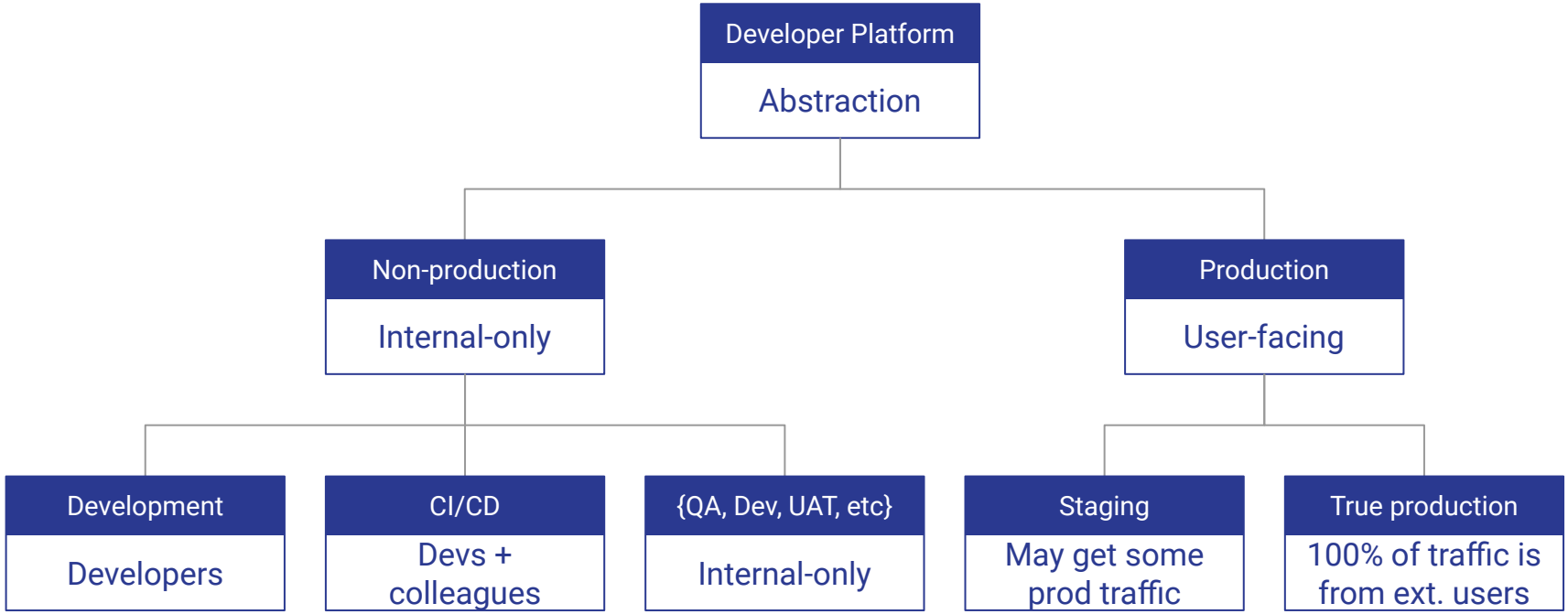
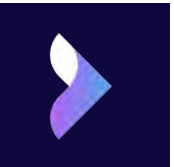
Declarative everything: a GitOps/automation-based approach to building efficient developer platforms

