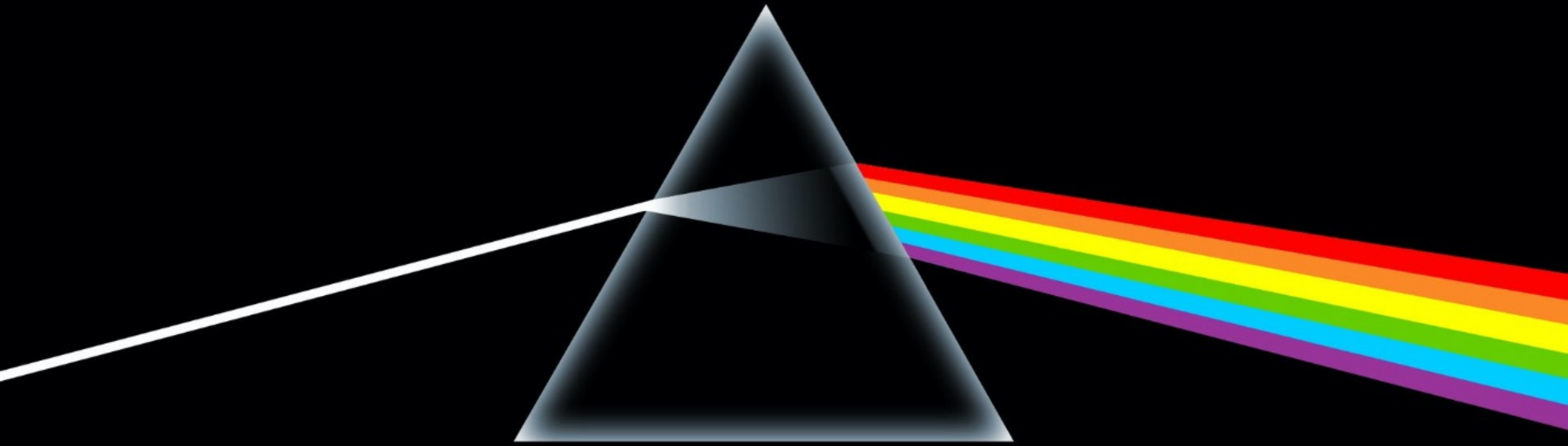
The uncharted side of your data

Important things you can learn from it

Javier Gonzalez Huerta
Associate Professor
Software Engineering Department, Blekinge Tekniska Högskola, Sweden

SERL Sweden
LEADING SOFTWARE ENGINEERING

February 16th, 2022



The uncharted side of your data

Important things you can learn from it

Javier Gonzalez Huerta
Associate Professor
Software Engineering Department, Blekinge Tekniska Högskola, Sweden

SERL Sweden
LEADING SOFTWARE ENGINEERING

February 16th, 2022

Who am I?

- Associate Professor in Software Engineering at Blekinge Institute of Technology
  - Program Manager: MSc of Engineering is Software Development (5 years program)

- Software Engineer: studied Computer Science and Software Engineering in Valencian Polytechnic University in Spain
- Worked as a developer, Project manager (not only in software projects), and Project leader during more than 10 years

- PhD in Software Development (Software Engineering)
- PostDoc here at Latece working with Hafedh and Naouel Moha
- Now I'm a researcher / practitioner (not necessarily in that order)
- Collaborate (doing research) with several companies in Sweden:

Spotify®

ERICSSON

VOLVO

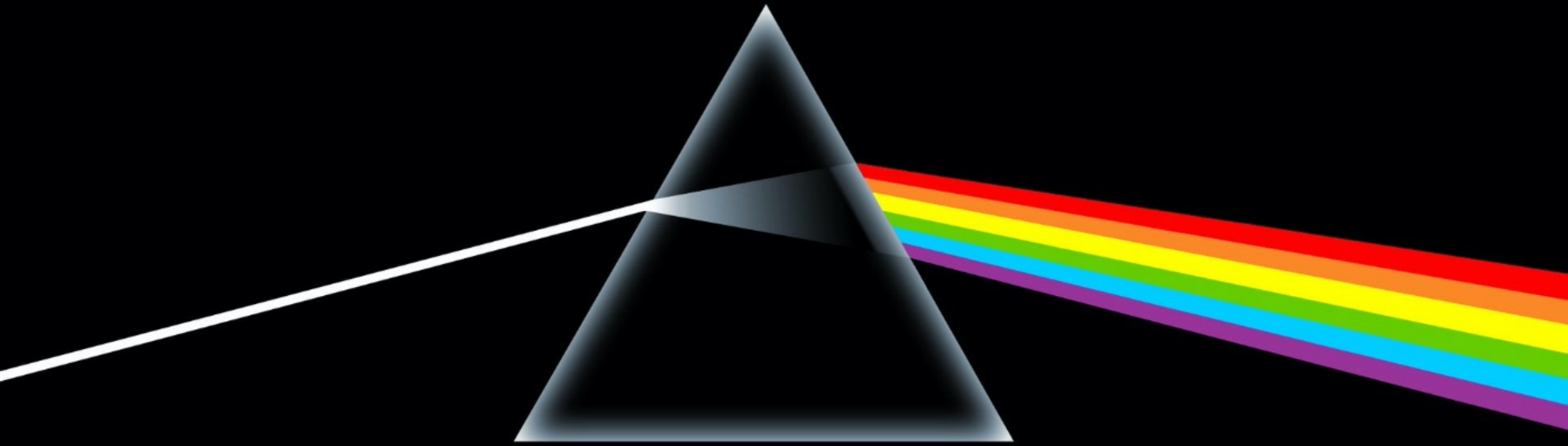
FORTNOX
Småföretagens bästa vän

Two words about BTH




- BTH is a small university in southern Sweden
- 21 Programs
- However we are a strong research environment in Software Engineering
- 7th most active institution in the world and 1st in Europe in Software Engineering Research [1]

[1] Wong, W. E., Mittas, N., Arvanitou, E. M., Li, Y. (2021). A bibliometric assessment of software engineering themes, scholars and institutions (2013–2020). Journal of Systems and Software, 180. https://doi.org/10.1016/j.jss.2021.111029



The uncharted side of your data

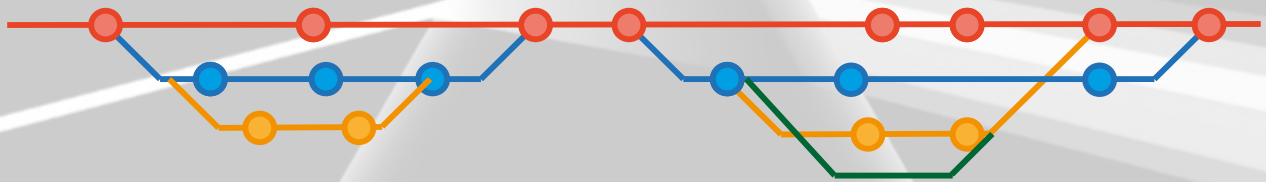
Important things you can learn from it

Can we learn more from data we already have?

- Technology allows organizations collecting vast amounts of data from the product and from their processes
- Organizations are now *hungry for their data* to improve their product and processes
- We are in the days of *Big Data,* but I would rather prefer being in the days of *Better Data*
- Giving another angle to the data can allow us to learn new lessons

What Data? Let's see what these are first...

Version Control Data



- Version Control Systems allow to manage file changes over time
- Are used by Software Organizations to control the code they produce
- When developers “commit” their code then is integrated into the main repository (after a QA process)
- Version control data includes *authorship data files changed, size of the change…*

What Data? Let's see what these are first...

