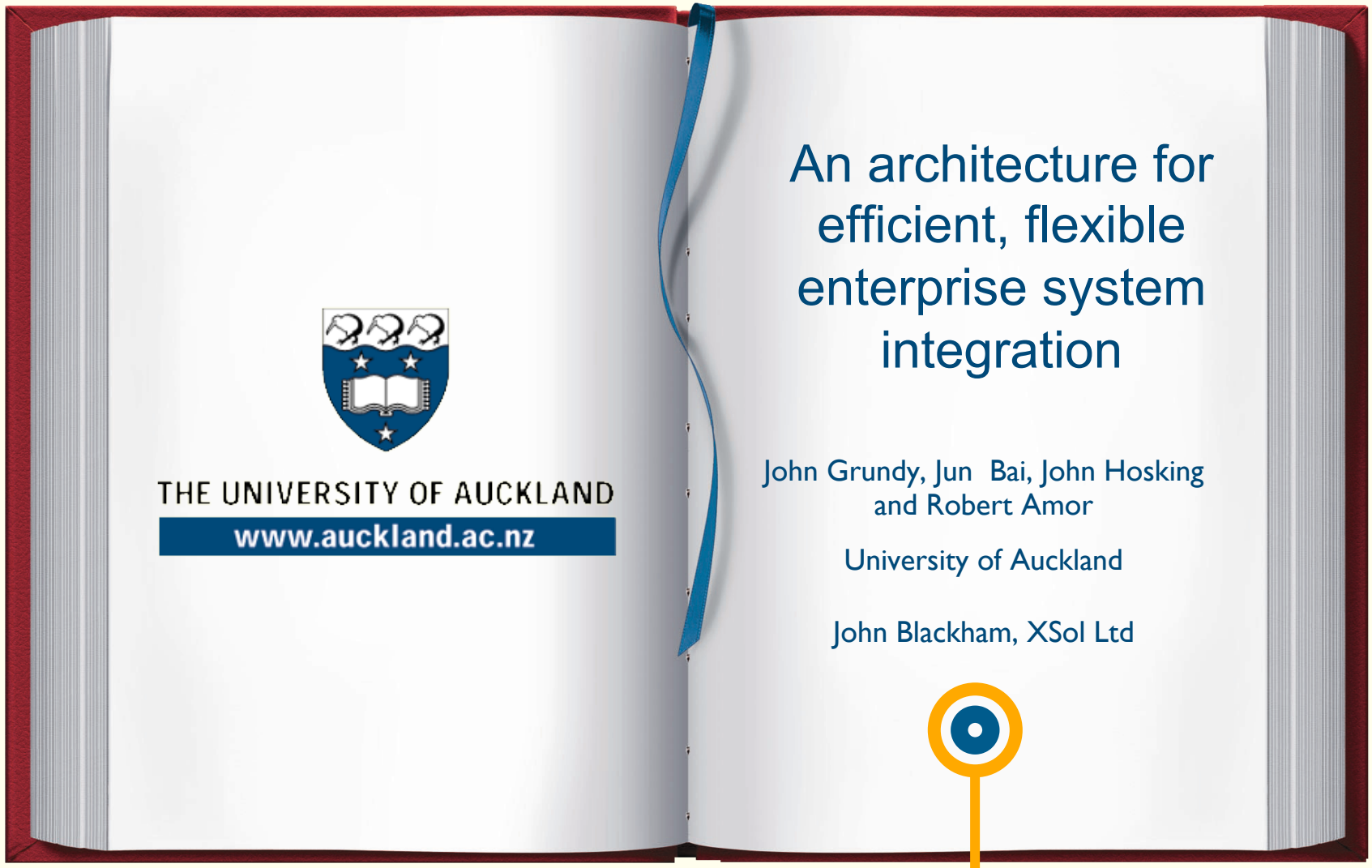


An architecture for efficient, flexible enterprise system integration

John Grundy, Jun Bai, John Hosking and Robert Amor

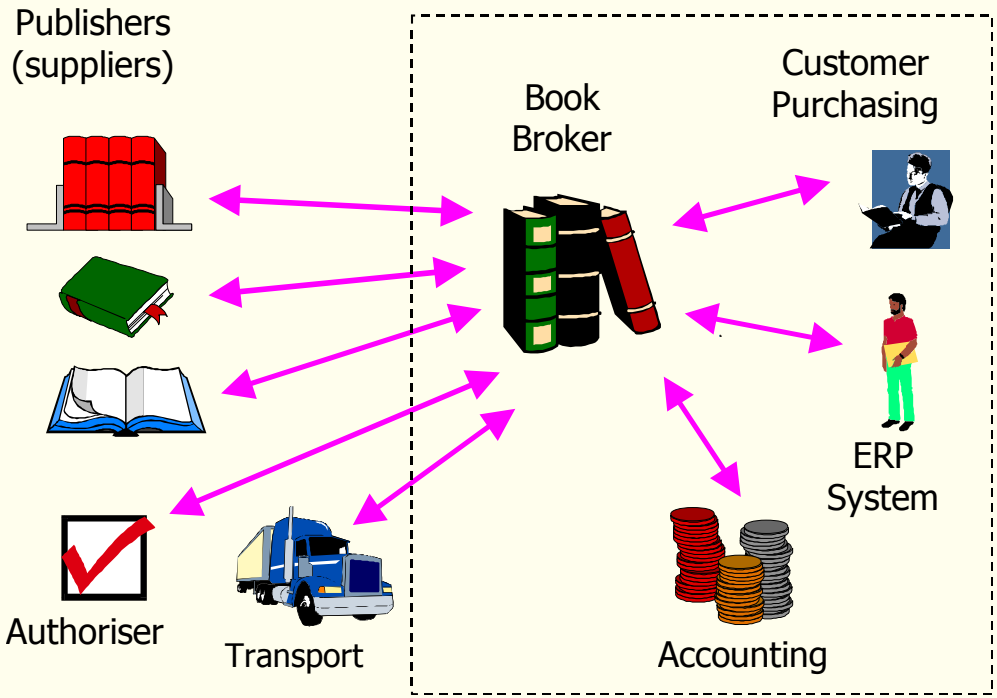
University of Auckland

John Blackham, XSol Ltd

Outline

