

# **Application Integration Solutions**

Cyberweave LLC

Anil Gupta & Suresh Rao

April 2004

## Integration issues

### **Migration/integration components**

Legacy applications currently dominate

Component/Service based applications will grow

Need to integrate with existing legacy apps

### **Browser based solution ideal for inter/intra nets**

Browser is ubiquitous

User interface standardized among apps

Leverage existing security protocols HTTPS and SSL

Minimal installation issues

## Thin client versus thick client

### **II tier Client/Server**

Only needs Web Server – like Apache or IIS  
Applications Bundled into Jars/Dlls in Web folders  
Needs JRE/ASP.Net runtime in client workstation  
Ideal for simple applications

### **III tier Middleware**

Web Sphere, Web Logic or Apache/Tomcat  
Full control over application integration issues  
Logical separation of presentation and business logic  
Most complex applications will benefit  
In most cases no need for client side software

## Strategy Considerations

Software standards constantly evolve

The pace of evolution is forcing shorter development time span

Large-scale projects very often fail to meet expectations

The luxury to spend years in system redesign is not there

Installations burdened with large-scale legacy applications

- Cannot be redesigned effectively
- Ongoing maintenance is still required
- Very difficult to re-engineer on a piece-meal basis

## Large Vendor packages

Single vendor package cannot fit diverse applications

Drain on corporate resources

- Manpower investment to redesign considerable
- New Hardware/Software costs must be justified

Locks up applications to vendor specific technology

Applications should be modified to fit into vendor technology

Licensing / Maintenance issues

## **Application/Code Development Tools**

Fits into specific technology/architecture

Less expensive to customize since native code is involved

Can be integrated into other frameworks with less risks

Typically vendor bundled IDE for porting/code generation

Increased productivity due to known application interfaces

## Application Integration Strategy

Best implemented in phases with a bottom-up approach

**Phase I** Business Logic Encapsulation

**Phase II** Common middle tier integration

**Phase III** Service Oriented Architecture

## **Business Logic Encapsulation**

Reusable code for business functions

Avoid presentation at component level

Standardized session access API to be defined

Session API on top of Host/SQL/MQ base functions

- TN3270/TN5250 host access API for CICS/IMS
- Industry standard JDBC/ODBC for SQL databases
- Message oriented Middleware for message queues

Core application functions split into adapters

- Stand-alone mode testing prior to integration
- Should install as an independent application with
- Applet/Servlet/ASP.Net wrappers

Adapters should work in Request/Response mode

## **Middle tier integration**

Overwhelming case for XML

License-Free, platform independent and universal  
Standard glue for connecting diverse applications  
SQL Server, DB2 and Oracle already support XML

Application Code generator should automate

Parsing XML streams into adapter request parameters  
Building XML streams from adapter response data  
Handle XML document validation

# Service Oriented Architecture

High-level abstraction

- J2EE, CORBA and DCOM are tightly coupled
- SOA is loosely coupled architecture
- 

Supports both online and batch mode interactions

Flexible programming model

Uses Standard HTTP(S) transport medium

Very wide support from all software vendors

Does not replace existing technology, complements it

Still need procedural languages to develop XML bridges

## **Options available for businesses**

No off-the-shelf package can solve all problems

Vendor's end-to-end solutions still need connectors/Adapters

Application code generators provide development relief

Prudent investment in in-house development will pay off

## Cyberweave Toolkit for Application integration

Best of the breed toolkit

Supports Java as well as .Net technology

Legacy emulator with record and playback options

Tightly coupled IDE for front end design

SQL wizard simplifies ad-hoc queries

Applications can be designed in days to weeks

Wide choice for adapter outputs

- XML for middle tier components \*
- Applets for rich client experience
- Servlets for thin client design

\* Release date - December 2004