Boulders, Rocks, and Pebbles

Executing On a Technology Strategy

Randy Shoup
@randyshoup

Executing Technology Strategy

- Make It Measurable
- Make It Concrete
- Make It Happen
- Make It Better

Executing Technology Strategy

- Make It Measurable
- Make It Concrete
- Make It Happen
- Make It Better

Velocity Initiative

Velocity Initiative: Accelerate Metrics

Aspect of Software Delivery Performance*	Elite	High	Medium	Low
Deployment frequency For the primary application or service you work on, how often does your organization deploy code to production or release it to end users?	On-demand (multiple deploys per day)	Between once per day and once per week	Between once per week and once per month	Between once per month and once every six months
Lead time for changes For the primary application or service you work on, what is your lead time for changes (i.e., how long does it take to go from code committed to code successfully running in production)?	Less than one day	Between one day and one week	Between one week and one month	Between one month and six months
Time to restore service For the primary application or service you work on, how long does it generally take to restore service when a service incident or a defect that impacts users occurs (e.g., unplanned outage or service impairment)?	Less than one hour	Less than one day[a]	Less than one day[a]	Between one week and one month
Change failure rate For the primary application or service you work on, what percentage of changes to production or released to users result in degraded service (e.g., lead to service impairment or service outage) and subsequently require remediation (e.g., require a hotfix, rollback, fix forward, patch)?	0-15%[b,c]	0-15%[b,d]	0-15%[c,d]	46-60%

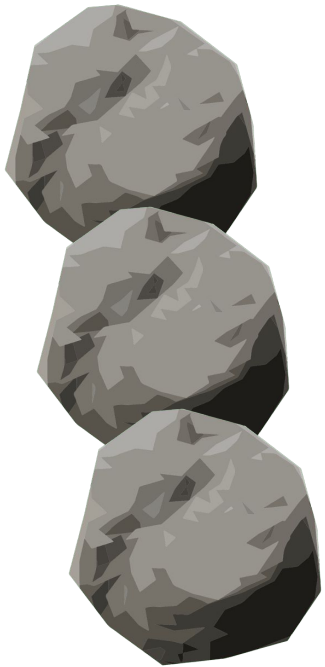
Velocity Initiative: Input Metrics

- Mobile App Release Frequency
  - Monthly -> biweekly -> weekly
- Server Build Time
  - 100min -> 5min at p95 [4.4min]
- Server Startup Time
  - 6min -> 1min at p95 [3min]
- PR Validation Time
  - 132min -> 60min [25min]
- Staging Availability
  - 60% -> 99% [97%]
- Traffic Mirror Adoption
  - 0 -> 100 apps [436]
- Canary Deployment Adoption
  - 0 -> 100 apps [~700]

Executing Technology Strategy

- Make It Measurable
- Make It Concrete
- Make It Happen
- Make It Better

Boulders, Rocks, and Pebbles




Google™

App Engine Outage - Oct 2012

- 4:00 am - Load begins increasing on traffic routers in one of the App Engine datacenters.
- 6:10 am - The load on traffic routers in the affected datacenter passes our paging threshold.
- 6:30 am - We begin a global restart of the traffic routers to address the load in the affected datacenter.
- 7:30 am - The global restart plus additional load unexpectedly reduces the count of healthy traffic routers below the minimum required for reliable operation. This causes overload in the remaining traffic routers, spreading to all App Engine datacenters. Applications begin consistently experiencing elevated error rates and latencies.
- 8:28 am - google-appengine-downtime-notify@googlegroups.com is updated with notification that we are aware of the incident and working to repair it.
- 11:10 am - We determine that App Engine's traffic routers are trapped in a cascading failure, and that we have no option other than to perform a full restart with gradual traffic ramp-up to return to service.
- 11:45 am - Traffic ramp-up completes, and App Engine returns to normal operation.

