



National Snow and Ice Data Center
Supporting Cryospheric Research Since 1976



Continuous Integration and Delivery at NSIDC

Julia Collins

**National Snow and Ice Data Center
Cooperative Institute for Research in Environmental
Sciences, University of Colorado, Boulder, CO**

**UCAR SEA 2014
7 April 2014**

The Goal

**Continuous integration
and
Continuous delivery**

?

Definitions

Continuous integration: Merge changes frequently into shared code base (and run unit/integration tests) to catch conflicts early.

Requires:

- * **Code repository**
- * **Automated build (build all commits)**
- * **Automated tests**
- * **If fail, fail fast**

Definitions

Continuous delivery: A software development discipline where you build software in such a way that the software can be released to production at any time.

<http://martinfowler.com/bliki/ContinuousDelivery.html>

Definitions

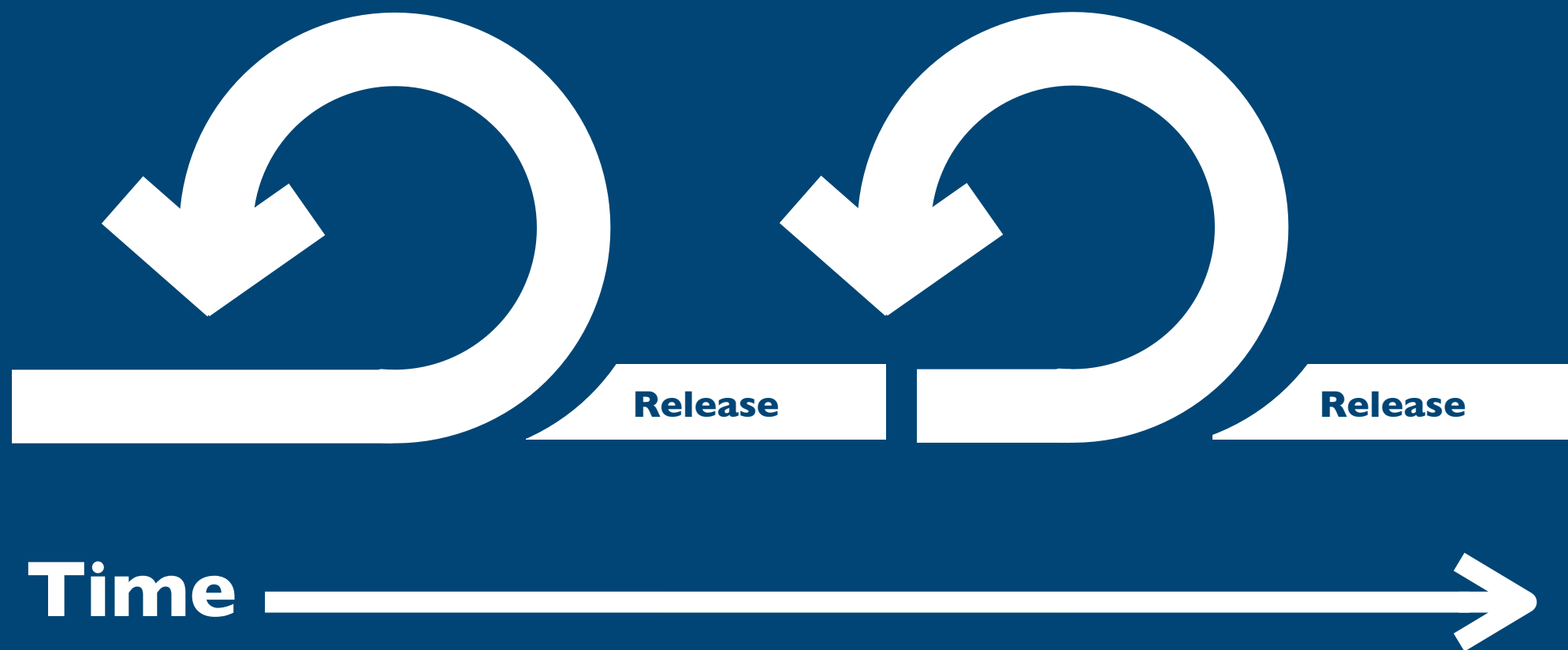
Continuous deployment Passing builds are
automatically deployed to the *production*
environment



