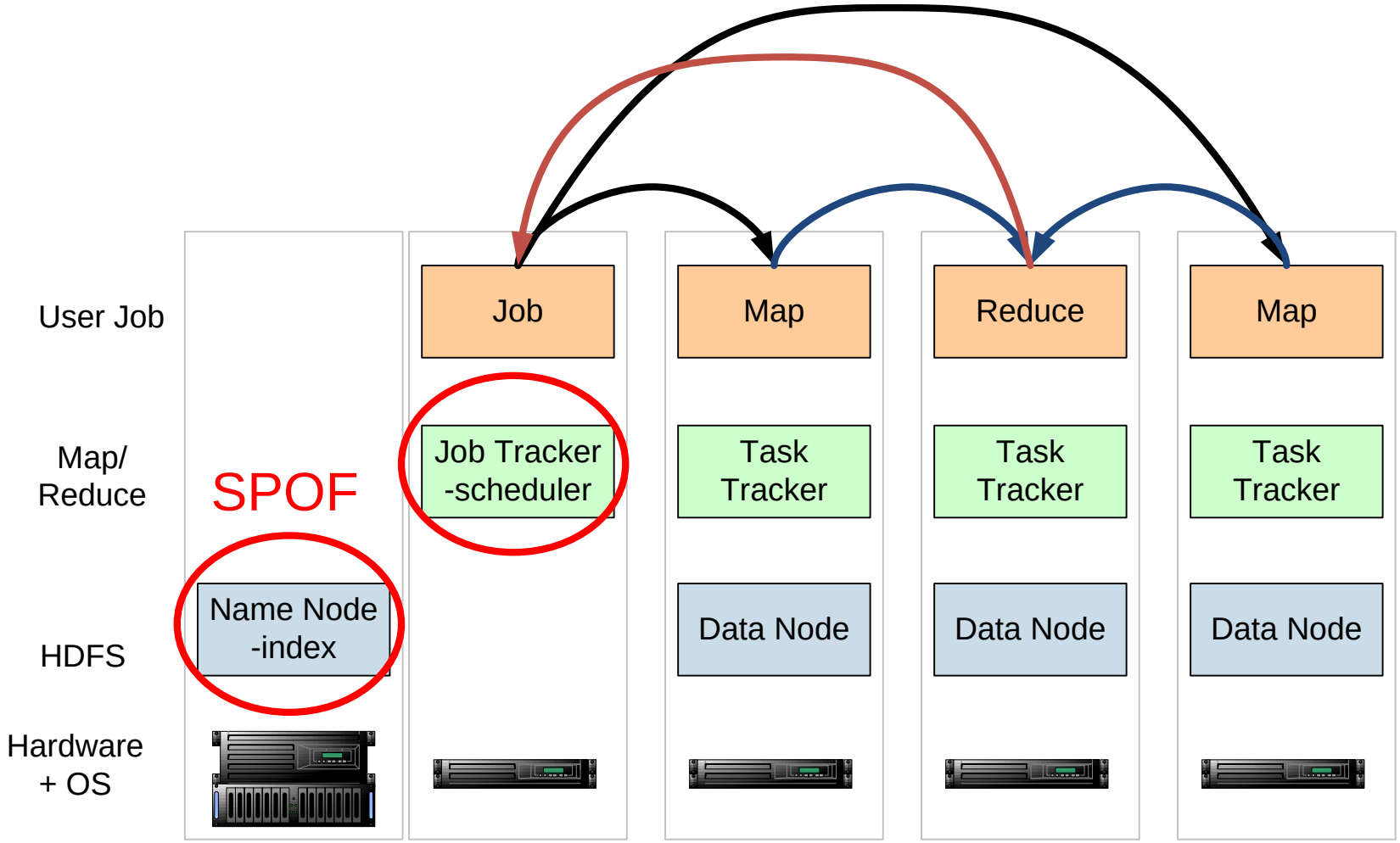
Dynamic Hadoop Clusters

Steve Loughran
Julio Guijarro

LABS hp
Automated Infrastructure Lab


My other computer
is a data center

Hadoop on a cluster



1 namenode, 1+ Job Tracker, many data nodes and task trackers

Cluster design

- Name node: high end box, RAID + backups.
  -this is the SPOF. Nurture it.
- Secondary name node —*as name node*
- Data nodes: mid-range multicore blade systems
  2 disks/core. No RAID.
- Job tracker: standalone server
- task trackers: on the data servers
- Secure the LAN
- Everything will fail -learn to read the logs

Management problems big applications

1. Configuration
2. Lifecycle
3. Troubleshooting

The hand-managed cluster

- Manual install onto machines
- SCP/FTP in Hadoop tar file
- Edit the -site.xml and log4j files
- edit /etc/hosts, /etc/rc5.d, ssh keys …