- Motivation
- Related Work
- Our Approach
- Architecture & Implementation
- Example
- Future work

Motivation – An Example

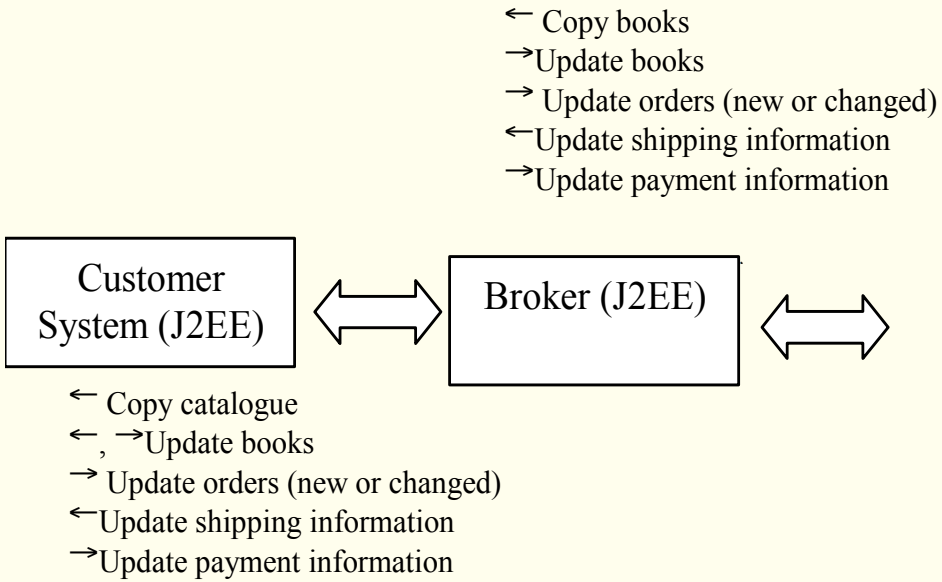


- Book orders brokering
- Range of systems to integrate
- Want to do in seamless, open way
- See BRE project

