Peer-to-Peer Architectures, Grid Infrastructures, and Service-oriented Architectures for Digital Libraries

Heiko Schuldt
Information & Software Engineering
UMIT, Hall in Tyrol, Austria

Joint work together with
• Donatella Castelli (CNR, Pisa)
• Gerhard Weikum (MPI, Saarbrücken)
• and other DELOS WP 1 partners
Digital Libraries – The Past

- Isolated and/or monolithic systems
- Proprietary interfaces
- Limited to access to content of one single provider
- …

Diagram:
- DL Management System
- DL Content
Digital Libraries – The Future

- Self-contained services rather than monolithic systems / DLs
- Flexible access to services and content across DLs / content providers

- DL Services
- DL Content
Outline

• Requirements of Next-Generation Digital Libraries (or Dynamic Ubiquitous Knowledge Environments)
• Service-oriented Architectures (SoA)
• Peer-to-Peer Infrastructures (P2P)
• Grid Architectures
• P2P, Grid, and SoA for DL: Towards putting everything together
Example 1: The Virtual Campus of a University

Rope Bridges
Eine Besonderheit besteht darin, dass sich die mit speziell geformten, variabel geneigten Kragarmen ausgestatteten Querträger direkt auf den Tragseilen absetzen. Am auffälligsten ist wohl der geringe Stich der Tragkabel, der mit 2,30 m in der 144 m langen Mittelspannweite das traditionelle Stich-Spannweiten-...
Example 2: Health Monitoring and eHealth Digital Libraries …

Blood pressure

ECG

oxygen saturation

activity

context

Pre-processing

ECG variability

Pre-processing

Has person fallen?

oven, light off

historical data

„e-Inclusion“

Electronic patient record

Call emergency physicians/neighbors

Analysis & prognosis

correlation and processing
Example 2: Health Monitoring and eHealth Digital Libraries

- Information is physically distributed
- Virtual integration
- Preservation of privacy

Electronic Patient Record

Family doctor

Hospital

Former family doctor

Former family doctor

Laboratory
Example 2: Health Monitoring and eHealth Digital Libraries

- Similarity search in virtual digital health records
  - e.g., multi-object multi-feature query
  - Query by example
  - Relevance feedback
Requirements ...

• Specialized services, local to a content provider
  – Search
    • Different media types
    • Content-based
    • Multi-object, multi-feature
    • Relevance feedback
    • ...
  – Indexing
  – Annotation
  – Metadata management
  – Content management
  – Resource Management
  – ...
... Requirements ...

• Virtual DL (virtual collection)

• Management of services which are
  – Distributed
  – Heterogeneous
  – Autonomous
… Requirements …

- Specialized services, across different providers
  - Search
    - Different media types
    - Content-based
    - Multi-object, multi-feature
    - Relevance feedback
    - …
  - Metadata management
  - Content management
  - Resource Management
  - Indexing
  - without central control
  - (no censorship)
... Requirements ...

- Management of computationally intensive services
  - Scale-out
  - Load balancing

- Composition of services
  - Defining complex services / processes / workflows on the basis of existing services
  - Flexibility: automatic adaptation
… Requirements …

- Notification of changes
- Guaranteed consistency of derived data
- Personalization
- Visualization
- Access anywhere
  - Mobile devices
- Context- and location-aware services
- Authentication & authorization
... Requirements ...

- High degree of availability: Access anytime
  - Replication

- High degree of dependability / reliability
  - Systems their users can count on

- High degree of scalability
... Requirements

- Continuously generated data
  - Sensor networks
  - Sensor data streams (hardware / software sensors)
- Monitoring of users 24/24
- Preservation of privacy
Service-oriented Architectures

- Self-contained functional units of applications encapsulated as services
- Interaction between services by well-defined interfaces
- Implementation details are hidden behind the service interface
- Examples for service-oriented architectures:
  - DCOM
  - CORBA
  - Message-oriented middleware (e.g., MQSeries, ...)
  - Web services
Web Services

- Services to be invoked in a uniform way, independent of their implementation details (programming language, platform, etc.)
- Use common protocols (HTTP) for transport and XML for data representation
  - **SOAP** (Simple Object Access Protocol) for transport
  - **WSDL** (Web Services Description Language) for service description & discovery
  - **UDDI** (Universal Description, Discovery, and Integration) as service registry
  - **BPEL4WS** (Business Process Execution Language for Web Services) for the definition of composite services
  - And many other (more or less accepted) standards on the web service stack
Web Services: Overview

HTTP

- accessMetaData
- accessDLObjects
- GetListOfStocks
- GetLastTradePrice
- Order

www.myDL.com
DL Services

www.stockquoteserver.com
Stock Data

www.pizza.at
Pizza Service
SOAP: Architecture

- SOAP spec: [www.w3.org/TR/soap/](http://www.w3.org/TR/soap/)
- Unidirectional information exchange
- No common distributed object management infrastructure needed
- Encapsulation of remote calls
WSDL: Architecture

- WSDL spec: www.w3.org/TR/wsdl/
- Implementation-independent description of
  - methods of web services and
  - their interfaces

Query: Services of the StockQuoteServer

WSDL Document (XML Document with Method Specification)

SOAP Request Document

Client

WSDL Server

SOAP Server

HTTP, ...

GetLastTradePrice

www.stockquoteserver.com

Stock Data
UDDI: Architecture & Usage

Client

Query: stock
Services

HTTP

... 

