AEM – Developer Friendly Tools

Listing of AEM tools that makes developers life easier

Ashokkumar T A | 10-Oct-2018

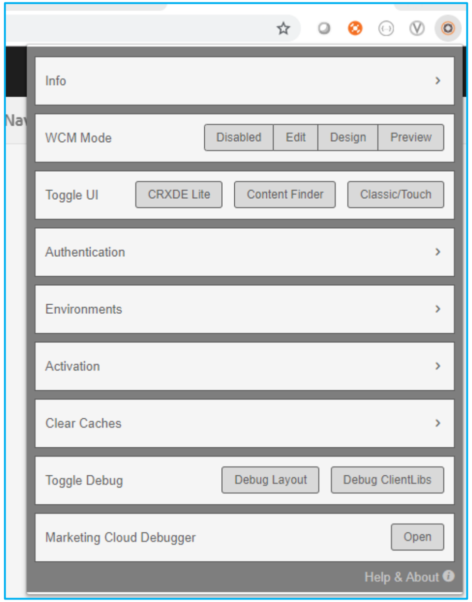
Browser Plugins

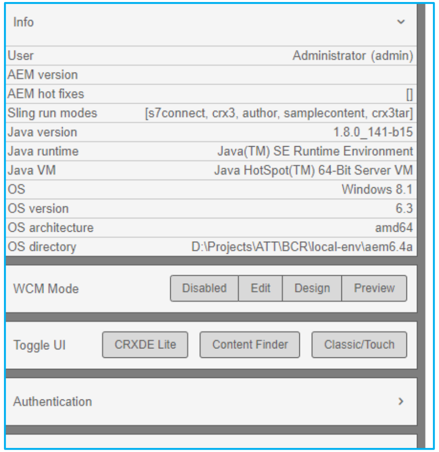
Chrome extension for AEM developers

This is a collection of tools and links to help the daily AEM developer work a little faster.

Features:

- Shows the user, AEM, Sling, Java & OS details of the instance
- Helps to toggle between edit, design & preview WCM modes or disable WCM mode
- Helps to toggle between Touch, Classic & CRXDE view
- Provides single click activate / deactivate functionality
- Has links to delete the client libs and compiled jsp caches




Info

User	Administrator (admin)
AEM version	
AEM hot fixes	[]
Sling run modes	[s7connect, crx3, author, samplecontent, crx3tar]
Java version	1.8.0_141-b15
Java runtime	Java(TM) SE Runtime Environment
Java VM	Java HotSpot(TM) 64-Bit Server VM
OS	Windows 8.1
OS version	6.3
OS architecture	amd64
OS directory	D:\Projects\ATT\BCR\local-env\aem6.4a

More details at

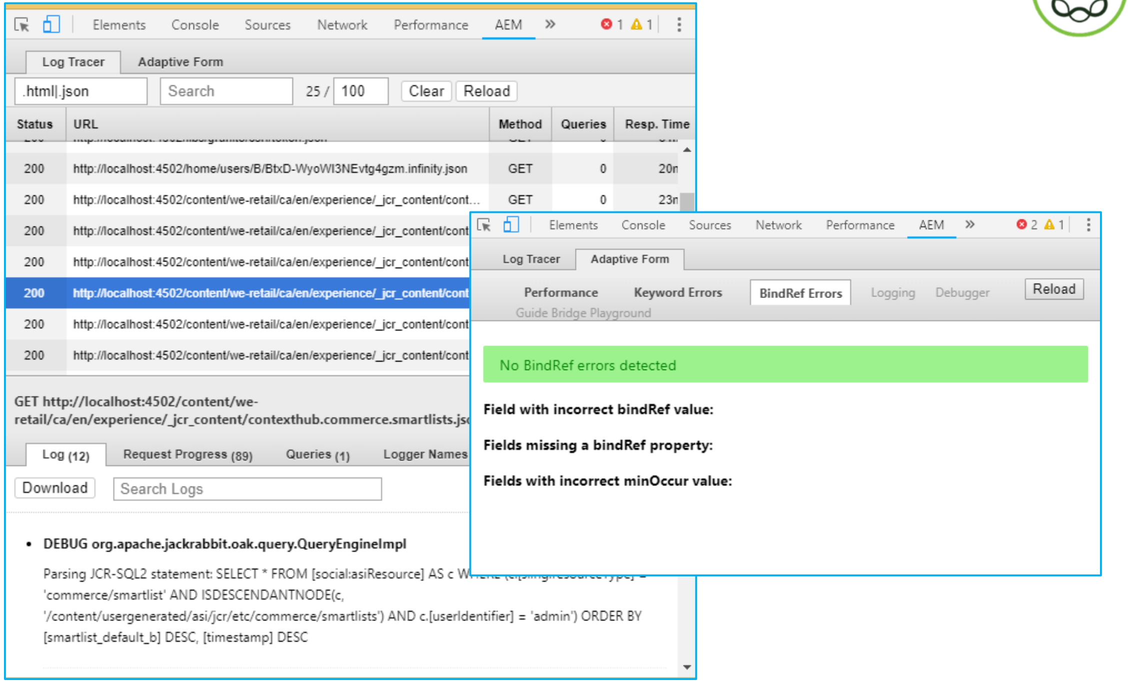
https://chrome.google.com/webstore/detail/aem-developer/hgjhcngmldfpgpakbnffnbnmcmohfmfc

AEM Chrome Plug-in

Provides a devtools panel for AEM development. This plugin includes an in-browser Log Tracer and also has Adaptive Forms development tooling

Features:

- Log Tracer surfaces per-request server side data quickly and conveniently into your Chrome dev tools panel
- Adaptive Forms: a robust suite of AEM Adaptive Form developer tools!



Status	URL	Method	Queries	Resp. Time
200	http://localhost:4502/home/users/B/BtxD-WyoWI3NEvtg4gzm.infinity.json	GET	0	20n
200	http://localhost:4502/content/we-retail/ca/en/experience/_jcr_content/cont...	GET	0	23n
200	http://localhost:4502/content/we-retail/ca/en/experience/_jcr_content/cont			
200	http://localhost:4502/content/we-retail/ca/en/experience/_jcr_content/cont			
200	http://localhost:4502/content/we-retail/ca/en/experience/_jcr_content/cont			
200	http://localhost:4502/content/we-retail/ca/en/experience/_jcr_content/cont			
200	http://localhost:4502/content/we-retail/ca/en/experience/_jcr_content/cont			

GET http://localhost:4502/content/we-retail/ca/en/experience/_jcr_content/contexthub.commerce.smartlists.js

```
• DEBUG org.apache.jackrabbit.oak.query.QueryEngineImpl
  Parsing JCR-SQL2 statement: SELECT * FROM [social:asiResource] AS c W
  'commerce/smartlist' AND ISDESCENDANTNODE(c,
  '/content/usergenerated/asi/jcr/etc/commerce/smartlists') AND c.[userIdentifier] = 'admin') ORDER BY
  [smartlist_default_b] DESC, [timestamp] DESC
```

No BindRef errors detected

Field with incorrect bindRef value:

Fields missing a bindRef property:

Fields with incorrect minOccur value:

More details at

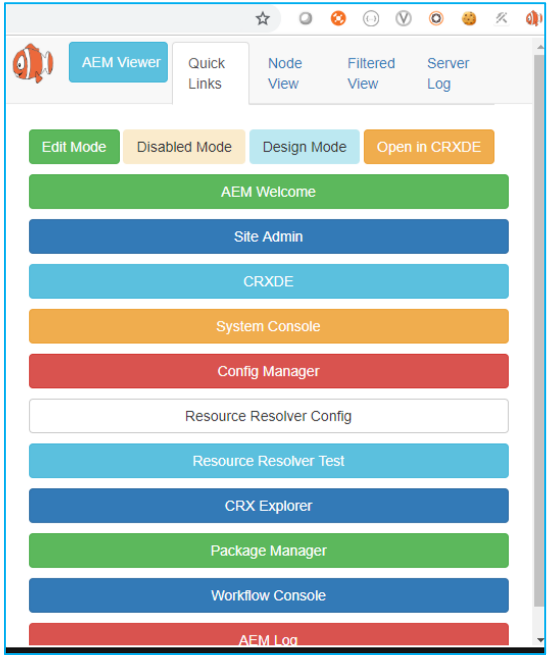
https://chrome.google.com/webstore/detail/aem-chrome-plug-in/ejdcnikffjleeffpigekhccpepplaode

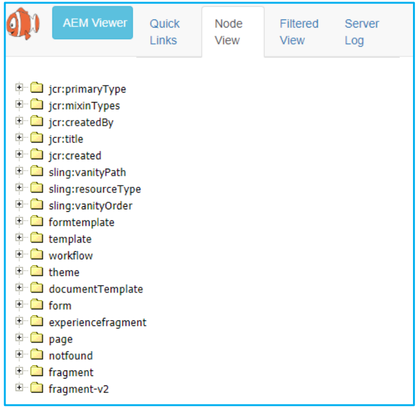
AEM Viewer

An extension which helps developers and testers to quickly check information about a page which is hosted in Adobe AEM/Day CQ.

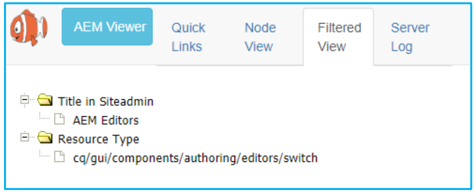
Features:

- The result has been split into two views , first tab shows the RAW data and the second tab shows the Skimmed Data .
- Intention is to show data in a non geeky way in the "Skimmed Data" tab and dumping the raw data in the "RAW Data" tab.





jcr:primaryType
jcr:mixinTypes
jcr:createdBy
jcr:title
jcr:created
sling:vanityPath
sling:resourceType
sling:vanityOrder
formtemplate
template
workflow
theme
documentTemplate
form
experiencefragment
page
notfound
fragment
fragment-v2



Title in Siteadmin
AEM Editors
Resource Type
cq/gui/components/authoring/editors/switch

More details at

https://chrome.google.com/webstore/detail/aem-viewer/hjacgomkpcmhaobhmgijlfpakaocchmm

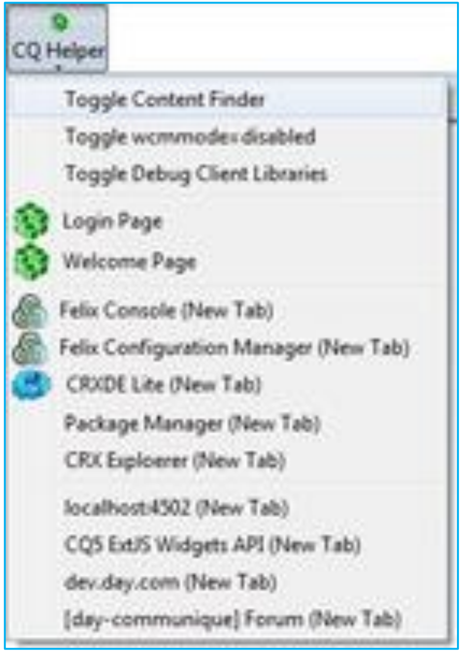
`

CQ Helper

With this tool, you can say "goodbye" to the days of cluttered bookmark menus full of links to individual CQ tools.

Features:

- It allows you to make a single bookmark to each of the CQ environments you regularly work in, then use the menu to jump between different tools provided by CQ.
- For example, if you're currently viewing CRXDE Lite in your browser, you can open the menu and click Felix Console to quickly open a new tab with Felix console on the same server.
- Additionally, it provides common shortcuts used when browsing pages in author mode.