Related Work

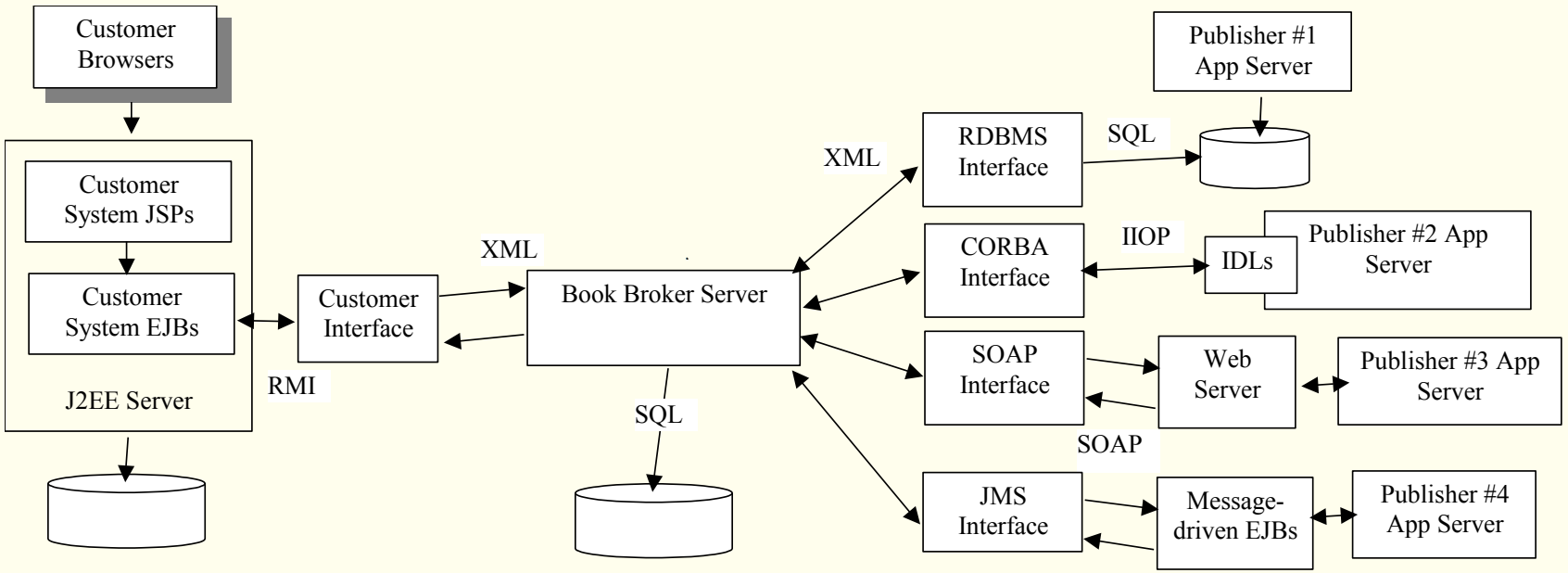
- Bespoke solutions e.g. CONNET – BRE book broker
- CORBA, SOAP, XSLT, Portal infrastructures
- Virtual databases e.g. OpenLink Virtuoso™
- Message integration systems e.g. MQ Integrator™
- B2B integration e.g. BizTalk™, Vitria™, eXcelon™
- Business Process Management (BPM) systems

Our Approach



- Broker data between systems
- Seamless integration
- Optimistic transactions
- Extensible

