

# Deploying Alfresco EA 6.0 using Docker

Docker compose Deployment Overview:

<https://github.com/Alfresco/acs-deployment/blob/master/docs/diagrams/docker-compose/docker-compose-components.png>

On linux:

## 1. Pre-requisite

Installation of docker and docker-compose

- a) `wget https://download.docker.com/linux/centos/7/x86_64/stable/Packages/docker-ce-17.09.1.ce-1.el7.centos.x86_64.rpm`
- b) `sudo yum install ./docker-ce-17.09.1.ce-1.el7.centos.x86_64.rpm`

```
Dependencies Resolved
```

Package	Arch	Version	Repository	Size
Installing: <b>docker-ce</b>	x86_64	17.09.1.ce-1.el7.centos	/docker-ce-17.09.1.ce-1.el7.centos.x86_64	76 M
Installing for dependencies: container-selinux	noarch	2:2.55-1.el7	extras	34 k

```
Transaction Summary
```

```
Install 1 Package (+1 Dependent package)
```

- c) `sudo systemctl start docker`
- d) `docker -v`
- e) `sudo systemctl status docker`

```
[root@localhost alf community]# docker --version
Docker version 17.09.1-ce, build 19e2cf6
[root@localhost alf community]# sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; disabled; vendor preset: disabled)
   Active: active (running) since Mon 2018-05-21 22:19:35 EDT; 4min 18s ago
     Docs: https://docs.docker.com
   Main PID: 6282 (dockerd)
    Tasks: 20
   Memory: 20.3M
   CGroup: /system.slice/docker.service
           └─6282 /usr/bin/dockerd
             └─6289 docker-containerd -l unix:///var/run/docker/libcontainerd/docker-containerd.sock --metrics-interval=0 --start-timeout 2m --state...
```

```
May 21 22:19:33 localhost.localdomain dockerd[6282]: time="2018-05-21T22:19:33.878464905-04:00" level=info msg="libcontainerd: new container... 6289"
May 21 22:19:34 localhost.localdomain dockerd[6282]: time="2018-05-21T22:19:34.888310480-04:00" level=warning msg="failed to rename /var/li...nously"
May 21 22:19:35 localhost.localdomain dockerd[6282]: time="2018-05-21T22:19:35.005828907-04:00" level=info msg="Graph migration to content-...econds"
May 21 22:19:35 localhost.localdomain dockerd[6282]: time="2018-05-21T22:19:35.007162732-04:00" level=info msg="Loading containers: start."
May 21 22:19:35 localhost.localdomain dockerd[6282]: time="2018-05-21T22:19:35.254455532-04:00" level=info msg="Default bridge (docker0) is...ddress"
May 21 22:19:35 localhost.localdomain dockerd[6282]: time="2018-05-21T22:19:35.516210606-04:00" level=info msg="Loading containers: done."
May 21 22:19:35 localhost.localdomain dockerd[6282]: time="2018-05-21T22:19:35.554954530-04:00" level=info msg="Docker daemon" commit=19e2c...09.1-ce
May 21 22:19:35 localhost.localdomain dockerd[6282]: time="2018-05-21T22:19:35.555334278-04:00" level=info msg="Daemon has completed initialization"
May 21 22:19:35 localhost.localdomain dockerd[6282]: time="2018-05-21T22:19:35.598809753-04:00" level=info msg="API listen on /var/run/docker.sock"
May 21 22:19:35 localhost.localdomain systemd[1]: Started Docker Application Container Engine.
Hint: Some lines were ellipsized, use -l to show in full.
[root@localhost alf_community]# docker run hello-world

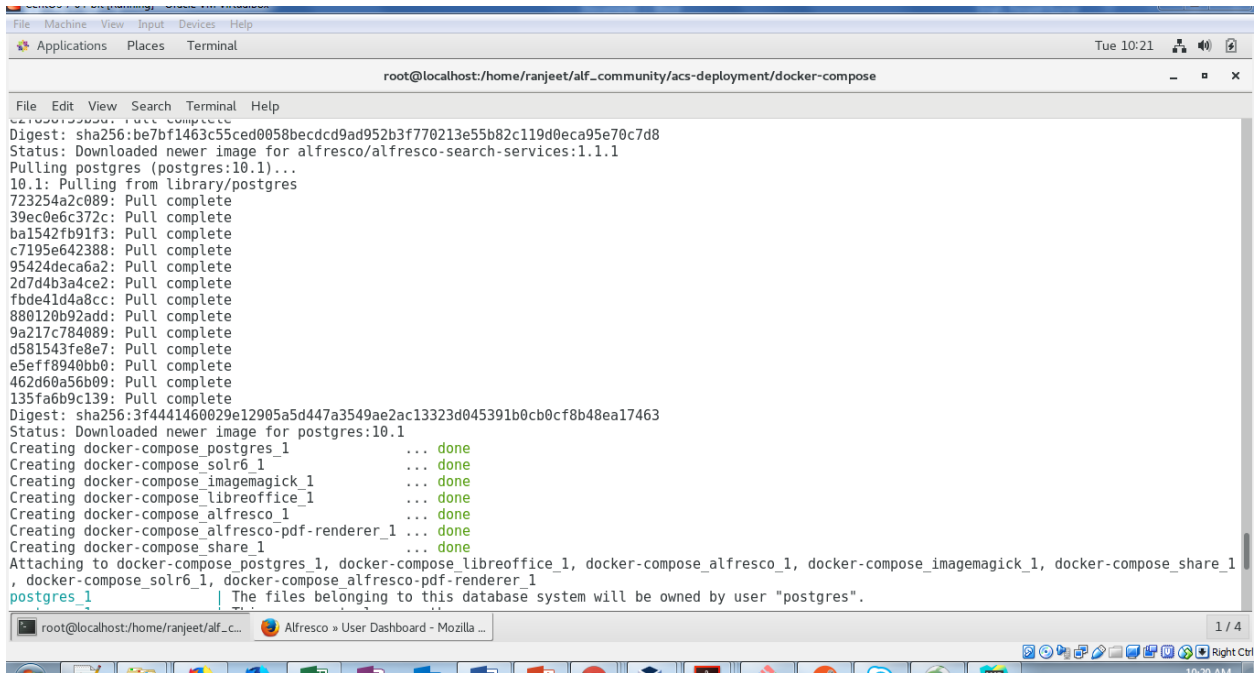
Hello from Docker!
```

