Keeping Your CI / CD Pipeline as Fast as It Needs to Be

fromfragiletoagile.com
@AbrahamMarin
#ACCUConf

About Me

MENDELEY

About Me

About Me

About Me

LJC

London Java Community

About Me

J.P.Morgan

OpenBet

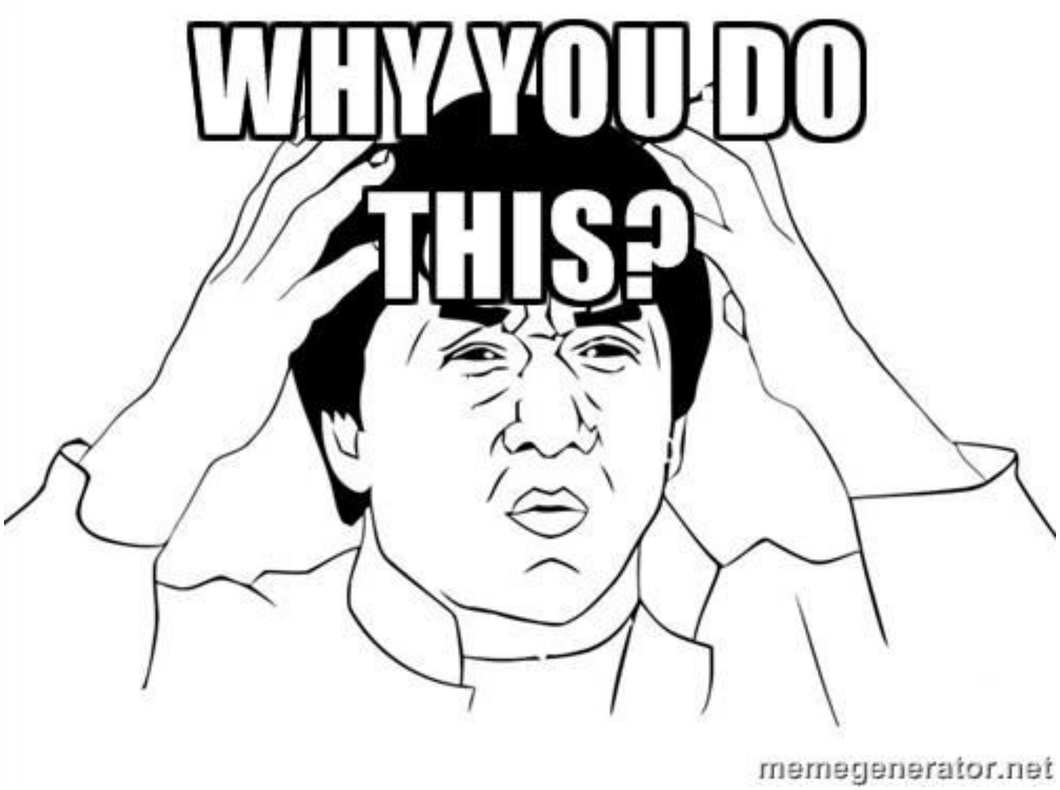
Tech Survey!

What is CI / CD?

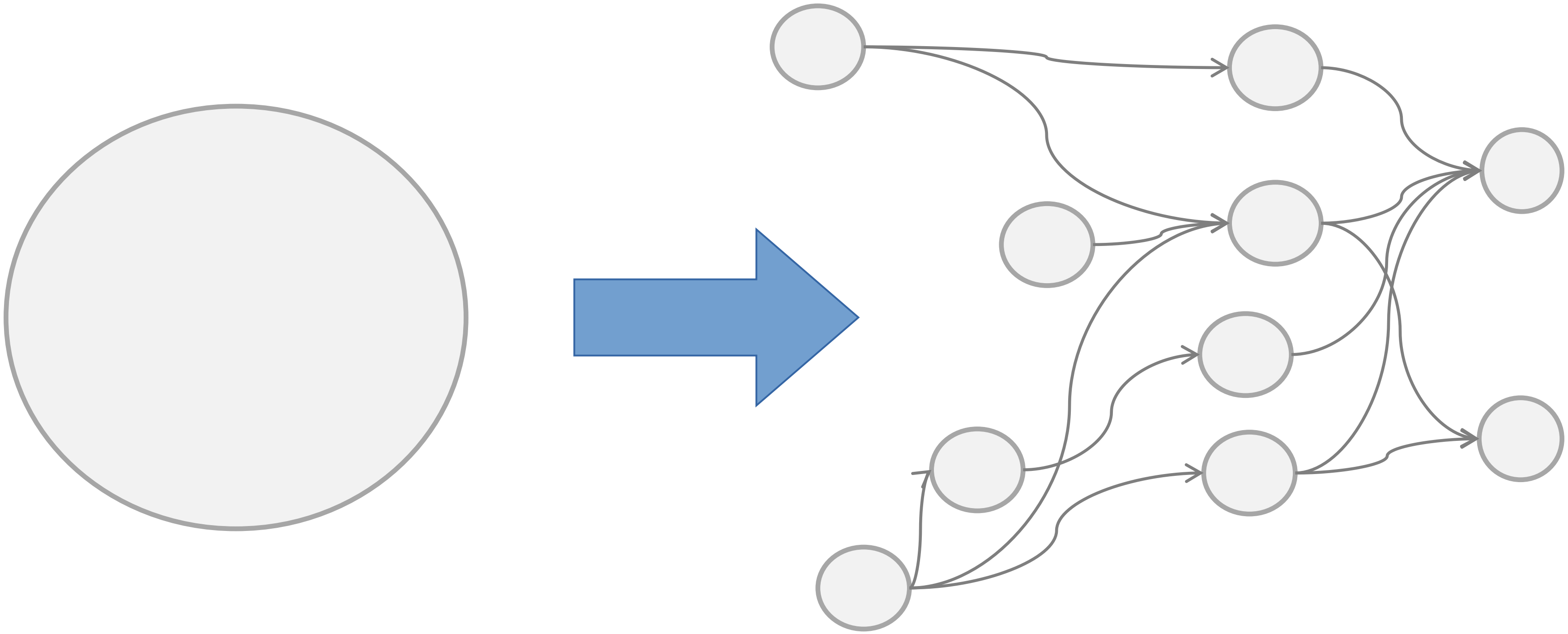
- Continuous Integration: check everything is still working after every commit
- Continuous Deployment: every successful commit turns into a release

About This Talk

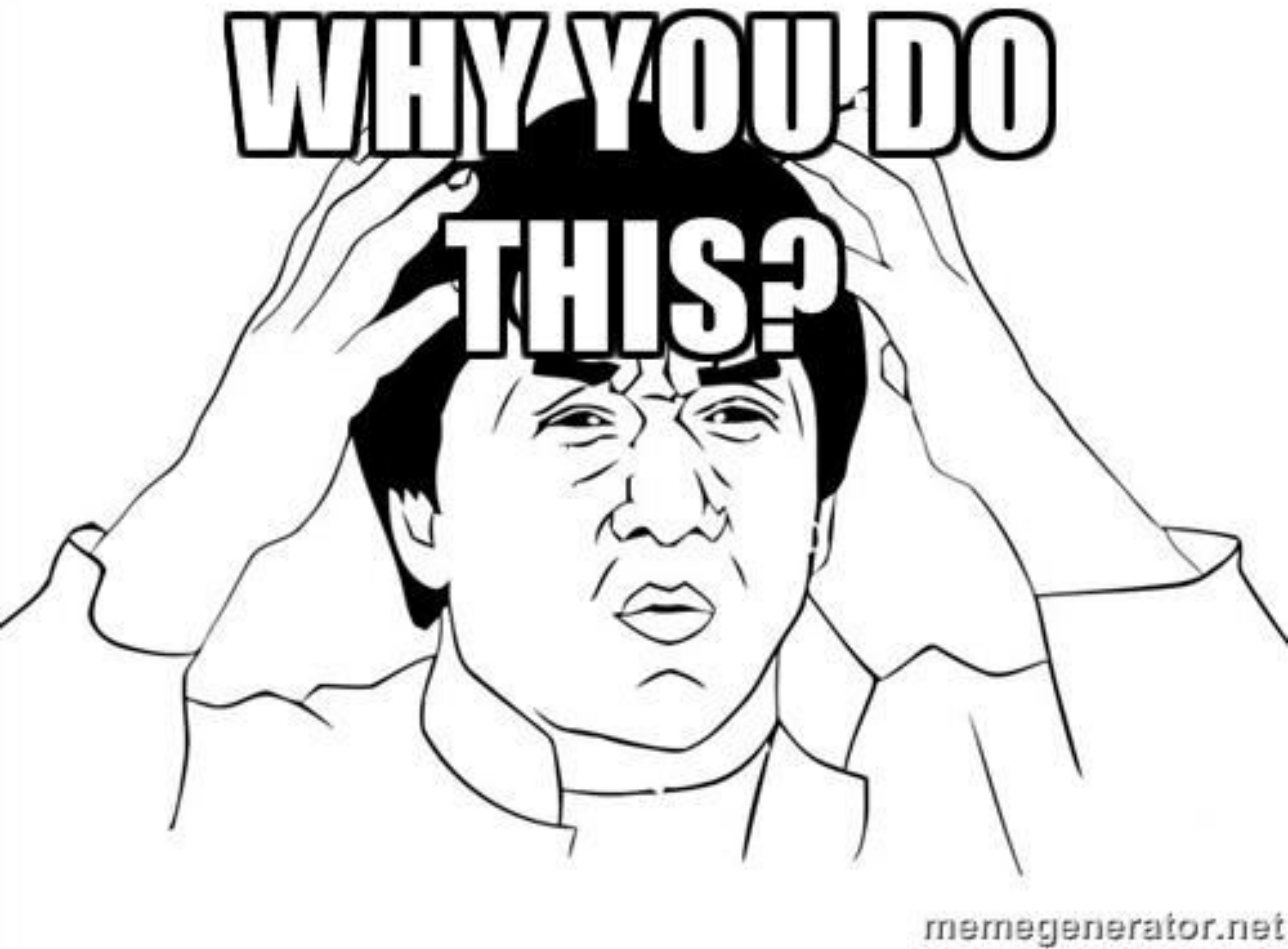
About This Talk

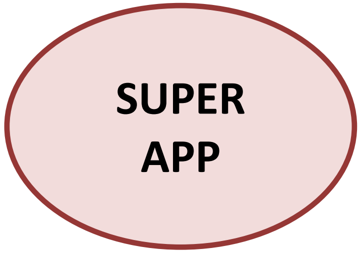


About This Talk



About This Talk


WHY YOU DO THIS?
memegenerator.net



SUPER
APP

# Files: 75
# Tests: 800
Build Time: 4 min
Output: **superapp.war**

SUPER APP

# Files: 113
# Tests: 1200
Build Time: 6 min
Output: **superapp.war**

SUPER APP

# Files: 169
# Tests: 1800
Build Time: 9 min
Output: **superapp.war**

The Problems Of Size

Slow feedback

Broken builds mask issues

Development paralysis

Impact on ability to meet our SLAs

Missed business opportunities

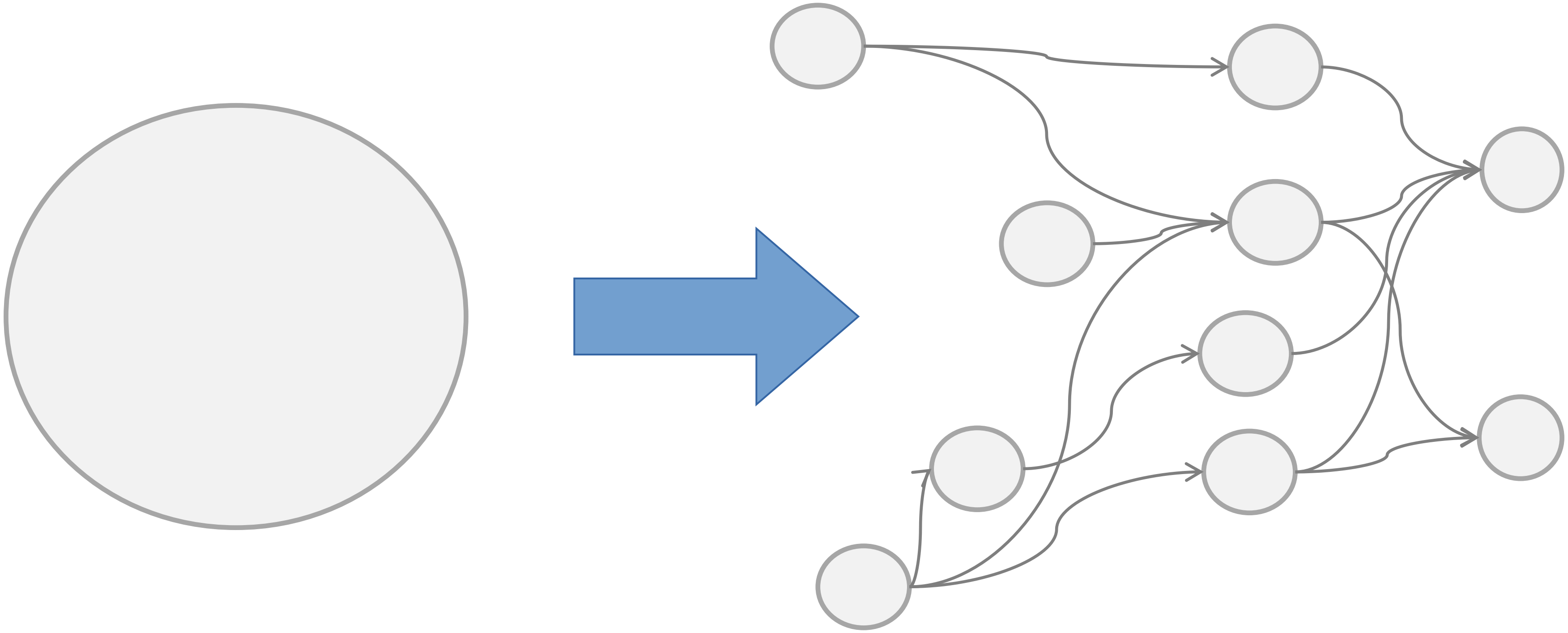
How Organisations Manage Size

Ignore it

Partial CD: only quick tests

Phased CD: split into components

Test Deprecation Policy



Maven™

TC
TeamCity

artifactory

SUBVERSION

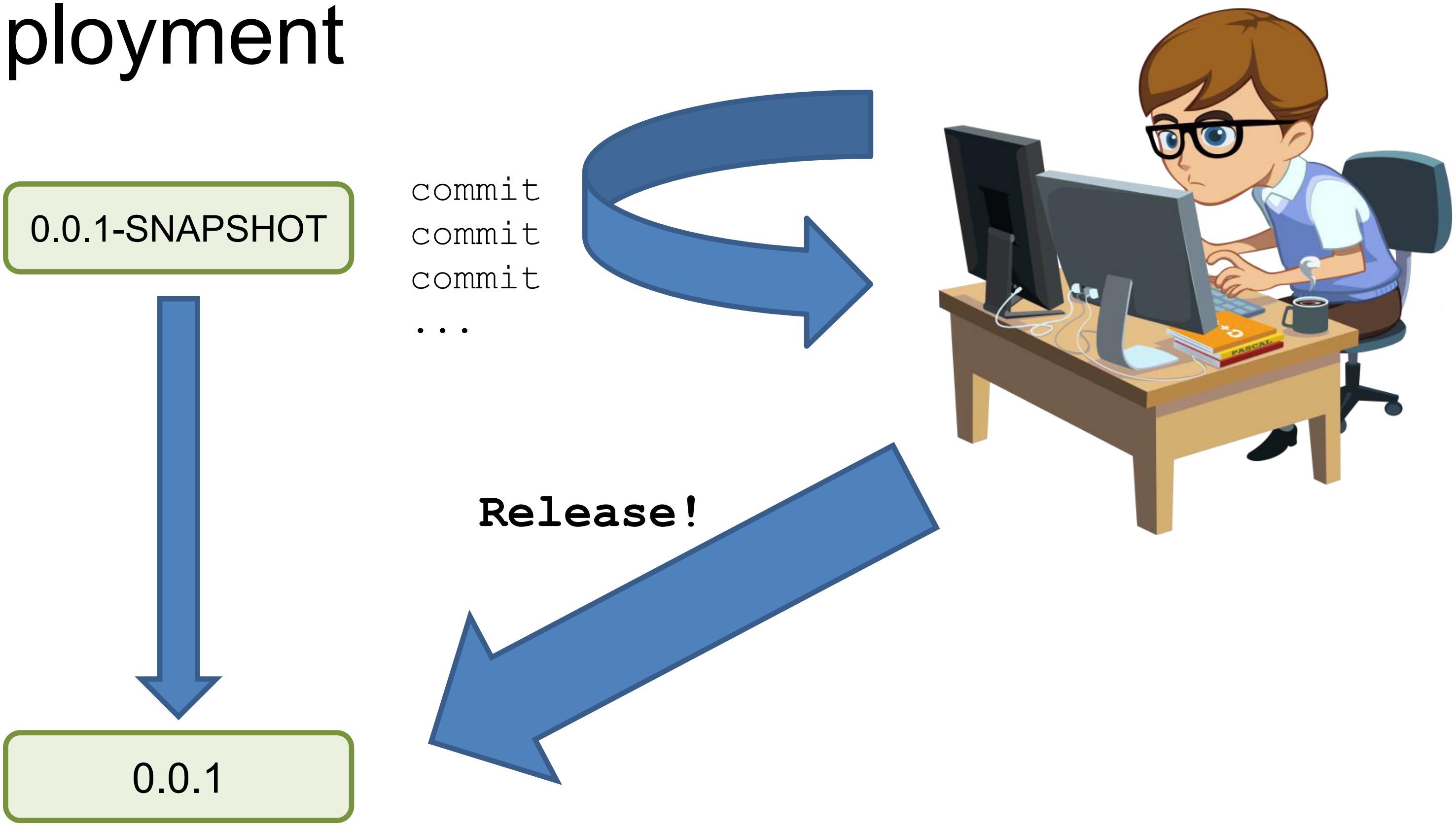
To Caesar what is Caesar’s

Based on John Ferguson Smart’s
“Real-World Strategies for Continuous Delivery with maven and Jenkins”

http://youtu.be/McTZtyb9M38

John’s approach

Maven wasn’t built for Continuous Deployment



0.0.1-SNAPSHOT

```
commit
commit
commit
...
```

Release!

0.0.1

John’s approach

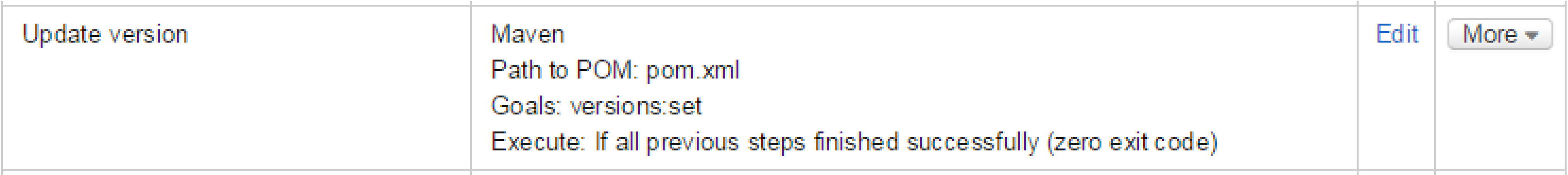
Don’t use RELEASE plugin
Use VERSIONS plugin

Set version to `<version scheme>.<build number>`
Run `mvn deploy`
Commit pom file to repository

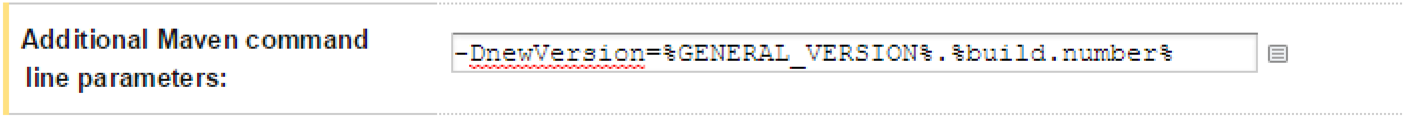
John’s approach

Set version to `<version scheme>.<build number>`

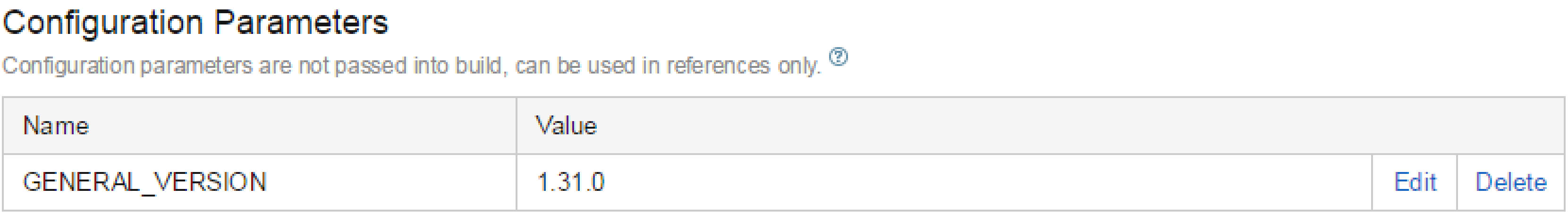
```
mvn versions:set -DnewVersion=**your version**
```



Update version	Maven Path to POM: pom.xml Goals: versions:set Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾



Additional Maven command line parameters: `-DnewVersion=%GENERAL_VERSION%.%build.number%`



Configuration Parameters

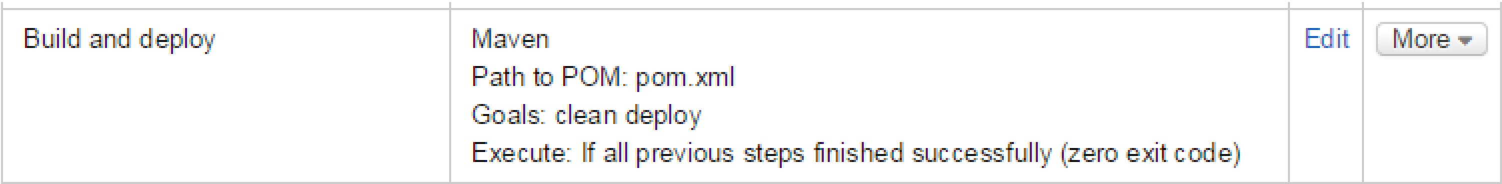
Configuration parameters are not passed into build, can be used in references only. ?