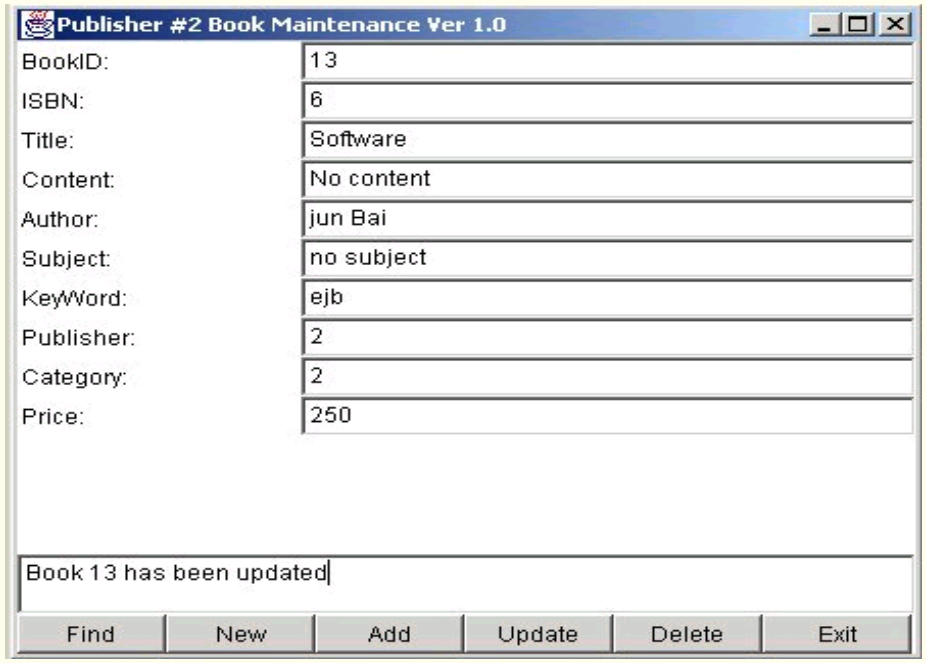
Our Architecture



Implementation

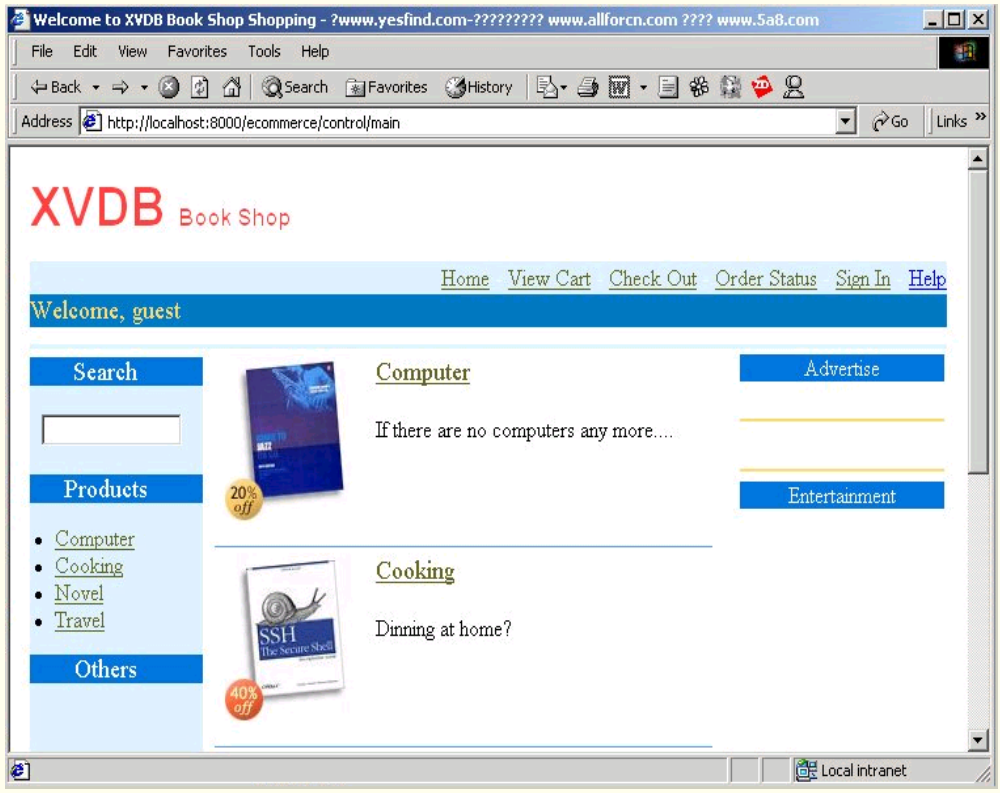
- Integration engine, integration connectors ("agents")
- XML datasets to/from external systems
- Data caching for reliability, asynchronous support
- Technologies:
  - JDBC, XML, XSLT for broker
  - J2EE, EJBs, J2EE Connector architecture for broker/customer system
  - CORBA, RMI, SOAP, Java Messaging System remote system IFs
  - JSPs, Swing for UIs