- f) `sudo pip install docker-compose`
- g) `docker-compose --version`

## 2. Deploying the ACS Deployment project

- a) `git clone https://github.com/Alfresco/acs-deployment.git`
- b) Ports 5432, 8080, 8082, 8083 are configured by default and Can be edit in the docker-compose.yml file.

- c) Move to acs-deployment folder and then to docker-compose and run the command:
- d) docker-compose up



- e) The services can then be accessed using urls:
  - http://localhost:8080/share to have access to Alfresco Share (login: admin, password: admin).
  - http://localhost:8082/alfresco for the repository default page.
  - http://localhost:8083/solr to have access to Solr Administration panel.

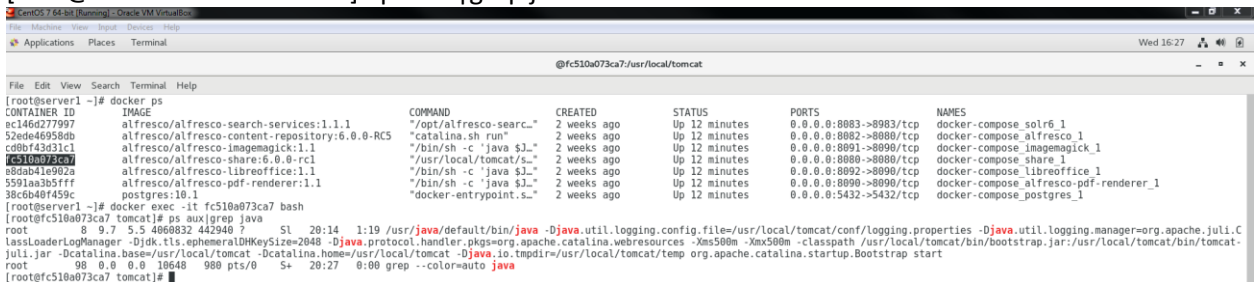
Installation can be inspected under /usr/local/tomcat when server started(As below)

