checkpoint
TECHNOLOGIES
Software Quality. Assured.

TESTING IN A CIT ENVIRONMENT

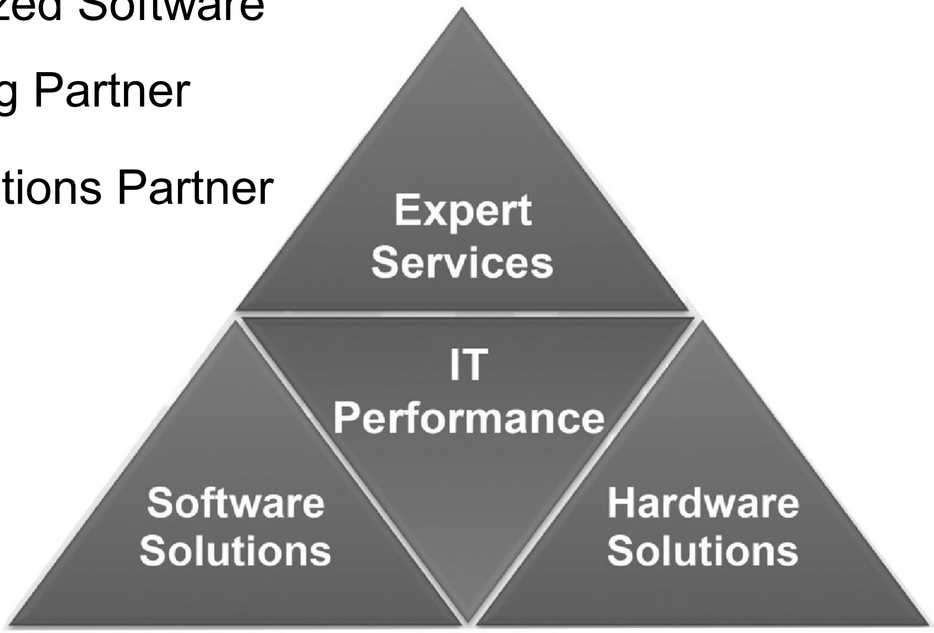
PRESENTER

BRIAN WHITE

Senior Technical Consultant

ABOUT US

- Incorporated in January, 2003
- QA and QC expertise focused on functional, performance and application security validation
- HP Software Gold Partner, HP Authorized Software Support Partner & HP Certified Training Partner
- Mobile Labs, Zephyr and Turnkey Solutions Partner
- QAI Training Partner



