

25 Alfresco things you might not know (but should!)

Nick Burch, Quanticate



Quanticate
The Clinical Data Experts

Who are you to pick?

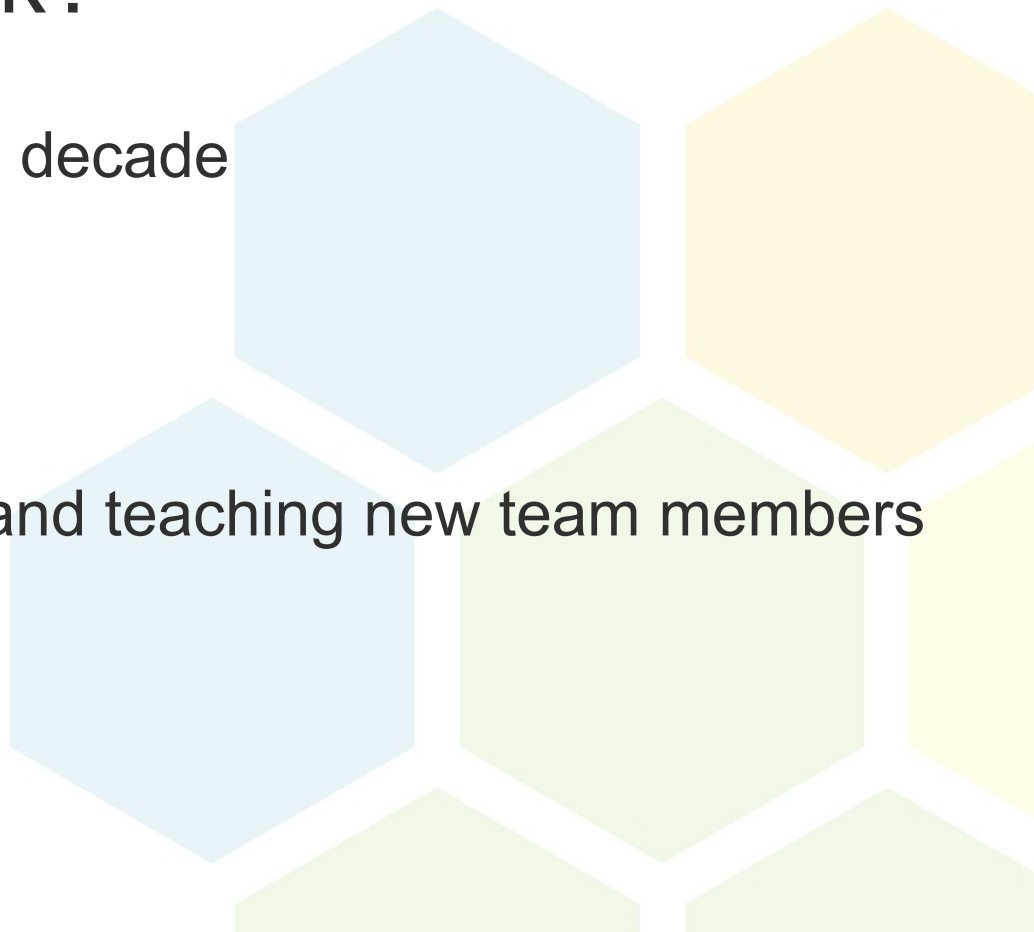
Apache committer for over a decade

Former Alfresco Engineer

CTO at Quanticate

Still coding, plus mentoring and teaching new team members

>25k rep on Stack Overflow



Where do these things come from?

Mistakes I've made!

Things new members of my team have struggled with

Neat tricks in open source + private Alfresco extensions

Helping new Alfresco developers on Stack Overflow

Seeing what others have been helping new developers with on the forums and Stack Overflow

25-ish Alfresco things you might not know (but should!)

Depending on how you count them.....



Source Code

Read the source, Luke

Alfresco is Open Source!

Community - <https://github.com/Alfresco>
Probably start with community-edition-old

Enterprise

<https://svn.alfresco.com/repos/alfresco-enterprise-mirror/>
Ask your account rep to set you up account details

Most extensions are too!

<https://github.com/OrderOfTheBee>

Links from <https://addons.alfresco.com/>

The background features a pattern of overlapping hexagons in light blue, light green, and light yellow. A large, green, stylized bracket shape is positioned on the left side, framing the text.

JavaScript Console

Admin + Testing from a browser

<https://github.com/share-extras/js-console>

Probably the add-on I use the most day-to-day!

Run saved scripts, write new ones on the fly

Full access to all repo root objects

Friendly interface with highlighting + tab complete

Lets you do lots of admin without redeploying / Java

Admin Tools

run with **var space** = /Company Home

Select ...

