

GEMSTONE

GLASS



GemStone, Linux, Apache, Seaside, Smalltalk




Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Interest Inspired by Ruby on Rails


- ▶ Ruby is a object-oriented language
 - Reflective, dynamic, interpreted
 - Provides much Smalltalk functionality
 - Syntax broadly similar to Perl and Python
 - Appeared in 1995
- ▶ Ruby on Rails is a web application framework
 - Provides out-of-the-box scaffolding which can quickly construct most of the logic and views needed for a basic web site

2 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Why the interest in Seaside?

- ▶ Ruby on Rails has provided a buzz that is moving developer mindshare into Ruby as a web development environment
- ▶ Seaside is serving a similar role for Smalltalk
 - Not as complete as Rails (no DB support)
 - What it does, it does much better



3 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Example of Commercial Use

▶ <http://www.dabbledb.com/>



4 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

What is Seaside?

- ▶ Squeak/Smalltalk
- ▶ Enterprise
- ▶ Application
- ▶ Server
- ▶ Integrated
- ▶ Development
- ▶ Environment

5 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

What is Seaside?

- ▶ Seaside is a framework for developing sophisticated web applications
- ▶ Seaside provides a layered set of abstractions over HTTP and HTML that lets developers quickly build highly interactive web applications
- ▶ Application is written in Smalltalk
 - Framework uses TCP/IP for request/response
 - Application uses Smalltalk message sends
 - HTML markup is hidden
- ▶ No client GUI is involved

6 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

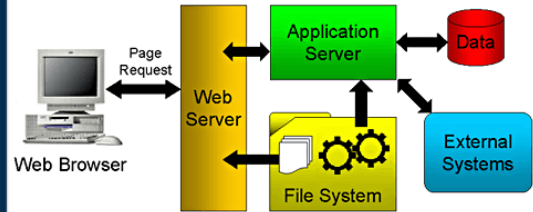
Where did Seaside come from?

- ▶ First, Avi Bryant created a web development framework in Ruby modeled on Apple's WebObjects
- ▶ Second, Avi ported his framework to Smalltalk to take advantage of "a generally richer environment and set of libraries and a deeper, more experienced community"
- ▶ Third, Avi added *continuations* to Smalltalk to avoid limitations of other approaches

7

Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

A Canonical Web Architecture



http://www-106.ibm.com/developerworks/library/it-booeh_web/

8

Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Why is Seaside Attractive?

- ▶ Building complex web apps is hard
 - HTTP was designed to serve static pages. It provides a request/response model, where no *state* is saved across pages
 - Any non-trivial application requires *state*, which calls for various hacks
 - Other frameworks make it difficult to build user interface in an abstract, maintainable manner.
 - Code for each UI interaction must know about others—it can't stand alone
 - <Back> button creates havoc with saved state

9

Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

How Does Seaside Solve the Problem?

- ▶ Each served page is associated with a *continuation* (or a program stack), that can be resumed when the user submits a subsequent request
- ▶ This magic allows the programmer to treat each web interaction as part of a series rather than as a stand-alone request/response
- ▶ Disparate components can also be mixed on a web page with more ease

10

Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Demo

- ▶ Smalltalk
 - WACounter

11

Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Aside on Some Smalltalk Dialects

- ▶ Squeak
- ▶ VisualWorks (Cincom)
- ▶ VA Smalltalk (Instantiations)
- ▶ Dolphin (Object Arts)
- ▶ GemStone



12

Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

What is Needed to Support Seaside?

- ▶ Smalltalk needs to allow *continuations* in order to support Seaside
 - Supported in
 - Squeak
 - VisualWorks
 - Dolphin
 - GemStone/S (as of 64-bit 2.2.0)
 - Not supported in
 - VA Smalltalk (yet)

13 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Demo

- ▶ Smalltalk
 - BlockClosure>>callCC
 - Continuation

14 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Why Port Seaside to GemStone/S?

- ▶ The other dialects of Smalltalk are single-user and non-persistent
 - This means that a Seaside application needs to work around built-in limitations to handle multi-user persistence
- ▶ GemStone's value has always been providing scalable multi-user persistence to a Smalltalk application
 - GemStone's lack of a GUI is okay!

15 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Challenge of Persistence

- ▶ Persistence approaches in Smalltalk
 - In the image
 - Loss of data if image quits
 - Not shared across images
 - In a binary file-out
 - Limited size
 - Object identity is not be preserved
 - In an external database
 - Object/relational mapping overhead
 - Extra coding to foreign interface
- ▶ GemStone/S solves this problem!

16 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Challenge of Multi-User Coordination

- ▶ Multi-user approaches in Smalltalk
 - One image serving multiple clients
 - Requires layer directing query to image
 - Scalability limit
 - Coordinate through external database
 - Object/relational mapping issues
 - Extra coding to foreign interface
- ▶ GemStone/S solves this problem!

17 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Challenge of Scalability

- ▶ Scalability approaches in Smalltalk
 - Add hardware
 - Still basically single-threaded
 - Run more images
 - Presents all the persistent/multi-user issues
- ▶ GemStone/S solves this problem!