Examples in Use…



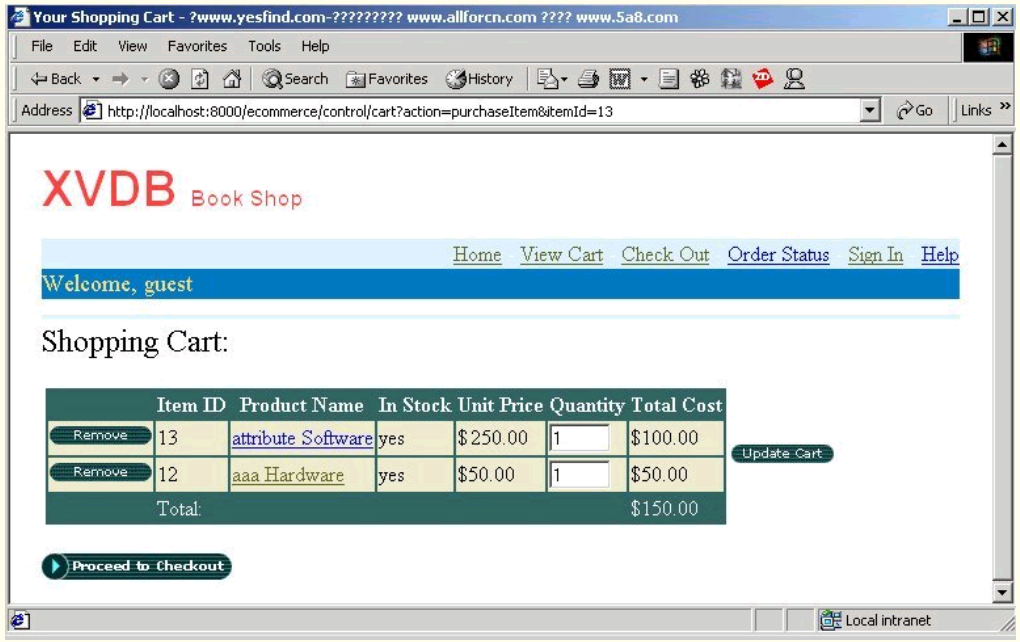
1. Copy book data (+ refreshed…)
2. Customer order
3. Publisher order
4. Order shipping (may change order…)
5. Payment

Examples in Use…



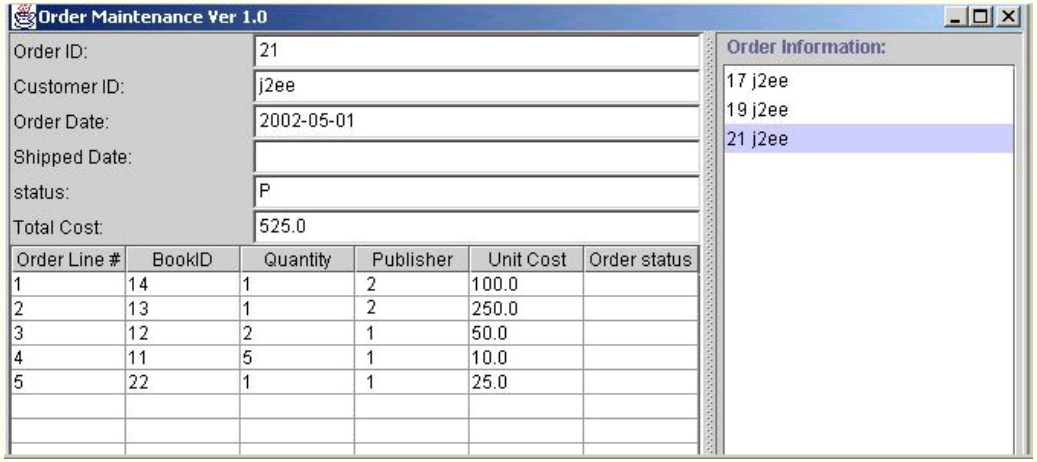
1. Copy book data (+ refreshed…)
2. Customer order
3. Publisher order
4. Order shipping (may change order…)
5. Payment

