



Java is a trademark of Sun Microsystems, Inc.



JavaOneSM

Enterprise Build and Test in the Cloud

Carlos Sanchez
G2iX

About me

- > Sr. Solutions Architect at G2iX
- > Member of Apache Maven PMC, Continuum, Archiva
- > Eclipse IAM co-lead
- > etc...



Index

1. The tools
2. Maven Integration Tests
3. Maven & Selenium
4. Adding Continuum to the Mix
5. Testing in Different Environments
6. And Now in the Cloud

1. The tools

Build: Apache Maven

maven

Build: Apache Maven

> Build tool and more

maven

Build: Apache Maven

- > Build tool and more
- > Dependency management

maven

Build: Apache Maven

- > Build tool and more
- > Dependency management
- > Execution of unit/integration tests

maven

Build: Apache Maven

- > Build tool and more
- > Dependency management
- > Execution of unit/integration tests
- > Start/stop application server

maven

Build: Apache Maven

- > Build tool and more
- > Dependency management
- > Execution of unit/integration tests
- > Start/stop application server
- > Automatic application deployment

maven

Test Cases: TestNG

Test Cases: TestNG

> Unit/integration tests

Test Cases: TestNG

- > Unit/integration tests
- > parameterized tests

Test Cases: TestNG

- > Unit/integration tests
- > parameterized tests
- > parallel testing

Integration Tests: Selenium



Integration Tests: Selenium

- > UI and Integration testing



Integration Tests: Selenium

- > UI and Integration testing
- > Tests run in the browser
 - support for multiple browsers



Integration Tests: Selenium

- > UI and Integration testing
- > Tests run in the browser
 - support for multiple browsers
- > Tests can be recorded
 - no need for developers writing tests
 - Selenium IDE Firefox plugin



Continuous Integration: Apache Continuum



Continuous Integration: Apache Continuum

- > Tight integration with Apache Maven
 - dependency handling



Continuous Integration: Apache Continuum

- > Tight integration with Apache Maven
 - dependency handling
- > Trigger builds
 - when integration tests change
 - when the webapp changes
 - when the webapp dependencies change



Cloud Computing: Amazon Web Services



Cloud Computing: Amazon Web Services

- > Amazon S3
 - storage



Cloud Computing: Amazon Web Services

- > Amazon S3
 - storage
- > Amazon SQS
 - queue service



Cloud Computing: Amazon Web Services

- > Amazon S3
 - storage
- > Amazon SQS
 - queue service
- > Amazon Mechanical Turk
 - manpower



Cloud Computing: Amazon Web Services

- > Amazon S3
 - storage
- > Amazon SQS
 - queue service
- > Amazon Mechanical Turk
 - manpower
- > Amazon EC2
 - computation



2. Maven integration tests

maven

The webapp to test

> appfuse-struts



A new Maven project

> appfuse-selenium

```
<groupId>org.appfuse</groupId>  
<artifactId>appfuse-selenium</artifactId>  
<version>2.0.2-SNAPSHOT</version>  
<packaging>war</packaging>  
<name>AppFuse Selenium testing</name>
```

that depends on

> appfuse-struts

```
<dependency>  
  <groupId>org.appfuse</groupId>  
  <artifactId>appfuse-struts</artifactId>  
  <version>2.0.2</version>  
  <type>war</type>  
</dependency>
```

using Cargo to run it

> automatically

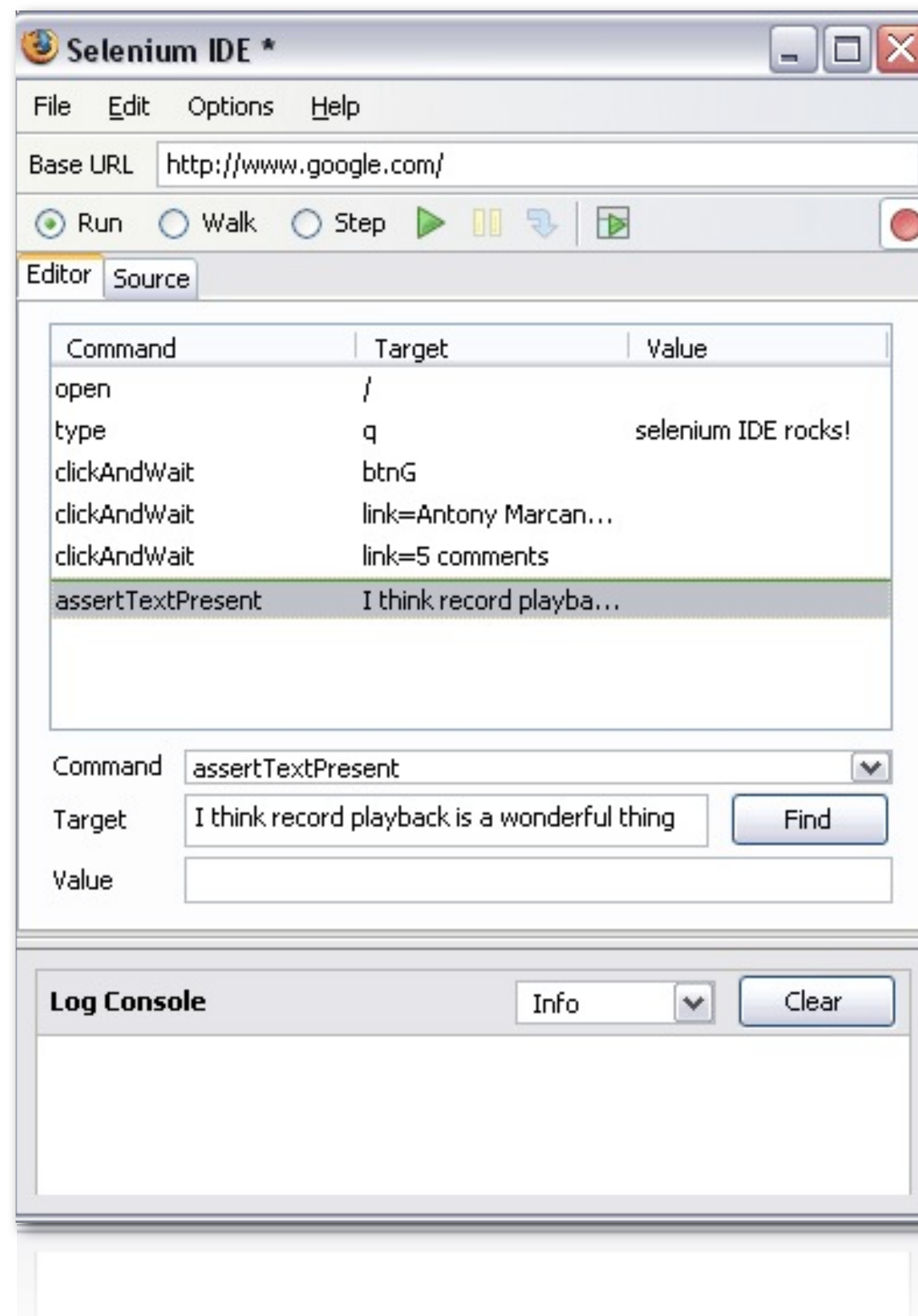
```
<plugin>
```

```
  <groupId>org.codehaus.cargo</groupId>
```

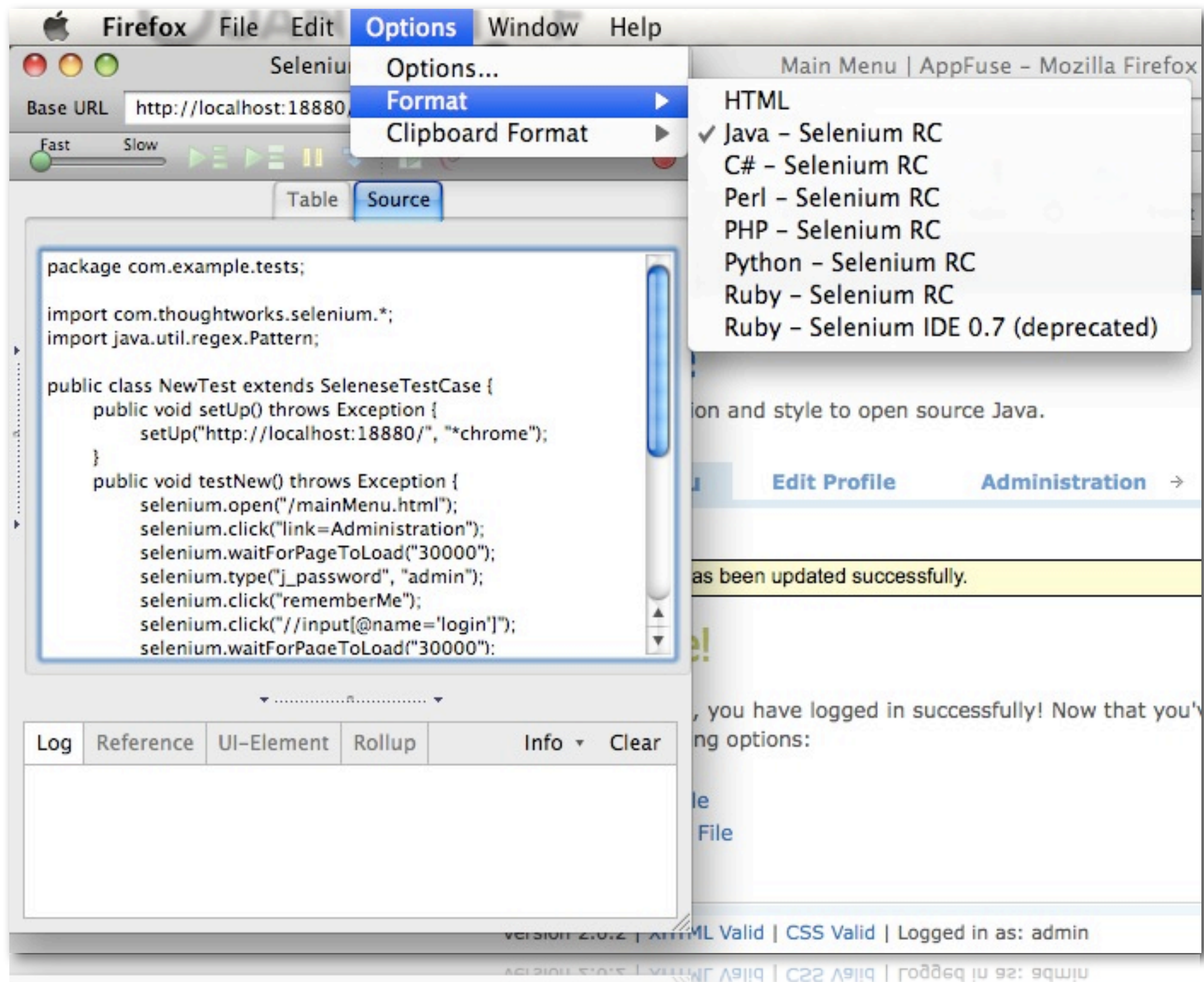
```
  <artifactId>cargo-maven2-plugin</artifactId>
```


Record Selenium tests

> SeleniumIDE



as Java



test cases

> using Selenium API

```
public void setUp() throws Exception {  
    setUp("http://localhost:18880/", "*chrome");  
}  
public void testNew() throws Exception {  
    selenium.open("/mainMenu.html");  
    selenium.click("link=Administration");  
    selenium.waitForPageToLoad("30000");  
    selenium.type("j_password", "admin");  
    selenium.click("rememberMe");  
    selenium.click("//input[@name='login']");  
    selenium.waitForPageToLoad("30000");  
    selenium.click("link=Edit Profile");  
    selenium.waitForPageToLoad("30000");  
    selenium.click("link=Main Menu");  
    selenium.waitForPageToLoad("30000");  
    selenium.click("link=Edit Profile");  
    selenium.waitForPageToLoad("30000");  
    selenium.click("document.forms[0].elements[24]");  
    selenium.waitForPageToLoad("30000");  
}
```

breakpoint

- > Selenium tests in Maven IT project

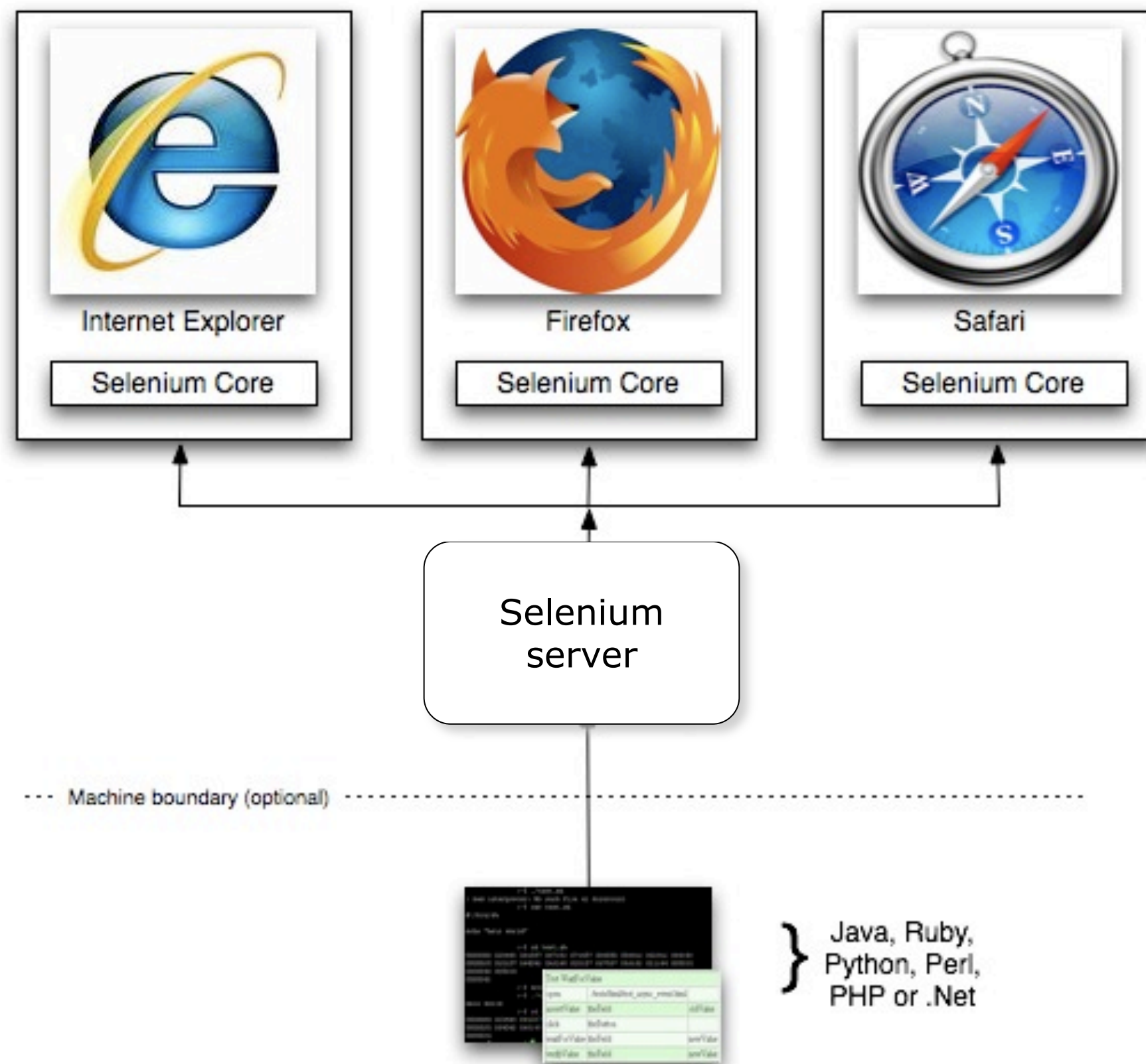
3. Maven & Selenium

maven



Selenium Architecture

Windows, Linux, or Mac (as appropriate)...



