

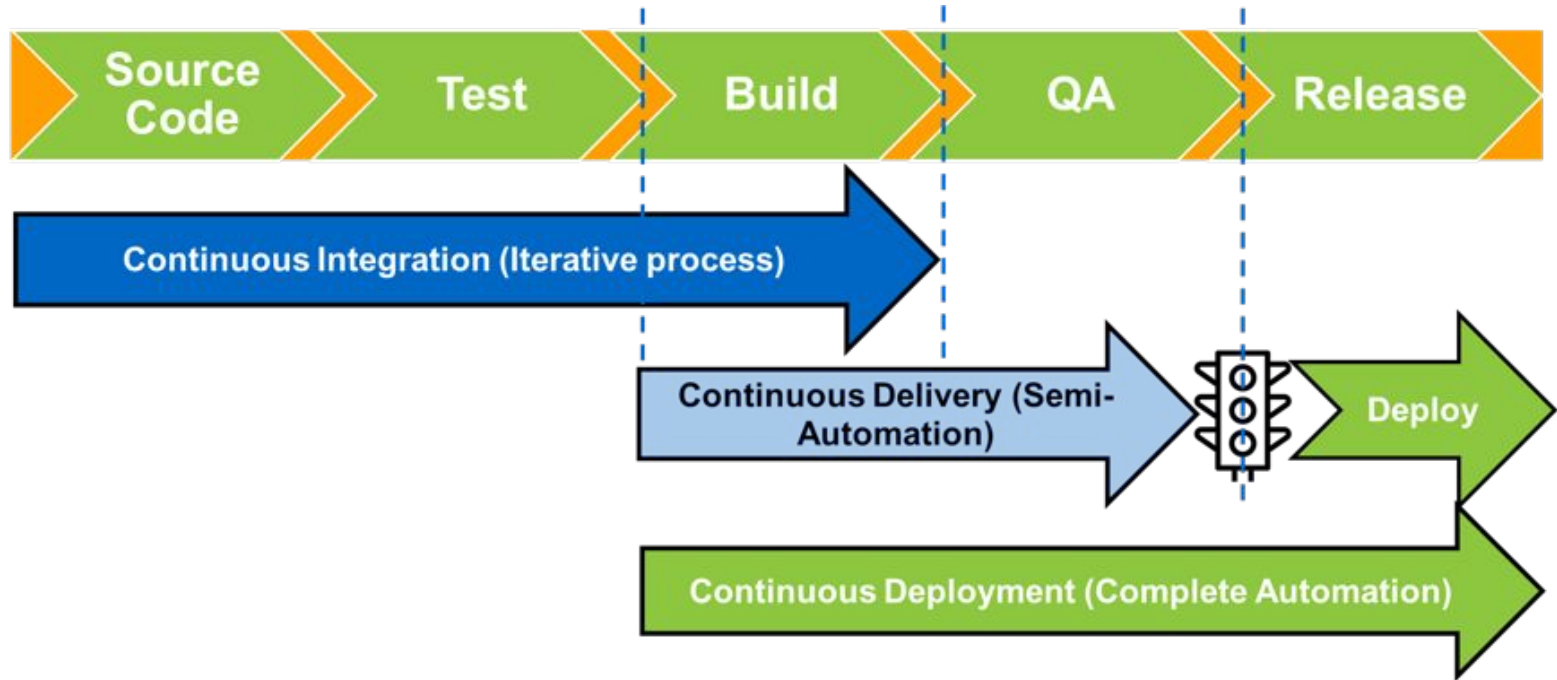


# Pipes, Blocks and Pipelines

David Becvarik



# CD vs CD





Promised Failures!