Examples in Use…



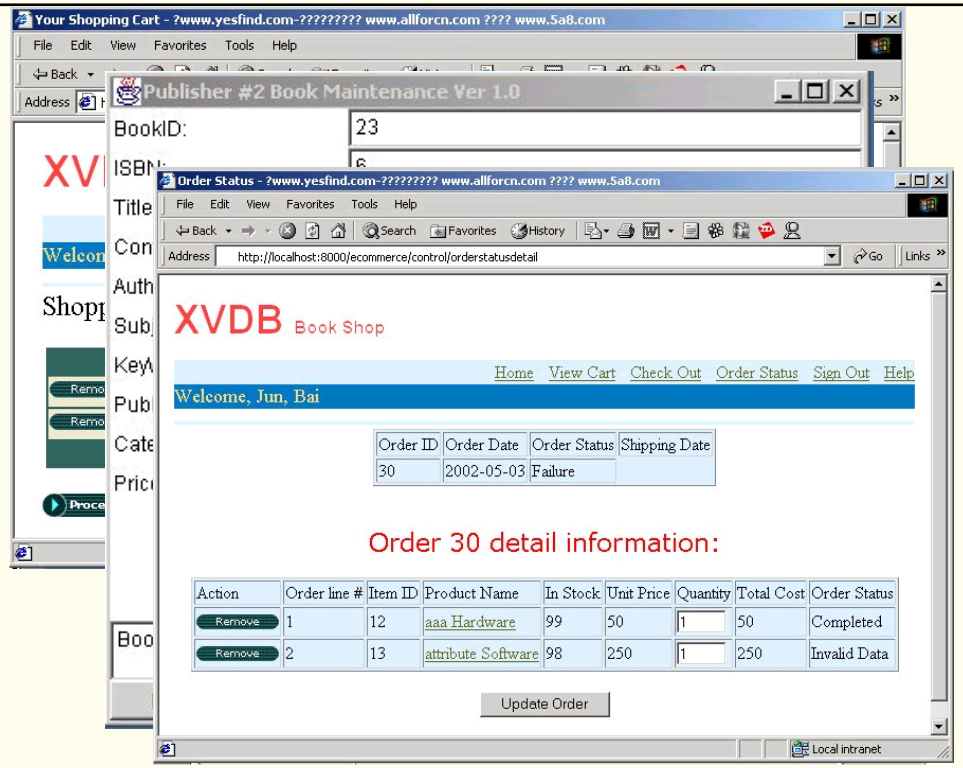
1. Copy book data (+ refreshed…)
2. Customer order
3. Publisher order
4. Order shipping (may change order…)
5. Payment

