# Jenkins

## Maintenance

# Section Contents

- Back Ups
- Archiving Projects
- Discarding Old Builds
- Monitor Server Load

# Back Ups

# Back Ups

- As its simplest form you just need to back up `$JENKINS_HOME`
- Folders that you can ignore
  - `$JENKINS_HOME/war`
  - `$JENKINS_HOME/cache`
  - `$JENKINS_HOME/tools`

# Specific backups

- If you do back up the `$JENKINS_HOME` you can be selective of your jobs
- There is no need to backup the `jobs/workspace` directory
- `builds` directory is important since it contains test results, artifacts, and timestamps, if that isn't necessary do not include it

# Verifying Backups

- `export` or `set` the `JENKINS_HOME` that points to the Jenkins backup
- Run Jenkins in a separate port

# Using the Backup Plugin

- Plugin used to backup Jenkins data
- Configure and run backups of
  - build job configurations
  - build history
- You can choose to shutdown or not to shutdown
- Can be attached to a cron by invoking with `wget` or `curl`: `http://<host>:<port>/backup/backup`

# Thin Backups

- Thin backups can be performed with the Thin Backup Plugin
- Build History and Artifacts are not stored! (It's thin)
- Includes cron settings to time your backups (Backup Plugin does not)
- Contains a restore page to restore the plugin

# Lab: Create a Backup and Restore

- Install the Backup Plugin
- Configure the backup without shutting down the server.
- Create a Backup
- Verify the Backup
- Delete the simple project job
- Restore from the Backup
- Ensure the simple project job is back

# Archiving Projects

# Archiving a Project

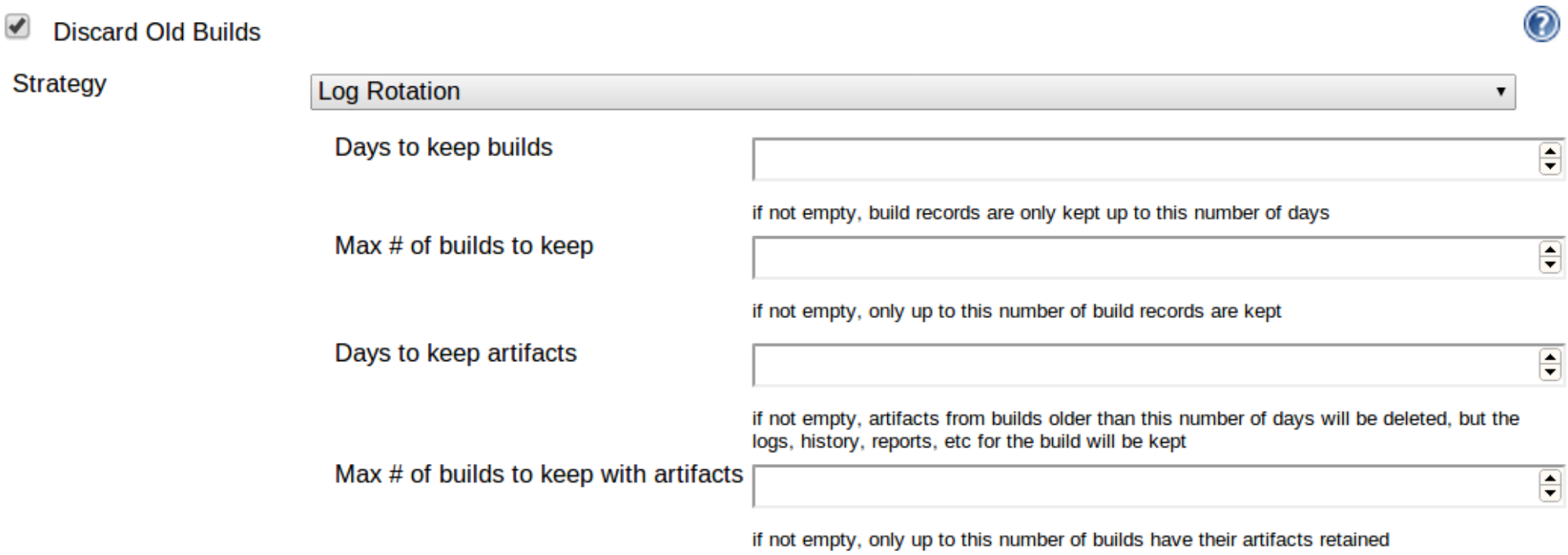
- In a command prompt go to the `$JENKINS_HOME\jobs` directory
- List the contents of the directory using `dir` or `ls`
- `jar`, `tar`, or `zip` the directory of the project
- Copy the archive to any desired backup directory
- Ensure that the project has no active jobs
- Delete the project from the `$JENKINS_HOME\jobs` directory
- 
  Reload Configuration from Disk
  Discard all the loaded data in memory and reload everything from file system. Useful when you modified config files directly on disk.