<http://continuousdelivery.com/2010/08/continuous-delivery-vs-continuous-deployment/>

Development Context

Iterative development with frequent feature releases



Jenkins (was Hudson)

The screenshot shows the Jenkins web interface for the 'Icebridge' project. The browser address bar indicates the URL is <https://scm.nsidc.org/jenkins/view/Icebridge/>. The Jenkins logo is visible at the top left, and the user 'collinsj' is logged in. The main content area displays a list of jobs with their status (Success, Failure, or Pending) and a table of build executor status.

Jobs List:

S	W	Name ↓
		Icebridge-nu-portal (1) quick fail
		Icebridge_Daily_Feedback_Report
		IceBridge Flight Report Tool 1 Compile and Deploy to Integration
		Icebridge Flight Report Tool 2 QA
		Icebridge Flight Report Tool 3 Prod
		Icebridge Ingest 0 test and build
		Icebridge Ingest 1 integration wrapper
		Icebridge Ingest 2 QA wrapper
		Icebridge Ingest 3 Staging Wrapper
		Icebridge Ingest Main Job
		Icebridge Manual stuff
		Icebridge NAS Ingest
		Icebridge Portal2 build test integrate
		Icebridge Portal 1 Compile and Deploy

Build Queue: No builds in the queue.

Build Executor Status:

#	Status
1	Idle
2	Idle
3	Idle
4	Idle
5	Idle
6	Idle
7	Idle
8	Idle
9	Idle
10	Idle

Jenkins

Open source (MIT license)

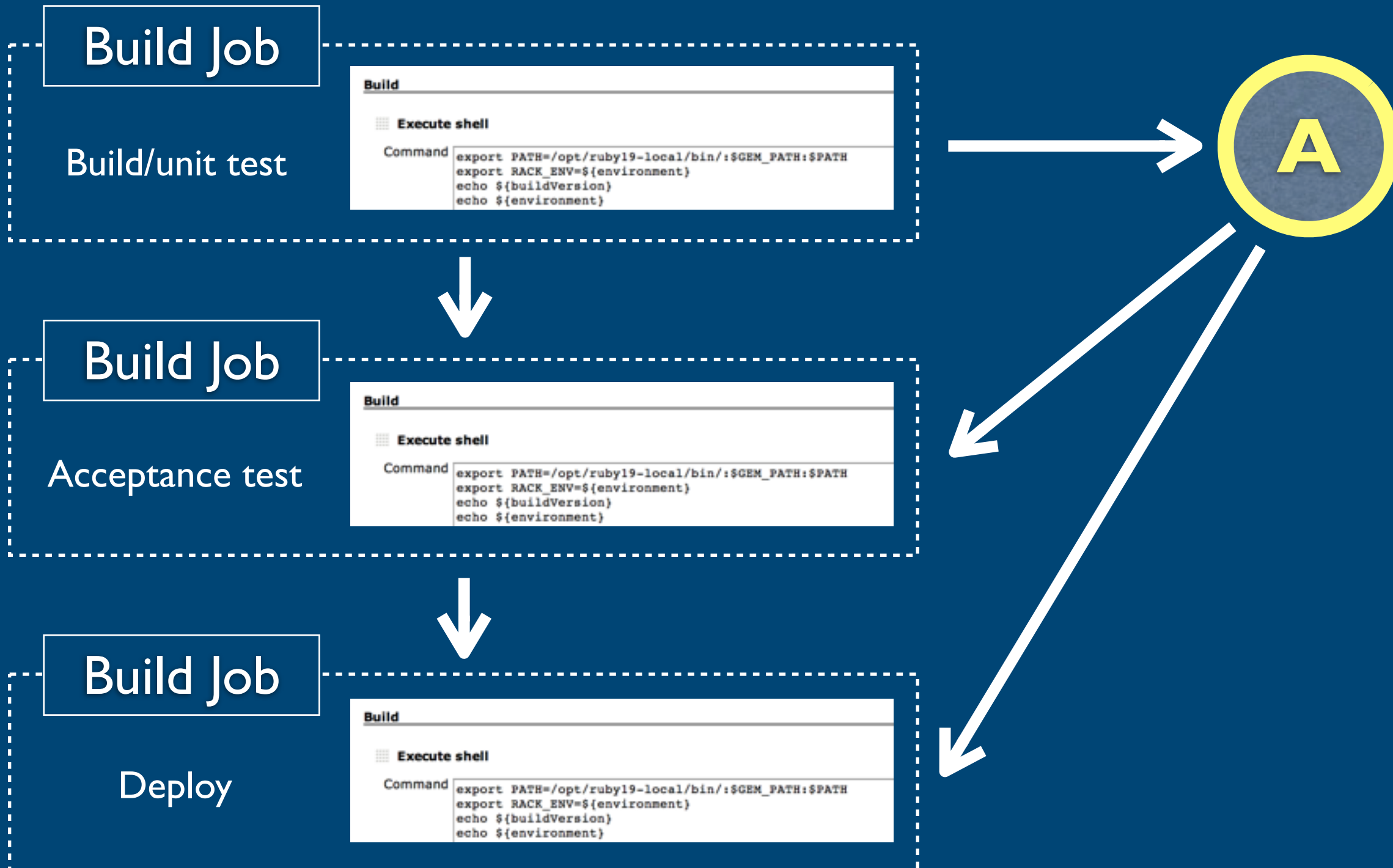
Core installation great for running scripts