**\$ docker ps**

**\$ docker exec -it 5591aa3b5fff bash**

**[root@server1 ~]# docker exec -it fc510a073ca7 bash**

**[root@fc510a073ca7 tomcat]# ps aux|grep java**



## On Windows:

For below have used windows 10: And docker version 18.03.1-CE

The first step is to get the acs\_deployment module for alfresco 6.0 from the path  
git clone <https://github.com/Alfresco/acs-deployment.git>

and clone it to a directory – the same way done for UNIX.

You can check the docker and docker compose versions on command prompt , but docker compose up command may not run on command prompt – giving unauthorized issue.

```
C:\>docker version
Client:
Version:      18.03.1-ce
API version:  1.37
Go version:   go1.9.5
Git commit:   9e9f40
Built:        Thu Apr 26 07:12:48 2018
OS/Arch:     windows/amd64
Experimental: false
Orchestrator: swarm

Server:
Engine:      18.03.1-ce
Version:     18.03.1-ce
API version: 1.37 (minimum version 1.12)
Go version:  go1.9.5
Git commit:  9e9f40
Built:       Thu Apr 26 07:22:38 2018
OS/Arch:    linux/amd64
Experimental: false

C:\>docker-compose --version
docker-compose version 1.21.1, build 7641a569

C:\>cd github

C:\github>git clone https://github.com/Alfresco/acs-deployment.git
Cloning into 'acs-deployment'...
remote: Counting objects: 944, done.
remote: Compressing objects: 100% (26/26), done.
remote: Total 944 (delta 10), reused 6 (delta 2), pack-reused 915
Receiving objects: 100% (944/944), 308.92 KiB | 1.13 MiB/s, done.
Resolving deltas: 100% (584/584), done.

C:\github>cd acs-deployment

C:\github\acs-deployment>docker-compose up
```

It would work when you logged into docker account and using the docker hub login credentials in kinematics powershell as below:

```
C:\WINDOWS\system32\WindowsPowerShell\v1.0\powershell.exe
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Ranjeet\Kinematic-Windows> cd \github
PS C:\github> cd .\acs-deployment\
PS C:\github\acs-deployment> cd .\docker-compose\
PS C:\github\acs-deployment\docker-compose> docker-compose up
Pulling alfresco_alfresco_alfresco-content-repository:6.1.0-EAL...
6.1.0-EAL: Pulling from alfresco/alfresco-content-repository
188eb7c7401: Pull complete
a2bb57b67b9a: Pull complete
0bc0bab4a3d2: Pull complete
10f3da3e4b1c: Pull complete
d1a10398dfb0: Pull complete
f0832a16da99: Pull complete
831f104cb55e: Pull complete
d87d63d0ff1e: Pull complete
c2085a124ffc: Pull complete
011ed34f985: Pull complete
9919907dc10: Pull complete
4d2198eb92ea: Pull complete
fad7e82ee8ec: Pull complete
acc718af4f3: Pull complete
c69b7729a899: Pull complete
018bbac432f1: Pull complete
57c8895172b2: Pull complete
9b762589c668: Pull complete
077cc3d1e23: Pull complete
a0202f632e06: Downloading [=====] 58.46MB/90.8MB
```