Javascript input

Freemarker input

Script execution parameters



```
1 // To create the next monthly folder in the EMT site, and setup
2 // the rules to apply to the child folders therein
3 var emtSite = siteService.getSite("emt");
4 if (emtSite == null) {
5     throw "No EMT site found!";
6 }
7 var doclib = emtSite.getContainer("documentLibrary");
8
9 // When are we running for? This month, or next?
10 var whenFor = new Date();
11 if (whenFor.getDate() >= 15) {
12     whenFor.setMonth(whenFor.getMonth()+1);
13 }
14 whenFor.setDate(1);
15 // Reports are always the month before
16 var repMonth = new Date(whenFor.getTime());
17 repMonth.setMonth(repMonth.getMonth()-1);
18 // Get the friendly date names
19 var nameFmt = new java.text.SimpleDateFormat("YYYY-MM");
20 var monthFmt = new java.text.SimpleDateFormat("MMMM");
21 var monthName = monthFmt.format(whenFor);
22 var repMonthName = monthFmt.format(repMonth);
```

Line 19 - Column 56 - Errors/Warnings 0

Execute

or Ctrl+Enter - executed in 7044ms (2017-4-24 22:22:9)

Console output

Freemarker HTML output

Freemarker text output

JSON output

Performance

Server infos

Dump Infos

```
DEBUG - Setting up folder for 2017 May
DEBUG - 2017 available as workspace://SpacesStore/bd615551-a382-429c-b574-fc5310c1e566
DEBUG - Creating folder for month 2017-05
DEBUG - Creating folder: Board Reports April 2017
DEBUG - Creating folder: Raw Board Reports
```



Alfresco Koans

Thanks to all contributors, especially Robin!

Bringing new developers up-to-speed

<https://github.com/rbramley/Alfresco-Koans>

Series of self-learning tasks for new Alfresco developers

Helps learn all the basics

Great introduction given by Robin last year!

http://beecon.buzz/2016/assets/data/files/20160125047/RBramley_AlfrescoKoans_BeeCon2016_LightningTalk.pdf

The background features a pattern of overlapping hexagons in light blue, light green, and light yellow. A thick green line starts on the left, forms a partial hexagon, and then continues as a horizontal line across the middle of the page.

Aikau Resources

How to learn the Aikau way

Aikau Intro

<http://docs.alfresco.com/5.2/concepts/aikau-intro.html>

Aikau Tutorial

<https://github.com/Alfresco/Aikau/blob/master/tutorial/chapters/About.md>

Developing Share extensions with Aikau from the Ground Up

<https://docs.google.com/document/d/1q25jA5EQ5PRYekr8tpM3ELlwOQ8Ht3Ng6D4VWsKoZtY/pub>

Aikau Alfresco blogs

<https://community.alfresco.com/community/ecm/blog/tags#/?tags=aikau>

The background features a cluster of hexagons in light blue, light green, and light yellow. A thick green line starts on the left, forms a partial hexagon, and then continues as a horizontal line that passes behind the text.

Node Locator

Finding Nodes

AncestorNodeLocator, CompanyHomeNodeLocator, DocLibNodeLocator, SelfNodeLocator, SharedHomeNodeLocator, SitesHomeNodeLocator, UserHomeNodeLocator, XPathNodeLocator

<http://dev.alfresco.com/resource/docs/java/org/alfresco/repo/nodelocator/NodeLocator.html>

WebScript: /alfresco/service/api/nodelocator/{node_locator_name}?params
/alfresco/service/api/{store_type}/{store_id}/{node_id}/nodelocator/{locator_name}

In Java, bit less user-friendly

<http://dev.alfresco.com/resource/docs/java/org/alfresco/repo/nodelocator/NodeLocatorService.html>

Finding Nodes - Examples

xpath?query=/app:company_home

companyhome

userhome

```
{  
  data: {  
    nodeRef: "workspace://SpacesStore/07e29d80-80c3-4e12-8b3a-e0f655505d63"  
  }  
}
```

siteshome

api/workspace/SpacesStore/12363b7a-79a5-4b0b-ad41-e34cd63ffffff/nodelocator/doclib



Node Cleaning

Making your nodes 50% brighter!

Why is my database so big?

This mostly applies to Alfresco 4.x , 5.0 and 5.1

alf_prop_link and alf_audit_entry can grow huge!

For the latter, either turn off Auditing, or purge older entries, using `api/audit/clear/alfresco-access?fromTime=0&toTime=`

For the former, read

<http://docs.alfresco.com.cn/5.1/concepts/prop-tables.html>

then enable `attributes.propcleaner.cronExpression`

What happens to deleted stuff?

Or why has my Alfresco disk run out of space?

When a user deletes something, unless it has the temporary aspect, it gets moved to the Archive Store

You can view these in the Trashcan (or in code in the Archive Store, eg via `sys:archivedOriginalParentAssoc`)

When purged from Archive, orphan content store cleaner moves it to `contentstore.deleted`, but by default no further!