App Engine Outage - Oct 2012

In response to this incident, we have increased our traffic routing capacity and adjusted our configuration to reduce the possibility of another cascading failure. Multiple projects have been in progress to allow us to further scale our traffic routers, reducing the likelihood of cascading failures in the future.

During this incident, no application data was lost and application behavior was restored without any manual intervention by developers. There is no need to make any code or configuration changes to your applications.

We will proactively issue credits to all paid applications for ten percent of their usage for the month of October to cover any SLA violations. This will appear on applications' November bills. There is no need to take any action to receive this credit.

http://googleappengine.blogspot.com/2012/10/about-todays-app-engine-outage.html

App Engine Reliability Fixit

- Step 1: Identify the Problem
  - All team leads and senior engineers met in a room with a whiteboard
  - Enumerated all known and suspected reliability issues
  - Too much technical debt had accumulated
  - Reliability issues had not been prioritized
  - Identify 8-10 themes

App Engine Reliability Fixit

- Step 2: Understand the Problem
  - Each theme assigned to a senior engineer to investigate
  - Timeboxed for 1 week
  - After 1 week, all leads came back with
    - Detailed list of issues
    - Recommended steps to address them
    - Estimated order-of-magnitude of effort (1 day, 1 week, 1 month, etc.)

App Engine Reliability Fixit

- Step 3: Consensus and Prioritization
  - Leads discussed themes and prioritized work
  - Assigned engineers to tasks

App Engine Reliability Fixit

- Step 4: Implementation and Follow-up
  - Engineers worked on assigned tasks
  - Simple spreadsheet of task status, which engineers updated weekly
  - Minimal effort from management (~1 hour / week) to summarize progress at weekly team meeting

App Engine Reliability Fixit

- Results
  - 10x reduction in reliability issues
  - Improved team cohesion and camaraderie
  - Broader participation and ownership of the future health of the platform
  - Still remembered several years later