Expert Services

IT Performance

Software Solutions

Hardware Solutions

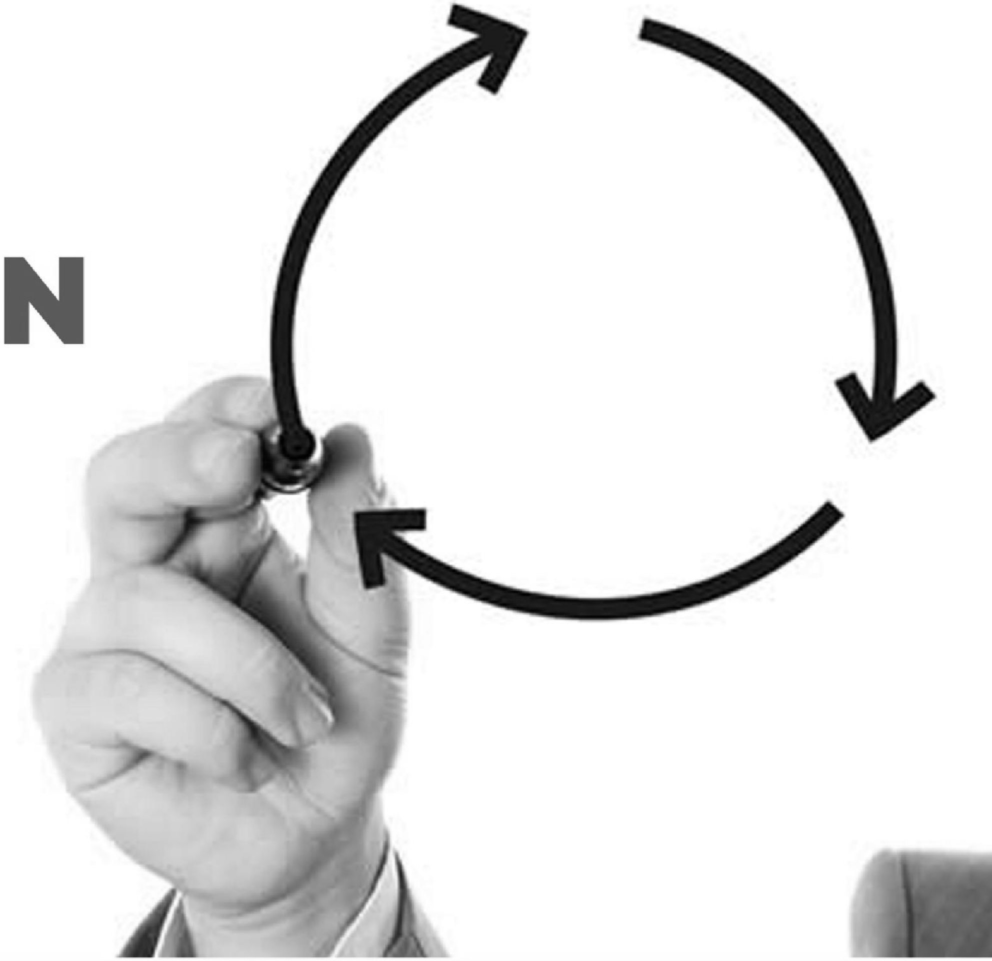
TESTING IN A CIT ENVIRONMENT

STAFF AUGMENTATION	• Long-term contract services, on-site and remote • QA leads, manual testers, automation experts etc.
CONSULTING	• Assessment, installation, configuration, analysis etc. • Long-term and short-term, on-site and remote
OUTSOURCING	• U.S. based functional & performance testing • Software testing performed at our test lab by experts resources
TRAINING	• HP Authorized Training Partner • On-site, virtual or public • Structured classroom format using HP materials
MENTORING	• Customized training essential to your team – in your environment • Cost-effective $$$

checkpoint TECHNOLOGIES

Software Quality. Assured.

ON TO THE PRESENTATION



How would you test this?

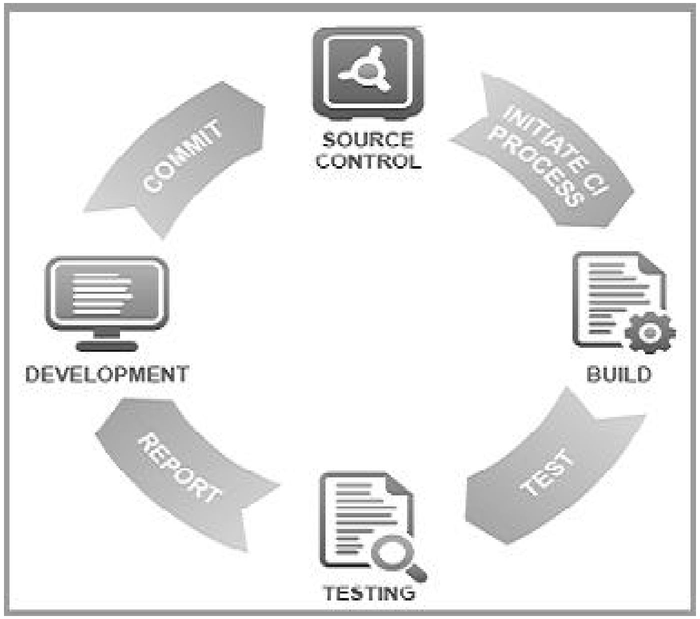
Bill Hamilton is an independent truck driver who just got a big contract with a national coffee chain. He has to deliver a new espresso machine to each of their 200 stores. He knows the distance between each pair of stores and he has to figure out the shortest route to save gas.

