Distributed Information Management: From Coordination to Market-based Cooperation

Karl Aberer
EPFL-DSC

DSC Open Day, July 5, 2000
Outline of Talk

• Introduction
  – information systems for market-based cooperation

• New functional requirements (example)
  – optimization of service quality for flexible workflows

• Implementation
  – light-weight infrastructures based on XML architecture

• Terminode Scenario
  – self-organizing data integration application

• Conclusions
Outline of Talk

• Introduction
  – information systems for market-based cooperation
• New functional requirements (example)
  – optimization of service quality for flexible workflows
• Implementation
  – light-weight infrastructures based on XML architecture
• Terminode Scenario
  – self-organizing data integration application
• Conclusions
Markets vs. Hierarchical Forms of Organization

- Communication networks, standardized information formats and security infrastructure
  - facilitate the coordination of activities
  - thus decrease coordination cost
  - thus are in favor of market forms of organization
Scenario: Insurance Workflow

Central workflow management system
- initiates and controls tasks
- provides access to data and resources
- assigns tasks to organizational units

receive call
assign garage
notify insurance
contact garage
estimate cost
assign adjustor
agree repair
check invoice
inspect car
repair car
invoice

estimate < 500

notify adjustor
fill claim form
amend estimate
reconcile info
finalize claim
Today: Virtual Organization

Crossorganizational workflow management
- outsourcing of services
- cooperating WFMS systems
- alternative providers for same service

Call Centre Workflow Management System

Receive call → Assign garage → Notify insurance

Garage Workflow Management System

Receive car → Estimate cost

Adjustor Workflow Management System

Inspect car → Agree repair → Check invoice

Insurance Workflow Management System

Notify adjustor → Fill claim form → Amend estimate → Reconcile info → Finalize claim
Problem: Information System Cooperation

Service provider(s)

Service consumer(s)

contract

inspect car

estimate < 500

notify adjustor

contact garage

assign adjustor

agree repair

check invoice

invoke

monitor

control

result

agree repair

check invoice

notify adjustor

fill claim form

amend estimate

reconcile info

finalize claim

adjustor WFMS

insurance WFMS
Basic System Architecture for Market Based Cooperation

- Service provider
- Service consumer
- Market place
- Contract establishment (searching, negotiation)
- Support services (logging, optimization)
- Contract execution process
- Resource systems (WFMS, DBMS, payment, authentication and other service providers)
- Communication management
- Offer request
- Contract models (process, content or service, price and quality)
- Contract representation (internal and external)
Outline of Talk

• Introduction
  – information systems for market-based cooperation
• New functional requirements (example)
  – optimization of service quality for flexible workflows
• Implementation
  – light-weight infrastructures based on XML architecture
• Terminode Scenario
  – self-organizing data integration application
• Conclusions
Outsourcing of Processes Across Organizational Boundaries

- Contract model for service outsourcing
  - specification of workflow process
    - activities
    - control and data flow among activities
  - quality of service parameters
    - execution time, cost
- Outsourcing requires a more declarative, goal-oriented specification of the workflow process
  - consumer allows flexible control flows in the contract
    - non-vital activities
    - preferred activities, with alternatives
    - preferred, but not required, sequential order of activity set
  - producer optimizes global quality of service parameters of outsourced process through the resolution of alternatives
Flexible Workflows

- Example OR-split

Activity A
output parameters
$p_1, ..., p_n$

$\varphi_1(p_1, ..., p_n)$

$\varphi_k(p_1, ..., p_n)$

Activity $A_1$

Activity $A_k$

Select alternative which contributes optimally to global quality of service goals
Optimization problem

- Quality of service goals
  - total execution time less than deadline
    \[ \max_{p \in P} \sum_{A \in p} \text{dur}(\text{res}(A)) \leq d \]
    - P paths from source to sink
    - A flexible activities, res(A) selected alternative
  - maximize accumulated utility
    \[ \sum_{A \in W} \text{util}(\text{res}(A)) \]
    - W activities in workflow
- Optimization problem of service producers
  - find a resolution of the remaining alternatives such that
    the goals are optimally achieved
  - times and utilities of remaining activities are known
Algorithms

• Search problem is NP complete (number of flexible activities)
  – even for well-structured workflows with AND-splits only
  – pruning of search space
    • alternatives may dominate others
      (both better execution time and utility)

• Practical solution
  – limit lookahead in search
  – estimate QoS of the remaining activities
    • current approach: worst case estimation
    • alternative: consider activity transition probabilities
      (continuous Markov chain model obtained from monitoring)

• Implemented as a support service for crossorganizational workflow management
Outline of Talk

• Introduction
  – information systems for market-based cooperation
• New functional requirements (example)
  – optimization of service quality for flexible workflows
• Implementation
  – light-weight infrastructures based on XML architecture
• Terminode Scenario
  – self-organizing data integration application
• Conclusions
Information Commerce

- Implementation Requirements
  - standardized: infrastructure, data format (offers, contracts), ...
  - light weight systems: thin clients, mobile users, ...
  - openness: content delivery, payment, authentication, ...
XML Architecture

- **Standard XML applications**
  - XHTML, SMIL, P3P, MathML

- **API**
  - DOM
  - SAX

- **Layout**
  - XSL
  - CSS

- **Hyperlinks**
  - XLink
  - XPointer

- **Schemas**
  - XSDL
  - XDTL

- **Metadata**
  - RDF, RDFS
  - PICS

- **Queries**
  - XSLT, XPath
  - XQL, XML-QL

- **XML 1.0**
  - DTD
  - Namespaces
  - Unicode
  - URI

**Specific XML Applications**

- **ICE** information and content exchange protocol
- **IOTP** Internet open trading protocol
- **Micropayment** markup language