<http://docs.alfresco.com/5.1/concepts/clean-content.html>

The background features a pattern of hexagons in light blue, light green, and light yellow. A large, green, stylized bracket shape is positioned on the left side, framing the text.

Python + cmislib

Quick integrations and scripted tasks

For pure-Alfresco admin, consider JavaScript console / Rules
For standard integrations, Public Rest APIs or CMIS

For CRUD tasks with simple/quick logic and external integrations, eg with FileSystem or other libraries, Python + cmislib can be a quick way to implement things

Groovy + OpenCMIS can work too, but some non-Groovy things in OpenCMIS make things a bit more verbose

Quick integrations and scripted tasks

Quick-wins I've had with it include:

- Sync folders+files locally with an Alfresco site
- Query DB, create stub nodes with custom metadata based on results
- Quick report of all PDFs in a site as XLS (could also be done with Spreadsheet webscript, but that's more work!)
- Python about 2/3rds size of Groovy, Groovy half Java



Custom NodeRefs

Vanity plates for your new Nodes!

Children (21)

Name	Child Reference	Primary	Association Type	Index
app:space_templates	workspace://SpacesStore/24d84	true	cm:contains	0
app:content_templates	workspace://SpacesStore/d1a5	true	cm:contains	1
app:email_templates	workspace://SpacesStore/c3dfd	true	cm:contains	2
app:rss_templates	workspace://SpacesStore/00f04	true	cm:contains	3
app:saved_searches	workspace://SpacesStore/c6d78	true	cm:contains	4
app:scripts	workspace://SpacesStore/aa203	true	cm:contains	5
app:node_templates	workspace://SpacesStore/5c4f0	true	cm:contains	6
cm:webscripts	workspace://SpacesStore/d6506	true	cm:contains	7
cm:extensionwebscripts	workspace://SpacesStore/755bl	true	cm:contains	8
app:models	workspace://SpacesStore/a9a9f	true	cm:contains	9
app:messages	workspace://SpacesStore/aced7	true	cm:contains	10
app:webclient_extension	workspace://SpacesStore/dd71	true	cm:contains	11
app:workflow_defs	workspace://SpacesStore/05590	true	cm:contains	12
app:imap_configs	workspace://SpacesStore/bf47c	true	cm:contains	13
app:transfers	workspace://SpacesStore/496df	true	cm:contains	14
app:rendering_actions	workspace://SpacesStore/rendering_actions_space	true	cm:contains	15
app:replication_actions	workspace://SpacesStore/replication_actions_space	true	cm:contains	16
cm:Scheduled Actions	workspace://SpacesStore/888291	true	cm:contains	17
app:publishing_root	workspace://SpacesStore/publishing_root_space	true	cm:contains	18
cm:Logos	workspace://SpacesStore/0b46e	true	cm:contains	19
srft:facets	workspace://SpacesStore/solr_facets_root_space	true	cm:contains	20

But how to get those nice noderefs?

Sadly, not possible via the normal NodeService

Not even possible via any other friendly services

You need an ACP! Either bootstrap or manual import

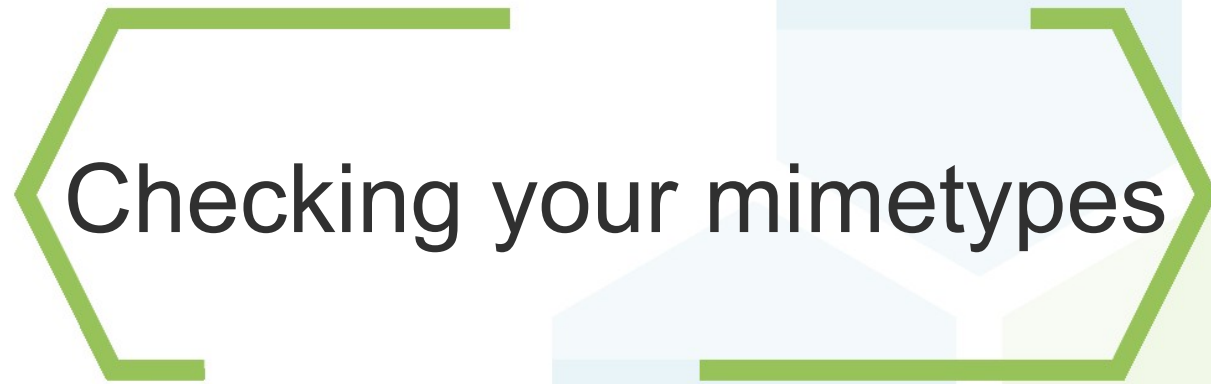
```
<pub:Environment view:childName="${spaces.publishing.root.childname}">
  <view:properties>
    <sys:store-protocol>workspace</sys:store-protocol>
    <sys:store-identifier>SpacesStore</sys:store-identifier>
    <sys:node-uuid>publishing_root_space</sys:node-uuid>
    <cm:name>${spaces.publishing.root.name}</cm:name>
    <cm:title>${spaces.publishing.root.name}</cm:title>
    <cm:description>${spaces.publishing.root.description}</cm:descrip
  </view:properties>
</pub:Environment>
```

The background features a cluster of hexagons in light blue, light green, and light yellow. A thick green line starts on the left, forms a partial hexagon, and then continues as a horizontal line that passes behind the word 'Debugging'.

