e.g. U.S. CIA World Fact Book, CRED EM-DAT Disaster Database, public domain spatial data)
 - ↪ R&D support from Universities and Student Groups

Related R&D Projects

- Project MEDSI

- ↳ „Management Decision Support for Critical Infrastructures“
- ↳ Funded by the European Commission
 - 6th Framework Programme
 - Start: 01/01/2004, Duration 18 months
 - 11 Partners from 8 Countries



- Large-scale Testbed for Infrastructure Approach

- ↳ Prototypes to test and evaluate infrastructure-based applications for Emergency Management
- ↳ Results / Lessons Learned will identify strengths and weaknesses and will help to build next generation Emergency Management Applications



Outlook

- Web Services can provide valuable input for Emergency Management Applications
- (Web) Service Infrastructures have the potential to be the basis for powerful EM applications
 - ↳ Added values can be identified (e.g. 4D scenarios)
 - ↳ Have strengths and weaknesses on their own
- How do they perform? Testbeds/Projects are underway
 - ↳ Strengths & weaknesses complementary to other architectures?
 - ↳ Can strengths be cumulated? Can weaknesses cancel out each other?
- Open Process, Contributions welcome!