...
ICE - Information and Content Exchange Protocol

- Standardized mechanism for content syndication
  - automatizes establishment of syndication relationships, data transfer and result analysis
  - supports flexible delivery policies (push/pull, time)
  - uses XML to encode protocol messages and http for transport
  - does not define content, vocabularies, mechanisms for payment, security and authentication → OPELIX specific extensions

request catalog
- step 1: subscriber ➔ syndicator: ice-get-catalog
- step 2: subscriber ➔ syndicator: ice-catalog
- step 3: subscriber ➔ syndicator: ice-offer (select an offer)

negotiate offer/establish contract
- step 4: subscriber ➔ syndicator: ice-offer (counteroffer, e.g. pull info)
- step 5: subscriber ➔ syndicator: ice-offer (accept counteroffer)
- step 6: subscriber ➔ syndicator: ice-subscription
- step 7: subscriber ➔ syndicator: get-package, initial state ICE-INITIAL
- step 8: subscriber ➔ syndicator: ice-package P1

request/deliver

xml code snippet:
```xml
<ice-delivery-policy
  startdate="1998-07-02T12:00:00">
  <ice-delivery-rule
    mode="pull"
    starttime="02:00:00"
    duration="P7200S"
    maxcount="1"/>
  <ice-delivery-rule
    mode="push"
    url="http://xmlbroker.de/ice-in/"/>
</ice-delivery-policy>
```
OPELIX System Kernel
Based on XML Architecture

Light-weight implementation in JAVA

Resource systems: Payment systems, delivery systems

Contract establishment: hardcoded ICE offer protocol

Support service: user access to catalog and log data (XSLT)

Support service: persistent logging of messages (non-repudiability)

Contract execution: ICE offer interpreter working on a DOM representation (message generation and validation)

ICE catalog

ICE offer

HTTP protocol

XML parsing

XML messages

ICE get-catalog

ICE catalog

ICE offer

ICE get-package

ICE packages

ICE offer

subscription

subscription-id

ICE change-subscription

ICE offer

Persistent DOM

ICE offer

counter-offer

Resource systems: Payment systems, delivery systems

Support service: persistent logging of messages (non-repudiability)
OPELIX System (B2C)

XML encoded messages

Customer (Browser)

OPELIX kernel

Wallet

Merchant

Server Interface

Function API

Deliverer

Delivery (WebServer)

Clearing House

Payment Gateway

Java servlets

Implementation: Java applets

data access: DOM

storage: PDOM

layout: XSLT

OPELIX kernel
OPELIX System (B2B)

- DeliveryClient
- Delivery (PushSystem)
- Customer (Browser)
- Customer Service Provider
- WAP Interface
- OPELIX kernel
- Wallet
- Merchant
- Server Interface
- OPELIX kernel
- Function API
- Deliverer
- Delivery (PushSystem)
- Clearing House
- Payment Gateway

Conversion to WML using XSLT
Design Goals for Business Offer
Language for Information Commerce

• flexible business models
  – pay-per-subscription, pay-per-view etc.
  – processes with payment, authentication, delivery
    • e.g. IOTP (XML-based B2C EDI Format)

• relationships among information items
  – repeated selling of the same or overlapping information
  – selling of aggregate information
  – dynamic selection of the information being sold
  – rights, price, delivery, certificate
Outline of Talk

- Introduction
  - information systems for market-based cooperation
- New functional requirements (example)
  - optimization of service quality for flexible workflows
- Implementation
  - light-weight infrastructures based on XML architecture
- Terminode Scenario
  - self-organizing data integration application
- Conclusions
Scenario: Price Comparison

- **Static set of shops**
- **Slow change rate**
- **Data accessed through Internet**
- **Standardized views**

Data warehouse:
- Streetprices.com

- USA>sports
- USA>toys
- Europe

- Golfshop.com
- Chipshot.com
- Buygolf.com
- Bestvaluegolf.com

Data models:
- Relational?
- XML?
- ASCII?
- HTML?

- HTML
- HTML
- HTML
- HTML
Terminode Price Comparison

Shoppers

- Buyers with mobile devices visiting physical shops
  - product data made available through mobile LAN
  - buyers communicate with shop and other buyers

![Diagram showing interest profile and data storage connections](image)
Proposal: Self-Organizing Information Space

Dynamic, location-sensitive data with restricted access
Mobile users with dynamic individual profiles

Local information market
Terminode Data Manager

Light-weight implementation in JAVA

Contract establishment: locate Terminode neighbors and their offering

Support services:
- optimize contracts and storage w.r.t users
- Goals util(data, loc, time, quality)
- Plans loc(time)
- Budget bud(time)

Contract execution: request data, answer requests, forward requests, push data

Resource systems: Database with product data and directory of product data providers

Data exchange/routing

Terminode comm. infrastructure

Data/routing offer

Virtual data access structure

XML messages
Conclusions

• Changing forms of organization
  – market based organization
  – virtual enterprises, new information economy

• Impact on required information systems
  – new functional solutions
  – implementation platforms

• New organization of information systems
  – self-organization through computational markets
    • Terminode: self-organizing data integration application
  – challenging (interdisciplinary) research

• DSC + DI at EPFL excellent platform