Conf42 DevOps 2024

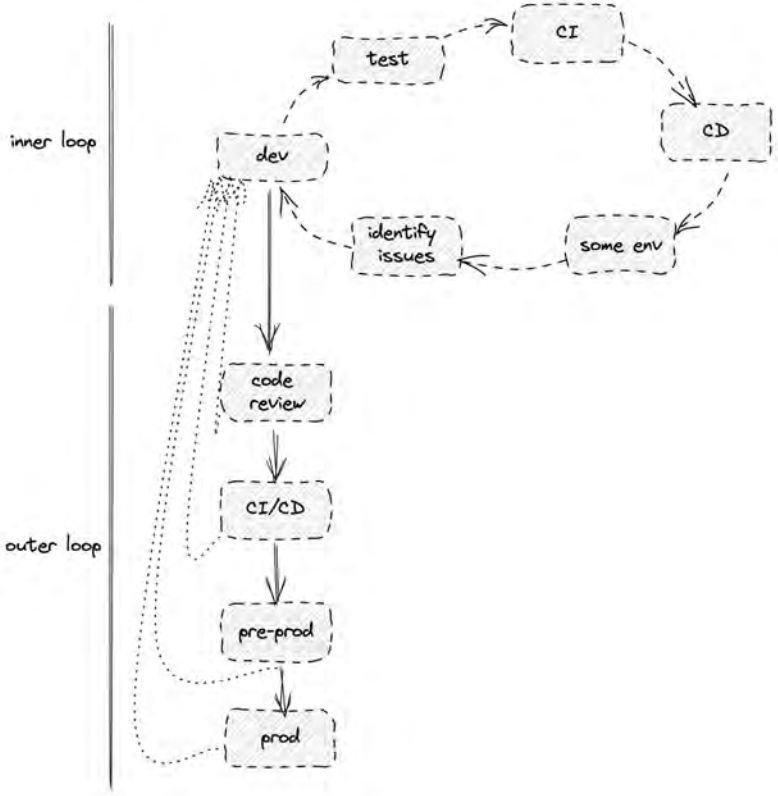
Debo Ray

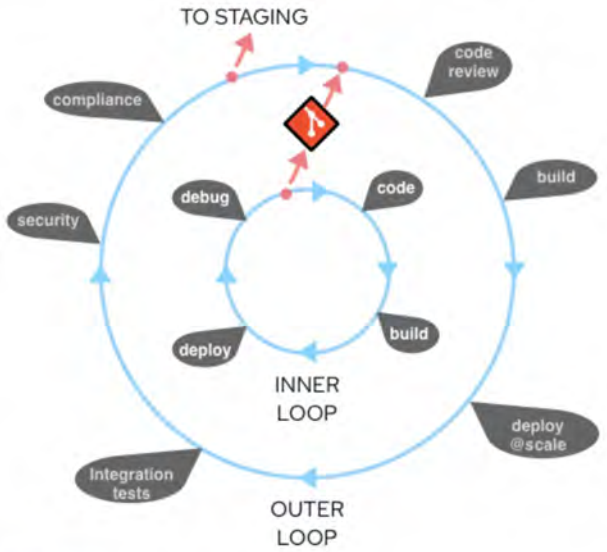
Developer platform coverage



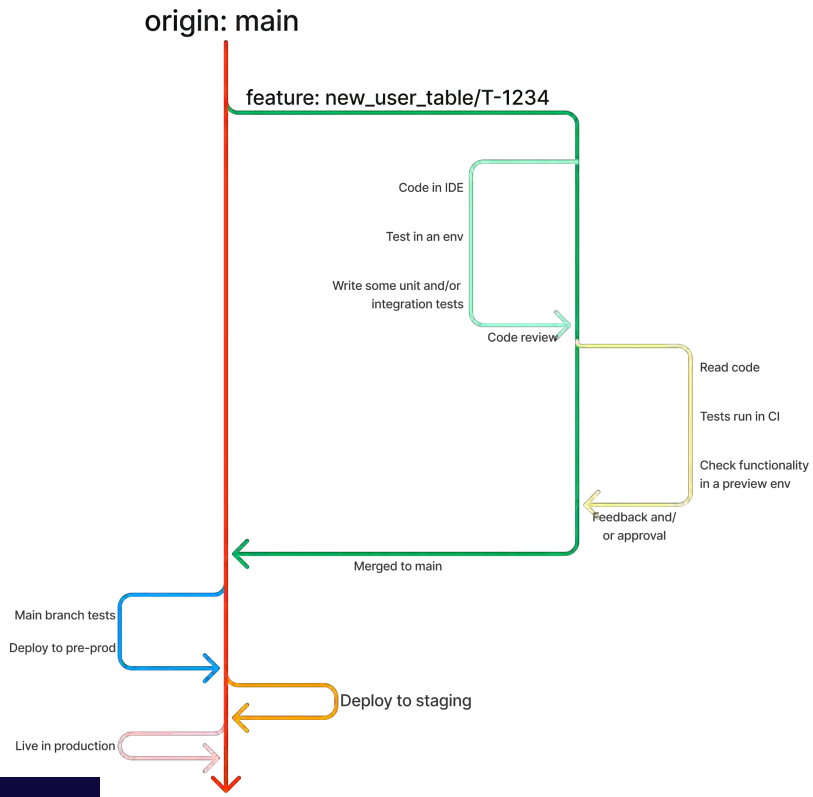


Understanding Developer Workflows

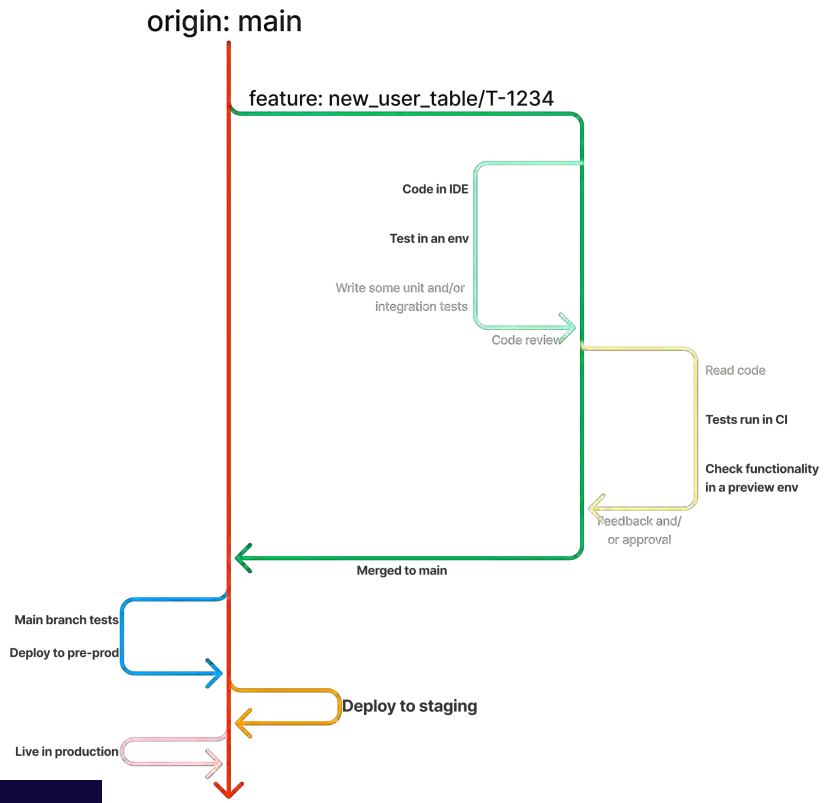




Source: Red Hat

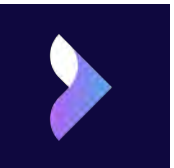


How changes get to production



Where can we benefit from having automation?