Maven & Selenium

- > Start Selenium server locally
- > Run integration tests
- > Stop selenium server

maven

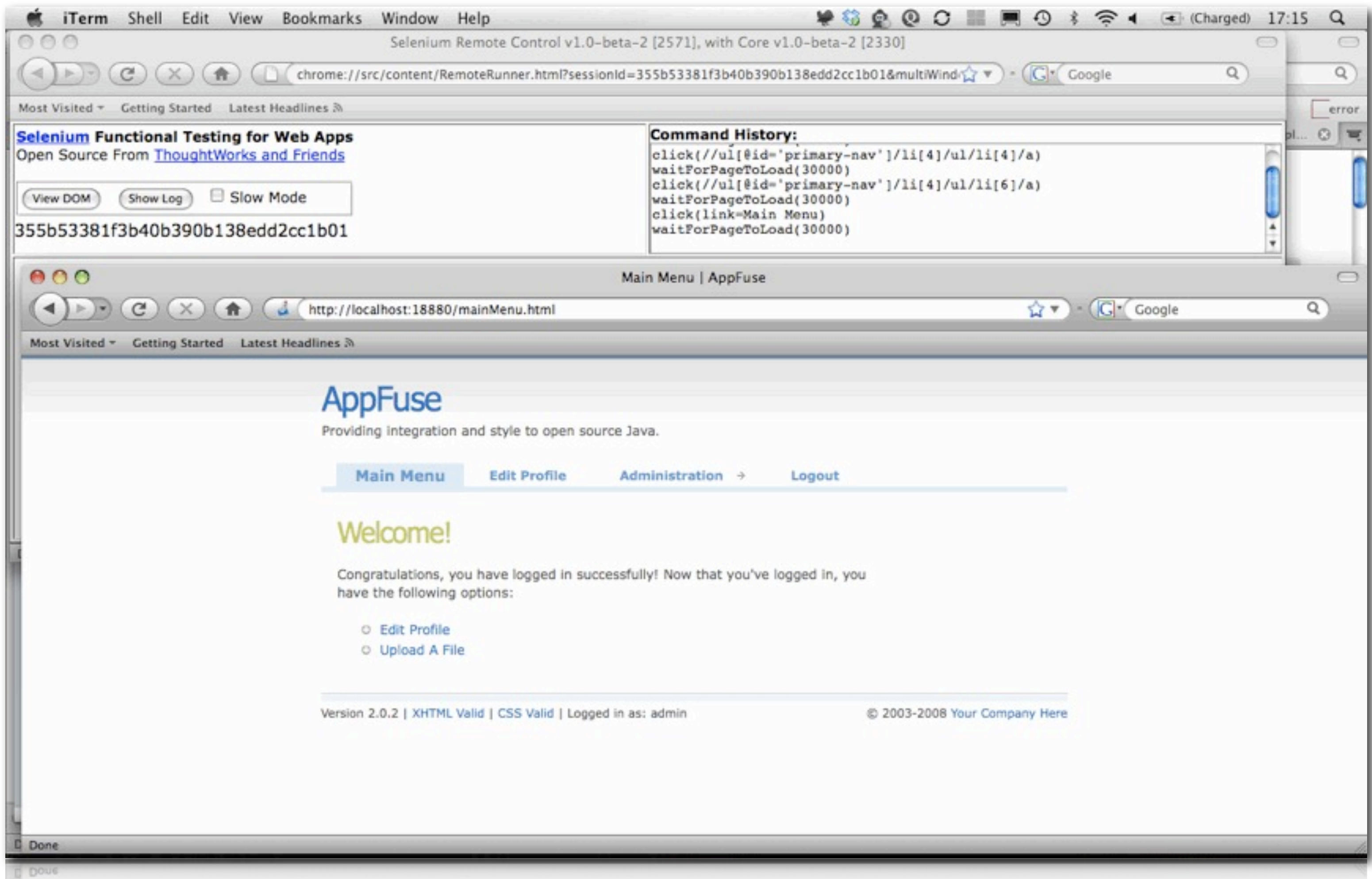
Selenium Maven plugin

- > <http://mojo.codehaus.org/selenium-maven-plugin>
- > selenium:start-server
- > selenium:stop-server

maven

Selenium Maven plugin

```
<plugin>
  <groupId>org.codehaus.mojo</groupId>
  <artifactId>selenium-maven-plugin</artifactId>
  <version>1.0-rc-1</version>
  <executions>
    <execution>
      <id>start-selenium</id>
      <phase>pre-integration-test</phase>
      <goals>
        <goal>start-server</goal>
      </goals>
    </execution>
  </executions>
  <configuration>
    <background>${selenium.background}</background>
  </configuration>
</plugin>
```



The screenshot shows a Mac desktop with two windows. The top window is titled "Selenium Remote Control v1.0-beta-2 [2571], with Core v1.0-beta-2 [2330]". It displays a Selenium Remote Runner interface with a "Command History" section containing the following commands:

```
click("//ul[@id='primary-nav']/li[4]/ul/li[4]/a)
waitForPageToLoad(30000)
click("//ul[@id='primary-nav']/li[4]/ul/li[6]/a)
waitForPageToLoad(30000)
click(link=Main Menu)
waitForPageToLoad(30000)
```

The bottom window is titled "Main Menu | AppFuse" and shows the AppFuse web application. The page has a navigation bar with links: "Main Menu", "Edit Profile", "Administration", and "Logout". The main content area displays a "Welcome!" message and a list of options: "Edit Profile" and "Upload A File". The footer of the page includes the text "Version 2.0.2 | XHTML Valid | CSS Valid | Logged in as: admin" and "© 2003-2008 Your Company Here".

T E S T S

Running TestSuite

```
log4j:WARN No appenders could be found for logger
(com.thoughtworks.selenium.grid.tools.ThreadSafeSeleniumSessionStorage).
log4j:WARN Please initialize the log4j system properly.
DEBUG [btpool0-5] UserAction.edit(117) | checking for remember me login...
DEBUG [btpool0-8] LookupDaoHibernate.getRoles(20) | Retrieving all role names...
DEBUG [btpool0-8] StartupListener.setupContext(102) | Drop-down initialization complete [OK]
INFO [btpool0-8] ReloadAction.execute(34) | reload complete, reloading user back to: http://localhost:18880/admin/activeUsers.html
DEBUG [btpool0-3] UserAction.edit(117) | checking for remember me login...
DEBUG [btpool0-3] UserAction.edit(117) | checking for remember me login...
Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 89.703 sec
```

Results :

Tests run: 1, Failures: 0, Errors: 0, Skipped: 0

```
[INFO] [cargo:stop {execution: stop-container}]
2009-03-23 16:41:42.176:/appfuse-selenium-2.0.2-SNAPSHOT:INFO: Closing Spring root
WebApplicationContext
[INFO] [install:install]
[INFO] Installing /Users/csanchez/dev/maestro/appfuse-selenium/target/appfuse-selenium-2.0.2-
SNAPSHOT.war to /Users/csanchez/.m2/repository/org/appfuse/appfuse-selenium/2.0.2-SNAPSHOT/appfuse-
selenium-2.0.2-SNAPSHOT.war
[INFO] -----
[INFO] BUILD SUCCESSFUL
[INFO] -----
[INFO] Total time: 2 minutes 20 seconds
[INFO] Finished at: Mon Mar 23 16:41:45 CET 2009
[INFO] Final Memory: 43M/63M
[INFO] -----
2009-03-23 16:41:46.168::INFO: Shutdown hook executing
2009-03-23 16:41:46.169::INFO: Shutdown hook complete
```


breakpoint

- > Selenium tests in Maven IT project
- > running locally in the browser

4. Adding Continuum to the Mix



Continuum



Continuum

> Build the webapp if



Continuum

- > Build the webapp if
 - the webapp changes



Continuum

- > Build the webapp if
 - the webapp changes
 - the dependencies change



Continuum

- > Build the webapp if
 - the webapp changes
 - the dependencies change
- > Run the integration tests



Continuum

- > Build the webapp if
 - the webapp changes
 - the dependencies change
- > Run the integration tests
 - if the ITs change



Continuum

- > Build the webapp if
 - the webapp changes
 - the dependencies change
- > Run the integration tests
 - if the ITs change
 - if the webapp changes




Continuum

- > Add appfuse-struts
- > Add appfuse-selenium

Project Group Last Build Result Overview

Success : 0  Errors : 0  Failed : 1 

Member Projects

<input type="checkbox"/>		Project Name	Version
<input type="checkbox"/>		AppFuse Selenium testing	2.0.2-SNAPSHOT
<input type="checkbox"/>		AppFuse Struts 2 Module	2.0.2-SNAPSHOT
<input type="checkbox"/>		AppFuse Struts 2 Module	2.0.2-SNAPSHOT

Selenium headless

- > Most likely needed
- > Use *nix X Virtual Frame Buffer
- > In Maven
 - selenium:xvfb

breakpoint

- > Selenium tests in Maven IT project
- > running locally in the browser
- > triggered by Continuum when there is any change to webapp, tests or dependencies