It would take around 30 mins -60 mins to download all dependencies and would create docker containers as below:

```
C:\WINDOWS\System32\WindowsPowerShell\v1.0\powershell.exe
d581943fe8e7: Pull complete
e5ff8940b0: Pull complete
462460a36b09: Pull complete
135fab69c139: Pull complete
Digest: sha256:3f4441460029e12905a5d447a3549ae2ac13323d04539190c8b0c8b48ea17463
Status: Downloaded newer image for postgres:10.1
Pulling solr6 (alfresco/alfresco-search-services:1.1.1)...
1.1.1: pulling from alfresco/alfresco-search-services
6fd4836e300: Already exists
6ff6188dadfc: Already exists
d1616fb0228: Already exists
fed33e3c39c: Already exists
96f2431a7b31: Pull complete
6r41ee0720: Pull complete
e2f85f839b3d: Pull complete
Digest: sha256:b07b746355ce0058becc49aed952b3f770213e55b82c119d0eca95e70c7d8
Status: Downloaded newer image for alfresco/alfresco-search-services:1.1.1
Creating docker-compose_solr6_1 ... done
Creating docker-compose_imagmagick_1 ... done
Creating docker-compose_alfresco-pdf-renderer_1 ... done
Creating docker-compose_tika_1 ... done
Creating docker-compose_postgres_1 ... done
Creating docker-compose_alfresco_1 ... done
Creating docker-compose_libreoffice_1 ... done
Creating docker-compose_share_1 ... done
Attaching to docker-compose_alfresco_1, docker-compose_tika_1, docker-compose_alfresco-pdf-renderer_1, docker-compose_share_1, docker-compose_solr6_1, docker-compose_imagmagick_1, docker-compose_postgres_1, docker-compose_alfresco_1
Replace 'REPO_HOST' with 'alfresco' and 'REPO_PORT' with '8080'
NEW -csrf.filter.referer is ''
NEW -csrf.filter.origin is ''
/var/local/tomcat/shared/classes/alfresco/substituter.sh: line 19: [: !: unary operator expected
The files belonging to this database system will be owned by user "postgres".
This user must also own the server process.

The database cluster will be initialized with locale "en_US.utf8".
The default database encoding has accordingly been set to "UTF8".
The default text search configuration will be set to "english".

Data page checksums are disabled.

fixing permissions on existing directory /var/lib/postgresql/data ... ok
creating subdirectories ... ok
selecting default max_connections ... 100
selecting default shared_buffers ... 128MB
selecting dynamic shared memory implementation ... posix
creating configuration files ... ok
running bootstrap script ... ok
performing post-bootstrap initialization ... ok
syncing data to disk ... ok

Success. You can now start the database server using:

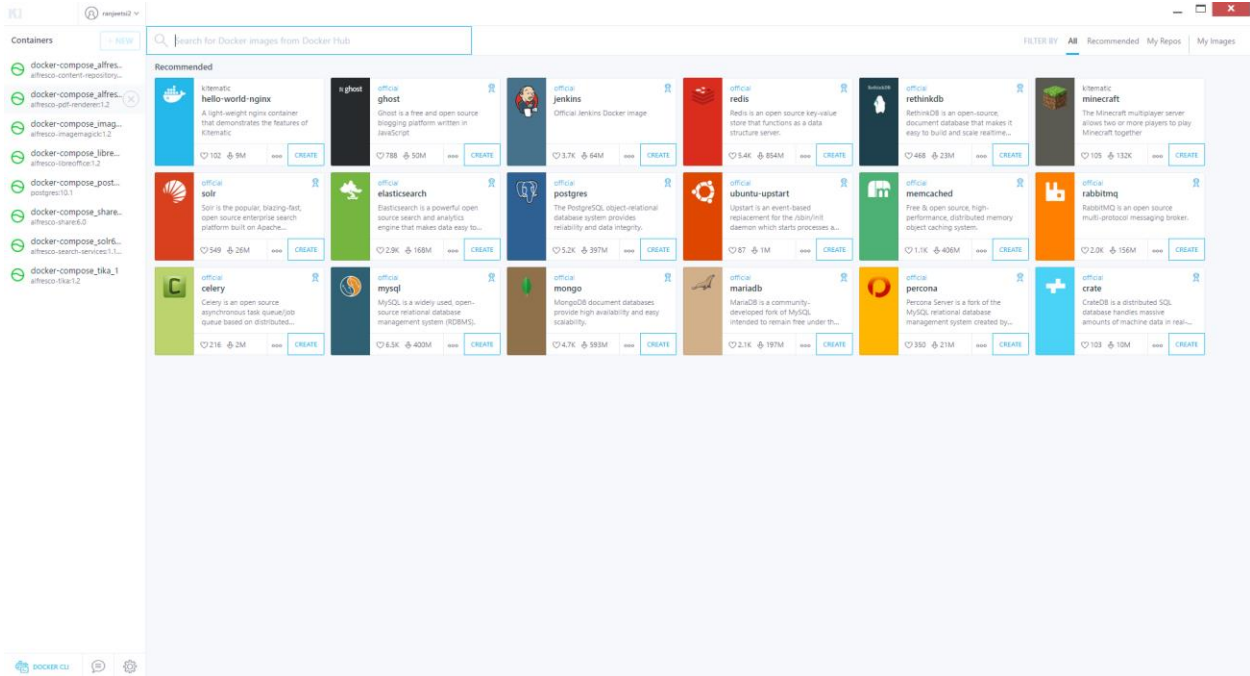
    pg_ctl -D /var/lib/postgresql/data -l logfile start

WARNING: enabling "trust" authentication for local connections
You can change this by editing pg_hba.conf or using the option -A, or
--auth-local and --auth-host, the next time you run initdb.
waiting for server to start..2018-07-14 16:14:15.937 UTC [40] LOG: listening on IPv4 address "127.0.0.1", port 5432
2018-07-14 16:14:15.938 UTC [40] LOG: could not bind IPv6 address "::1": cannot assign requested address
2018-07-14 16:14:15.938 UTC [40] HINT: Is another postmaster already running on port 5432? If not, wait a few seconds and retry.
2018-07-14 16:14:16.220 UTC [40] LOG: listening on Unix socket "/var/run/postgresql/.s.PGSQL.5432"
2018-07-14 16:14:16.665 UTC [41] LOG: database system was shut down at 2018-07-14 16:14:12 UTC
```

