Continuous integration

Slides from Tim Storer

Java Build tools

- Ant and Ivy
- Maven
- **Gradle (Lab)**

Assignment Project Exam Help

https://powcoder.com

Add WeChat powcoder

Continuous Integration

Assignment Project Exam Help

https://powcoder.com

Add WeChat powcoder

- Integration Hell

- Change Management

- Monitoring and Maintaining Software Quality

- Continuous Deployment

- Continuous Integration Environments

Integration Hell

Assignment Project Exam Help

https://powcoder.com

Add WeChat powcoder
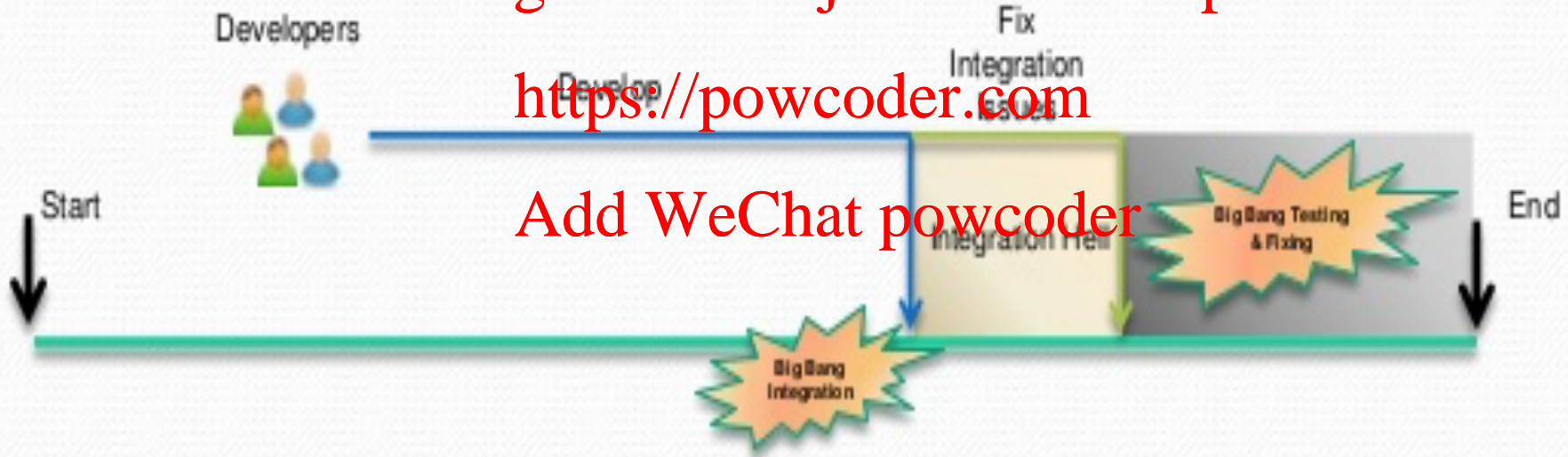
Practices in CI [Fowler, 2006]

Change management:

- Maintain a single source repository
- Everyone commits to the mainline every day
- Every commit should build the mainline [the trunk] on an integration machine
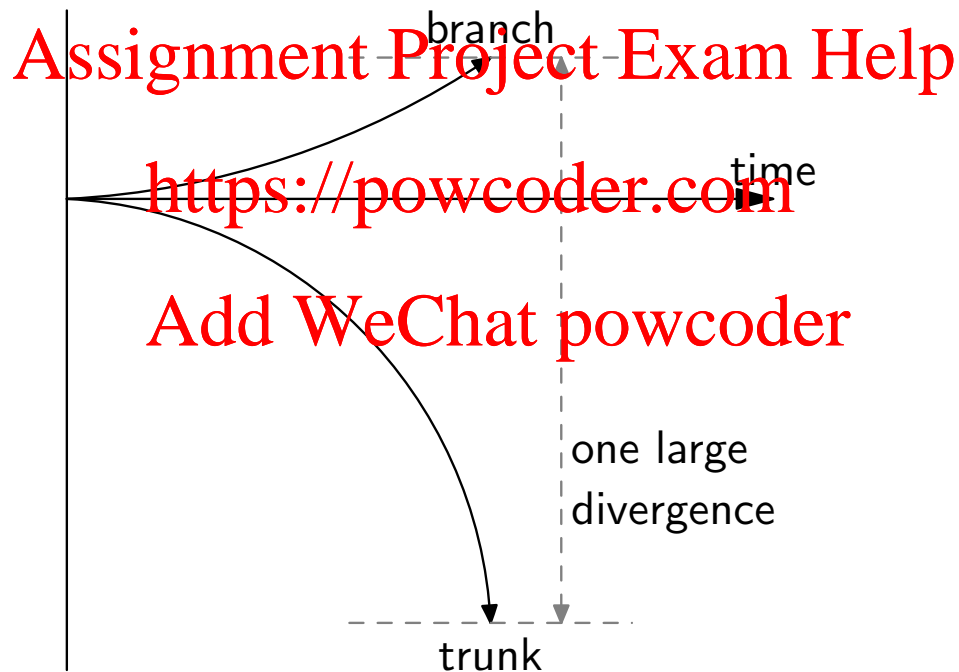- Everyone can see what's happening