Can run on a schedule, poll SCM for changes

Can be extended via plugins. Many (many many) plugins already exist.

Cons: Plugin developers not necessarily writing with other (unrelated) plugin behaviors in mind.

Jenkins Build



Another definition

“Artifact”

The by-product of the build; the testable entity

tar file

jar file

SCM tag or revision number

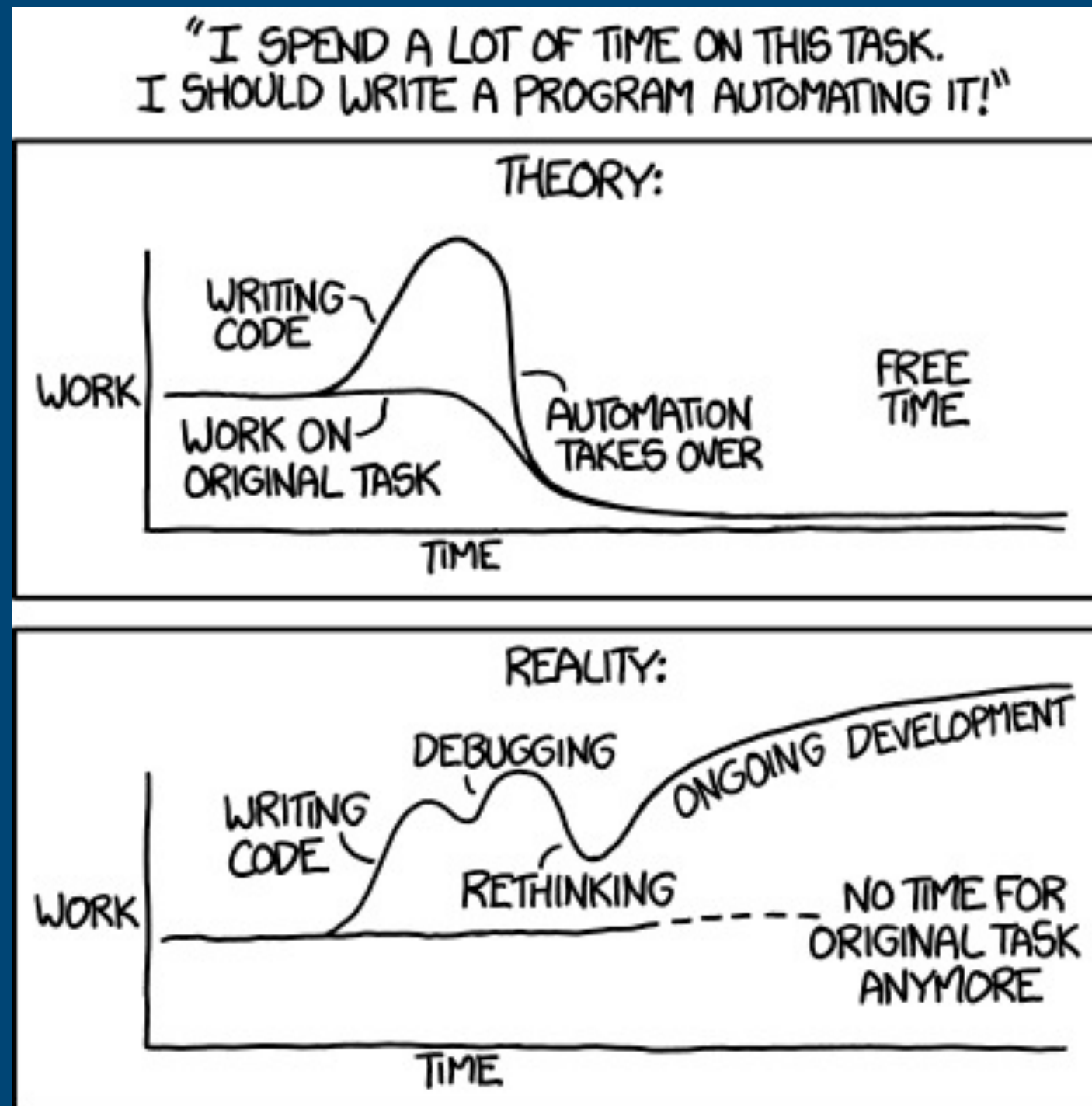
a running virtual machine

Artifacts