Name	Value		
GENERAL_VERSION	1.31.0	Edit	Delete

John’s approach

Run `mvn deploy`

```
mvn clean deploy
```



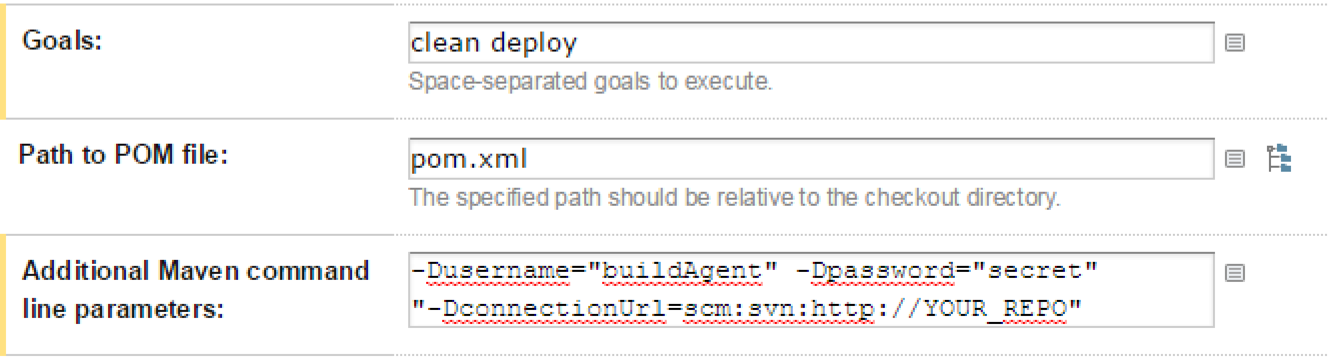
Build and deploy

Maven
Path to POM: pom.xml
Goals: clean deploy
Execute: If all previous steps finished successfully (zero exit code)

John’s approach

Commit pom file to repository

```
<plugin>
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-scm-plugin</artifactId>
    <version>1.9.1</version>
    <executions>
        <execution>
            <id>commit</id>
            <phase>deploy</phase>
            <goals>
                <goal>checkin</goal>
            </goals>
        </execution>
    </executions>
</plugin>
```



Goals: clean deploy

Space-separated goals to execute.

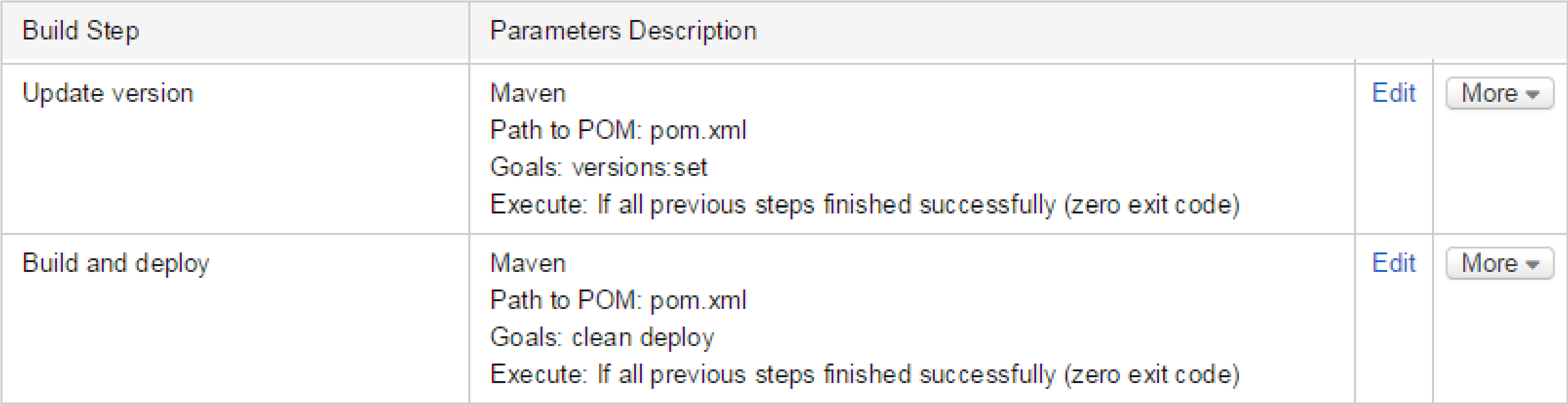
Path to POM file: pom.xml

The specified path should be relative to the checkout directory.

Additional Maven command line parameters:

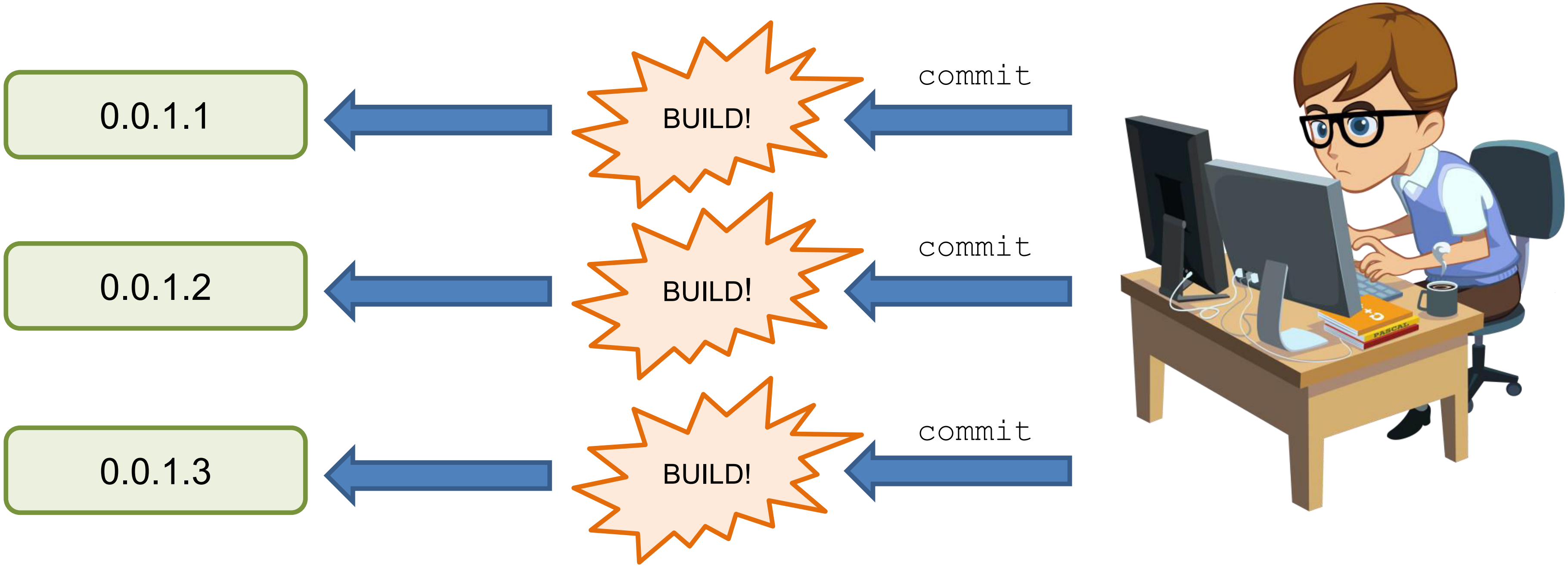
```
-Dusername="buildAgent" -Dpassword="secret"
"-DconnectionUrl=scm:svn:http://YOUR_REPO"
```

John’s approach



Build Step	Parameters Description		
Update version	Maven Path to POM: pom.xml Goals: versions:set Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾
Build and deploy	Maven Path to POM: pom.xml Goals: clean deploy Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾

John’s approach



0.0.1.1

BUILD!

commit

0.0.1.2

BUILD!

commit

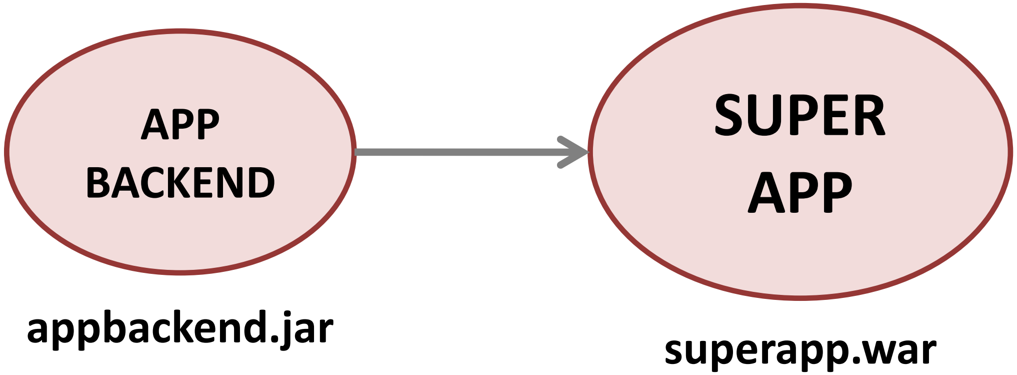
0.0.1.3

BUILD!

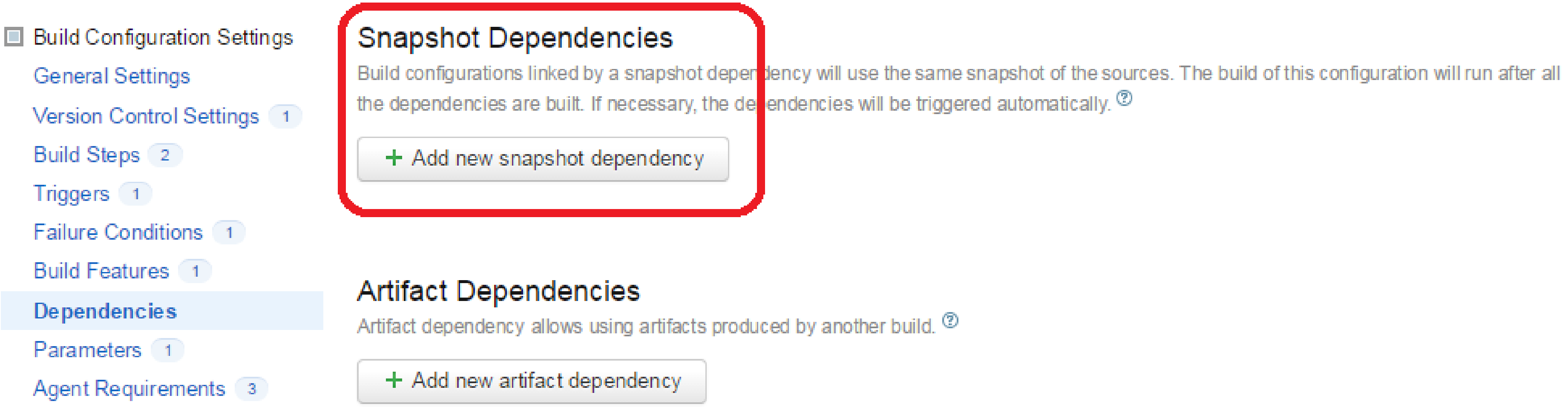
commit

How do you scale this?

Setting up dependencies



```xml
<dependency>
    <groupId>com.superappfactory</groupId>
    <artifactId>appbackend</artifactId>
    <version>LATEST</version>
</dependency>
```

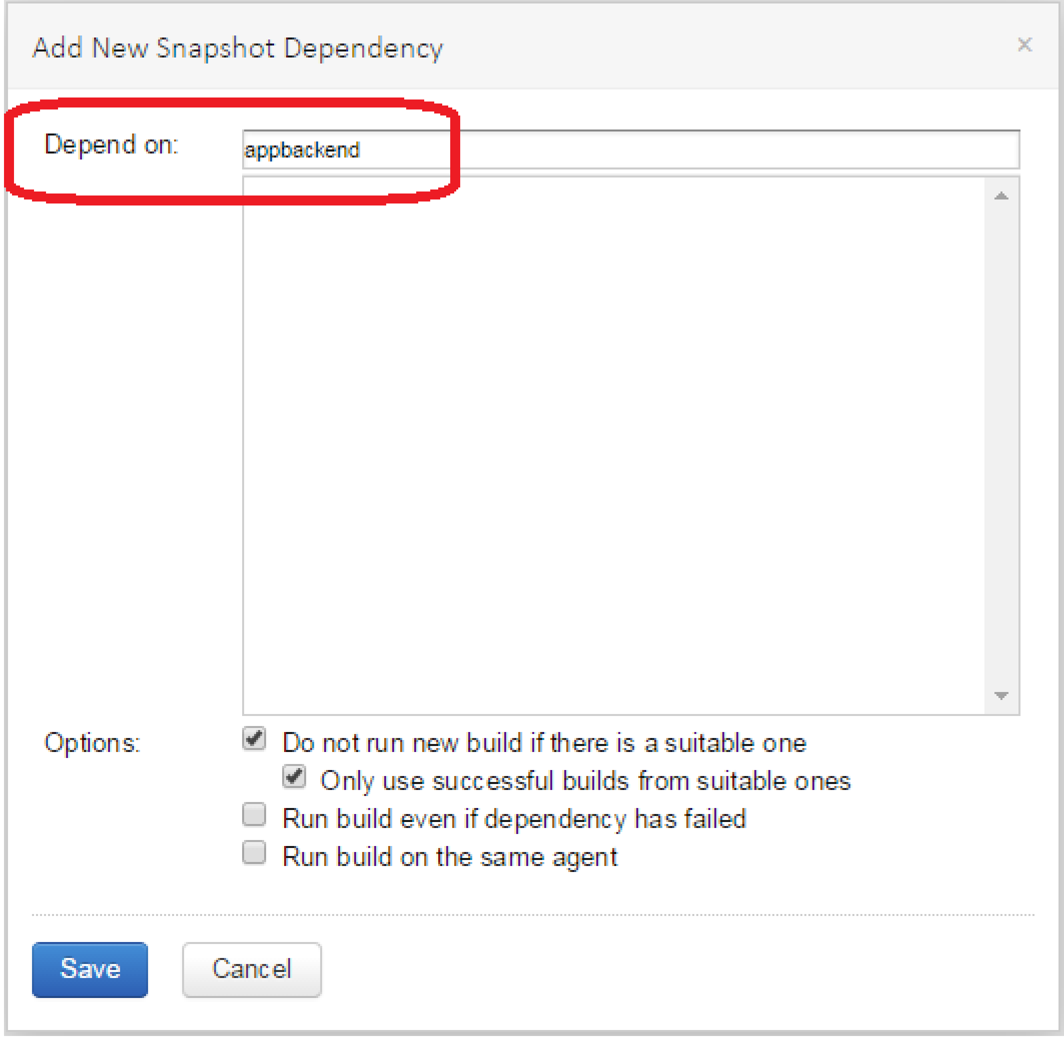


Snapshot Dependencies

Build configurations linked by a snapshot dependency will use the same snapshot of the sources. The build of this configuration will run after all the dependencies are built. If necessary, the dependencies will be triggered automatically.

Artifact Dependencies

Artifact dependency allows using artifacts produced by another build.



Add New Snapshot Dependency

Depend on: appbackend

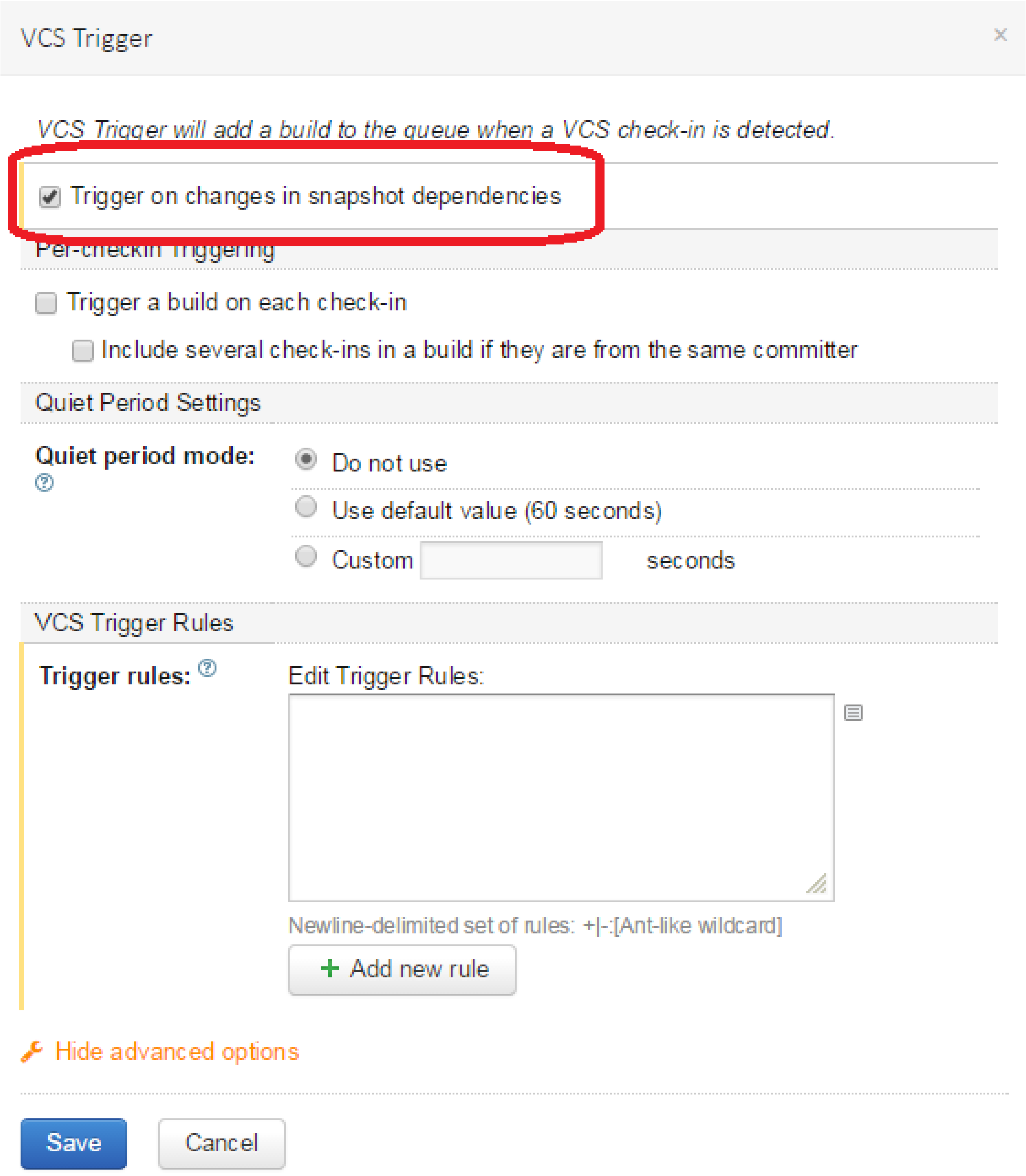
Options:

☑ Do not run new build if there is a suitable one

☑ Only use successful builds from suitable ones

☐ Run build even if dependency has failed

☐ Run build on the same agent



VCS Trigger

VCS Trigger will add a build to the queue when a VCS check-in is detected.