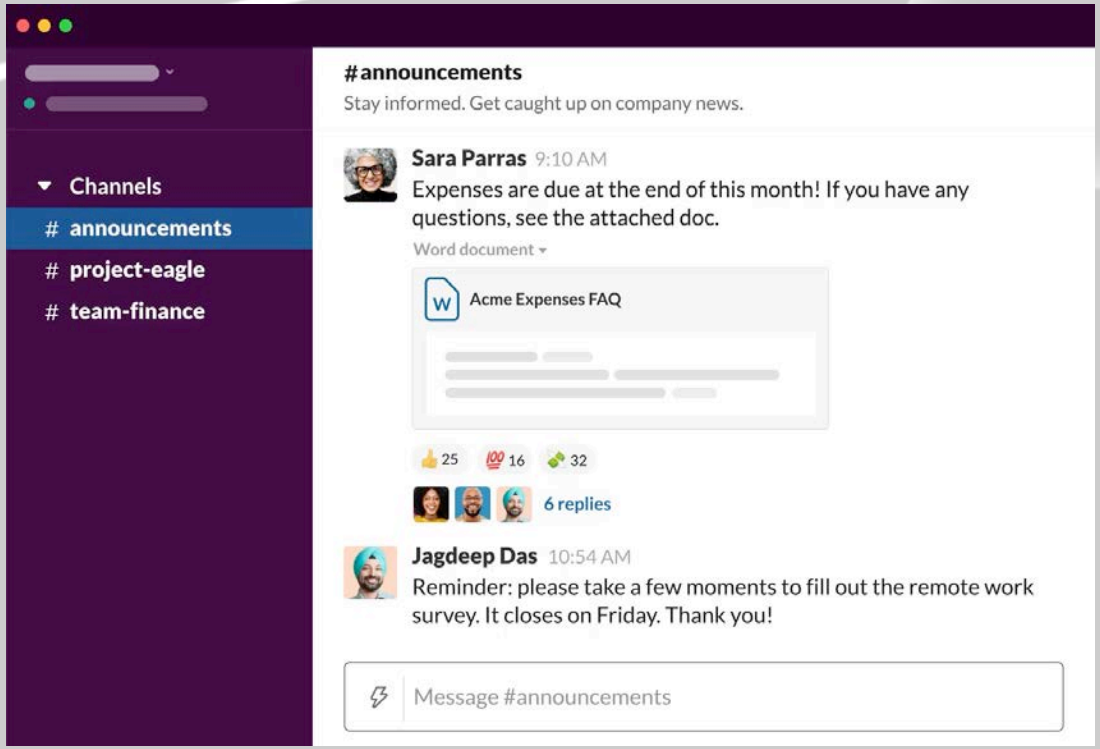
Code Review Data

Gerrit

- Code Review is a software quality assurance mechanism
- Is the act of inspecting your fellow programmers code to check for mistakes or improvements
- It can be supported by tools like
- Code review data include *who authors the code*, *who reviews*, *the verdict*, *comments*, *the size and files affected*,….

What Data? Let's see what these are first...

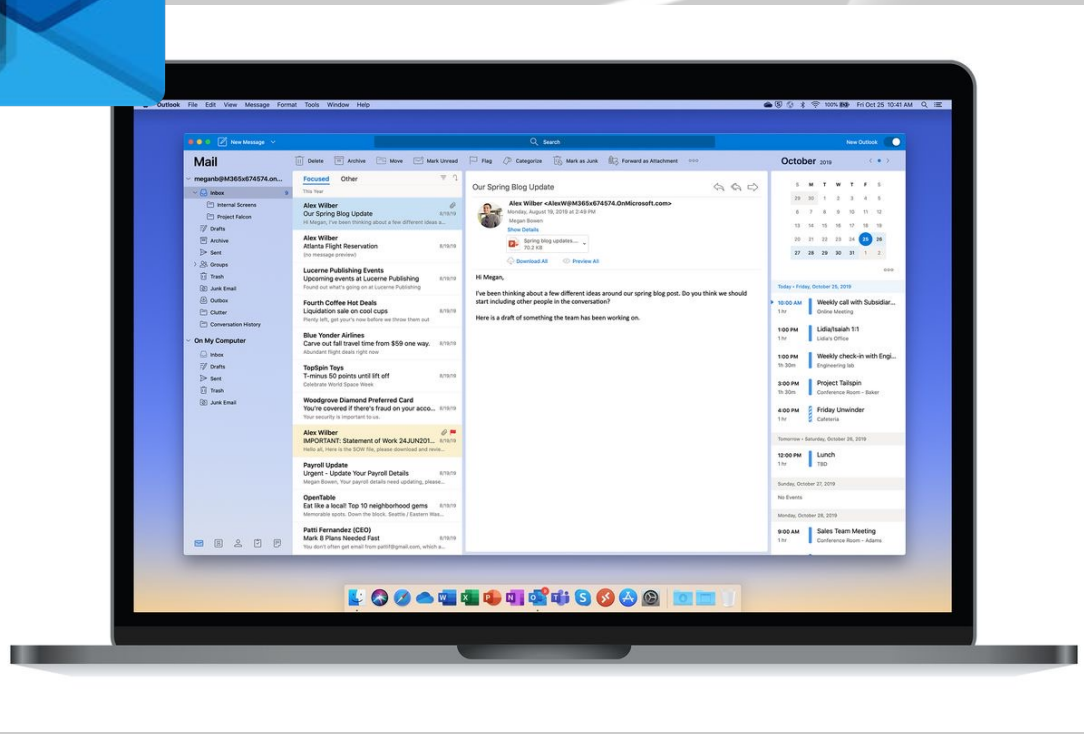
Slack as a communication tool



- Slack is a communication tool for professionals (like Teams or Skype)
- There are public and private channels people subscribe to
- Allows also point to point communications

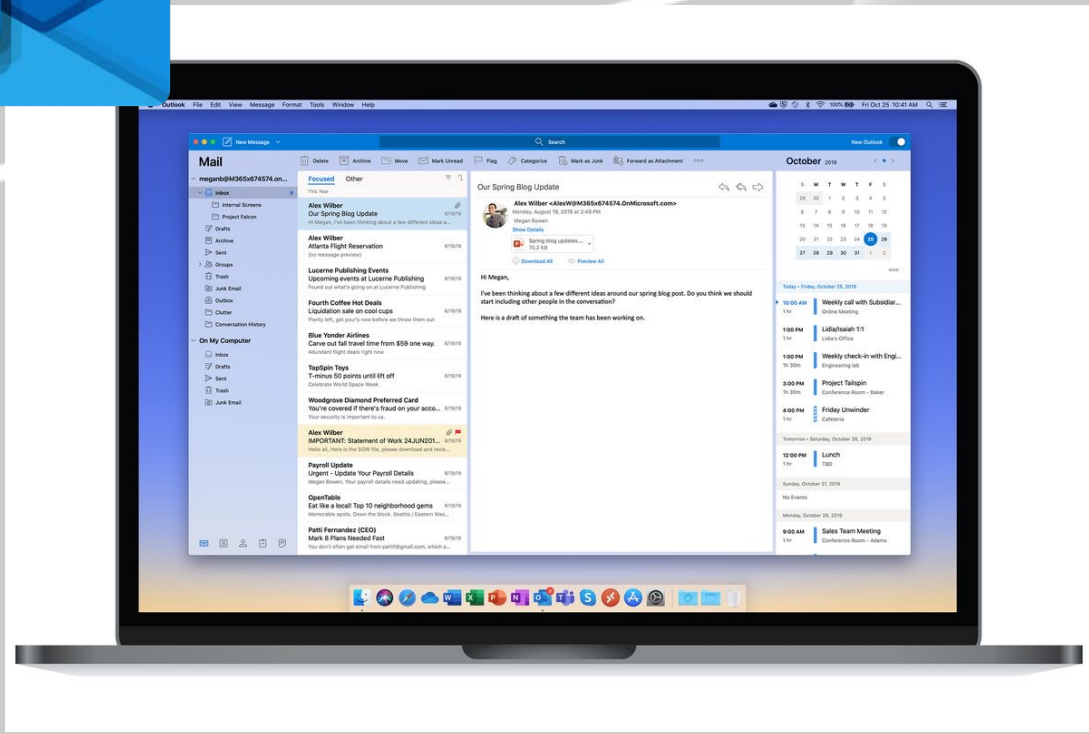
What Data? Let's see what these are first...