- Installation scales O(N)
- Maintenance, debugging scales worse

Do not try this more than once

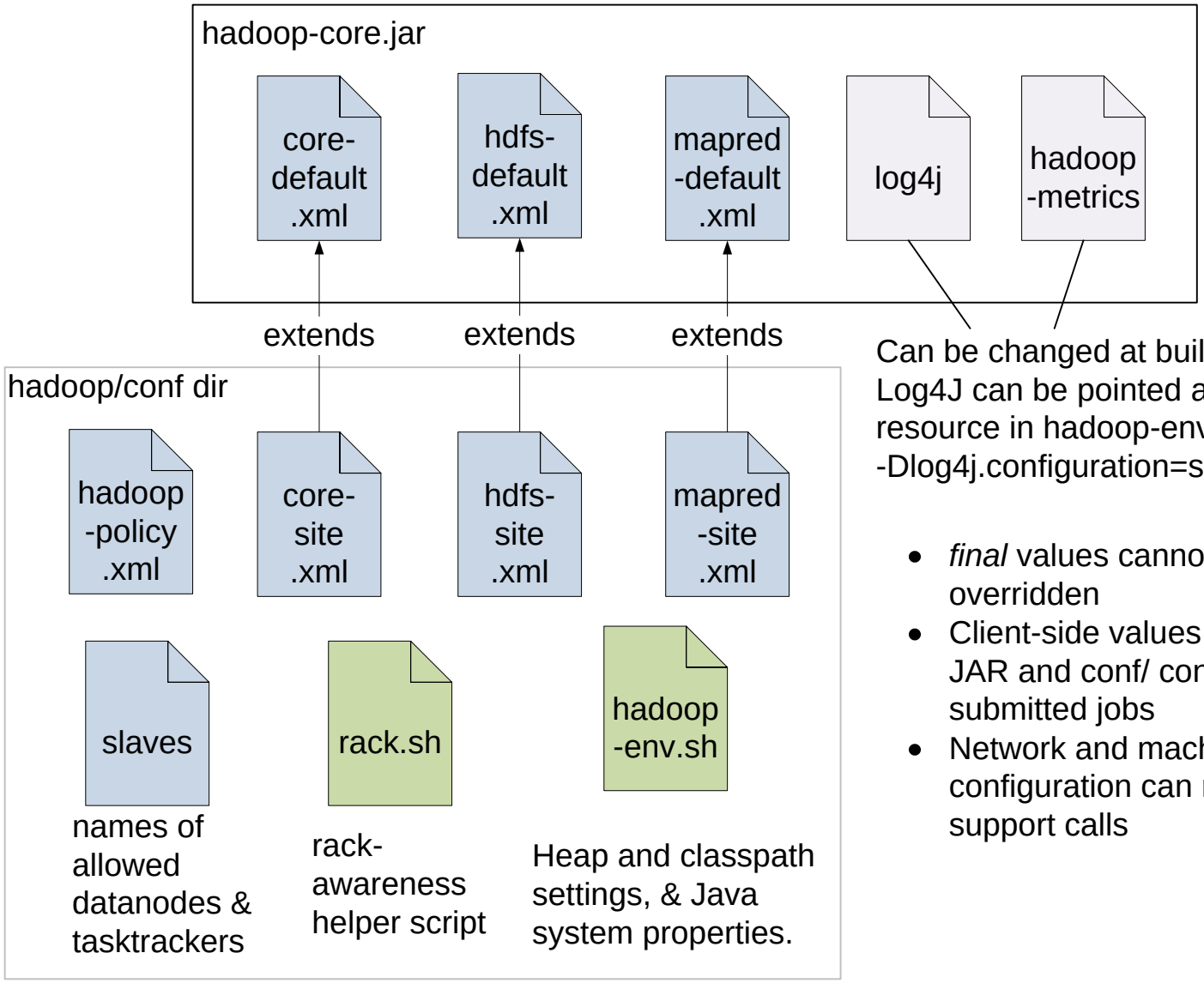
The locked-down cluster

- PXE/gPXE Preboot of OS images
- RedHat Kickstart to serve up (see instalinux.com)
- Maybe: LDAP to manage state
- Chukwa for log capture/analysis

uniform images, central LDAP service, good ops team, stable configurations, home-rolled RPMs

How Yahoo! work?

How do you configure Hadoop 0.21?

