Issues in Transaction Integration with a Case Study

Biplav Srivastava, Ph.D.

Data v/s Transaction Sources

- Transaction sources are dynamic; Data source can be static.
- Transaction sources show a process at the source; data sources may show none.
- Correctness of data involved in a transaction is critical. A data source will need no input.
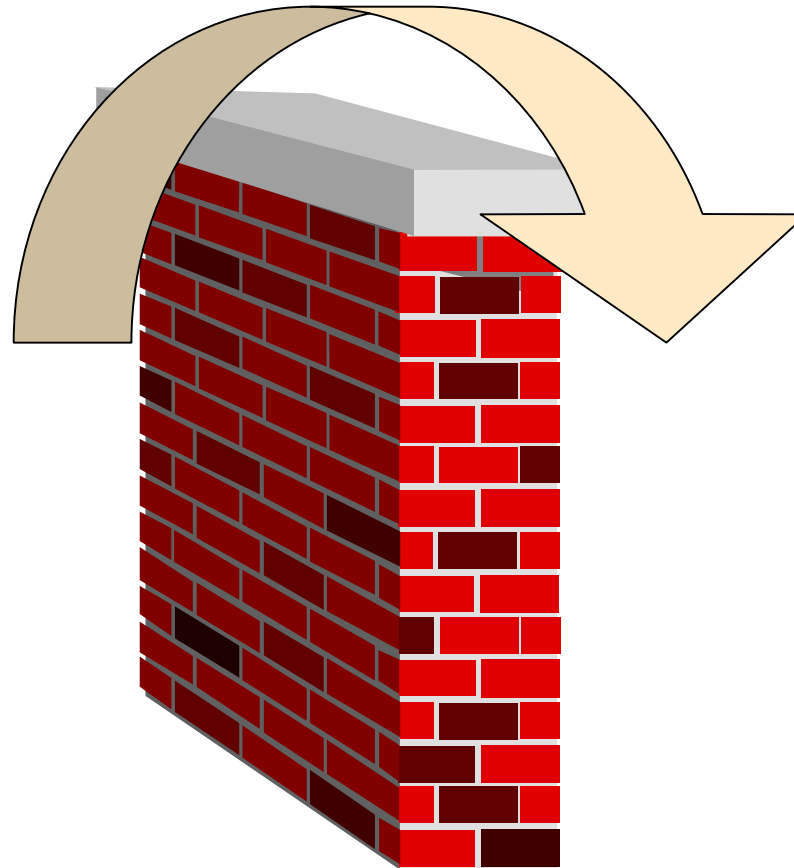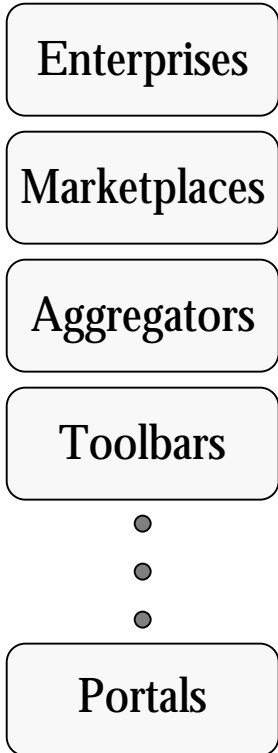
Integration Vision

Connect businesses with their
- Partners
- Suppliers
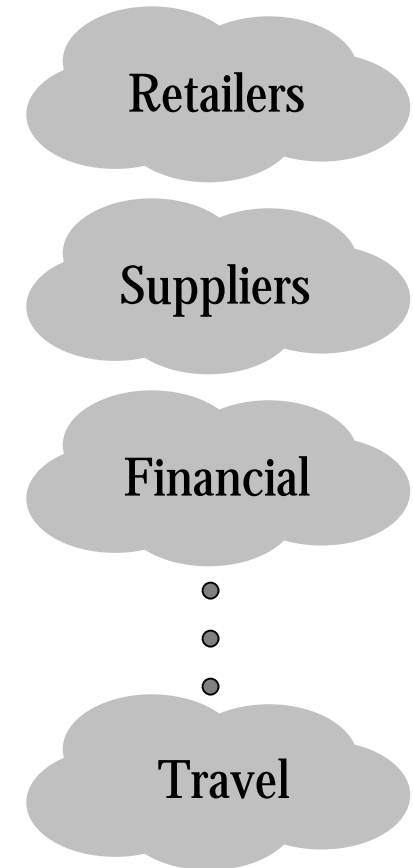- Customers, other businesses