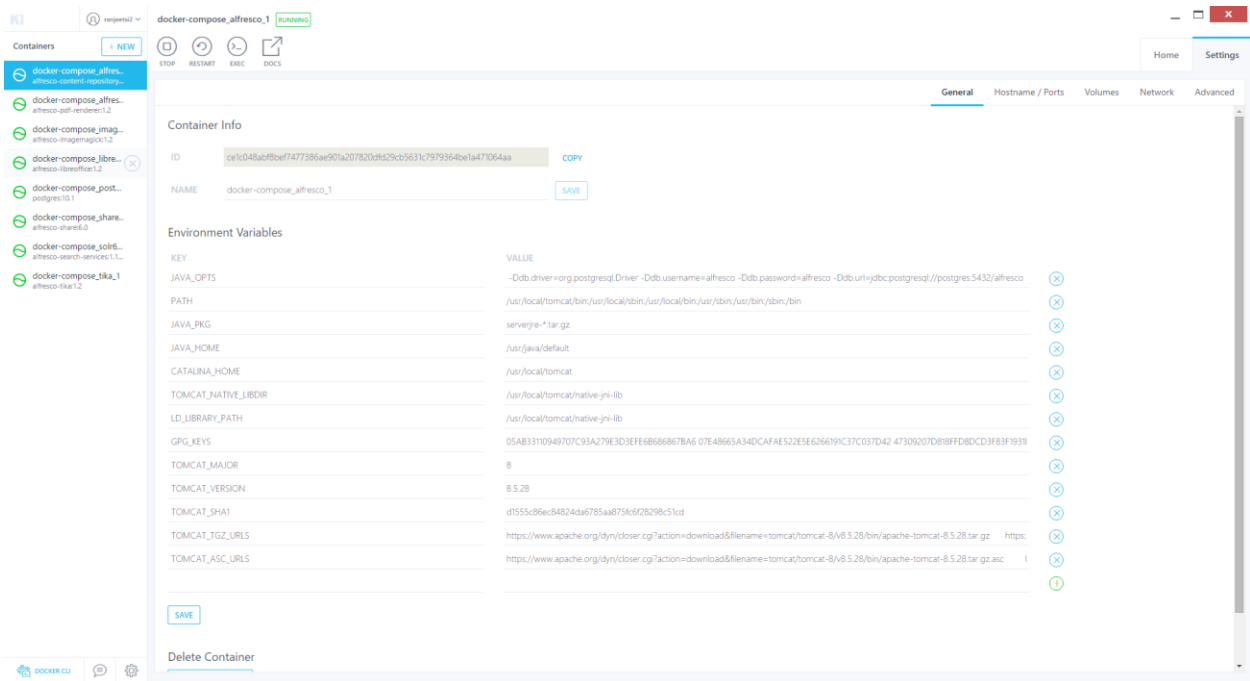
The individual docker container logs can be viewed via the kinematics dashboard as below:

The screenshot shows the Docker Desktop interface. On the left, a sidebar lists several containers: 'docker-compose\_alfresco-pdf-renderer\_1', 'docker-compose\_alfresco-pdf-renderer\_1', 'docker-compose\_imagmagick\_1' (selected), 'docker-compose\_alfresco-share\_1', 'docker-compose\_alfresco-share\_1', 'docker-compose\_solr6\_1', 'docker-compose\_solr6\_1', and 'docker-compose\_tika\_1'. The main area displays the logs for 'docker-compose\_imagmagick\_1'. The logs show the container starting successfully, with various system messages and application startup logs. The logs include information about the Spring Boot application, the application's profile, and the initialization of the database. The logs also show the application's startup time and the port it is running on.

When all the docker containers gets started – all docker containers would be in running as below.



Can check all settings as below for individual containers:



This runs alfresco at <http://localhost:8082/alfresco>

share application at <http://localhost:8080/share>

solr application at <http://localhost:8083/solr>



The screenshot displays the Solr Admin interface. On the left, there's a navigation menu with options like Dashboard, Logging, Core Admin, Java Properties, and Thread Dump. The main area shows the 'Instance' details, including its start time (7 minutes ago), versions (solr-spec 6.3.0, solr-impl 6.3.0, lucene-spec 6.3.0, lucene-impl 6.3.0), and JVM configuration. The JVM configuration lists various parameters such as -DSTOP.KEY=solrlocks, -Djetty.home, -Dsolr.install.dir, and -Duser.timezone=UTC. On the right, there are system resource usage graphs for Physical Memory (98.2%), Swap Space (0.0%), File Descriptor Count (0.0%), and JVM-Memory (17.3%).