GitOps



What: DevOps best practices applied to infrastructure automation (*“golden paths”*)

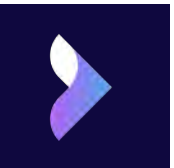
Focus: Automation/standardization improves the dev experience and helps reduces cost

How: Git-based access to consistent/symmetric environments

Why: Continuous integrations leads to faster development and deployment

An Environment

What *exactly* is it?



Application runtime

Relevant {dev, test, ..} data

Access to downstream services

Accessible from upstream services

Accessible to a human

Isolated from production (if not prod)

Source code, build tools (+ IDE)

{Shared, dedicated} tenancy

We need to rethink our general “software verification” strategies

Why does testing exist (emphasis on services, **not** libs/modules)?

Functionality → Interoperability → Confidence

Acceptance criteria

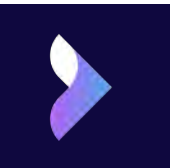
- Is my feature/module meeting product requirements?
- Am I calling downstream APIs per their contracts?

Integrate w/ existing apps/services/patterns

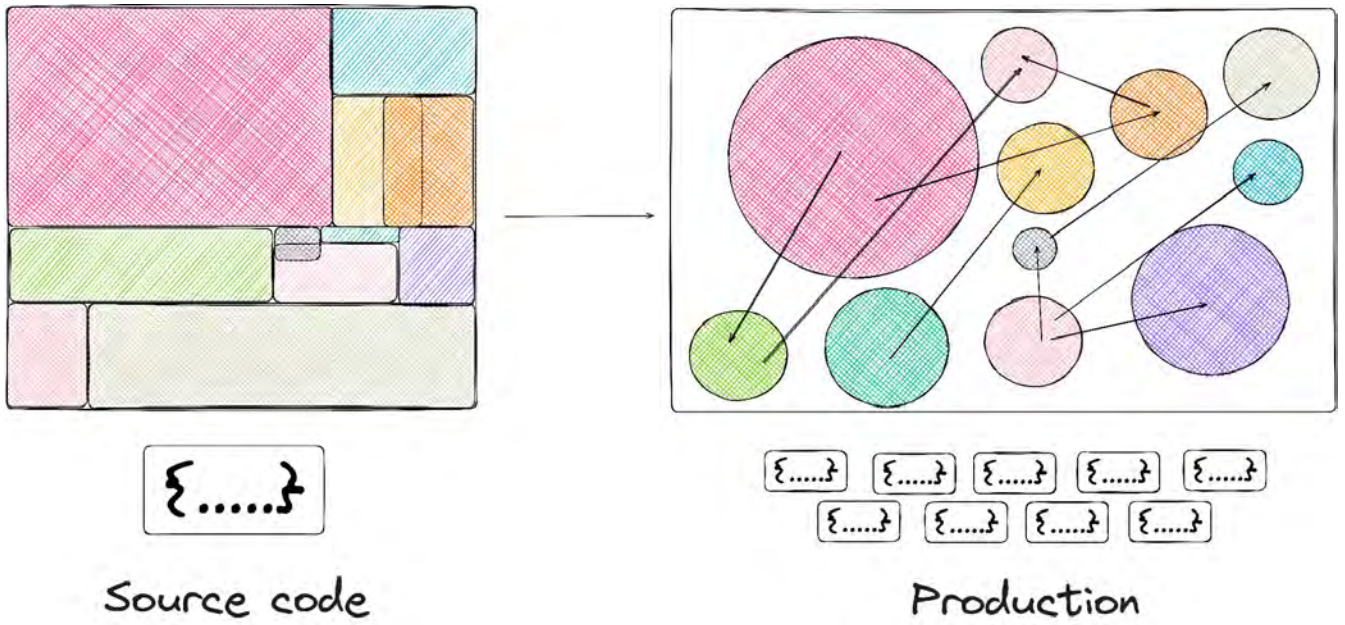
- Although API contracts are met, will these calls actually {return the data, perform the action} I need?
- Are the {latencies, request/response shapes} acceptable?

Reliability in prod

- Will this make future deployments error-prone?
- Is this reducing idempotence?