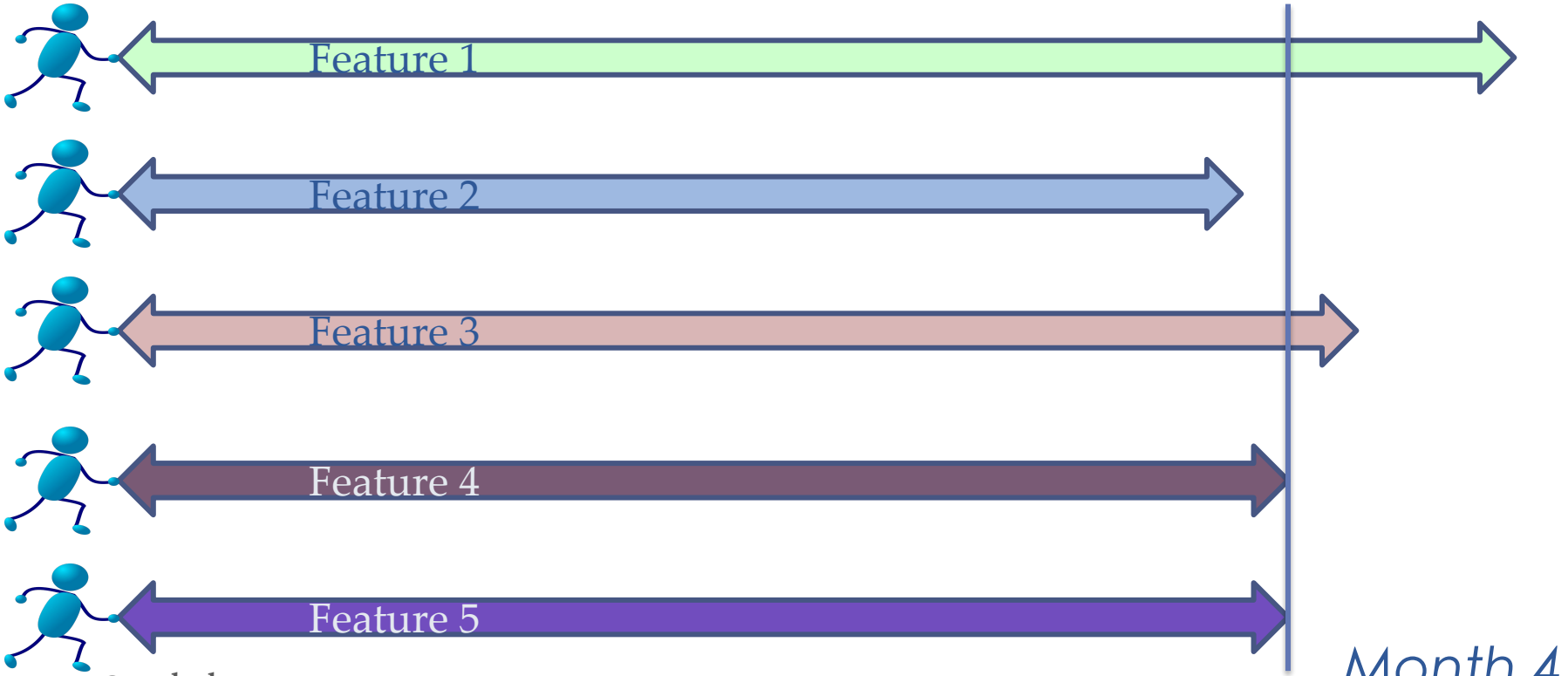
Executing Technology Strategy

- Make It Measurable
- Make It Concrete
- **Make It Happen**
- Make It Better

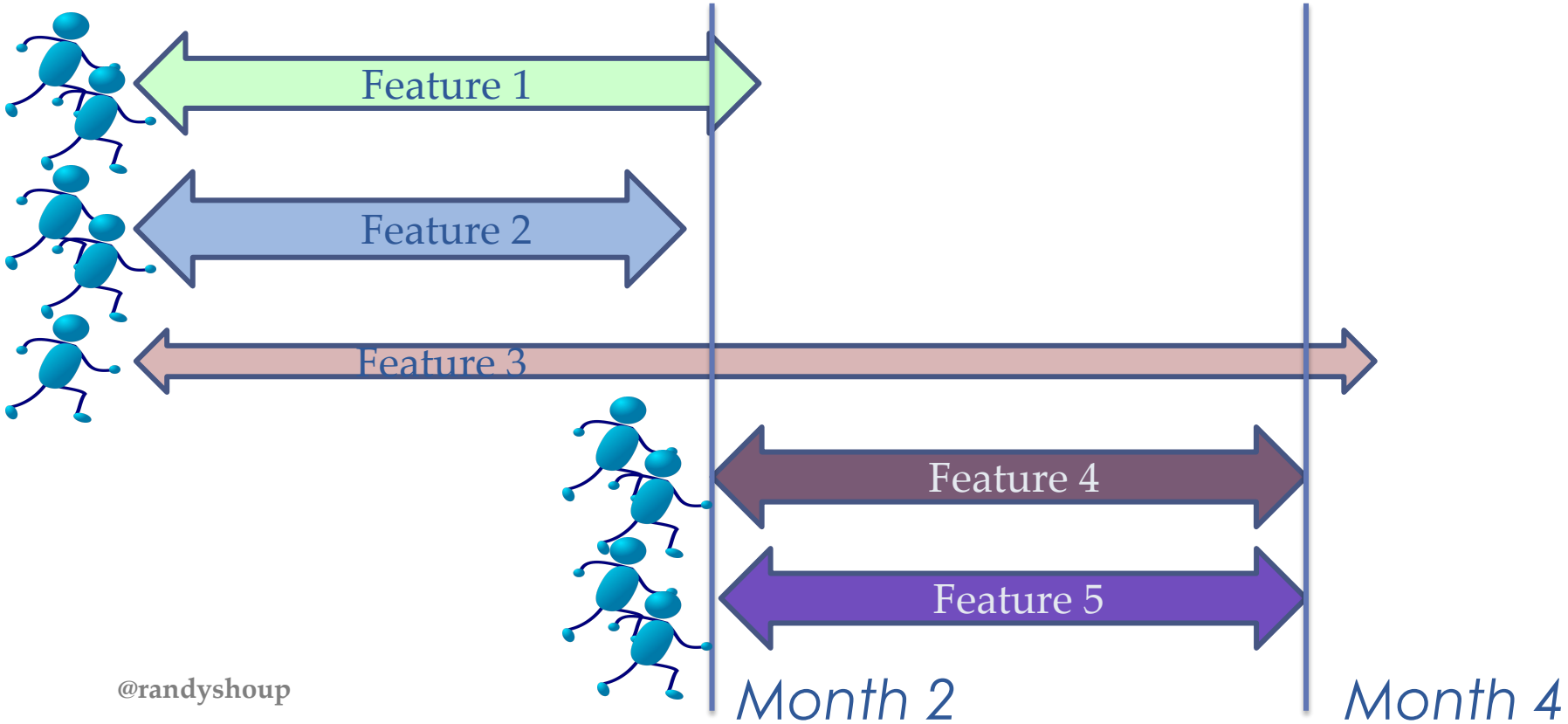
Continuous Delivery

- Show a consistent trajectory of incremental value
- Deliver on commitments
- When things don't go as planned …
  - Be Honest
  - Provide Options