Meetings Invitations



- Meeting invitations are used nowadays to book meetings everywhere
- The information on a meeting includes: organized, meeting id, whether is recurring, whether I have accepted…

Meetings Invitations

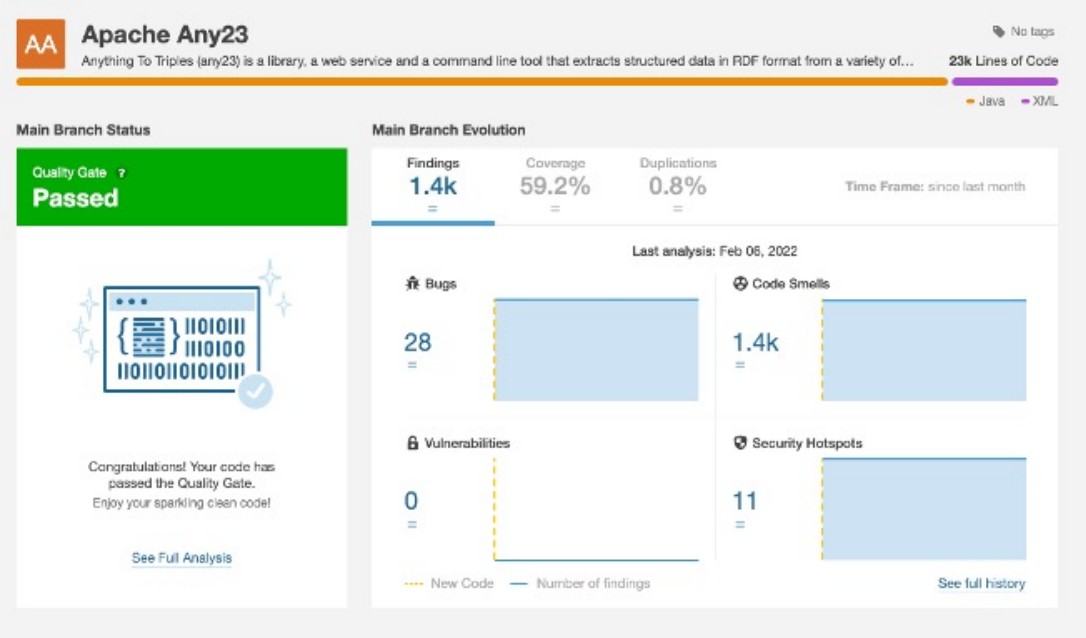


- Meeting invitations are used nowadays to book meetings everywhere
- The information on a meeting includes: organized, meeting id, whether is recurring, whether I have accepted…

What Data? Let's see what these are first...

Technical Debt (aka: TD)

sonarqube

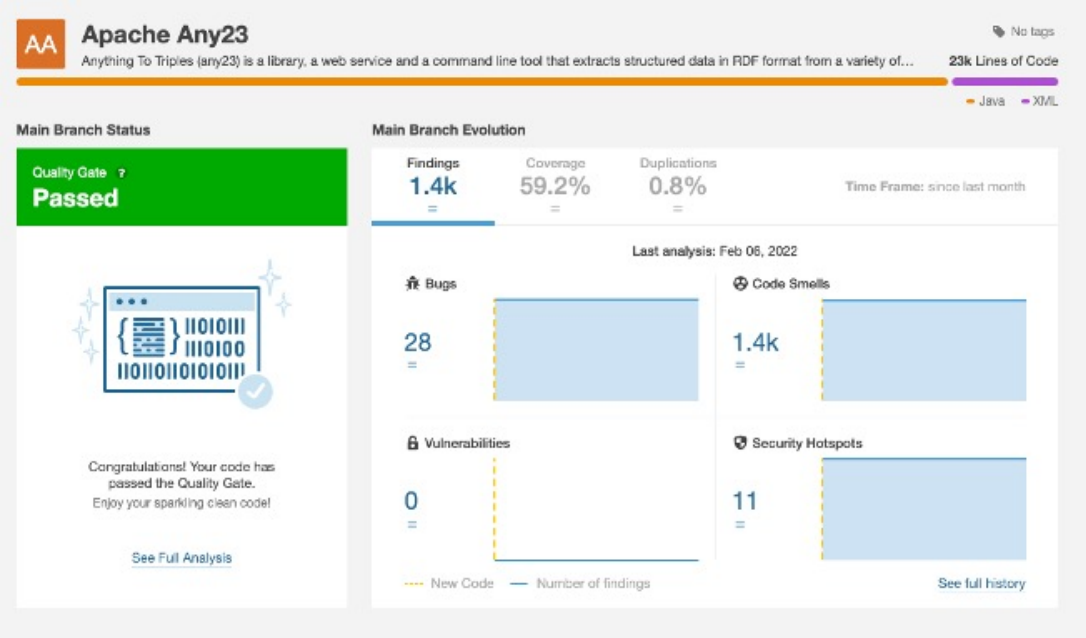


- Technical Debt is a metaphor to discuss the long-term consequences of *sub-optimal* decisions taken to favor the *need-for-speed*
- Technical Debt seems an umbrella term for all negative consequence that happen to our software assets (code, tests, requirements…)
- **“Yes! we have a lot of this…”** is what we hear when we refer to it.

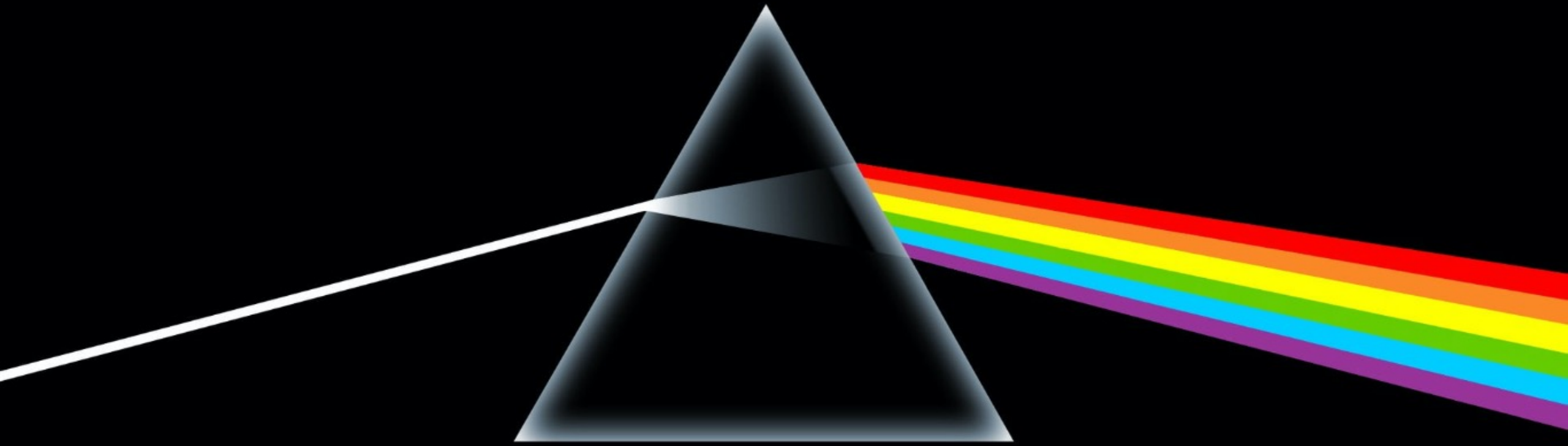
Technical Debt (aka: TD)

Technical Debt (aka: TD)

sonarqube



- Technical Debt data typically includes:
  - Issues (Code-Smells, Bugs, Vulnerabilities)
    - Where each Issue has been found
    - Severity
    - Remediation time for each Issue



The uncharted side...