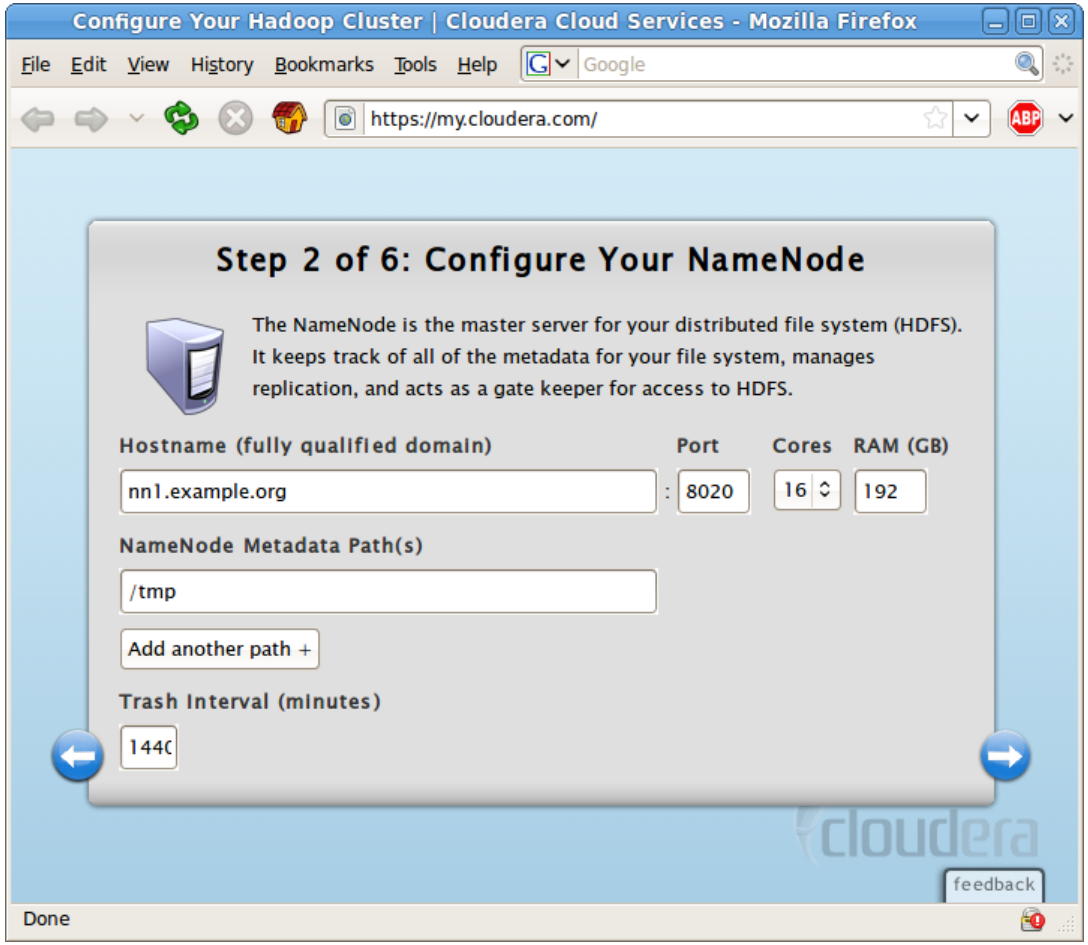


Can be changed at build time
Log4J can be pointed at new resource in hadoop-env.sh:
-Dlog4j.configuration=site2.xml

- *final* values cannot be overridden
- Client-side values in JAR and conf/ configure submitted jobs
- Network and machine configuration can raise support calls

cloudera

cloudera.com/hadoop



- RPM-packaged Hadoop distributions
- Web UI creates configuration RPMs
- Configurations managed with "alternatives"

Applications Places System 3:39 PM

smartfrog@dundee:/home/smartfrog

File Edit View Terminal Tabs Help

```
[root@dundee smartfrog]# /sbin/service hadoop-datanode start
Starting Hadoop datanode daemon (hadoop-datanode): starting datanode, logging to
 /var/log/hadoop/hadoop-hadoop-datanode-dundee.out
09/03/04 15:37:02 INFO dfs.DataNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting DataNode
STARTUP_MSG:   host = dundee/0:0:0:0:0:0:0:1
STARTUP_MSG:   args = []
STARTUP_MSG:   version = 0.18.3-CH-0.3.0
STARTUP_MSG:   build =  -r ; compiled by 'matt' on Wed Feb 25 14:54:00 PST 2009
************************************************************/
                                                           [  OK  ]
[root@dundee smartfrog]#
```