More details at

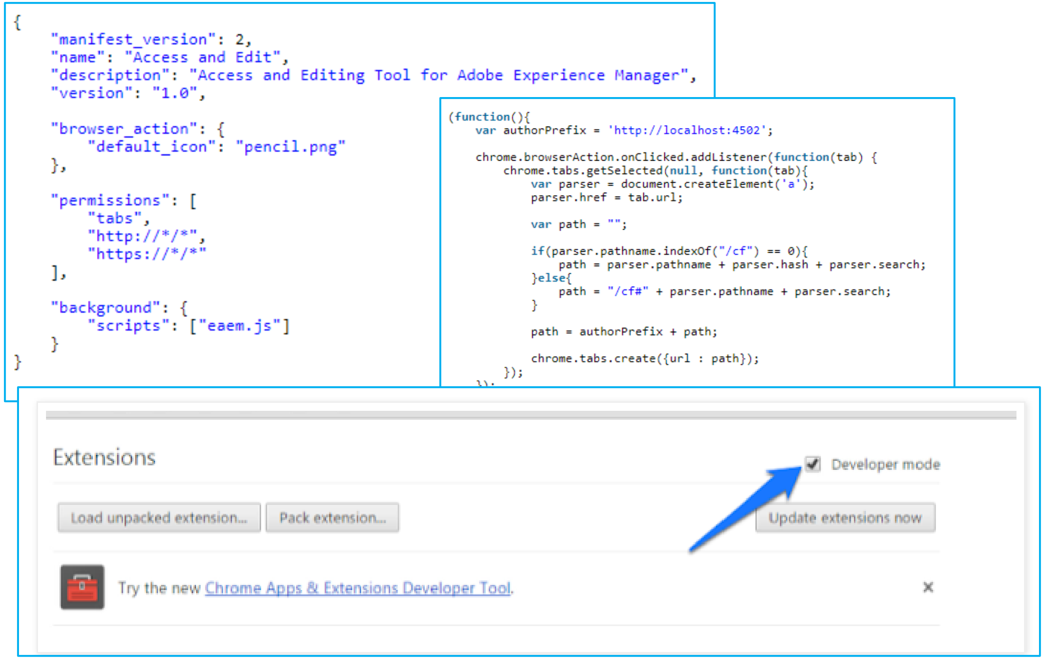
https://addons.mozilla.org/en-US/firefox/addon/cq-helper

How to develop Browser Extensions for AEM

This blog post gives detailed instructions on how to develop an AEM browser extension for chrome and firefox

Key details:

- Has the sample code for developing a simple browser extension for opening a published page from Author instance
- Covers all the steps from creating the code, extension icon to downloading the created plugin as zip



```
{
    "manifest_version": 2,
    "name": "Access and Edit",
    "description": "Access and Editing Tool for Adobe Experience Manager",
    "version": "1.0",

    "browser_action": {
        "default_icon": "pencil.png"
    },

    "permissions": [
        "tabs",
        "http://*/*",
        "https://*/*"
    ],

    "background": {
        "scripts": ["eaem.js"]
    }
}
```

```
(function(){
    var authorPrefix = 'http://localhost:4502';

    chrome.browserAction.onClicked.addListener(function(tab) {
        chrome.tabs.getSelected(null, function(tab){
            var parser = document.createElement('a');
            parser.href = tab.url;

            var path = "";

            if(parser.pathname.indexOf("/cf") == 0){
                path = parser.pathname + parser.hash + parser.search;
            }else{
                path = "/cf#" + parser.pathname + parser.search;
            }

            path = authorPrefix + path;

            chrome.tabs.create({url : path});
        });
    });
```

Extensions
☑ Developer mode
Load unpacked extension...
Pack extension...
Update extensions now
Try the new Chrome Apps & Extensions Developer Tool.
×

More details at

http://experience-aem.blogspot.com/2015/01/aem-browser-extension-to-open-publish-page-in-author.html

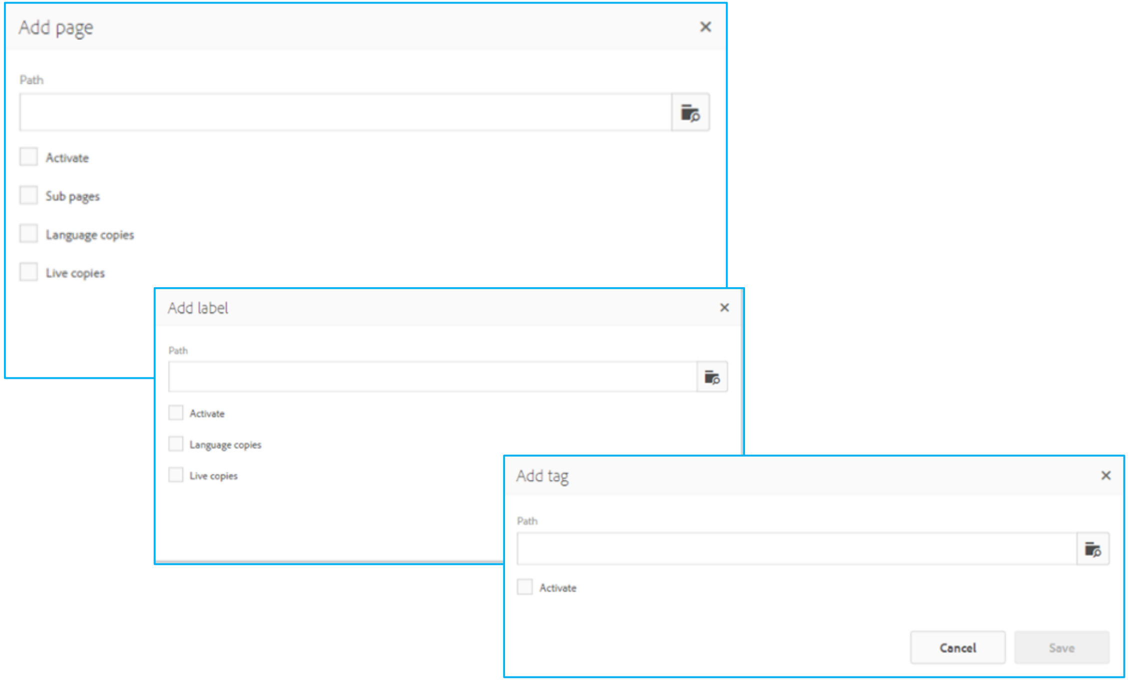
Content Migration

AMPLEXOR’s Content Package Tool for AEM

A Content Package Tool (CPT), it is ideal for multilingual content management.

Advantages of this tool:

- With its familiar user interface built in the updated AEM Touch-Enabled UI, this tool enables any content author to create content packages in a fast and easy-to-use way, without requiring any technical skills
- It offers a simple three step process to create content package



Add page ×

Path

☐ Activate

☐ Sub pages

☐ Language copies

☐ Live copies

Add label ×

Path

☐ Activate

☐ Language copies

☐ Live copies

Add tag ×

Path

☐ Activate

More details at

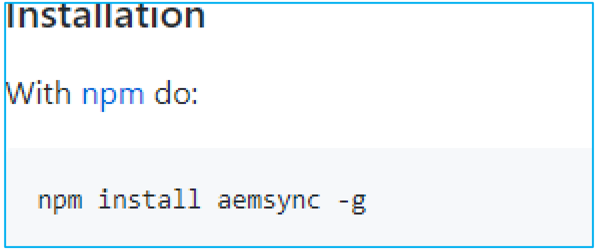
https://blog.amplexor.com/digitalexperience/en/tips-and-tools-to-import-and-export-content-in-aem

aemsync

An AEM (Adobe CQ) Synchronization Tool. This tool pushes code changes to AEM instance(s) upon a file change.

Key Features are

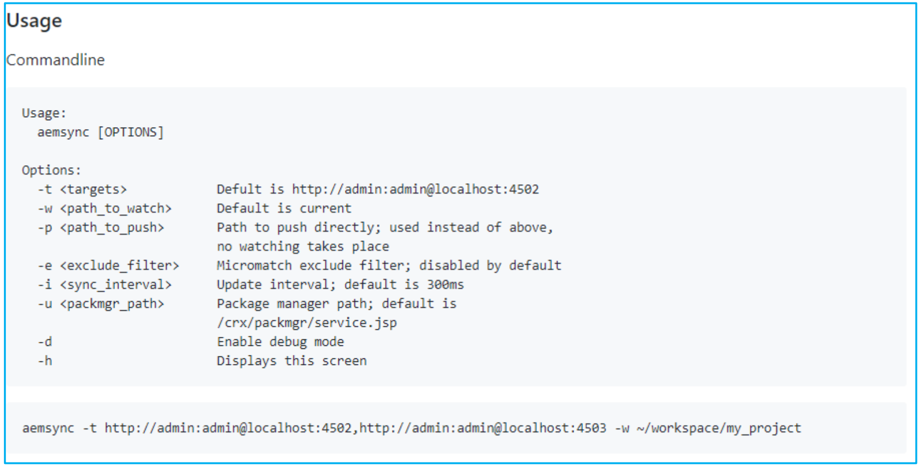
- There is no vault dependency.
- It can push to multiple instances at the same time (e.g. author and publish).
- IDE/editor agnostic.
- Works on Windows, Linux and Mac.



Installation

With npm do:

```
npm install aemsync -g
```



Usage

Commandline

```
Usage:
  aemsync [OPTIONS]

Options:
  -t <targets>           Defult is http://admin:admin@localhost:4502
  -w <path_to_watch>     Default is current
  -p <path_to_push>      Path to push directly; used instead of above,
                         no watching takes place
  -e <exclude_filter>    Micromatch exclude filter; disabled by default
  -i <sync_interval>     Update interval; default is 300ms
  -u <packmgr_path>      Package manager path; default is
                         /crx/packmgr/service.jsp
  -d                     Enable debug mode
  -h                     Displays this screen
```

```
aemsync -t http://admin:admin@localhost:4502,http://admin:admin@localhost:4503 -w ~/workspace/my_project
```

More details at

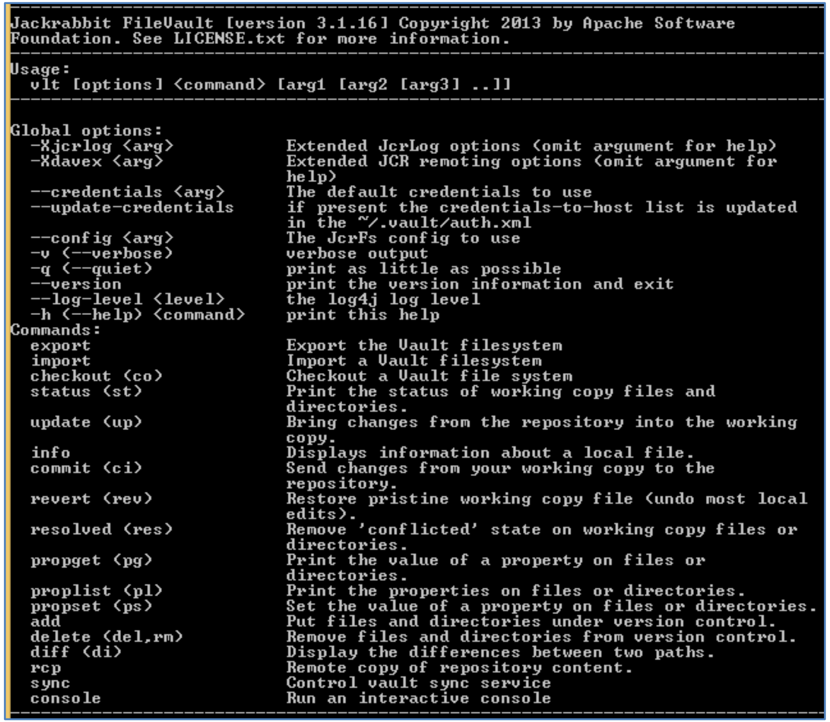
https://github.com/gavoja/aemsync

VLT

The Jackrabbit FileVault tool (VLT) is a tool developed by The Apache Foundation that maps the content of a Jackrabbit/AEM instance to your file system.

