



Automated communication for improved deliveries

Lars Bendix
lars.bendix@cs.lth.se

*Department of Computer Science
Lund University
Sweden*

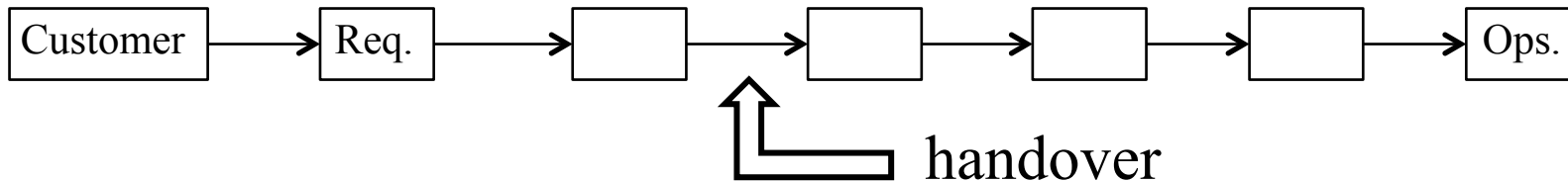
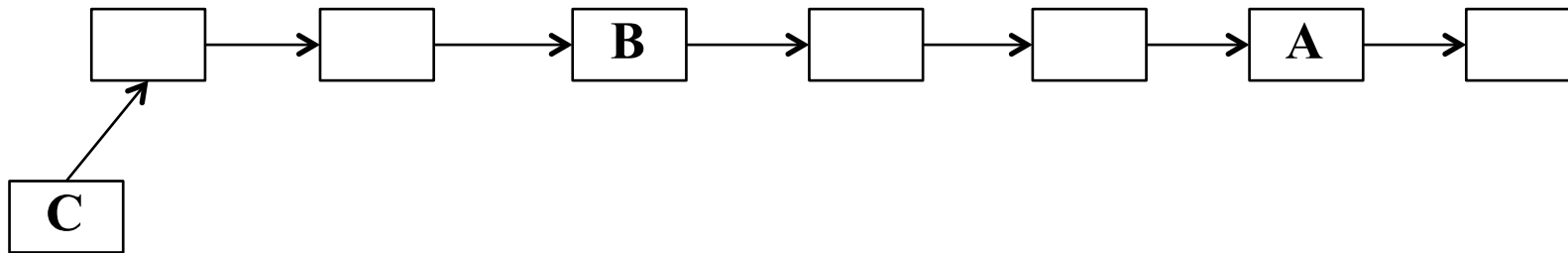
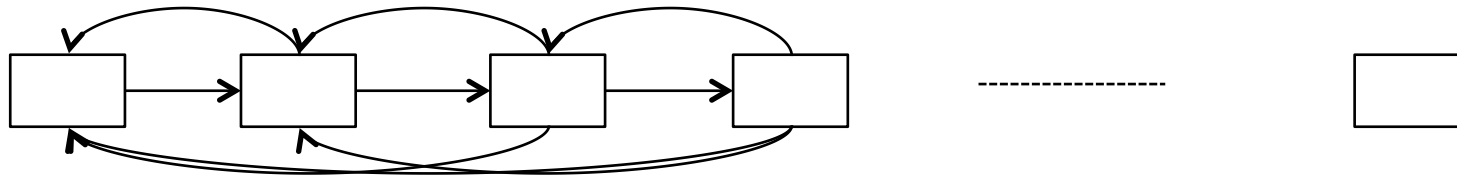


The DevOps dream





DevOps realities





Why do you want to do DevOps?

-
-
-

What can you learn from our students?