Quality assurance:

- Automate the build
- Make your build self-testing
- Test in a clone of the production environment
- Keep the build fast

Deployment:

- Make it easy for anyone to get the latest executable
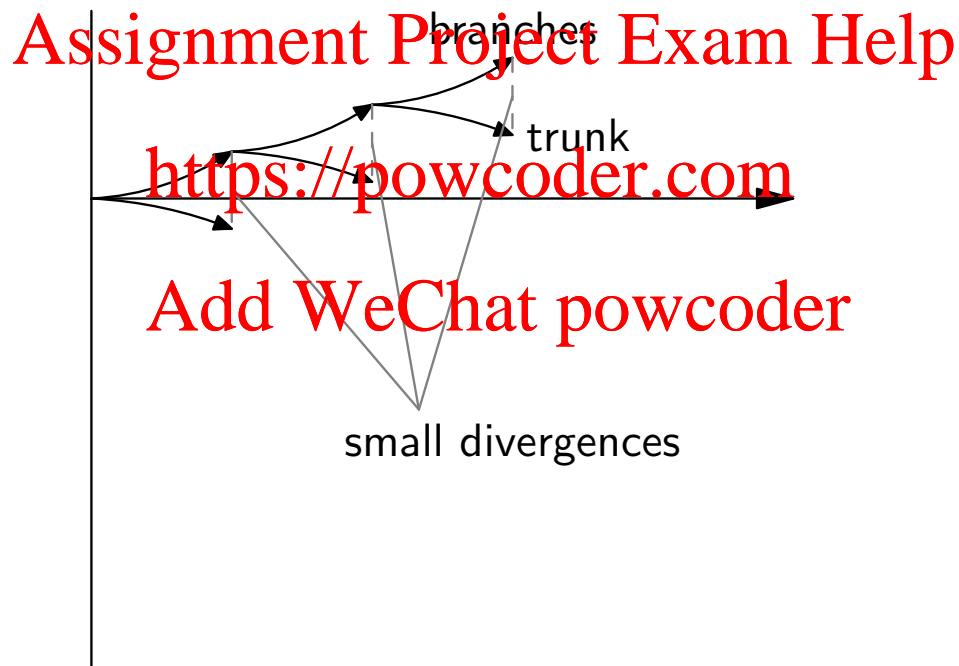- Automate deployment

Integration frequency

(a) large, infrequent commits

Figure: divergence between trunk and feature branches over time

Integration frequency

branches

trunk

small divergences

(b) small, frequent commits

Figure: divergence between trunk and feature branches over time

Feature branching in continuous integration

- Implement the new feature on the trunk, with frequent commits of small changes.

- Implement the new feature in a branch and merge changes from the trunk frequently.

- Implement a feature as a prototype branch and then re-implement the feature on the trunk in small changes.

- Implement the feature as a permanent branch.

Detecting broken builds

every time a change is made to the system:

- The system should be compiled from source in a clone of the production environment.

- A suite of automated regression tests should be executed on the complete system.