Debugging

Why didn't it work as expected?

- Remote Debugger – jpda – attach breakpoints from IDE
- Rhino JS Debugger – step through server-side JavaScript (eg WebScripts or Rules), fiddly/hard on production
- Log4j – Java, JavaScript, Activiti, Alfresco itself
alfresco/extension/dev-log4j.properties
- Share - Surf and Aikau debugging
- Un-minimised JS resources + in-browser dev tools



Checking your mimetypes



What does Alfresco know about?

To see what mimetypes are defined, including extensions and the description shown in Share, as JSON:

`/alfresco/service/api/mimetypes/descriptions`

To see the mimetypes in a list, use:

`/alfresco/service/mimetypes`

To add new custom ones, follow instructions from `extension/mimetype/mimetypes-extension-map.xml.sample`

The background features a pattern of hexagons in light blue, light green, and light yellow. A thick green line forms a large, irregular hexagonal shape that frames the text. The text "And your transformers" is centered within this green outline.

And your transformers

What can turn into what?

Start in your browser at: [/alfresco/service/mimetypes](#)


Click to view details of one – only currently active shown:

audio/mp4 - m4a

Extractors: org.alfresco.repo.content.metadata.TikaAudioMetadataExtractor
Transformable To:

- audio/mpeg = Proxy via: org.alfresco.repo.content.transform.RuntimeE with gcc 5.4.0 (Ubuntu 5.4.0-6ubuntu1~16.04.4) 20160609 configurat --incdir=/usr/include/x86_64-linux-gnu --cc=cc --cxx=g++ --enable-gp avisynth --enable-gnutls --enable-ladspa --enable-libass --enable-libblur libgme --enable-libgsm --enable-libmodplug --enable-libmp3lame --enab libsoxr --enable-libspeex --enable-libssh --enable-libtheora --enable-libt openal --enable-opengl --enable-x11grab --enable-libdc1394 --enable-lib 60.100 libavformat 56. 40.101 / 56. 40.101 libavdevice 56. 4.100 / 56. 4 libpostproc 53. 3.100 / 53. 3.100)

Transformable From: *Cannot be generated from anything else*

The background features a pattern of hexagons in light blue, light green, and light yellow. A thick green line forms a large, irregular hexagonal shape that frames the text. The text is centered within this green frame.

New previews &
transformers

For new/custom formats

First, check Share Extras and Alfresco Add-Ons, your work may already have been done by someone else!

Next, consider pure-JS previewers – often delivers better User Experience, may be easier to develop / configure

If using Command Line tools, base on `RuntimeExecutableContentTransformerWorker`, copy eg `ffmpeg`

Check registration using `mimetype` details webscript

Consider Composite Transformations, via several mimetypes



Eventual Consistency

But I'm sure I had that node a moment ago...

Fun with SOLR...

Most services in Alfresco are now “Eventually Consistent”, so generally you can’t add/change then search/see the result immediately. It’s often there almost straight after, not always!

<http://docs.alfresco.com/5.2/concepts/solr-event-consistency.html>

There are some methods which query the DB directly, and so don’t wait on the lag for SOLR to notice + fetch + index

“Canned Query” powered ones include

`NodeService.getChildAssocs` and `NodeService.getChildByName`



JS & FTL Root Objects

Handy objects and helpers

Biggest thing to understand – you get different root objects when running in Share and on the Repository

Second biggest – FTL and JavaScript have different root objects, and some different behaviour on wrappers

Repo JS: <http://docs.alfresco.com/5.2/references/API-JS-rootscoped.html>

Repo FTL: <http://docs.alfresco.com/5.2/references/API-FreeMarker-defaultmodel.html>

Share JS: <http://docs.alfresco.com/5.2/references/APISurf-rootscoped.html>

Services – Built-in and Your Own

On Repo side, most things you'll want to do can be done in JS using the root service objects, eg actions / search / people (On Share / Surf side, you'll mostly be using REST APIs)

Repo helpers almost all work with wrapped objects and JavaScript-like data types, eg ScriptNode

You can add your own helpers / services, just extend `BaseScopableProcessorExtension` and give bean an `extensionName` . Try to return `Scriptable` or wrapper objects

`ServiceRegistry` hard to use, security implications if exposed!

The background features a cluster of hexagons in light blue, light green, and light yellow. A large, green, stylized bracket shape is positioned on the left side, framing the text.