We had no artifact repository

- * Solution: Roll our own**
- * Result: Overhead added to job configurations to support the artifact handling**

One day you look around . . .



But really, automation is good

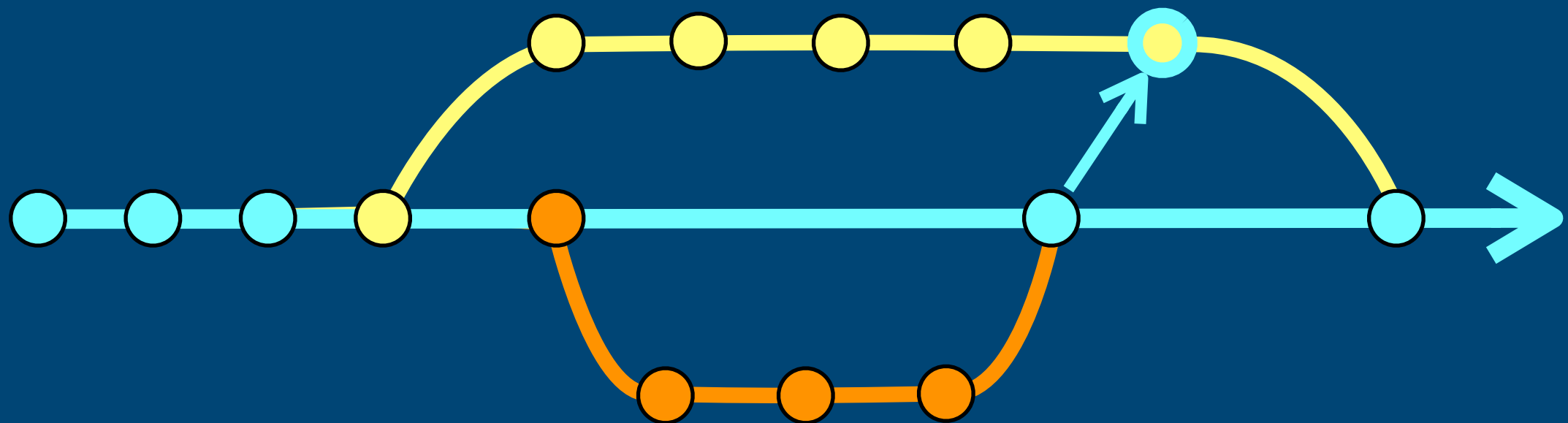
- * Consistent steps, no matter who's running the build**
- * History and reproducibility**
- * Build configuration conveys information about application environment and structure**

Meanwhile . . .