☑ Trigger on changes in snapshot dependencies

Per-checkin Triggering

□ Trigger a build on each check-in

□ Include several check-ins in a build if they are from the same committer

Quiet Period Settings

Quiet period mode:

● Do not use

○ Use default value (60 seconds)

○ Custom seconds

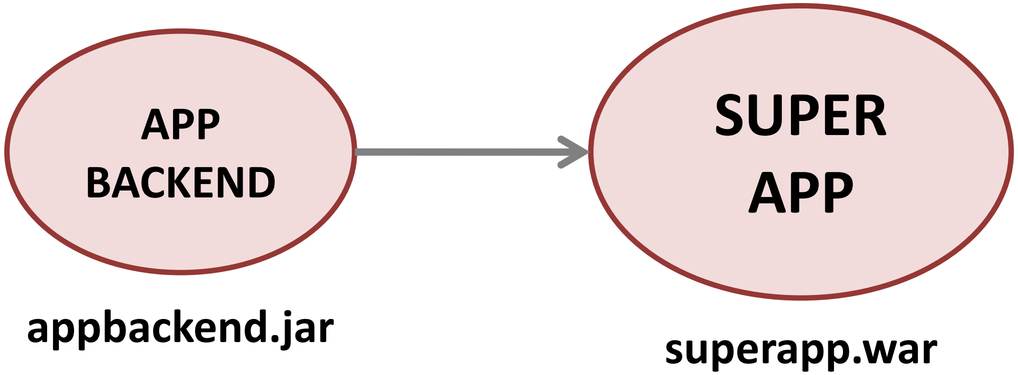
VCS Trigger Rules

Trigger rules:

Edit Trigger Rules:

Newline-delimited set of rules: +|-:[Ant-like wildcard]

Setting up dependencies



appbackend.jar

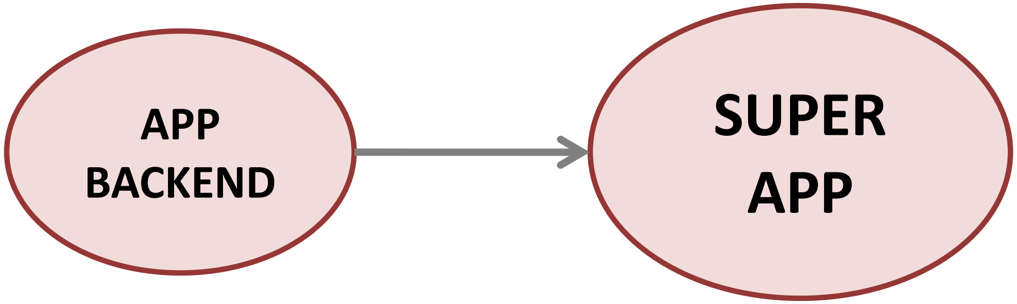
superapp.war

Rebuilding old versions

Rebuilding old versions

Using “LATEST” makes it impossible to build old versions correctly

Rebuilding old versions



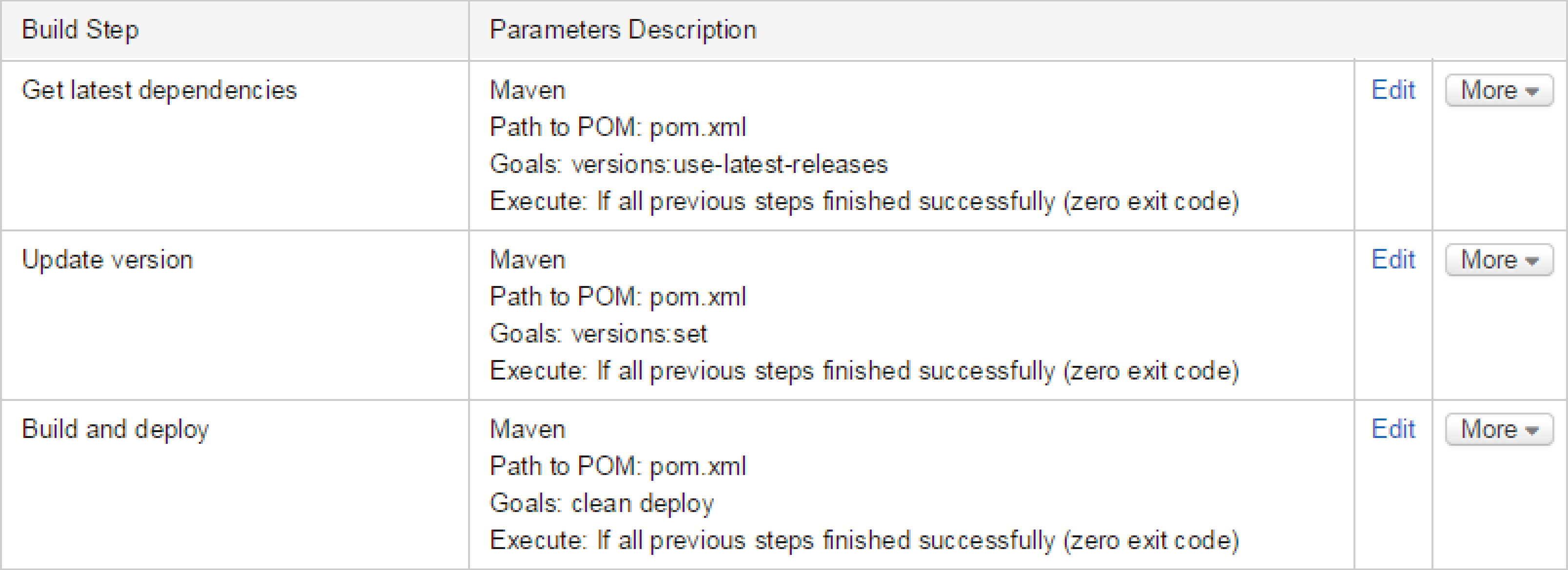
appbackend.jar

superapp.war

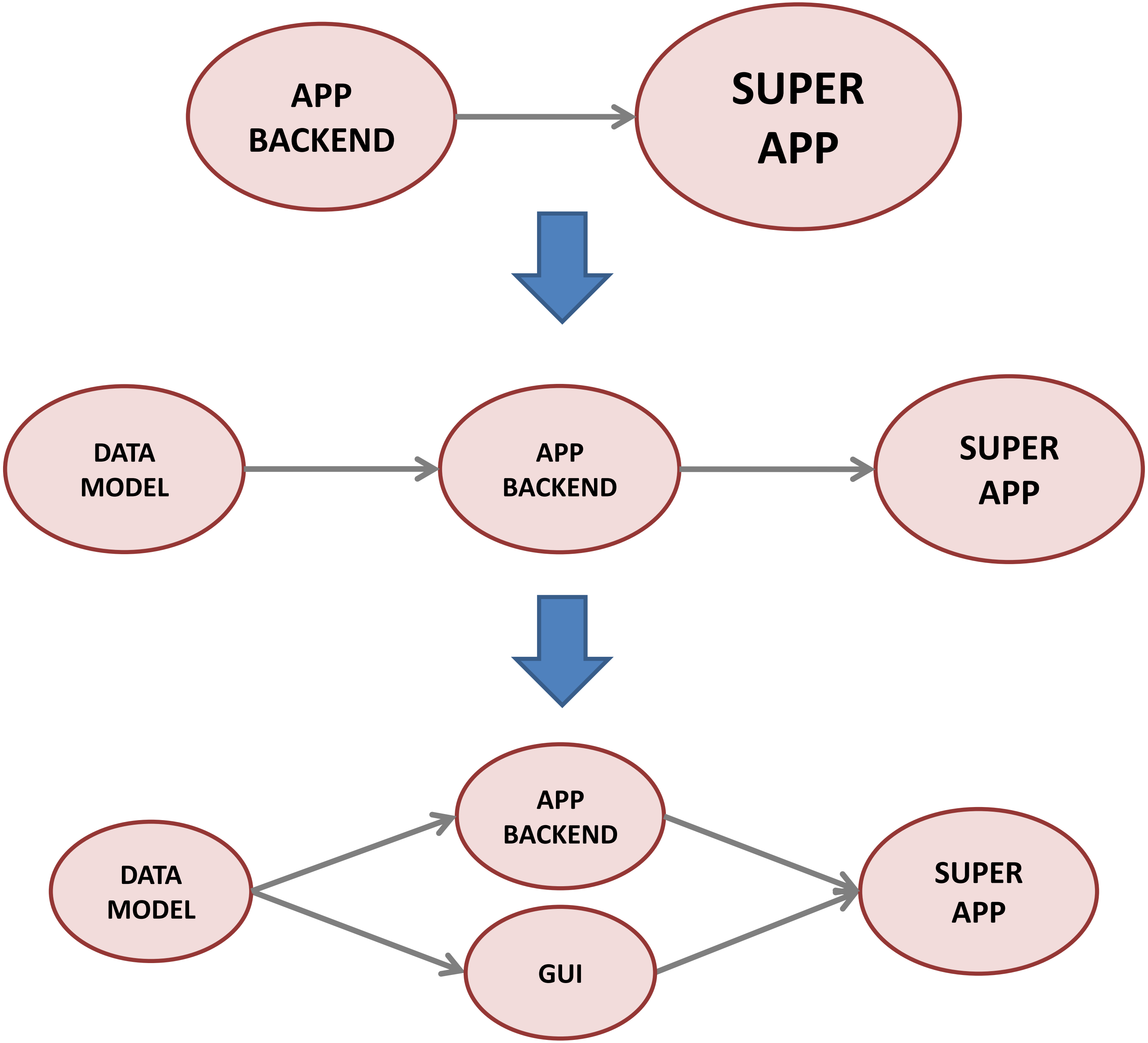
```
<dependency>
    <groupId>com.superappfactory</groupId>
    <artifactId>appbackend</artifactId>
    <version>???????</version>
</dependency>
```

Update versions of dependencies

```
mvn versions:use-latest-releases
```



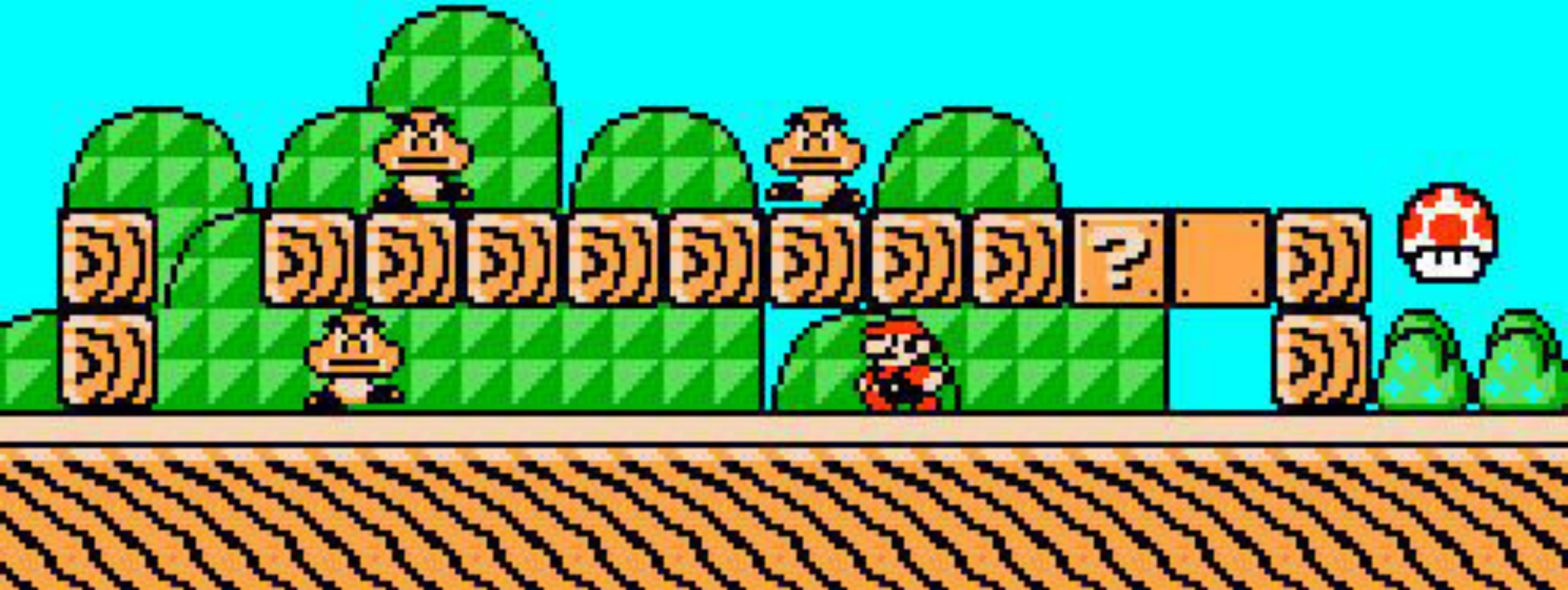
Build Step	Parameters Description		
Get latest dependencies	Maven Path to POM: pom.xml Goals: versions:use-latest-releases Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾
Update version	Maven Path to POM: pom.xml Goals: versions:set Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾
Build and deploy	Maven Path to POM: pom.xml Goals: clean deploy Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾


APP BACKEND
SUPER APP
DATA MODEL
APP BACKEND
SUPER APP
DATA MODEL
APP BACKEND
GUI
SUPER APP

Like it?

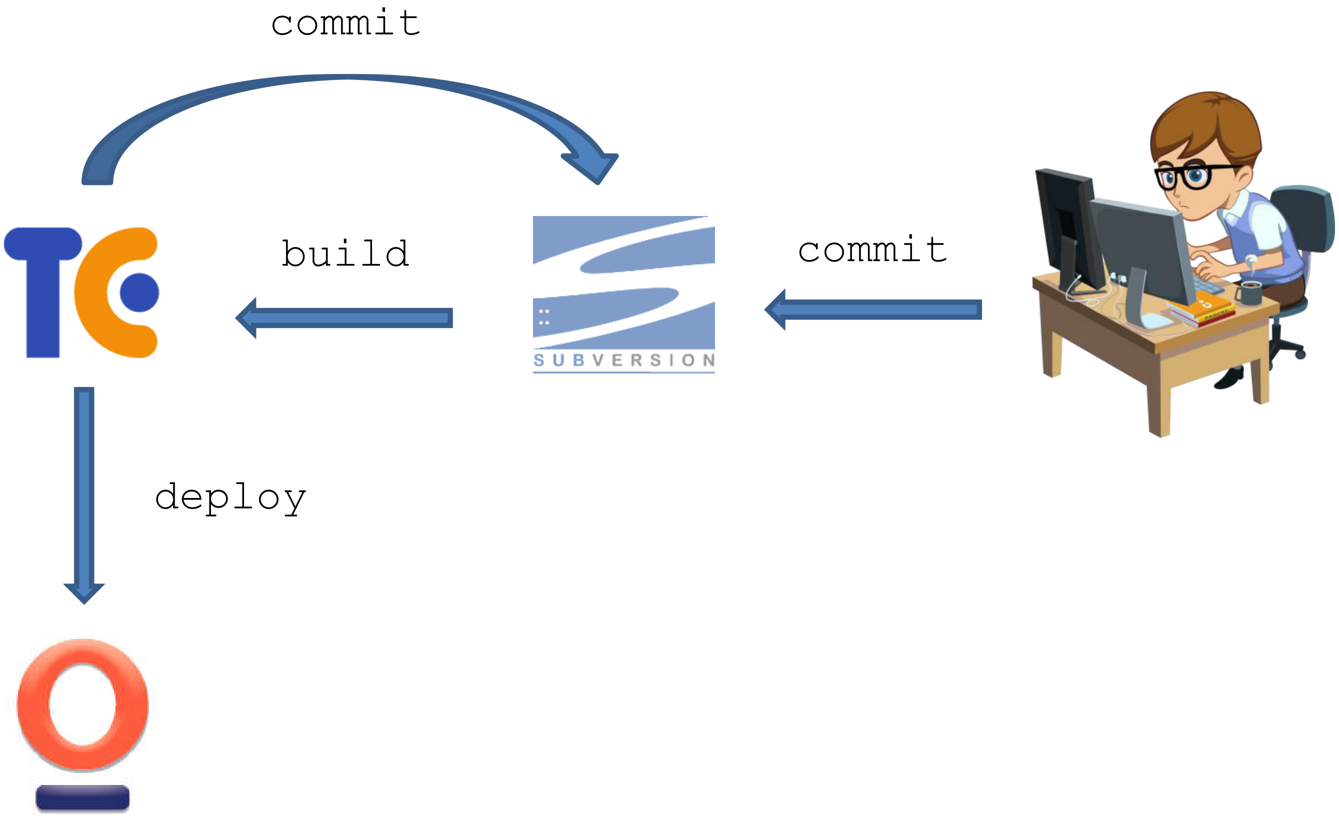
THUMBS&AMMO

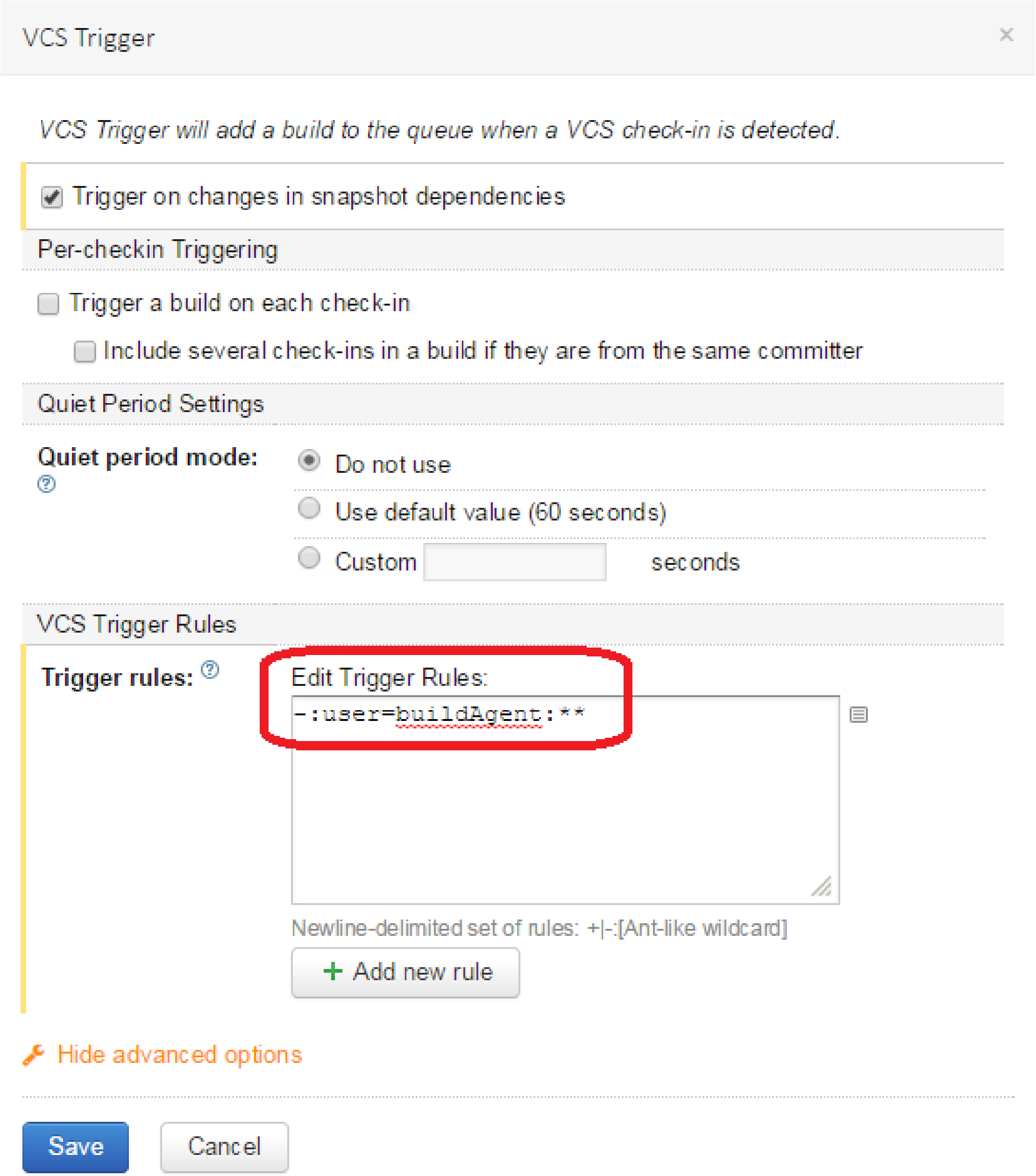
Somehow, I don't think you thought your cunning plan all the way through.



I Can Help ☺

Problem: Infinite Trigger





VCS Trigger ×

VCS Trigger will add a build to the queue when a VCS check-in is detected.