Activiti Scripts & Delegates

Custom logic in your Workflow

Two main ways to inject your own logic for validation / calling other systems / copying variables / performing tasks

Java – Extends `org.alfresco.repo.workflow.activiti.BaseJavaDelegate` , registered in Spring, called from `activiti:delegateExpression`

JavaScript – In your workflow in `<activiti:field name="script">` in start or complete event via `<activiti:taskListener event="complete" class="org.alfresco.repo.workflow.activiti.tasklistener.ScriptTaskListener">`

Java has more services, better errors, but not versioned with WF



Share custom evaluators

What shows as Share Actions or Icons

docs.alfresco.com/5.2/concepts/dev-extensions-share-evaluators.html

docs.alfresco.com/5.2/concepts/doclib-predefined-evaluators-reference.html

Lots of built-in ones, eg locked, rules, checkedOutAspect

Easy to configure up other ones as spring beans + config, eg by Site Role, Node Type, Mime Type

Writing your own custom ones in Java isn't too bad – extend BaseEvaluator, check node details JSON, review properties or aspects. Try to avoid additional REST fetches if possible!

```
<bean id="evaluator.doclib.action.isMimeTypePDF"
      parent="evaluator.doclib.action.isMimeType">
  <property name="mimetypes">
    <list>
      <value>application/pdf</value>
    </list>
  </property>
</bean>
```

```
<bean id="evaluator.doclib.action.isSiteContributor"
      parent="evaluator.doclib.action.siteRole">
  <property name="requiredRole" value="SiteContributor" />
</bean>
```

```
<bean id="evaluator.doclib.action.isSiteConsumer"
      parent="evaluator.doclib.action.siteRole">
  <property name="requiredRole" value="SiteConsumer" />
</bean>
```

```
<bean id="evaluator.doclib.action.isQuanticateStudyDocument"
      parent="evaluator.doclib.action.nodeType">
  <property name="types">
    <list>
      <value>quanticate:studyDocument</value>
    </list>
  </property>
</bean>
```



RunAs / Retrying Txn

More advanced Java coding

Pattern you'll see a lot in the Alfresco codebase, if you're learning from that (as you should!), may be needed for you

RunAs – allows you to run a block of Java code as another user, eg as an Admin or as the System Account

Retrying Transaction – has work done in a transaction, which will retry if another process touched the same nodes

Both have pattern where you pass in a special block of code to do the work, single return value, not as complex as looks!



Getting URLs of things

For external links, emails etc

In Java – SysAdminParams

dev.alfresco.com/resource/docs/java/org/alfresco/repo/admin/SysAdminParams.html

In Java, more helpfull - UrlUtil

dev.alfresco.com/resource/docs/java/org/alfresco/util/UrlUtil.html

WebScript or Java (hacky) – SiteShareViewUrlGet

[/api/sites/shareUrl?nodeRef={nodeRef}](http://api/sites/shareUrl?nodeRef={nodeRef})

```
{  
  site: "tlftest",  
  url: "http://grenache:8081/share/page/site/tlftest/documentlibrary?path=/Metric Reports"  
}
```

The background features a cluster of hexagons in light blue, light green, and light yellow. A thick green line forms a large, irregular hexagonal shape that frames the text. The text "Rules & Behaviours" is centered within this green frame.

Rules & Behaviours

Similar idea, coming from different ends

Behaviours – Java code that's bound to certain global events, eg “OnUpdate for node of type/subtype cm:content”

dev.alfresco.com/resource/docs/java/org/alfresco/repo/node/NodeServicePolicies.htm

Not everything that should have a policy has one :(

Allows for unit-testable code to implement custom logic when things happen, eg validation, notification, sync, specialisation

Generally needs to consider transactions, may need to disable other behaviours while running

Similar idea, coming from different ends

Rules – Configured using GUI, based on a set of conditions, can trigger common Actions, or custom JS Script for harder stuff

Can be re-used, generally only tested manually

Allows power users and non-coders to set things up

With Scripts, can do pretty much anything you might want to

Rules are great for power-users and prototyping, don't always jump to behaviours to start, but consider when heavily re-used

Define Rule

When:

Items are created or enter this folder ▼

☒ If all criteria are met:

All Items ▼

All Items

Size

Created Date

Modified Date

Creator

Modifier

Author

Mimetype

Encoding

Description

Name

Title

Has tag

Has category

Content of type or sub-type

Has aspect

Show more...

☐ Run rule in background

If errors occur run script:

Select...

Create

Create and Create Another

Cancel

Select...

Execute script

Copy

Move

Check in

Check out

Link to category

Add aspect

Remove aspect

Add simple workflow

Send email

☒ Transform and copy content

Transform and copy image

Extract common metadata fields

Import

Specialise type

Increment Counter

Set property value

Review, Approve then Create PDF

Select...

The background features a cluster of hexagons in light blue, light green, and light yellow. A thick green line forms a large, irregular hexagonal outline that frames the central text.