Moving from svn towards git

- * Bitbucket repositories**
- * Take advantage of git workflows**

Branching Strategies



- Master (trunk): Always in deployable state
- Feature A
- Feature B
- Merge master into Feature A

Many jobs, one build server

- * Applications using different versions of same language or libraries**
- * Conflict between projects needing upgrades vs. those frozen at a particular version**
- * Would like option to build, test and run applications on different OS than default**

The Wish List

Feature workflow:

- * The ability to build/test/approve a feature branch.**
- * The ability to merge master into a branch prior to build/test.**
- * The ability to merge all successfully integrated (built/tested) feature branches into master for review within the QA environment.**
- * The ability to push the local master (with all approved feature changes) to origin/master on the remote repository, along with tags indicating released code.**

The Wish List

Build and deployment:

- * The ability to create multiple build environments to satisfy different application needs for language versions and supporting libraries.**
- * A continuous build environment that produces/ manages deployment-ready artifacts, including managing the state of approval (or not).**
- * Deployment: Ops must be able to identify the approved artifact(s) and run the required steps for deployment.**
- * Deployment: Ops must be able to easily roll back to a previous release.**

Road Map

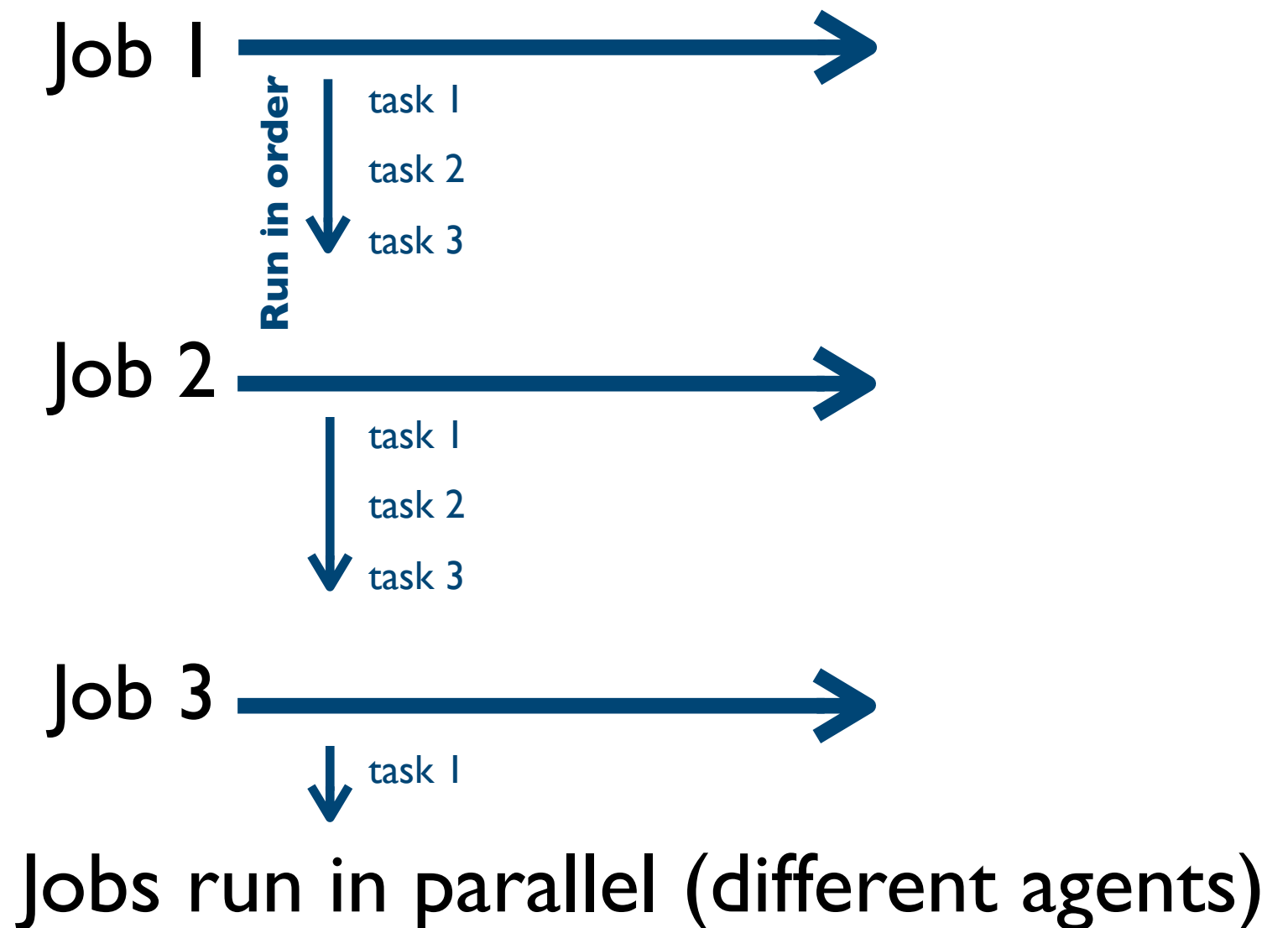
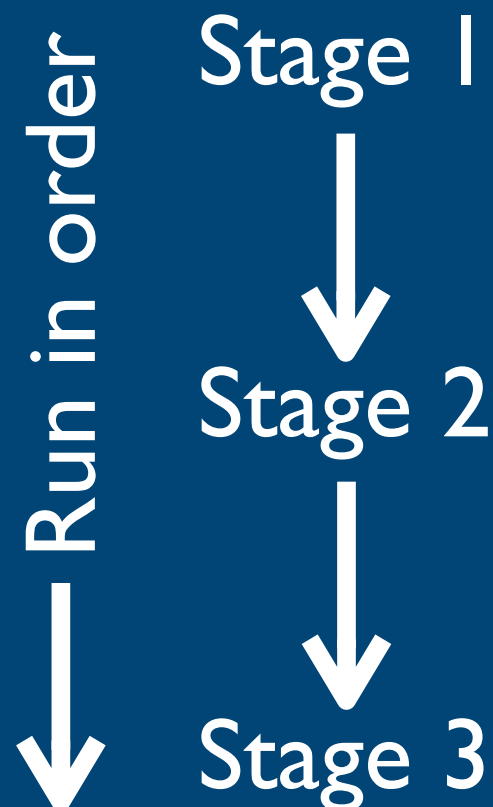
- * **Stick with Jenkins, OR**
- * **Transition to a different build server**

Jenkins in the future

- * Use existing plugins to provide needed functionality (artifact handling, master/slave implementation for multiple build agents)**
- * Contribute to plugin development**
- * Move in-line shell commands to application task runners (remove redundancies)**

Bamboo (Atlassian)