smartfrog@dundee:/home/smartfrog

Configuration in RPMs

Applications Places System 2:00 PM

root

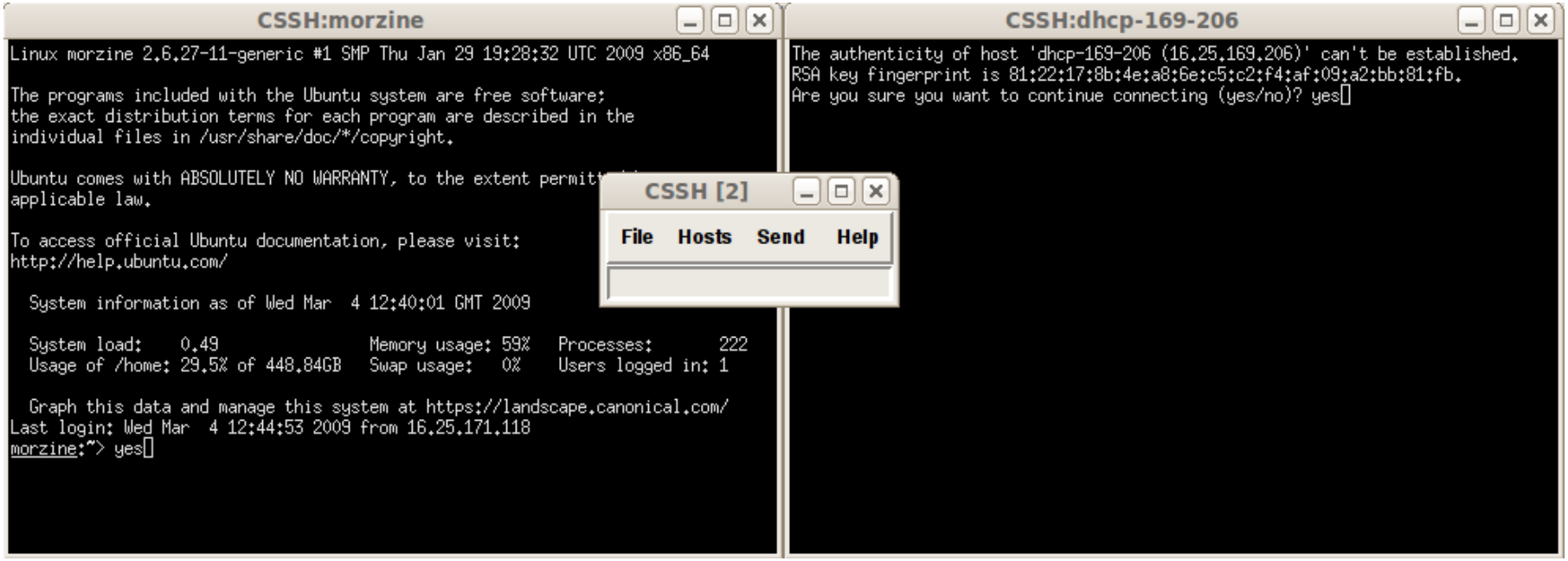
File Edit View Terminal Tabs Help

```
[root@dundee hadoop]#
[root@dundee hadoop]# ls -al
total 36
drwxr-xr-x  3 root   root    4096 Mar  4 14:25 .
drwxr-xr-x 97 root   root   12288 Mar 12 16:23 ..
lrwxrwxrwx  1 root   root      24 Mar  4 14:25 conf -> /etc/alternatives
/hadoop
drwxr-xr-x  2 hadoop hadoop  4096 Mar  4 14:24 conf.empty
[root@dundee hadoop]# rpm -qf conf/hadoop-site.xml
hadoop-0.18.3-1.cloudera.CH0_3
[root@dundee hadoop]#
```

+Push out, rollback via kickstart.

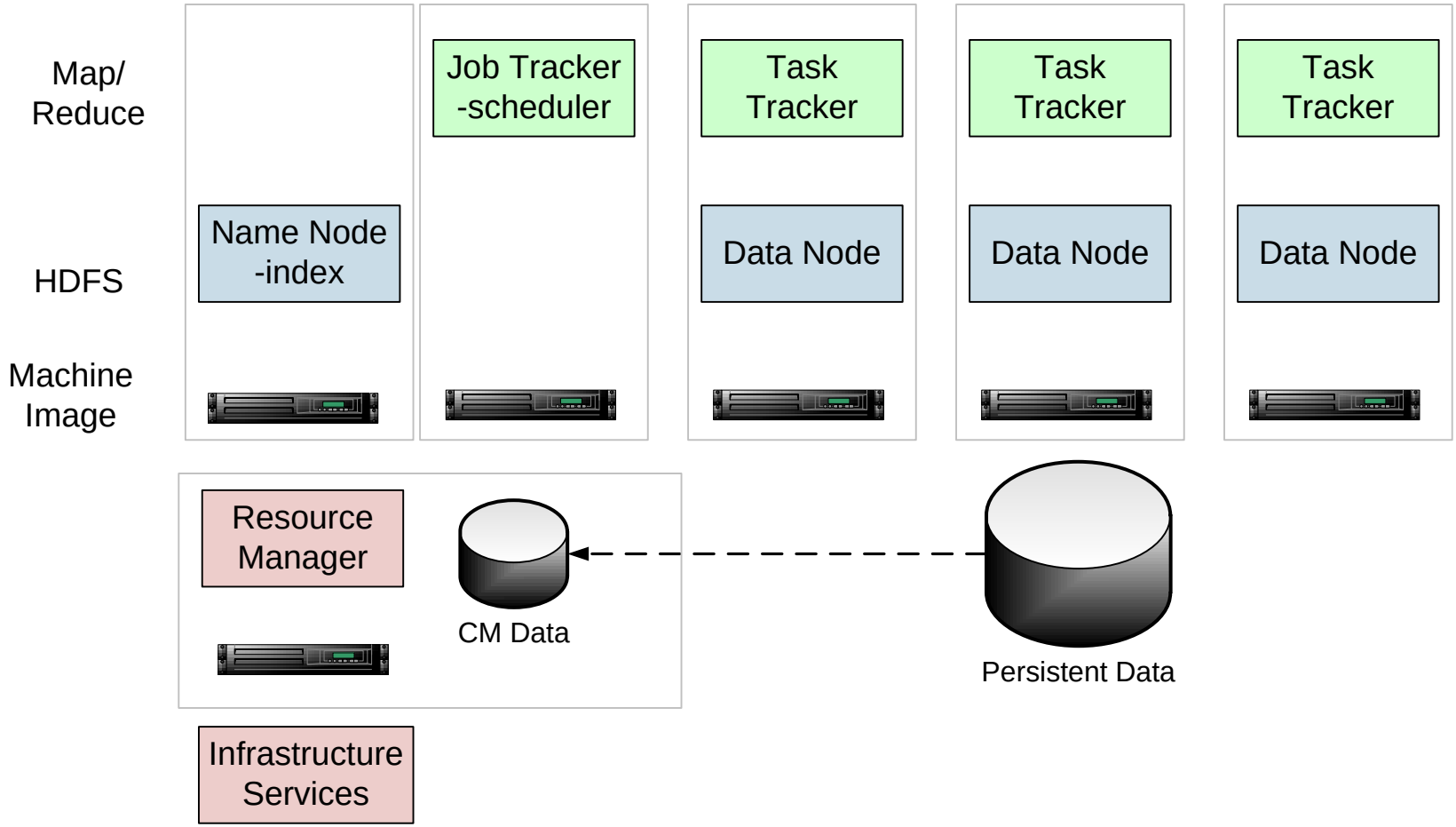
- Extra build step, may need kickstart server

