

Contrasting Development & Release Stabilization on Linux Kernel

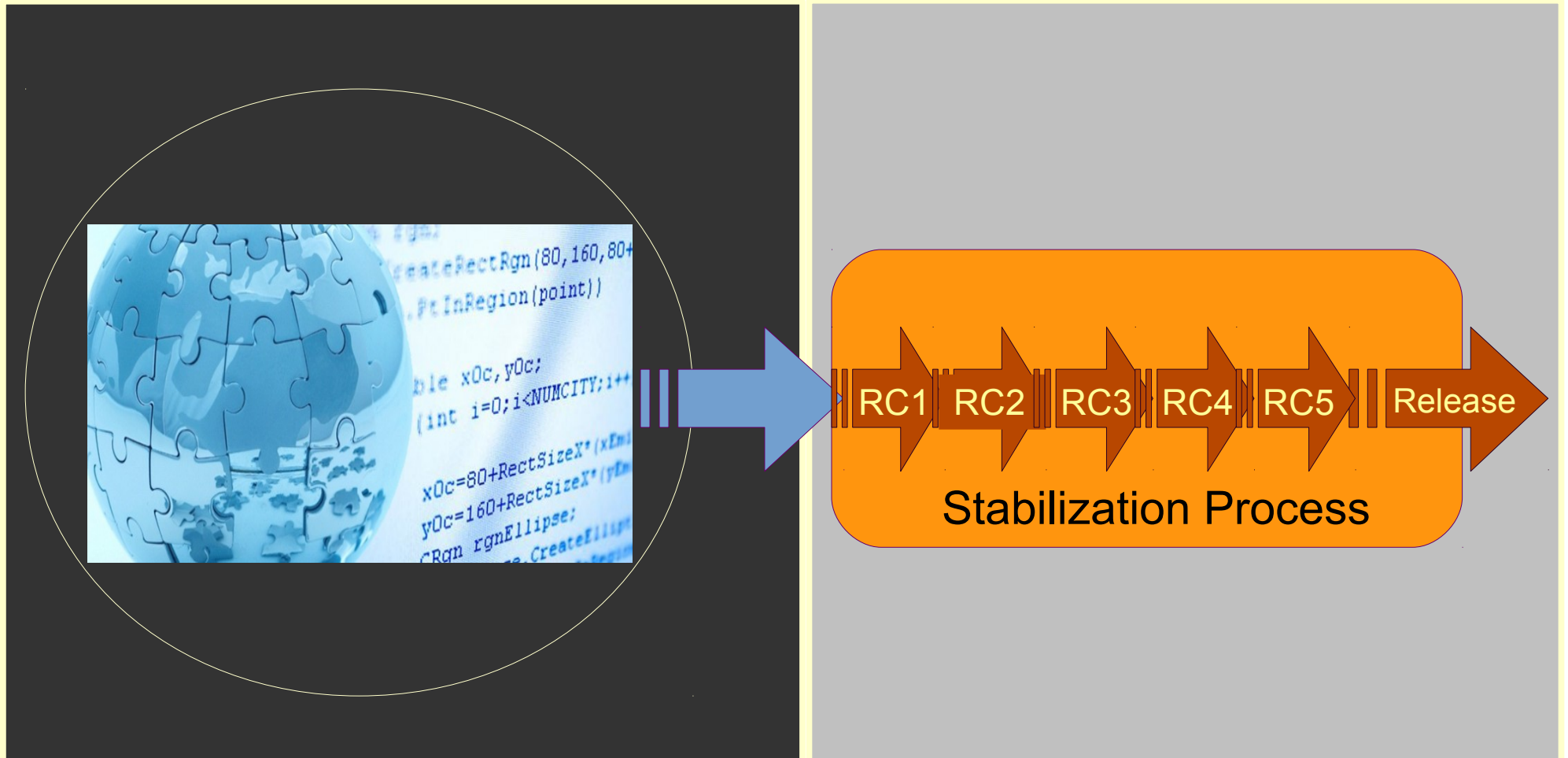
Md Tajmilur Rahman
Peter C Rigby

Concordia University
Montreal, Quebec, Canada

mdt_rahm@encs.concordia.ca
peter.rigby@concordia.ca



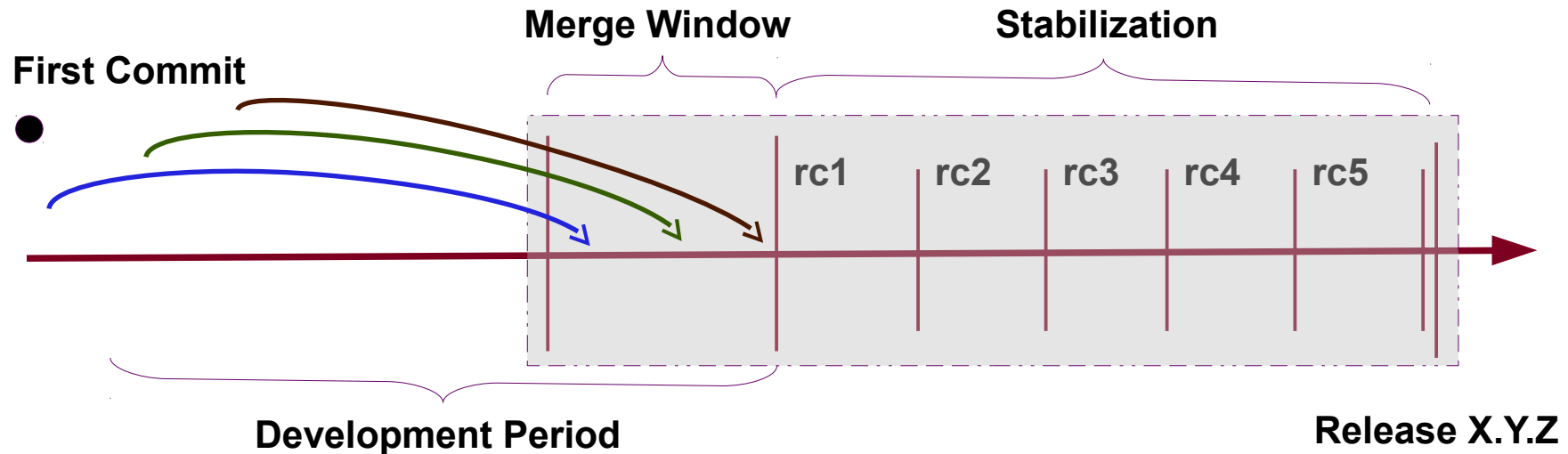
Development vs Stabilization



Research Questions

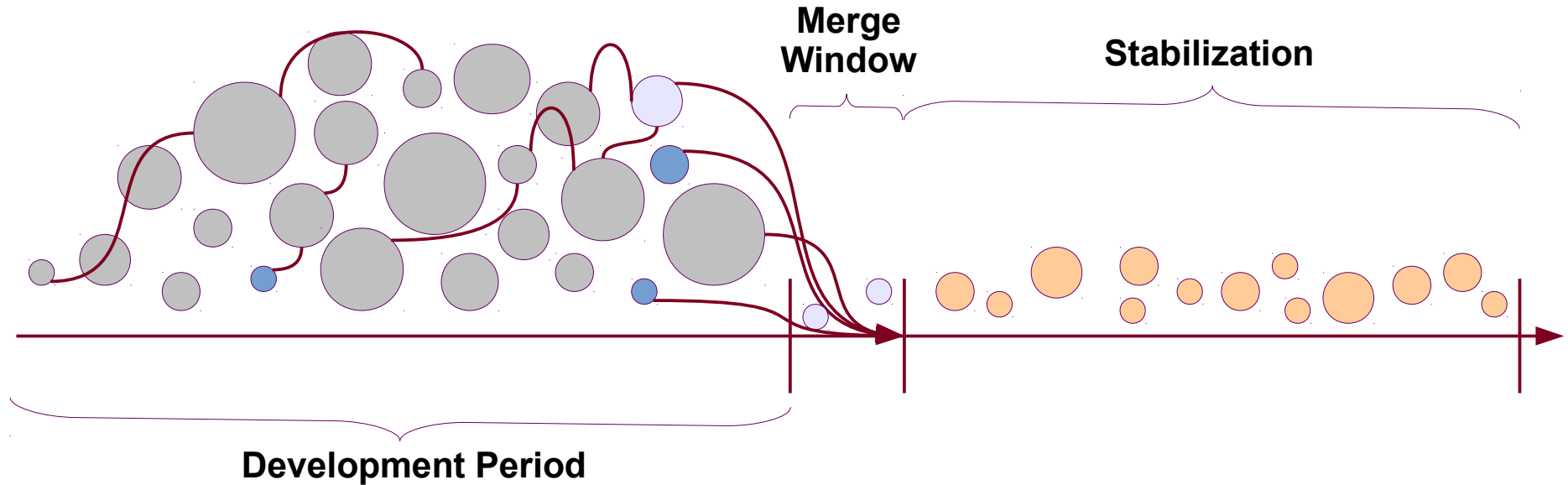
- **RQ1:** What **release process** does Linux use?
- **RQ2:** How much **effort** is expended during development vs stabilization?
- **RQ3:** Do **developers work** in the same set of files during development and stabilization?
- **RQ4:** How long does it take (**lag/transit time**) for development vs stabilization changes to be released?
- **RQ5:** Is there a **rush-to-release** period?