5. Testing in different environments

Multi environment

Multi environment

Multiple Browsers

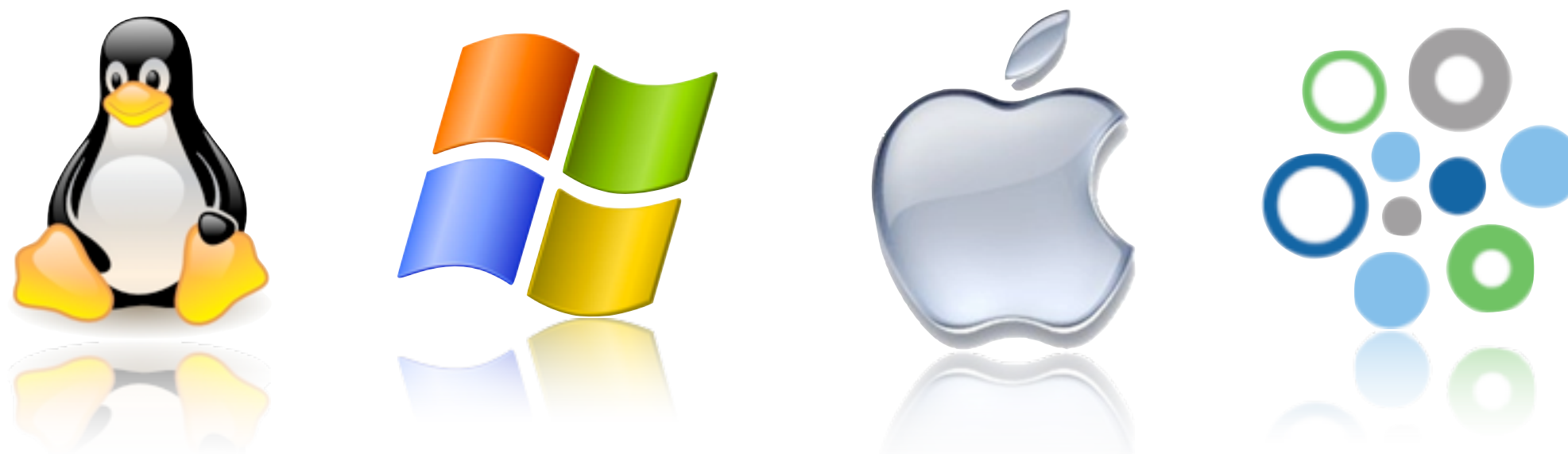


Multi environment

Multiple Browsers



Multiple Operating Systems



Multi environment

Multiple Browsers



Multiple Operating Systems

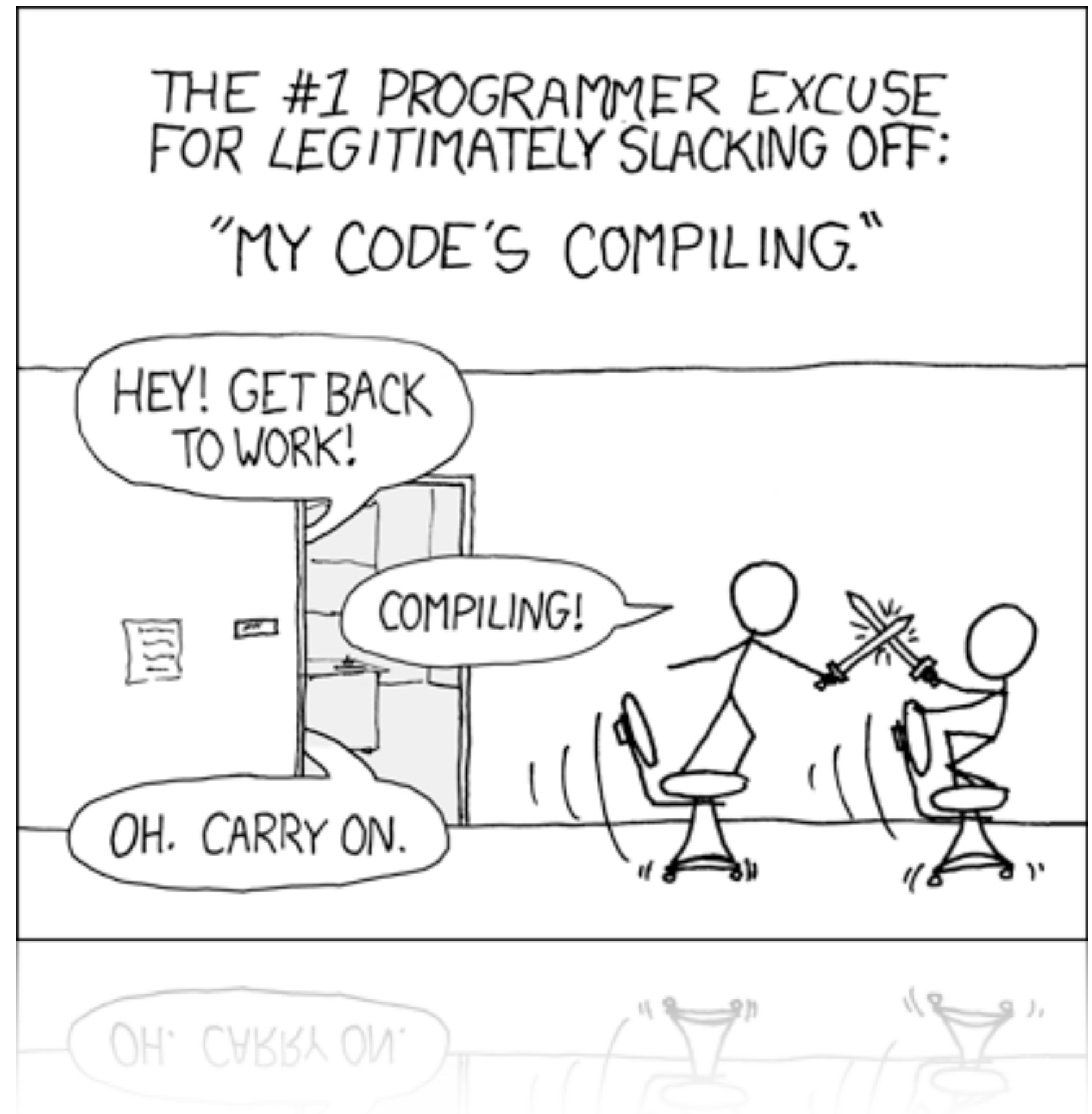


Parallel

- > Run tests in parallel
 - in each browser
 - several browsers per test

Parallel

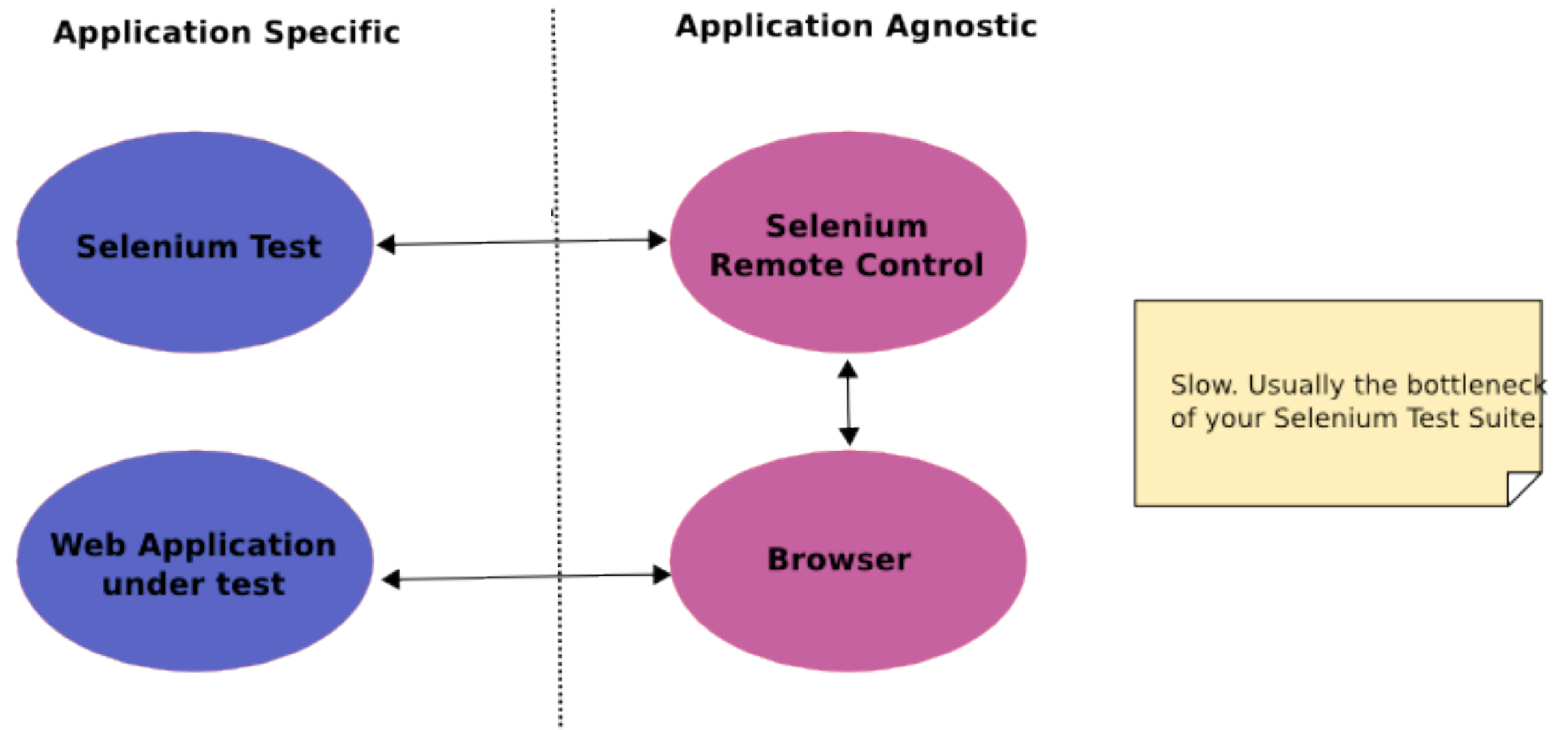
- > Faster testing
- > Load testing



Credits: XKCD

Selenium Grid

Traditional Selenium Setup



Selenium Grid Setup

- * No change required
- * Write them exactly as you would in the traditional setup
- * Make them run in parallel to take advantage of the grid

Application Specific

Selenium Tests

Web Application
under test

Selenium Grid - Application Agnostic

- * Unique entry point for Selenium tests
- * Load balance selenese requests to "real" remote controls
- * Scale transparently by adding more remote controls
- * Run multiple remote controls per machine (typically)

Selenium
Hub

Selenium
Remote Control

Selenium
Remote Control

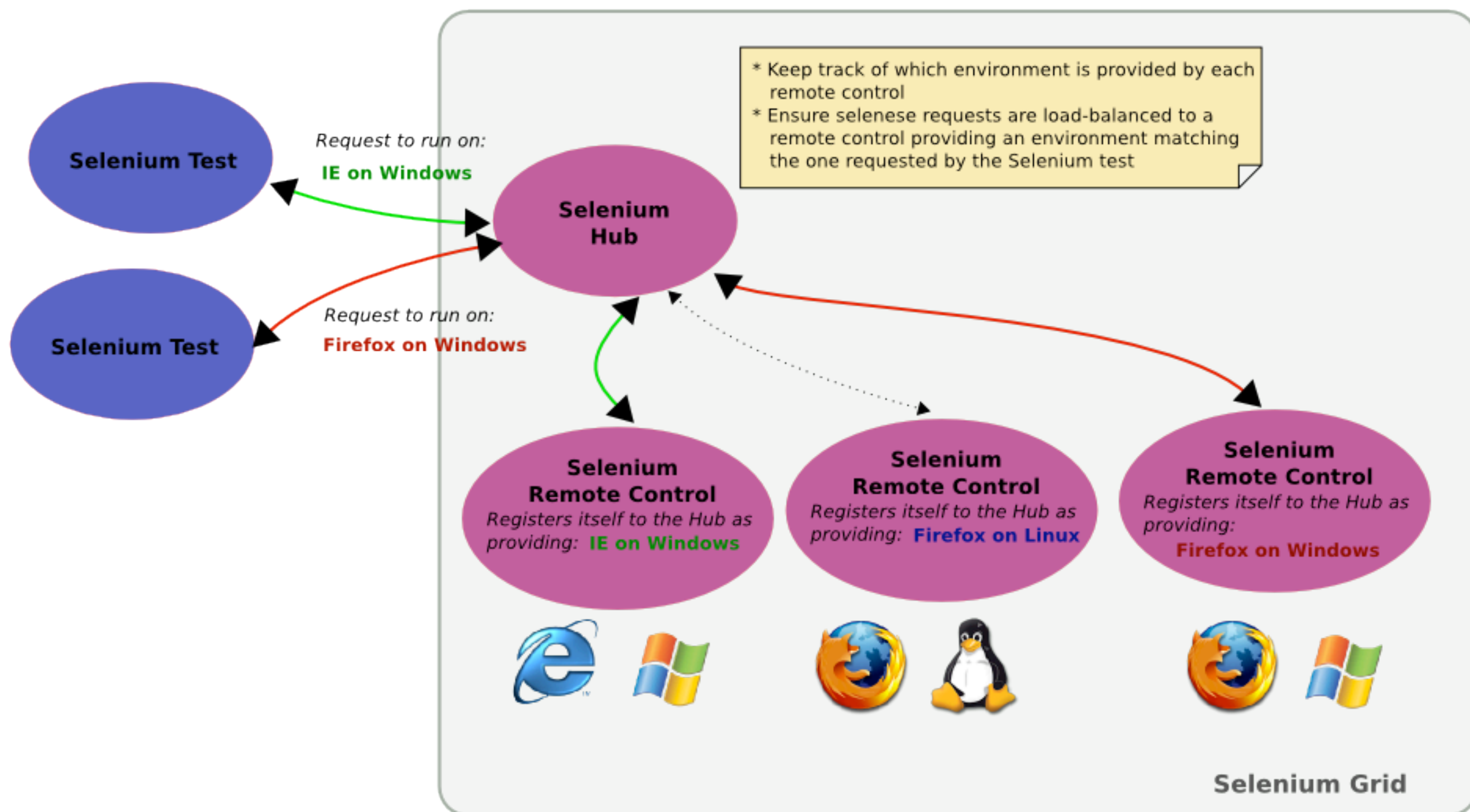
Selenium
Remote Control

Browser

Browser

Browser

Selenium Grid : Requesting a Specific Environment



TestNG properties

- > Allows to run the same tests in different environments
- > and in parallel

TestNG properties

```
<suite name="Example" parallel="tests" thread-count="3">
  <test name="iexplore">
    <parameter name="browser" value="*iexplore"/>
    <packages>
      <package name="org.appfuse.webapp.selenium" />
    </packages>
  </test>
  <test name="firefox2">
    <parameter name="browser" value="*firefox2"/>
    <packages>
      <package name="org.appfuse.webapp.selenium" />
    </packages>
  </test>
  <test name="opera">
    <parameter name="browser" value="*opera"/>
    <packages>
      <package name="org.appfuse.webapp.selenium" />
    </packages>
  </test>
</suite>
```

TestNG properties

```
<suite name="Example" parallel="tests" thread-count="3">
  <test name="iexplore">
    <parameter name="browser" value="*iexplore"/>
    <packages>
      <package name="org.appfuse.webapp.selenium" />
    </packages>
  </test>
  <test name="firefox2">
    <parameter name="browser" value="*firefox2"/>
    <packages>
      <package name="org.appfuse.webapp.selenium" />
    </packages>
  </test>
  <test name="opera">
    <parameter name="browser" value="*opera"/>
    <packages>
      <package name="org.appfuse.webapp.selenium" />
    </packages>
  </test>
</suite>
```

breakpoint

- > Selenium tests in Maven IT project
- > running locally in several browsers

6. and now in the Cloud

\$\$\$



\$\$\$



> Machines cost money

\$\$\$



- > Machines cost money
- > Bandwidth costs money

\$\$\$



- > Machines cost money
- > Bandwidth costs money
- > Electricity costs money

\$\$\$

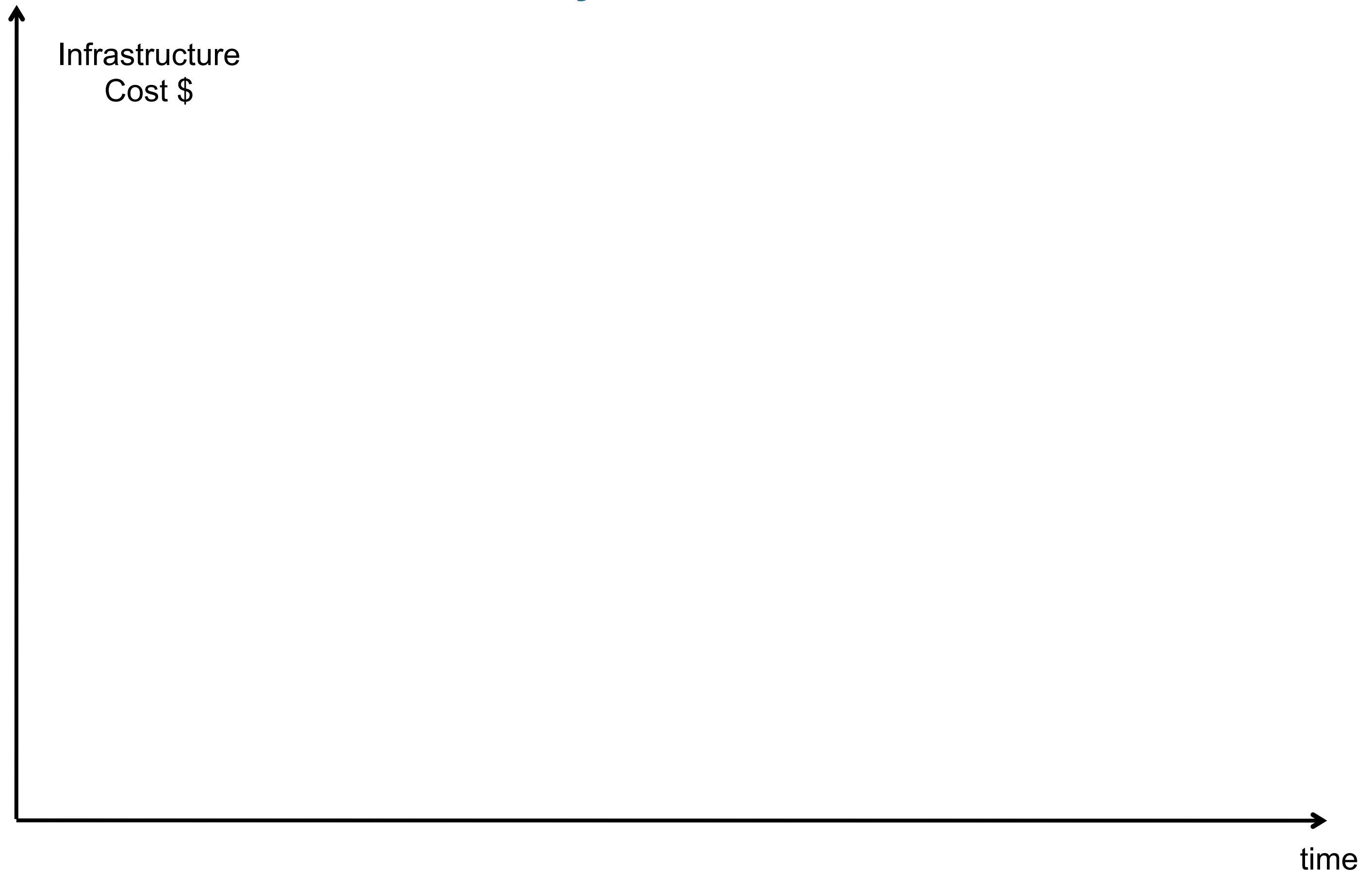


- > Machines cost money
- > Bandwidth costs money
- > Electricity costs money
- > Server administration costs money