WSDL
Server

SOAP
Server

UDDI-Repository

exchange

Register Service

UDDI-Repository

Stock Data

www.stockquoteserver.com

www.nyse.com

www.uddi.org

www.stockquoteserver.com

www.nyse.com

www.stockquoteserver.com

Query: stock
Services

www.stockquoteserver.com

www.nyse.com
BPEL4WS

- **BPEL4WS**: Business Process Execution Language for Web Services
- Specification of business process behavior based on web services
  - **Executable business processes model**: actual behavior of a participant in a business interaction
  - **Business protocols (abstract processes)**: specify the mutually visible message exchange behavior of each of the parties involved in the protocol, without revealing their internal behavior
The “Rest” of the Web Service Stack

- WS-Discovery
- WS-Interoperability
- WS-Inspection Language
- WS-Manageability
- WS-Addressing
- WS-MessageDelivery
- WS-Routing
- WS-Reliability
- WS-Eventing
- WS-CompositeApplicationFramework
- WS-Notification (WS-BaseNotification / WS-BrokeredNotification)
- WS-Topics
- WS-AtomicTransaction
- WS-Coordination
- WS-BusinessActivity
- WS-TransactionManagement
- WS-Security
- WS-Trust
- WS-Federation
- WS-Choreography
- WS-ConversationLanguage
- WS-Context
- …
SoA and Future Digital Libraries

- SoA is the core backbone of DUKEs
- Definition, invocation, description of services
- Service directories
- Basic service composition
Peer-to-Peer Architectures …

Main goals and features:

- Decentralization
- Sharing of distributed resources
- Autonomy
- Self-organization and autonomic behavior

Resource sharing

• Example: Napster, featuring a central index
Peer-to-Peer Architectures

- Example: gnutella
  - Completely distributed
  - Uses flooding algorithm
Peer-to-Peer Architectures

- 1\textsuperscript{st} generation: focus on MP3 sharing. But also:
- Computer resources
  - Processors
  - Memory
  - Disks, …
- Intellectual resources like
  - User annotations
  - Recommendations
  - …
- Other applications
  - Collaborative authoring
  - Groupware
  - Publish-subscribe applications, …
Next generation P2P systems:

• Distribute index over a larger number of (super-)peers

• Reduce number of messages by
  – By appropriate routing protocols
  – Efficient localization of data objects

• Apply strategies for
  – Load balancing
  – Failure resilience
  – Replication
  – Self-organization

• Other aspects
  – Trust
  – Privacy
  – Anonymity
Examples for P2P Indexing

- Scalable & self-organizing distributed Hash Tables (DHTs), e.g.
  - Pastry
  - Tapestry
  - CAN
  - CHORD
  - ...

- m-bit identifier assigned to each peer
- Peers organized on a ring (mod $2^m$)
- Key $k$ assigned to the first node whose identifier is $\geq k$
- Reassignment (peer joins or leaves) just considers successor
- Each peer maintains a partial local routing table (finger table)
P2P and Future Digital Libraries

- Complete decentralization
  - No global control
  - No censorship
- Data and service management
- Indexing
- Self-organization …
  - … in the presence of many failures
  - … to address the high dynamics of large-scale DLs
Grid Infrastructure …

- Hardware and software infrastructure for coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organization
- Generations of grid infrastructures
  - Computational grid
  - Data grid
  - Service grid
... Grid Infrastructure

- Key concepts: sharing of
  - Computing and storage devices (heterogeneous)
  - Data
  - Software and Services
  - More general: networked resource usable in a remote way

- Goals:
  - Provide a high degree of scalability, even if components are geographically distributed (latency, etc.)
  - Provide a high degree of adaptability (failure handling, account for dynamics)
  - Support load balancing between components
Service Grid

• Open Grid Service Architecture (OGSA)
  – Encapsulation of resources as services
  – Providing grid technology to dynamically create, manage, discover, etc. these services

• WS-Resource framework (WSRF)
  – Bringing web service technology to the grid
  – Creation, addressing, inspection, and lifetime management of stateful resources (WS-Resources)
Grid Infrastructures in Practice

- Globus Toolkit (GT): basic grid environment
  - GT4 to support OGSA and WSRF
- Condor (resource management)
- gLite
Grid and Future Digital Libraries

- Efficient resource management
- Load balancing and scheduling
- Self-adaptability
  - Automatic service installation and deployment
- Availability
- Replication
- Authentication & authorization
SoA, P2P & Grid: How does it fit together

- These are by far no orthogonal technologies
- Differences more and more diminish
  - e.g., grid services
- There is not a unique recipe on …
  - how to construct a DL/DUKE
  - what base technology to use
- Strongly dependent on particular requirements
SoA, P2P & Grid: All DL-related Problems Solved?

- Several of the challenges can be addressed by SoA, P2P and/or the Grid
- But there is still some work left, e.g.,
  - Context- and location-aware services
  - Support for dependable and reliable systems
  - Continuous sensor data, monitoring of users
  - Guaranteed consistency of derived data
  - Support for mobile devices (switching between connected / disconnected mode), advanced replication and synchronization mechanisms
  - Complex queries across media types
  - …
Summary & Outlook

• SoA, P2P, Grid are well suited as base technology for DLs / DUKEs
• Trend to integrate and merge functionality will increase

• Work in DELOS will be continued by defining
  A Reference Model for Digital Library Management Systems