Results - RQ1: Release Process



- Rolling Release Model
- Has longer release cycle in comparison to Firefox and Google projects
- Has more branches to get merged in merge window

Results – RQ2: Effort Spent



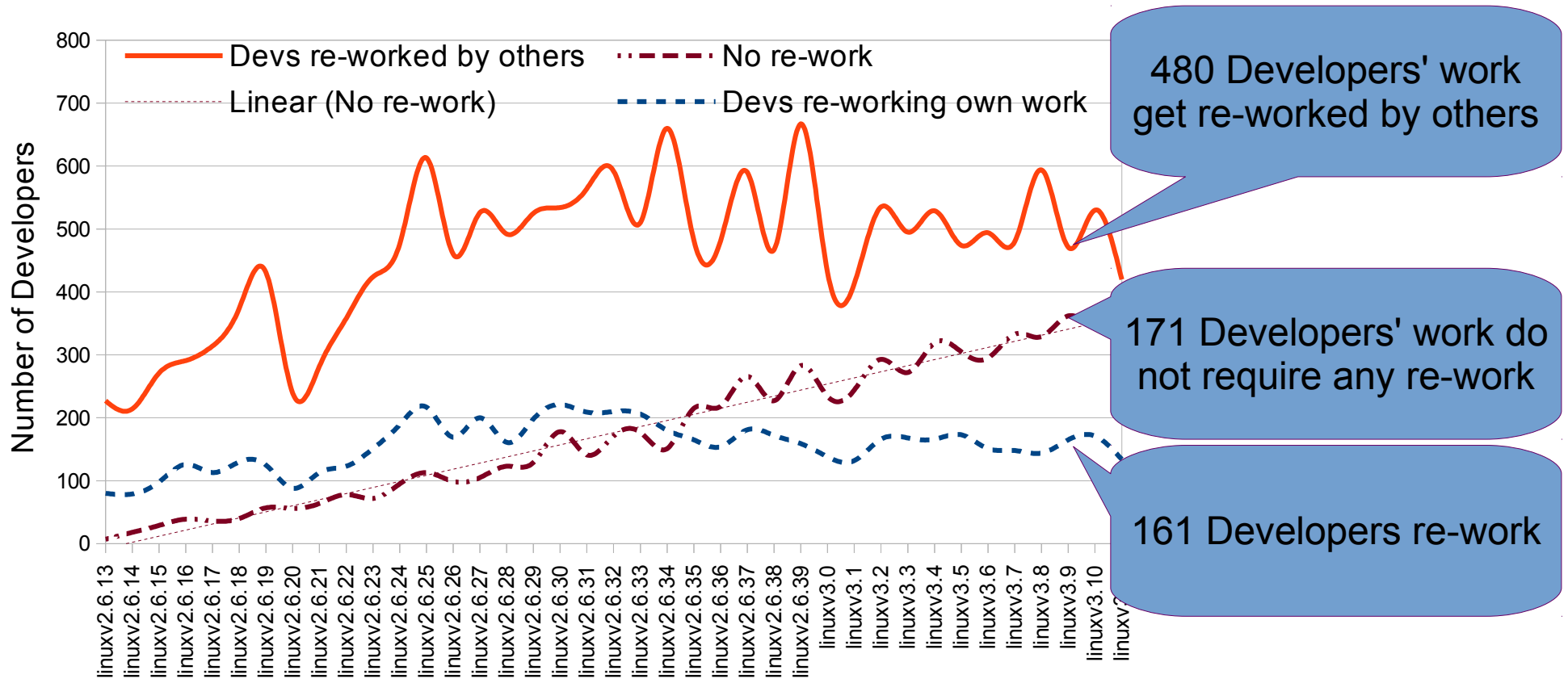
- Data: From 2005 to 2013, 331 releases, **39** are stable
- Commits: 381k (Development: **77%**, Stabilization: **23%**)
- Churns: Development: **91%**, Stabilization: **9%**
- LOC/Commit: Development: **105**, Stabilization: **41**