**Stephen Diehl**

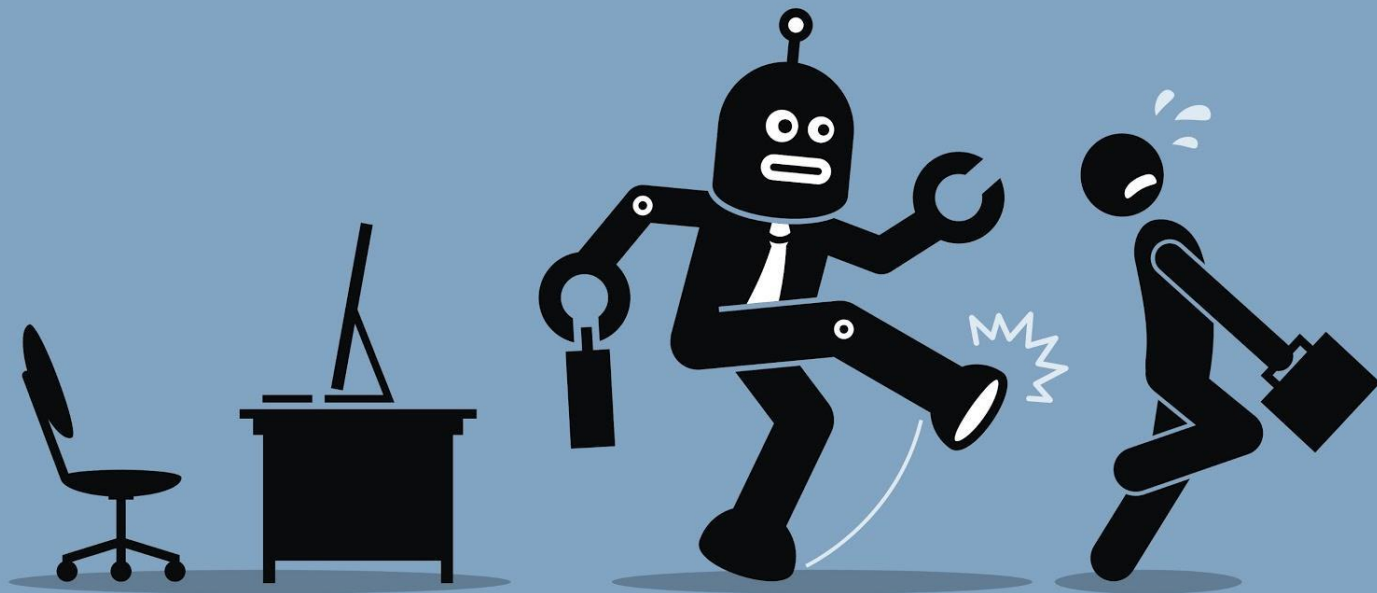
@smdiehl



I think software has just reached a point where we like artificial complexity because it creates the illusion of progress in maintenance.

12:18 PM · Oct 30, 2019 · [Twitter Web App](#)

# Wrong CI/CD server



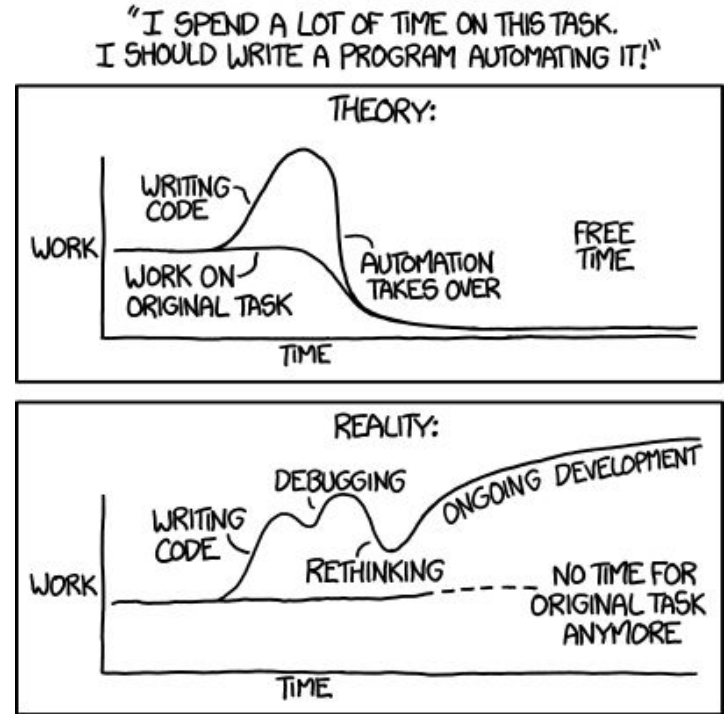
# Configuring jobs from UX

- Not reproducible builds
- Missed steps
- Hard to debug



# Wrong stuff to automate

- Repetitive task first
- Start with easier stuff like containers
- Do not ignore legacy!
- Integrate with other tools (Ansible)
- Be prepared for scale



# Random build results

- Not reproducible builds
- Missed steps
- Hard to debug

```
int getRandomNumber()  
{  
    return 4; // chosen by fair dice roll.  
              // guaranteed to be random.  
}
```



# Delivering failures

- Ignoring test results
- Issues in production
- Hotfixes
- Pretending DevOps/Agile

# Forgetting Code Review

- CICD is not replacement for code review
- Missing test means bad CI
- Code smell is slowing you down



How to spot good CI/CD story?



# Pipeline as a code

- Easy to version
- Easy to extend
- Easy tintegration

```
apiVersion: tekton.dev/v1alpha1
kind: Task
metadata:
  name: check-stuff-file-exists
spec:
  inputs:
    resources:
      - name: workspace
        type: git
        targetPath: newworkspace
    params:
      - name: args
        type: array
  steps:
    - name: read
      image: ubuntu
      command: ["/bin/bash"]
      args: ['$(inputs.params.args)']
  ---
```

Security is crucial

Security is integrated

Security is continuous

# ~~Freedom~~ Responsibility to the Devs!

Lets the dev teams choose tools and way

Define deliverables

Define requirements



Understand the code you deploy





I have a dream, my CI/CD ...