Rules & Workflows

For starting a workflow by Rules

Define a rule on your folder, eg Content Created

Perform Action – Execute Script

Upload a script to Data Dictionary / Scripts, eg

```
var workflowAction = workflow.getDefinitionByName('activiti$WFID');  
var package= workflow.createPackage();  
package.addNode(document);  
var wfparams = new Array();  
wfparams["model_prefix:req_props_name"] = value2;  
wfparams["bpm:assignee"] = people.getPerson("ADMIN");  
workflowAction.startWorkflow(package, wfparams);
```



Raw DB / Content Store

When querying the database directly...

Generally, STOP! Don't go there!

You almost never should be talking to the Alfresco DB directly, save for doing a full backup/restore

If you need to backup/restore, ensure you also backup the whole of the content store in the Alfresco home folder!

Just DB = metadata only, no content. Just store = all your content but no idea what it is...

Use Alfresco service to search / update / store / etc, not raw!



JMX

And performance

Community and Enterprise

Firstly, change your passwords!

See <http://docs.alfresco.com/5.2/tasks/jmx-access.html> for how to configure at the Alfresco level

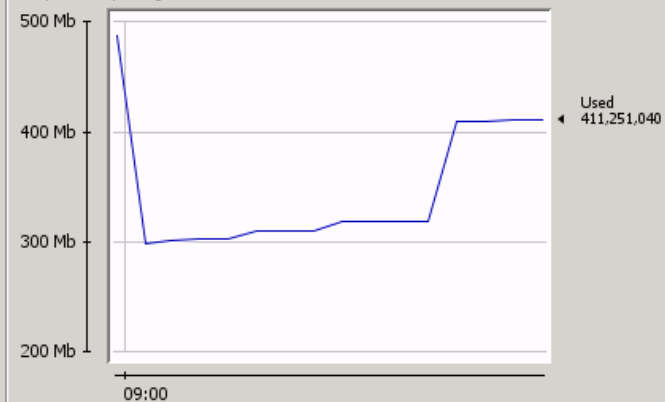
Can also do at Tomcat level

```
<!-- Give JXM RMI static, non-clashing ports -->  
<!-- This requires catalina-jmx-remote.jar and some -D settings on start -->  
<Listener className="org.apache.catalina.mbeans.JmxRemoteLifecycleListener"  
          rmiRegistryPortPlatform="9090" rmiServerPortPlatform="9095" />
```

See what the JVM is doing, why, and fix/change some stuff!

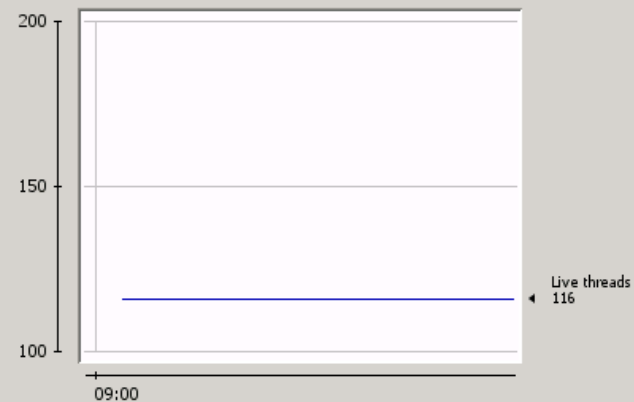
Time Range: All

Heap Memory Usage



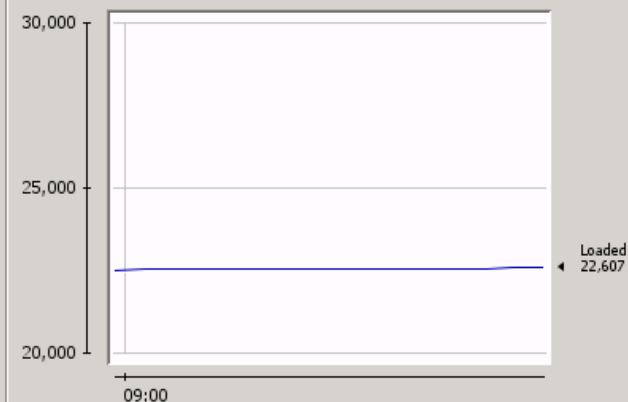
Used: 411.3 Mb Committed: 1.3 Gb Max: 1.9 Gb