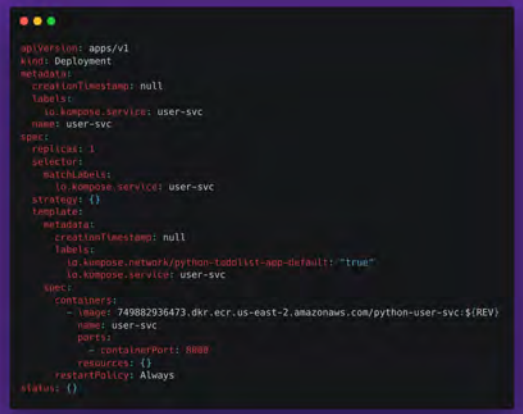


Source code …. Running in production

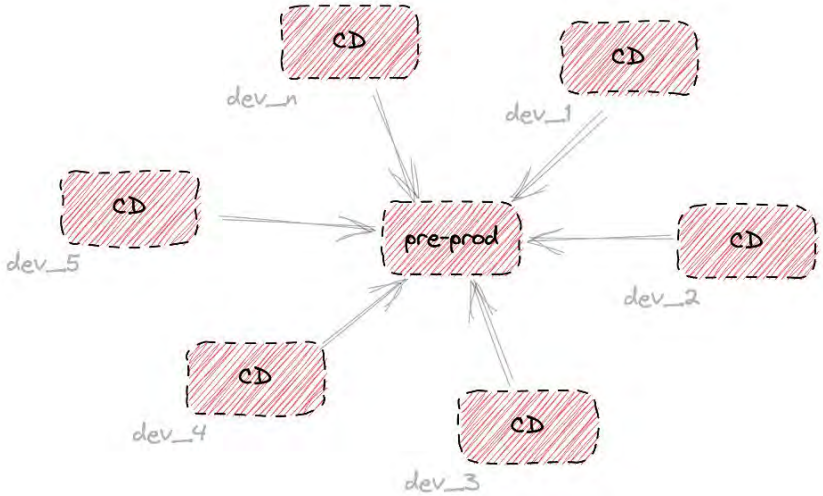


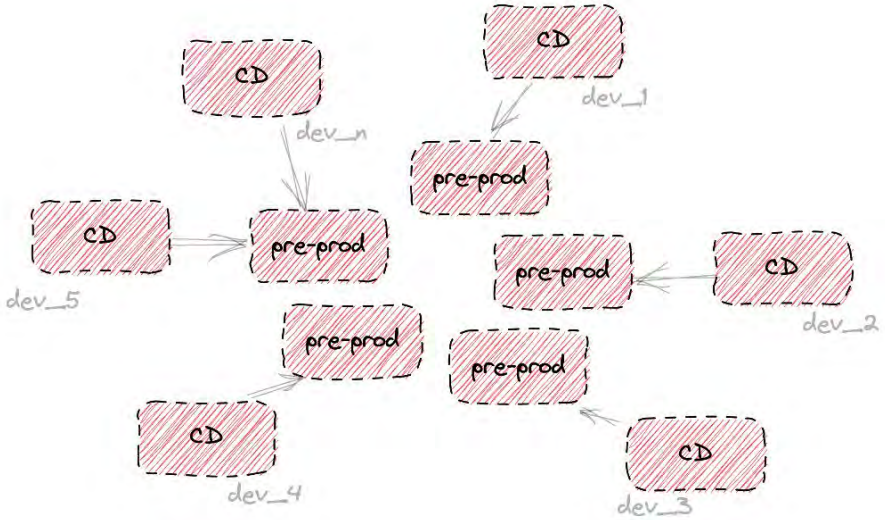
[Production] K8s Deployment YAML, Helm charts etc

```yaml
apiVersion: v1
kind: Service
metadata:
  annotations:
    kompose.cmd: kompose convert -f python-todolist-app/docker-compose.yml
    kompose.version: 1.28.0 (c4137012e)
  creationTimestamp: null
  labels:
    io.kompose.service: app
  name: todolist-app
spec:
  ports:
    - name: "6000"
      port: 6000
      targetPort: 6000
  selector:
    io.kompose.service: todolist-app
status:
  loadBalancer: {}
```