-
-



Context



Our students – since 2001:

- second year students
- groups of 8-12 developers
- two coaches (fourth year students)
- one customer (to share between three teams)
- agile development (eXtreme Programming)
- course part and project part



Setup



Six weekly iterations:

- 2-hour Planning Game on Wednesdays
- 4 hours of individual/pair work
- 8-hour programming session on Mondays
- release to customer every other week (+1)
- using all XP practices:
 - Frequent Releases (“DevOps”)
 - Continuous Integration
 - Test Driven Development
- something more



Real life

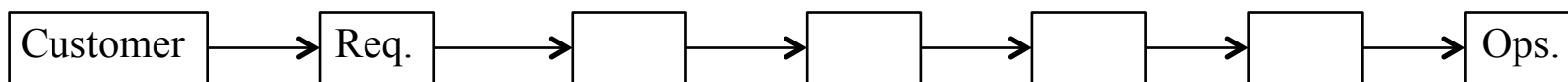
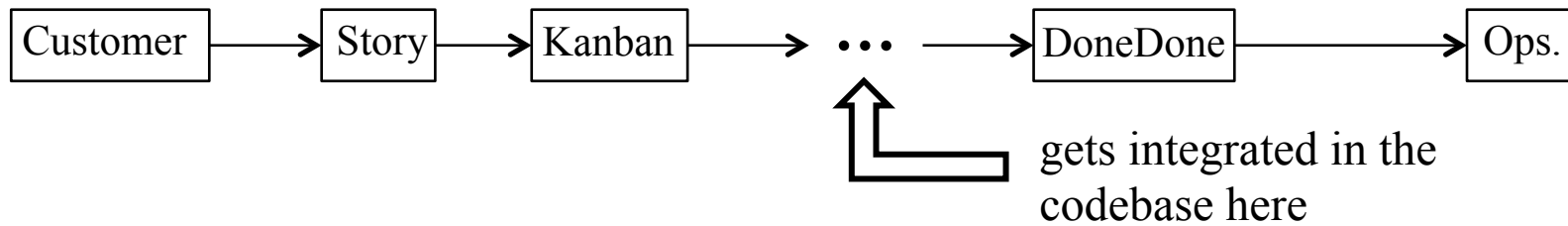


Can our students do:

- DevOps?
- Continuous Delivery?
- Frequent Releases?
- On-demand releases?
- Anything at all? ;-)