- Appropriate static analysis checks (such as code style conformity) should be performed and compared to benchmarks.
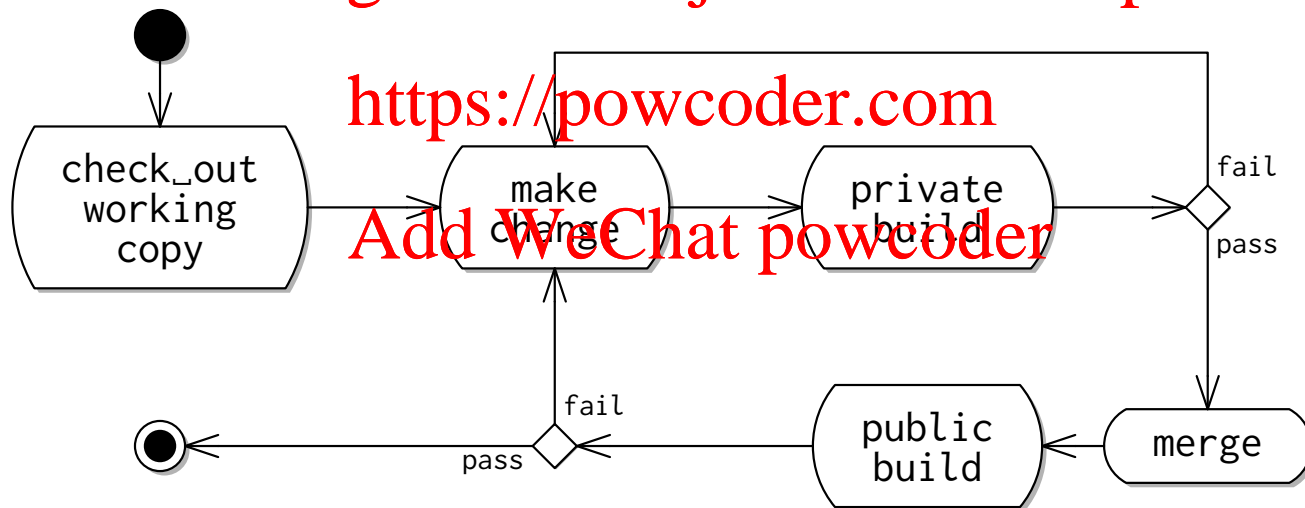
Preventing broken public builds
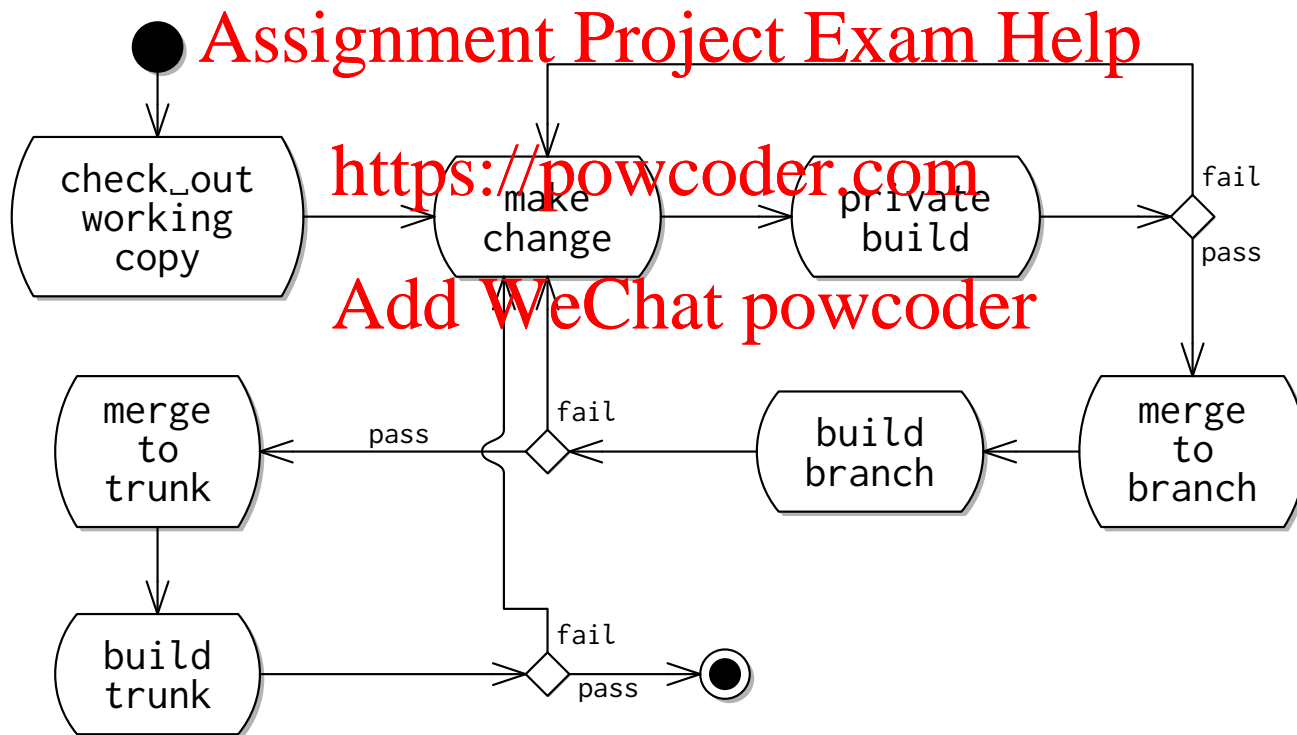