Threads



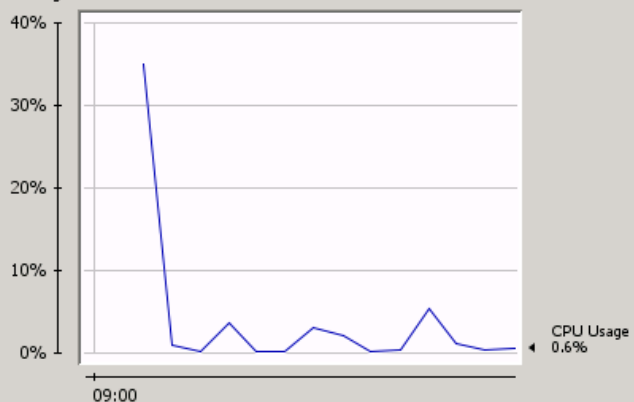
Live: 116 Peak: 118 Total: 1,149

Classes



Loaded: 22,607 Unloaded: 166 Total: 22,773

CPU Usage

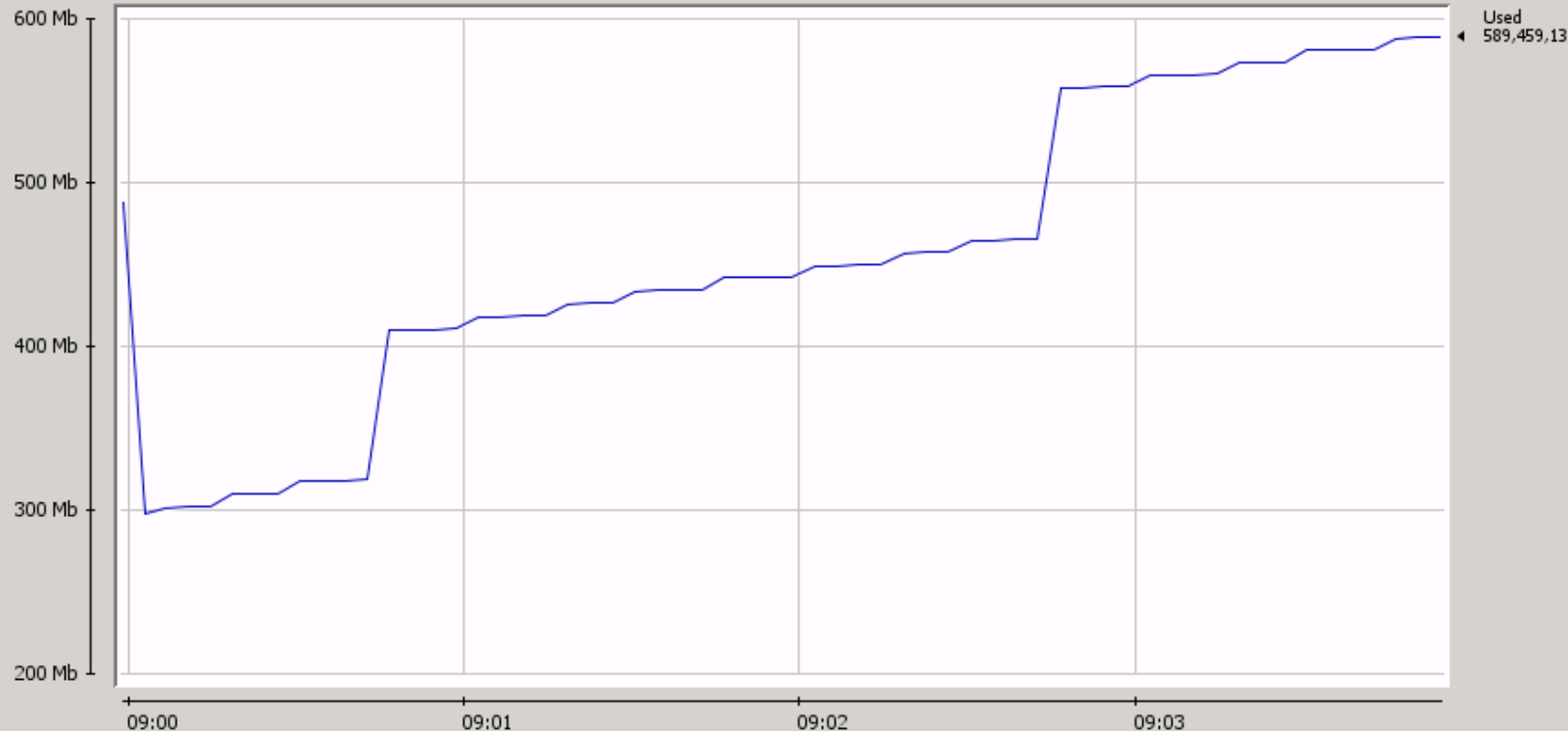


CPU Usage: 0.6%

Chart: Heap Memory Usage

Time Range: All

Perform GC



Details

Time: 2017-04-28 09:03:54

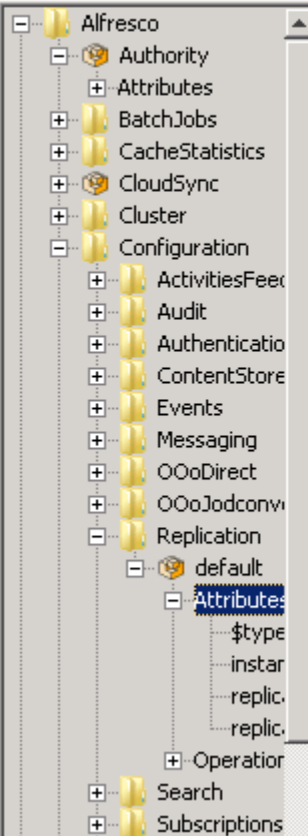
Used: 575,383 kbytes

Committed: 1,307,648 kbytes

100% ..

