The Future of Feature Flags and Continuous Delivery

Shifting to Feature First Organizations

Erez Rusovsky,
Director of Product Management

DEVOPS WORLD
by CloudBees

Continuous Delivery

Code is always in a deployable state, even in the face of teams of thousands of developers making changes on a daily basis

↓

Continuous Deployment

Every change goes through the pipeline and automatically gets put into production

DEVOPSWORLD
by CloudBees

Where Can Continuous Deployment Fall Short?

- ## HARD!!

- But even when achieved:

  - Testing primarily occurs pre-production without "real world" factors

  - Deployments and releases are conjoined as one step

  - Limited ability to roll out features to specific customer subsets

  - Some customers may not want continuous updates

DEVOPSWORLD
by CloudBees

Traditional CD vs. CD with Feature Flags

Traditional CD

- Hard

- Emphasis on intensive pre-production testing

- Users get different versions

- "All or nothing" rollout or rollback of a release based on feature dependencies and performance

- Business teams and developer must work in tandem to deploy a release to production at the ideal time
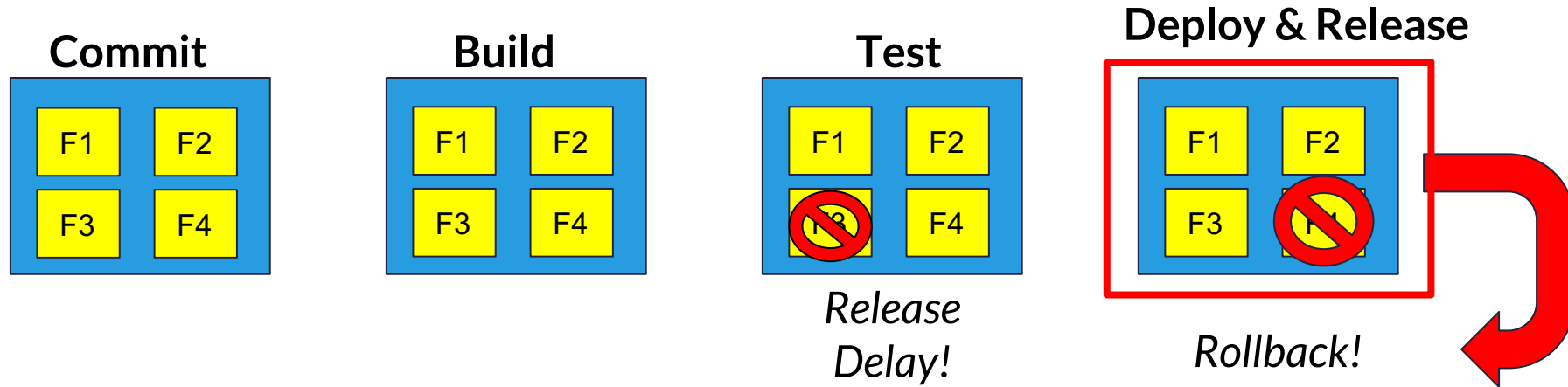
CD + Feature Flags

- Easier reaching CD

- Greater shift to testing in production

- Everyone gets the same version but different features

- Independent feature releases can be rolled out or back without impact the others

- Deploying code to production while the business (product or executives) decide on when to release the feature to customers decouples teams and increases productivity

DEVOPSWORLD
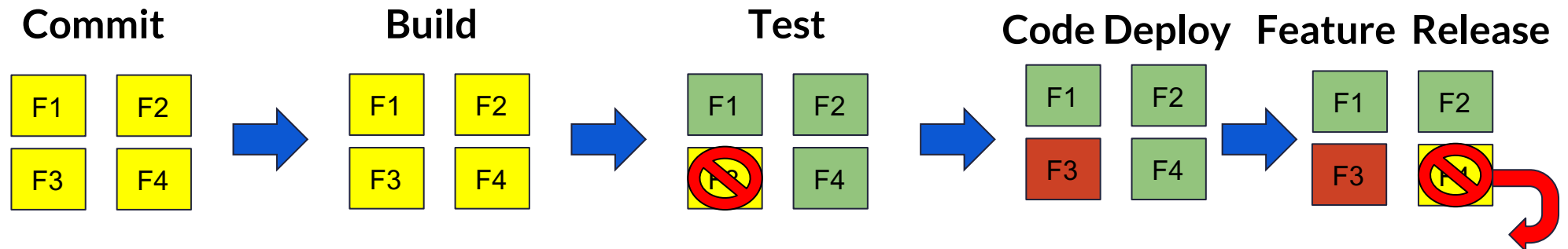by CloudBees

Full Release vs. Feature Granularity

- Shifts organizations from product (full release) to feature focus and mindset
- Features are no longer bound by the release as a whole, changing the game

Traditional CI/CD

Release v1.1

Commit

F1	F2
F3	F4

Build

F1	F2
F3	F4

Test

F1	F2
🚫	F4

Release Delay!

Deploy & Release

F1	F2
F3	🚫

Rollback!

CI/CD with Feature Flags

Commit

F1	F2
F3	F4

Build

F1	F2
F3	F4

Test

F1	F2
🚫	F4

Code Deploy

F1	F2
F3	F4

Feature Release

F1	F2
F3	🚫

CD + FF = CD++

- Feature granularity

- Release consistency

- Better, faster testing

- Reduced delivery risk

- Shared visibility and
governance across development and delivery teams

Feature Flag Maturity in CI/CD

Feature Flags	Enterprise FF Management	Enterprise Progressive Delivery	Enterprise Feature Lifecycle Management
Standalone development pattern used in silos	Using a standardized platform for scalability, visibility, governance, and advanced functionality	**Integrated CI/CD + FF = CI/CD++**	The ideal future state!
Standard targeted benefits in CI/CD		Shared view of flags across CI/CD tool sets across org	Fully shifted from product to feature focus
Either a limited "homegrown" management system or none at all	Generally where true FFDD starts given common visibility and governance	QA, Ops can see flags to pass or deploy	Organization aligned around visibility and impact of new features
		Increased emphasis on feature vs. product release	

Feature Flag-Driven Development

Thank You

DEVOPS WORLD
by CloudBees