clusterssh: cssh



If all the machines start in the same initial state, they should end up in the same exit state

CM-Managed Hadoop



Resource Manager keeps cluster live; talks to infrastructure

Persistent data store for input data and results

Configuration Management tools

	State Driven	Workflow
Centralized	Radia, ITIL, lcfg	Puppet
Decentralized	bcfg2, SmartFrog	Perl scripts, makefiles

CM tools are how to manage big clusters

SmartFrog - HPLabs' CM tool

- Language for describing systems to deploy —everything from datacentres to test cases
- Runtime to create *components* from the model
- Components have a *lifecycle*
- Apache 2.0 Licensed from May 2009
- http://smartfrog.org/

Model the system in the SmartFrog language

extending an existing template

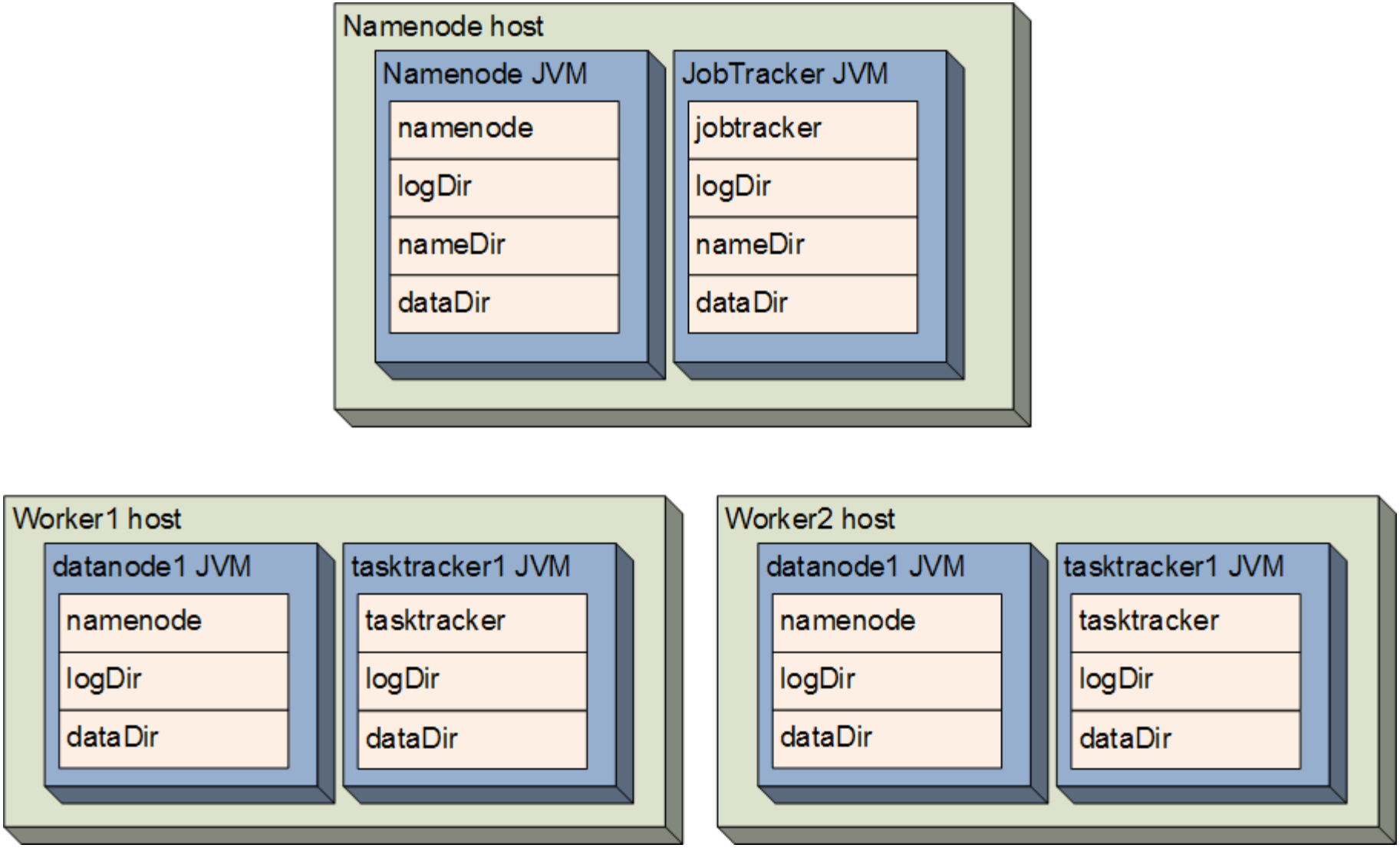
```
TwoNodeHDFS extends OneNodeHDFS {

  localDataDir2 extends TempDirWithCleanup {

  }

  datanode2 extends datanode {
    dataDirectories [LAZY localDataDir2];
    dfs.datanode.https.address "https://0.0.0.0:8020";
  }
}
```