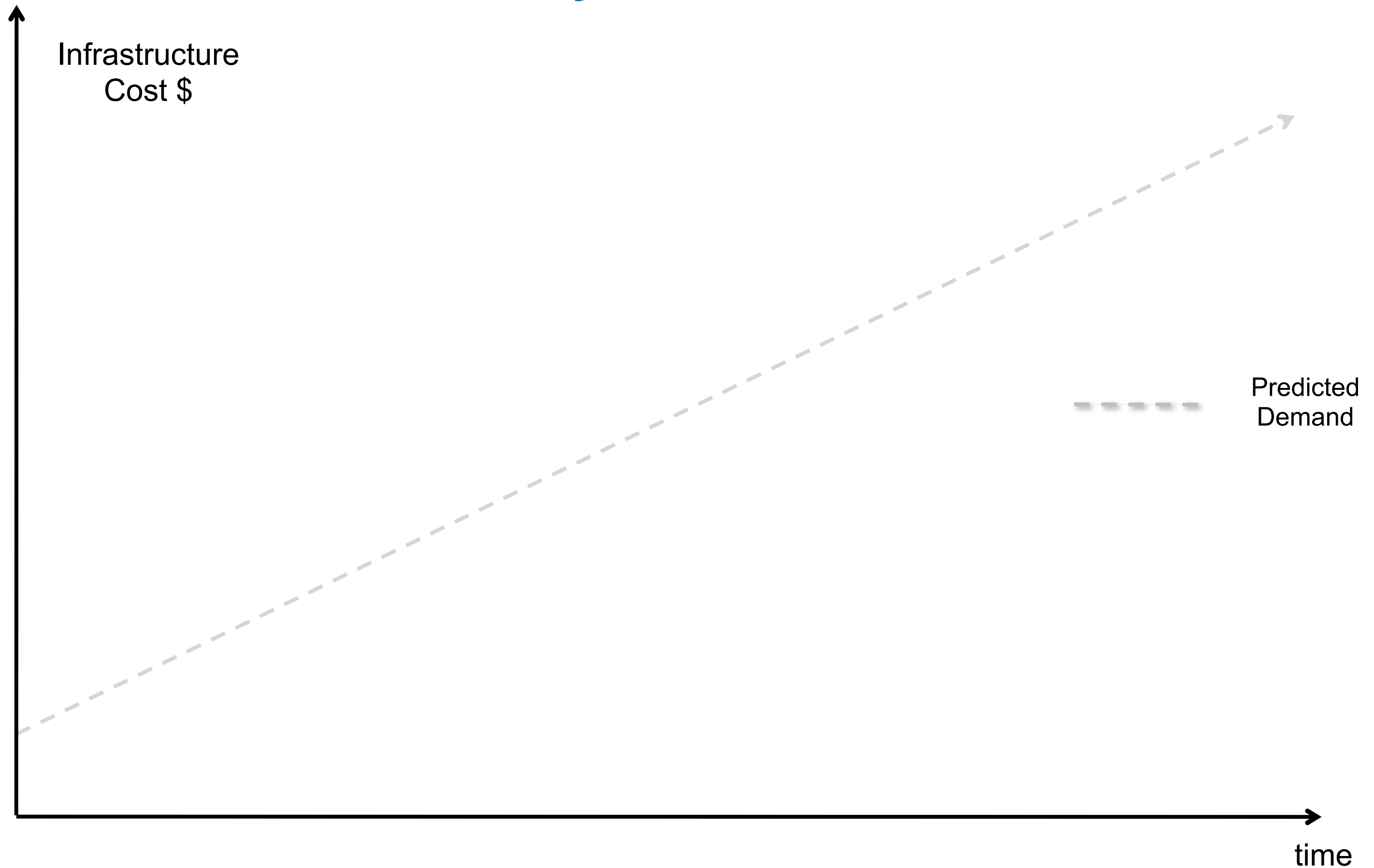
Resource Optimization

- > test servers working 10-20% of the time

Predictions cost money

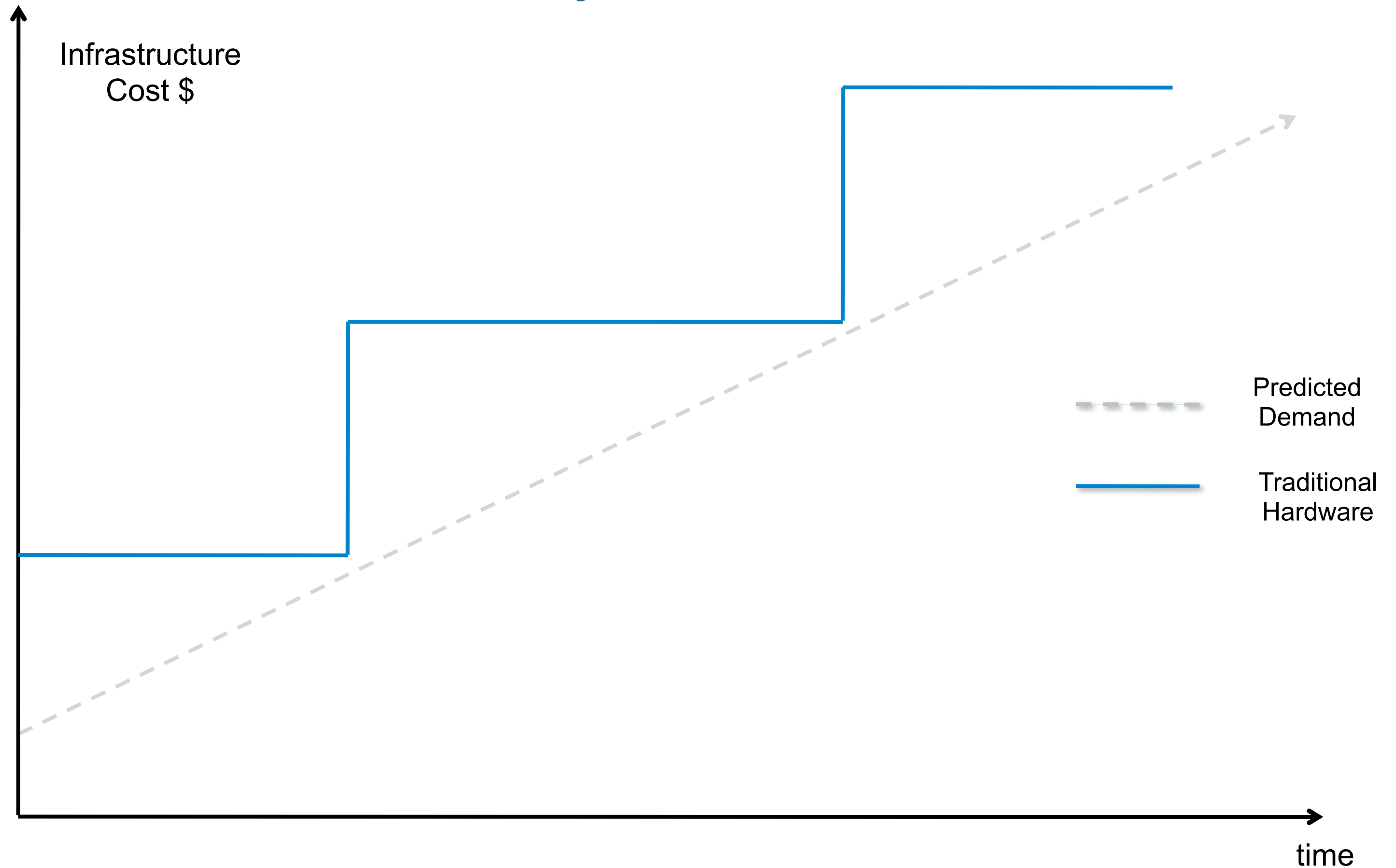


Predictions cost money



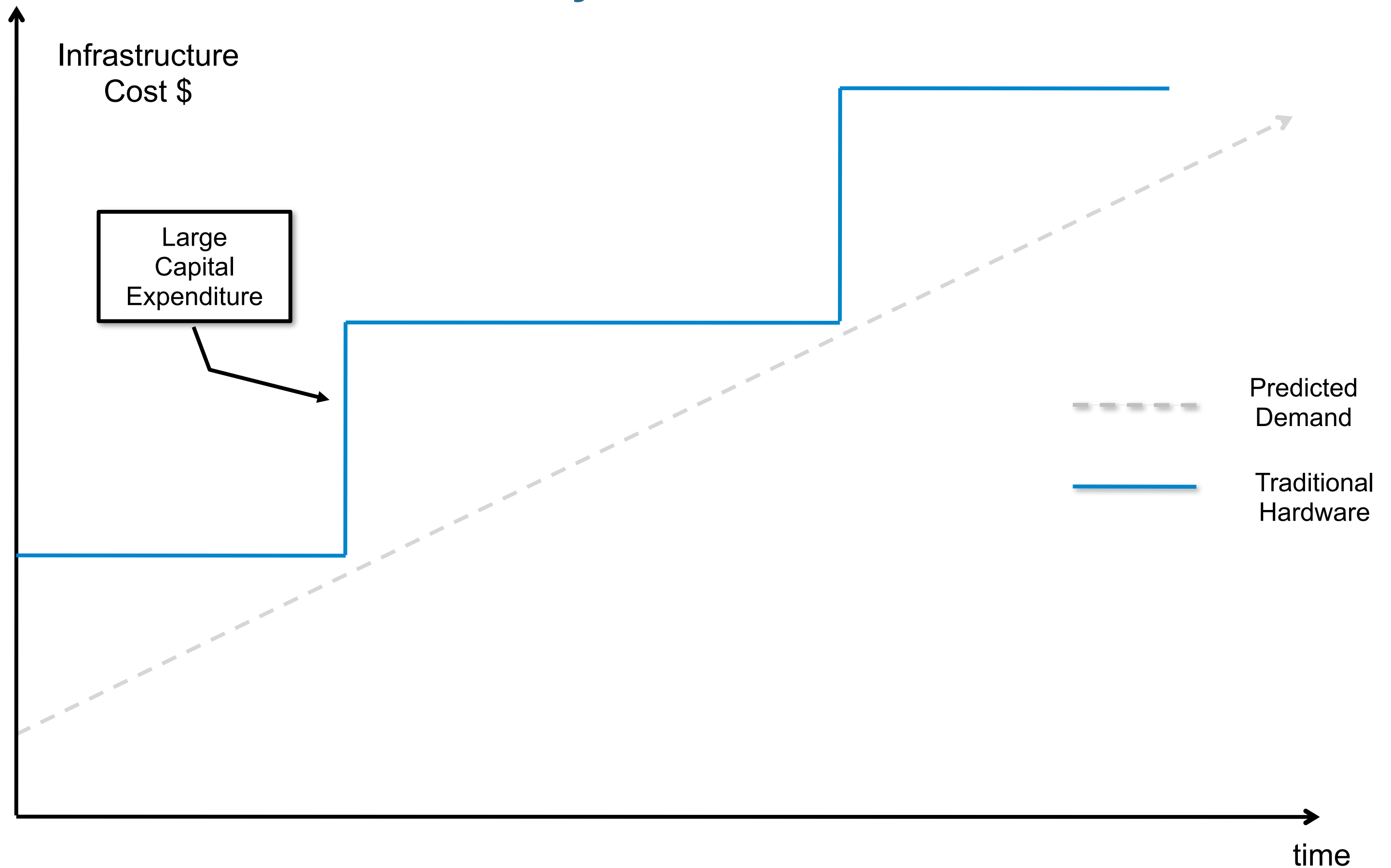
Source: Amazon

Predictions cost money



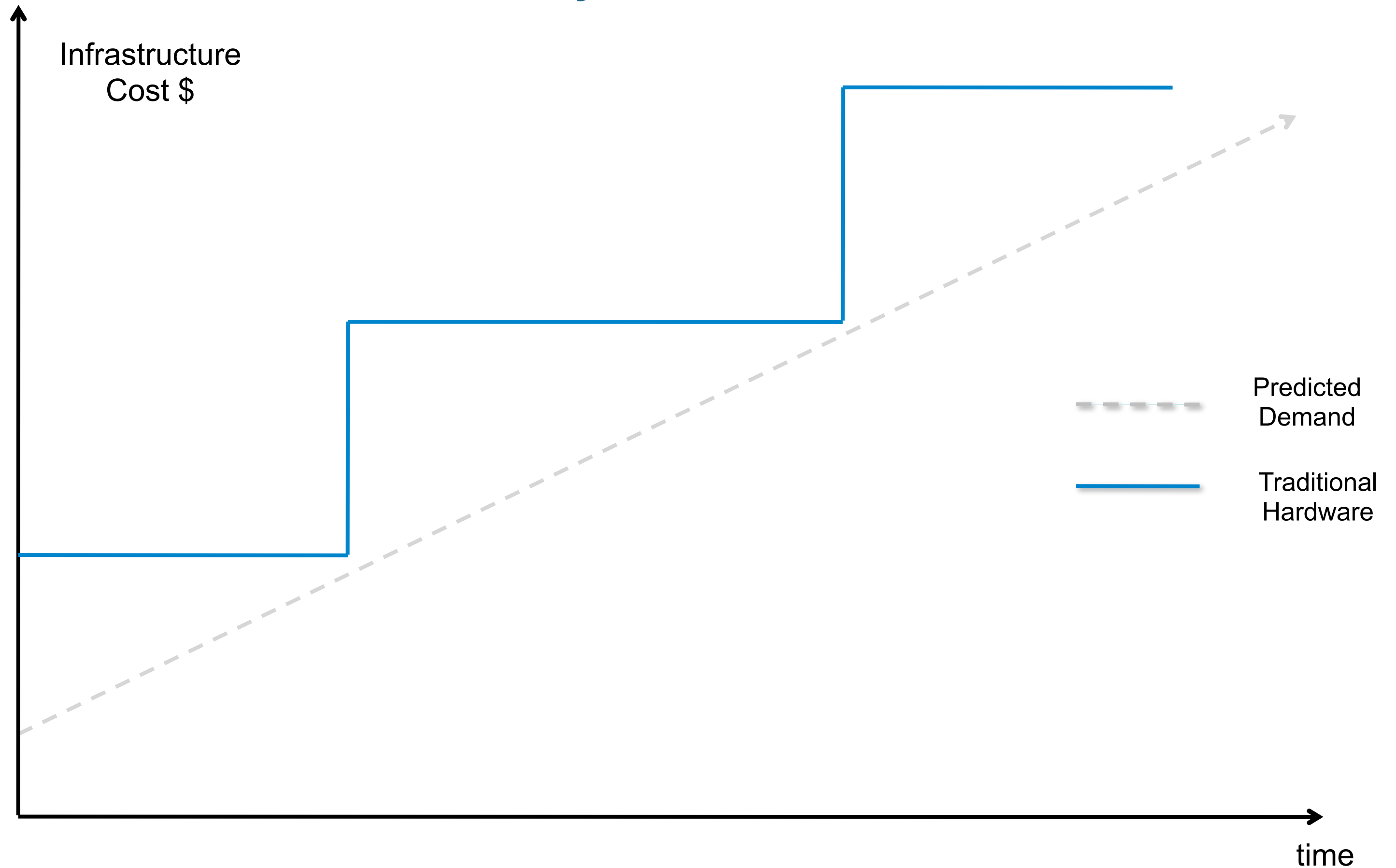
Source: Amazon

Predictions cost money



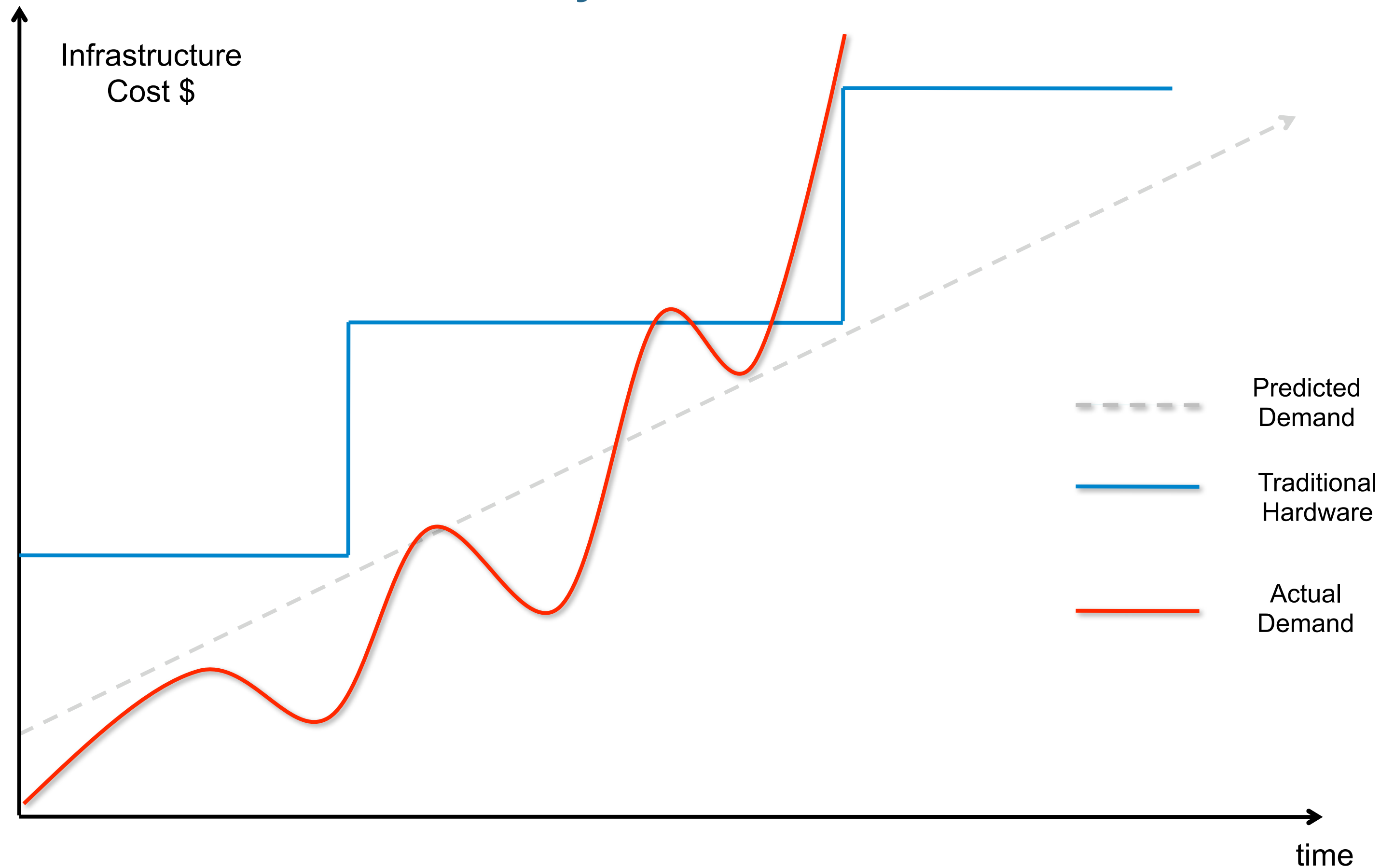