Features

- A command line tool that has similar functions as source control system client (such as a Subversion (svn), providing normal check-in, check-out and management operations, as well as configuration options for flexible representation of the project content.
- Comes with AEM, but is not installed by default. To install, copy the archive file from the cq-quickstart/opt/filevault, extract it and add **<archive-dir>/vault-cli-<version>/bin** to your environment **PATH**



```
-------------------------------------------------------------------------------
Jackrabbit FileVault [version 3.1.16] Copyright 2013 by Apache Software
Foundation. See LICENSE.txt for more information.
-------------------------------------------------------------------------------
Usage:
  vlt [options] <command> [arg1 [arg2 [arg3] ..]]
-------------------------------------------------------------------------------

Global options:
  -Xjcrlog <arg>           Extended JcrLog options (omit argument for help)
  -Xdavex <arg>            Extended JCR remoting options (omit argument for
                           help)
  --credentials <arg>      The default credentials to use
  --update-credentials     if present the credentials-to-host list is updated
                           in the ~/.vault/auth.xml
  --config <arg>           The JcrFs config to use
  -v (--verbose)           verbose output
  -q (--quiet)             print as little as possible
  --version                print the version information and exit
  --log-level <level>      the log4j log level
  -h (--help) <command>    print this help
Commands:
  export                   Export the Vault filesystem
  import                   Import a Vault filesystem
  checkout (co)            Checkout a Vault file system
  status (st)              Print the status of working copy files and
                           directories.
  update (up)              Bring changes from the repository into the working
                           copy.
  info                     Displays information about a local file.
  commit (ci)              Send changes from your working copy to the
                           repository.
  revert (rev)             Restore pristine working copy file (undo most local
                           edits).
  resolved (res)           Remove 'conflicted' state on working copy files or
                           directories.
  propget (pg)             Print the value of a property on files or
                           directories.
  proplist (pl)            Print the properties on files or directories.
  propset (ps)             Set the value of a property on files or directories.
  add                      Put files and directories under version control.
  delete (del,rm)          Remove files and directories from version control.
  diff (di)                Display the differences between two paths.
  rcp                      Remote copy of repository content.
  sync                     Control vault sync service
  console                  Run an interactive console
-------------------------------------------------------------------------------
```

More details at

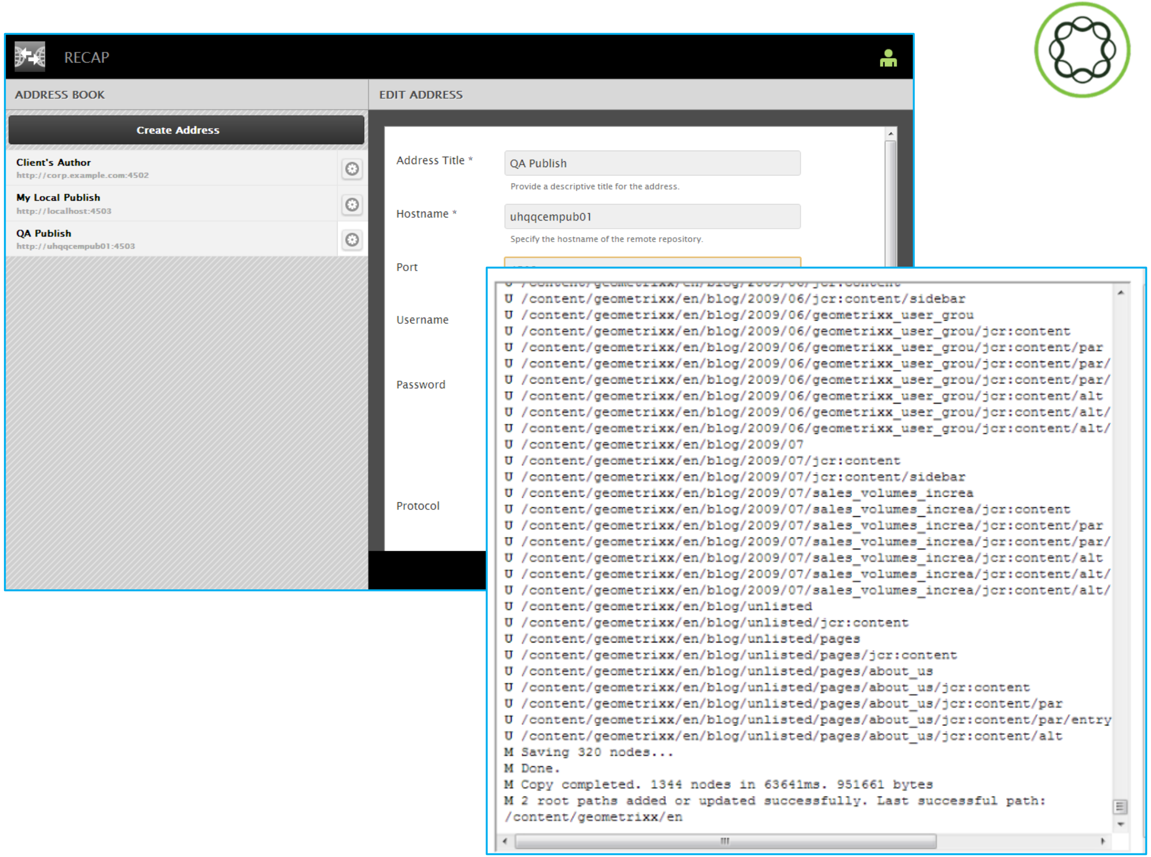
https://helpx.adobe.com/experience-manager/6-3/sites/developing/using/ht-vlttool.html

Recap

An rsync tool for CRX! Recap is based on the 'vlt rcp' command, but focuses on providing a simple web interface for syncing content between CRX instances, using a browser instead of the command-line.

Primary Objective:

- To enable a team of CQ developers in a shared testing environment to work more easily with content creators and QA testers to troubleshoot bugs and other issues on their local machines without having to waste as much time as they currently do with the overhead of creating packages specific to the problem, downloading those packages, and installing them locally.



Client's Author
http://corp.example.com:4502

My Local Publish
http://localhost:4503

QA Publish
http://uhqqcempub01:4503

Address Title * QA Publish
Provide a descriptive title for the address.

Hostname * uhqqcempub01
Specify the hostname of the remote repository.

Port

Username

Password

Protocol

```
U /content/geometrixx/en/blog/2009/06/jcr:content/sidebar
U /content/geometrixx/en/blog/2009/06/geometrixx_user_grou
U /content/geometrixx/en/blog/2009/06/geometrixx_user_grou/jcr:content
U /content/geometrixx/en/blog/2009/06/geometrixx_user_grou/jcr:content/par
U /content/geometrixx/en/blog/2009/06/geometrixx_user_grou/jcr:content/par/
U /content/geometrixx/en/blog/2009/06/geometrixx_user_grou/jcr:content/par/
U /content/geometrixx/en/blog/2009/06/geometrixx_user_grou/jcr:content/alt
U /content/geometrixx/en/blog/2009/06/geometrixx_user_grou/jcr:content/alt/
U /content/geometrixx/en/blog/2009/06/geometrixx_user_grou/jcr:content/alt/
U /content/geometrixx/en/blog/2009/07
U /content/geometrixx/en/blog/2009/07/jcr:content
U /content/geometrixx/en/blog/2009/07/jcr:content/sidebar
U /content/geometrixx/en/blog/2009/07/sales_volumes_increa
U /content/geometrixx/en/blog/2009/07/sales_volumes_increa/jcr:content
U /content/geometrixx/en/blog/2009/07/sales_volumes_increa/jcr:content/par
U /content/geometrixx/en/blog/2009/07/sales_volumes_increa/jcr:content/par/
U /content/geometrixx/en/blog/2009/07/sales_volumes_increa/jcr:content/alt
U /content/geometrixx/en/blog/2009/07/sales_volumes_increa/jcr:content/alt/
U /content/geometrixx/en/blog/2009/07/sales_volumes_increa/jcr:content/alt/
U /content/geometrixx/en/blog/unlisted
U /content/geometrixx/en/blog/unlisted/jcr:content
U /content/geometrixx/en/blog/unlisted/pages
U /content/geometrixx/en/blog/unlisted/pages/jcr:content
U /content/geometrixx/en/blog/unlisted/pages/about_us
U /content/geometrixx/en/blog/unlisted/pages/about_us/jcr:content
U /content/geometrixx/en/blog/unlisted/pages/about_us/jcr:content/par
U /content/geometrixx/en/blog/unlisted/pages/about_us/jcr:content/par/entry
U /content/geometrixx/en/blog/unlisted/pages/about_us/jcr:content/alt
M Saving 320 nodes...
M Done.
M Copy completed. 1344 nodes in 63641ms. 951661 bytes
M 2 root paths added or updated successfully. Last successful path:
/content/geometrixx/en
```

More details at

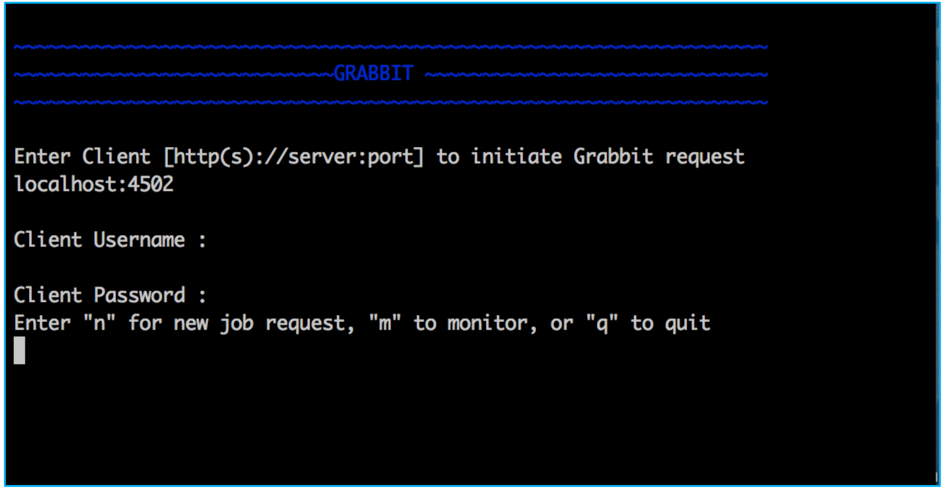
http://adamcin.net/net.adamcin.recap/

Grabbit

The purpose of this tool is to provide a fast and reliable way of copying content from one Sling (specifically Adobe CQ/AEM) instance to another.

How it works:

- Grabbit creates a stream of data using Google’s Protocol Buffers aka "ProtoBuf".
- Protocol Buffers are an extremely efficient (in terms of CPU, memory and wire size) binary protocol that includes compression.
- Doing a continuous stream avoids the latency issues. Depending on the size and nature of the data, as well as network latency, can achieve speed improvements ranging from 2 to 10 times that of Recap/vlt

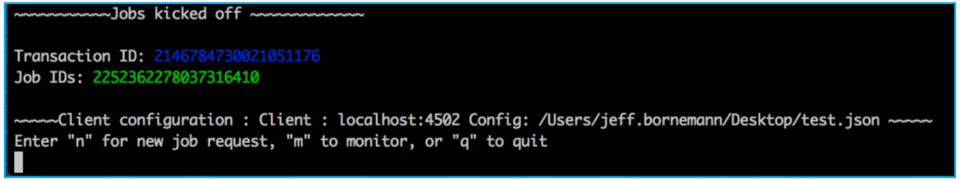


```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~GRABBIT ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Enter Client [http(s)://server:port] to initiate Grabbit request
localhost:4502

Client Username :

Client Password :
Enter "n" for new job request, "m" to monitor, or "q" to quit
█
```



```
~~~~~~~~~~~Jobs kicked off ~~~~~~~~~~~~~

Transaction ID: 2146784730021051176
Job IDs: 2252362278037316410

~~~~~Client configuration : Client : localhost:4502 Config: /Users/jeff.bornemann/Desktop/test.json ~~~~~
Enter "n" for new job request, "m" to monitor, or "q" to quit
█
```

More details at

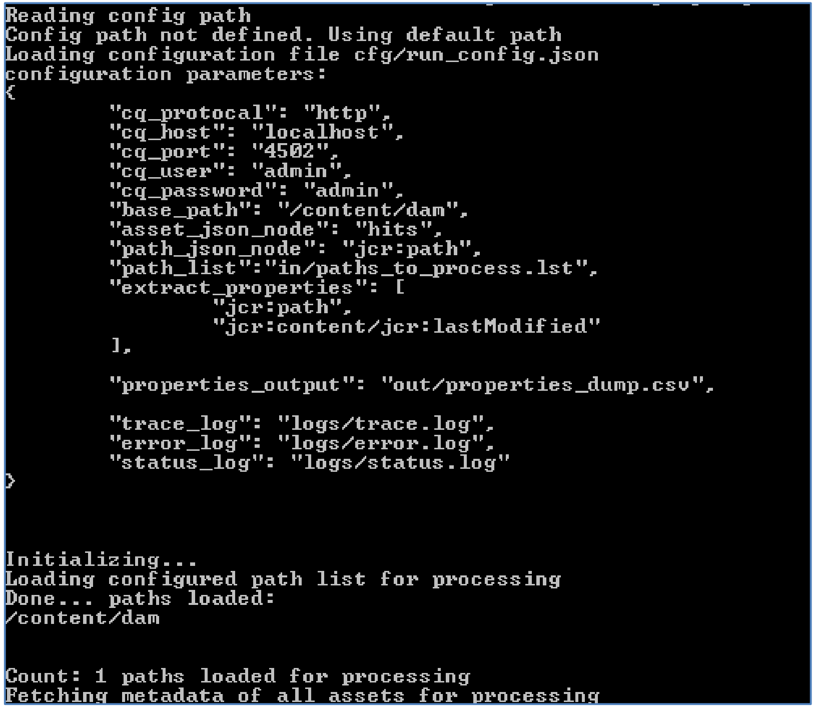
https://github.com/TWCable/grabbit

aemigrate

A set of AEM migration utilities. It includes scripts for migration of asset binaries, asset properties, asset folders & handling tag hierarchies. These scripts are developed in Python