What would change in our ways of working during a pandemic: The Tale of the Two

SERL Sweden
LEADING SOFTWARE ENGINEERING

Companies, data and analysis

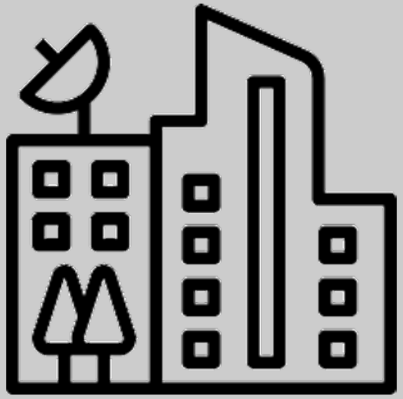
Company A

Scope: whole company data (2,026 engineers)

Locations: Sweden, the UK, USA

Quantitative analysis:

- Volume of code produced
- Calendar meeting patterns: number of meetings, duration, time spent in meetings
- Slack communication in public channels
- Work patterns – distribution of activities (commits, meetings, slack posts) during the day



Company B

Scope: one product in the company data (178 engineers)

Locations: Sweden, Brazil, India

Quantitative analysis:

- Volume of code committed to review (from pull requests)
- Deployment history and time to deploy
- Work patterns – distribution of activities (commits, reviews) during the day

Work done together with Darja Šmithe, Erik Klotnins, and Nils Broede Moe https://arxiv.org/abs/2101.08315

Amount of code produced

Work done together with Darja Šmithe, Erik Klotnins, and Nils Broede Moe https://arxiv.org/abs/2101.08315

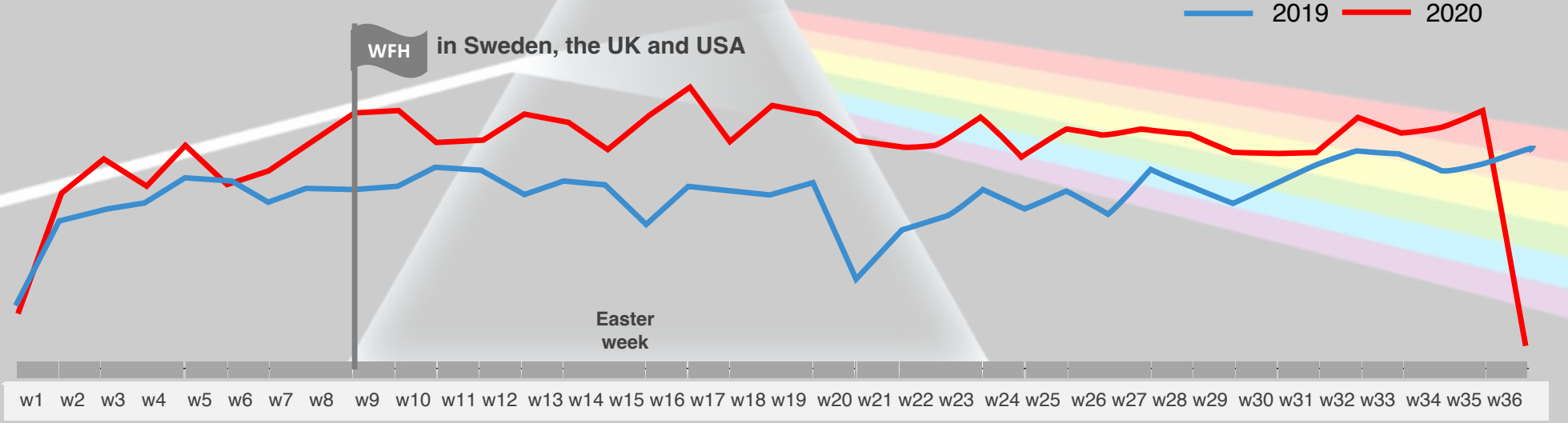
Disclaimer!

We are not using this data to measure productivity!
Only the volume of code being produced!

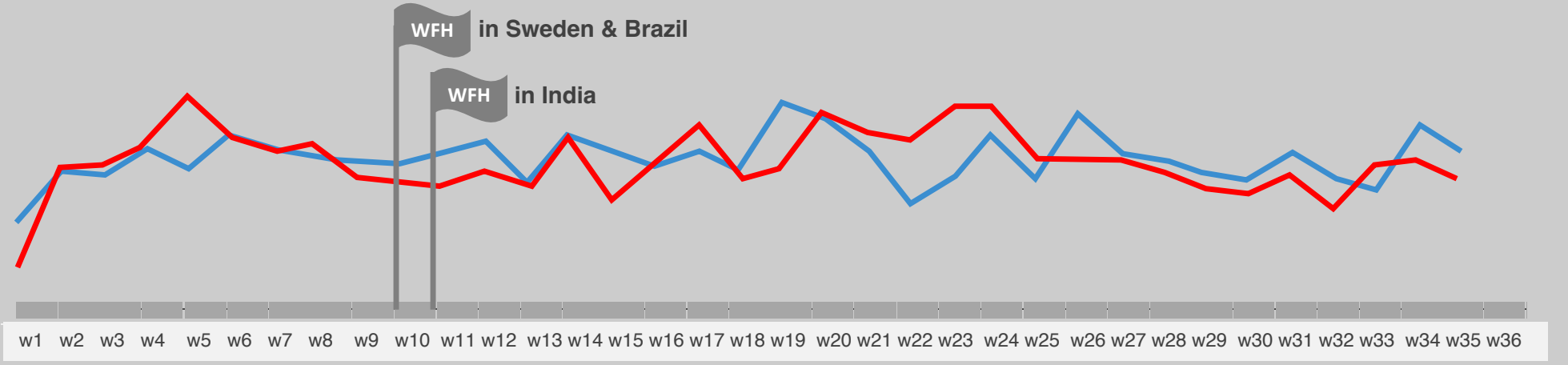
Work done together with Darja Šmithe, Erik Klotnins, and Nils Broede Moe https://arxiv.org/abs/2101.08315