Source: Amazon

Predictions cost money



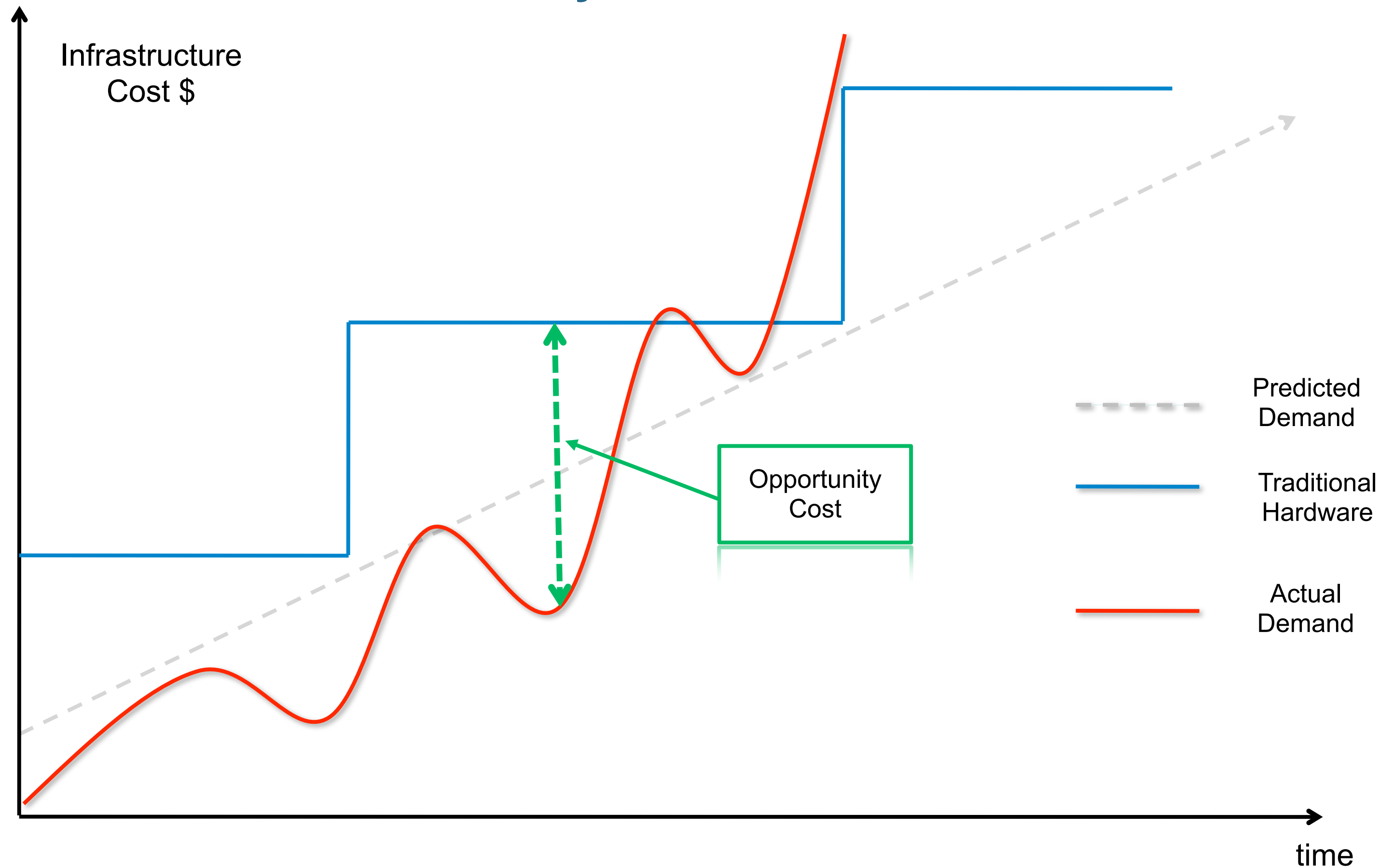
Source: Amazon

Predictions cost money



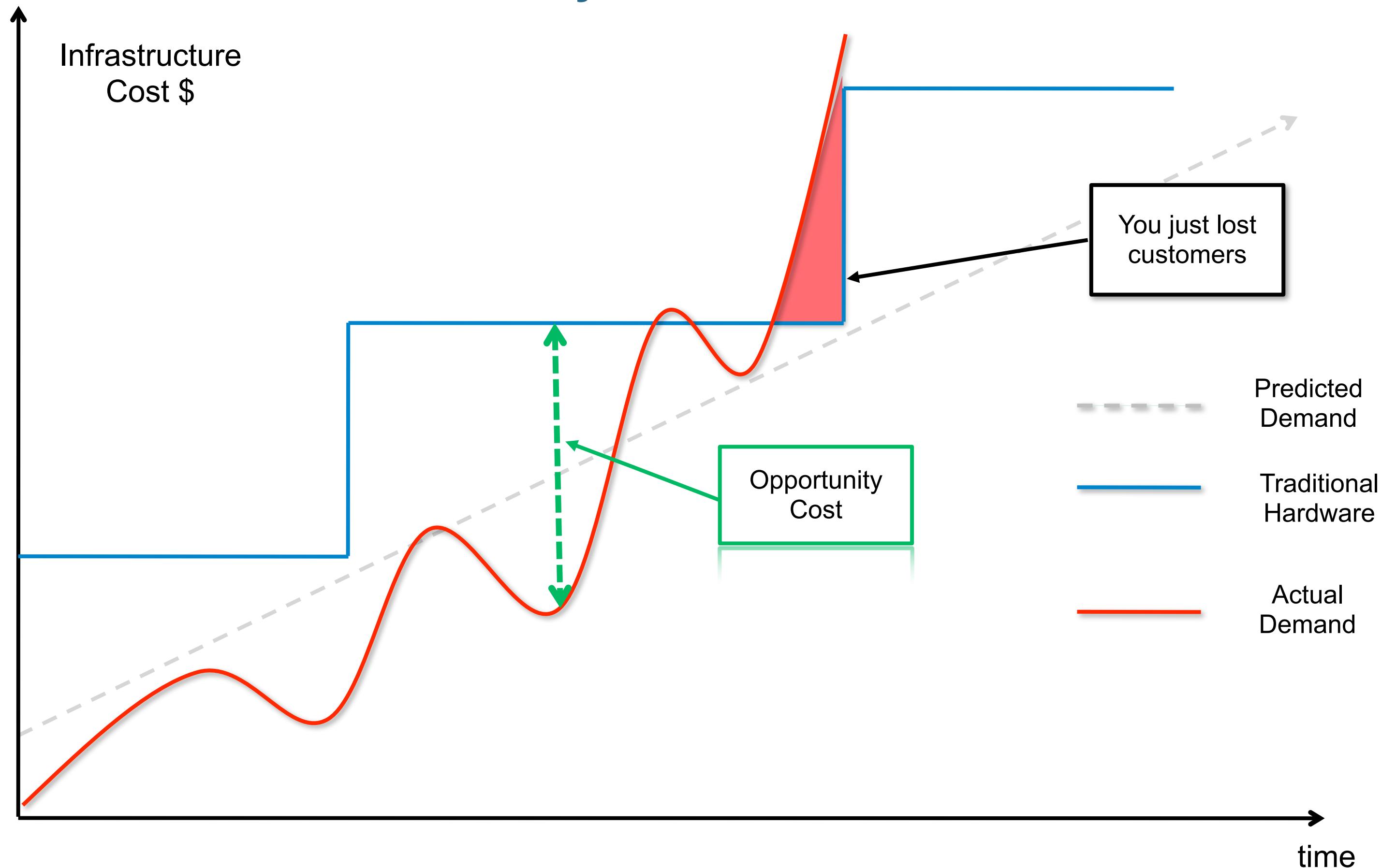
Source: Amazon

Predictions cost money



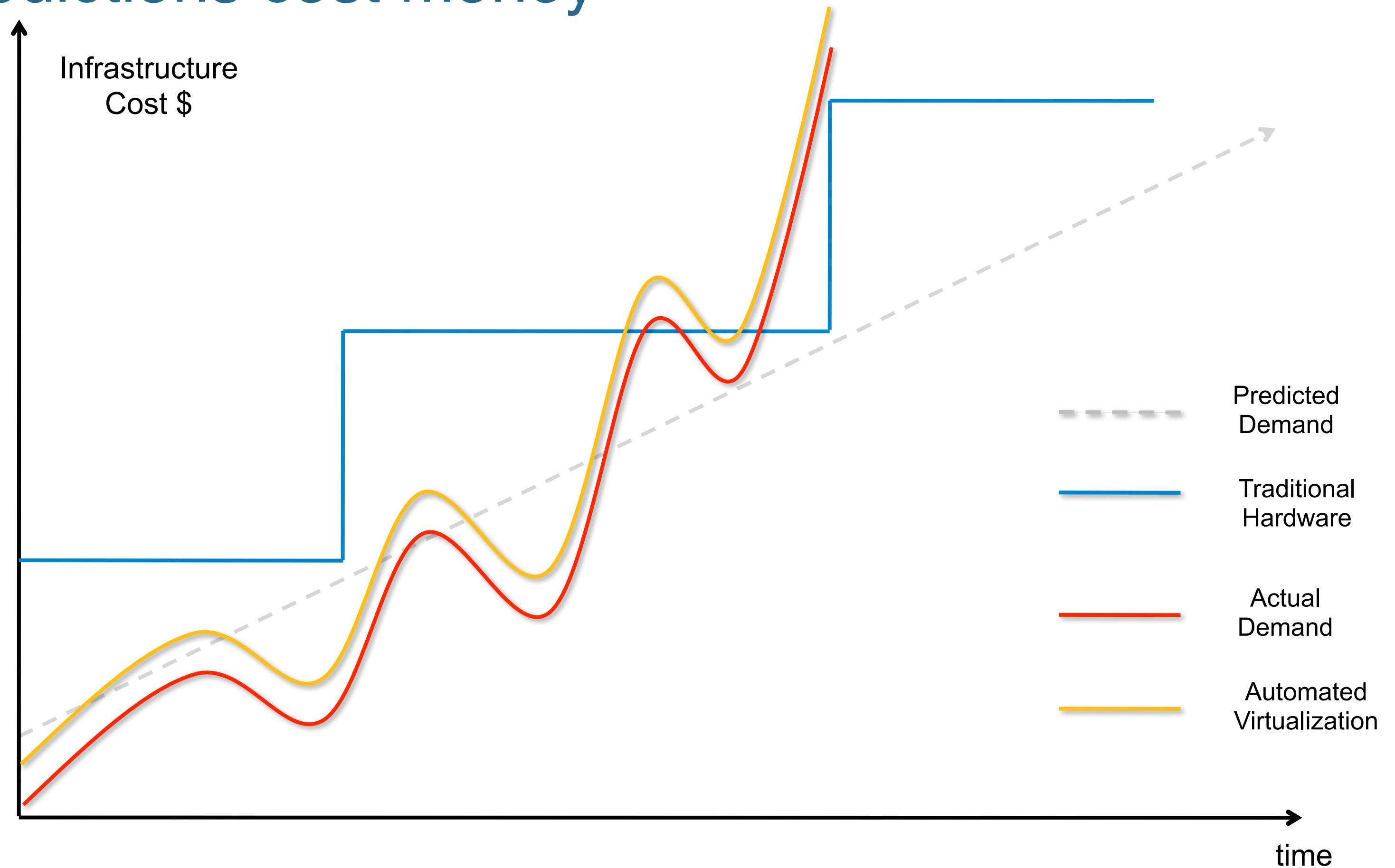
Source: Amazon

Predictions cost money



Source: Amazon

Predictions cost money



Availability

> Can you get 3000 machines in 3 days?