Examples in Use…



1. Copy book data (+ refreshed…)
2. Customer order
3. Publisher order
4. Order shipping (may change order…)
5. Payment

Exception Handling

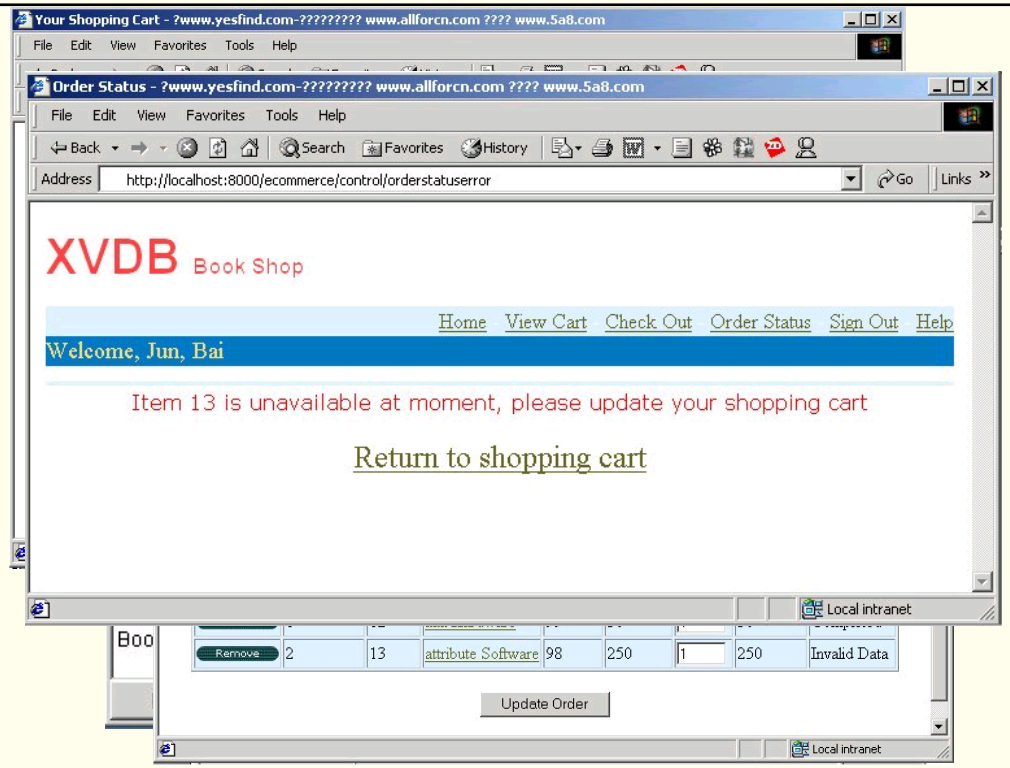


Exception handling:

- Changes to e.g. book data
- Locked data
- Transaction failure e.g. order rejected

Inform user & support LRT across systems

Exception Handling

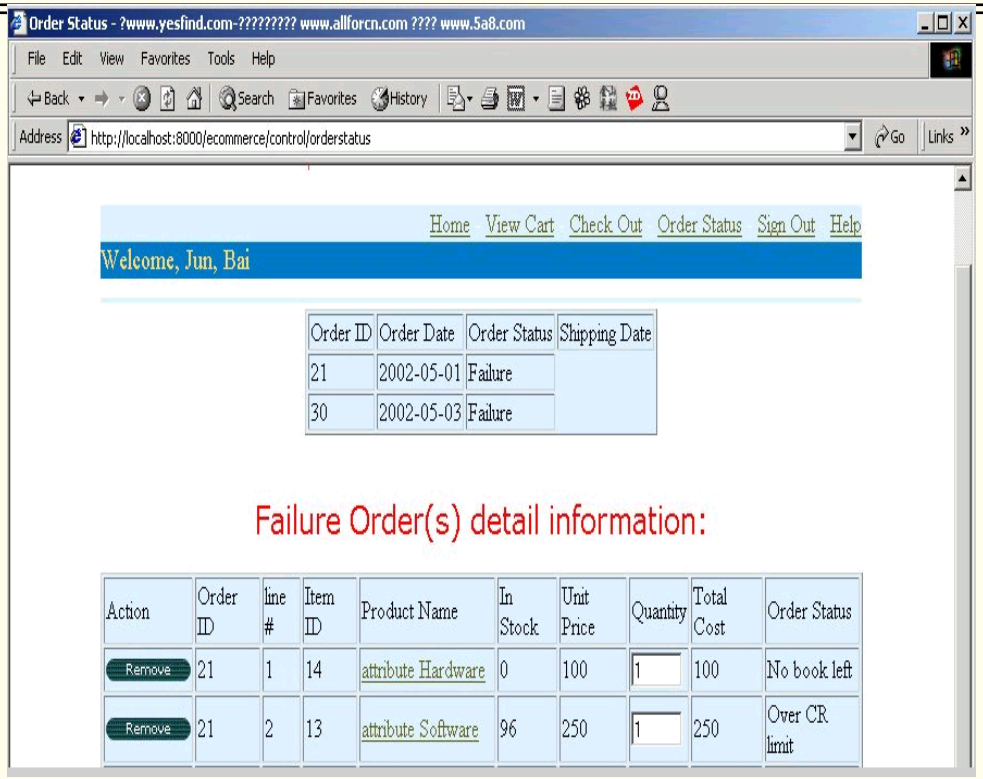


Exception handling:

- Changes to e.g. book data
- Locked data
- Transaction failure e.g. order rejected

Inform user & support LRT across systems

Exception Handling



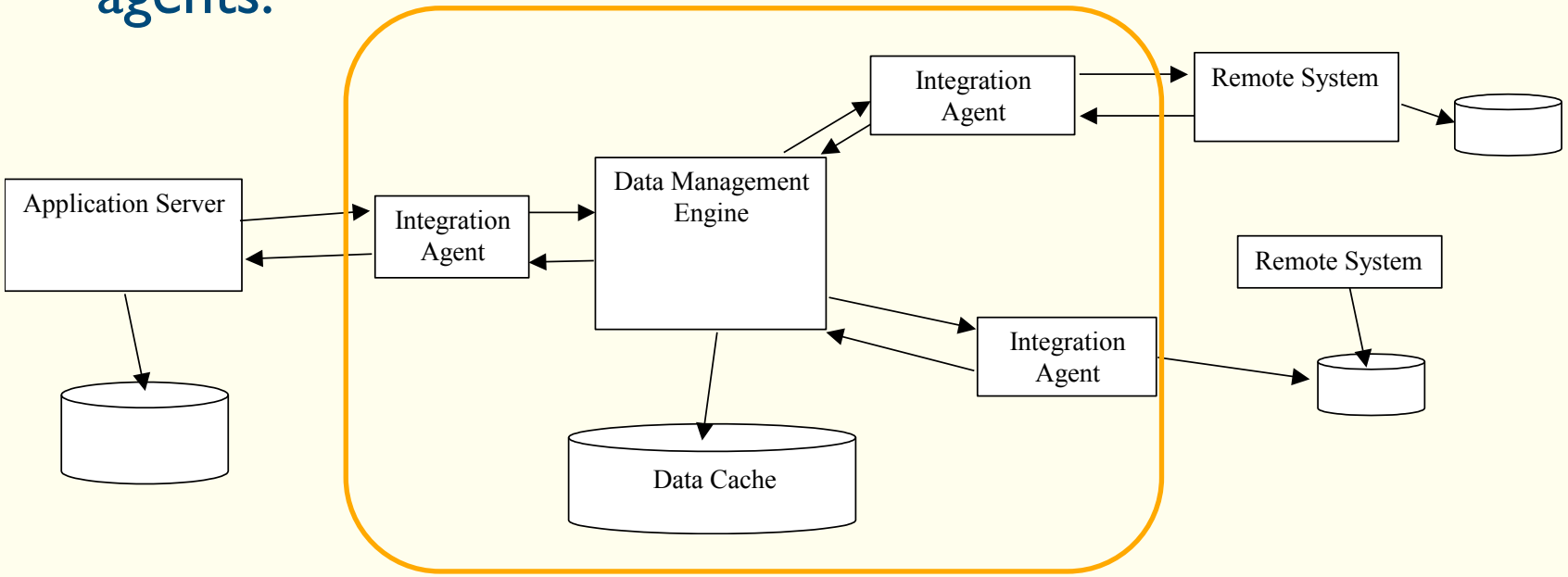
Exception handling:

- Changes to e.g. book data
- Locked data
- Transaction failure e.g. order rejected

Inform user & support LRT across systems

Current Work

- Developing a generic integration engine + integration agents:



See: www.xsol.com

Summary

- Evaluated with:
  - Performance analysis
  - Multiple users of customer/publisher systems
- Performance for most tasks very good; exception handling suitable; users generally satisfied with system
- Integration approach isolates systems and provides data-oriented mechanism to integration problem
- Generic data integration prototype developed