☑ Trigger on changes in snapshot dependencies

Per-checkin Triggering

□ Trigger a build on each check-in

□ Include several check-ins in a build if they are from the same committer

Quiet Period Settings

Quiet period mode:

● Do not use

○ Use default value (60 seconds)

○ Custom ______ seconds

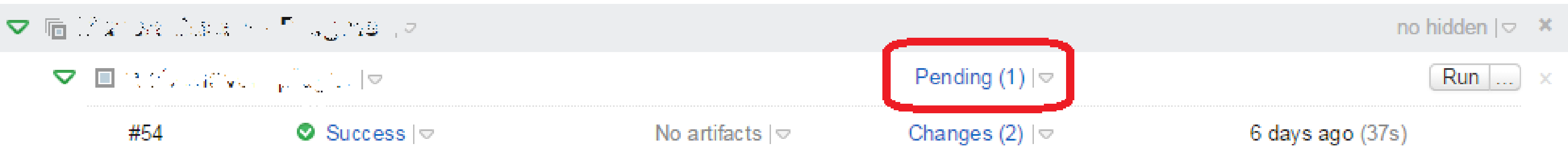
VCS Trigger Rules

Trigger rules:

Edit Trigger Rules:

```
-:user=buildAgent:**
```

Newline-delimited set of rules: +|-:[Ant-like wildcard]



[illegible]

no hidden

[illegible]

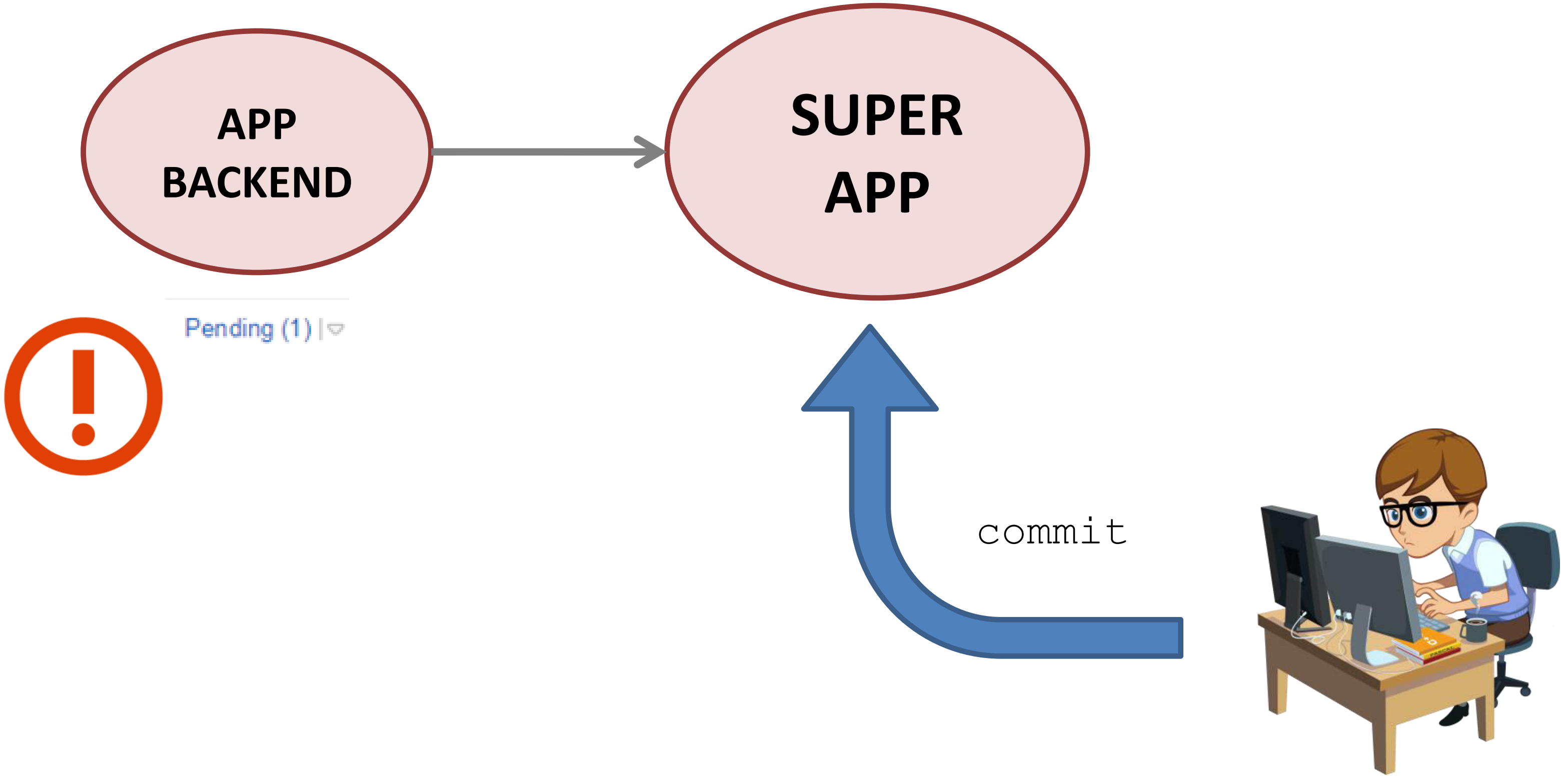
Pending (1)

Run

...

#54 Success No artifacts Changes (2) 6 days ago (37s)

Problem: Unnecessary rebuilds



APP BACKEND

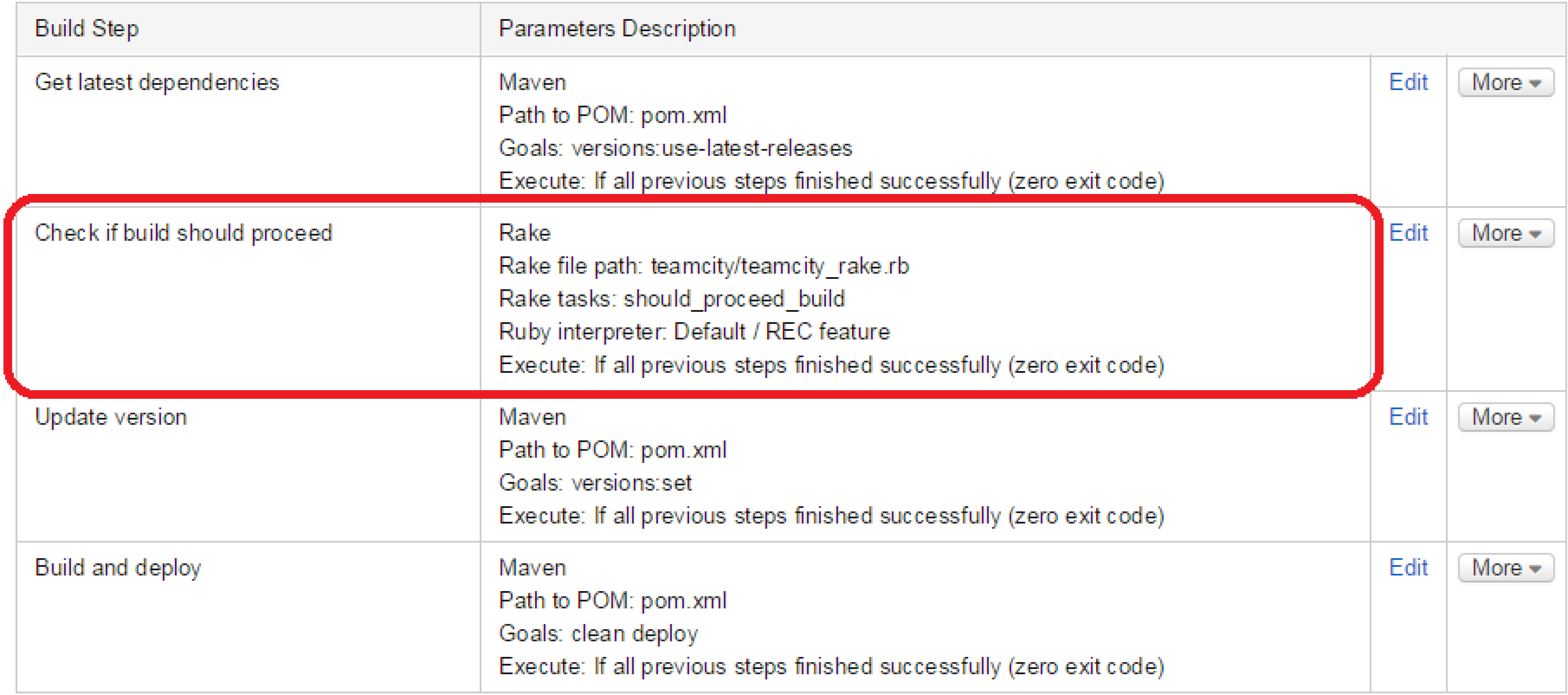
SUPER APP

Pending (1)

commit

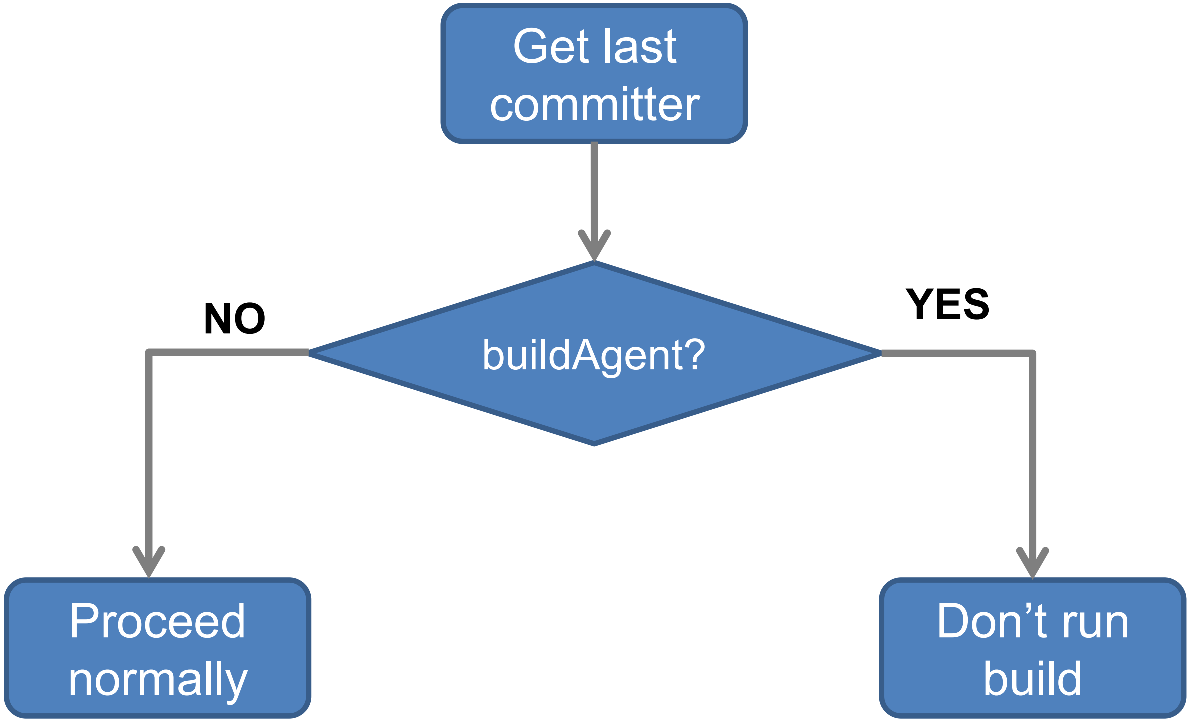
FAIL.

Problem: Unnecessary rebuilds



Build Step	Parameters Description		
Get latest dependencies	Maven Path to POM: pom.xml Goals: versions:use-latest-releases Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾
Check if build should proceed	Rake Rake file path: teamcity/teamcity_rake.rb Rake tasks: should_proceed_build Ruby interpreter: Default / REC feature Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾
Update version	Maven Path to POM: pom.xml Goals: versions:set Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾
Build and deploy	Maven Path to POM: pom.xml Goals: clean deploy Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾

Problem: Unnecessary rebuilds



`touch skip_build`

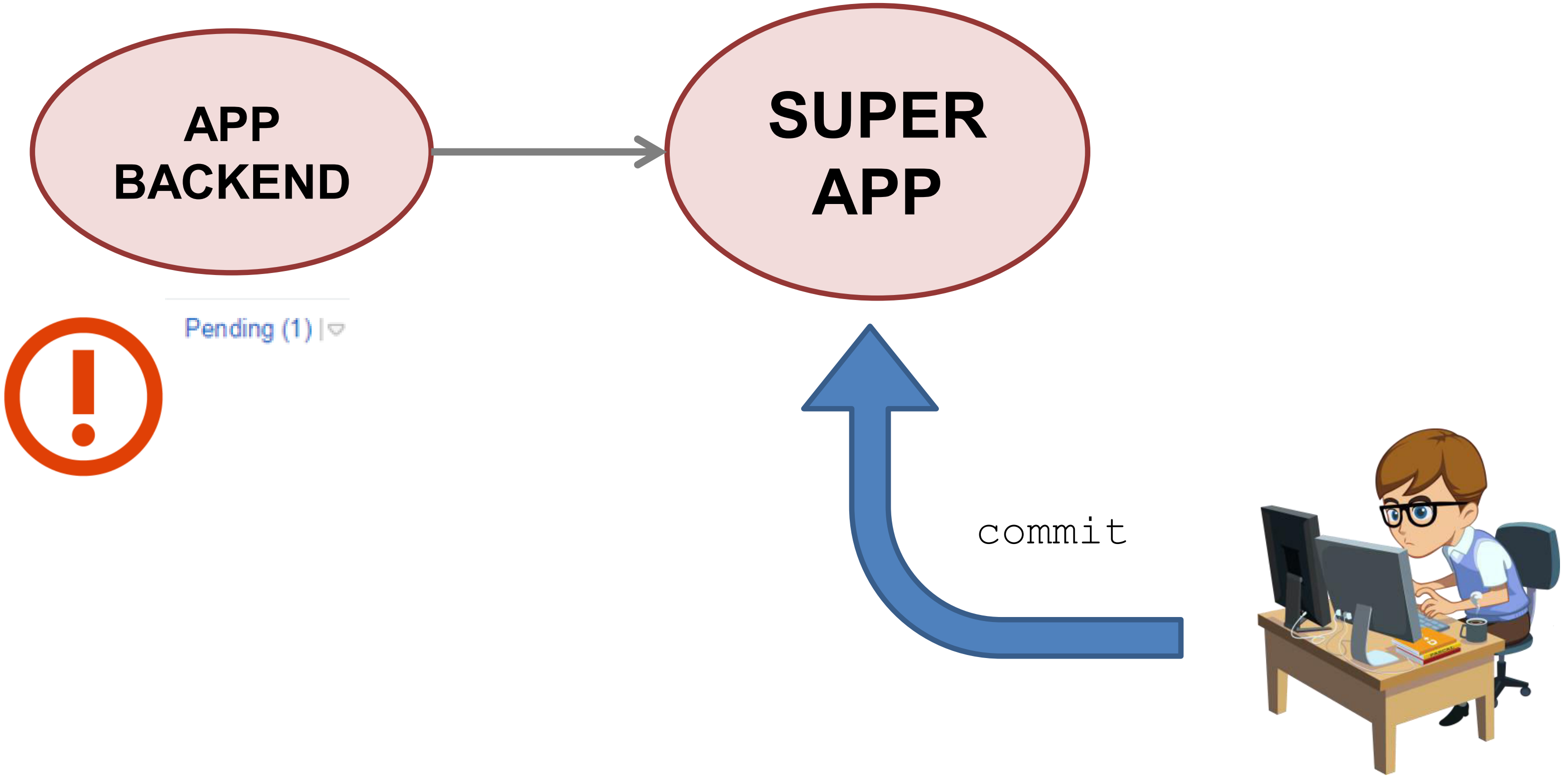
Problem: Unnecessary rebuilds

```
<profiles>
<!-- Plugins that need to be disabled when doing a no-run -->
<profile>
<id>do.nothing</id>
<activation>
    <file>
         <exists>skip_build</exists>
    </file>
</activation>

<build>
    <plugins>
        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-compiler-plugin</artifactId>
            <version>3.1</version>
            <configuration>
                <skipMain>true</skipMain>
                <skip>true</skip>
            </configuration>
        </plugin>
```

Problem: Unnecessary rebuilds

```
<plugin>
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-install-plugin</artifactId>
    <version>2.5.2</version>
    <configuration>
        <skip>true</skip>
    </configuration>
</plugin>

<plugin>
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-deploy-plugin</artifactId>
    <version>2.8.2</version>
    <configuration>
        <skip>true</skip>
    </configuration>
</plugin>
```

Problem: Unnecessary rebuilds



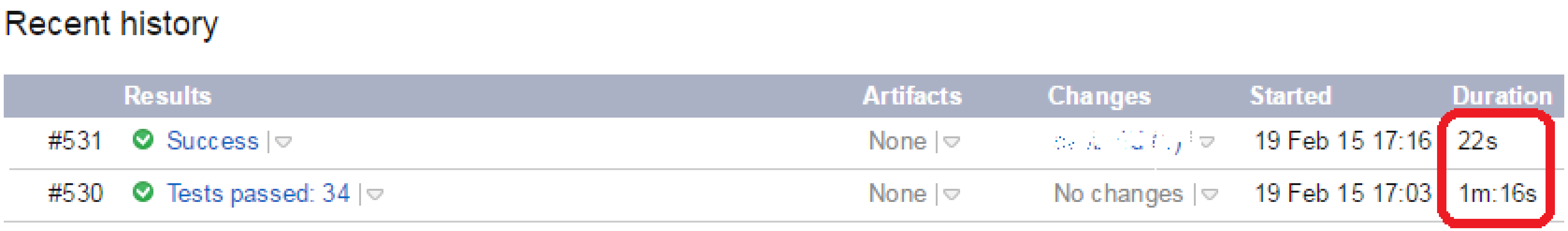
APP BACKEND

SUPER APP

Pending (1)

commit

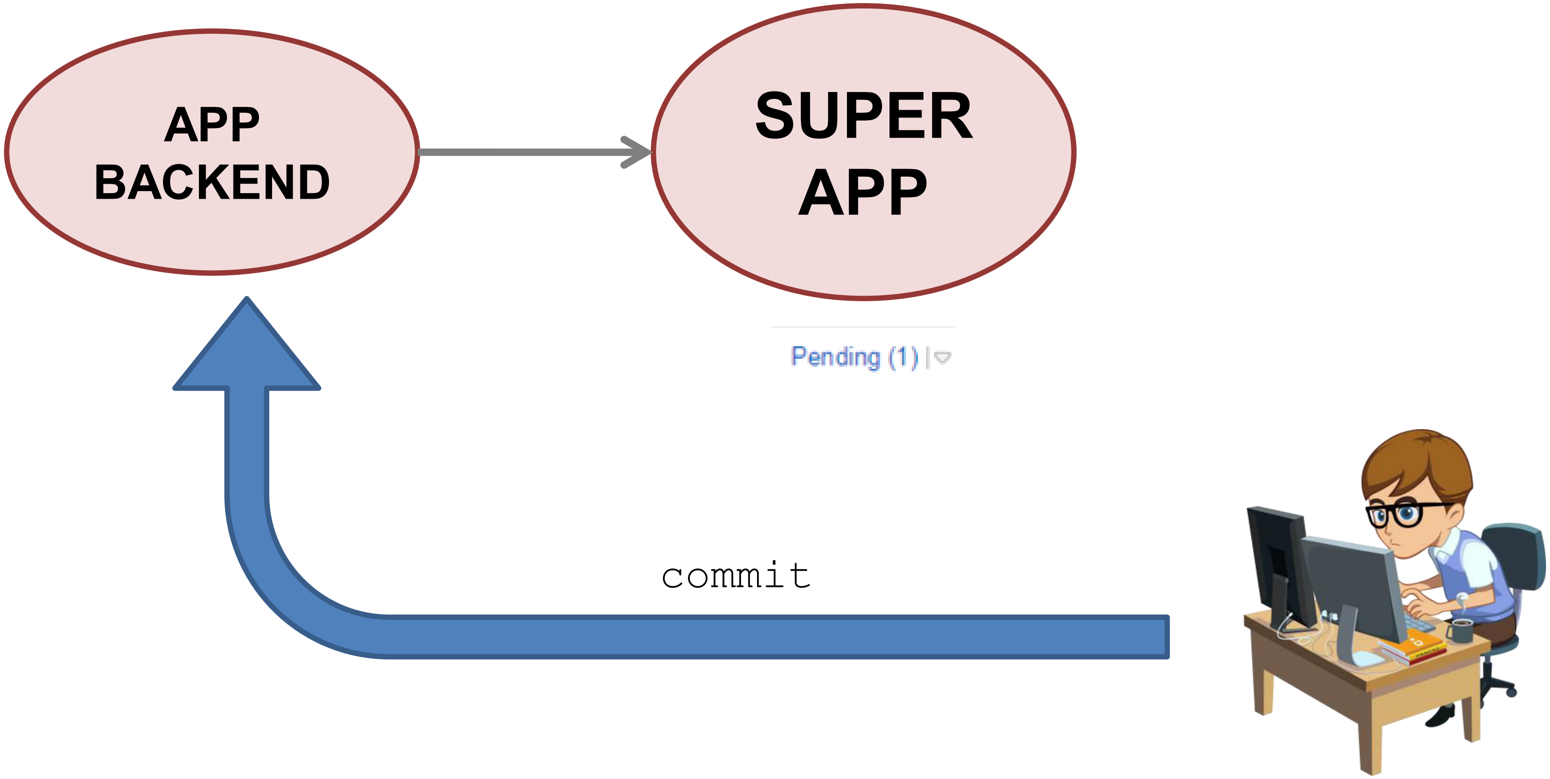
Problem: Unnecessary rebuilds



Recent history

	Results	Artifacts	Changes	Started	Duration
#531	✓ Success | ▽	None | ▽	[illegible] | ▽	19 Feb 15 17:16	22s
#530	✓ Tests passed: 34 | ▽	None | ▽	No changes | ▽	19 Feb 15 17:03	1m:16s

Problem: Necessary rebuilds



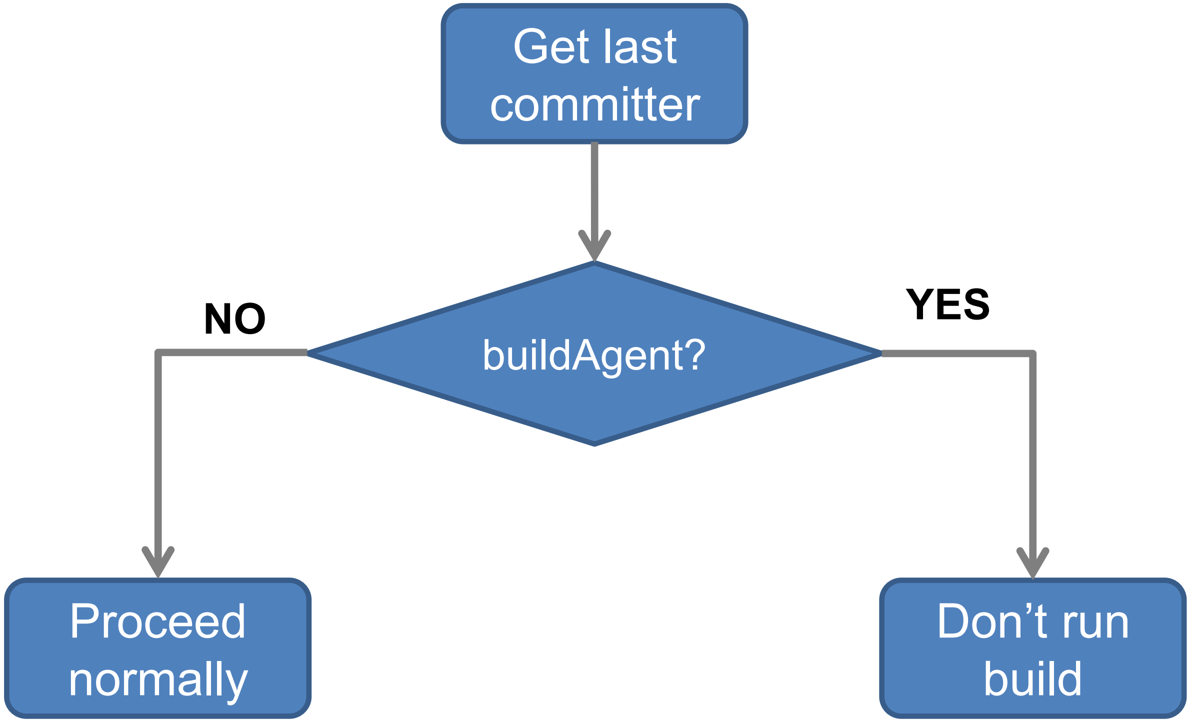
APP BACKEND

SUPER APP

Pending (1)

commit

Problem: Necessary rebuilds



Get last committer

buildAgent?