While executing
- Rapidly
- With Zero-integration
- Leveraging existing web infrastructure

Problem

Businesses

- Enterprises
- Marketplaces
- Aggregators
- Toolbars
- ⋮
- Portals

Partners

- Retailers
- Suppliers
- Financial
- ⋮
- Travel

➢ Lack of integration in cross-enterprise interactions (sites, processes)
➢ Fragmented experience, no control/visibility of users' activities

Research Issues

- How to model process ?
  - Eliminate false-positive transactions – e.g., unauthorized purchases
  - General framework
- How to build wrappers ?
  - Fast and resilient
  - Fast development
- How to monitor changes ?
  - Content change v/s down time; Batching
- How to do all in Economical, Scaleable and Secure manner ?

Modeling Process

- Example: Purchase of an item
- Declarative specifications are general
  - Operation: Read, Write
  - Role: Accounts, Ship, Tax, Details, Order
- Overall process
  - Procedural description of flow
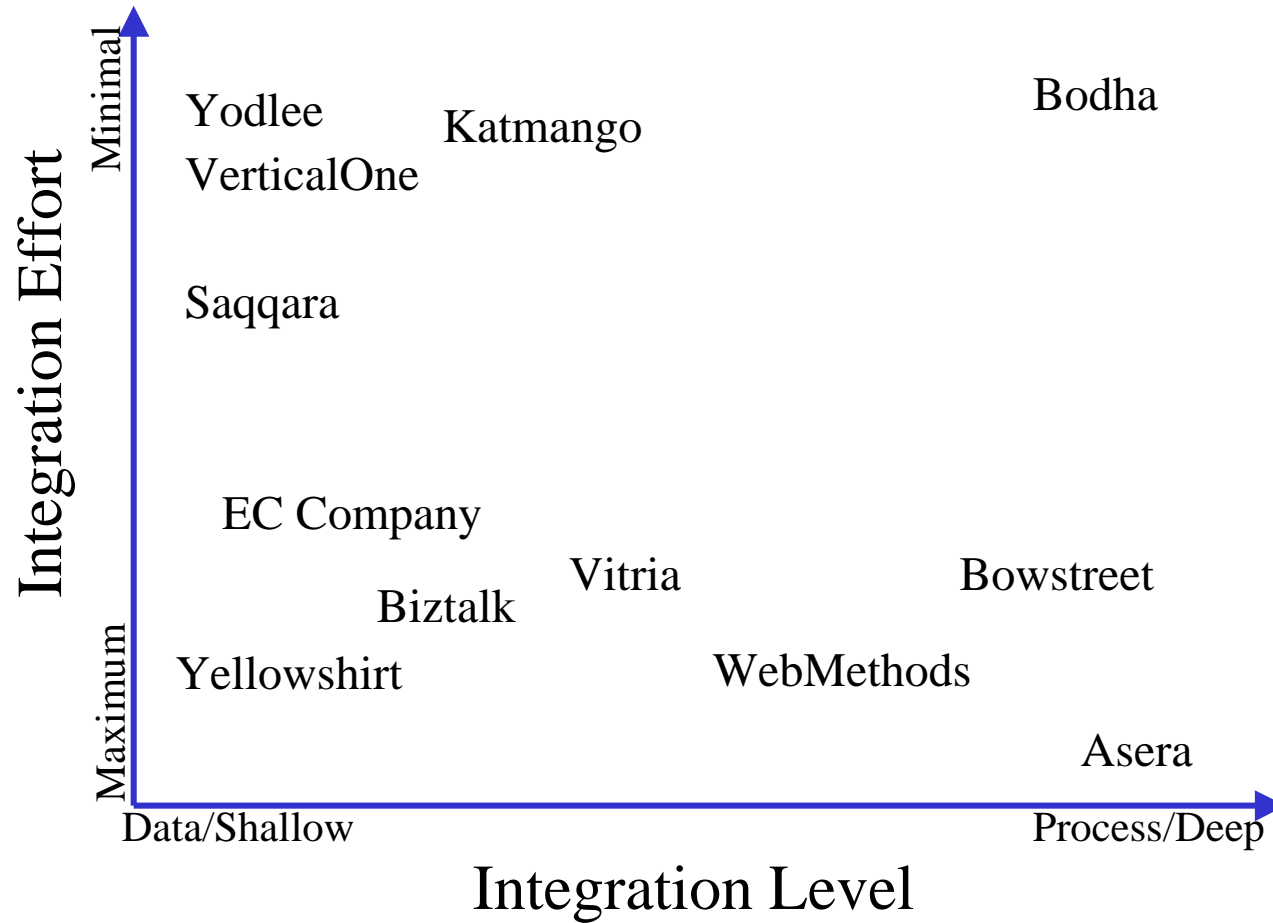 - Source-specific customization

