


# Some Virgo Repositories

steve powell - [spowell@vmware.com](mailto:spowell@vmware.com)

# artifact-repository

- only one project (bundle)  
**org.eclipse.virgo.repository**
- exporting packages:  
**org.eclipse.virgo.repository**  
**org.eclipse.virgo.repository.builder**  
**org.eclipse.virgo.repository.codec**  
**org.eclipse.virgo.repository.configuration**  
**org.eclipse.virgo.repository.management**  
**org.eclipse.virgo.repository.util**  
*is this necessary to export?* 

# content

- create repository chains, configurations and instance.
- main interface is **Repository**
- deals in **ArtifactDescriptors**, not only artifacts (TxNxV)

```
public interface Repository {  
  
    String getName();  
  
    Query createQuery(String key, String value);  
  
    Query createQuery(String key, String value, Map<String,  
Set<String>> properties);  
  
    RepositoryAwareArtifactDescriptor get(String type, String name,  
VersionRange versionRange);  
  
    void stop();  
  
}
```

```
public interface Query {  
  
    Query addFilter(String name, String value);  
  
    Query addFilter(String name, String value, Map<String,  
Set<String>> properties);  
  
    Set<RepositoryAwareArtifactDescriptor> run();  
  
}
```

# ...content

- management – for jmx interface;
- configuration – for encapsulating various types of repository config;
- codec – used by hosted-repository app;  
builder – used by kernel **Bridges**.

# util

- Utilities for file handling, directory watching, conversions, parsing

# content

org.eclipse.virgo.util.common  
org.eclipse.virgo.util.env  
org.eclipse.virgo.util.io  
org.eclipse.virgo.util.jmx  
org.eclipse.virgo.util.math  
org.eclipse.virgo.util.osgi  
org.eclipse.virgo.util.parser.manifest  
org.eclipse.virgo.util.test

it's fairly obvious  
what each of  
these is for



# medic

- log
- eventlog
- dump

log

# eventlog

dump

# test

- kernel framework - DmKernelTestRunner

```
@RunWith(DmKernelTestRunner.class)
public abstract class AbstractKernelIntegrationTest {
```

- on OsgiTestRunner
  - on JUnit test runner

```
@RunWith(OsgiTestRunner.class)
public class
BasicOsgiEnvironmentTests {
```

- Allows integration tests

# Documentation

the documents and how we (should) maintain them

# Built by documentation repo

- built just like any other repository:
  - `build-documentation/build.xml`  
(with `publish-documentation.xml`)
  - builds all three documents in pdf and html - when a release is built also publishes them to doc website
  - single source docbook markup

# Getting-Started

## {OpenGrok

xref: /virgo/org.eclipse.virgo.documentation/getting-started/src/preface.xml

Home | History | Annotate | Line # | Download |  Search  only in src

```
1 <?xml version="1.0"?>
2 <!DOCTYPE preface PUBLIC "-//OASIS//DTD DocBook XML V4.5//EN"
3     "http://www.oasis-open.org/docbook/xml/4.5/docbookx.dtd">
4
5 <preface>
6
7     <title>Preface</title>
8
9     <para>
10         This Guide introduces some basic concepts, explains how to in
11         and the associated Eclipse<trademark class="trade"/> tools, a
12         web application in detail, including how the application is b
13     </para>
14
15     <para>It is intended for Spring application programmers and assumes l
16         understanding of OSGi<trademark class="trade"/> or @virgo@ @w
17
18     <para>Questions about @webserv@ and @sts@ (or this Guide) may be post
19     (<ulink url="@webserv.community.forums.url@">@webserv.community.forum
20
21 </preface>
22
```



# content

- Maven-built sample application (**GreenPages**) in two forms - complete and skeleton.
- Instructions for using Eclipse and Virgo tools to build sample.
- Guidance about how to set up automatic unit and integration tests for the sample.

# User's Guide

{OpenGrok

xref: /virgo/org.eclipse.virgo.documentation/user-guide/src/virgo-user-guide.xml

Home | History | Annotate | Line # | Download |  Search  only in src

```
93
94     <toc />
95     <xi:include href="introduction.xml" />
96     <xi:include href="installing.xml" />
97     <xi:include href="installing-kernel.xml" />
98     <xi:include href="starting-stopping.xml" />
99     <xi:include href="kernel-user-region.xml" />
100    <xi:include href="admin-shell.xml" />
101    <xi:include href="admin-console.xml" />
102    <xi:include href="repository.xml" />
103    <xi:include href="serviceability.xml" />
104    <xi:include href="applications.xml" />
105    <xi:include href="configuring.xml" />
106    <xi:include href="log-codes.xml" />
107    <xi:include href="known-issues.xml" />
108    <xi:include href="further-reading.xml" />
109
110 </book>
111
```

# content

- **Guide to installing, configuring and using (e.g. error codes) the kernel and the server.**

