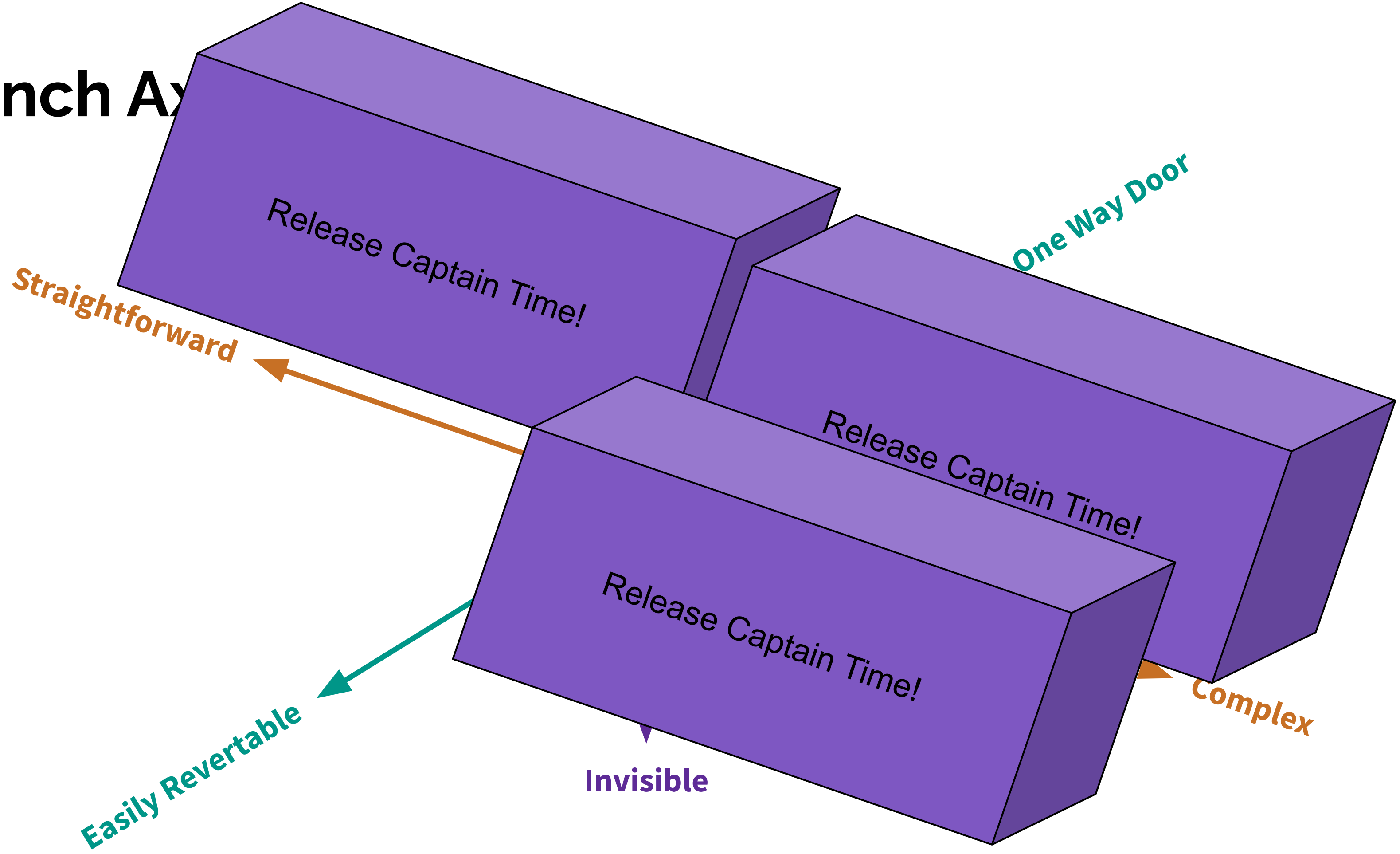
Supporting Major Launches as a Staff+ IC

Erin Sardo

she/her/hers

Launch Ax



Release Captain Time!

Release Captain Time!

Release Captain Time!

One Way Door

Straightforward

Complex

Easily Revertable

Invisible

What is a Release Captain?