Figure: the continuous integration workflow for a developer

Preventing broken builds with branches

Assignment Project Exam Help

https://powcoder.com

Add WeChat powcoder

Maintaining visibility

- Push notifications such as SMS or email.

- Broadcast mechanisms such as displaying a status monitor on a large screen display in the development area.

CI metrics for project stability

Unsuccessful builds over last period, which measures how frequently a project is broken due to integration conflicts.

Average unsuccessful tests per build can be used to estimate the rate at which defects are introduced into a project by the implementation of new features

Time to build is important to monitor to ensure that a CI process remains viable.

Jenkins build stability reports

Assignment Project Exam Help

https://powcoder.com

Add WeChat powcoder
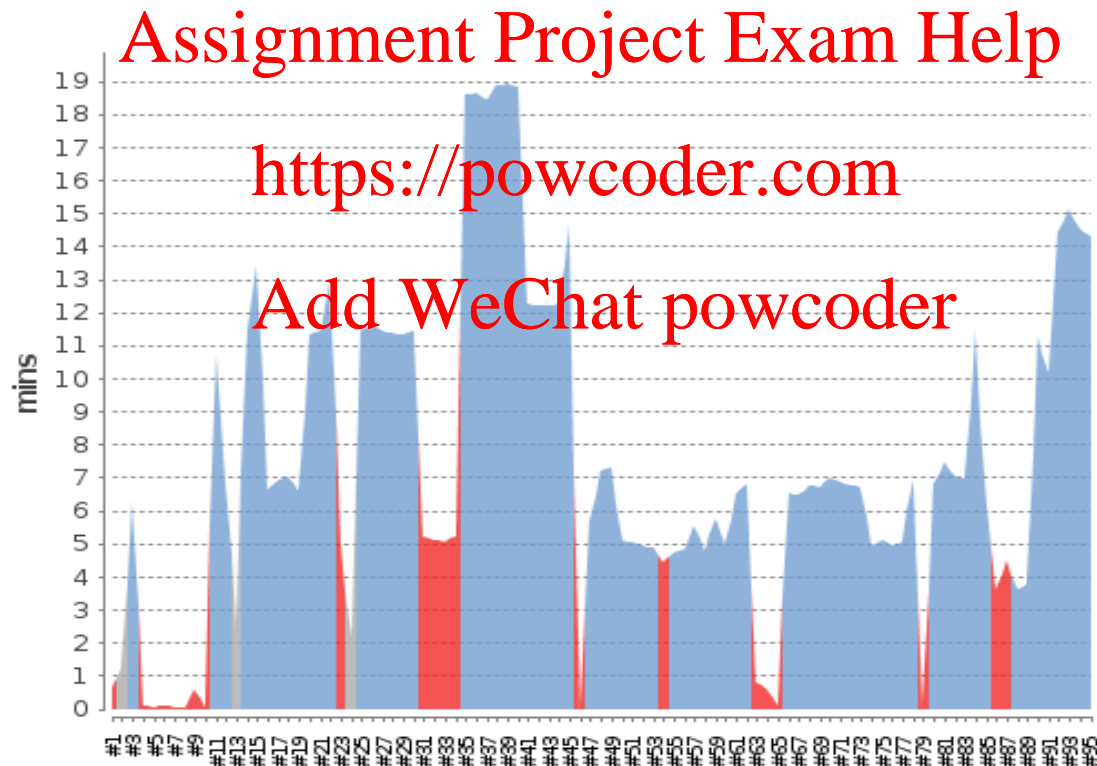
The importance of fast builds

- Delays in completing a build will also delay the discovery of any problems.