# Programmer's Guide

...and I quote:

**In this Programmer Guide, we explore the runtime portions of the [@umbrella.product.name@](#), the [@product.name@](#) and the [@kernel.product.name@](#), and learn how to develop applications to benefit from their capabilities.**

# content

[xref: /virgo/org.eclipse.virgo.documentation/programmer-guide/src](#)

[Home](#) | [History](#) | [Annotate](#) |    only in /virgo/org

*Up to higher level directory*

Name	Date	Size
<a href="#">architecture.xml</a>	02-Nov-2010	24.7K
<a href="#">common-libraries.xml</a>	08-Sep-2010	4.2K
<a href="#">developing-applications.xml</a>	02-Nov-2010	54.7K
<a href="#">formtags-case-study.xml</a>	02-Nov-2010	33.7K
<a href="#">introduction.xml</a>	02-Nov-2010	10.8K
<a href="#">known-issues.xml</a>	08-Oct-2010	3.9K
<a href="#">migrating-to-osgi.xml</a>	08-Sep-2010	8.6K
<a href="#">preface.xml</a>	02-Nov-2010	1.7K
<a href="#">prerequisites.xml</a>	02-Nov-2010	1.2K
<a href="#">tooling.xml</a>	05-Nov-2010	13.7K
<a href="#">virgo-programmer-guide.xml</a>	02-Nov-2010	2K



# Images

- Images are explicitly sized and centered:

Name
<code>architecture-bundle-versioning.graffle</code>
<code>architecture-bundle-versioning.png</code>
<code>bundle-dependencies-layers.graffle</code>
<code>bundle-dependencies-layers.png</code>
<code>bundle-dependencies-verticals.graffle</code>
<code>bundle-dependencies-verticals.png</code>
<code>BundleDependencies.png</code>
<code>BundleDependencies.vsd</code>
<code>BundleDependenciesAlongLayers.png</code>
<code>BundleDependenciesAlongLayers.vsd</code>
<code>BundleDependenciesAlongVerticals.png</code>
<code>BundleDependenciesAlongVerticals.vsd</code>
<code>create-new-par-complexDomain-skeleton-bundles.png</code>
<code>create-new-par-complexDomain-web-page.png</code>
<code>create-new-par-dataSource-classNotFound-eclipse.png</code>
<code>create-new-par-infrastructure-add-infrastructure-to-par.png</code>
<code>create-new-par-infrastructure-created-eclipse.png</code>
<code>create-new-par-infrastructure-databasepopulator-before-manifest.png</code>

- either drawn (graffle) or snapshots — .png files.
- all sources in repository

# Process

- Every change potentially affects the documentation. Even 'pure' bugs should give rise to 'Release Notes'.
- Part of the review process for bugzilla patches/contributions is to provide documentation updates. (Git commits are not linked.)