Bill goes to Software Now, Inc and asks them to build him a quick application he can run on his mobile phone to tell him the shortest route that will take him to each store. Your company employs the agile process of Extreme Programming (XP). How will you and your company test Bill's application?

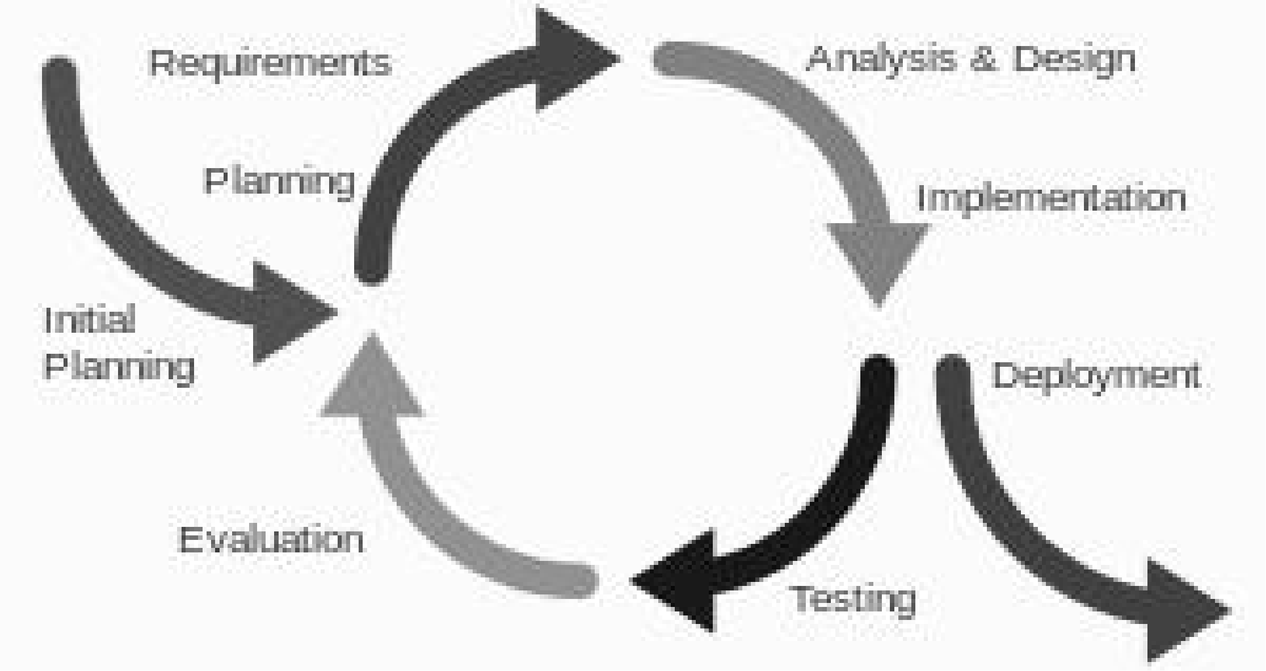
What is CIT?

John Walker in his blog:
"Beyond Scrum: Continuous Integration with Build and Test Automation

"Organizations clearly need to invest in automated build and test processes if they want to scale up and deliver features faster and release more frequently. This investment can be expensive, but manual methods are obviously not scalable. Also, automated build and test processes tend to produce much higher software quality."