75% ..





Attribute values

Name	Value
\$type	default
instancePath	[default]
replication.enabled	false
replication.transfer.readonly	true

The background features a pattern of hexagons in light blue, light green, and light yellow. A large, green, open hexagonal outline is centered on the left side, framing the title text.

Naming & Clashes

Would a rose by any other name.... ?

Java – Use your own package name, very few reasons to put things under `org.alfresco` or `org.activiti`

Config / WebScripts – Use `alfresco/module` or `alfresco/extension`

Modules – ID will be used to build module install info directory, AMP tool has issues, ensure no clashes! Directory should use `_` not `.` so that context file references work properly

Keep your stuff in source control, redeploy, don't edit live!

The background features a pattern of hexagons in light blue, light green, and light yellow. A thick green line is drawn across the middle, forming a large, irregular shape that frames the text.

Node Browser

So that's what I've got!

Low level checking of all your nodes

Depending on your version, may be available in Share, or the Alfresco Repo admin console, or both!

`/alfresco/s/enterprise/admin/admin-nodebrowser`
`/share/page/console/admin-console/node-browser`

Lets you check all properties, type + aspects, all children (including normally hidden ones), names + qnames, secondary associations, parents, permissions etc

Invaluable for checking what you really have, and why!

Node Information

Reference workspace://SpacesStore/rendering_actions_space

Primary Path /app:company_home/app:dictionary/app:rendering_actions

Type cm:folder

Parent [workspace://SpacesStore/6a27093e-58ef-4364-bbcc-e38d44b09f95](#)

Aspects (4)

cm:auditable

sys:referenceable

cm:titled

Properties (12)

Name	Type	Value	Residual
cm:name	d:text	Rendering Actions Space	false
sys:node-dbid	d:long	492	false
sys:store-identifier	d:text	SpacesStore	false
sys:locale	d:locale	en_GB	false
cm:title	d:mltext	Rendering Actions Space	false
cm:modified	d:datetime	26 Sep 2015 16:31:35 GMT+0100 (BST)	false
sys:node-uuid	d:text	rendering_actions_space	false
cm:created	d:datetime	14 Aug 2015 22:01:55 GMT+0100 (BST)	false
sys:store-protocol	d:text	workspace	false
cm:creator	d:text	System	false
cm:description	d:mltext	A space used by the system to persist rendering actions.	false
cm:modifier	d:text	System	false

Parents (1)

Name	Parent Type
app:rendering_actions	cm:folder

Associations (0)

Type	Target	Reference	Association
------	--------	-----------	-------------

Source Associations (0)

Type	Source	Reference	Association
------	--------	-----------	-------------

Permissions

Inherits: true

Owner: System

Permission	Authority	Access
Consumer	GROUP_EVERYONE	ALLOW

The background features a light gray grid of hexagons. Some hexagons are filled with light blue, light green, or light yellow. A thick green line starts on the left, forms a partial hexagon, then extends horizontally to the right, passing behind the text. It then turns down and to the right, forming another partial hexagon shape.

Unit Testing

Just move to the SDK version 3!

You really should be unit testing your code

Unit test your services, integration test your WebScripts, use Selenium or similar to test the full webapps

Upgrade to SDK v3 – unit testing of Services and WebScripts is a lot nicer and a lot quicker there!

Check the Alfresco source tree for examples of the other kinds of unit tests, but be aware that some do things older ways...

Wanted – Good example unit tests!

Maybe I've missed this, but it'd be good to pull together a list of example unit/integration tests of all types, for new developers to learn from

SDK 3 has some, but perhaps not complex enough to be a good example for each layer?

Aikau has great blog post at

<https://community.alfresco.com/community/ecm/blog/2014/02/26/unit-testing-aikau>
but going from that to working examples hard for newbies!

Others I missed?

The background features a cluster of hexagons in light blue, light green, and light yellow. A thick green line forms a large, irregular hexagonal shape that frames the text.

Benchmarking Tools

Not just for benchmarking

Can be used to load test your system

Can be used to test the effect of performance parameters, eg “If I increase the memory but also add 5 more threads, what happens?”

Good examples for how to use the Share-PO Selenium wrappers for automated UI testing, including setup and teardown

Can use Alfresco provided tests, fairly easy to write your own



Nick Burch
@Gagravarr

orderofthebee.org
stackoverflow.com/questions/tagged/alfresco