Code production in the companies

Company A

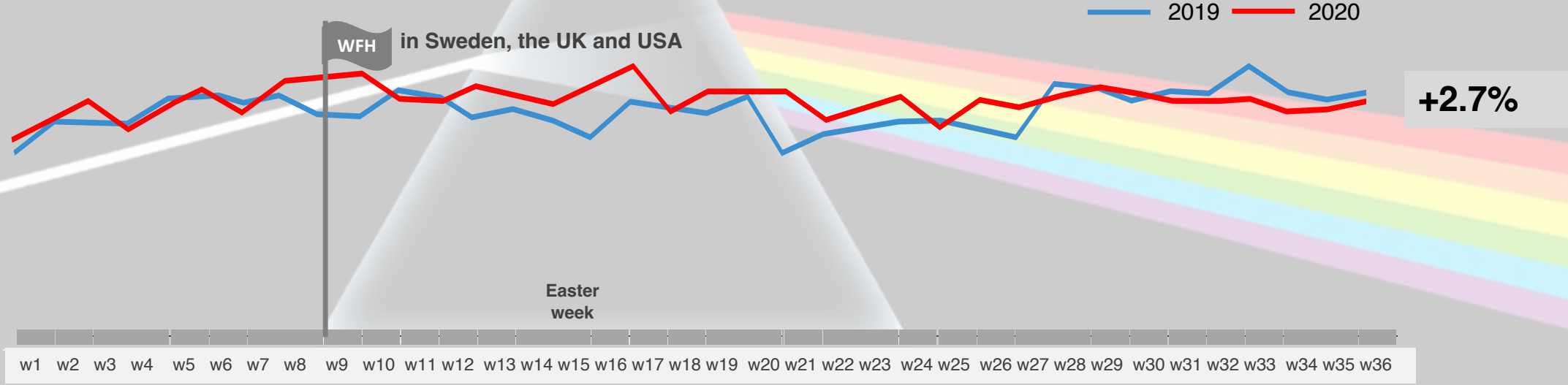


Company B

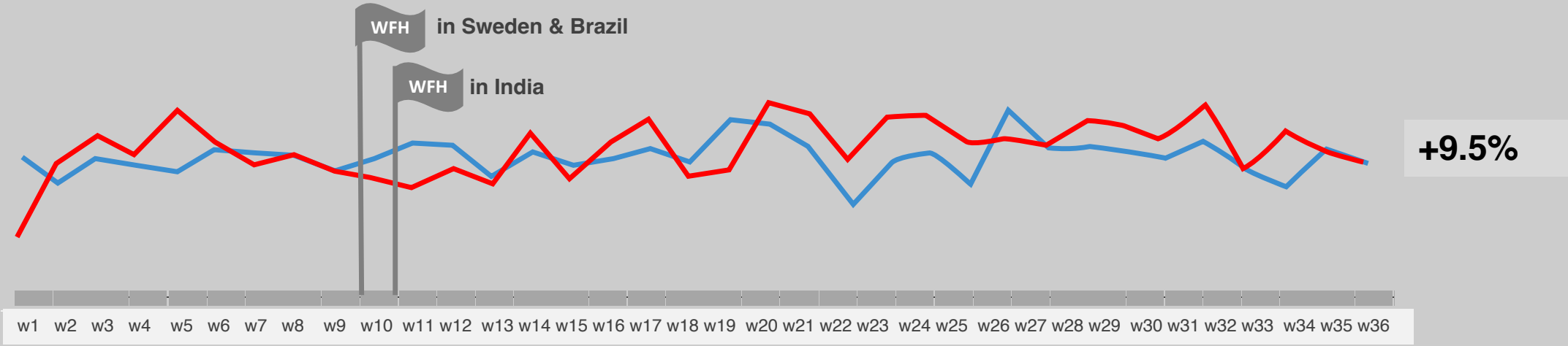


Code production normalized per developer

Company A



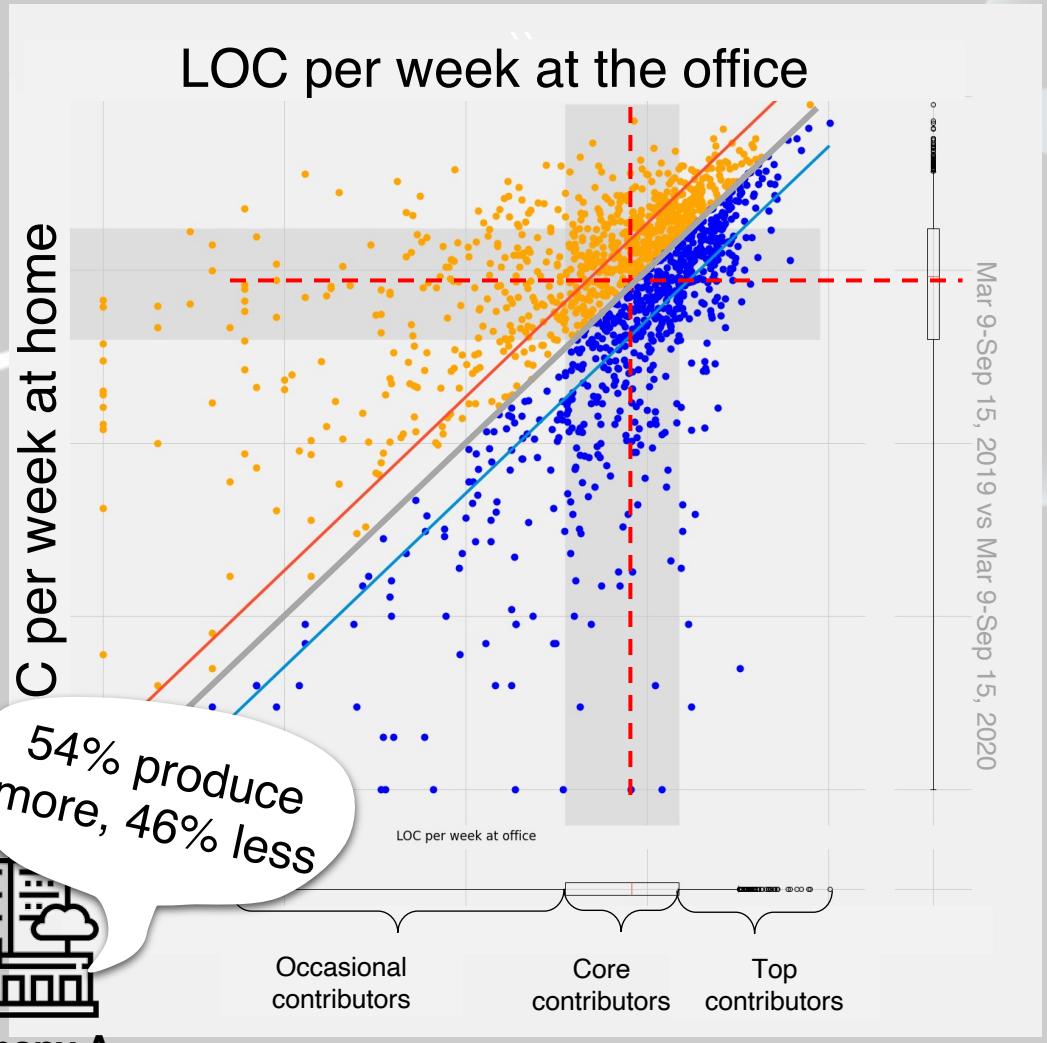
Company B



Work done together with Darja Šmithe, Erik Klotnins, and Nils Broede Moe https://arxiv.org/abs/2101.08315

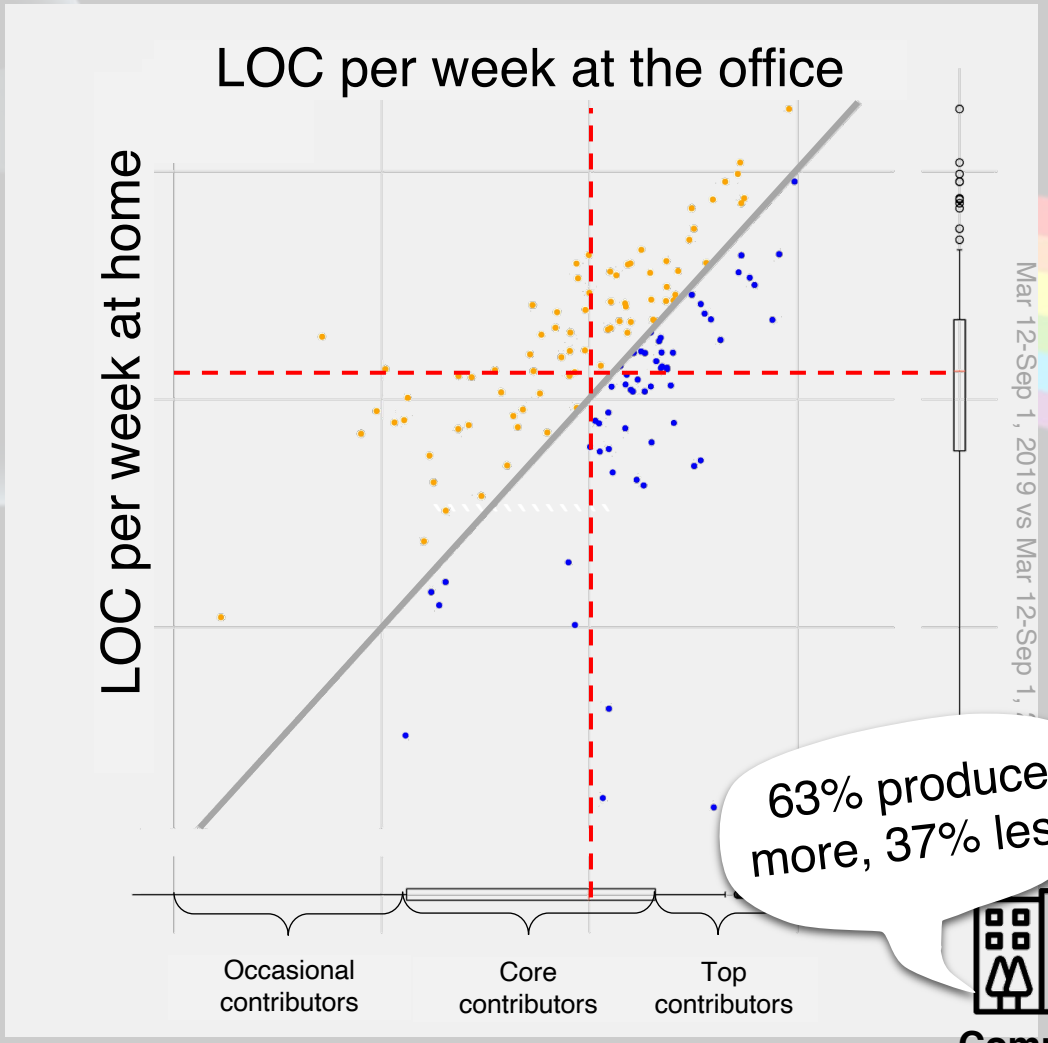
Work done together with Darja Šmithe, Erik Klotnins, and Nils Broede Moe https://arxiv.org/abs/2101.08315