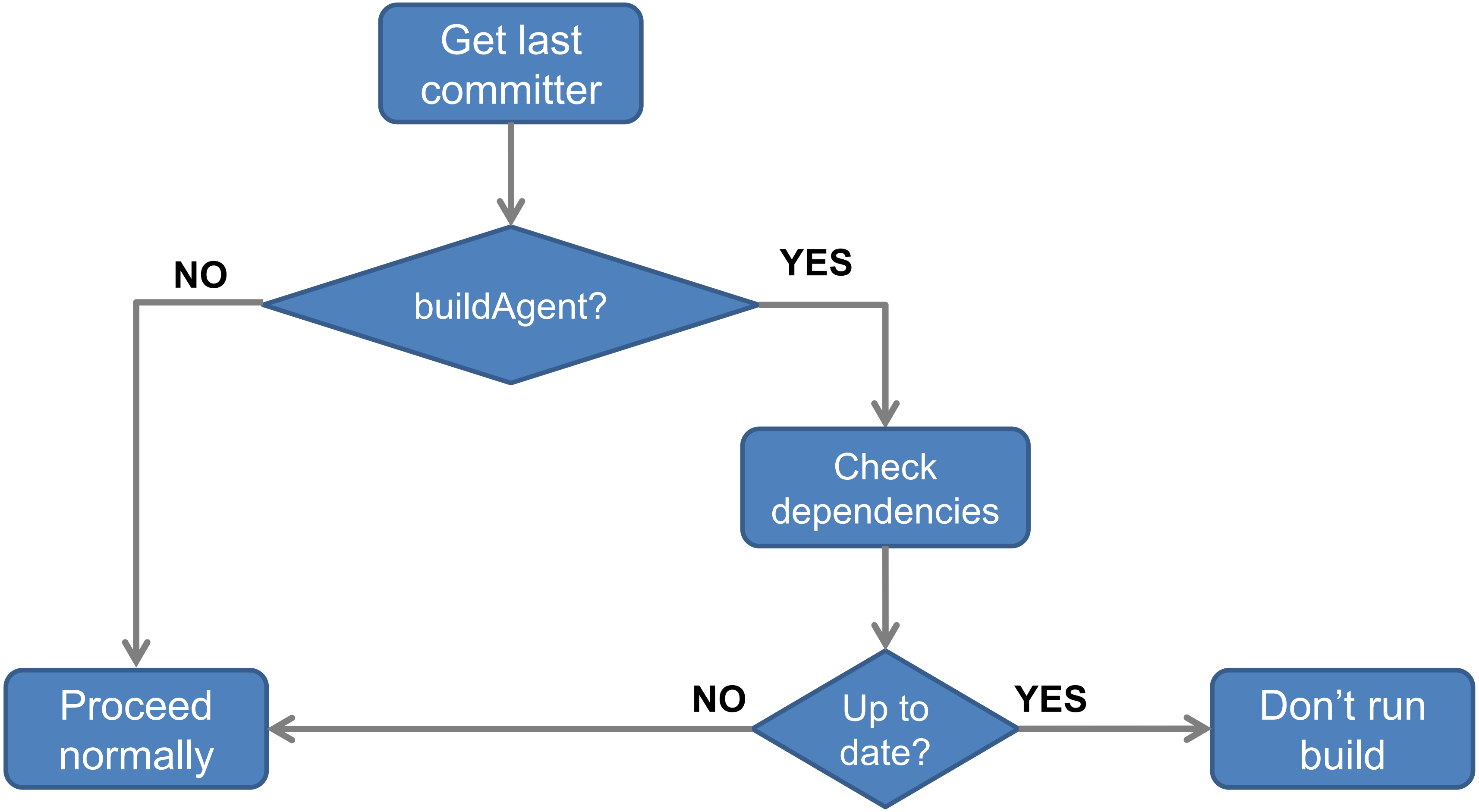
NO

YES

Proceed normally

Don’t run build

FAIL



Get last committer

buildAgent?

NO

YES

Check dependencies

Up to date?

NO

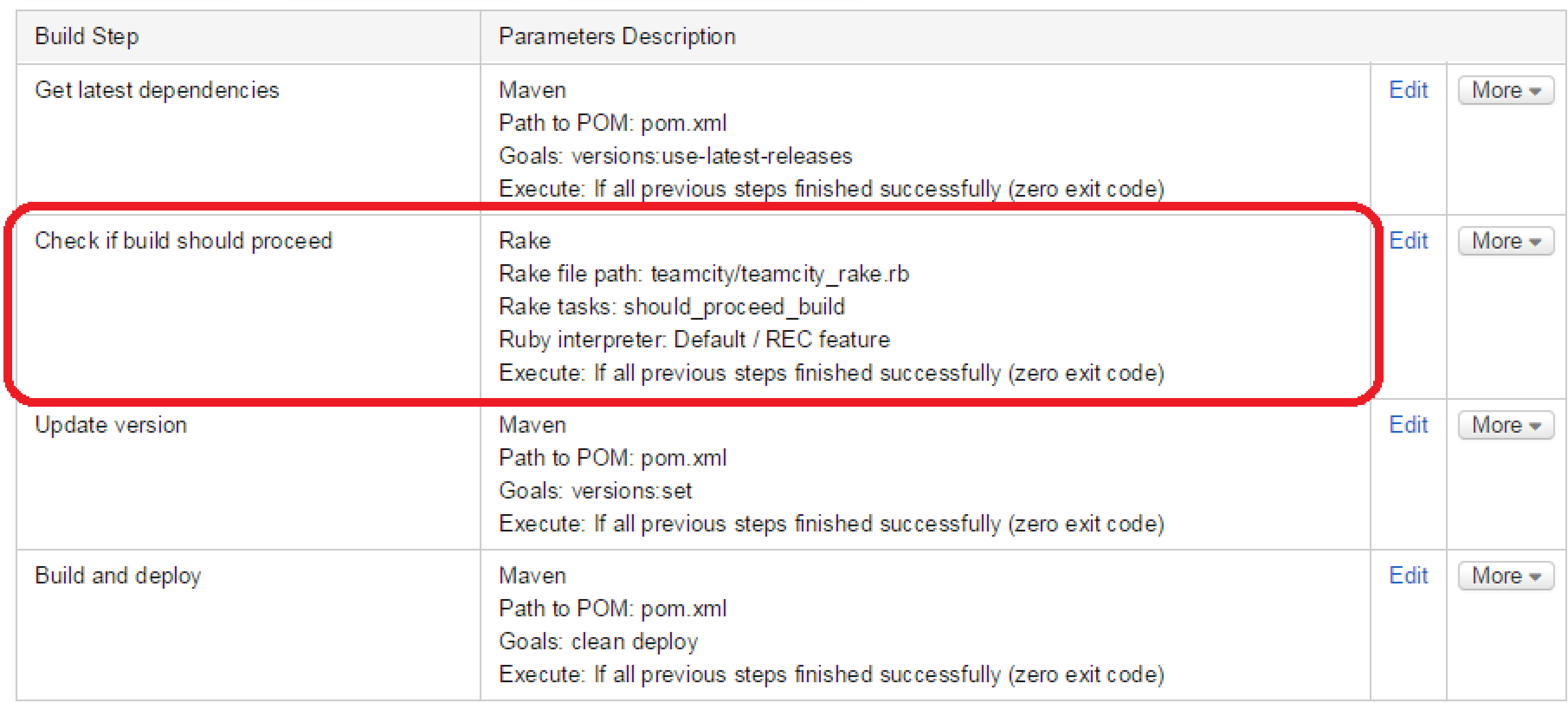
YES

Proceed normally

Don’t run build

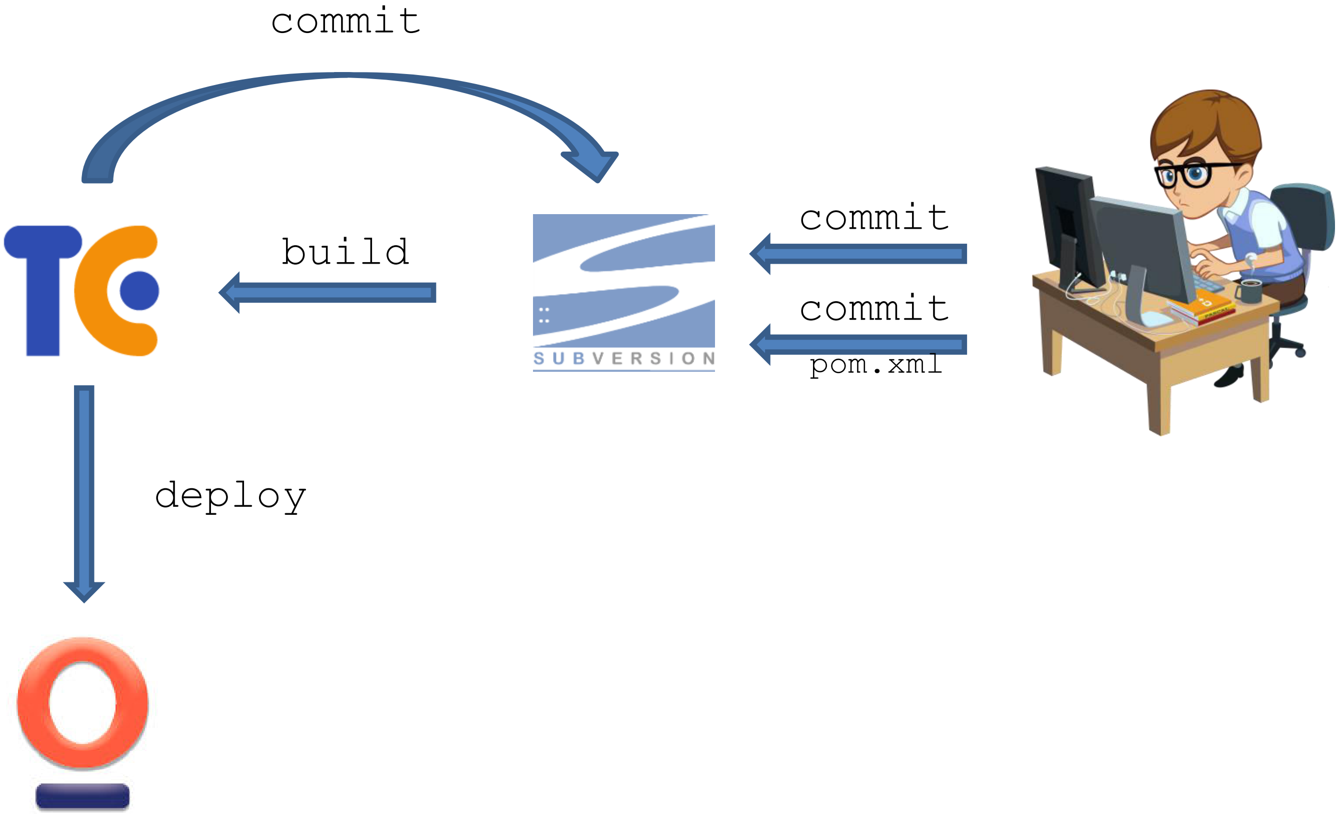
```
touch skip_build
```

Problem: Necessary rebuilds



Build Step	Parameters Description		
Get latest dependencies	Maven Path to POM: pom.xml Goals: versions:use-latest-releases Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾
Check if build should proceed	Rake Rake file path: teamcity/teamcity_rake.rb Rake tasks: should_proceed_build Ruby interpreter: Default / REC feature Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾
Update version	Maven Path to POM: pom.xml Goals: versions:set Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾
Build and deploy	Maven Path to POM: pom.xml Goals: clean deploy Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾

Problem: Doomed Build



commit

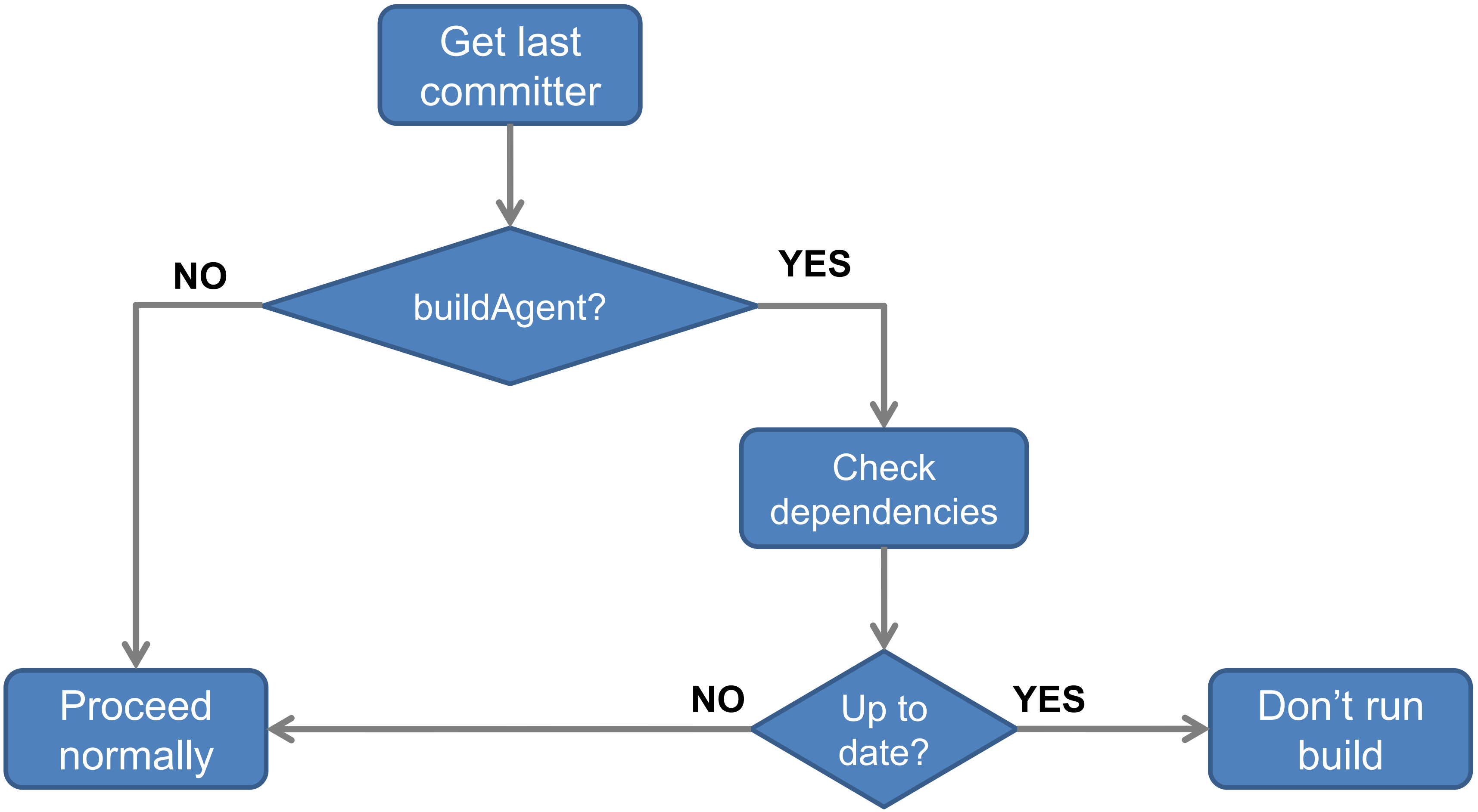
build

commit

commit

pom.xml

deploy



Get last committer

buildAgent?

NO

YES

Check dependencies

Up to date?

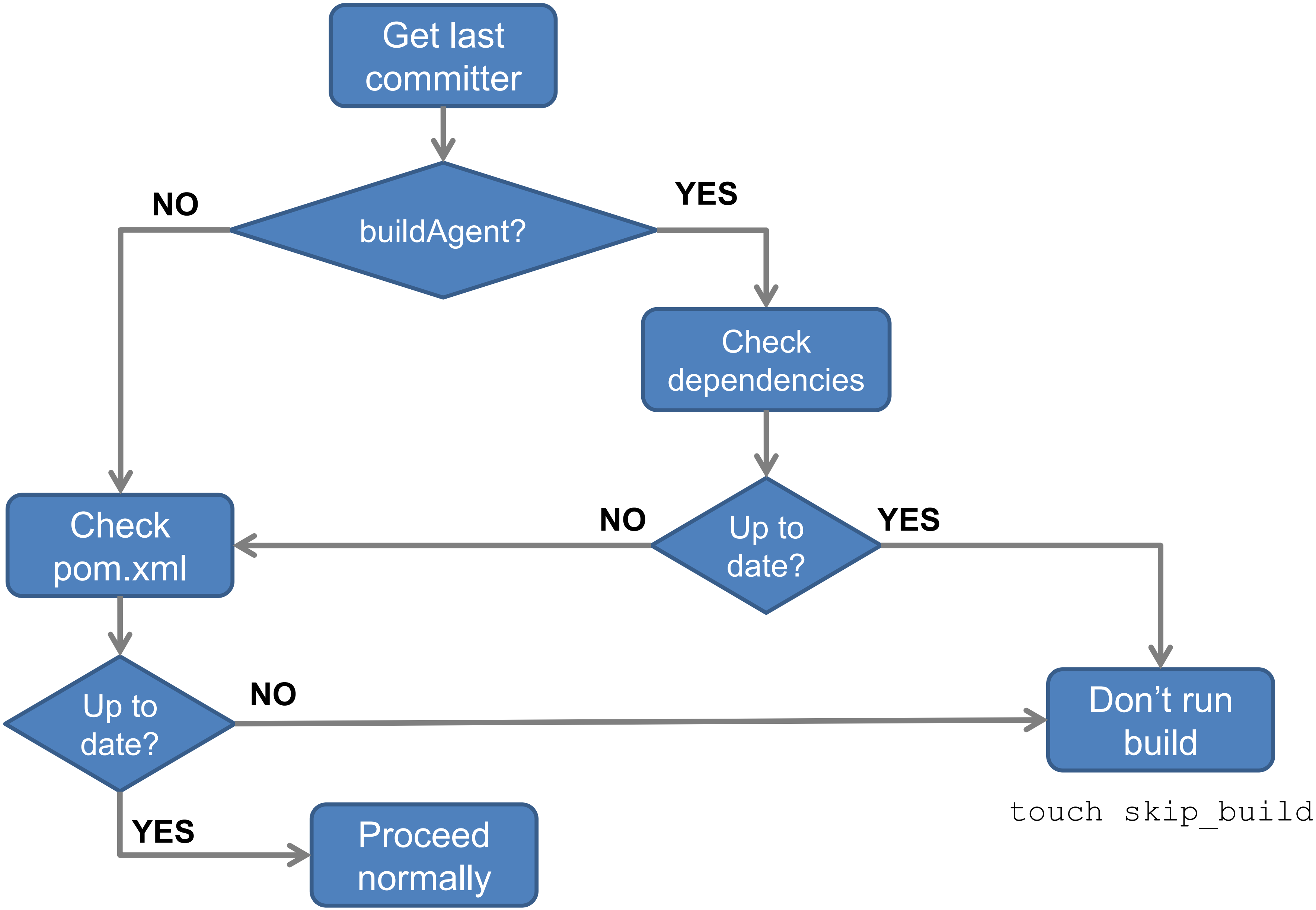
NO

YES

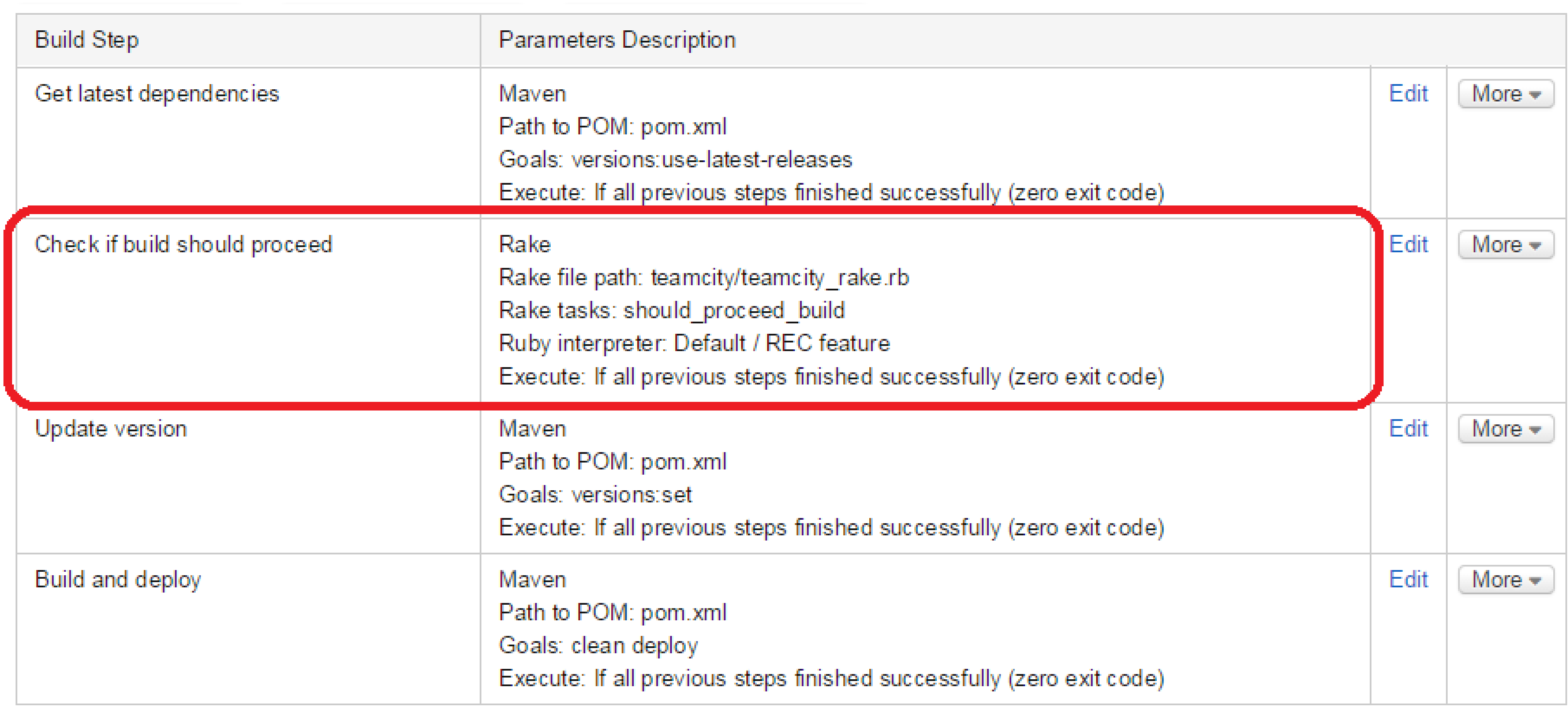
Proceed normally

Don’t run build

```
touch skip_build
```


Get last committer
buildAgent?
NO
YES
Check dependencies
Up to date?
NO
YES
Check pom.xml
Up to date?
NO
YES
Don't run build
Proceed normally
touch skip_build