Amazon EC2

- > Elastic Compute Cloud
- > Machines on demand

Amazon EC2

- > pay per hour
 - same price for
 - 10 machines / 1 hour
 - 1 machine / 10 hours
- > unlimited number of machines
- > start in couple minutes



AMIs

AMIs

Amazon Machine Images



AMIs

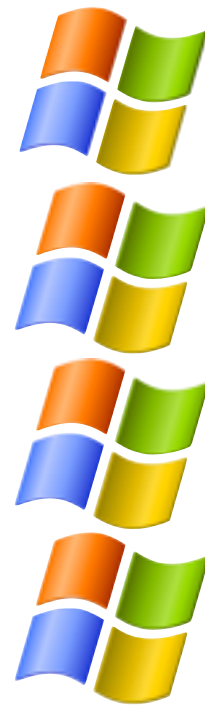
Amazon Machine Images



Bundle your own

Countless possibilities

- > Internet Explorer
- > Firefox 2
- > Opera
- > Safari
- > Firefox 3
- > Firefox 2
- > etc...
- > no OS X :(



Amazon AMIs

Amazon AMIs

- > Create AMIs for different environments to test

Amazon AMIs

- > Create AMIs for different environments to test
- > Read user data to customize the images at runtime
 - where is the Selenium Hub

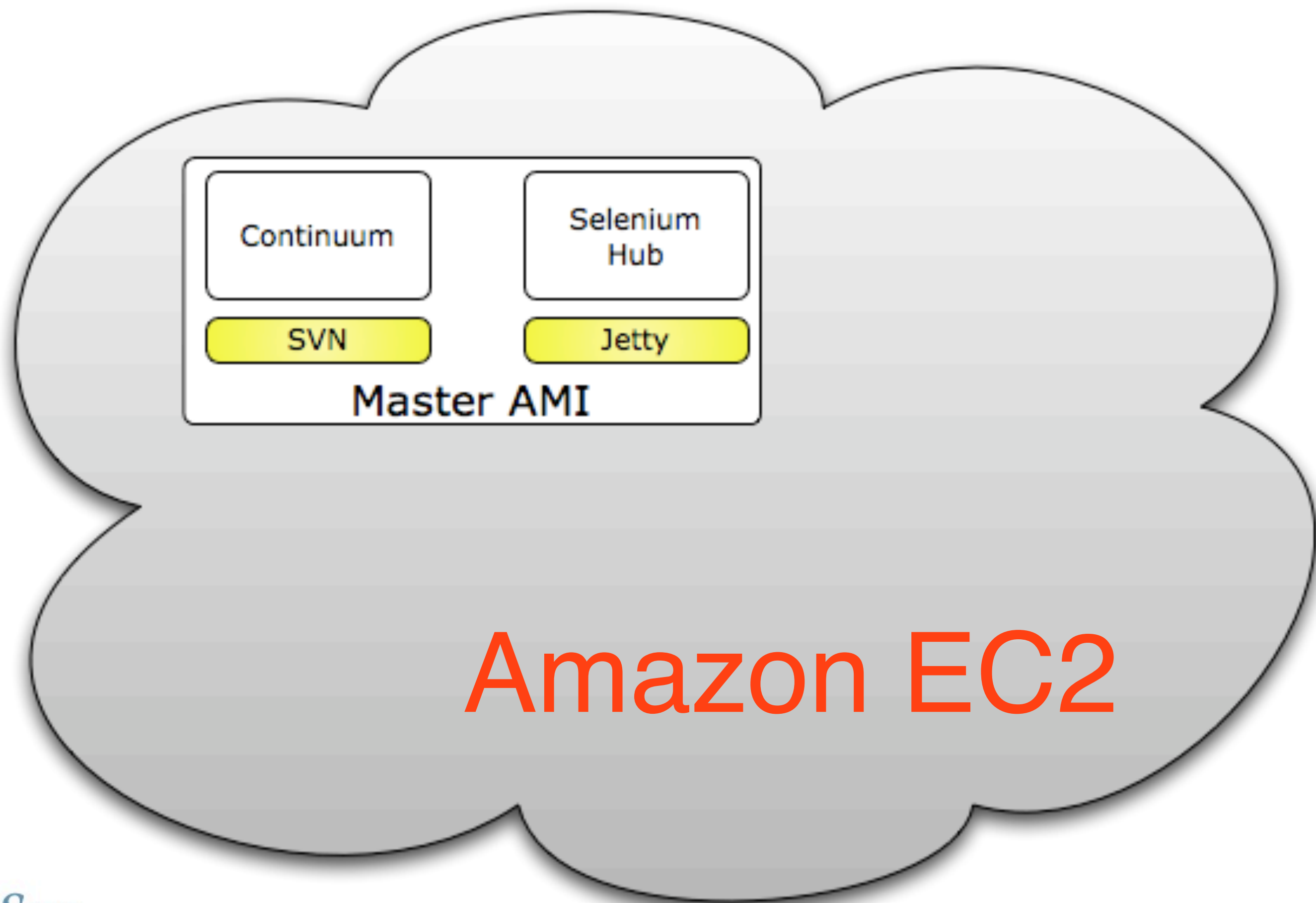
Maven EC2 plugin

- > *in progress*

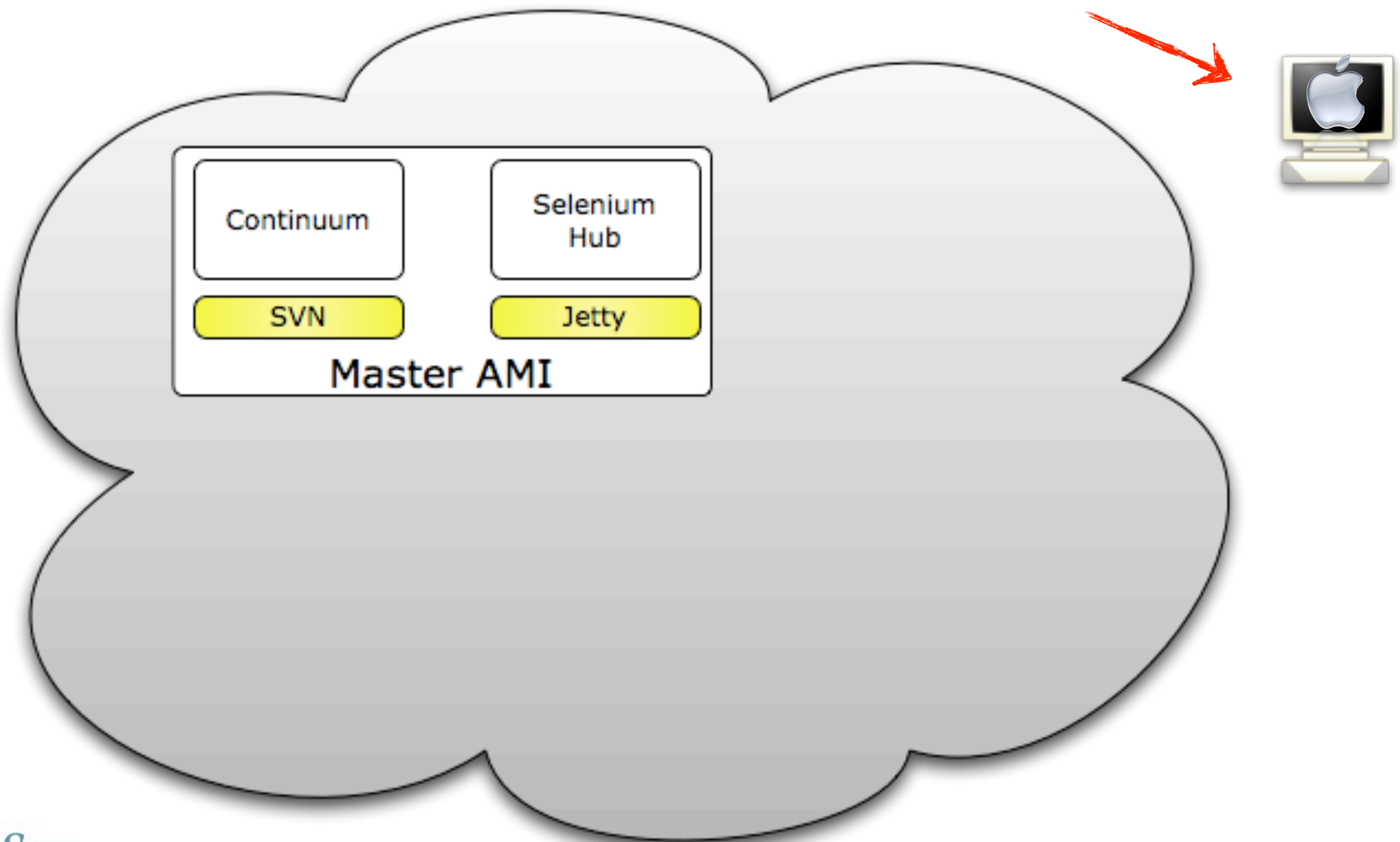
- > <http://mojo.codehaus.org/maven-ec2-plugin>
 - start Amazon AMIs
 - stop at the end of the build
 - pass on user data