Build Plan



Bamboo

The screenshot displays the Atlassian Bamboo web interface for editing a plan configuration. The browser window title is "IceBridge Portal 2.0 - master: Edit plan configuration - Atlassian Bamboo". The address bar shows the URL "bamboo.nsidc.org:8085/chain/admin/config/configureBranches.action?buildKey=IBP-COM". The left sidebar contains a navigation menu with the following items: Unit Tests, Prepare package, Prepare dist, Deploy to integration, Copy to integration, Acceptance Test, Run Acceptance Tests, and Branches (which is highlighted and shows a count of 3). The main content area is titled "Automatically manage branches" and includes a description: "Bamboo will automatically create plan branches when it detects new branches in your version control system. And automatically remove plan branches from Bamboo after a specified period of inactivity. Detection of new branches runs every 5 minutes." Below this, there are two sections: "Automatically manage branches" and "Merging". The "Automatically manage branches" section has a checkbox "Automatically manage branches" which is checked. It also includes a text input field for "Only create branches that match" with the value "feature-*" and a "Remove after" field set to "30 Days". The "Merging" section has a checkbox "Branch merging enabled" which is checked. It contains two sub-sections: "Branch updater" and "Gatekeeper". The "Branch updater" section has a "Checkout" field set to "Current branch", a "Merge from" dropdown set to "master", a "Build" field set to "Merge result", and a "Push on" checkbox which is checked. The "Gatekeeper" section has a "Checkout" dropdown set to "master", a "Merge from" field set to "Current branch", a "Build" field set to "Merge result", and a "Push on" checkbox which is checked. The "Notifications" section at the bottom has a description: "Notification preferences chosen below will be applied to any new plan branches created either automatically or manually." It includes two radio buttons: "Notify committers and people who have favoured the created branch" and "Use the plan's notification settings", with the latter being selected. A yellow banner at the bottom of the page states: "Powered by a free Atlassian Bamboo evaluation license. Please consider purchasing it today."

Bamboo

The screenshot shows the Atlassian Bamboo web interface. The browser address bar displays the URL: `bamboo.nsidc.org:8085/deploy/viewDeploymentProjectVersions.action?id=2752514`. The page title is "IceBridge Portal 2.0 - Deploy: Releases - Atlassian Bamboo". The navigation bar includes "Bamboo", "My Bamboo", "Build", "Deploy", "Reports", and a "Create" button. The main content area shows the "Releases" tab for the project "IceBridge Portal 2.0 - Deploy". It includes a "Currently deployed" table and a "History" table. A warning banner at the bottom states: "Powered by a free Atlassian Bamboo evaluation license. Please consider purchasing it today."

Deployment projects

IceBridge Portal 2.0 - Deploy

Project summary Releases

Currently deployed

Version	Release branch	Flagged	Created	Deployed on
feature-geo-filter-1	feature-geo-filter	APPROVED	20 November 2013 09:50 AM	QA

History

Version	Release branch	Flagged	Created	Was deployed on
feature-geo-filter-1	feature-geo-filter	APPROVED	20 November 2013 09:50 AM	QA
feature-southern-hemisphere-28	feature-southern-hemis...		19 November 2013 11:17 PM	QA
feature-all-flight-reports-2	feature-all-flight-rep...		19 November 2013 10:50 PM	
feature-requirejs-2	feature-requirejs	APPROVED	19 November 2013 10:45 PM	QA
0.1.0-rc5	master	APPROVED	29 October 2013 11:34 AM	QA

Powered by a free Atlassian Bamboo evaluation license. Please consider purchasing it today.

Go to "http://bamboo.nsidc.org:8085/deploy/viewDeploymentProjectVersions.action?id=2752514"

The Critical Facts

Jenkins: “free”

TeamCity

- * **Free edition: 3 build agents, 20 build configurations**
- * **\$299: 1 additional build agent, 10 additional configs**
- * **\$1999+: Enterprise options**

Bamboo

- * **Hosted vs. local options**
- * **\$10: No remote agents, 10 jobs**
- * **\$800: 1 remote agent, unlimited jobs**
- * **\$2200: 4 remote agents, unlimited jobs**

Lessons Learned

- * Build configurations tend to proliferate.**
- * With some effort, Jenkins can support many requirements.**
- * It's not just using the right tool, it's how you use it.**

References

- **Jenkins:** <http://jenkins-ci.org>
- **TeamCity:** <http://www.jetbrains.com/teamcity/>
- **Bamboo:** <https://www.atlassian.com/software/bamboo>
- **CI:** <http://martinfowler.com/bliki/ContinuousDelivery.html>
- **CD:** <http://continuousdelivery.com/2010/08/continuous-delivery-vs-continuous-deployment/>

Julia Collins
collinsj@nsidc.org