Wrapper Choices

Interaction	Wrappers	Comments
Cooperation	XML-interface	Easy to maintain, Robust, Fast
Loose but gives advance change information	Prefix/Suffix without regular expression	Medium maintenance, Fragile, Fast information
	Proxy with limited DOM manipulation	Medium maintenance, Robust, Fast
No co-operation	DOM element manipulation	Medium maintenance, Medium robust, Medium speed
	Prefix/Suffix with regular expression	Medium maintenance, Medium robust, Fast
	Proxy with limited DOM manipulation	Medium maintenance, Medium robust, Fast

Monitoring

- ## Source changes
  - Page changes: content v/s style
  - Implementation changes (do we care ?)
  - Process changes
- ## Source downtime
  - What to do: batch transactions ?
- ## Reliable, Economical and Fast deployment
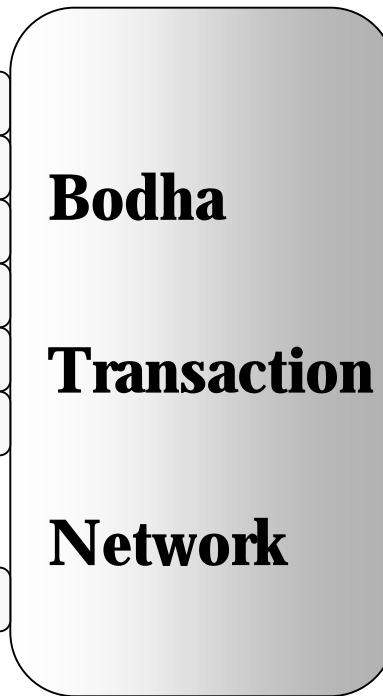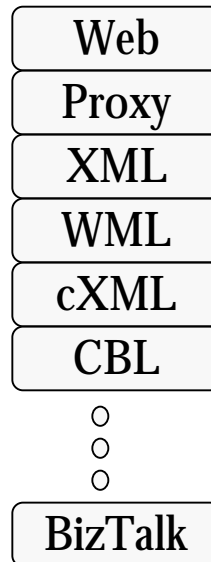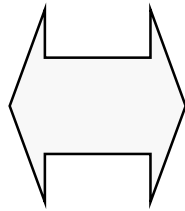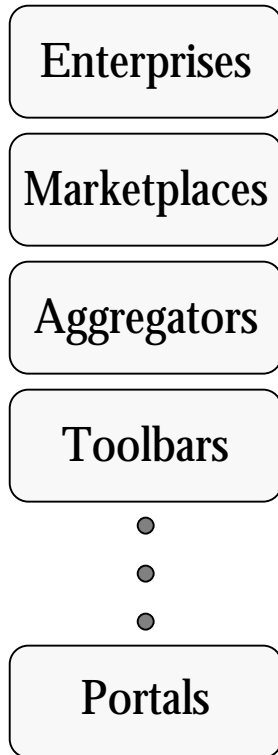 - Self correction
  - How to guarantee uptime ?

