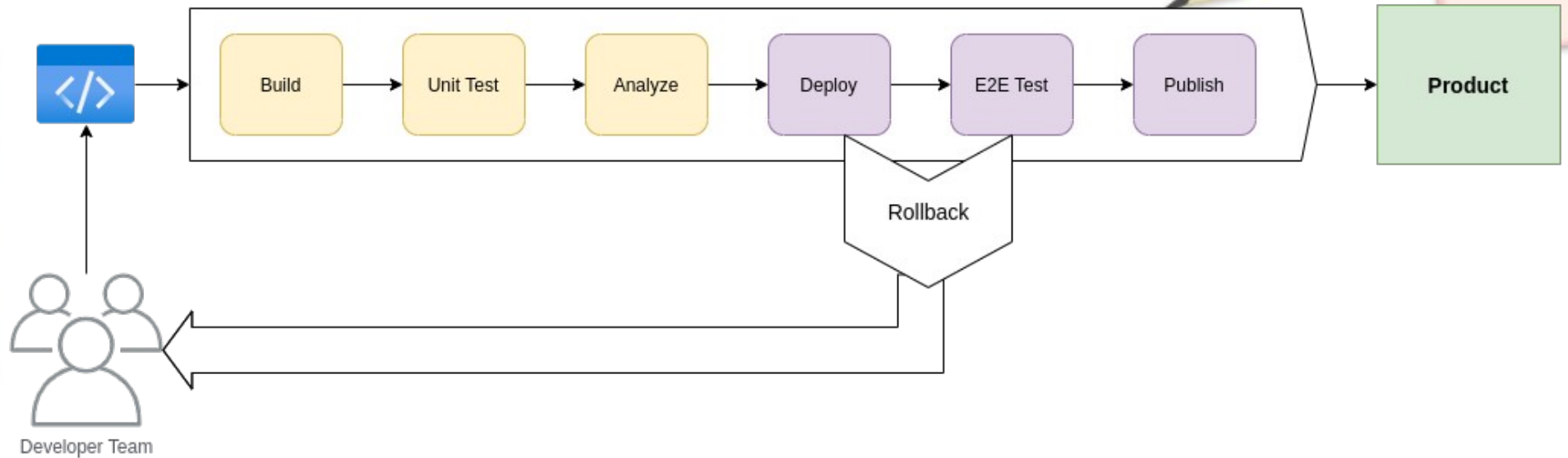




CI/CD Pipelines

Fundamentals and Benefits

Fundamentals



- CI = Continuous Integration
 - Merging Code from several developers to one code base
- CD = Continuous Deployment
 - Deploy and verify that the product is working fine
- CI + CD = Continuous Delivery
 - Transforming code to a product, automatically

Automate and remove failures

Where Does CI/CD Fit In?

Stage	<i>Before CI/CD</i>	<i>After CI/CD</i>
Coding	<i>Human</i>	<i>Human</i>
Code Review	<i>Human, Subjective, Inconsistent</i>	<i>Human/CI - Static Analysis</i>
Compile/Lint	<i>Human</i>	CI
Merge/Integrate	<i>Human</i>	CI
Run Unit Tests	<i>Human, Hit or Miss, Easily Bought Off with Pressure</i>	CI
Run Integration Tests	<i>Human, Hit or Miss, Easily Bought Off with Pressure</i>	CI
Verify Dependency Security	<i>Human, Often Not Done</i>	CI
Deploy to Test Env	<i>Human, Problematic, Missed Steps</i>	CD
Team Test	<i>Human, Time Consuming</i>	CD - Automated Acceptance Tests
Deploy to Client Test Env	<i>Human, Problematic, Missed Steps</i>	CD

How do we know we need CI/CD ?



- Investing **more time** in a release cycle than delivering value
- Going through integration hell every time we finish a feature
- **Code gets lost** because of botched merges
- Unit test suite hasn't been green in ages
- Deployments contribute to **schedule slip**
- Friction between ops and development departments
- **Only one engineer** can deploy a system
- *Deployments are not cause for celebration*

Benefits

Technical Language	Value	Translation
Catch Compile Errors After Merge	Reduce Cost	Less developer time on issues from new developer code
Catch Unit Test Failures	Avoid Cost	Less bugs in production and less time in testing
Detect Security Vulnerabilities	Avoid Cost	Prevent embarrassing or costly security holes
Automate Infrastructure Creation	Avoid Cost	Less human error, Faster deployments
Automate Infrastructure Cleanup	Reduce Cost	Less infrastructure costs from unused resources
Faster and More Frequent Production Deployments	Increase Revenue	New value-generating features released more quickly
Deploy to Production Without Manual Checks	Increase Revenue	Less time to market
Automated Smoke Tests	Protect Revenue	Reduced downtime from a deploy-related crash or major bug
Automated Rollback Triggered by Job Failure	Protect Revenue	Quick undo to return production to working state