Results – RQ2: Core Developers

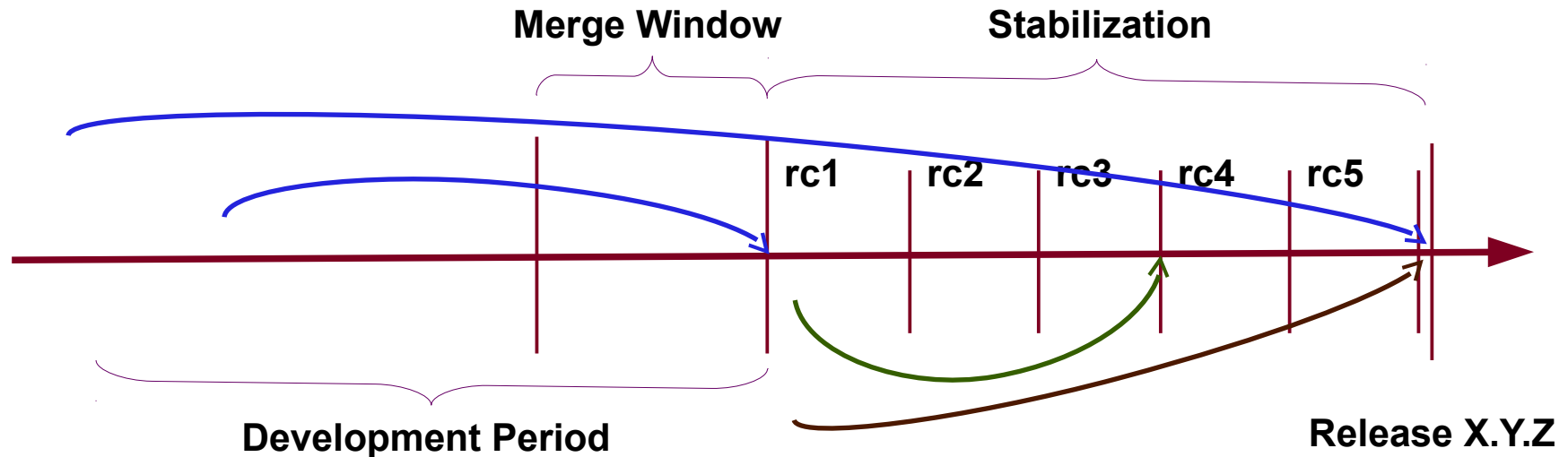


- 80% changes in development by: **55** devs
- 80% changes in stabilization by: **23** devs