Dev -> Code Review -> CI/CD -> **Pre-Prod** -> Prod

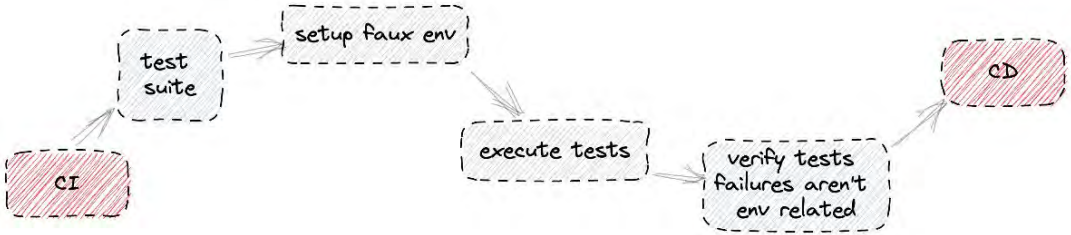


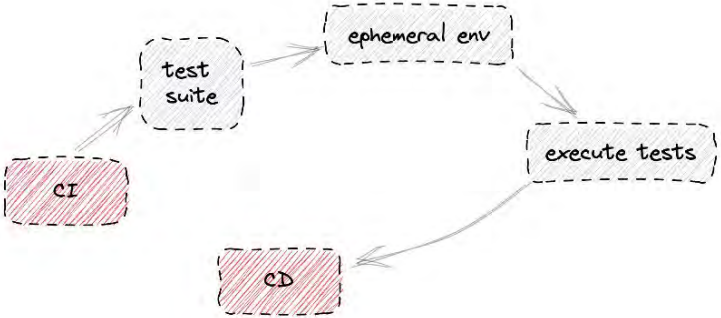


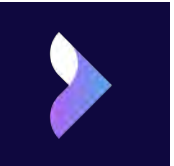
CI might have a flavor of building images, usually not

```yaml
name: Build and Deploy TodoList App
on:
  push:
    branches:
      - main

jobs:
  build-and-deploy:
    runs-on: ubuntu-latest

    steps:
    - name: Checkout code
      uses: actions/checkout@v2

    - name: Set up AWS CLI
      uses: aws-actions/configure-aws-credentials@v1
      with:
        aws-access-key-id: ${{ secrets.AWS_ACCESS_KEY_ID }}
        aws-secret-access-key: ${{ secrets.AWS_SECRET_ACCESS_KEY }}
        aws-region: us-east-1

    - name: Login to Amazon ECR
      uses: aws-actions/amazon-ecr-login@v1

    - name: Build and Push Docker Image
      run: |
        docker build -t your-ecr-repo-uri:latest .
        docker push your-ecr-repo-uri:latest

    - name: Deploy to Kubernetes
      run: |
        aws eks --region us-east-1 update-kubeconfig --name your-eks-cluster

        kubectl apply -f path/to/kubernetes/deployment.yaml
        kubectl apply -f path/to/kubernetes/service.yaml
```

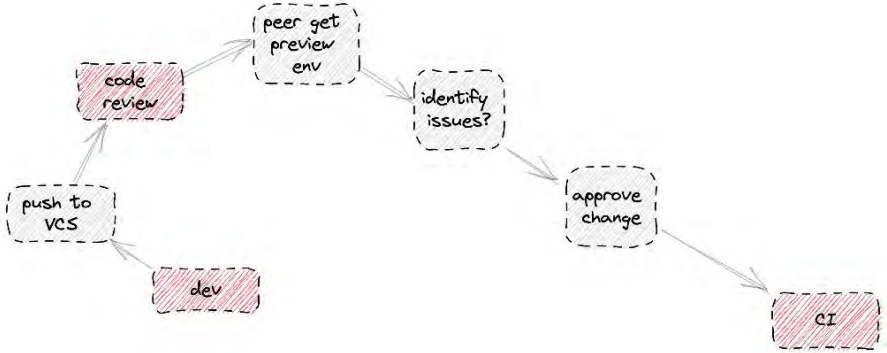
Dev -> Code Review -> **CI/CD** -> Pre-Prod -> Prod







Dev -> **Code Review** -> CI/CD -> Pre-Prod -> Prod

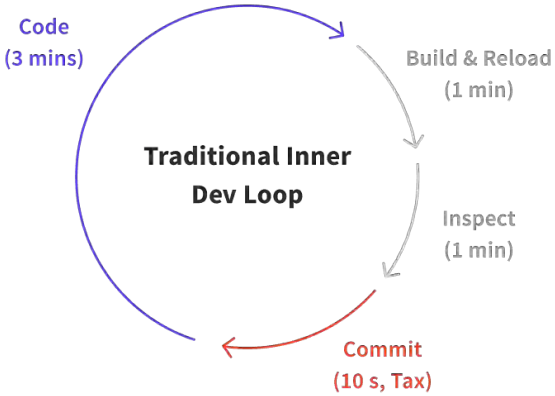


Dev (and CI) usually resorts to docke

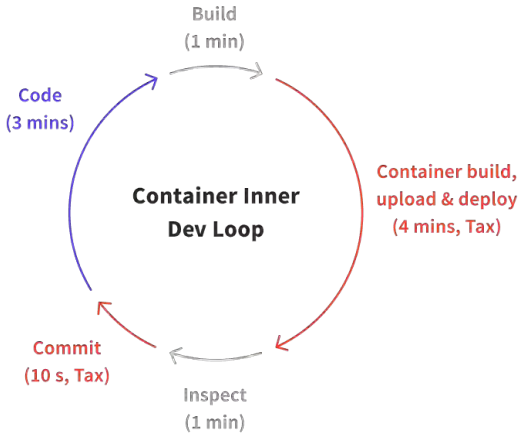
```yaml
version: "2"
services:
  todolist-backend:
    build: ./app
    links:
      - db
      - user-svc
    ports:
      - "6000:6000"
    environment:
      USER_SVC_HOSTPORT: http://user-svc:8000
  db:
    image: mysql:5.7.28
    ports:
      - "32000:3306"
    environment:
      MYSQL_ROOT_PASSWORD: root
    volumes:
      - ./db:/docker-entrypoint-initdb.d/:ro
  user-svc:
    build: ../python-user-svc/app
    links:
      - db
    ports:
      - "8000:8000"
  todolist-web:
    build: ../web-client
    links:
      - user-svc
      - todolist-backend
    environment:
      USER_SVC_HOSTPORT: http://user-svc:8000
      TODOLIST_APP_SVC_HOSTPORT: http://todolist-app:6000
    ports:
      - "8088:8080"
```

```yaml
version: "2"
services:
  app:
    build: ./app
    links:
      - db
    ports:
      - "8000:8000"
  db:
    image: mysql:5.7.28
    ports:
      - "32000:3306"
    environment:
      MYSQL_ROOT_PASSWORD: root
    volumes:
      - ./db:/docker-entrypoint-initdb.d/:ro
```