It includes:

- Utilities for handling asset binaries (uploading, downloading)
- Utilities for handling asset properties (extracting, updating, fetching the asset list)
- Utility for creating asset folder taxonomy
- Utilities for managing tags (creating, extracting)



```
Reading config path
Config path not defined. Using default path
Loading configuration file cfg/run_config.json
configuration parameters:
{
        "cq_protocal": "http",
        "cq_host": "localhost",
        "cq_port": "4502",
        "cq_user": "admin",
        "cq_password": "admin",
        "base_path": "/content/dam",
        "asset_json_node": "hits",
        "path_json_node": "jcr:path",
        "path_list":"in/paths_to_process.lst",
        "extract_properties": [
                "jcr:path",
                "jcr:content/jcr:lastModified"
        ],

        "properties_output": "out/properties_dump.csv",

        "trace_log": "logs/trace.log",
        "error_log": "logs/error.log",
        "status_log": "logs/status.log"
}



Initializing...
Loading configured path list for processing
Done... paths loaded:
/content/dam


Count: 1 paths loaded for processing
Fetching metadata of all assets for processing
```

More details at

https://github.com/ashokkumarta/aemigrate

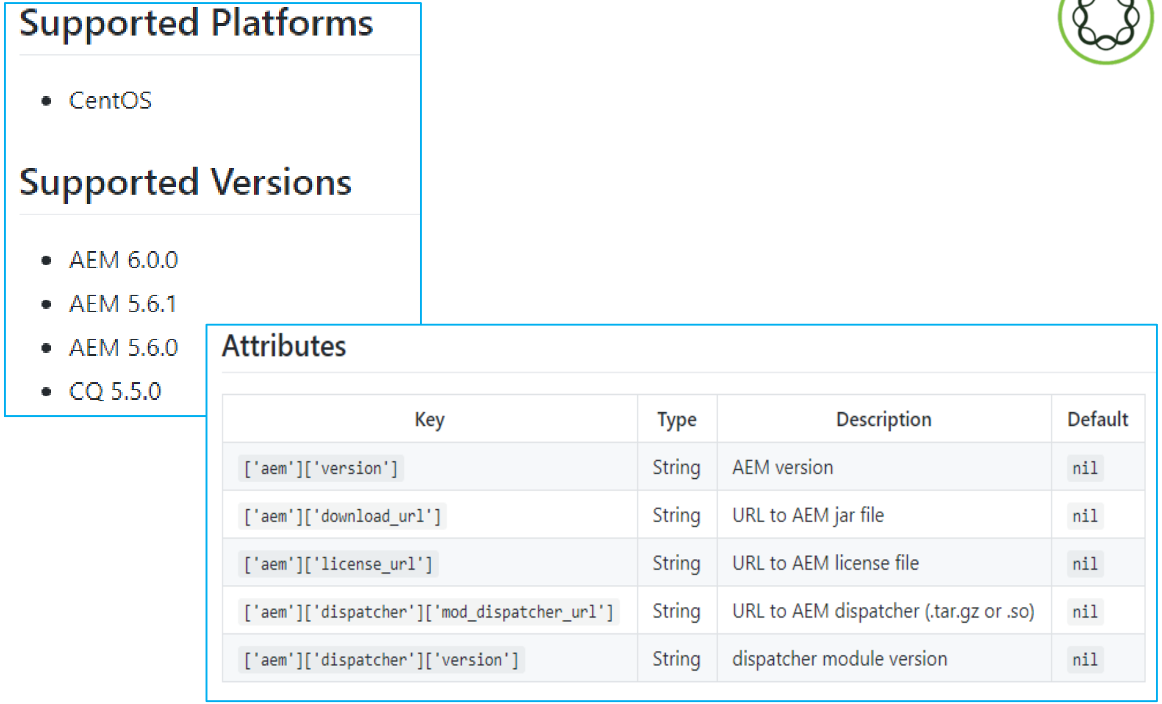
DevOps

aem-cookbook

This cookbook installs and configures Adobe Experience Manager (AEM). Includes install recipes for author, publish & dispatcher module

Features:

- Unattended installation of AEM author, publish, and dispatcher nodes.
- Automatically search for and configure AEM cluster members using chef searches.
- Configure replication agents using the replicator provider.
- Configure dispatcher farms with the farm provider.
- Deploy and remove AEM packages with the package provider.



Supported Platforms

- CentOS

Supported Versions

- AEM 6.0.0
- AEM 5.6.1
- AEM 5.6.0
- CQ 5.5.0

Attributes

Key	Type	Description	Default
`['aem']['version']`	String	AEM version	`nil`
`['aem']['download_url']`	String	URL to AEM jar file	`nil`
`['aem']['license_url']`	String	URL to AEM license file	`nil`
`['aem']['dispatcher']['mod_dispatcher_url']`	String	URL to AEM dispatcher (.tar.gz or .so)	`nil`
`['aem']['dispatcher']['version']`	String	dispatcher module version	`nil`

More details at

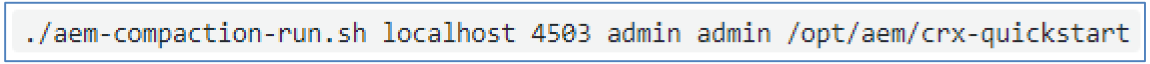
https://github.com/tacitknowledge/aem-cookbook

aemaint

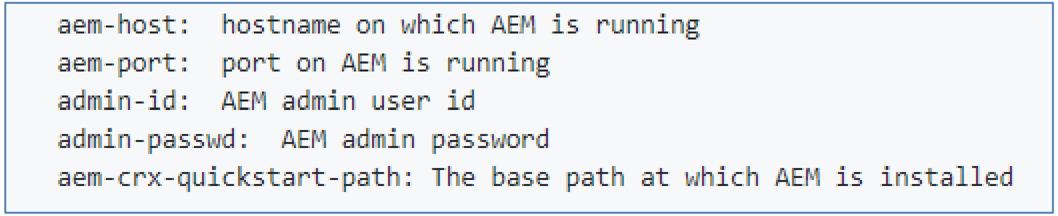
A set of scripts and utilities to perform AEM maintenance activities. It includes script for performing tar compaction.

Features:

- Automates the manual process of compaction
- Fully automated, checks for each step completion before reporting its status and moving to the next step
- Easy integration with automatic scheduling systems like Jenkins



```
./aem-compaction-run.sh localhost 4503 admin admin /opt/aem/crx-quickstart
```



```
aem-host:  hostname on which AEM is running
aem-port:  port on AEM is running
admin-id:  AEM admin user id
admin-passwd:  AEM admin password
aem-crx-quickstart-path: The base path at which AEM is installed
```

More details at

https://github.com/ashokkumarta/aemaint

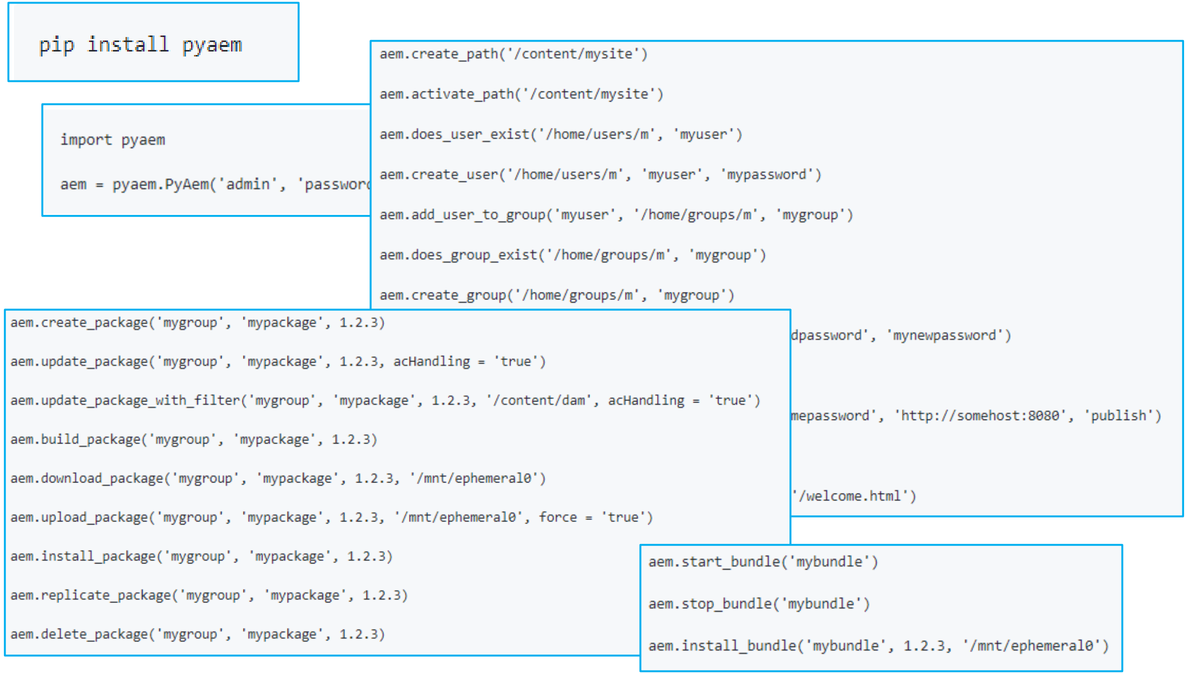
Development Utilities

PyAEM

PyAEM is a Python client for Adobe Experience Manager (AEM) API.

Features includes python APIs for

- Content management – Create, update, Activate
- User management – Check, create user/group, add, remove permission
- Package management – Create, update, build, download, upload, install, replicate, delete
- Bundle management – Install, stop, start



```
pip install pyaem
```

```
import pyaem

aem = pyaem.PyAem('admin', 'passwor
```

```
aem.create_path('/content/mysite')

aem.activate_path('/content/mysite')

aem.does_user_exist('/home/users/m', 'myuser')

aem.create_user('/home/users/m', 'myuser', 'mypassword')

aem.add_user_to_group('myuser', '/home/groups/m', 'mygroup')

aem.does_group_exist('/home/groups/m', 'mygroup')

aem.create_group('/home/groups/m', 'mygroup')

dpassword', 'mynewpassword')

mepassword', 'http://somehost:8080', 'publish')

'/welcome.html')
```

```
aem.create_package('mygroup', 'mypackage', 1.2.3)

aem.update_package('mygroup', 'mypackage', 1.2.3, acHandling = 'true')

aem.update_package_with_filter('mygroup', 'mypackage', 1.2.3, '/content/dam', acHandling = 'true')

aem.build_package('mygroup', 'mypackage', 1.2.3)

aem.download_package('mygroup', 'mypackage', 1.2.3, '/mnt/ephemeral0')

aem.upload_package('mygroup', 'mypackage', 1.2.3, '/mnt/ephemeral0', force = 'true')

aem.install_package('mygroup', 'mypackage', 1.2.3)

aem.replicate_package('mygroup', 'mypackage', 1.2.3)

aem.delete_package('mygroup', 'mypackage', 1.2.3)
```

```
aem.start_bundle('mybundle')

aem.stop_bundle('mybundle')

aem.install_bundle('mybundle', 1.2.3, '/mnt/ephemeral0')
```