Results – RQ3: Developers' work

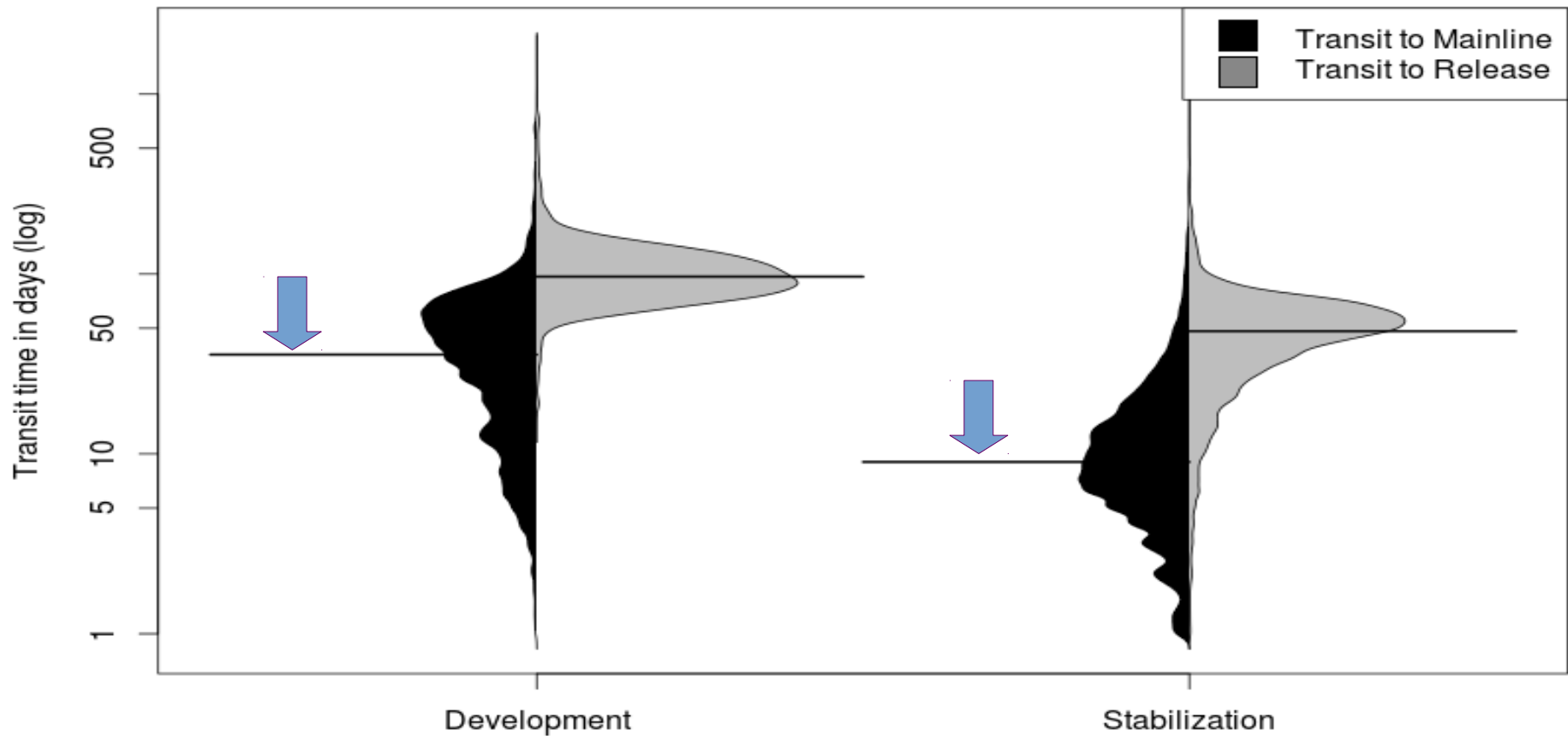


Results – RQ4: Transit Time



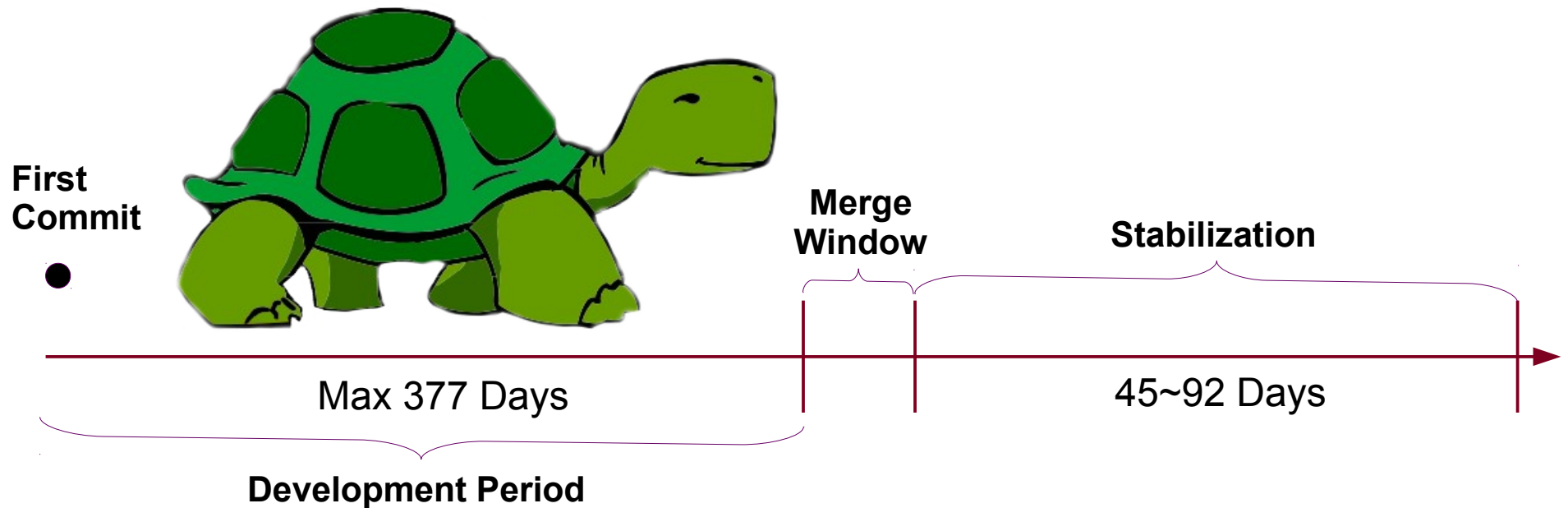
- Development Commit Date to Release Date
- Development Commit Date to rc1