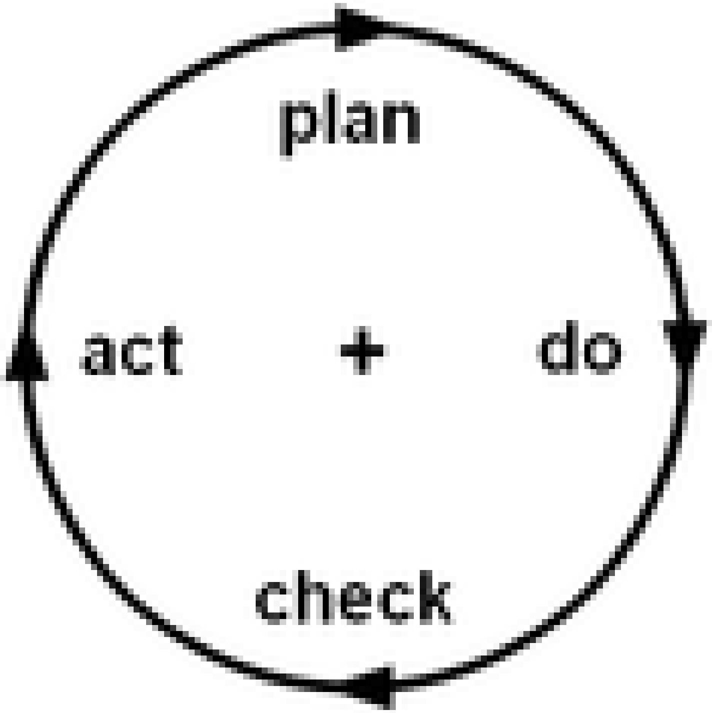
Zbyszek Mockun in his technology blog says



"CI allows us to run automated tests after each commit and send feedback with results to developers. Good automated tests should cover all functionality, or at least most of it."

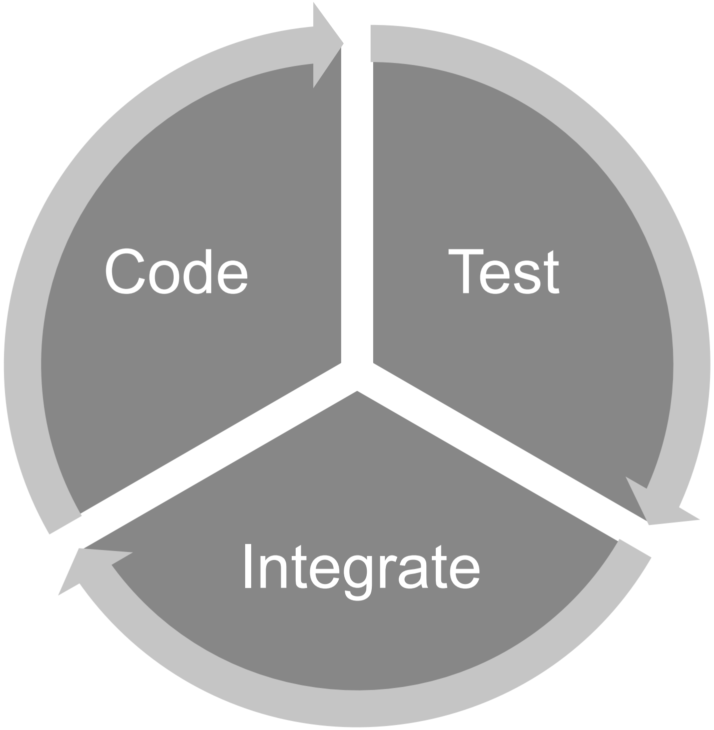
A blog post from: **Margret Rouse**

"Continuous integration (CI) is a software engineering practice in which isolated changes are immediately tested and reported on when they are added to a larger code base."



Wikipedia says

(and what discussion would be complete without a good “crowdsourced” definition?)



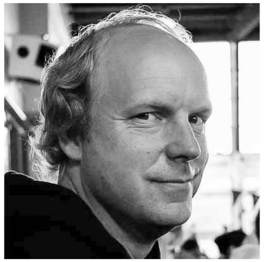
“Continuous integration (CI) is the practice, in software engineering, of merging all developer working copies to a shared mainline several times a day.”