18 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Web Server in Smalltalk

- ▶ Seaside requires a web server
 - Built-in
 - Swazoo
 - Hyper (branch from Swazoo)
 - Kom (formerly Comanche)
 - External – requires FastCGI
 - Apache (<http://apache.org>)
 - Lighttpd (<http://www.lighttpd.net>)
 - We want to support both internal (for developers) and external (for production)

19 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Port Hyper to GemStone/S

- ▶ Hyper is Bruce Badger's branch of Swazoo's HTTP server
- ▶ Bruce has a version running on an earlier version of GS/S
 - GsRandom
 - Sport
 - OpenSkills
 - Hyper

20 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Demo

- ▶ Smalltalk
 - HTTPServer
 - OSkSocketListenerService

21 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Where Do We Get Seaside?

- ▶ The "official" version is in Squeak
 - <http://www.squeaksource.com/Seaside>
- ▶ Existing Seaside ports to
 - Dolphin
 - VisualWorks
- ▶ Typical porting process is somewhat complex
 - Export from Squeak
 - Import into other dialect

22 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Source Code Control for Squeak

- ▶ *Monticello*
 - A distributed concurrent versioning system based on a declarative representation of Squeak source code
 - It allows packages to be safely saved, loaded and updated, and provides tools for branching and merging package versions
 - It does not depend on (and does not provide) any form of central repository

23 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Port *Monticello* from Squeak to GemStone/S

- ▶ Complexities of multi-user image
 - Each user can load different packages
 - A package can add/replace methods for classes in another package
 - What if other class/method is shared?
- ▶ GsPackage
 - sessionMethods
 - Prereqs
 - symbolDict

24 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Session Methods-1

- ▶ Traditional method lookup:
 - Array
 - SequenceableCollection
 - Collection
 - Object

25 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Session Methods-2

- ▶ New method search (top down):
 - SessionMethods for Array
 - Array
 - SessionMethods for SequenceableCollection
 - SequenceableCollection
 - SessionMethods for Collection
 - Collection
 - SessionMethods for Object
 - Object
- ▶ Method is cached after first lookup

26 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Demo

- ▶ Smalltalk
 - GsPackage
 - SessionMethods

27 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

How Do We Get Seaside?

- ▶ Load it from SqueakSource.com using Monticello
- ▶ GemStone/S compiler was modified to allow underbar as the assignment statement:


```
url _ 'http://seaside.gemstone.com/' .
```
- ▶ This allows code to be loaded directly
- ▶ This allows use of Monticello tools to compare versions w/o clutter

28 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

SqueakSource

- ▶ A Seaside application that acts as a server for Monticello packages
 - A Squeak image with SqueakSource installed can act as a server
- ▶ A particular Internet domain that hosts a SqueakSource server
 - <http://www.squeaksource.com/>
- ▶ GemStone/S can host SqueakSource as an installed application

29 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

How Does Apache Fit In?

- ▶ Most Seaside applications use a Smalltalk HTTP server (e.g., Kom)
- ▶ Large-scale applications will generally want a separate web server
 - Serve static pages
 - Handle SSL (https requests)
 - Load balancing
 - Fail-over backup
 - Security (hacker-resistant)
 - Separate server administration

30 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Demo

- ▶ Apache
 - /srv/www/htdocs/index.html

31 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

FastCGI

- ▶ FastCGI provides a well-defined, well-used protocol for communication between a web server (e.g., Apache) and an application server
- ▶ Rather than listening on port 80 on public host and serving all HTTP requests, listen on another port behind firewall and only serve Seaside requests

32 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Demo

- ▶ Apache
 - /etc/apache2/conf.d/mod_fastcgi.conf
- ▶ Smalltalk
 - FSDemoHandler->answerResponderRole:
 - FSGsDemoHandler
 - FSSeasideHandler

33 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Performance

- ▶ Pages per second
 - Squeak: ~30
 - VisualWorks: ~60
 - GemStone
 - Commit every page: ~60
 - Commit every 200 pages: ~200

34 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Scaling

- ▶ Multiple VMs
 - Each is a separate OS process
 - Each has full access to the database
 - Close to linear scaling
- ▶ Multiple hosts
 - Customer production systems
 - 1500 VMs; 200 hosts
 - Tested
 - 3000 VMs; 1 terabyte data; 16 billion objects

35 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Demo

- ▶ Multiple virtual machines
 - Browser to daleh-t60

36 Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Jade: GemStone/S Tools in Squeak - 1

- ▶ Client access is through a shared library (DLL on Windows)
 - GemStone C Interface (GCI)
- ▶ Use Squeak's Foreign Function Interface (FFI) to access GCI
 - Wrap C functions with Smalltalk methods

37

Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Jade: GemStone/S Tools in Squeak - 2

- ▶ User Interface available
 - Login
 - Workspace
 - Code browser (edit code)
 - Debugger (resume/terminate)
 - Method list
- ▶ Yet to come
 - Step/breakpoint in debugger
 - Inspector

38

Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Demo

- ▶ Squeak Tools
 - Login
 - Transcript
 - Workspace
 - Browser
 - Method List

39

Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Hosted Sandbox

- ▶ Avoid Setup, Configuration, and Management of GemStone Server
- ▶ Shared Server on Internet
- ▶ Apply for an Account
 - <http://seaside.gemstone.com/>

40

Copyright © 2007, GemStone Systems Inc. All Rights Reserved.

Question?

- ▶ Documentation (such as it is ;-)
 - <http://seaside.gemstone.com/>
- ▶ Mailing Lists
 - <http://www.seaside.st/Community/MailingList/>
 - subscribe-gemstone-smalltalk@earth.lyris.net
- ▶ Email
 - James.Foster@GemStone.com
 - Dale.Henrichs@GemStone.com

41

Copyright © 2007, GemStone Systems Inc. All Rights Reserved.