- Bug Fix Commit Date to rc*
- Bug Fix Commit Date to Release

Results – RQ4: Transit time



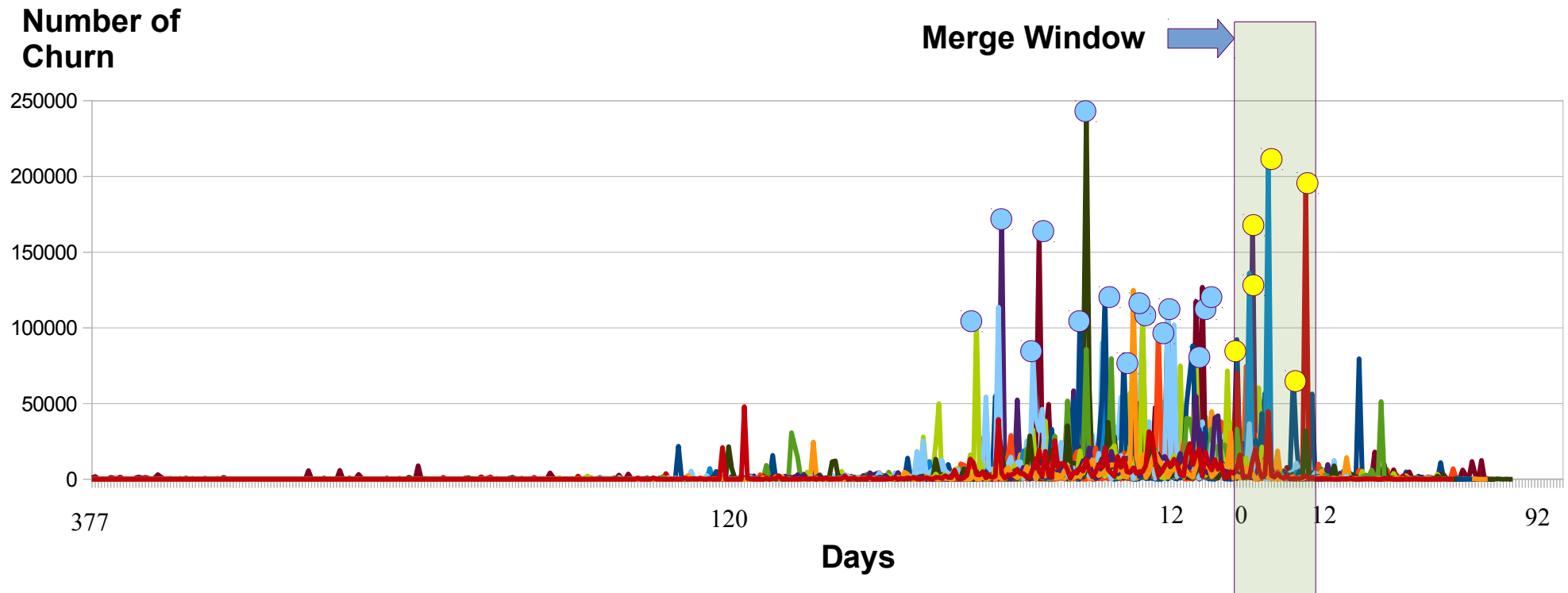
- Development commits: **35** days, Stabilization Commits: **8** days to be integrated to mainline