Workflow

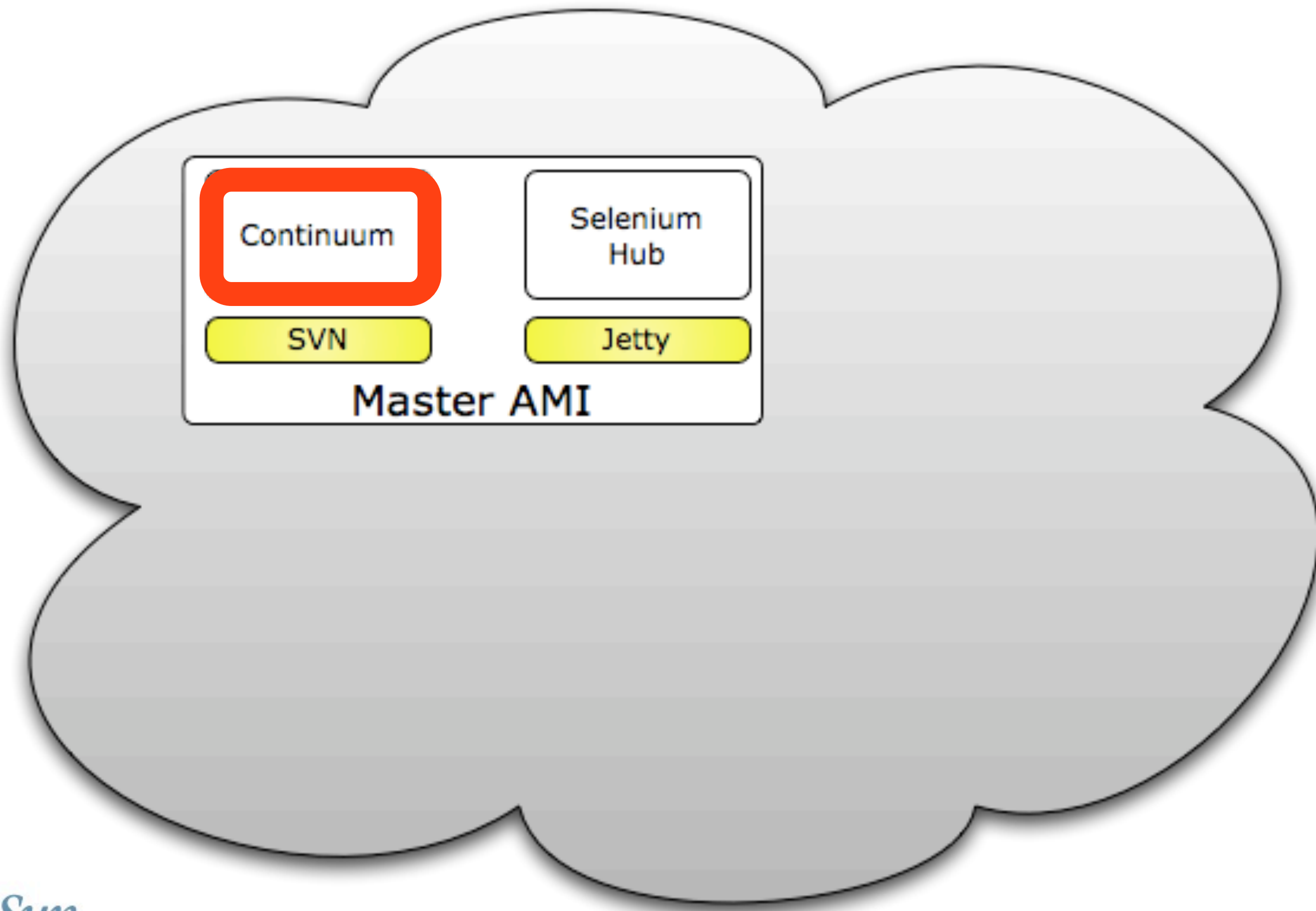
Starting point



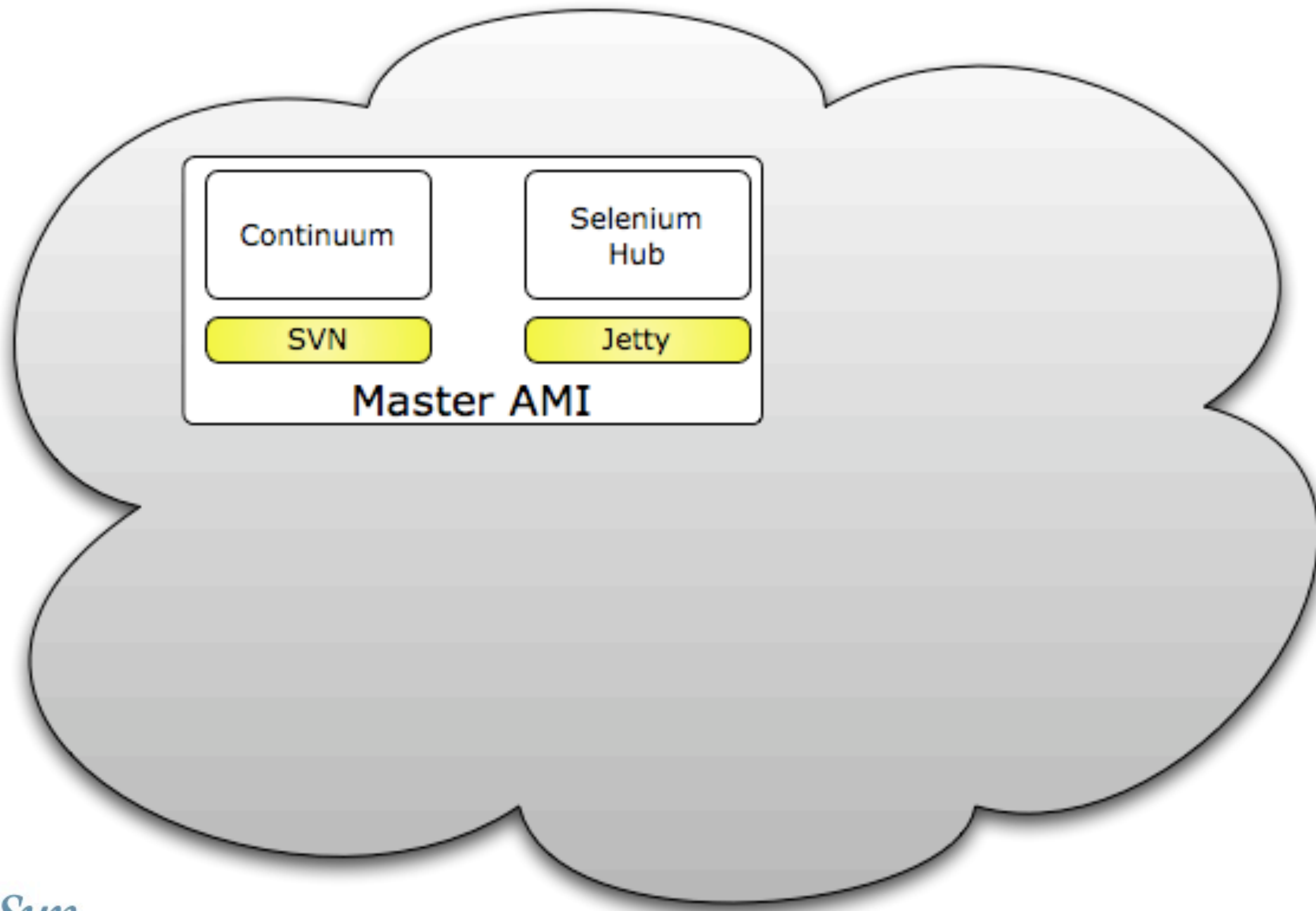
Starting point



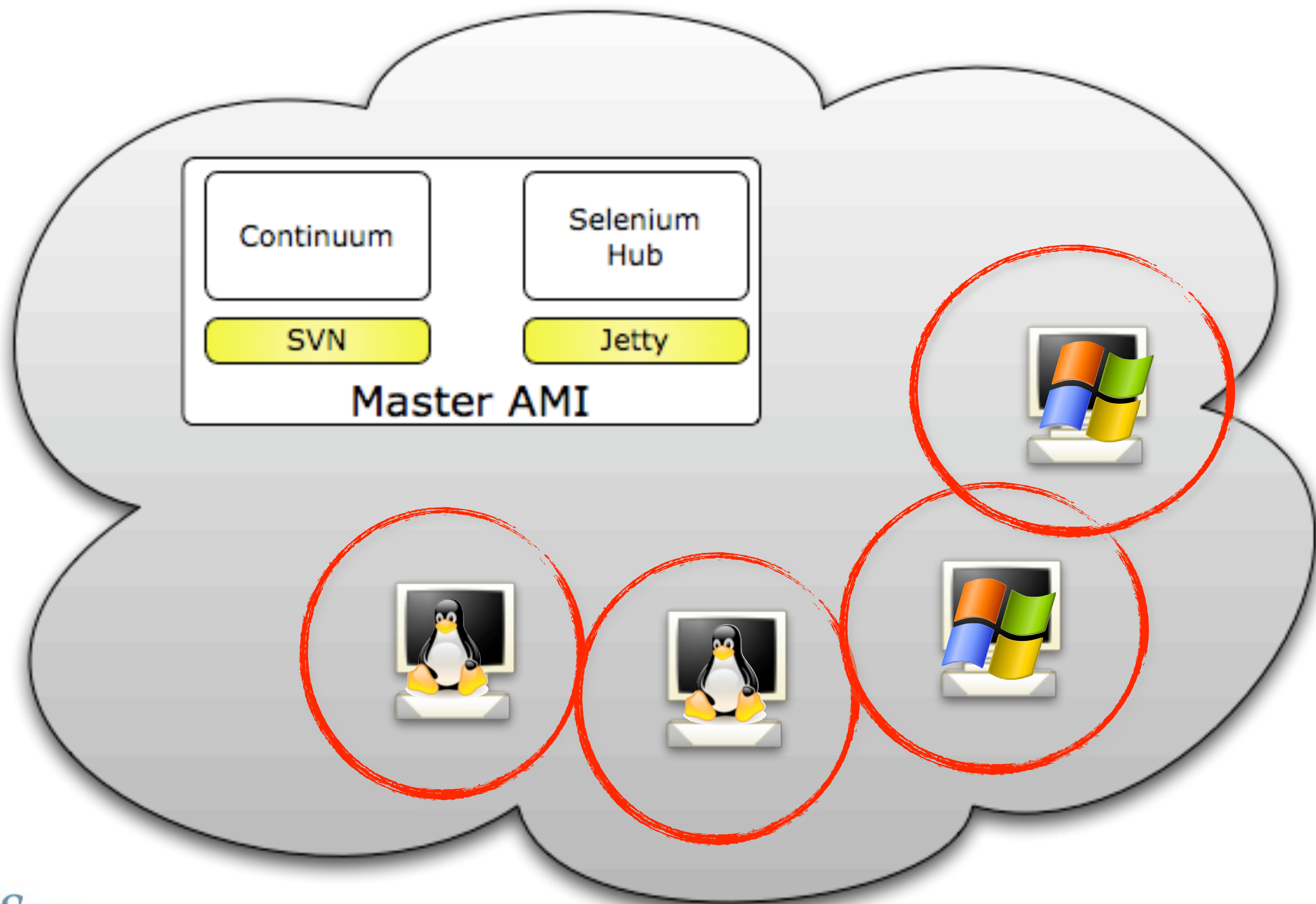
Build is triggered



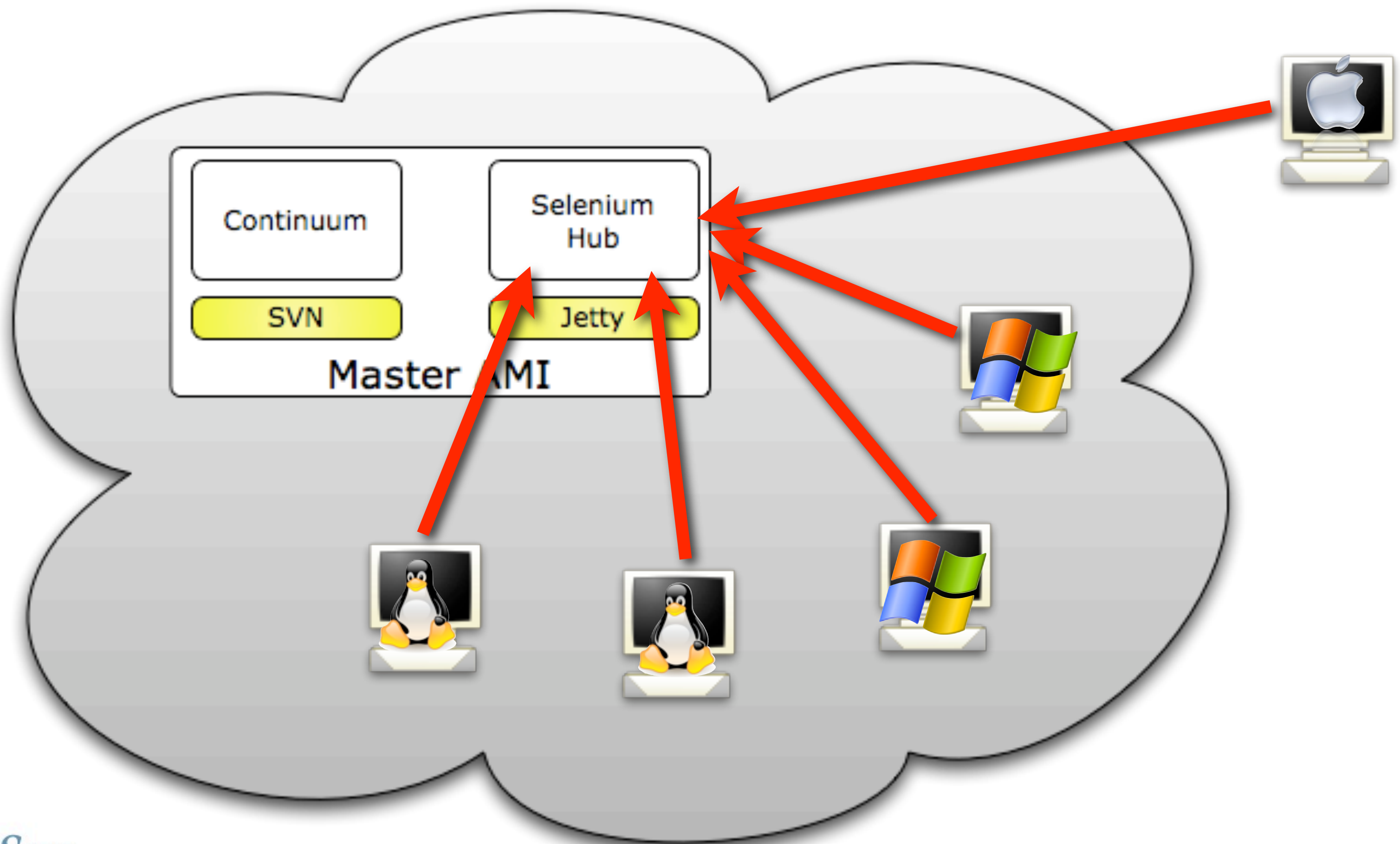
Start Images



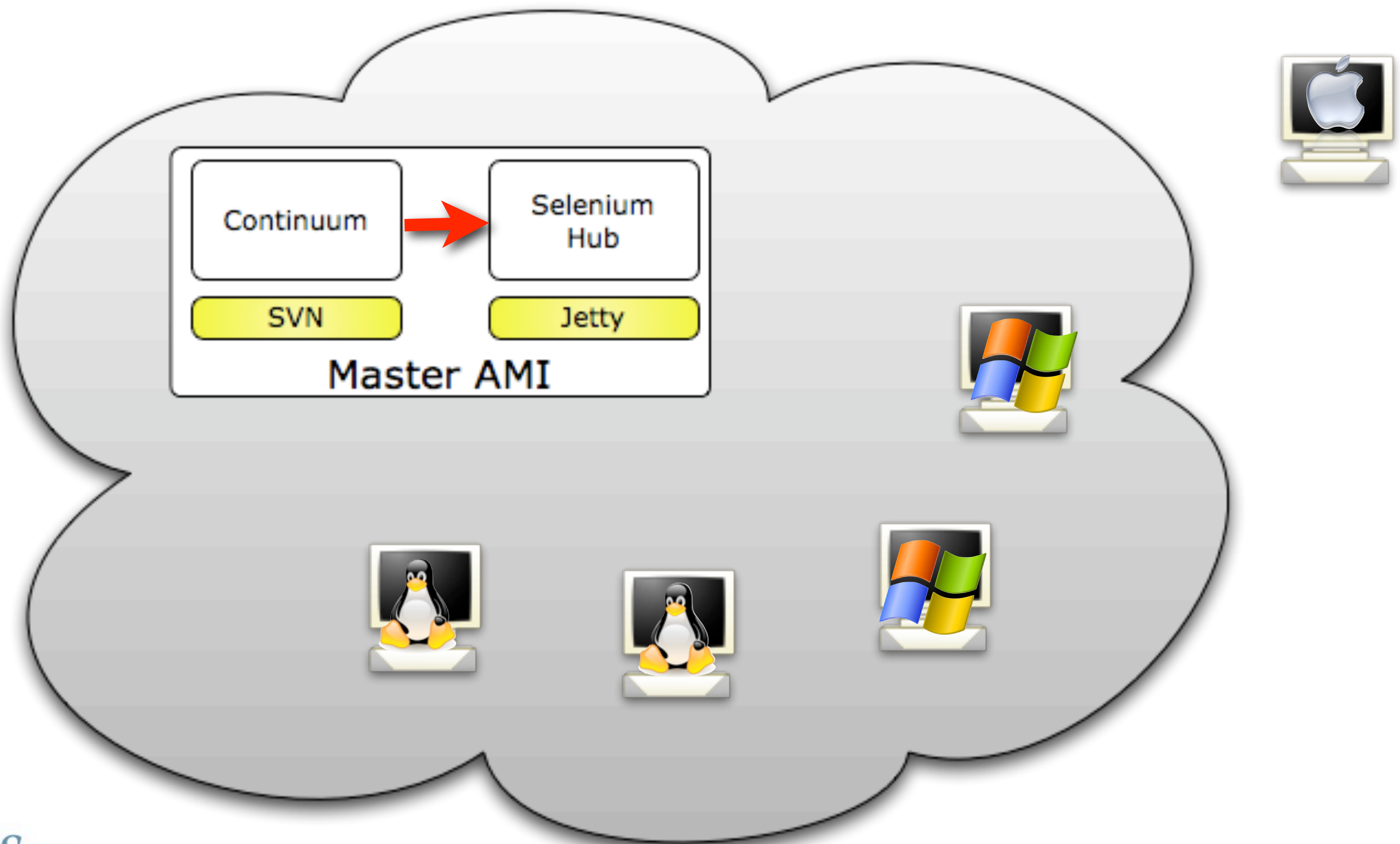
Start Images



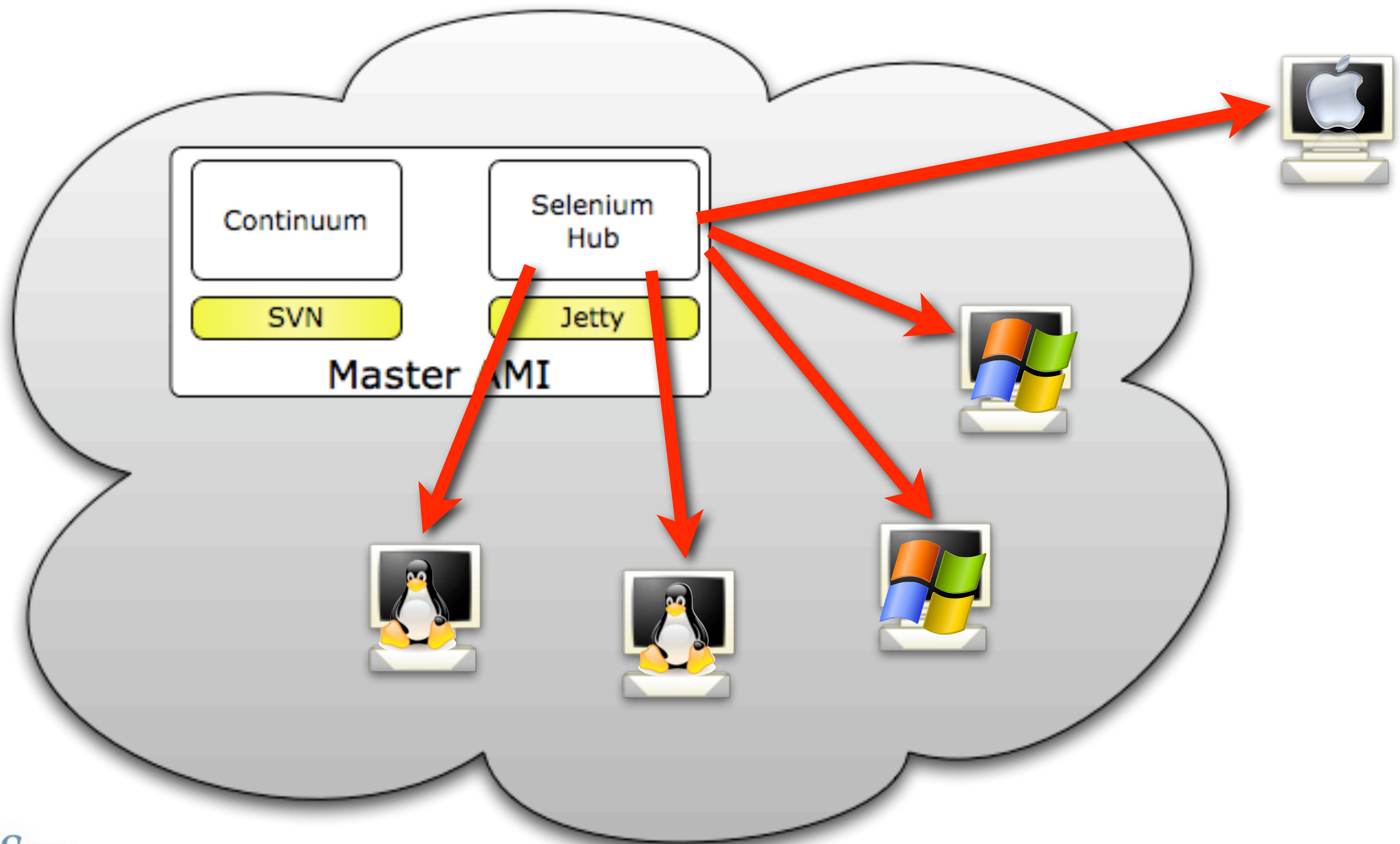
Register with Hub



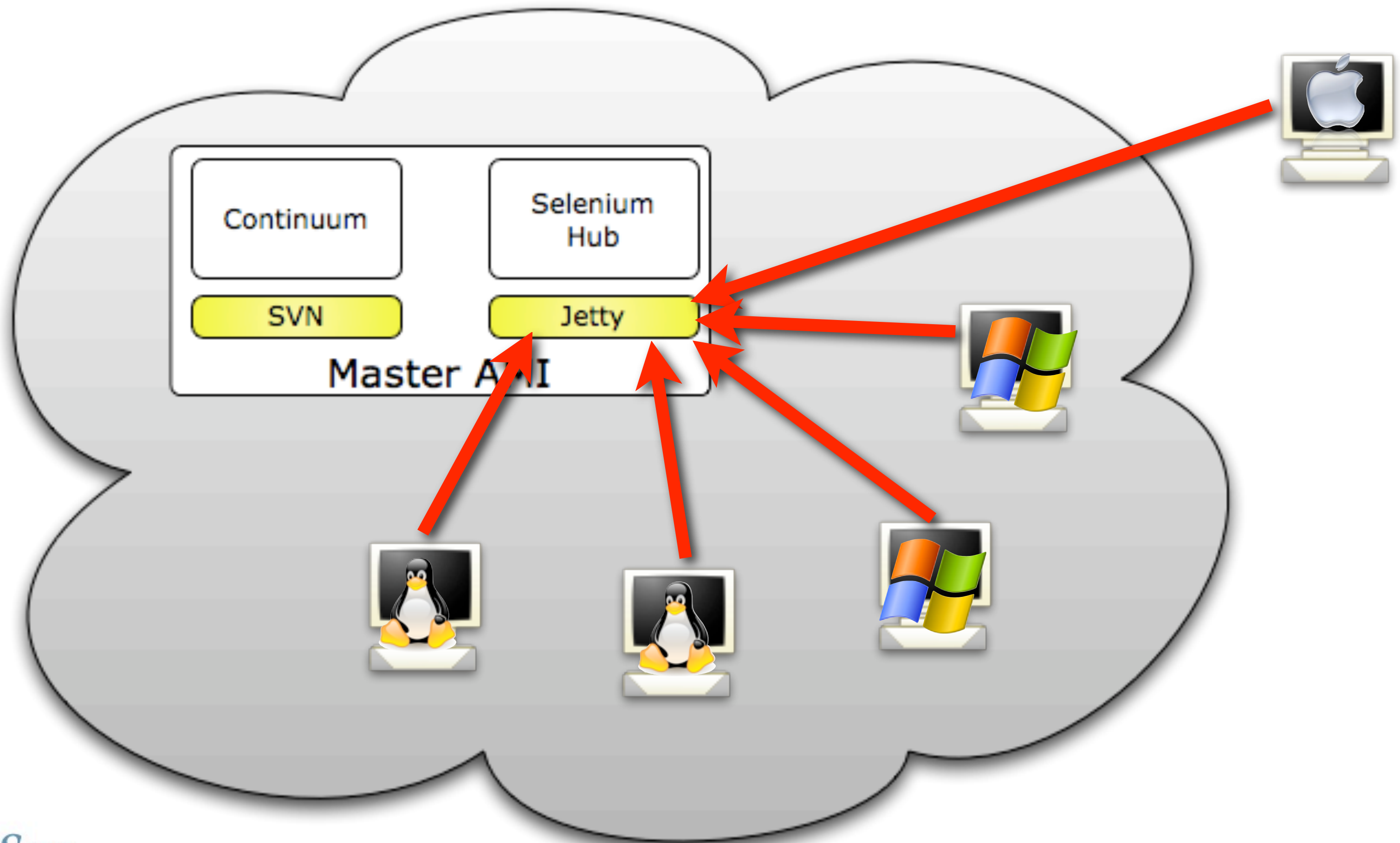
Tests call Selenium