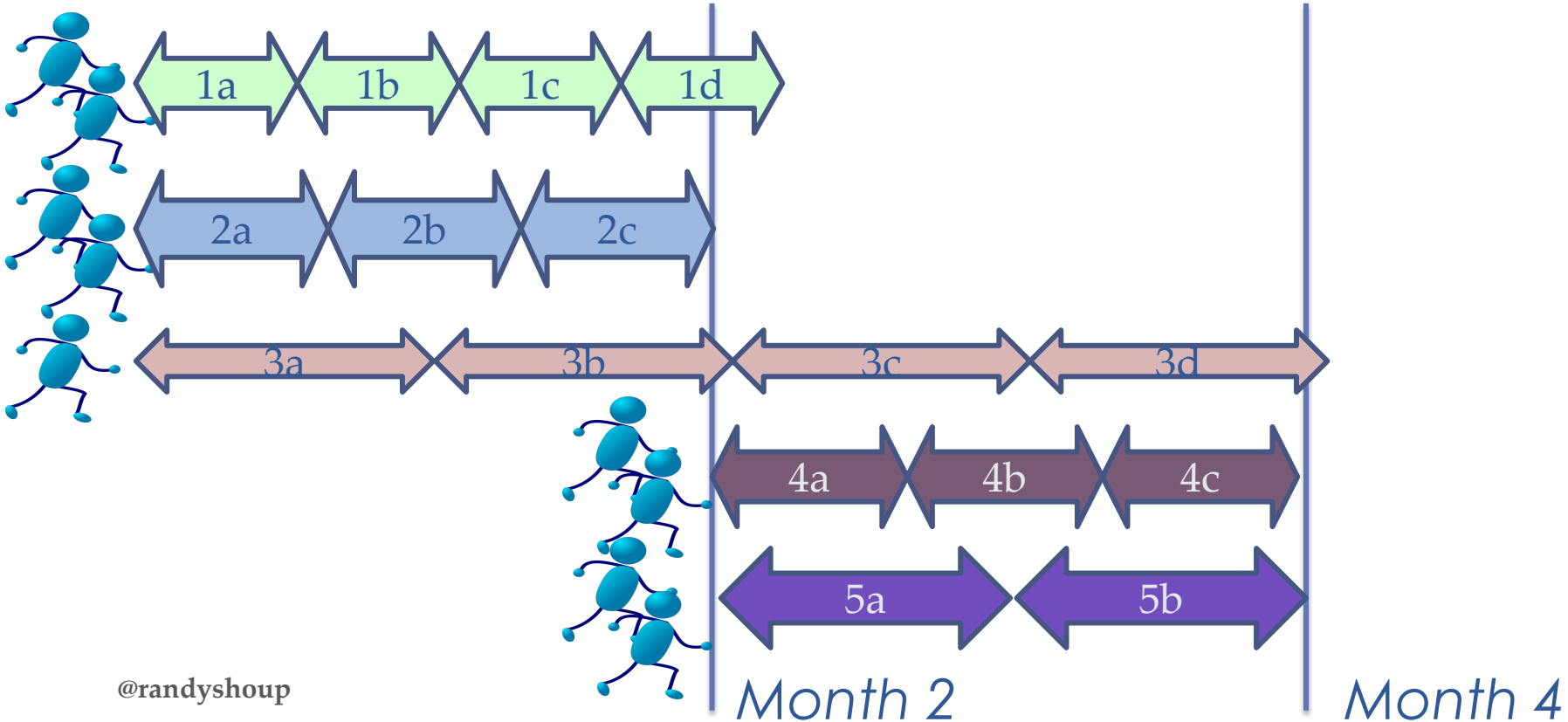
Too Much in Parallel



Fewer Things, More Done



Deliver Incremental Value

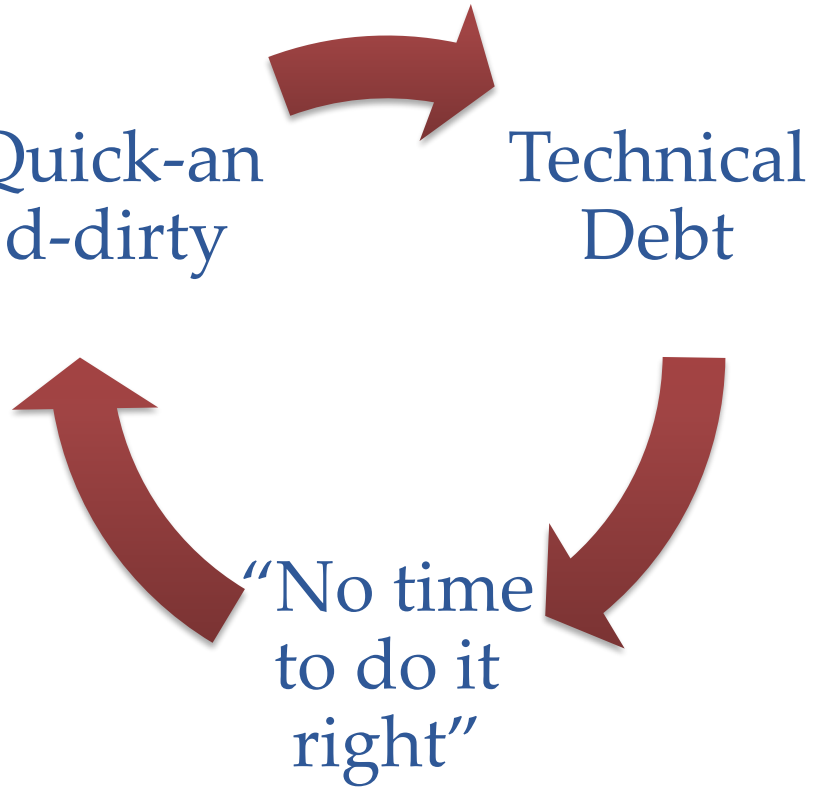



CHANGED
PRIORITIES
50 yds AHEAD
CCTV

Executing Technology Strategy

- Make It Measurable
- Make It Concrete
- Make It Happen
- Make It Better

Vicious Cycle of Technical Debt



Virtuous Cycle of Investment

Improving daily work is even more important than **doing daily work**.

-- Gene Kim, The DevOps Handbook

Executing Technology Strategy

- Make It Measurable
- Make It Concrete
- Make It Happen
- Make It Better

Thank you!

@randyshoup

linkedin.com/in/randyshoup

medium.com/@randyshoup