More details at

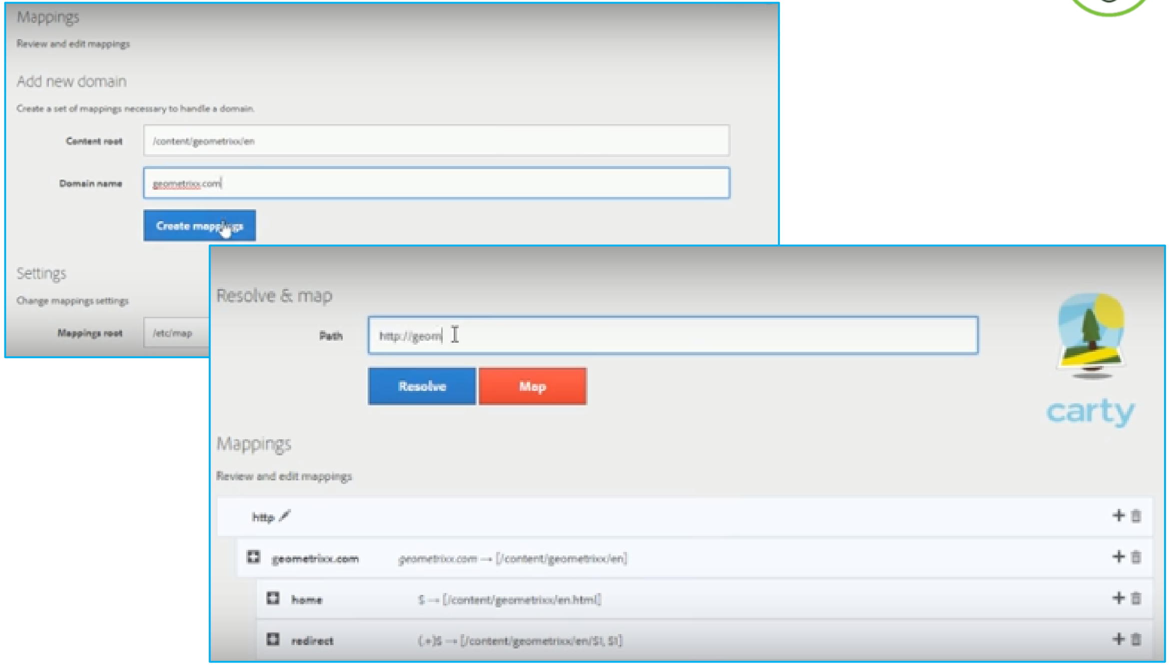
https://github.com/wildone/pyaem

Carty

Carty is an integrated tool to create, edit, review and test Sling mappings for Adobe CQ/AEM.

Advantages:

- No more switching between Felix Console and CRX DE tabs to check if the mappings finally work as expected.
- No more guessing why given URL resolves to a content path or vice versa - Carty will tell exactly which entries has been applied to a given path and why.
- You may use an automatic mapping generator (just enter domain and content path) or manually edit your configuration if you need more precise control.



Mappings

Review and edit mappings

Add new domain

Create a set of mappings necessary to handle a domain.

Content root: /content/geometrixx/en

Domain name: geometrixx.com

Settings

Change mappings settings

Mappings root: /etc/map

Resolve & map

Path: http://geom

Mappings

Review and edit mappings

http

geometrixx.com geometrixx.com → [/content/geometrixx/en]

home $ → [/content/geometrixx/en.html]

redirect (.+)$ → [/content/geometrixx/en/$1, $1]

More details at

http://cognifide.github.io/Carty/

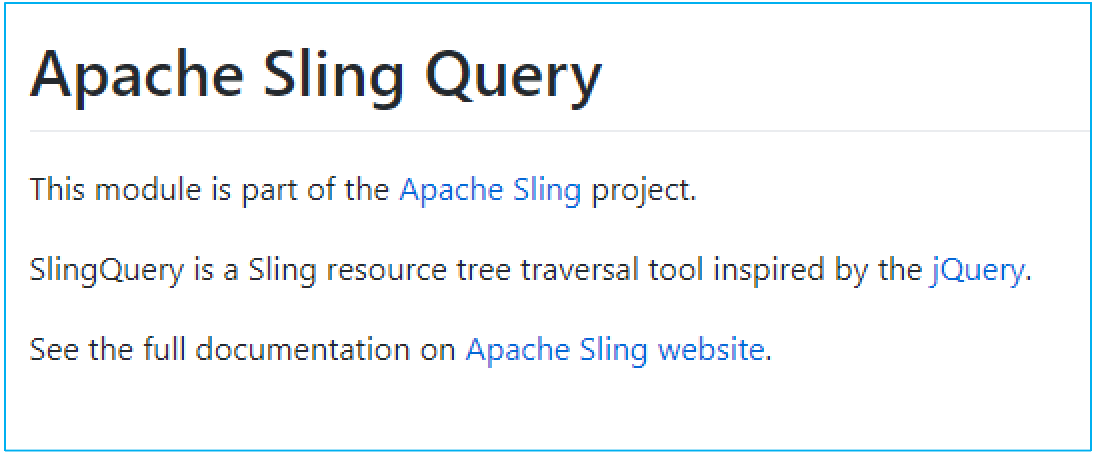
SlingQuery

SlingQuery is a Sling resource tree traversal tool inspired by the jQuery.

Features:

- Useful for operations to traverse the resource tree
- Provides flexible filtering syntax
- Supports lazy evaluation of the query result
- Immutable SlingQuery object which is thread-safe
- And its fluent, friendly, jQuery-like API.

Note: This tool has been contributed to Apache Sling project



Apache Sling Query

This module is part of the Apache Sling project.

SlingQuery is a Sling resource tree traversal tool inspired by the jQuery.

See the full documentation on Apache Sling website.

More details at

http://cognifide.github.io/sling-query/

AEM Rules for SonarQube

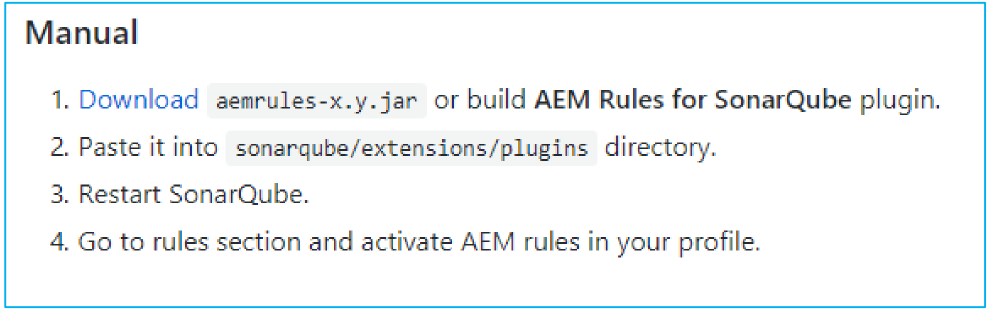
This tool is intended to find common bugs and bad smells specific for AEM development.

Documentation of each rule is available from SonarQube interface after plugin installation

This ruleset helps enforce

- Good practices in AEM coding
- Prevent possible bugs in AEM project development
- Sling & slice related coding Rules

AEM Rules
for SonarQube



Manual

1. Download `aemrules-x.y.jar` or build **AEM Rules for SonarQube** plugin.
2. Paste it into `sonarqube/extensions/plugins` directory.
3. Restart SonarQube.
4. Go to rules section and activate AEM rules in your profile.

More details at

http://cognifide.github.io/aem-rules-for-sonarqube/

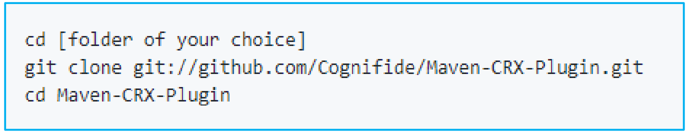
Maven CRX Plugin

This is a plugin designed for Maven 2.x+ based builds to automate CRX compliant packages deployment.

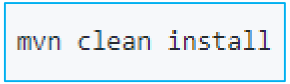
It allows to set up upload and install commands as a step in a build process speeding up entire development cycle

Features:

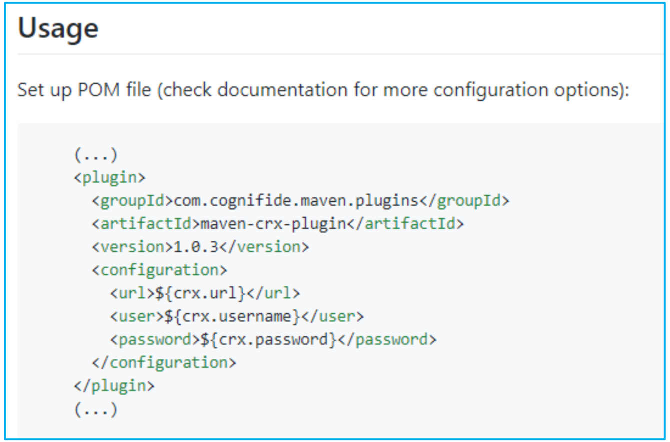
- Uploads a selected artifact to a running CRX instance.
- Installs previously uploaded artifact in a running CRX instance.
- Activates (replicates and installs) previously uploaded artifact on all publish instances (requires CRX 2.3 or later).



```
cd [folder of your choice]
git clone git://github.com/Cognifide/Maven-CRX-Plugin.git
cd Maven-CRX-Plugin
```



```
mvn clean install
```



Usage

Set up POM file (check documentation for more configuration options):

```
(...)
<plugin>
  <groupId>com.cognifide.maven.plugins</groupId>
  <artifactId>maven-crx-plugin</artifactId>
  <version>1.0.3</version>
  <configuration>
    <url>${crx.url}</url>
    <user>${crx.username}</user>
    <password>${crx.password}</password>
  </configuration>
</plugin>
(...)
```

More details at

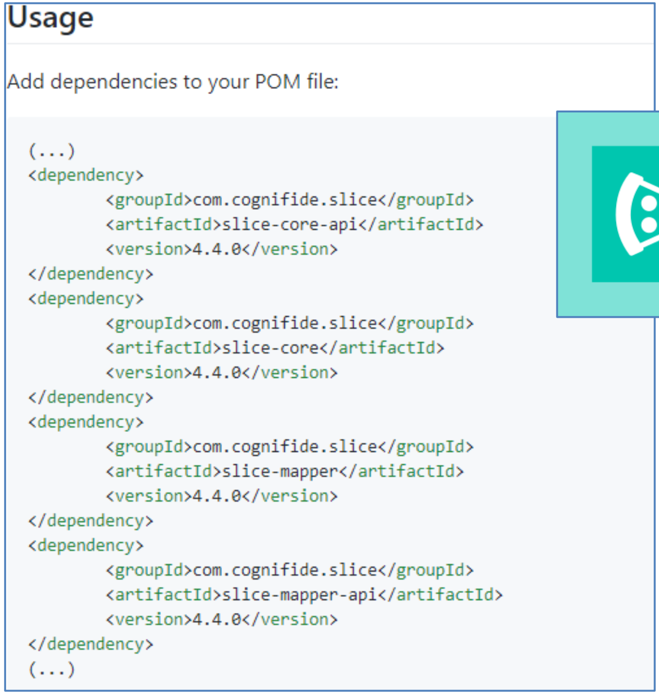
http://cognifide.github.io/crx-plugin/

Slice

Slice is a framework which simplifies AEM development by using dependency injection pattern. It glues Sling and Google Guice together, allowing you to map resources to Java models seamlessly

Features

- Lean and neat code, slicker design!
- Improved testability of code
- Improved maintenance of your code
- Easy to start with! Slice is easy to learn
- Faster development – with code reuse, dependency injection and simplicity
- Overall costs reduced!