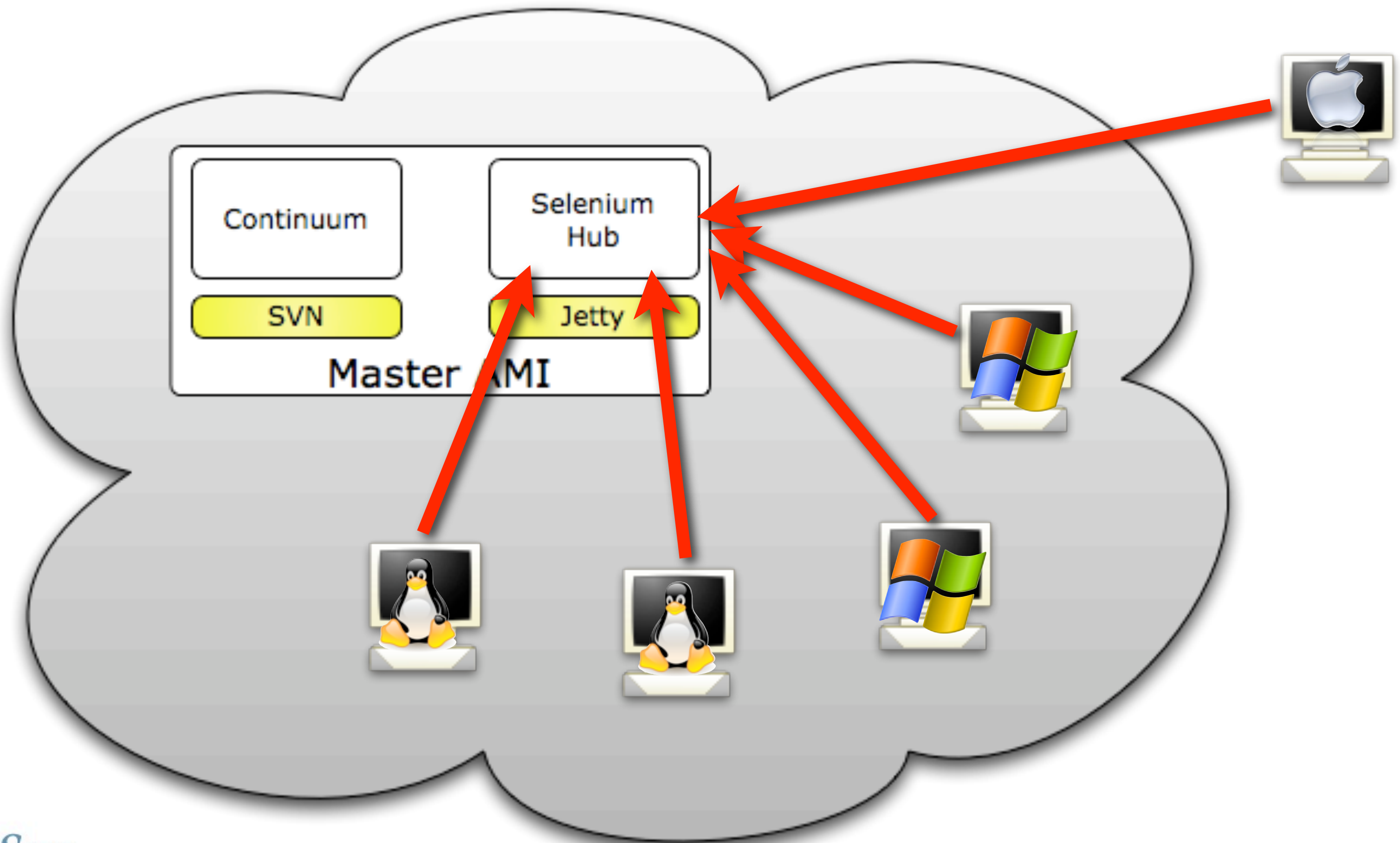
Distribute Tests



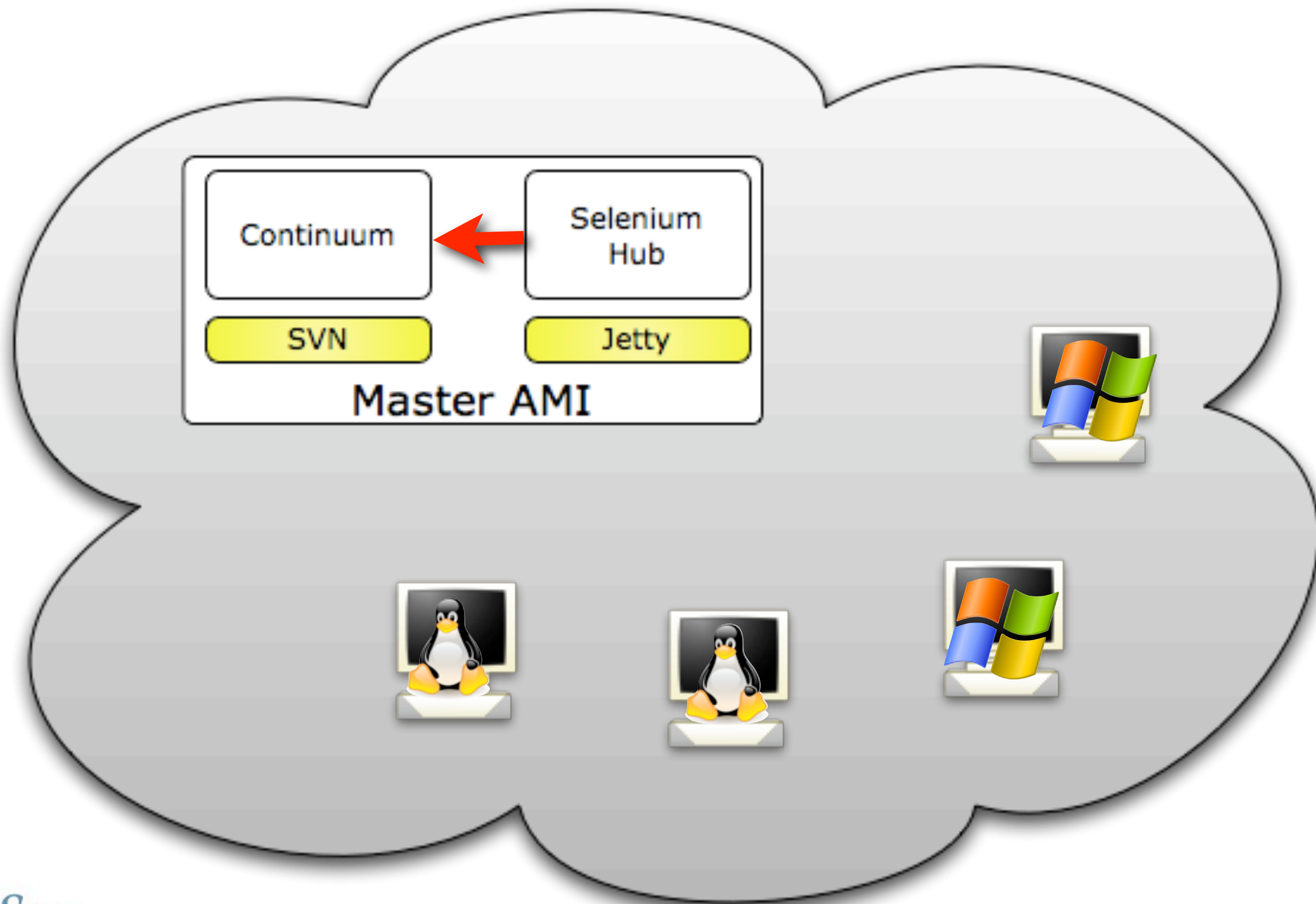
Run tests against webapp



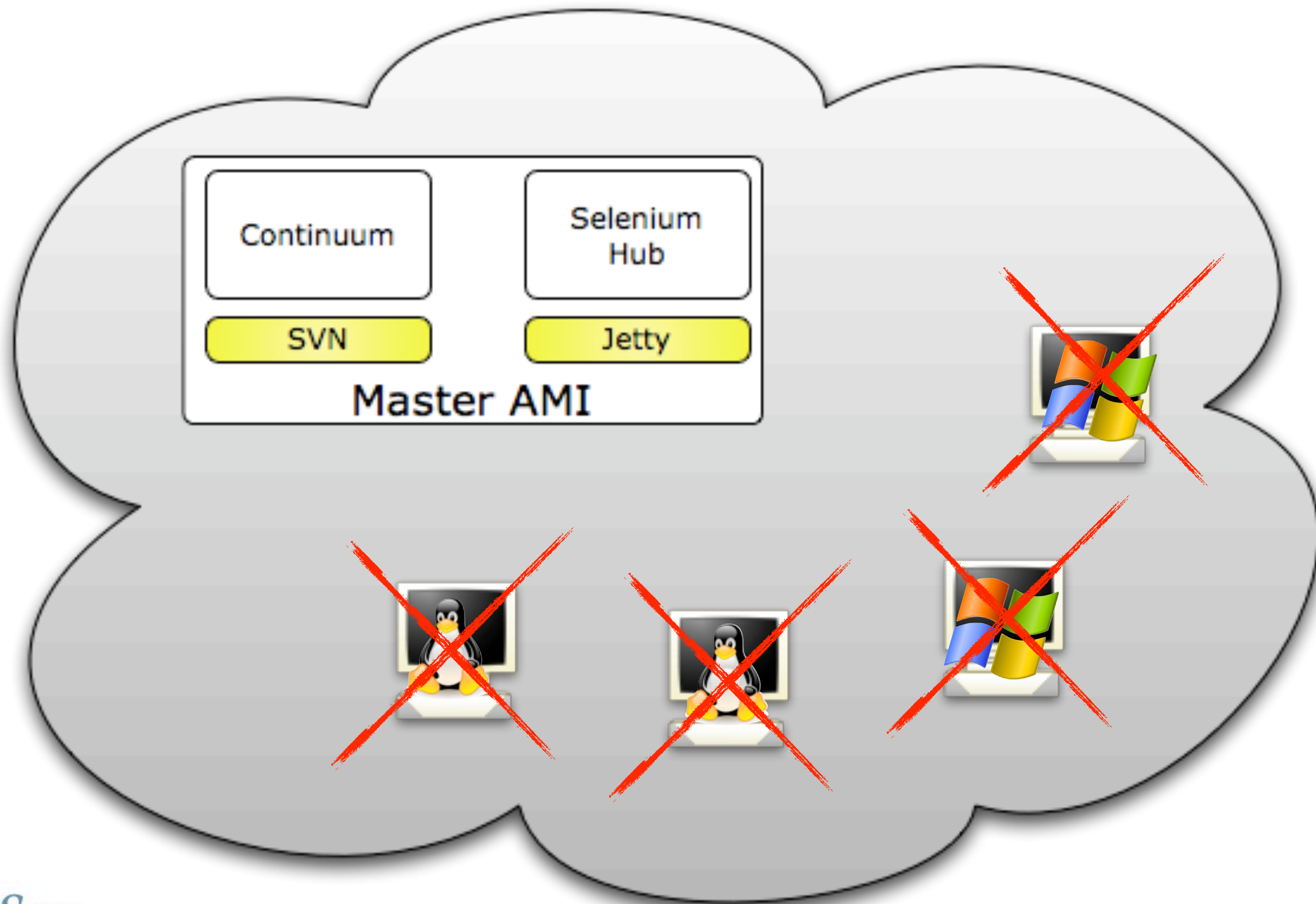
Return results to Hub



Return results to builder



Kill Images



G2iX

maestrodev

Release quality software and spend less building it.

Questions?



JavaOneSM

Thank You



Carlos Sanchez
csanchez@g2ix.com
carlos@apache.org

<http://www.carlossanchez.eu>

Twitter: **csanchez**

