

How to Run Fedora4 in AWS

Start AWS Instance

Puppet Setup

1. Locally clone the [fcrepo-aws-puppet](#) project
2. Follow the [README](#) from the fcrepo-aws-puppet project
3. There are two configurations to choose from:
 - a. Single node, async-indexing (default)
 - i. Includes YourKit agent for remote profiling
 - b. Clustered

Clustered Configuration

To run a clustered configuration, locally update **fcrepo-aws-puppet/modules/tomcat7/manifests/init.pp** uncommenting the clustered **\$repo_config** and **\$etc_default_tomcat** elements, and commenting out the previous values of those elements.

- From

```
$repo_config = 'classpath:/config/async-indexing/repository.json',
#   $repo_config = 'classpath:/config/clustered/repository.json',
$etc_default_tomcat = 'tomcat7/default-tomcat7.erb',
#   $etc_default_tomcat = 'tomcat7/default-tomcat7-clustered.erb',
```

- To

```
#   $repo_config = 'classpath:/config/async-indexing/repository.json',
$repo_config = 'classpath:/config/clustered/repository.json',
#   $etc_default_tomcat = 'tomcat7/default-tomcat7.erb',
$etc_default_tomcat = 'tomcat7/default-tomcat7-clustered.erb',
```

Note, for the clustered configuration, start as many instances as desired.

- All clustered instances will automatically join the same cluster

Puppet Run

Run

```
./fcrepo-aws-puppet/cloud-init/create-ec2-instance.sh
```

Clustered Options

Fedora 4 can be run in a clustered configuration on AWS. We ran performance benchmarks using these additional java opts:

```
###
# Clustering
###
JAVA_OPTS="${JAVA_OPTS} -Dfcrepo.infinispan.cache_configuration=config/infinispan/clustered/infinispan.xml"
JAVA_OPTS="${JAVA_OPTS} -Djgroups.tcp.address=<aws-private-ip>"
JAVA_OPTS="${JAVA_OPTS} -Dfcrepo.ispn.numOwners=2 -Djava.net.PreferIPv4Stack=true"

# The jgroups-ec2.xml file is included in ispn's jars
JAVA_OPTS="${JAVA_OPTS} -Dfcrepo.ispn.jgroups.configuration=jgroups-ec2.xml"

# This property overwrites the S3 bucketname variable in jgroups-ec2.xml
JAVA_OPTS="${JAVA_OPTS} -Djgroups.s3.bucket=fcrepo4-cluster-0"
```

Most importantly, we used the jgroups-ec2 configuration, which enables auto-discovery of nodes by writing content into an S3 bucket.

JMeter

...coming