The containers can elegantly be stopped using Cntrl+C command. And forcefully by again hitting Cntrl+C

```

failed)
al fresco_1 | 2018-07-14 22:29:19,775 WARN [scripts.servlet.X509ServletFilterBase] [localhost-startStop-1] clientAuth does not appear to be set for Tomcat. clientAuth must be
set to 'want' for X509 Authentication
al fresco_1 | 2018-07-14 22:29:19,775 WARN [scripts.servlet.X509ServletFilterBase] [localhost-startStop-1] Attempting to set clientAuth=want through JMX...
al fresco_1 | 2018-07-14 22:29:24,872 INFO [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory [/usr/local/to
ncat/webapps/al fresco] has finished in [182,153] ms
al fresco_1 | 14-Jul-2018 22:29:24,884 INFO [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["http-nio-8080"]
al fresco_1 | 14-Jul-2018 22:29:24,905 INFO [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["ajp-nio-8009"]
al fresco_1 | 14-Jul-2018 22:29:24,941 INFO [main] org.apache.catalina.startup.Catalina.start Server startup in 186057 ms
al fresco_1 | 2018-07-14 22:29:50,659 INFO [management.subsystems.ChildApplicationContextFactory] [http-nio-8080-exec-4] Starting 'Transformers' subsystem, ID: [Transformers
, default]
al fresco_1 | 2018-07-14 22:29:50,893 INFO [management.subsystems.ChildApplicationContextFactory] [http-nio-8080-exec-4] Startup of 'Transformers' subsystem, ID: [Transformers
, default] complete
al fresco_1 | 2018-07-14 22:29:52,139 INFO 1 --- [nio-8090-exec-3] org.al fresco.transformer.TikaController : Probe: livenessPercent=400
al fresco_1 | 2018-07-14 22:29:52,139 INFO 1 --- [nio-8090-exec-3] org.al fresco.transformer.TikaController : Probe: maxTransforms=10240
al fresco_1 | 2018-07-14 22:29:52,139 INFO 1 --- [nio-8090-exec-3] org.al fresco.transformer.TikaController : Probe: maxTransformsSeconds=1801
al fresco_1 | 2018-07-14 22:29:52,139 INFO 1 --- [nio-8090-exec-3] org.al fresco.transformer.TikaController : Probe: livenessTransformPeriodSeconds=920
al fresco_1 | 2018-07-14 22:29:53,988 DEBUG 1 --- [nio-8090-exec-4] org.al fresco.transformer.TikaController : 1 22:29:52 200 1.8s (75ms 1.4s 389ms) doc 72 KB txt 3.8 KB TextMin
ing --targetMimetype=text/plain --targetEncoding=UTF-8 doc txt Success
al fresco_1 | 2018-07-14 22:29:53,991 DEBUG 1 --- [nio-8090-exec-3] org.al fresco.transformer.TikaController : 3 22:29:52 200 1.8s (75ms 1.4s 386ms) doc 72.5 KB txt 3.9 KB TextMin
ing --targetMimetype=text/plain --targetEncoding=UTF-8 doc txt Success
al fresco_1 | 2018-07-14 22:29:53,991 DEBUG 1 --- [nio-8090-exec-5] org.al fresco.transformer.TikaController : 2 22:29:52 200 1.8s (65ms 1.4s 393ms) doc 72 KB txt 3.8 KB TextMin
ing --targetMimetype=text/plain --targetEncoding=UTF-8 doc txt Success
al fresco_1 | 2018-07-14 22:29:54,357 DEBUG 1 --- [nio-8090-exec-6] org.al fresco.transformer.TikaController : 4 22:29:52 200 2s (16ms 2s 34ms) ppt 2 MB txt 1.3 KB TikaAuto --ta
rgetMimetype=text/plain --targetEncoding=UTF-8 ppt txt Success
al fresco_1 | 2018-07-14 22:29:54,366 DEBUG 1 --- [nio-8090-exec-1] org.al fresco.transformer.TikaController : 7 22:29:54 200 286ms (5ms 275ms 6ms) ppt 2.7 MB txt 1 KB TikaAuto
--targetMimetype=text/plain --targetEncoding=UTF-8 ppt txt Success
al fresco_1 | 2018-07-14 22:29:54,776 DEBUG 1 --- [nio-8090-exec-8] org.al fresco.transformer.TikaController : 6 22:29:53 200 806ms (1ms 803ms 2ms) xls 25.5 KB txt 596bytes Tika
Auto --targetMimetype=text/plain --targetEncoding=UTF-8 xls txt Success
al fresco_1 | 2018-07-14 22:29:55,133 DEBUG 1 --- [nio-8090-exec-7] org.al fresco.transformer.TikaController : 5 22:29:53 200 2.1s (1ms 2.1s 2ms) pdf 372.8 KB txt 12.2 KB Pdfbox
--targetMimetype=text/plain --targetEncoding=UTF-8 pdf txt Success
al fresco_1 | 2018-07-14 22:32:58,868 INFO [web.site.EditionInterceptor] [http-nio-8080-exec-5] Successfully retrieved license information from Alfresco.
al fresco_1 | 2018-07-14 22:33:48,473 INFO [web.scripts.ImapServerStatus] [http-nio-8080-exec-2] Successfully retrieved IMAP server status from Alfresco: disabled
al fresco_1 | 2018-07-14 22:33:48,786 INFO [web.scripts.SyncModeConfig] [http-nio-8080-exec-2] Successfully retrieved sync mode configuration from Alfresco: OFF
Gracefully stopping... (press Ctrl+C again to force)
Stopping docker-compose_al fresco_1 ... done
Stopping docker-compose_al fresco-pdf-renderer_1 ... done
Stopping docker-compose_tika_1 ... done
Stopping docker-compose_imagick_1 ... done
Stopping docker-compose_postgres_1 ... done
Stopping docker-compose_share_1 ... done
Stopping docker-compose_solr6_1 ... done
Stopping docker-compose_1libreoffice_1 ... done
PS C:\github\acs-deployment\docker-compose>

```