- Router, Ring Leader, DRI, Single Source of Truth, etc for Engineering
- Plan and oversee engineering tasks supporting the rollout
- Unifying force across all sub-features

Release Captain Responsibilities

- Deriving required internal engineering milestones for release safety
- Environments for testing
- Rollout strategy
- Rollback strategy

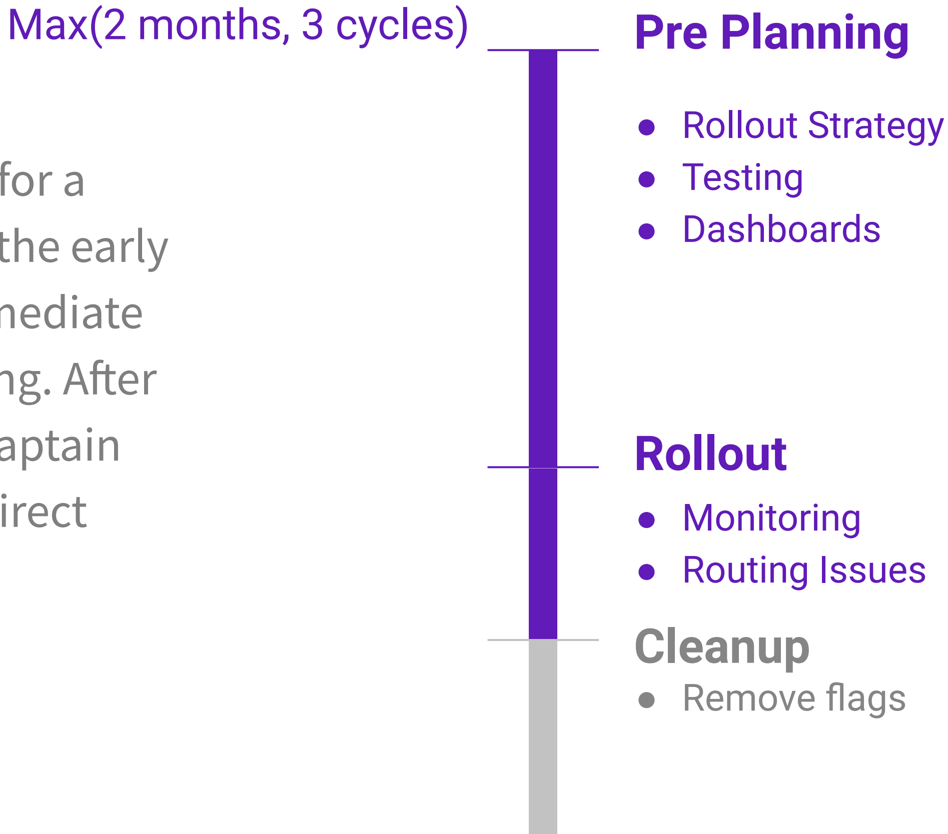
Other Tasks for Successful Launch

- QA
- Bug Triage
- Press/External Communications
- ✳ Internal Milestone communications
- ✳ Itemized launch agenda coordination
- Go/no-go decision tracking

✳ Can be taken by Release Captain in lieu of Program Manager

Timeline

The bulk of the work for a Release Captain is in the early planning and the immediate pre-launch fire-fighting. After a launch, a Release Captain should be able to redirect most issues to the appropriate DRI.



Max(2 months, 3 cycles)

Pre Planning

- Rollout Strategy
- Testing
- Dashboards

Rollout

- Monitoring
- Routing Issues

Cleanup

- Remove flags

Thoughts for Release Captains

Delegate, Aggregate, Communicate!

- See the puzzle, not just the pieces. Know where everything fits together.
- Frameworks, not implementations
- Frequent, predictable communication

Flags - Questions

- Consistency factors
  - Does the same user need the same experience across each platform at the exact same time?
  - Do all the users in an organization/cohort need the same experience at the same time?
- Customer ordering
  - Is there an ordering across cohorts (consumer vs enterprise, different paid tiers, etc)?
  - Do any customers require an advanced notice period?
  - Can customers opt-out of their predetermined rollout time? Opt-in to an earlier one?
- How quickly should each cohort get the features? Is there a percentage ramp?
- How will rollbacks work?
- Is a minimum version of the code required in production? How will this be enforced?

Strategies for Multiple Flags

- Consolidate
  - `s/feature_flag_foo/aggregated_feature_flag/g`
- Pre-launch dependent functionality
  - `aggregated_feature_flag` controls the ability to get into <Brand New Screen>. Within <Brand New Screen>, `feature_flag_foo` controls the Foo button. It is safe to roll out `feature_flag_foo` to 100% before `aggregated_feature_flag`
- Tie flags together
  - Some systems allow `feature_flag_foo` to be set to the exact state of `aggregated_feature_flag` automatically

Testing Environments

- Under what conditions should each feature be tested manually? In automated tests?
- How will you handle if Feature Foo drops out of the release but Feature Bar was tested with it?

Internal Engineering Milestones

- Feature decisions
    - Required minimum feature set that blocks launch
    - Final set for launch including non-blocking but desired items
- Flag consolidation/rollout completion
- Code complete/no additional features
- Launch/project-health dashboard completion and verification

Picking a Release Captain

Picking a Release Captain

- Have the time: ≥50% time commitment
- Enjoy working with others
- Can communicate clearly
- Familiarity with launching features

Consider a Shadow opportunity for more junior candidates

Thank you!