a temporary directory component

extend and override with new values, including a reference to the temporary directory

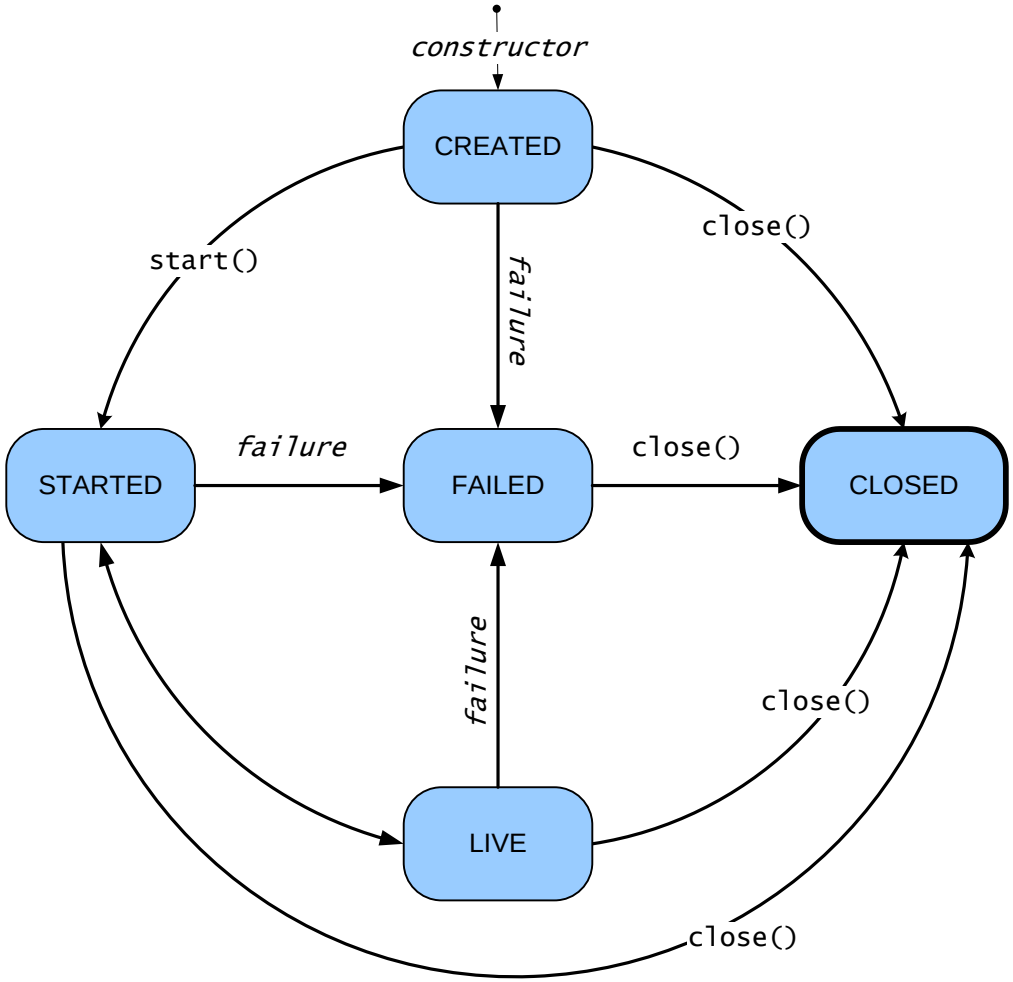
Inheritance, cross-referencing, templating

The runtime deploys the model



DEMO

HADOOP-3628: A lifecycle for services



Base Service class for all nodes

```
public class Service extends Configured implements Closeable {
  public void start() throws IOException;
 public void innerPing(ServiceStatus status)
                  throws IOException;
  void close() throws IOException;
  State getLifecycleState();
  public enum State {
    UNDEFINED,
    CREATED,
    STARTED,
    LIVE,
    FAILED,
    CLOSED
  }
}
```