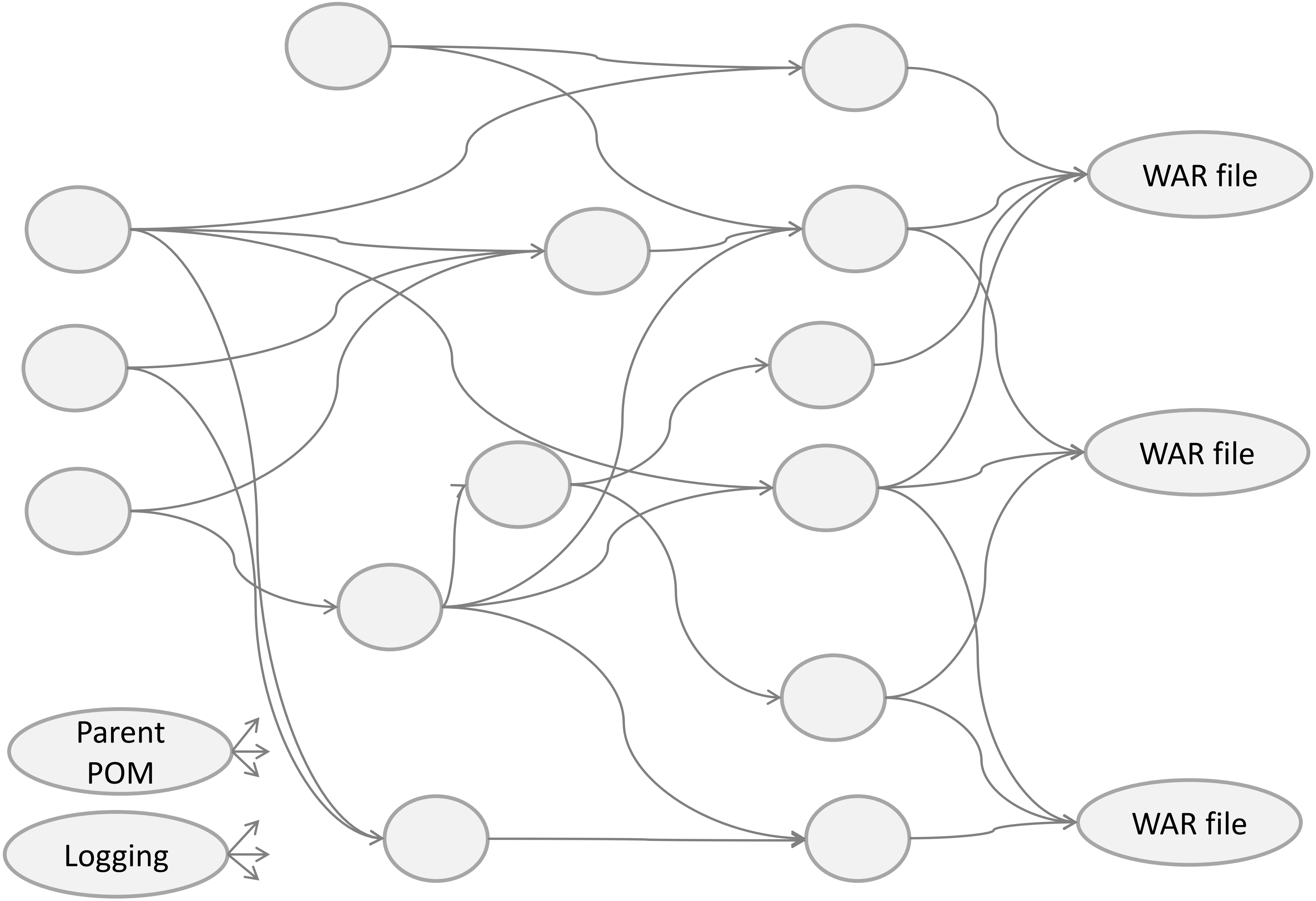
Problem: Doomed Build

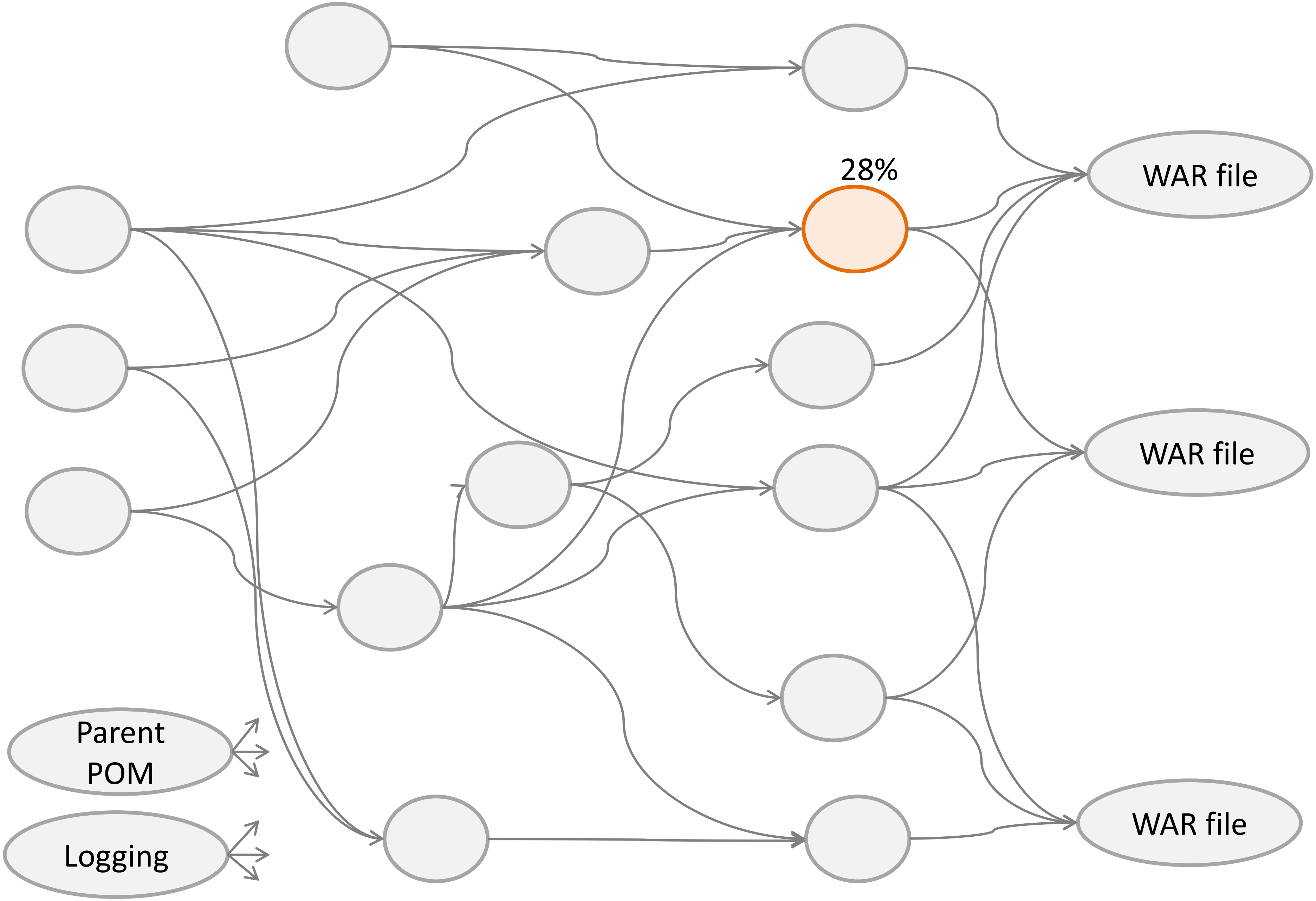


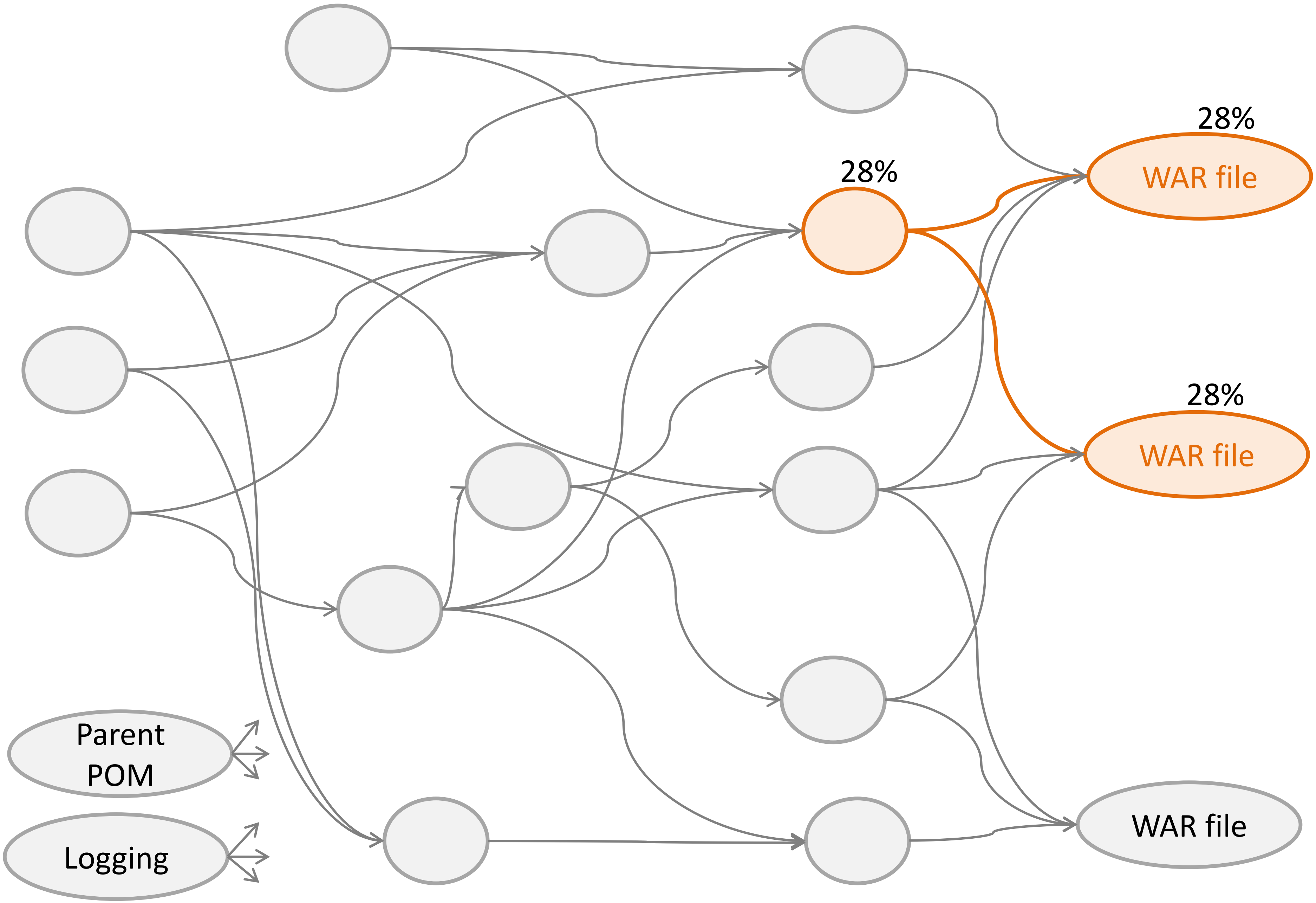
Build Step	Parameters Description		
Get latest dependencies	Maven Path to POM: pom.xml Goals: versions:use-latest-releases Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾
Check if build should proceed	Rake Rake file path: teamcity/teamcity_rake.rb Rake tasks: should_proceed_build Ruby interpreter: Default / REC feature Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾
Update version	Maven Path to POM: pom.xml Goals: versions:set Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾
Build and deploy	Maven Path to POM: pom.xml Goals: clean deploy Execute: If all previous steps finished successfully (zero exit code)	Edit	More ▾