# Lab: Archive a Single Project

- Archive the `simpleproject` project
- Delete the `simpleproject` project
- Reload the configuration from disk
- Bring back the `simpleproject` by putting it back in the jobs directory
- Reload the configuration from disk

# Discarding Old Builds

# Discarding Old Builds



# Discard Old Builds

- Overtime, the more build records on a job the longer it will take
- Discard Old Builds in each project can save on space
- Days to Keep Builds specifies how many days to keep
- Max # of Builds specifies how many builds to hold onto
- Days to Keep Artifacts specifies how many days to hold onto artifacts
- Max # of Builds to keep artifacts specifies how many builds to keep artifacts
- Warning: Everything will be gone, Keeping Artifacts is the best approach

# Disk Usage Plugin

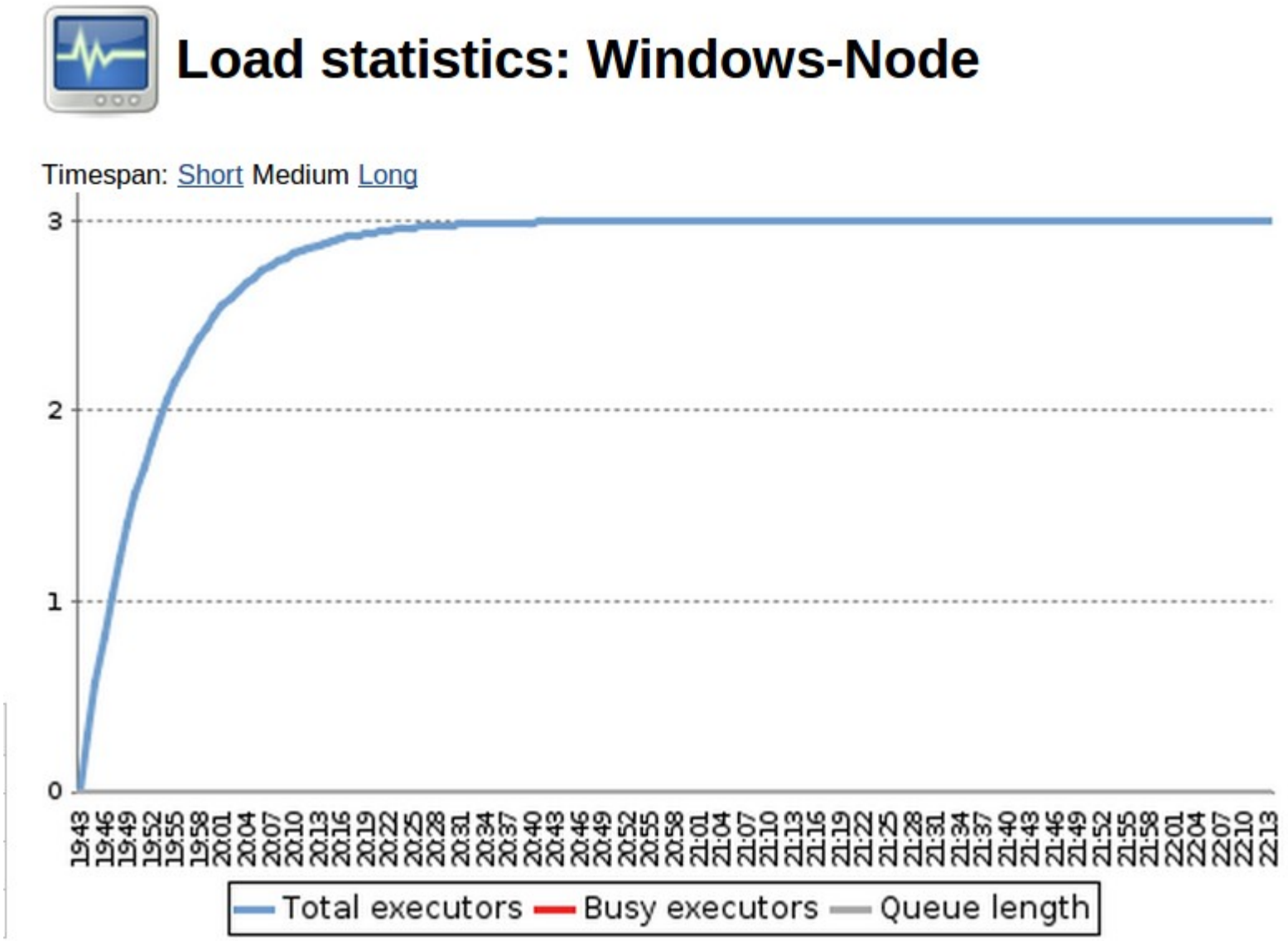
- Records Amount of Disk Space
- Allows you to fix Disk Space issues
- Shows the Project Name and the Build space and the Workspace space taken up
- This plugin allow you to select "Show disk usage trend" for each project

# Lab: Discard Builds & Disk Usage

- Set the `simpleproject` job to max number of builds to 50 to keep the artifacts
- Install the disk usage plugin
- Set `simpleproject` to show disk usage in the project

# Monitor Server Load

# Monitor Server Load



# Monitor Server Load Elements

- Server Load contains three lines
  - Total Executors – Number of executors used
  - Number of Busy Executors – Number of executors that are busy.  If the number matches total executors over a long period, it may be time to add new nodes
  - Queue Length – Number of Jobs waiting for an executor. If this goes above 0, you may need more nodes to handle the requests

# Monitor Server Load

- Server Load Graphs can viewed on:
  - 'Manage Jenkins' page of the master node
  - The node configuration page of the remote node

# Lab: View the Monitor Server Load

- View the Monitor Server Load on the master node and one of the Remote Nodes

# Thanks