Subclasses implement transitions

```
public class NameNode extends Service implements
  ClientProtocol, NamenodeProtocol, ... {

protected void innerStart() throws IOException {
  initialize(bindAddress, getConf());
  setServiceState(ServiceState.LIVE);
}

public void innerClose() throws IOException {
  if (server != null) {
    server.stop();
    server = null;
  }
  ...
}

}
```

Health and Liveness: ping()

```
public class DataNode extends Service {
...
  public void innerPing(ServiceStatus status)
                                throws IOException {
    if (ipcServer == null) {
      status.addThrowable(
       new LivenessException("No IPC Server running"));
    }
    if (dnRegistration == null) {
      status.addThrowable(
       new LivenessException("Not bound to a namenode"));
    }
  }
```

Ping issues

- If a datanode cannot see a namenode, is it still healthy?
- If a namenode has no data nodes, is it healthy?
- How to treat a failure of a ping? Permanent failure of service, or a transient outage?

How unavailable should the nodes be before a cluster is "unhealthy"?

Replace hadoop-*.xml with .sf files

```
NameNode extends FileSystemNode {
  nameDirectories TBD;
  dataDirectories TBD;
  logDir TBD;
  dfs.http.address "http://0.0.0.0:8021";
  dfs.namenode.handler.count 10;
  dfs.namenode.decommission.interval (5 * 60);
  dfs.name.dir TBD;
  dfs.permissions.supergroup "supergroup";
  dfs.upgrade.permission "0777"
  dfs.replication 3;
  dfs.replication.interval 3;
  . . .
}
```

Hadoop Cluster under SmartFrog

sfManagementConsole [sfManagementConsole connected to localhost:3800]

Display options Help Mng. Console

Refresh node | Refresh all tabs

rootProcess | cluster | output

- rootProcess
  - sfVersionInfo [tags_erro
  - namenode
  - datanode1
  - jobtracker
  - tasktracker1
  - cluster
    - namenode
    - datanode
      - logDir
      - localDataDir
      - datanode
      - workerHost
    - jobTracker
    - taskTracker

Attribute	Value	...
dfs.datanode.rack	/default-rack	..
dfs.socket.timeout	60000	..
dfs.datanode.simulatedda...	false	..
dfs.datanode.startup	"REGULAR"	..
hostname	"0.0.0.0"	..
logDir	LAZY PARENT:logDir	..
sfLivenessDelay	15L	..
sfLivenessFactor	2	..
sfHost	morzine/127.0.0.1	..
sfProcess	"datanode1"	..
sfLog	"HOST localhost:rootPro...	..
live.dfs.datanode.address	"http://0.0.0.0:8042/"	..
live.dfs.datanode.http.a...	"http://127.0.0.1:8030/"	..

```
* Attribute: live.dfs.datanode.address
 * Tags: []
 * Value:
"http://0.0.0.0:8042/"
 * Value resolved:
```

HOST localhost:rootProcess:cluster:datanode:datanode

Aggregated logs

```
17:39:08 [JobTracker] INFO mapred.ExtJobTracker : State change: JobTracker is now LIVE
17:39:08 [JobTracker] INFO mapred.JobTracker : Restoration complete
17:39:08 [JobTracker] INFO mapred.JobTracker : Starting interTrackerServer
17:39:08 [IPC Server Responder] INFO ipc.Server : IPC Server Responder: starting
17:39:08 [IPC Server listener on 8012] INFO ipc.Server : IPC Server listener on 8012:
  starting
17:39:08 [JobTracker] INFO mapred.JobTracker : Starting RUNNING
17:39:08 [Map-events fetcher for all reduce tasks on
  tracker_localhost:localhost/127.0.0.1:34072] INFO mapred.TaskTracker : Starting thread:
  Map-events fetcher for all reduce tasks on tracker_localhost:localhost/127.0.0.1:34072
17:39:08:960 GMT [INFO ][TaskTracker] HOST localhost:rootProcess:cluster - TaskTracker
  deployment complete: service is: tracker_localhost:localhost/127.0.0.1:34072 instance
  org.apache.hadoop.mapred.ExtTaskTracker@8775b3a in state STARTED; web port=50060
17:39:08 [TaskTracker] INFO mapred.ExtTaskTracker : Task Tracker Service is being offered:
  tracker_localhost:localhost/127.0.0.1:34072 instance
  org.apache.hadoop.mapred.ExtTaskTracker@8775b3a in state STARTED; web port=50060
17:39:09 [IPC Server handler 5 on 8012] INFO net.NetworkTopology : Adding a new node:
  /default-rack/localhost
17:39:09 [TaskTracker] INFO mapred.ExtTaskTracker : State change: TaskTracker is now LIVE
```

File and Job operations

```
TestJob extends BlockingJobSubmitter {
  name "test-job";
  cluster LAZY PARENT:cluster;
  jobTracker LAZY PARENT:cluster;
  mapred.child.java.opts "-Xmx512m";
  mapred.tasktracker.map.tasks.maximum 5;
  mapred.tasktracker.reduce.tasks.maximum 1;
  mapred.map.max.attempts 1;
  mapred.reduce.max.attempts 1;
}
```

DFS manipulation: DfsCreateDir, DfsDeleteDir, DfsListDir, DfsPathExists, DfsFormatFileSystem,

DFS I/O: DfsCopyFileIn, DfsCopyFileOut

What does this let us do?