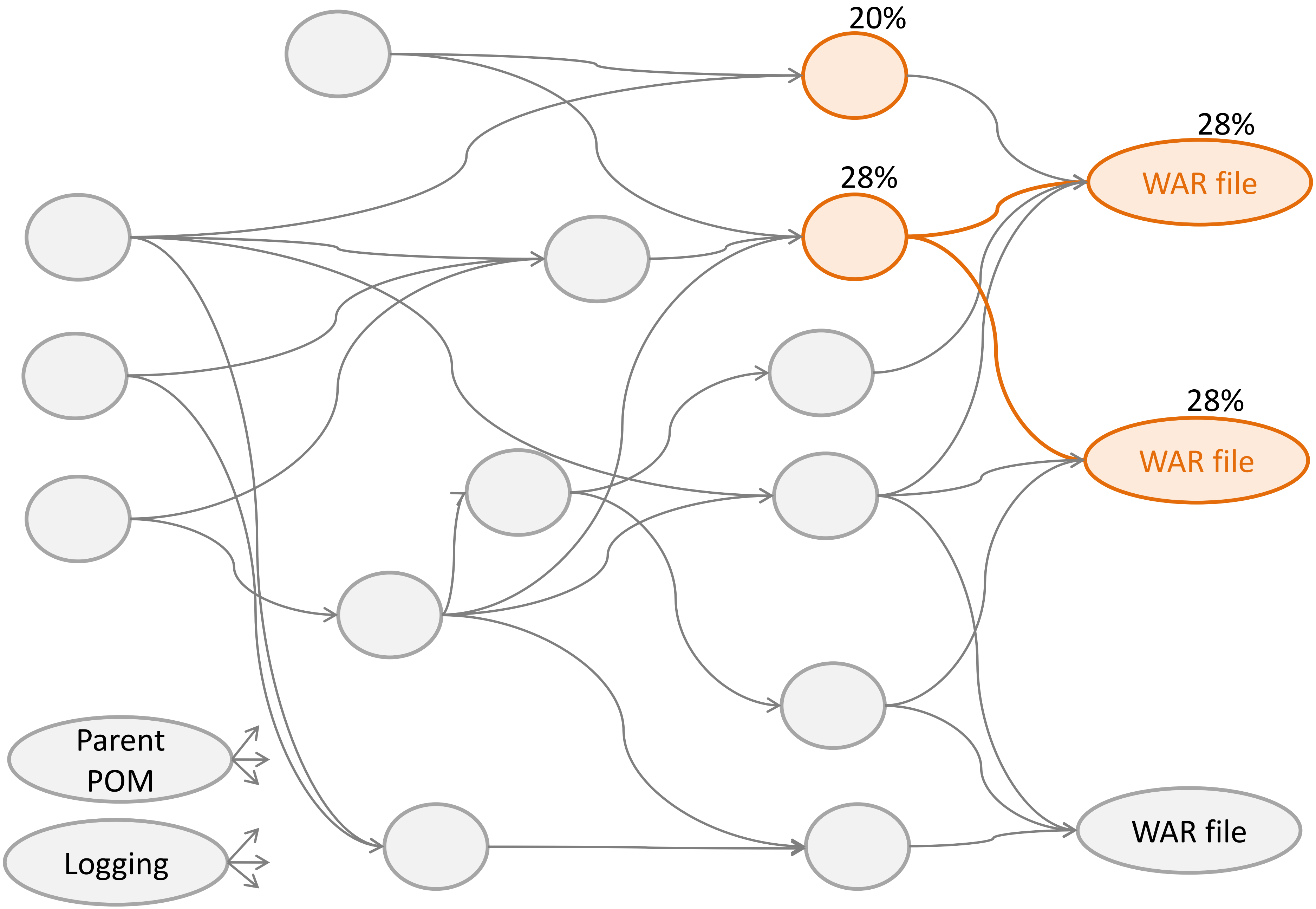
BAM!
SUCCESS

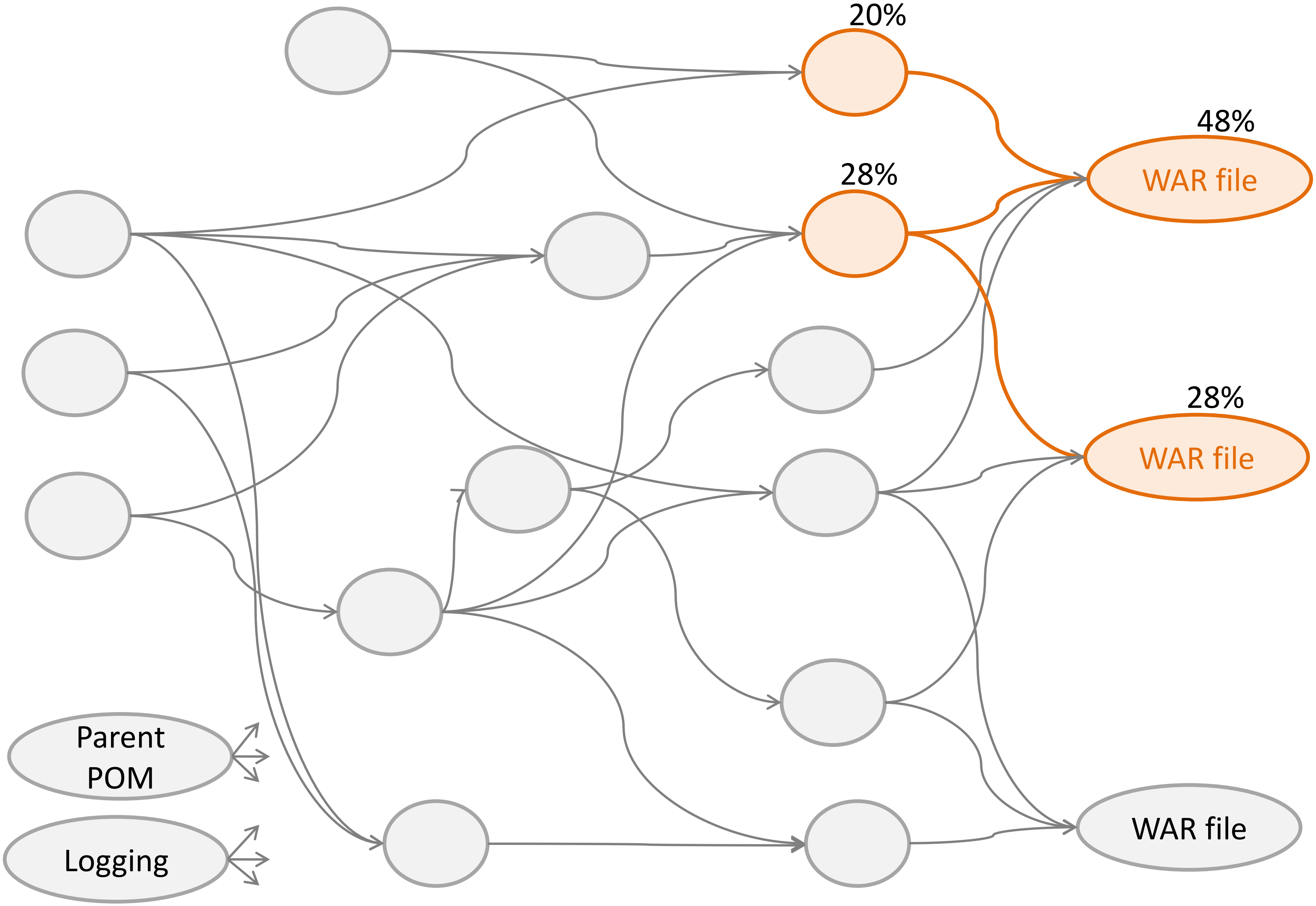
A real case scenario

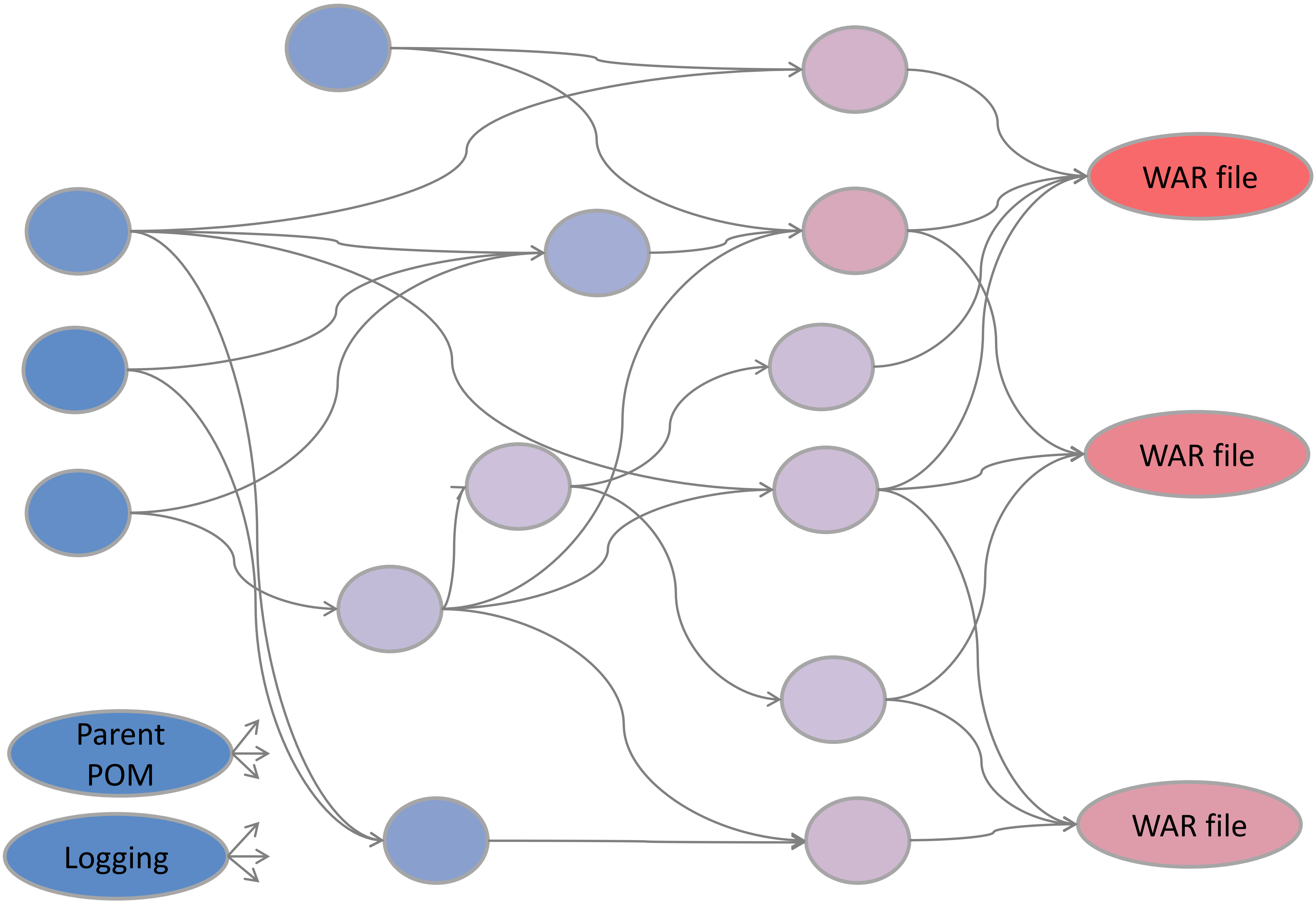

WAR file
WAR file
WAR file
Parent POM
Logging

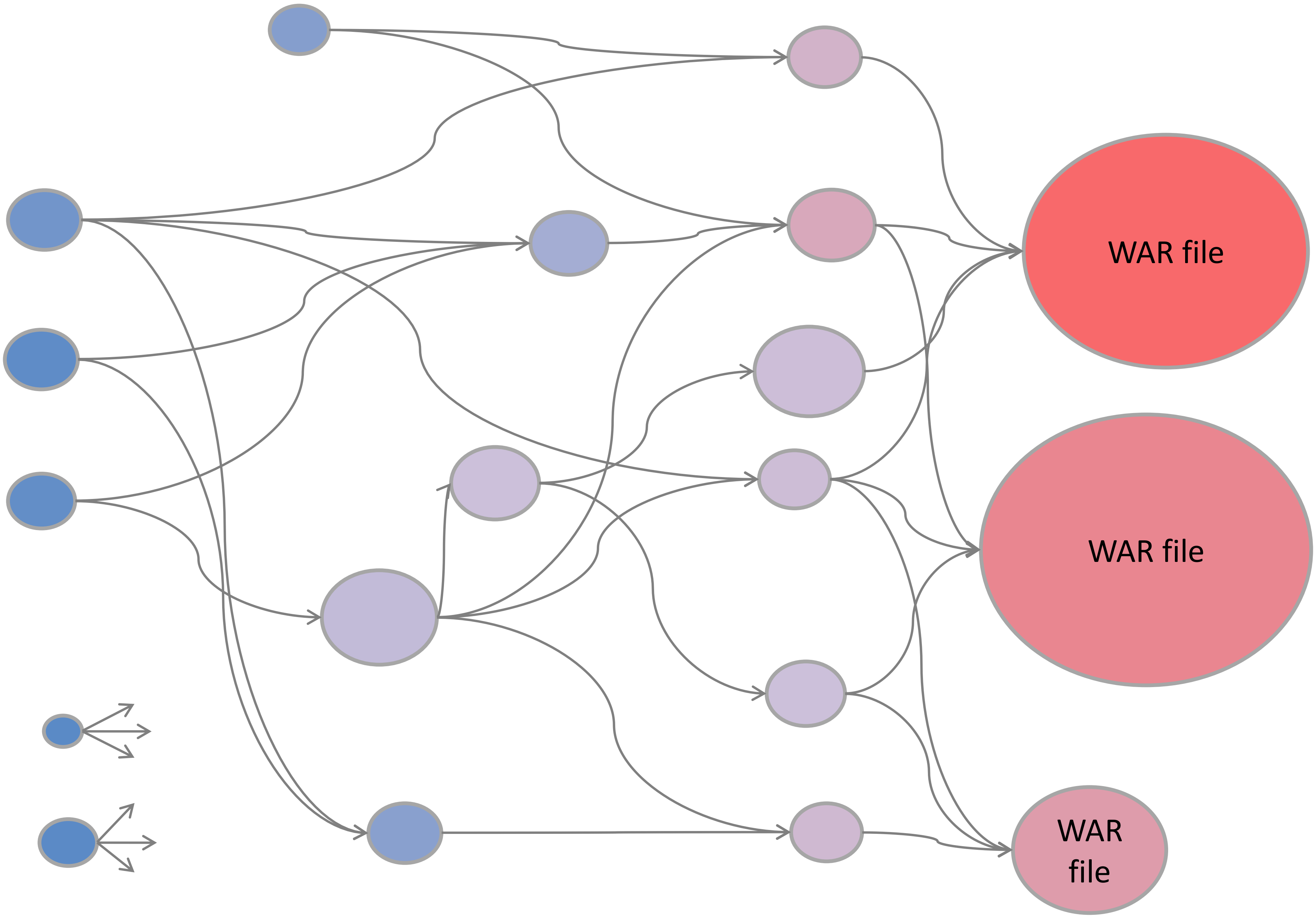

28%
WAR file
WAR file
WAR file
Parent POM
Logging


28%
28%
WAR file
28%
WAR file
Parent POM
Logging
WAR file


20%
28%
28%
WAR file
28%
WAR file
Parent POM
Logging
WAR file


20%
48%
WAR file
28%
28%
WAR file
Parent POM
Logging
WAR file

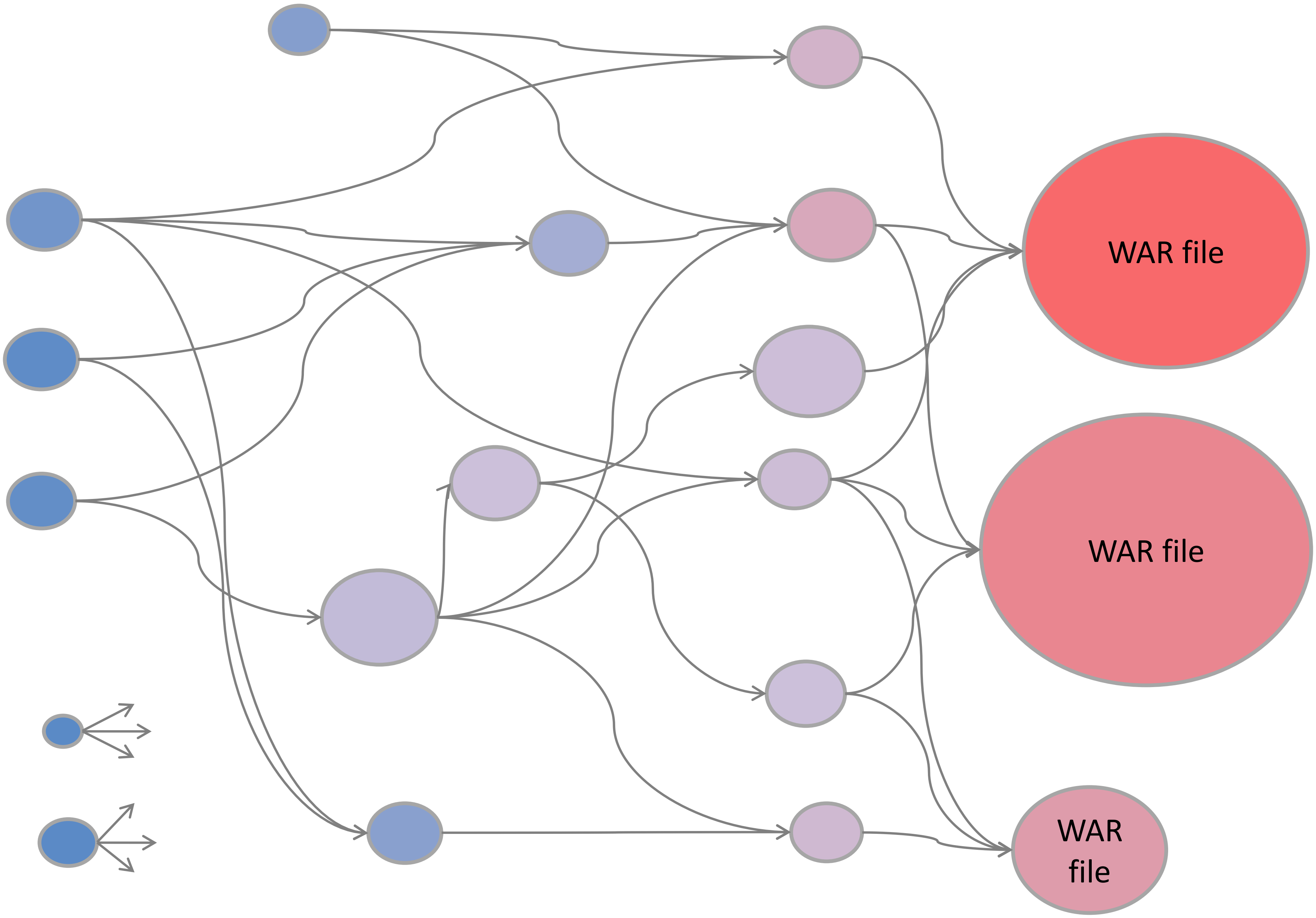

WAR file
WAR file
WAR file
Parent POM
Logging

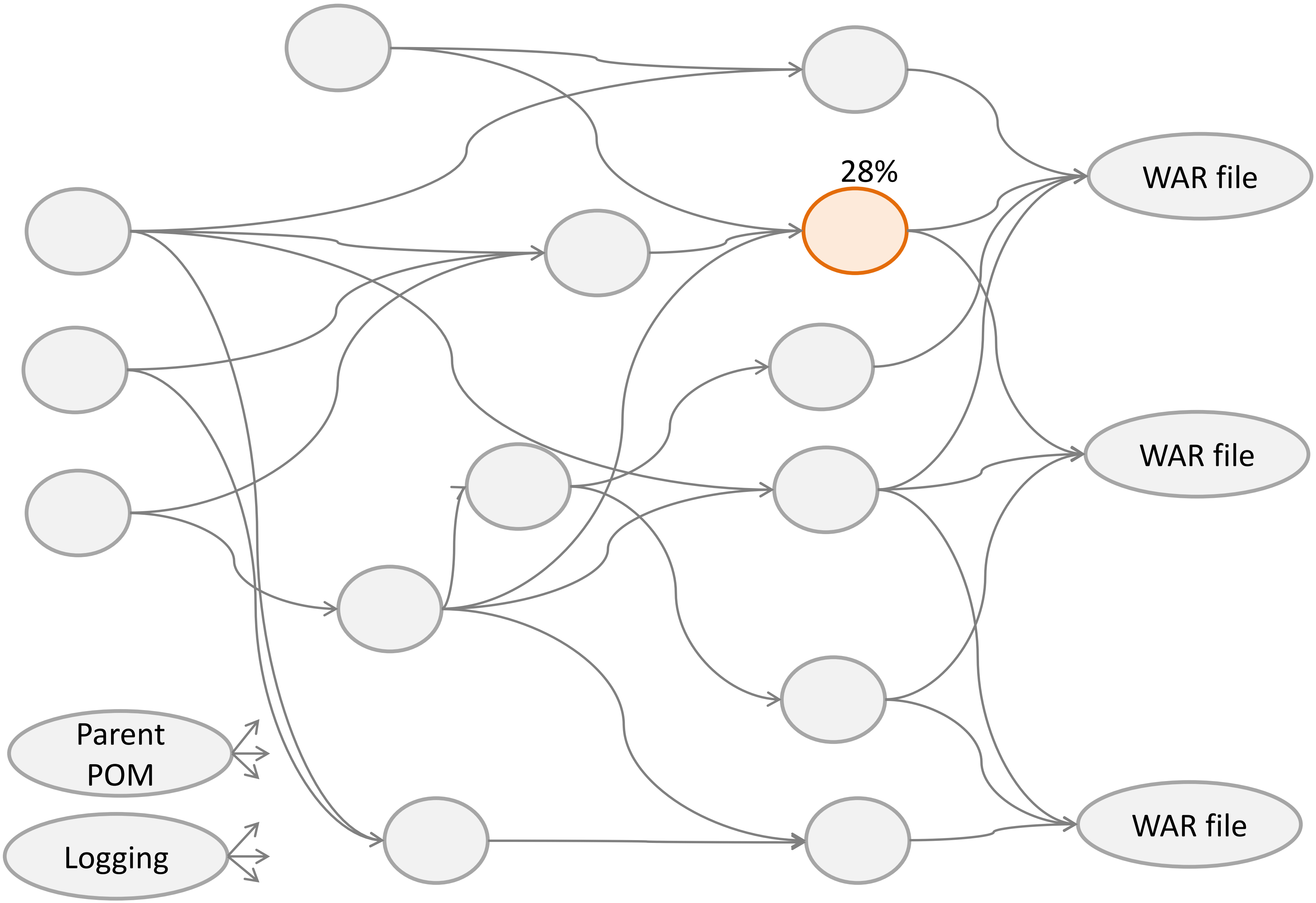

WAR file
WAR file
WAR file

Useful metrics

- Build Time (BT): time an individual build takes to run

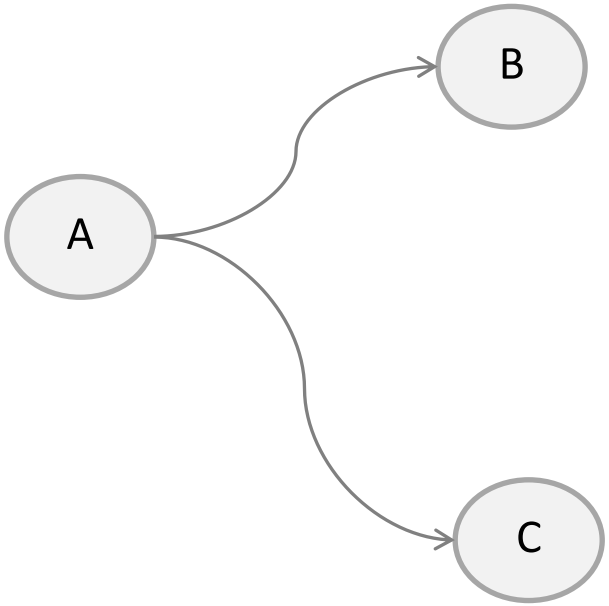
- Change Rate (CR): percentage of commits upon an individual build with respect to the whole system


WAR file
WAR file
WAR file


28%
WAR file
WAR file
WAR file
Parent POM
Logging

Useful metrics

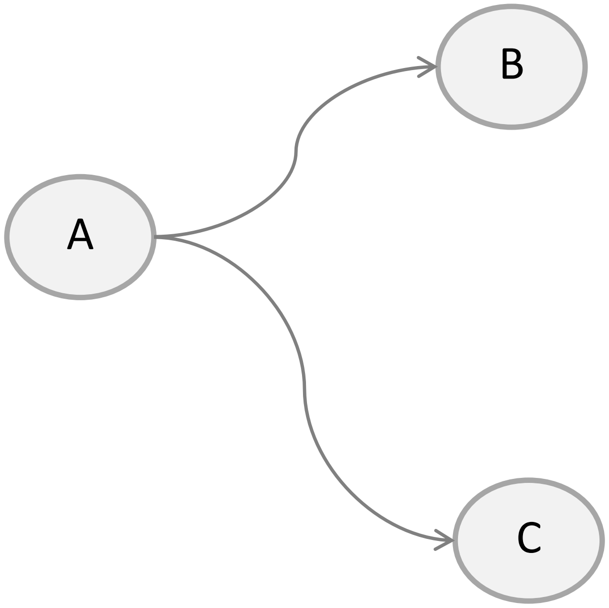
- Impact Time (IT): total time to run a build and all the builds that will be triggered as a result



Useful metrics

Serial execution →

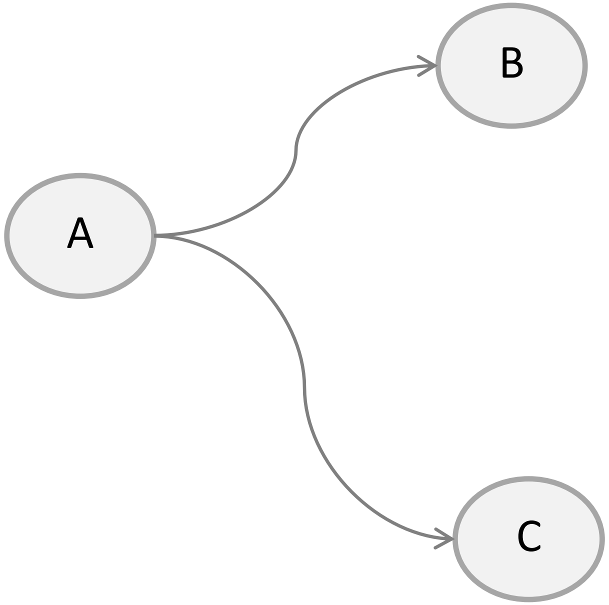
IT(A) = BT(A) + IT(B) + IT(C)

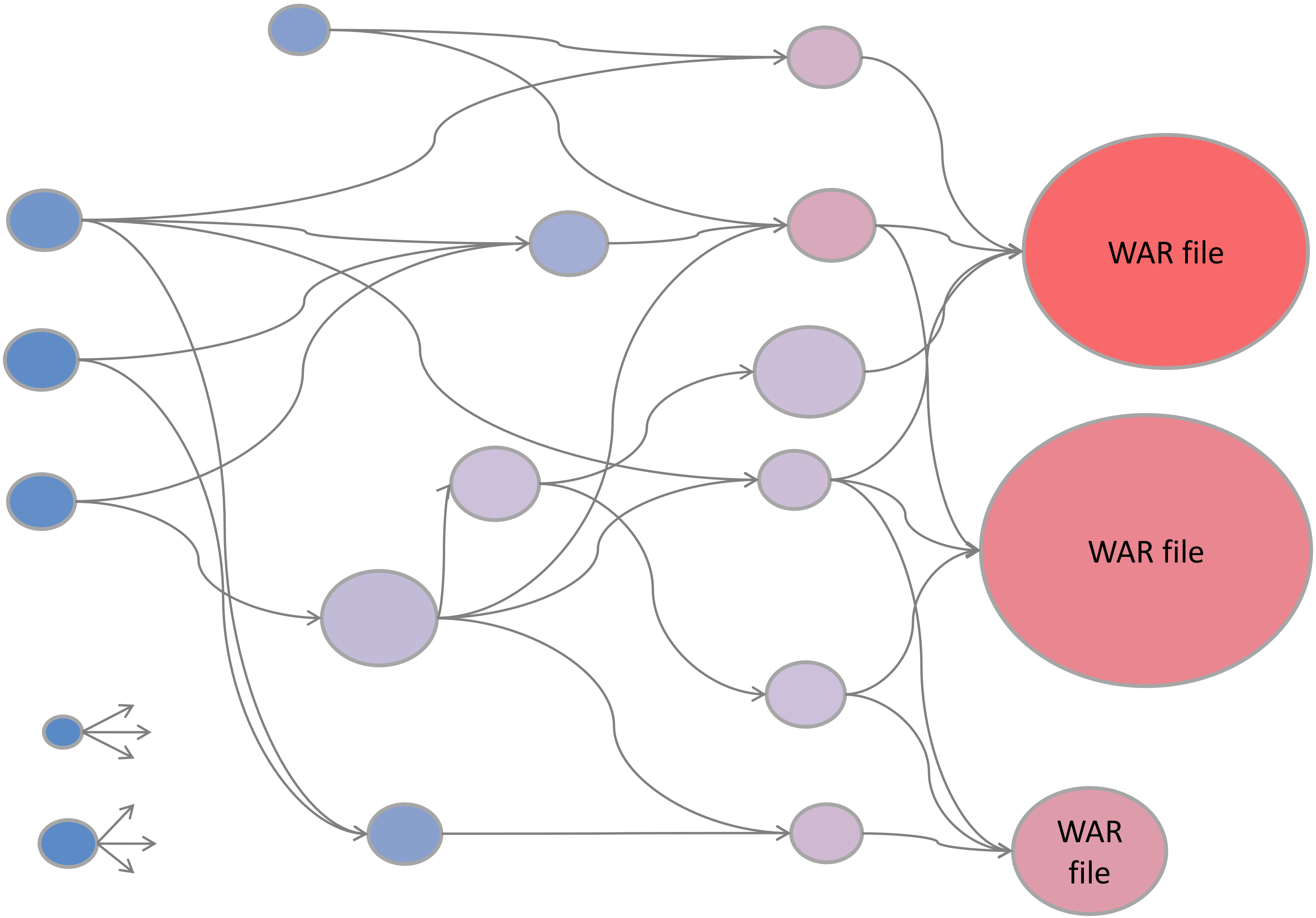


Useful metrics

Parallel execution →

IT(A) = BT(A) + max(IT(B), IT(C))




