



# Thots on Web Applications

NALIT 2008 – Louisville, KY

# Menu to Thots

Web Application - Advantages

Web Application - Disadvantages

The Circle of Life

Who is doing it?

Site Specific Browsers

Web Application Environment

Web Application Tools

Model View Controller

Creation Steps

Security

# Advantages of Web Applications

- Uses browser so PC configuration is less of an issue.
- Familiar interface (browser).
- Centralized data store.
- Deployment and updates are easier.
- Collaboration is facilitated.
- Many flexible tools for development (many are open source).
- Monitoring of use is easier.
- One app can deliver different looks.

[Back to Menu](#)

# Disadvantages of Web Applications

- Tend to have slower response times.
- Interfaces tend to be less sophisticated and capable.
- GUI development is traditionally more difficult.
- Different security concerns.
- All eggs in one basket, centralized.

[Back to Menu](#)



# The Circle of Life

Big Web  
Servers with  
Browsers  
Attached

Networked PC's  
Client Server

Anarchy Rules  
Web 2.0

[Back to Menu](#)

# Who is doing it?

- Difficult to find similar applications.
- Many web based work flow solutions developers.
- Kodak – Newsmanager 1.1
- Others?

[Back to Menu](#)

# Site Specific Browser

- Special browser tailored to application.
- Mozilla Prism
- Adobe Air
- Google Gears
- Fluid

[Back to Menu](#)

# Web Application Environment

- Common Acronyms
- L.A.M.P. (Linux, Apache, MySQL, PHP or Pearl or Python or Programming Language)
- W.I.M.P. (Windows, IIS, MS SQL, Programming Language)
- Others

[Back to Menu](#)



# Web Application Tools

AJAX (Asynchronous JavaScript and XML)

Silverlight

WPF (Windows Presentation Foundation)

Ruby on Rails

ASP

ColdFusion

Struts

XHTML

Perl

PHP

Java applets

ASP.NET

Python

Database (MySQL, PostgreSQL, MS SQL etc)

Flash

CGI

[Back to Menu](#)

# Model View Controller

MVC encompasses more of the architecture of an application than is typical for a [design pattern](#).

## Model

The **domain**-specific representation of the information on which the application operates. Domain logic adds meaning to raw data (e.g., calculating whether today is the user's birthday, or the totals, taxes, and shipping charges for shopping cart items).

Many applications use a persistent storage mechanism (such as a [database](#)) to store data. MVC does not specifically mention the data access layer because it is understood to be underneath or encapsulated by the Model.

## View

Renders the model into a form suitable for interaction, typically a [user interface](#) element. Multiple views can exist for a single model for different purposes.

## Controller

Processes and responds to events, typically user actions, and may invoke changes on the model.

[Back to Menu](#)

# Creation Steps

- The future for web applications consists of an architecture that consists of:
  - A data model + functionality
  - Composition of data for the interface
  - Mapping of this composition onto an presentation and interface
- This architecture offers advantages such as:
  - Device independent applications
  - Accessible applications
  - Order of magnitude savings in programming
- By using the backplane approach, it allows you to plug different application elements (Forms, Multimedia, Graphics) together, without committing to an output device until the last minute. XBL allows you to map the composition to a final presentation. Steven Pemberton - <http://homepages.cwi.nl/~steven/>

Back to Menu



# Security

- IBM Smash (Secure Mashup) “... allows information from different sources to communicate with one another while keeping the respective sources separate to prevent the transfer of malicious code. – GCN – Aug 4, 2008
- WASC (Web Application Security Consortium)
- OWASP (Open Web Application Security Project)

[Back to Menu](#)