- Set up and tear down Hadoop clusters
- Manipulate the filesystem
- Get a console view of the whole system
- Allow different cluster configurations
- Automate failover policies

Status as of March 2009

- SmartFrog code in sourceforge SVN
- HADOOP-3628 branch patches Hadoop source
  - ready to merge?
- Building RPMs for managing local clusters
- Hosting on VMs
- Submitting simple jobs
- Troublespots: hostnames, Java security, JSP

Not ready for production

Issue: Hadoop configuration

- Trouble: core-site.xml, mapred-site …
- Current SmartFrog support subclasses JobConf
- Better to have multiple sources of configuration
  - XML
  - LDAP
  - Databases
  - SmartFrog

Issue: VM performance

- CPU performance under Xen, VMWare slightly slower
- Disk IO measurably worse than physical
- Startup costs if persistent data kept elsewhere
- VM APIs need to include source data/locality
- Swapping and clock drift causes trouble

Cluster availability is often more important than absolute performance

Issue: binding on a dynamic network

- Discovery on networks without multicast
- Hadoop on networks without reverse DNS
- Need IP address only (no forward DNS)
- What if nodes change during a cluster's life?

Call to action

- Dynamic Hadoop clusters are a good way to explore Hadoop
- Come and play with the SmartFrog Hadoop tools
- Get involved with managing Hadoop
- Help with lifecycle, configuration issues

Come to Thursday's talk :
Cloud Application Architecture

LABS hp

XML in SCM-managed filesystem

conf

File Edit View Terminal Tabs Help

```
$ svn info core-site.xml
Path: core-site.xml
Name: core-site.xml
URL: http://hcluster/svn/hadoop/sites/work-cluster-3/conf/core-site.xml
Repository Root: http://hcluster/svn/
Repository UUID: 13f79535-47bb-0310-9956-ffa450edef68
Revision: 753193
Node Kind: file
Schedule: normal
Last Changed Author: slo
Last Changed Rev: 726883
Last Changed Date: 2009-03-17 00:15:03 +0000 (Tue, 17 Mar 2009)
Text Last Updated: 2009-03-17 00:15:03 +0000 (Tue, 17 Mar 2009)
Checksum: 5248c973f1cd3c22fefd056024434bcbi

$
```

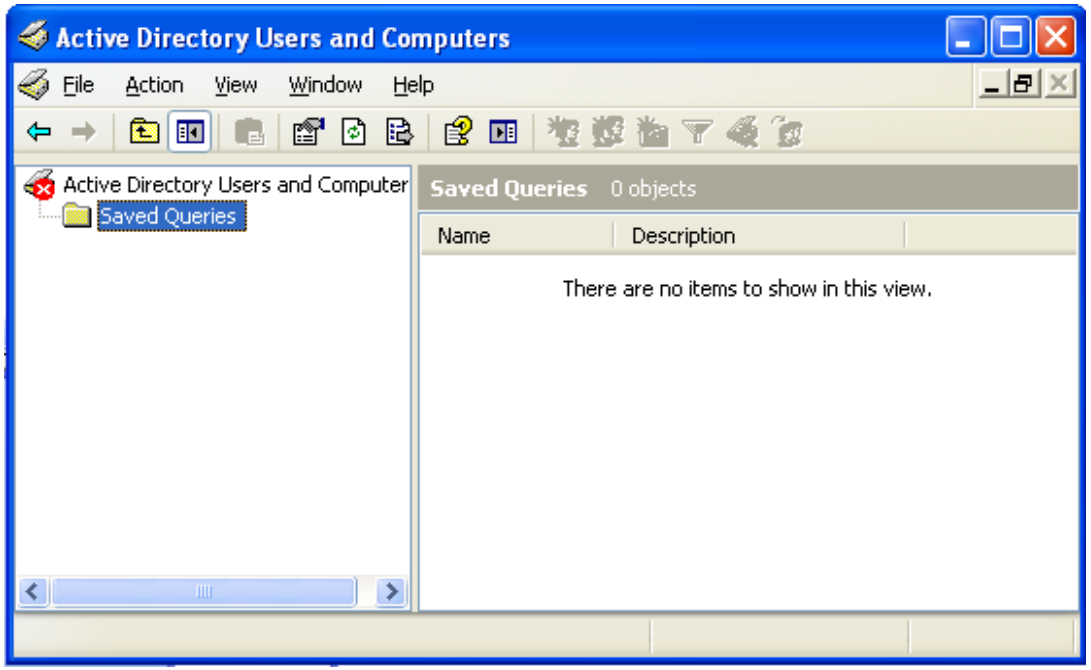
+push out, rollback.

- Need to restart cluster, SPOF?

Configuration-in-database

- JDBC, CouchDB, SimpleDB, …
- Other name-value keystore?
- Bootstrap problem -startup parameters?
- Rollback and versioning?

Configuration with LDAP



+View, change settings; High Availability

- Not under SCM; rollback and preflight hard