Results – RQ5: Rush to Release ??



Development seems to be slow and steady, although long distance between merge windows

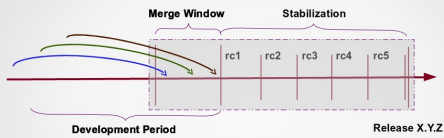
Results – RQ5: Rush to Release



- Kolmogrov-Smirnov test
- p-value: **0.18** indicates there is no significant difference between churn in merge window and before merge window



Results - RQ1: Release Process

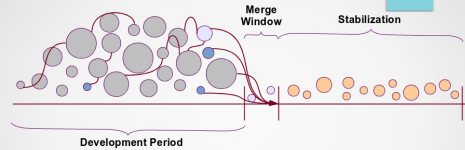


- Rolling Release Model
- Has longer release cycle in comparison to Firefox and Google projects
- Has more branches to get merged in merge window

RELENG 2014

4

Results – RQ2: Effort Spent



- Commits: 381k (Development: 77%, Stabilization: 23%)
- Churns: Development: 91%, Stabilization: 9%
- LOC/Commit: Development: 105, Stabilization: 41

RELENG 2014

5

Results – RQ2: Core Developers



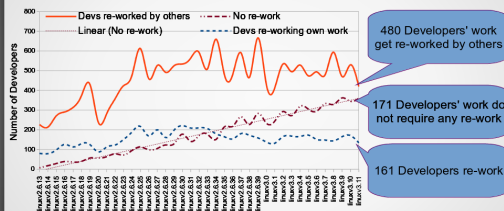
- Total developers: 1156 (Development: 1023, Stabilization: 590) per release cycle
- Common developers: 322

RELENG 2014

6

Question ?

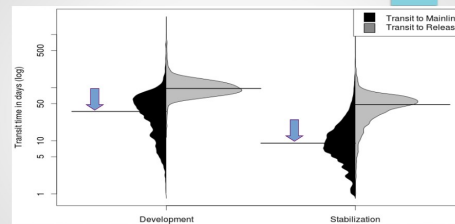
Results – RQ3: Developers' work



RELENG 2014

8

Results – RQ4: Lag

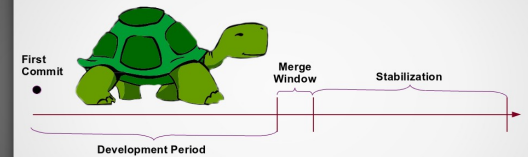


- Development commits: 35 days, Stabilization Commits: 8 days to be integrated to mainline

RELENG 2014

9

Results – RQ5: Rush to Release ??



Development seems to be slow and steady, although long distance between merge windows

RELENG 2014

10

Md Tajmilur Rahman
mdt_rahm@encs.concordia.ca

RELENG 2014

Peter C. Rigby
peter.rigby@concordia.ca

13