Bodha Integration Framework

Competitive Landscape

Bodha Solution
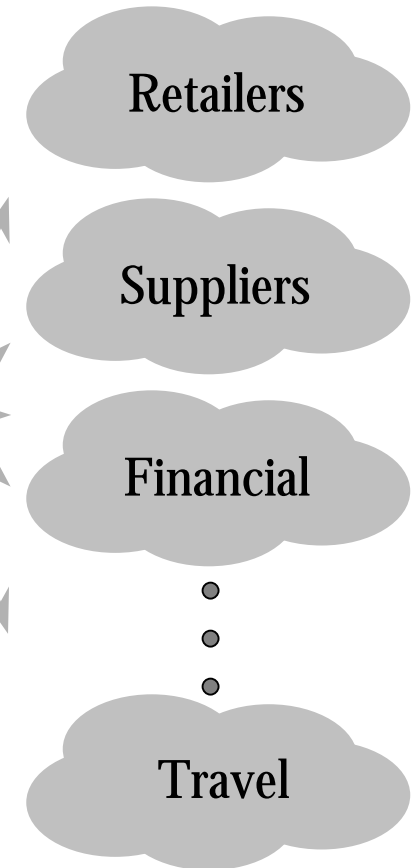
Marketplaces

Transaction Centers

Enterprises

Marketplaces

Aggregators

Toolbars

Portals

Web
Proxy
XML
WML
cXML
CBL

BizTalk

Bodha

Transaction

Network

Retailers

Suppliers

Financial
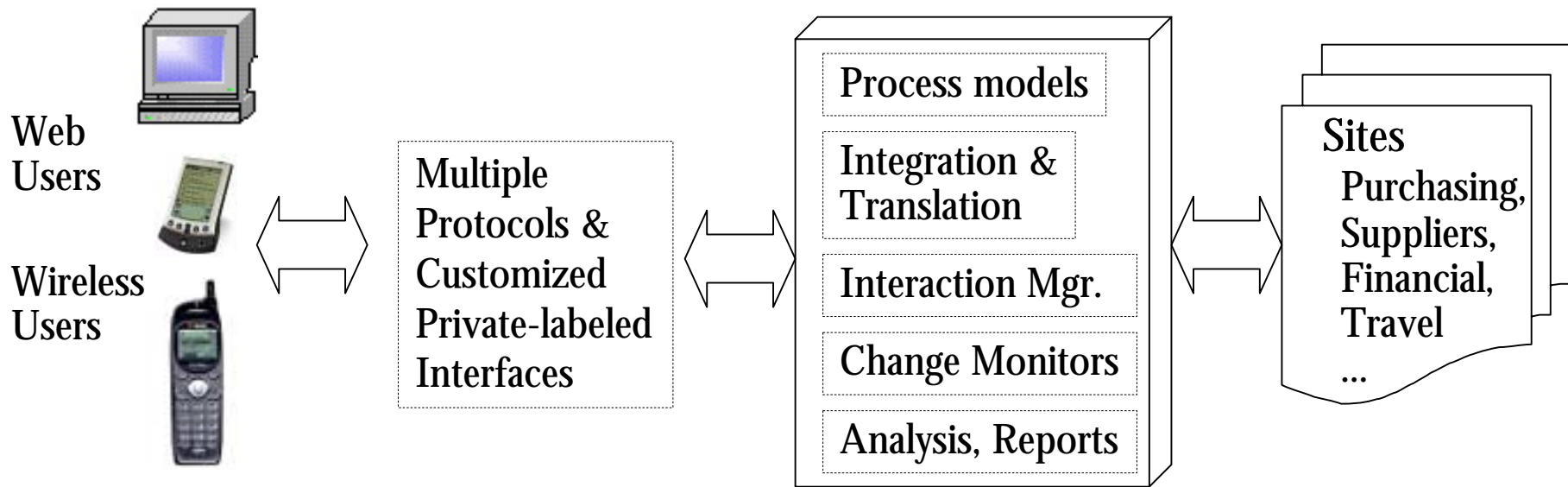
Travel

➢Zero integration for partners, minimal integration for businesses
➢Seamless interactions, seamless participation

BTN Solution Overview

Web Users

Wireless Users

Multiple Protocols & Customized Private-labeled Interfaces

Process models

Integration & Translation

Interaction Mgr.