Kent Beck says in his book, **Extreme Programming Explained: Embrace Change (2nd Edition)**

“Integrate and test changes after no more than a couple of hours”

How would you define CIT?

Why did we give Kent Beck that last word?



Kent Beck is credited with creating the practice of Extreme Programming (XP)

XP Practices:

- **Collaboration**
  - Sit Together
  - Whole Team
  - Pair Programming
- **Sharing Information**
  - Informative work place
- **Planning**
  - Stories
  - Energized Work
  - Weekly and Quarterly Cycles
  - Slack

Why aren't software developers usually the best software testers?

Glenford J. Myers gives the following definition of testing in his book, The Art of Software Testing:

"Testing is the process of executing a program with the intent of finding errors."

What is Kent Beck's CIT process?
(I am going to over simplify now!)

- Developers write code
- Every couple of hours developers integrate their new code into the build
- Build process is automated
- If…
  - Build breaks, responsible developer is notified
  - Build works
- Developers write more code either to fix problems or to build more functionality

Where do testers fit in?

- **In the section called Testers, Beck defines the role for that group on the XP team**
  - Help write automated system level tests
  - Coach programmers on testing techniques
  - Write tests that go beyond happy path testing for the given stories
  - Ensure system-level tests succeed only after the stories are fully implemented
  - Write more tests
  - Help automate and tune tests
  - Pair with programmers to help decipher “knotty testing” problems

Some possible pitfalls to avoid: Letting the fox guard the hen house

- Letting the development team take over testing
- Come on, even Earnest Hemingway and J.K. Rowling had editors

Some possible pitfalls to avoid:
Not setting expectations for your customers

- When your customers, be they internal or external, hear that testing is being automated they can get the wrong idea

- Make sure your customers understand what CIT covers and what SQA testing covers
  - Computers are incredibly fast, incredibly accurate and incredibly stupid
  - People are incredibly slow, incredibly inaccurate and incredibly brilliant
  - Make sure your customers understand the value of both automated testing and manual testing

Some possible pitfalls to avoid:

Not checking periodically that CIT tests are still valid and reliable

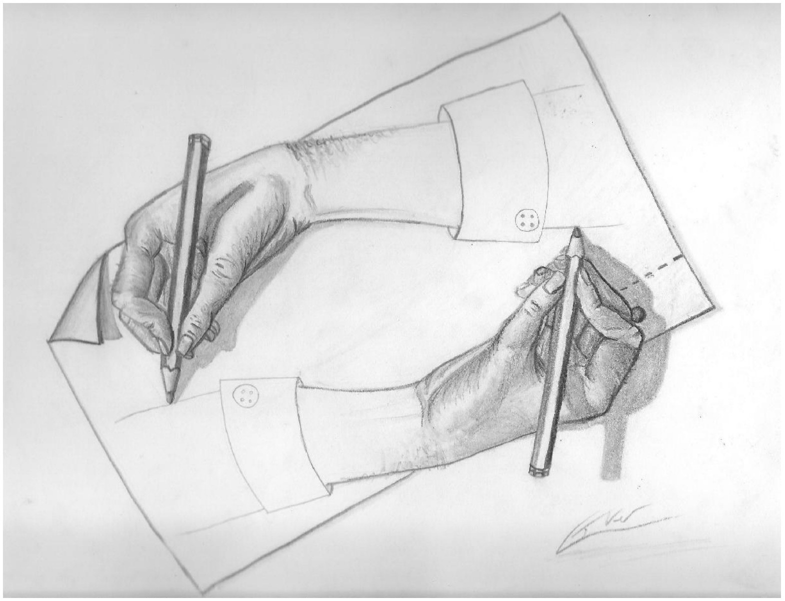
- What if the build doesn't break but some functionality changes in such a way the CIT test doesn't pick up the error?
- Make sure you keep an eye on legacy tests

Some “Best” practices:

Develop a CIT test strategy with development

Stagger CIT testing with SQA automated regression testing

- The, “Build Unit Test” should be run for every build and has to be fast
- Last build of the day kicks off an automated Regression test?
- SQA has to determine which tests get run when?
- What is the best “overlap” of SQA and Development testing?