Deep-dive: Inner-loop of SDLC



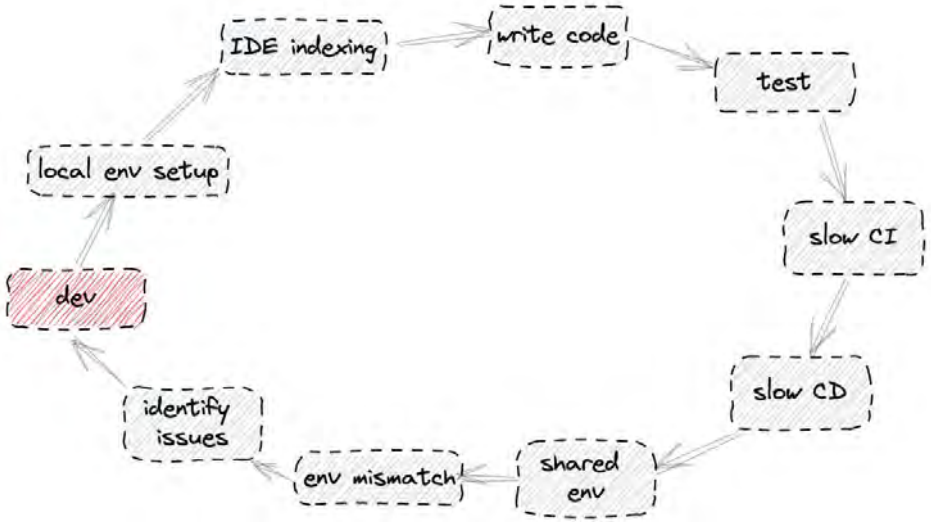
~5 mins

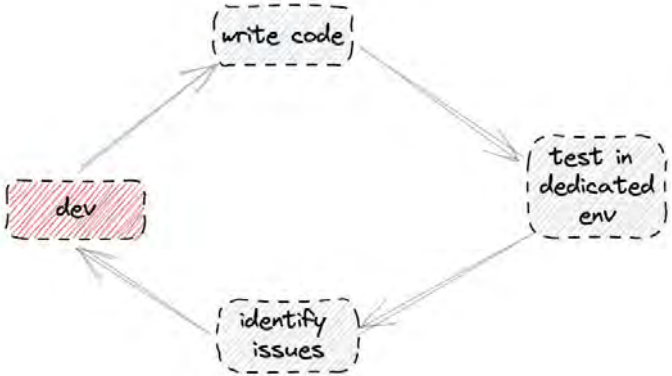


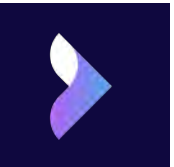
~9 mins

+45%

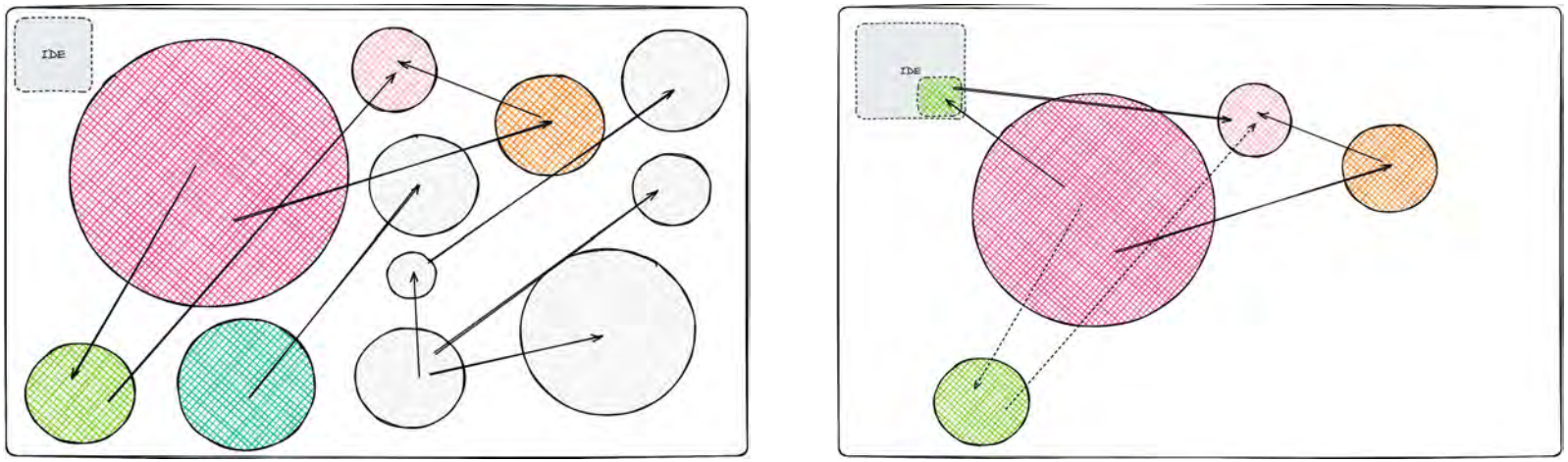
Dev -> Code Review -> CI/CD -> Pre-Prod -> Prod