Change Monitors

Analysis, Reports

Sites
Purchasing, Suppliers, Financial, Travel ...

- Wrappers for unique process models for sites/applications
  - All interactions with underlying apps/sites replaced by Wrapper interactions (e.g. for shopping, **R**ead, **W**rite, **D**etails, **O**rder-status, **A**ccount, **S** & **T)**
  - Overlay multiple process flows to front-end; bring up all flow options
- Build wrappers rapidly using advanced information integration technology
- Scalable, secure solution (use underlying sites/apps' security protocols)
- Automatic detection, quick repair of changing application/site formats

Integration Technology

- Based on BIL™ (Bodha Integration Language)
  - Framework for rapid, accurate development of wrappers
  - Represent interactions as BIL scripts
  - Supports myriad interaction complexities, conditionals
  - Use proprietary parsing library; Small footprint parser
  - Includes a proprietary extension of DOM
- Mediation based translation, information integration
- Integration over web sites, databases, other data sources
- Extending to add location-independence, content based identification, automatic BIL generation, XML format

Site Interaction Manager™

- Fast, secure, safe programmatic interaction with web sites

- Low foot print for scaling heavy concurrent usage

- Script controllable, preserve state

- Handles

  – HTTP1.0/1.1, Javascript, ECMAScript, Frames, Redirection

  – Transparent, correct, fast cookie and session management

  – Automated error correction on site HTMLs

  – All varieties of SSL libraries

- Fast, small footprint, built-in HTML parser

Change Monitoring Overview

- Framework for quick, automatic detection of site change

- Meaningful change (minimizes false positives, false negatives)

- Based on candidate queries, feature extraction

- Implemented as a background service; automatic removal of affected stores, notification

- Extending to improve accuracy, location-independence

ParserGen™

- Framework for quick, automatic wrapper/parser creation

- Supports multiple parser flows, HTML formats

- Simple, intuitive XML format, GUI tools

- Integrated quick test, change detection

- Extending to

    – Location-independence

    – Native BIL compilation

    – Resilience

Generalized Proxy Framework

- Fast, run-time proxy of any web site

- Handle all varieties of HTML, Javascript, cookies/sessions

- Seamlessly enables

  - Link-rewriting

  - Link-insertion, change of control-flow

  - Logging, parsing, recording, analysis

  - Multiple front-ends, UI displays

  - Dynamic administration

Architecture

Process Overview

Three-tiered environment

Client-tier code

- Presentation layer
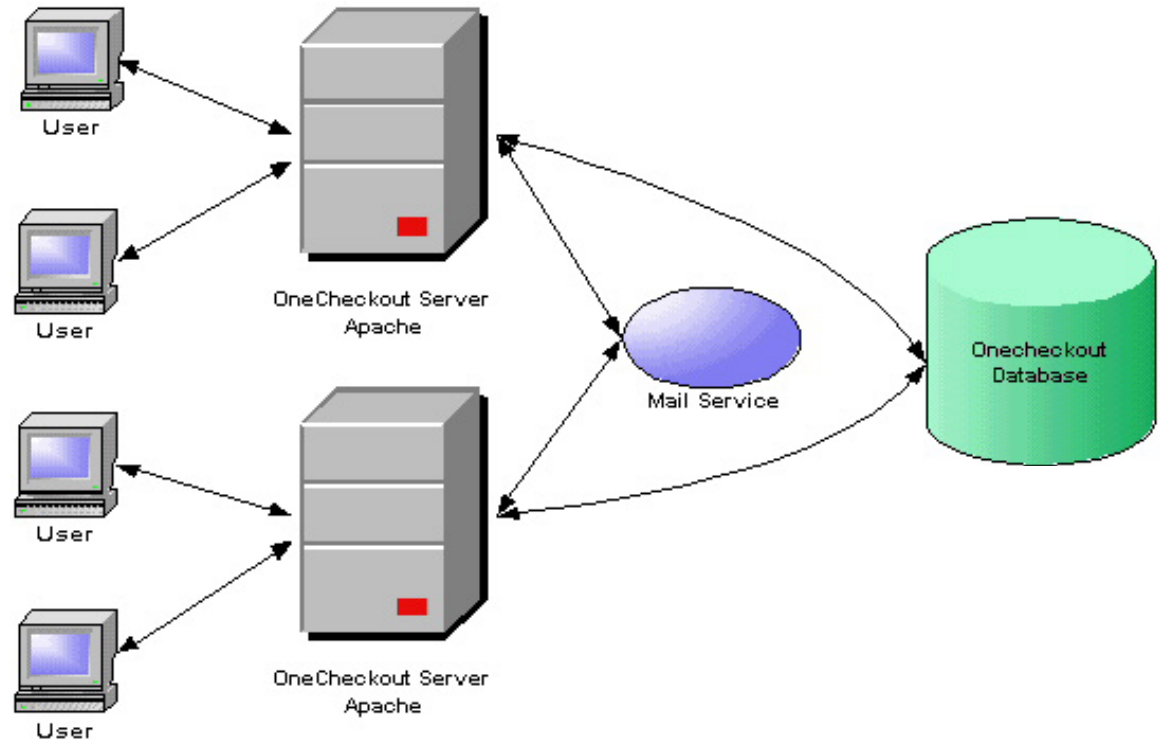- Client adapters / Output views