Some “Best” practices:
How does the testing approach adapt?

- CIT testing has to adapt to changes in code without exceeding code construction time
- How are tests added to the CIT test?
- How are tests added to the SQA suite that tests beyond CIT?
- When is the CIT test too big?
- What is the scope for CIT and how does your team know when they exceed that scope?
  - Why is that important?
- How do you know what your tests are missing?

Some “Best” practices:
Where is your test repository?

Where are tests run during CIT stored?

- Can your CIT test access all of your assets?
- Can you automate the test pull?

What does your system do on a fail?

- Does the CIT test move on to the next test and send notifications?
- Does it stop?
- Why does it do either?

Some “Best” practices:
Who is going to do all this work?

Everyone is already too busy, how do you get this done?

- Hire contractors to get the first set of tests up and running?
  - Once they have your CIT test running it may free up time spent manually regression testing so your team can take over
- Delay releases such that your teams can dedicate a specific time period to getting the CIT testing running
- Determine cost savings to see if hiring specific full-time CIT resource(s) makes sense

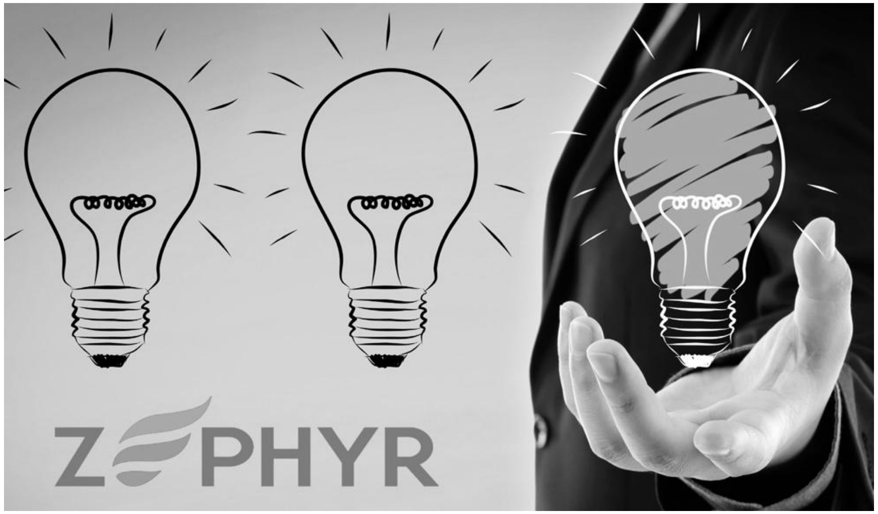
Don’t forget to ask:
What is your company’s economic commitment to test?

- Are SQA and Development resources costs to your company or do one or both generate revenue?
- Is your company committed to quality?
- Does your company only test to meet auditing requirements?
- Why does your company TEST?

What are some tools you can use?

- All the usual suspects for testing still apply:
  - HP Products like Application Lifecycle management, Unified Functional Test, Performance Center
  - Selenium
  - Jira
  - Zephyr
  - Jenkins
  - Microsoft Visual studio
  - Etc.
- Maybe you’ll write your own tools?

Let’s talk about one of these tools:



Zephyr is a test resource management tool

Why is it relevant to CIT?
It interfaces with Jenkins
For a demo of the Jenkins plug in, check this out:
https://www.youtube.com/watch?v=_IT-BvxNttM

Now lets take a quick look

ZEPHYR

DEMO

Q&A

checkpoint
TECHNOLOGIES
Software Quality. Assured.

THANK YOU FOR ATTENDING!

BRIAN WHITE
Senior Technical Consultant
bwhite@checkpointech.com
www.checkpointech.com