WAR file
WAR file
WAR file

Useful metrics

Weighted Impact Time (WIT): impact time of a build weighted according to its change rage

WIT(A) = IT(A) * CR(A)

Useful metrics

Average Impact Time (AIT): total time needed, on average, to execute all necessary builds after any given commit anywhere in the system

AIT = WIT(A) + WIT(B) + ... + WIT(Z)

Sample Thresholds

Sample Thresholds

Average Impact Time

Average Impact Time is what indicates how well you have scaled your system

Sample Thresholds

Maximum Impact Time

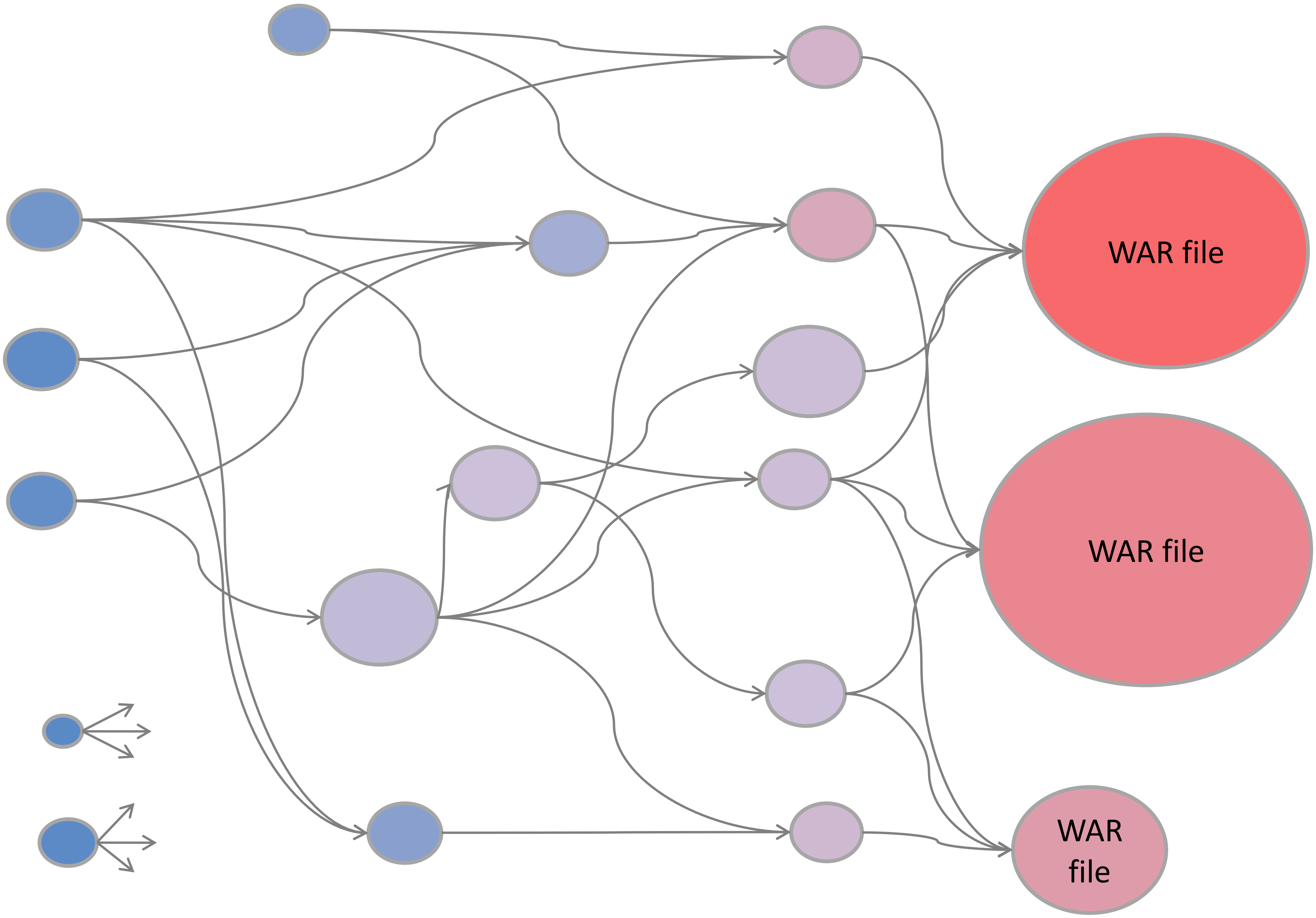
In a worst-case scenario, a build won’t take longer than this.

Sample Thresholds

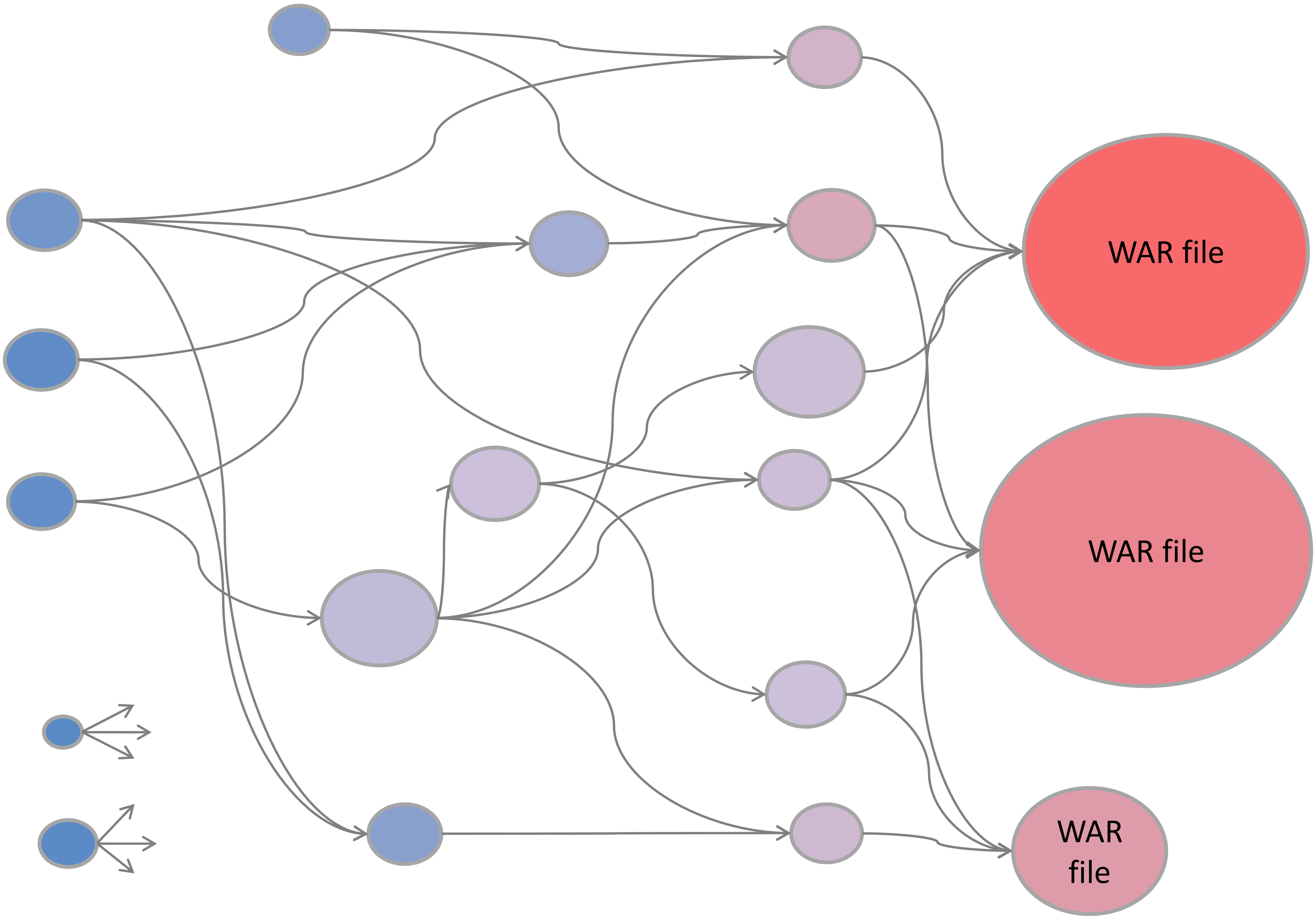
Maximum Impact Time for Critical Components

The same, but only for your most sensitive modules (log-in, payment gateway, etc.)

Beware of dependencies!


WAR file
WAR file
WAR file

https://commons.wikimedia.org/wiki/File:Trainera_Transbordador_Vizcaya.jpg


WAR file
WAR file
WAR
file

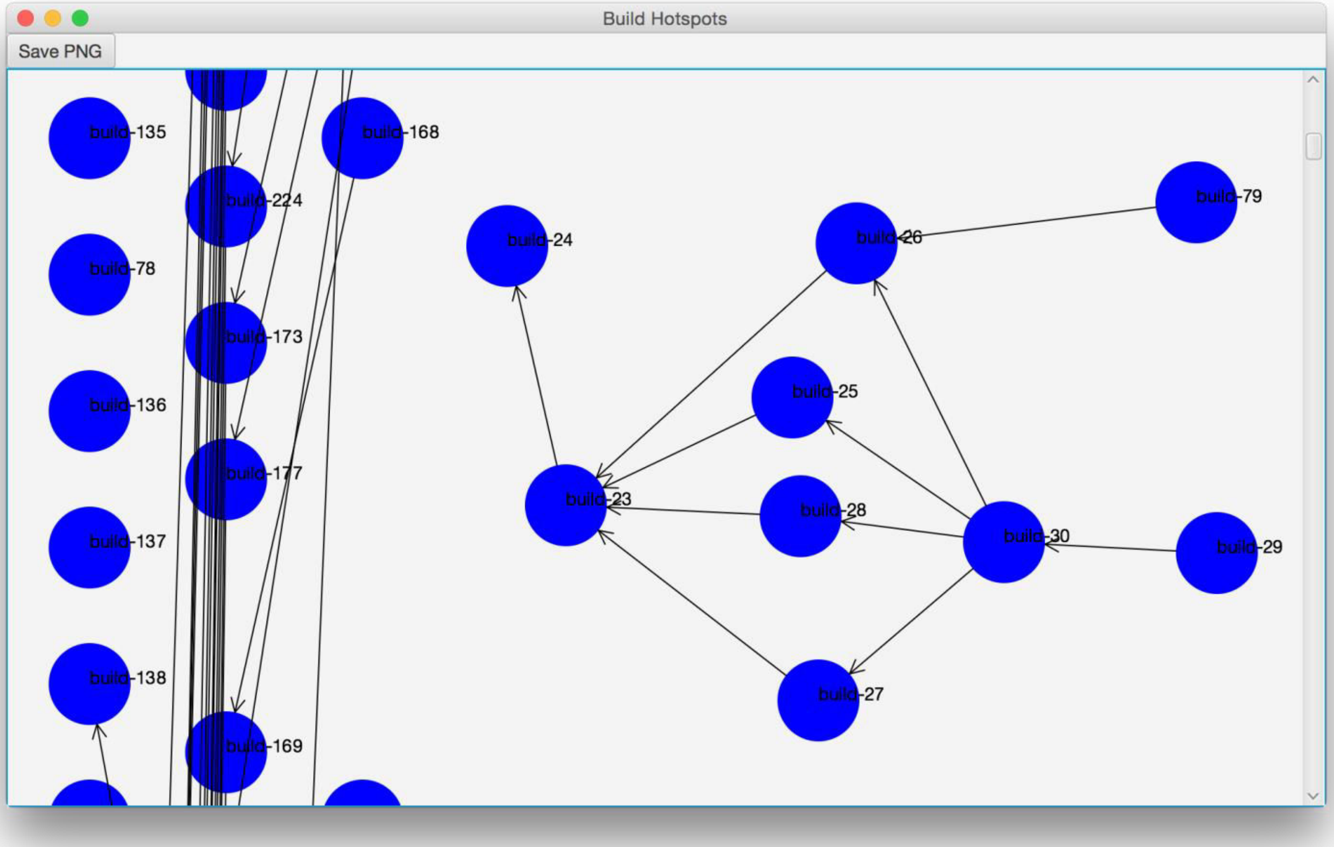
Manual processing takes time…

Automating Build Analysis

- Most CI systems provide an API
- Calculations aren’t complex
- Multiple graphical tools available

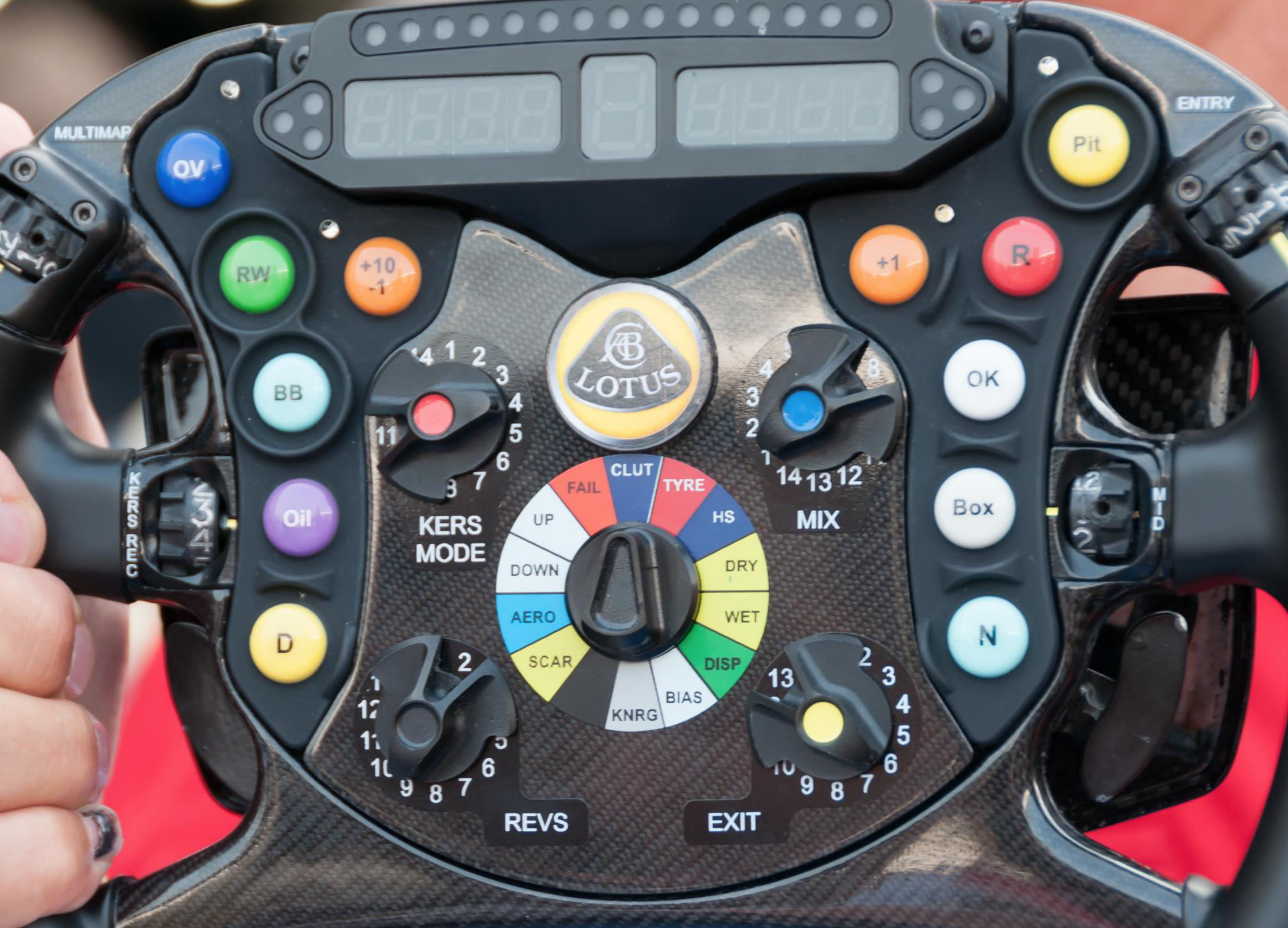
Build Hotspots

github.com/quiram/build-hotspots

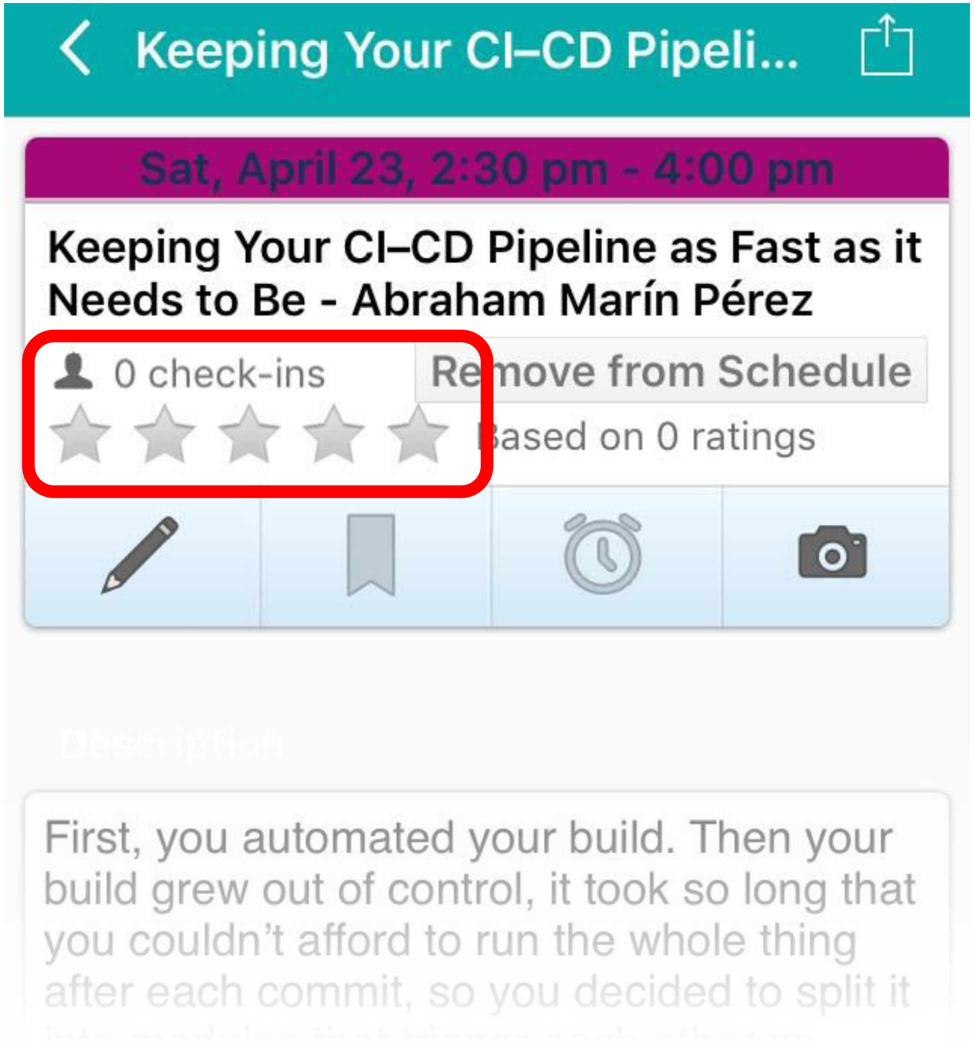


Automating Build Analysis

- Add size
- Add support for other CI systems
- Update data automatically (for build displays)
- Anything else you may find useful!



https://commons.wikimedia.org/wiki/File:2012_Italian_GP_-_Lotus_wheel.jpg



Sat, April 23, 2:30 pm - 4:00 pm

Keeping Your CI–CD Pipeline as Fast as it Needs to Be - Abraham Marín Pérez

First, you automated your build. Then your build grew out of control, it took so long that you couldn't afford to run the whole thing after each commit, so you decided to split it

Thank You!

fromfragiletoagile.com
@AbrahamMarin
#ACCUConf