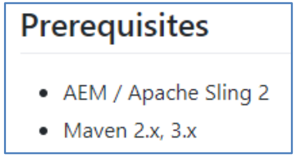


Usage

Add dependencies to your POM file:

```
(...)
<dependency>
        <groupId>com.cognifide.slice</groupId>
        <artifactId>slice-core-api</artifactId>
        <version>4.4.0</version>
</dependency>
<dependency>
        <groupId>com.cognifide.slice</groupId>
        <artifactId>slice-core</artifactId>
        <version>4.4.0</version>
</dependency>
<dependency>
        <groupId>com.cognifide.slice</groupId>
        <artifactId>slice-mapper</artifactId>
        <version>4.4.0</version>
</dependency>
<dependency>
        <groupId>com.cognifide.slice</groupId>
        <artifactId>slice-mapper-api</artifactId>
        <version>4.4.0</version>
</dependency>
(...)
```

Slice



Prerequisites

- AEM / Apache Sling 2
- Maven 2.x, 3.x

More details at

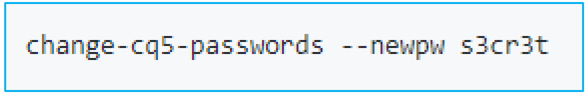
http://cognifide.github.io/slice/

cq5tools

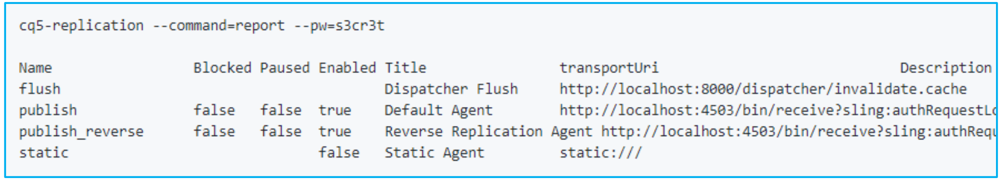
A collection of scripts to allow one to administrate parts of CQ5 without having to use the GUI and thus avoid RSI. Especially useful when handling many instances on many machines.

The tool set includes

- change-cq5-passwords - change the admin password in the five (CQ5.4) or the two (CQ5.5) places one needs to change it for it to be changed everywhere
- cq5-replication - display replication agent logs, test replication agent configuration, and create new replication agents all from the command line
- cq5-users - create an initial set of users

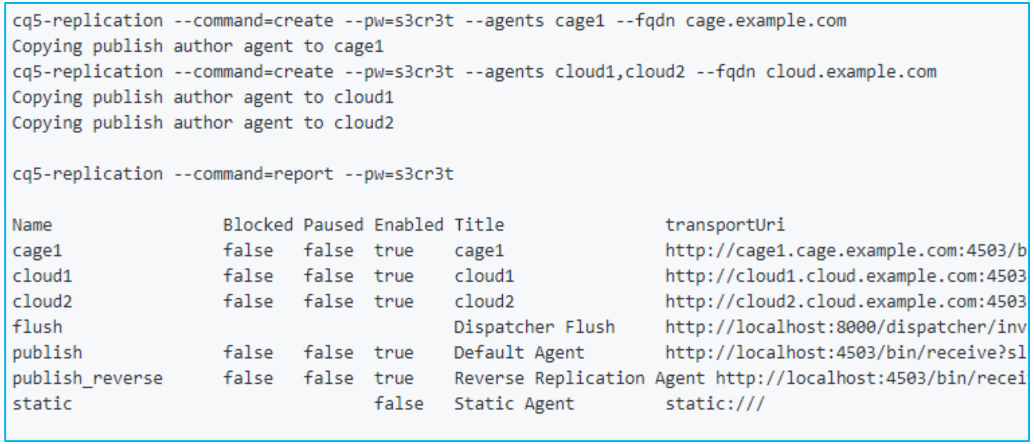


```
change-cq5-passwords --newpw s3cr3t
```



```
cq5-replication --command=report --pw=s3cr3t

Name                 Blocked Paused Enabled Title                transportUri                             Description
flush                                       Dispatcher Flush     http://localhost:8000/dispatcher/invalidate.cache
publish              false   false  true    Default Agent        http://localhost:4503/bin/receive?sling:authRequestLo
publish_reverse      false   false  true    Reverse Replication Agent http://localhost:4503/bin/receive?sling:authRequ
static                              false   Static Agent         static:///
```



```
cq5-replication --command=create --pw=s3cr3t --agents cage1 --fqdn cage.example.com
Copying publish author agent to cage1
cq5-replication --command=create --pw=s3cr3t --agents cloud1,cloud2 --fqdn cloud.example.com
Copying publish author agent to cloud1
Copying publish author agent to cloud2

cq5-replication --command=report --pw=s3cr3t

Name                 Blocked Paused Enabled Title                transportUri
cage1                false   false  true    cage1                http://cage1.cage.example.com:4503/b
cloud1               false   false  true    cloud1               http://cloud1.cloud.example.com:4503
cloud2               false   false  true    cloud2               http://cloud2.cloud.example.com:4503
flush                                       Dispatcher Flush     http://localhost:8000/dispatcher/inv
publish              false   false  true    Default Agent        http://localhost:4503/bin/receive?sl
publish_reverse      false   false  true    Reverse Replication Agent http://localhost:4503/bin/recei
static                              false   Static Agent         static:///
```

More details at

https://github.com/42Lines/cq5tools

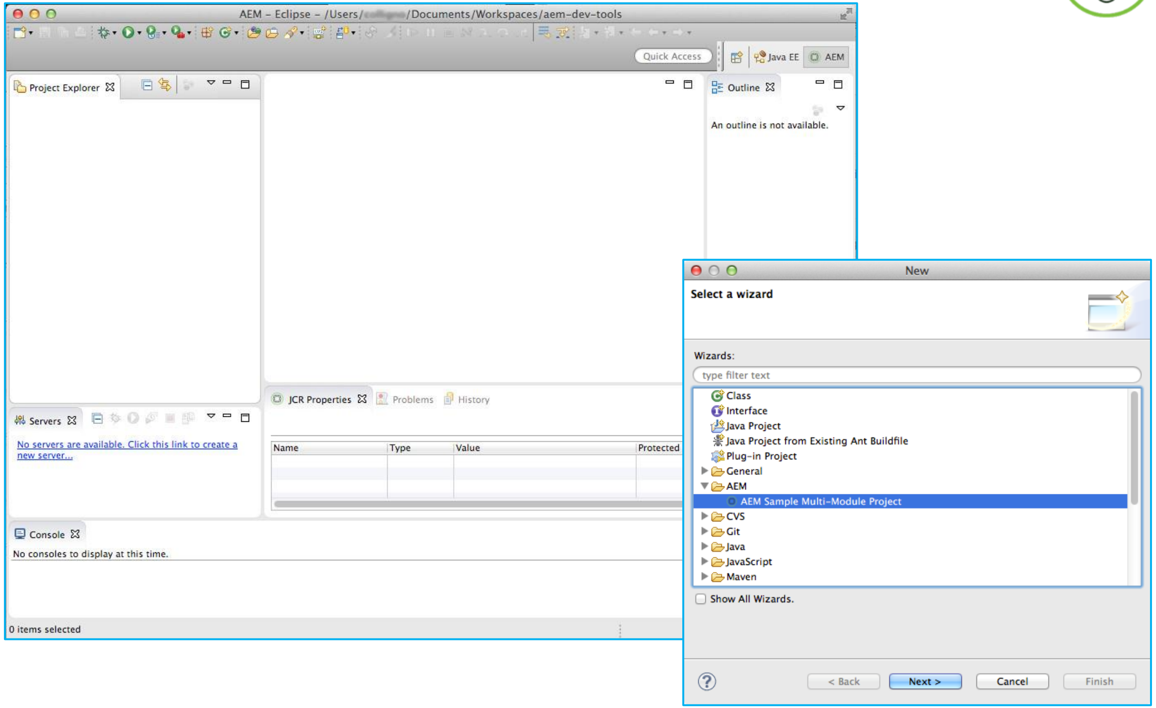
Integrated Development Environment (IDE) Support

AEM Developer Tools for Eclipse

It offers several features that make AEM development easier

Features:

- Seamless integration with AEM instances through Eclipse Server Connector.
- Synchronization for both content and OSGI bundles.
- Debugging support with code hot-swaping capability.
- Simple bootstrap of AEM projects via a specific Project Creation Wizard.
- Easy JCR properties edition.



AEM – Eclipse – /Users/[redacted]/Documents/Workspaces/aem-dev-tools

An outline is not available.

No servers are available. Click this link to create a new server...

No consoles to display at this time.

0 items selected

Select a wizard

Wizards: type filter text

☐ Show All Wizards.

More details at

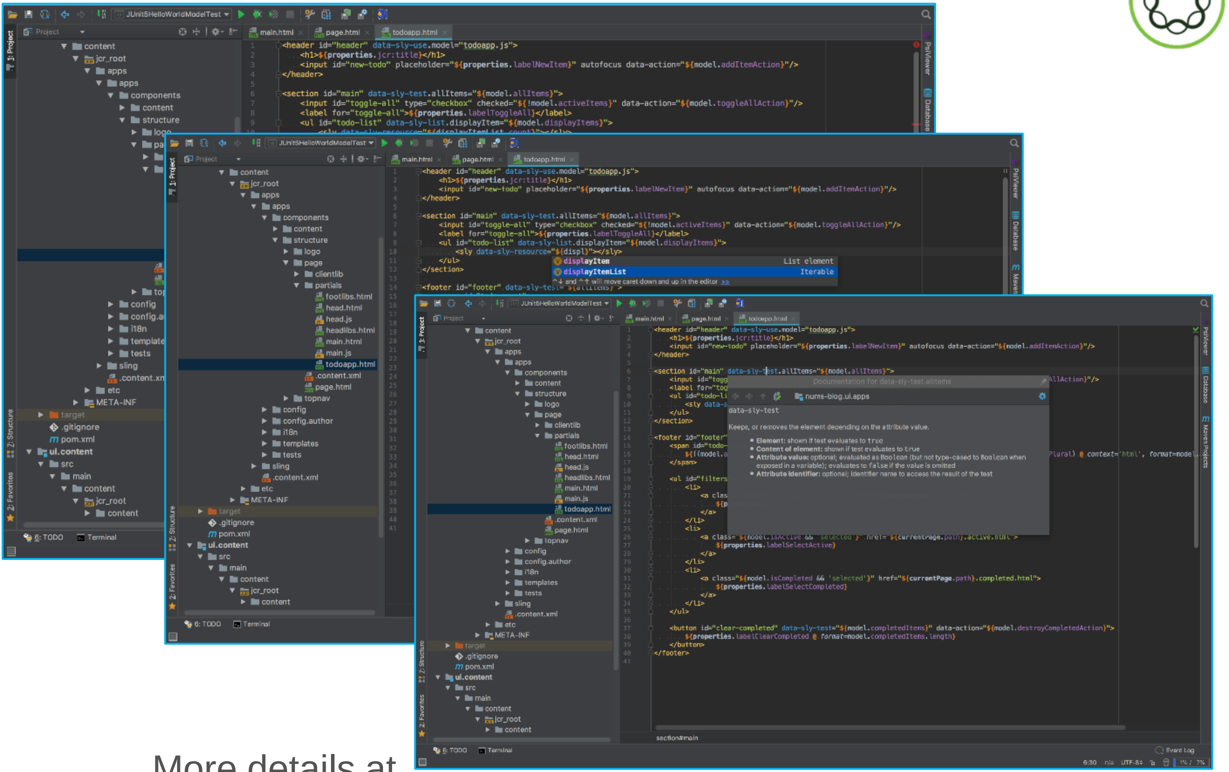
https://helpx.adobe.com/in/experience-manager/6-3/sites/developing/using/aem-eclipse.html

AEM IntelliJ Plugin

Intellij Platform plugin for Adobe Experience Manager

Features:

- JCR roots auto-detection
- HTL/Sightly support (HTML files under jcr_root directory)
- syntax validation and highlighting
- auto-completion, inspections and documentation for:
  - block attributes (data-sly-*)
  - global objects (properties, currentPage,…)
  - properties (jcr:title, cq:lastModified, …)
  - built-in expression options (context, …)
  - display contexts (html, scriptToken, …)
  - use objects, block variables, template parameters, …



```
<header id="header" data-sly-use.model="todoapp.js">
    <h1>${properties.jcr:title}</h1>
    <input id="new-todo" placeholder="${properties.labelNewItem}" autofocus data-action="${model.addItemAction}"/>
</header>

<section id="main" data-sly-test.allItems="${model.allItems}">
    <input id="togg                                                    AllAction}"/>
    <label for="tog
    <ul id="todo-li
        <sly data-s
    </ul>
</section>

<footer id="footer"
    <span id="todo-
        ${[(model.a                                                    Plural) @ context='html', format=model...
    </span>

    <ul id="filters
        <li>
            <a clas
                ${p
            </a>
        </li>
        <li>
            <a class="${model.isActive && 'selected'}" href="${currentPage.path}.active.html">
                ${properties.labelSelectActive}
            </a>
        </li>
        <li>
            <a class="${model.isCompleted && 'selected'}" href="${currentPage.path}.completed.html">
                ${properties.labelSelectCompleted}
            </a>
        </li>
    </ul>

    <button id="clear-completed" data-sly-test="${model.completedItems}" data-action="${model.destroyCompletedAction}">
        ${properties.labelClearCompleted @ format=model.completedItems.length}
    </button>
</footer>
```

Documentation for data-sly-test.allItems

nums-blog.ui.apps

data-sly-test

Keeps, or removes the element depending on the attribute value.