# Build

the build jobs and how to maintain them

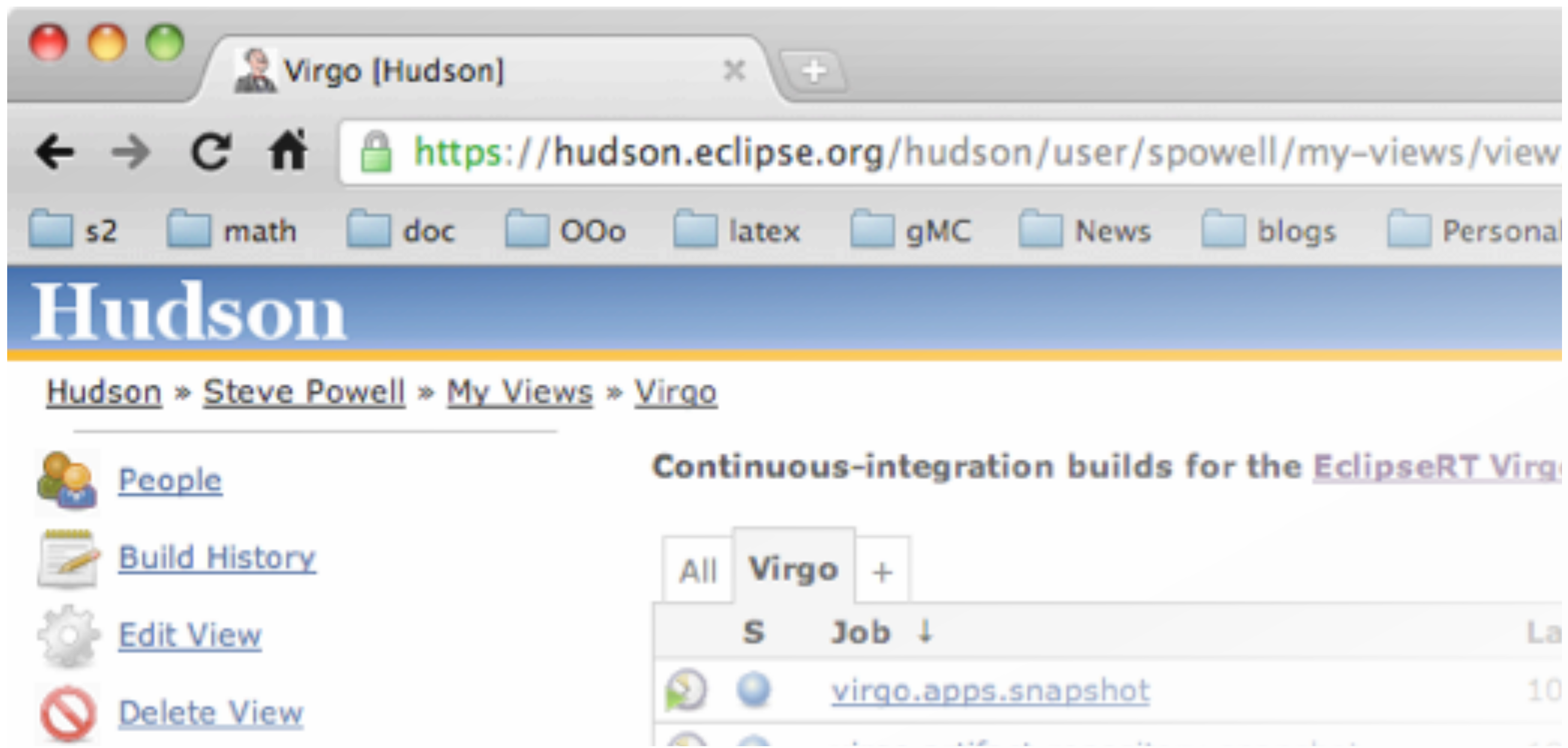


# Hudson

- where is it?
- the jobs
- how to create a new job
- the general scheme

# where is it?

- <http://hudson.eclipse.org>



The screenshot shows a web browser window with the following elements:

- Browser Tab:** Virgo [Hudson]
- Address Bar:** <https://hudson.eclipse.org/hudson/user/spowell/my-views/view>
- Navigation:** Back, Forward, Refresh, Home icons.
- Folder Bar:** s2, math, doc, OOo, latex, gMC, News, blogs, Personal.
- Page Header:** Hudson
- Breadcrumbs:** [Hudson](#) » [Steve Powell](#) » [My Views](#) » [Virgo](#)
- Left Sidebar:**
  - [People](#)
  - [Build History](#)
  - [Edit View](#)
  - [Delete View](#)
- Main Content:**
  - Continuous-integration builds for the [EclipseRT Virgo](#)**
  - Buttons: All, **Virgo**, +
  - Table with columns: S, Job ↓, La
  - Table Row 1: [virgo.apps.snapshot](#) 10

# jobs

virgo.	apps	
	artifact-repository	
	documentation	
	gemini-web-container	
	kernel	
	medic	
	osgi-extensions.snapshot	
	osgi-test-stubs.snapshot	
	test.snapshot	
	util.snapshot	
	web-server.snapshot	
	virgo.web.snapshot	
		.snapshot

# creating a new job

- raise a bugzilla: [bugs.eclipse.org/](http://bugs.eclipse.org/bugs)
- community/
  - hudson
  - give job name and owner userid on system
  - use existing Virgo job as template

# general info for a job

## **General settings:**

*Project name*

virgo.artifact-repository.snapshot

*Description*

Full build of **artifact-repository** repository in Virgo.

[X] *Discard Old Builds*

*Days to keep builds*

[5] *if not empty, build records are only kept up to this number of days*

*Max # of builds to keep*

[3] *if not empty, only up to this number of build records are kept*

*JDK*

[Java 6 R 21 64bit (SUN)] *JDK to be used for this project*

*Restrict where this project can be run*

[build2]

# get source

## Source Code Management

- None
- CVS
- Subversion
- Git

### Repositories

URL of repository

Name of repository (blank to create default)

Refspec (blank to create default)

### Branches to build

Branch Specifier (blank for default):

### Repository browser

# ...advanced settings

Merge options

Merge before build

Clean after checkout



Use commit author in changelog



Wipe out workspace



## **Build Triggers**

---

Build after other projects are built

Trigger builds remotely (e.g., from scripts)

Build periodically

Poll SCM

Schedule

```
#MIN HOUR DOM MONTH DOW
*/30 * * * 1-5
# (every 30 minutes, weekdays)
```

# Ant build step

## Build

---

### Invoke Ant

Ant Version

apache-ant-1.8.1

Targets

check-proxy clean clean-integration test javadoc-api javadoc-all package

Build File

build-repository/build.xml

Properties

```
ci.build=true  
build.stamp=CI-${BUILD_ID}  
eclipse.buildId=udsonbuild
```

Java Options



# Ant build step