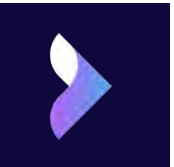




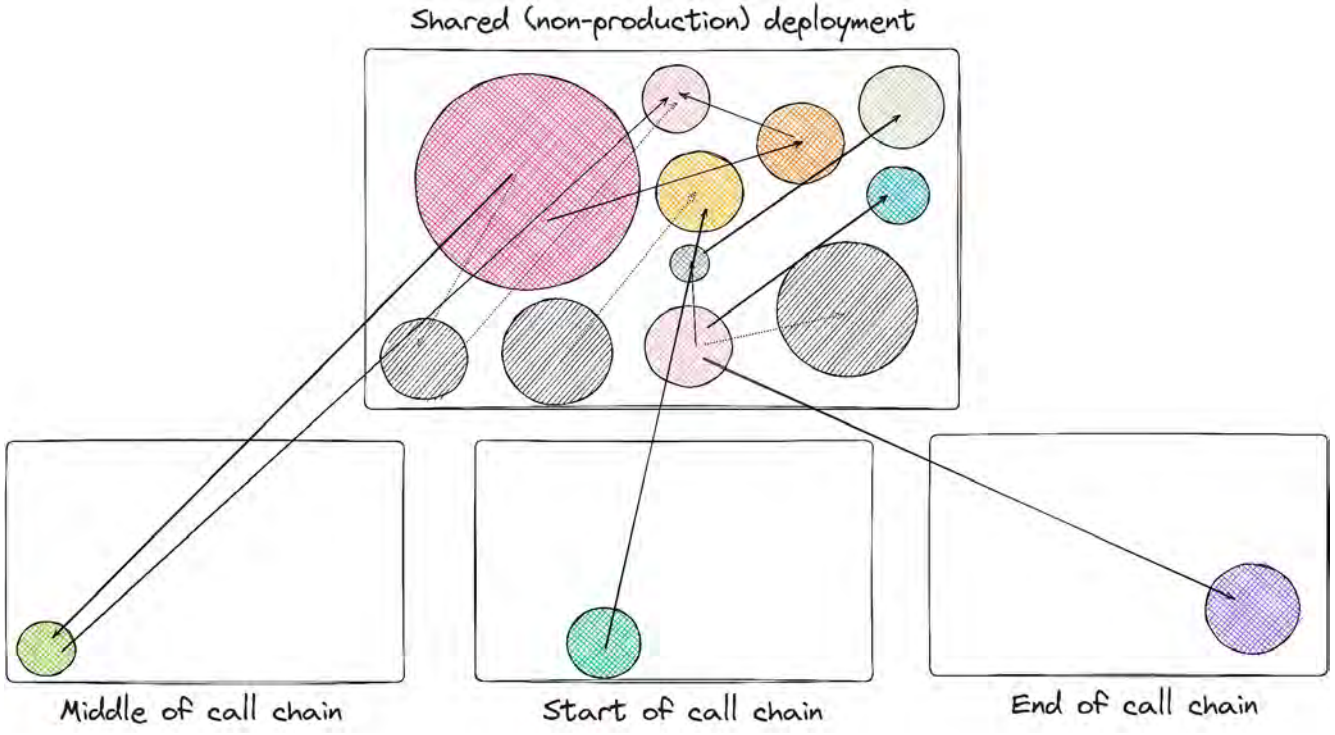
Development purposes (“local dev”, IDE etc)

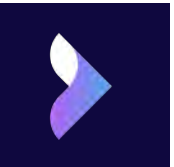


When service dependencies are somewhat isolated



Testing purposes (“local dev”, CI, etc)



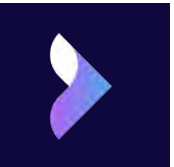


Takeaways

Focus: Issues are always easier to fix when caught before production

How: Easier access to production-symmetric environments

Why: Remove drift between SDLC stages w/ better dev ergonomics



3-up Context Slide

Company

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

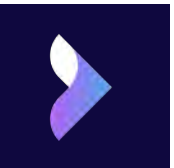
Context

Ut enim ad minim veniam, quis nostrud exercitation

- Duis aute irure dolor in reprehenderit in voluptate velit

Problem statement

Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.



3-up Flow Slide

Challenge 1

Expand audience

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Challenge 2

Up 30-day actives

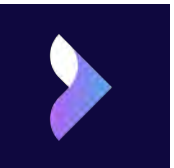
Ut enim ad minim veniam, quis nostrud exercitation

- Duis aute irure dolor in reprehenderit in voluptate velit

Challenge 3

Increase conversion

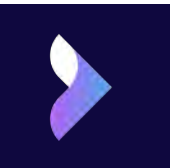
Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.



