Enterprise Mashups: An Industry Case Study

PHP meets Web 2.0

Dan Gisolfi
Internet Emerging Technology, IBM

New York PHP Conference & Expo 2006
THE PHP BUSINESS COMMUNITY
Why is IBM investing in PHP?

Enterprise Mashups: An Industry Case Study

- PHP empowers another segment of developers - expanding solutions for customers

- Broad community support & participation
  - Architecture by Participation
  - Users as co-developers - DIY Apps
  - Re-use & assembly culture

- Light weight programming model thru web services & REST
  - Embracing values for integration & interoperability

- Strong business value synergy & linkages
  - SMB & mid-market
  - Departmental
  - IT

IBM Initiatives
- Zend Core for IBM - Information Management
- Zend Core for iSeries
- SWG Emerging Technologies
  - PHP + SOA
  - Web 2.0 Instant Apps
  - Ajax Tooling & toolkits

- Re-use & assembly culture
- Architecture by Participation
- Users as co-developers - DIY Apps
- Embracing values for integration & interoperability
- SMB & mid-market
- Departmental
- IT

IBM Initiatives
- Zend Core for IBM - Information Management
- Zend Core for iSeries
- SWG Emerging Technologies
  - PHP + SOA
  - Web 2.0 Instant Apps
  - Ajax Tooling & toolkits
IBM Web 2.0 Technologies Current Focus Areas

Enterprise Mashups: An Industry Case Study

• Web 2.0 Technologies converging on a few key value proposition

• Broad Collaboration

• Simplicity & rich(er) internet experiences

• Remixability
  • Enabling “applications” that can be created by non-professional programmers
  • APIs based on open (defacto) standards

PHP On Forefront of Opportunities
• It’s about instant results
• It’s about empowering line-of-business professionals
Web 2.0 - What technologies are we talking about?

Enterprise Mashups: An Industry Case Study

- Many Web 2.0 Technologies still in innovation stage

- Customer Interest High In:
  - AJAX is most tangible in terms of potential business value
  - RSS/Atom - RSS & Atom/APP being seen as potential approaches to simplify specific content centric application architectures
  - Programmable Web - potential seen in building/extending business ecosystems
  - Web 2.0 “instant” applications
Web 2.0 - What customer pain points can it address?

**Customer Pain Points**

- Internet technologies continue enabling Businesses to **expand their ecosystems & partnerships**

- Partnerships cause **integration** work items it IT - usually requiring a minimum of 6 months per request

- **Change Happens** a number of business collaborations last less than 12 months - do the math

- **Attention Economy Factor** keeping a pulse on external business data that **could** impact results is constantly changing

A whole bunch of apps not being written today because they’re not affordable

**What’s Should Businesses Do?**
Web 2.0 - Rethinking Application Assumptions

How would we design middleware if assume:

• business organizations & relationships are continually changing - therefore solutions need are situational

• LOB teams just enough IT savvy to create their own services/solutions that drive their part of the business (Igniting the Phoenix: A New Vision for IT/Sapir)

• ...applications are disposable

“Situational” Apps

• Built to solve an immediate, specific business problem

• Blending externalities with business-private content & services

• Manipulates static & increasingly dynamic content information-centric

• Accelerated by community-based collaborations
Application Wikis are environments for **collaborative**, situational (ad hoc) dynamic content development

- Facilitate web solutions for non-programmers who are domain experts - i.e. Mash-ups, dashboards, etc..

- Further mark-up based client development strategy - plug-in model for easy extensibility

- Weave together a “good enough” solutions by scripters

- Combined runtime & light weight assembly capabilities
Case Study: Media & Entertainment Industry

• Background
  • Provide members of National Association of Broadcasters within the Media & Entertainment industry with lifecycle enabling tools for all aspects of media production.

• Challenge
  • Typical media production projects are of short durations (3 weeks, 3 months, 6 months..) thereby limiting justification for IT investment
  • Media production lifecycle solutions are often standalone and proprietary
  • Application development resources are siloed and disjoint from infrastructure IT

• Solution
  • Deploy IBM’s Media Hub, an enterprise service bus with media extensions and workflow capability
  • Integrate QEDWiki to provide
    • informal, just-in-time access to domain content supplied by Media Hub
    • Production assistants with the ability to create their own situational application for each individual project
    • IT groups with the ability to support hundreds of situational applications for short-term projects

• Benefit
  • Media production teams are empowered to create short-lived ad-hoc applications
  • Single point of visibility throughout entire organization
  • Flexibility of application tailoring for a variety of users
  • Potential simplification of support for IT organization
  • Ability to provide applications for shorter term projects. (3 weeks - 3 months)
Case Study: Solution Architecture

Media Hub is an enterprise-service-bus with media extensions and workflow capability based on IBM’s service-oriented-architecture using IBM’s WebSphere Application Server.
Case Study: Demo

- Created for the Annual National Association of Broadcasters Conference
  - Example *Mashboards*

  - Long Form production Workflow

  - Content management for media production Editors, Producers and Directors

  - **Resource Utilization Monitor**

- Short Form Production Workflow

  - Event based resource allocation for Journalists
Case Study: Resource Utilization Monitor

• Scenario:
  • Within the Media industry the majority of workers use timecards to track resource activity pertinent to a specific project.
  
  • Our Project Manager for the “Picasso Code” documentary needs a resource utilization mashboard to manage resource allocations for the project.

• Demo
  • Our current production project, code named Guernica after the famous Picasso painting, contains several production editors located in remote locations across the country.
  
  • Each editor is billing us at different contract rates and they are utilizing editing bay facilities that range in cost per location.
  
  • Our demo allows us to see that:
    • Peter has been inefficient, going over his time and cost budgets
    • Ed has been tracking inline with his budget
    • Fred has been very efficient, tracking under his time and cost budgets
QEDwiki - Architecture Overview
**QEDwiki - Architecture Overview**

Web 2.0 Instant Applications

1. Initiate Session in DB
2. Check ACL
3. Check/Start new version
4. Connect & collect e-mail sent to wiki page

![Diagram of QEDwiki architecture]

- **Commands**
  1. Pub/Sub component model
  2. Connect to external/internal source & return content
  3. Execute in-line PHP script
  4. Store Results
  5. Repeat until all Cmds processed
  6. End version & gen RSS
  7. Compose Page (Recursive)
     - General embedded wiki's
     - Compose Page Style
     - Use AJAX, etc.
     - Format for Presentation
  8. Return to Browser

---

```javascript
//** Results from Google will be displayed
// in a table. Items will have a drag icon
// on the right column. Dragging onto Yahoo maps to
// immediately locate an office & see any weather related info.

<
  google
  search="IBM New York Address" data="results" drag="true"
  yahooMap
    id="map" acceptLists="* *" }
  noaaforecast
  rssfeed
    url="http://www.weather.gov/alerts/ts.rss"
//** End of WikiPage, versioned, saved & immediately deployed
```
Conclusions & Summary

**Information**
- Content thru feeds, messages & documents
- Data integrity for transactions

**Integration**
- Mash-ups - bringing together content from multiple sources
- Seamless collaborations across business boundaries

**Independence**
- Freedom to employ any & all applications/services
- Application, platform, programming language, and delivery device

**Interoperability**
- Sharing content across applications & platforms
- Loosely-couple programming model

**Internationalization**
- Globally linking businesses

*PHP is going to continue fueling the Business Innovation-Integration cycle*

*...a web of data sources, services for exploring & manipulating data, and ways that (end) users can connect them together instantly*

*Tom Coates/Yahoo*