Middle-tier code

- Core components
- Application logic, core functionality

Database-tier code

- User info (orders, membership, billing)
- Stores, product info

Flow (MVC)

Design pattern for presentation/content separation

Beans (Model)

  process specific tasks

JSPs (View)

  Retrieve beans

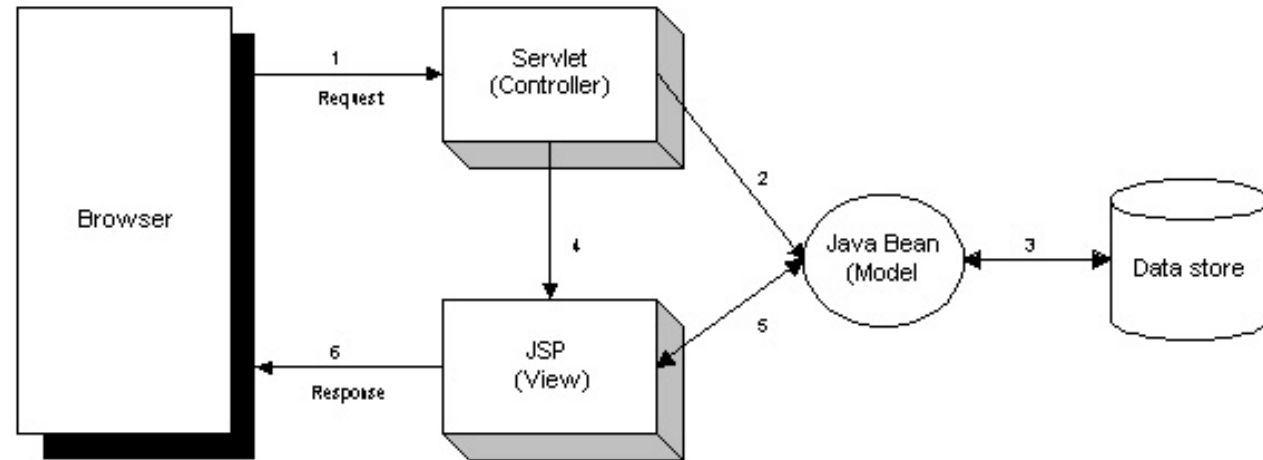 Extract content & present

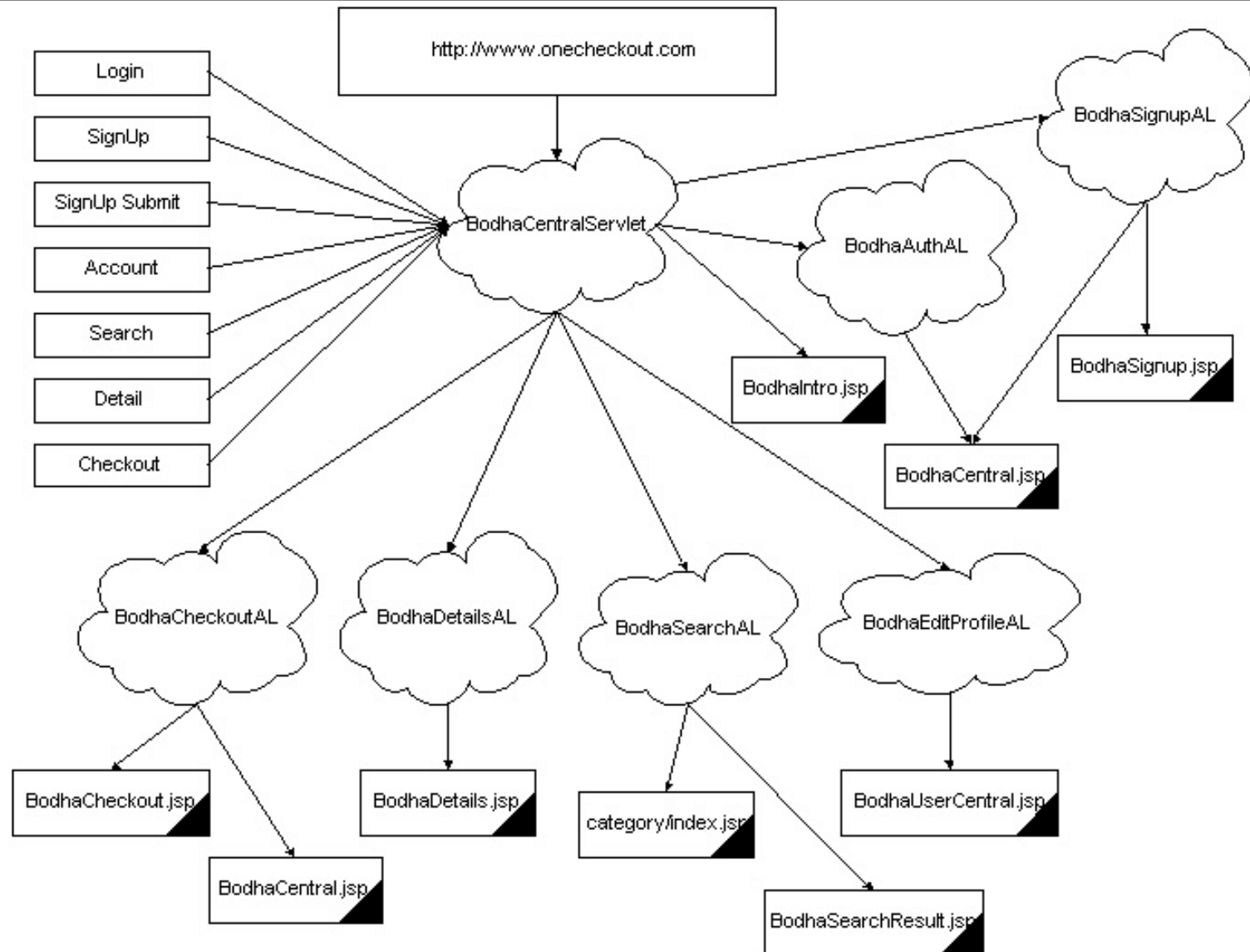Servlets (Controller)

  Input validation

  Request processing

  Bean/object creation

  Invoking JSPs

Front-end Interaction Flow

Practical Considerations

- Build security and scalability into the architecture

- No-frills, discourage gratuitous use of technology, question assumptions (Occam's Razor principle)

- Good net citizens (privacy, crawling, use of data)

- Reliable platform, partners: Apache, Oracle, Sun-JDKs

Demonstration

Summary

- Transaction Integration pushes data integration to the next level through process flow integration

- XML-based integration is ideal but not always practical. Demand for leveraging current infrastructure and thus eliminating the huge deployment challenges faced by other solutions

- A working example in Bodha Case Study