Thing

Moar Text Goes Here

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

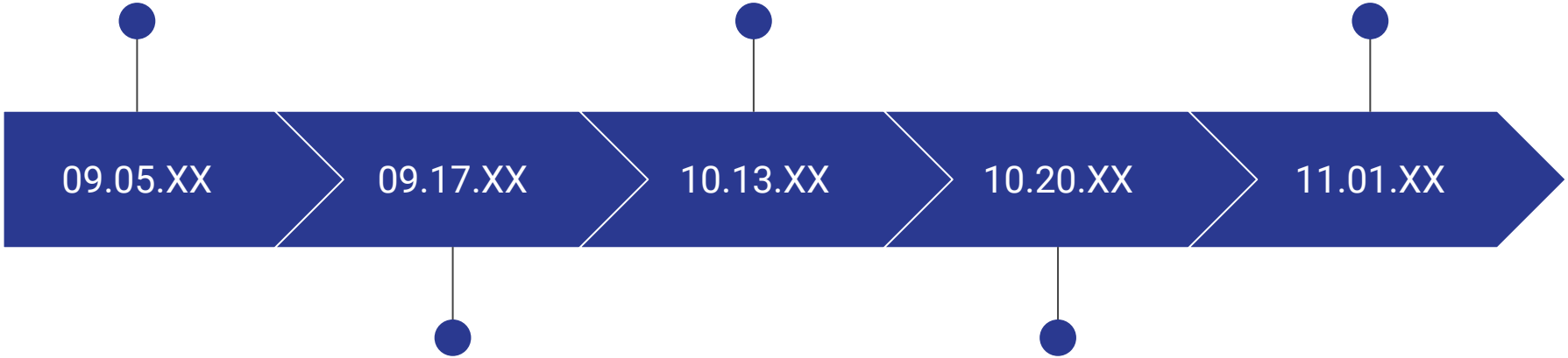


Big Thought/Idea

Lorem ipsum dolor sit amet, consectetur adipiscing elit

Lorem ipsum dolor sit amet, consectetur adipiscing elit

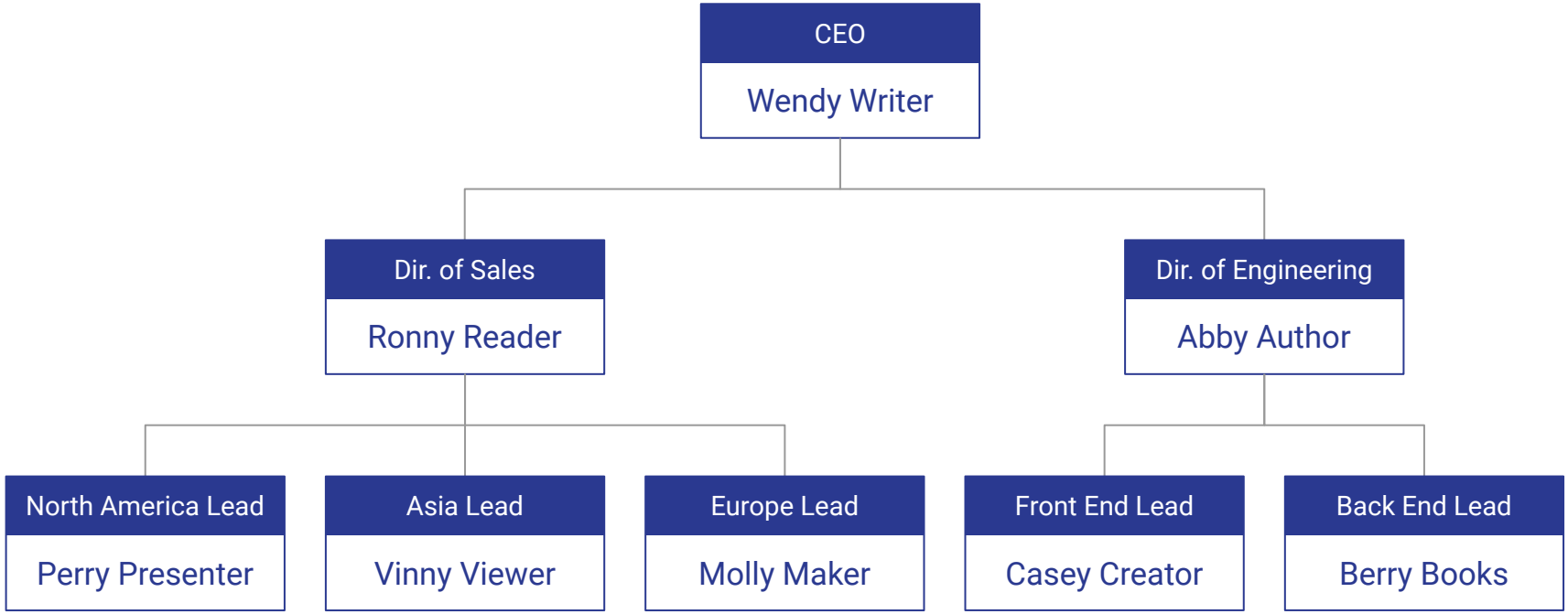
Lorem ipsum dolor sit amet, consectetur adipiscing elit



Lorem ipsum dolor sit amet, consectetur adipiscing elit

Lorem ipsum dolor sit amet, consectetur adipiscing elit

A tree of people



Impact

XX% sales increase