- Developers will be deterred from making frequent commits, because the need to wait for a build to complete will reduce productivity.

Tuning build speeds

- Reconfigure the build process to ensure that no unnecessary steps are being undertaken when a component is being built.
- Configure several different types of build that are performed for different parts of the project and/or causes of change.
- Prioritise test cases and then partition them into different categories for different types of build.
- Partition large projects into smaller well defined components that can be developed concurrently.

Continuous deployment

- Ensures that the deployment process is regularly used.

- Different feature variants can be deployed to different users.
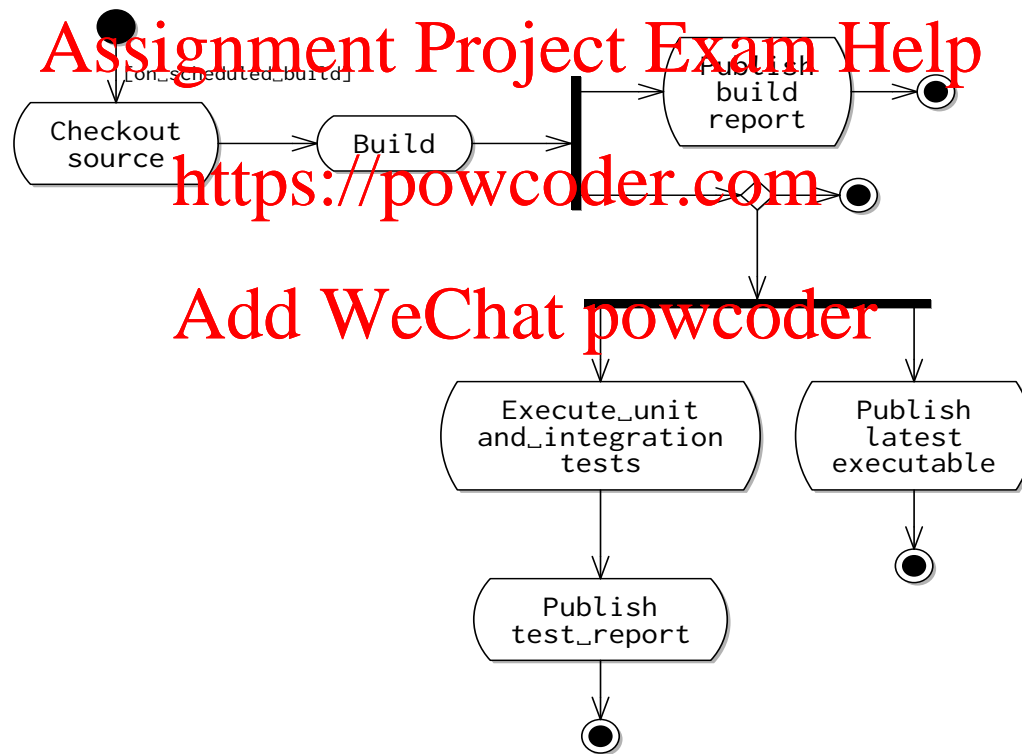
Continuous integration environments

Hudson/Jenkins

TeamCity

CruiseControl

BuildBot

Integrity

Table: some continuous integration environments

Continuous integration environment workflow

Assignment Project Exam Help

https://powcoder.com

Add WeChat powcoder

[on_scheduled_build]

Checkout source

Build

Publish build report

Execute_unit and_integration tests

Publish latest executable

Publish test_report

Key message

Continuous integration practices minimise the disruption caused by rapid, concurrent changes to software systems.

PI Question (session1280)

In AE2, How many bugs have you found?

- A: 1
- B: 2
- C: 3
- D: 4