- **Element:** shown if test evaluates to true
- **Content of element:** shown if test evaluates to true
- **Attribute value:** optional; evaluated as Boolean (but not type-cased to Boolean when exposed in a variable); evaluates to false if the value is omitted
- **Attribute identifier:** optional; identifier name to access the result of the test

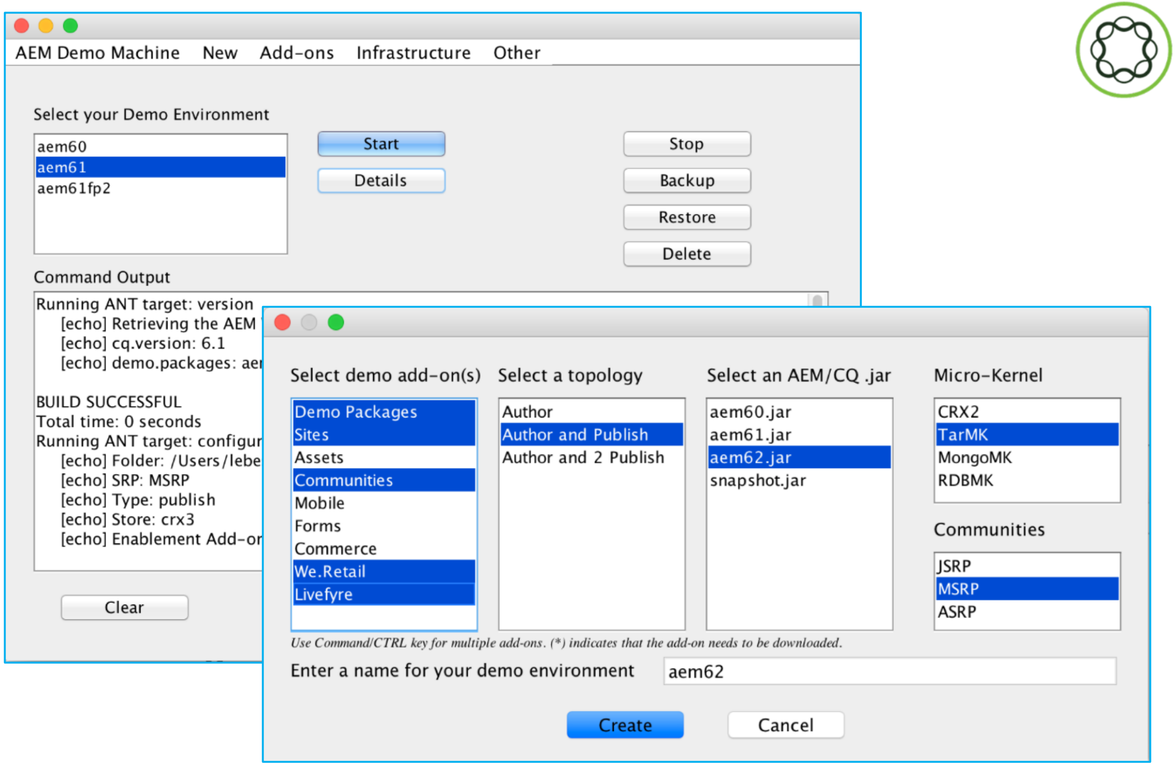
More details at

https://plugins.jetbrains.com/plugin/9269-aem-intellij-plugin

AEM Demo Machine

The AEM Demo Machine consists of

Graphical user interface / User Interface, scripts and packages to install, to manage and to run demos for AEM [Sites], Assets, Communities, Mobile and Forms, which in most situations require more setup than starting a QuickStart instance, and sometimes additional infrastructure such as MongoDB, SOLR, MySQL, FFMPEG or email servers.



Select your Demo Environment

- aem60
- aem61
- aem61fp2

Command Output

```
Running ANT target: version
    [echo] Retrieving the AEM
    [echo] cq.version: 6.1
    [echo] demo.packages: ae

BUILD SUCCESSFUL
Total time: 0 seconds
Running ANT target: configur
    [echo] Folder: /Users/lebe
    [echo] SRP: MSRP
    [echo] Type: publish
    [echo] Store: crx3
    [echo] Enablement Add-or
```

Select demo add-on(s): Demo Packages, Sites, Assets, Communities, Mobile, Forms, Commerce, We.Retail, Livefyre

Select a topology: Author, Author and Publish, Author and 2 Publish

Select an AEM/CQ .jar: aem60.jar, aem61.jar, aem62.jar, snapshot.jar

Micro-Kernel: CRX2, TarMK, MongoMK, RDBMK

Communities: JSRP, MSRP, ASRP

Use Command/CTRL key for multiple add-ons. (*) indicates that the add-on needs to be downloaded.

Enter a name for your demo environment: aem62

The AEM Demo Machine was tested successfully with CQ 5.5, CQ 5.6.1, AEM 6.0, AEM 6.1, AEM 6.2, AEM 6.3 and current internal 6.4 snapshots, on Windows, OS X, Linux and Docker.

More details at

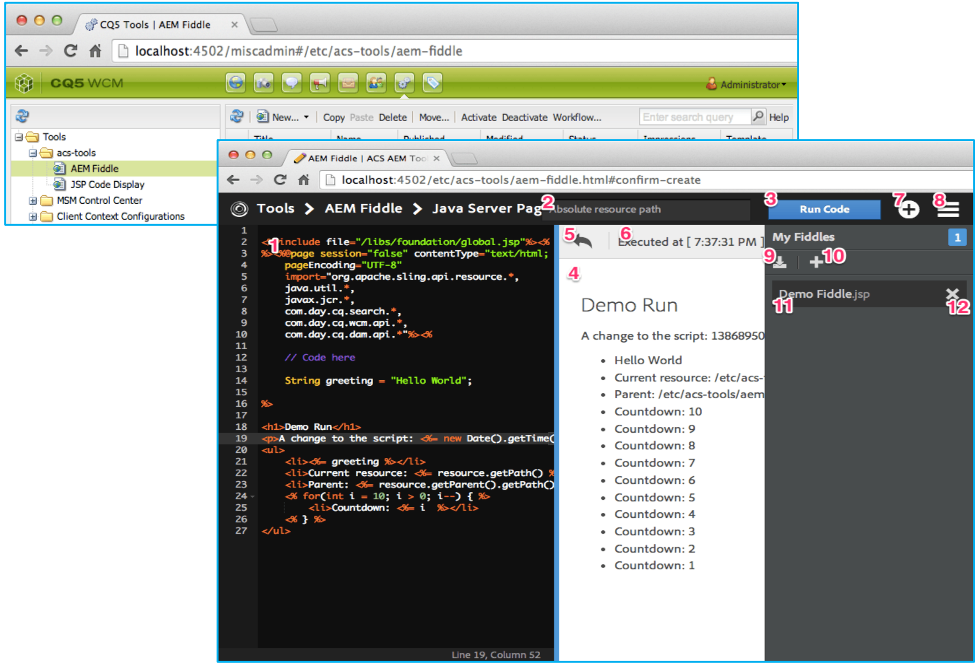
https://github.com/Adobe-Marketing-Cloud/aem-demo-machine/wiki

AEM Fiddle

The intent of AEM Fiddle is to provide an accessible, developer friendly environment for experimentation, POCs and other one-off tasks. AEM Fiddle is not intended to be used as a full fledged application development environment.

Advantages:

- AEM Fiddle’s fundamental advantage over other AEM IDEs is: time to execution.
- Most AEM IDE/development tooling requires the orchestration of scripts, resources and sling:resourceTypes.
- AEM Fiddle allows you to begin coding and executing your code in seconds (literally).



```
<%@include file="/libs/foundation/global.jsp"%><%
%><%@page session="false" contentType="text/html;
    pageEncoding="UTF-8"
    import="org.apache.sling.api.resource.*,
    java.util.*,
    javax.jcr.*,
    com.day.cq.search.*,
    com.day.cq.wcm.api.*,
    com.day.cq.dam.api.*"%><%

    // Code here

    String greeting = "Hello World";

%>

<h1>Demo Run</h1>
<p>A change to the script: <%= new Date().getTime(
<ul>
    <li><%= greeting %></li>
    <li>Current resource: <%= resource.getPath() %
    <li>Parent: <%= resource.getParent().getPath()
    <% for(int i = 10; i > 0; i--) { %>
        <li>Countdown: <%= i  %></li>
    <% } %>
</ul>
```

```
Demo Run

A change to the script: 13868950

  • Hello World
  • Current resource: /etc/acs-
  • Parent: /etc/acs-tools/aem
  • Countdown: 10
  • Countdown: 9
  • Countdown: 8
  • Countdown: 7
  • Countdown: 6
  • Countdown: 5
  • Countdown: 4
  • Countdown: 3
  • Countdown: 2
  • Countdown: 1
```

More details at

https://adobe-consulting-services.github.io/acs-aem-tools/features/aem-fiddle/index.html

Testing

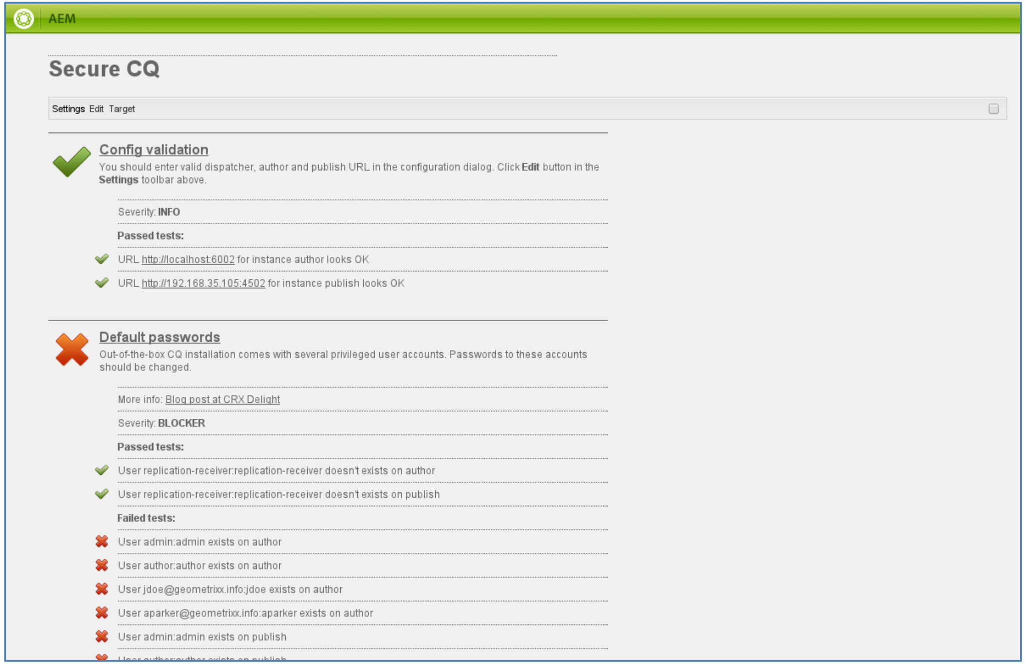
SecureCQ

Secure CQ is a tool which can be used to find the most popular security problems in your CQ instance.

It tests both author, publish instances and also validates the dispatcher

It checks for

- if the default passwords are changed
- if there are no unnecessary protocols enabled after being published
- if the the administrator console access is disabled
- if content-grabbing selectors are restricted on the dispatcher
- Other checks



Secure CQ

Config validation

You should enter valid dispatcher, author and publish URL in the configuration dialog. Click **Edit** button in the **Settings** toolbar above.