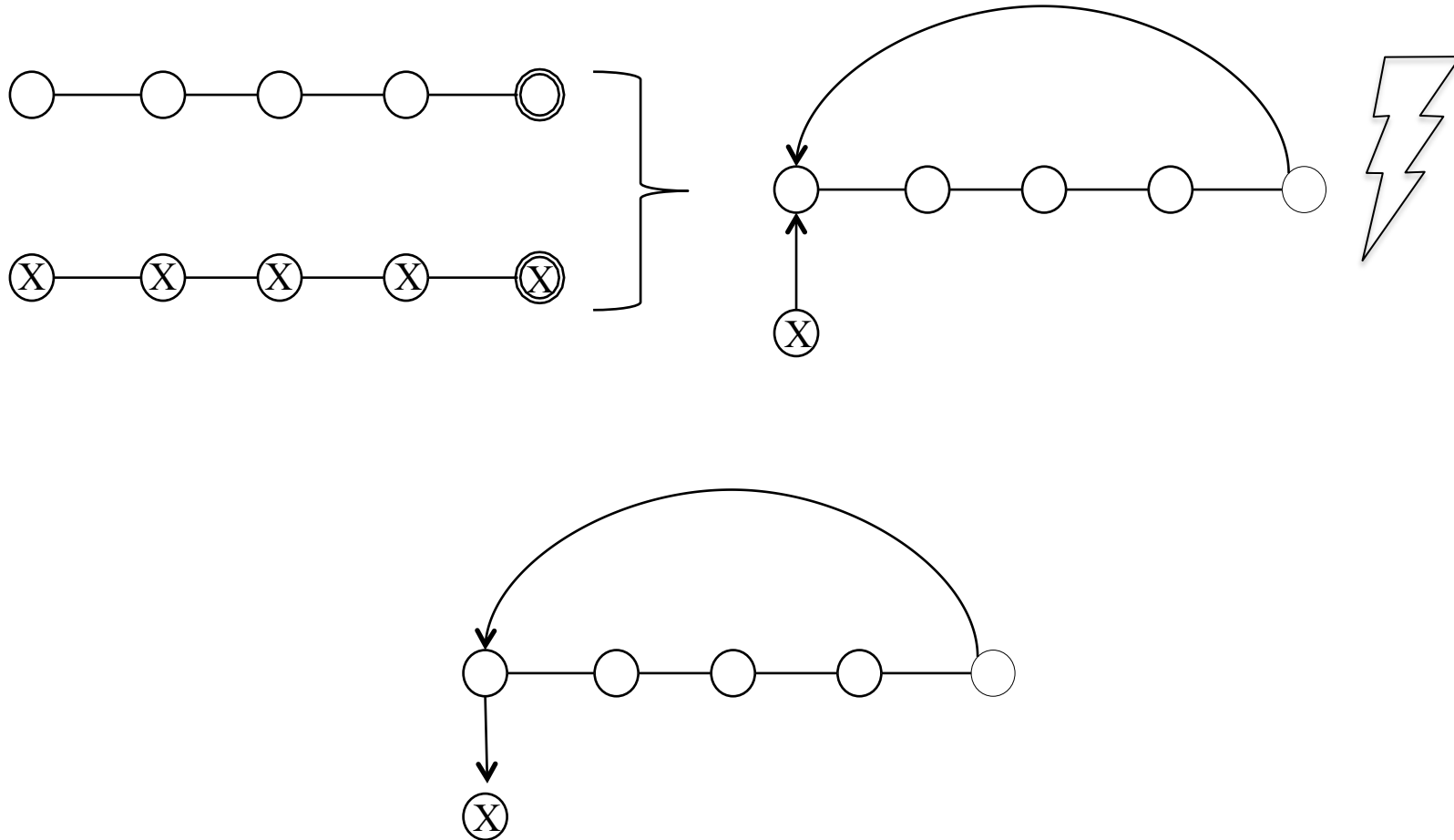
Bottlenecks and more



Done – DoneDone – ReallyDone - ...



Parallel work





Parallel strategies



Branching and merging

vs.

Change sets

vs.

Strict long transactions



Why look at DevDev?



High quality allows us to go from Dev to Ops in a nanosecond – and with probable success.

To create a good start for DevOps.



Situational awareness



Digital data means:

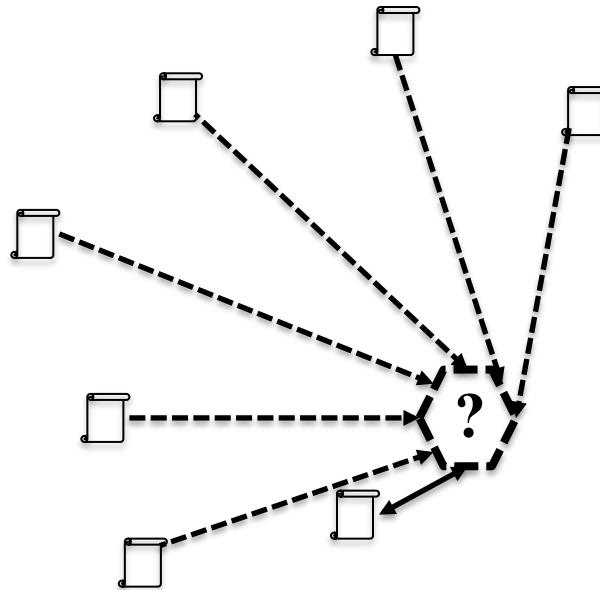
- copies and parallel work
- merge

Easing merge:

- locking
- (stable and traceable) APIs
- (decoupled) modularization
- strict long transactions
- (VCT) notification
- “virtual continuous integration”



Virtual Continuous Integration





What did you learn today?



Take-away #1: avoid bottlenecks – plan for scalability

Take-away #2: deal with parallelism – branching / change sets

Take-away #3: DevOps looks like TDD

Take-away #4: ...