Severity: **INFO**

Passed tests:

✓ URL http://localhost:6002 for instance author looks OK

✓ URL http://192.168.35.105:4502 for instance publish looks OK

Default passwords

Out-of-the-box CQ installation comes with several privileged user accounts. Passwords to these accounts should be changed.

More info: Blog post at CRX Delight

Severity: **BLOCKER**

Passed tests:

✓ User replication-receiver:replication-receiver doesn't exists on author

✓ User replication-receiver:replication-receiver doesn't exists on publish

Failed tests:

✗ User admin:admin exists on author

✗ User author:author exists on author

✗ User jdoe@geometrixx.info:jdoe exists on author

✗ User aparker@geometrixx.info:aparker exists on author

✗ User admin:admin exists on publish

✗ User author:author exists on publish

More details at

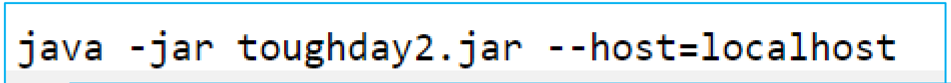
http://cognifide.github.io/secure-cq/

Tough Day

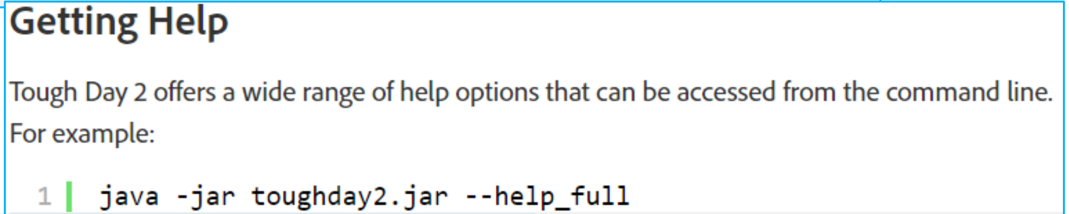
It is a an application that allows you to stress test the limits of your AEM instance. It can be run out of the box with the default test suite or it can be configured to fit your testing needs.

Default use cases tested:

- Create pages and live copies for them (including rollouts)
- Get Homepage
- Run queries in querybuilder
- Create asset hierarchies
- Delete assets



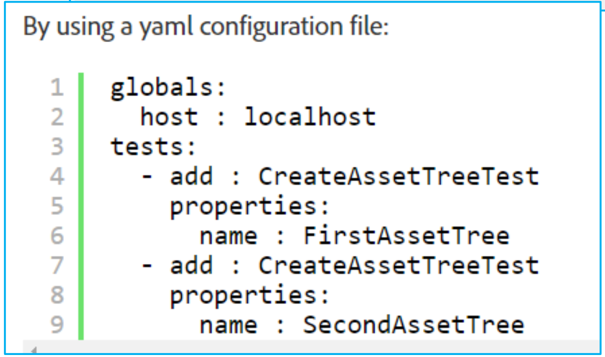
```
java -jar toughday2.jar --host=localhost
```



Getting Help

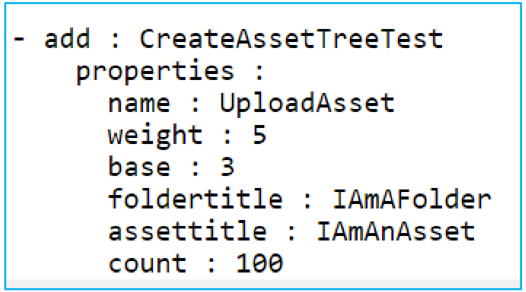
Tough Day 2 offers a wide range of help options that can be accessed from the command line. For example:

```
java -jar toughday2.jar --help_full
```



By using a yaml configuration file:

```
globals:
  host : localhost
tests:
  - add : CreateAssetTreeTest
    properties:
      name : FirstAssetTree
  - add : CreateAssetTreeTest
    properties:
      name : SecondAssetTree
```



```
- add : CreateAssetTreeTest
    properties :
      name : UploadAsset
      weight : 5
      base : 3
      foldertitle : IAmAFolder
      assettitle : IAmAnAsset
      count : 100
```

More details at

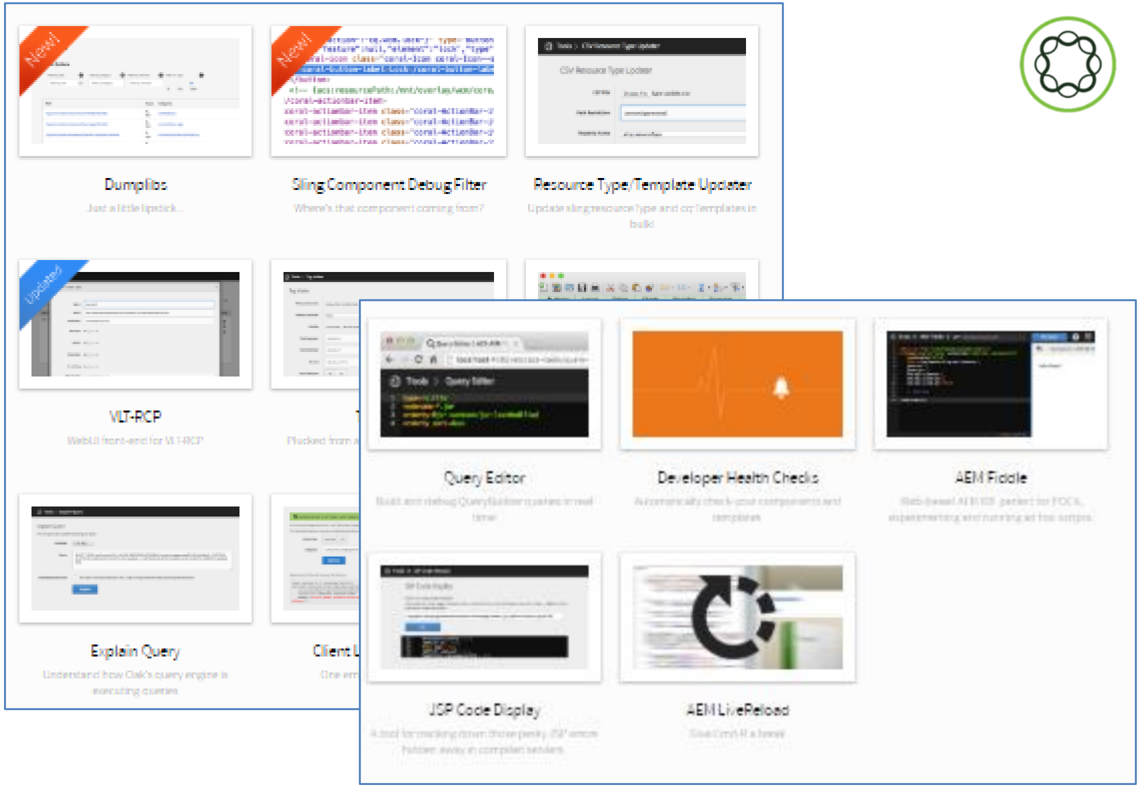
https://helpx.adobe.com/experience-manager/6-3/sites/developing/using/tough-day.html

ACS Tools

ACS AEM Tools

A growing set of tools maintained by Adobe Consulting Services. These tools are meant to make developers life easier.

Note: These tools are not recommended to be deployed to Production and are meant only for developers use



Dumplibs

Just a little lipstick...

Sling Component Debug Filter

Where's that component coming from?

Resource Type/Template Updater

Update sling:resourceType and cq:Templates in bulk!

VLT-RCP

WebUI front-end for VLT-RCP

Plucked from a

Explain Query

Understand how Oak's query engine is executing queries

Client L

One err

Query Editor

Build and debug QueryBuilder queries in real time

Developer Health Checks

Automatically check your components and templates

AEM Fiddle

Web-based AEM IDE perfect for POCs, experimenting and running ad hoc scripts

JSP Code Display

A tool for tracking down those pesky JSP errors hidden away in compiled servlets

AEM LiveReload

Give Cmd-R a break

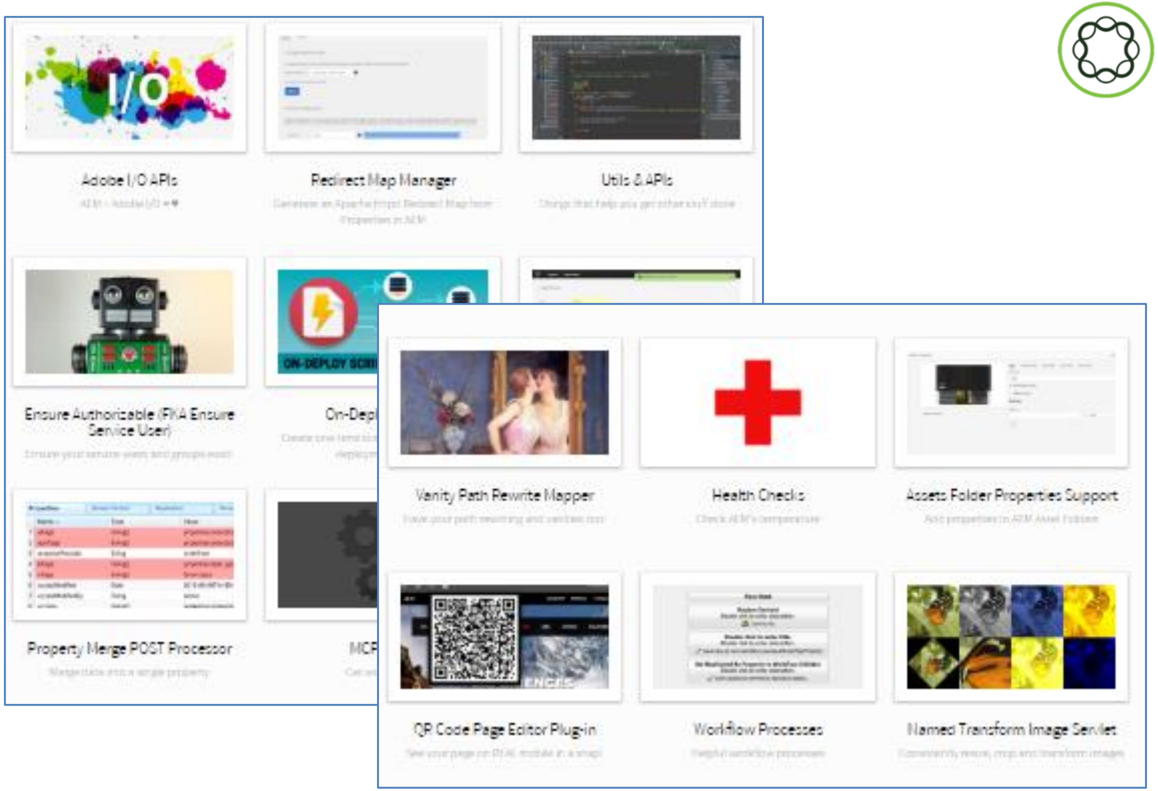
More details at

https://adobe-consulting-services.github.io/acs-aem-tools/

ACS AEM Commons

A growing set of utilities & reusable components maintained by Adobe Consulting Services; which can be used as-is or leveraged as a base to build solutions on.

Note: Unlike ACS AEM Tools, the components under ACS AEM Commons can be deployed to production. However note that AEM support license do not cover any issues or bugs with these components.


Adobe I/O APIs
AEM + Adobe I/O = ♥
Redirect Map Manager
Generate an Apache httpd Redirect Map from Properties in AEM
Utils & APIs
Things that help you get other stuff done
Ensure Authorizable (FKA Ensure Service User)
Ensure your service users and groups exist
On-Dep
ON-DEPLOY SCRI
Property Merge POST Processor
Merge data into a single property
MCP
Vanity Path Rewrite Mapper
Health Checks
Check AEM's temperature
Assets Folder Properties Support
Add properties to AEM Asset Folders
QR Code Page Editor Plug-in
See your page on REAL mobile in a snap
Workflow Processes
Helpful workflow processes
Named Transform Image Servlet
Conveniently resize, crop and transform images

More details at

https://adobe-consulting-services.github.io/acs-aem-commons/

Thank You

Feedback and suggestions welcome. Please write to

ashokkumar_ta / ashokkumar.ta@gmail.com