Individual developers



Company A

54% are producing more and 46% less

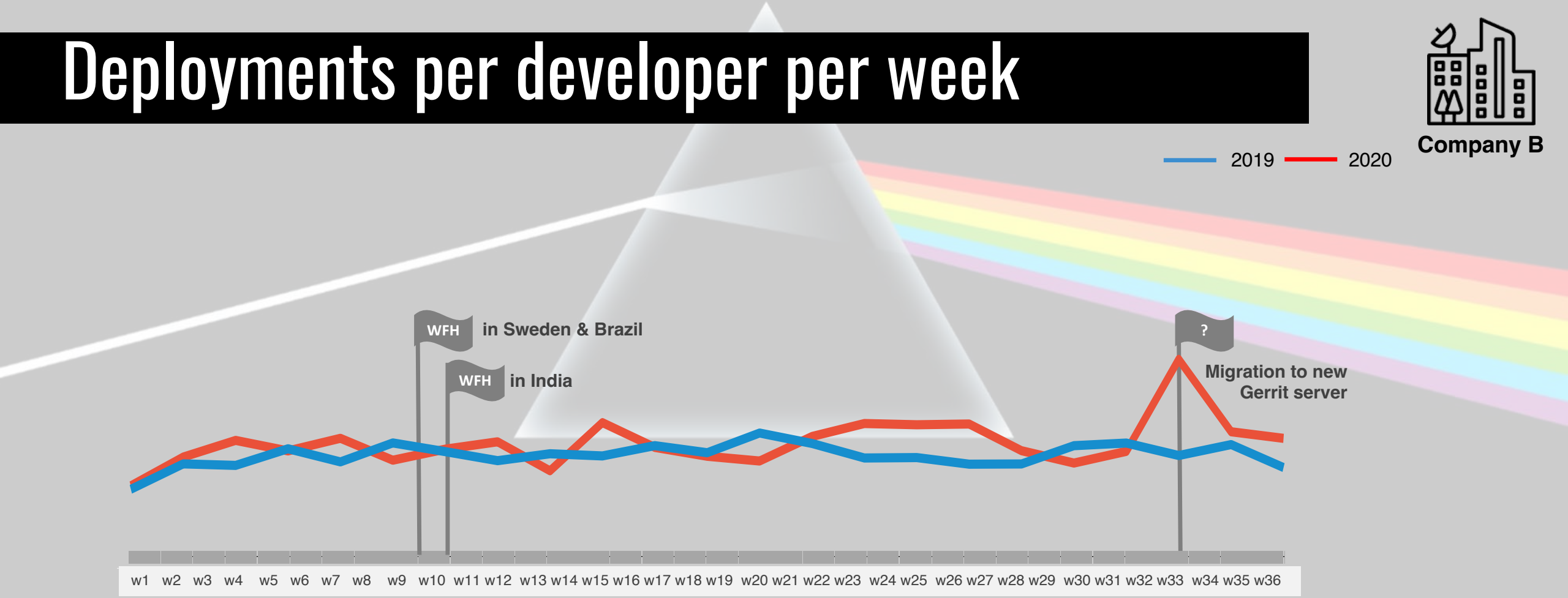


Company B

62.5% are producing more and 37.5% less

Work done together with Darja Šmithe, Erik Klotnins, and Nils Broede Moe https://arxiv.org/abs/2101.08315

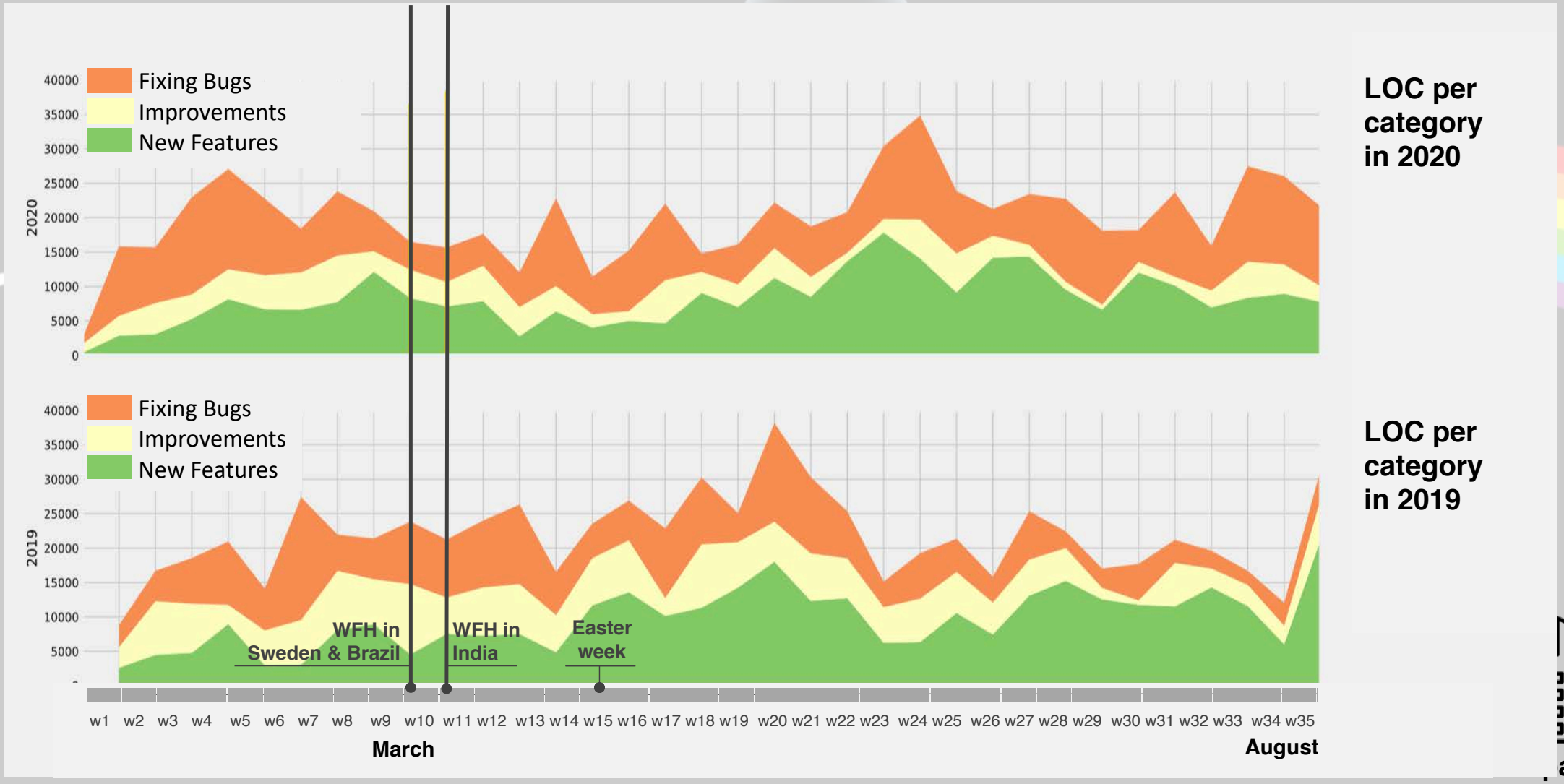
Deployments per developer per week



git Gerrit

Work done together with Darja Šmithe, Erik Klotnins, and Nils Broede Moe https://arxiv.org/abs/2101.08315

Analyzing the type of work



LOC per category in 2020

LOC per category in 2019

Work done together with Darja Šmithe, Erik Klotnins, and Nils Broede Moe https://arxiv.org/abs/2101.08315

Code Reviews: Time until finished

git Gerrit

Company B

Average time to merge a pull request:

36.5h Mar-Apr 2019		**35.8h** Mar-Aug 2019	
37.5h Mar-Apr 2020	**Slower**	**33.6h** Mar-Aug 2020	**Faster**

Median time to merge a pull request:

3.8h Mar-Apr 2019		**3.6h** Mar-Aug 2019	
3.4h Mar-Apr 2020	**Faster**	**2.7h** Mar-Aug 2020	**Faster**

Developers merge their code faster:

- Mar-May 2020 – 50% of developers merge PRs faster (than in Mar-May 2019)
- Mar-Sep 2020 – 61% of developers merge PRs faster (than in Mar-Sep 2019)

Work done together with Darja Šmithe, Erik Klotnins, and Nils Broede Moe https://arxiv.org/abs/2101.08315

Daily routines

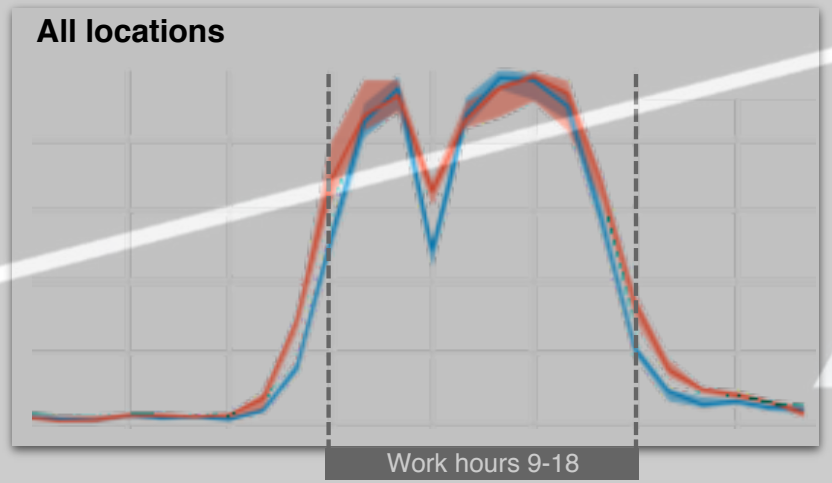
+1.5-2h

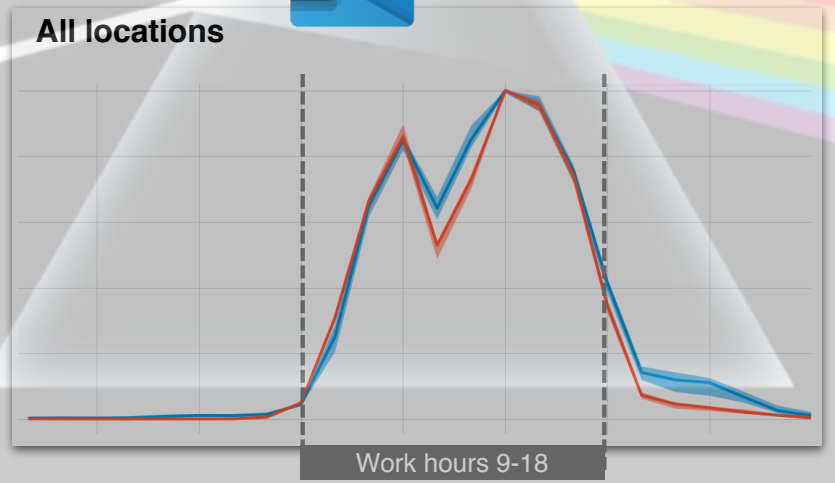
Company A

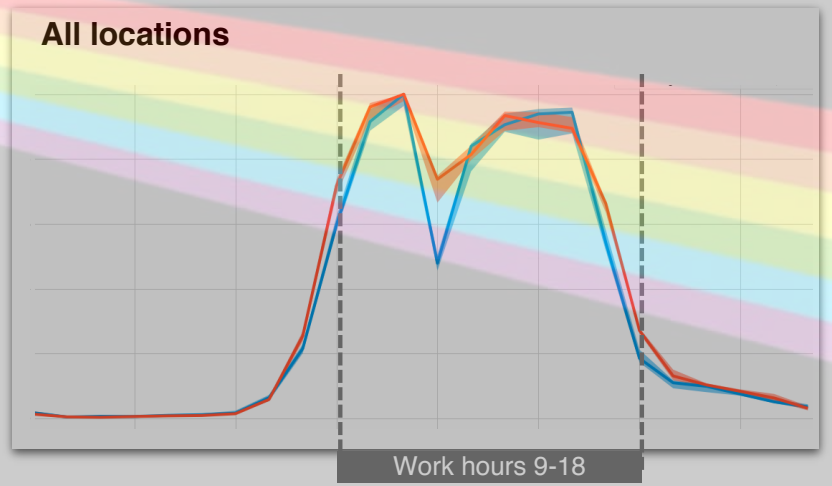
Deployments: git

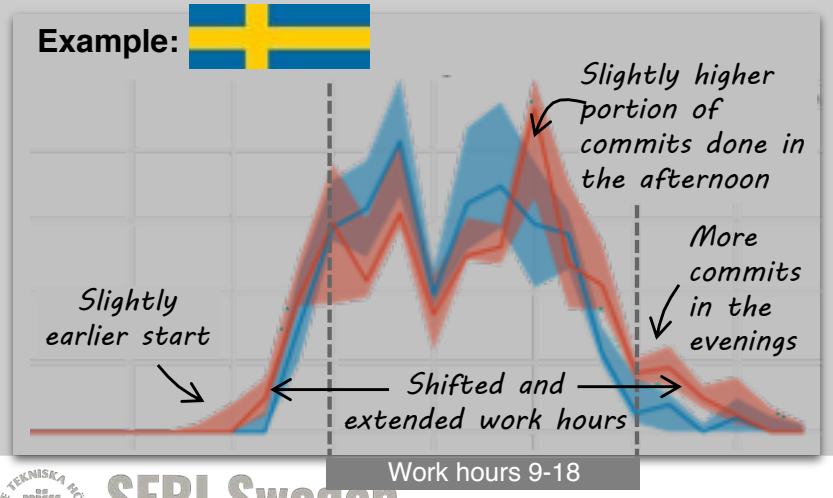
Meetings:

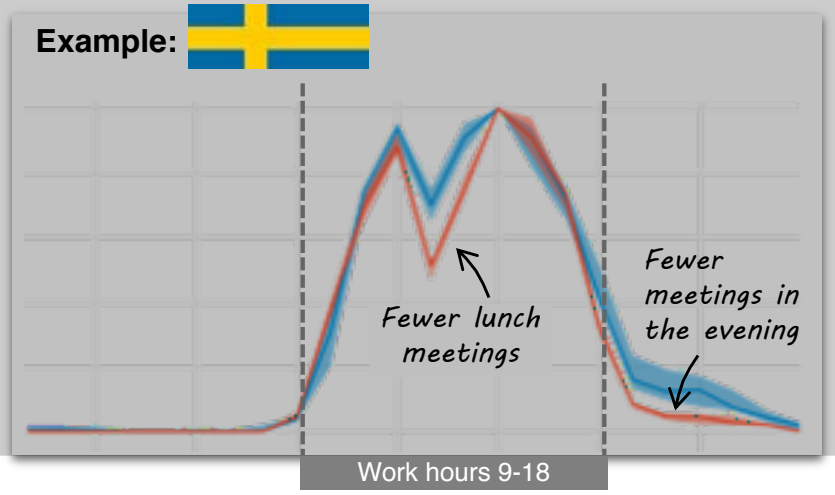
slack

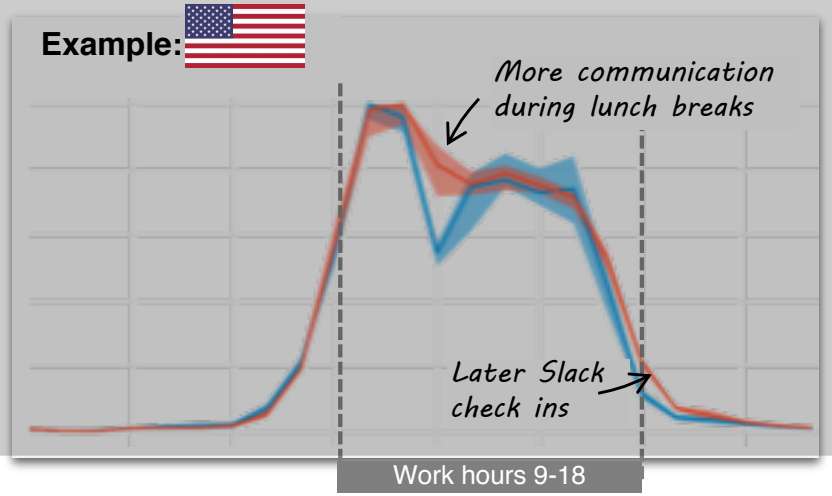












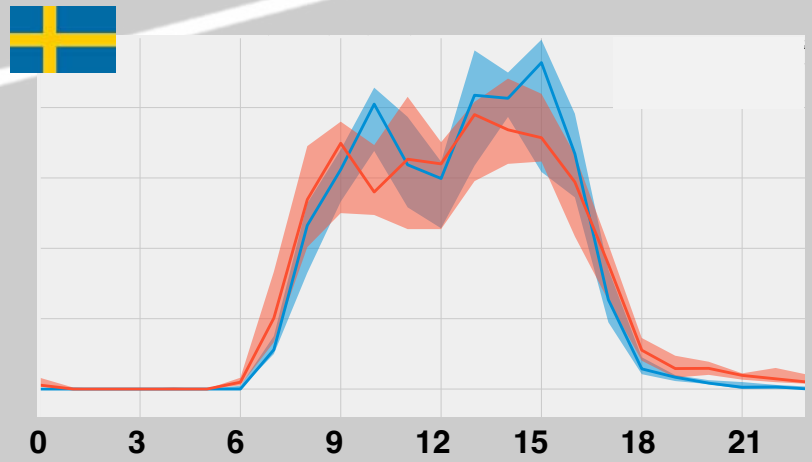
SERL Sweden
LEADING SOFTWARE ENGINEERING

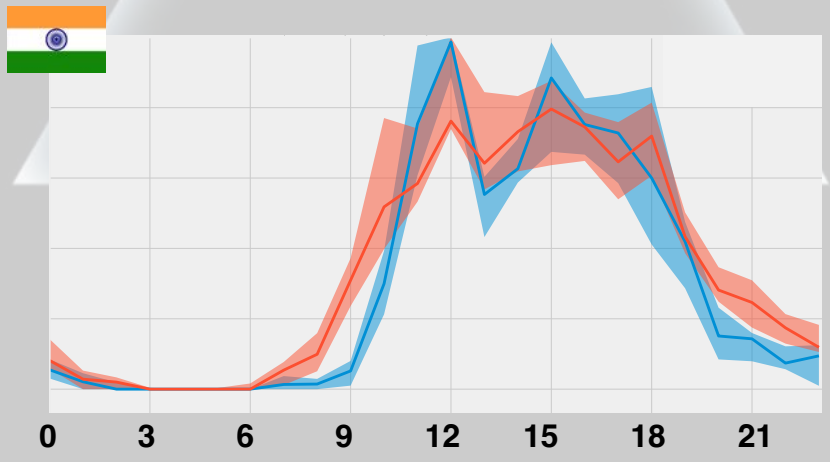
Office work (2020-01-01 – 2020-03-11) Work from home (2020-03-11 – 2020-05-30)

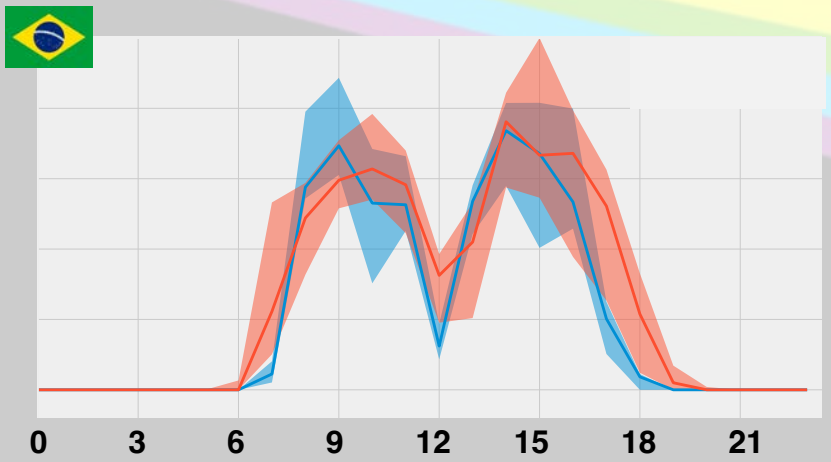
Distribution of pull requests during the day

Company B

git Gerrit



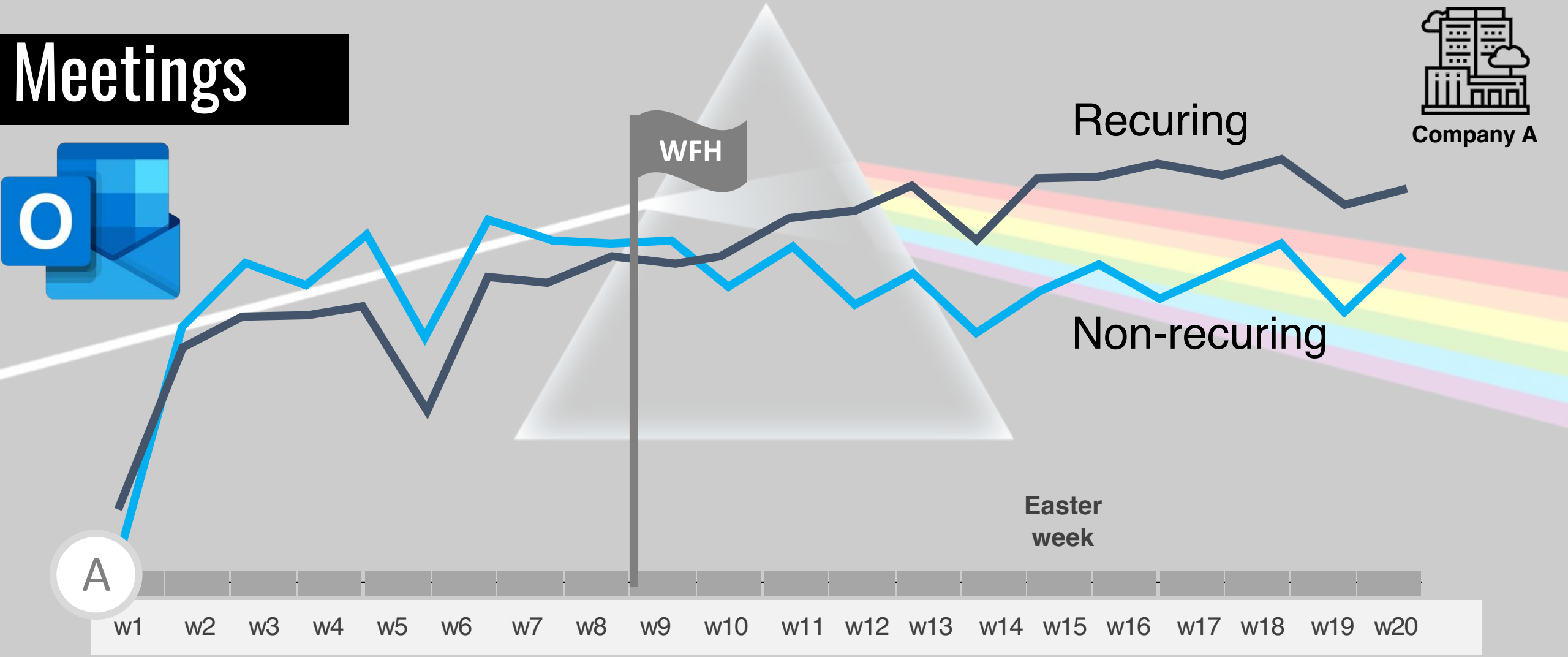




Office work (2020-01-01 – 2020-03-11) Work from home (2020-03-11 – 2020-05-30)

Work done together with Darja Šmithe, Erik Klotnins, and Nils Broede Moe https://arxiv.org/abs/2101.08315

Meetings



Recuring versus non-recuring meetings

Spontaneous meetings are now scheduled on a regular basis

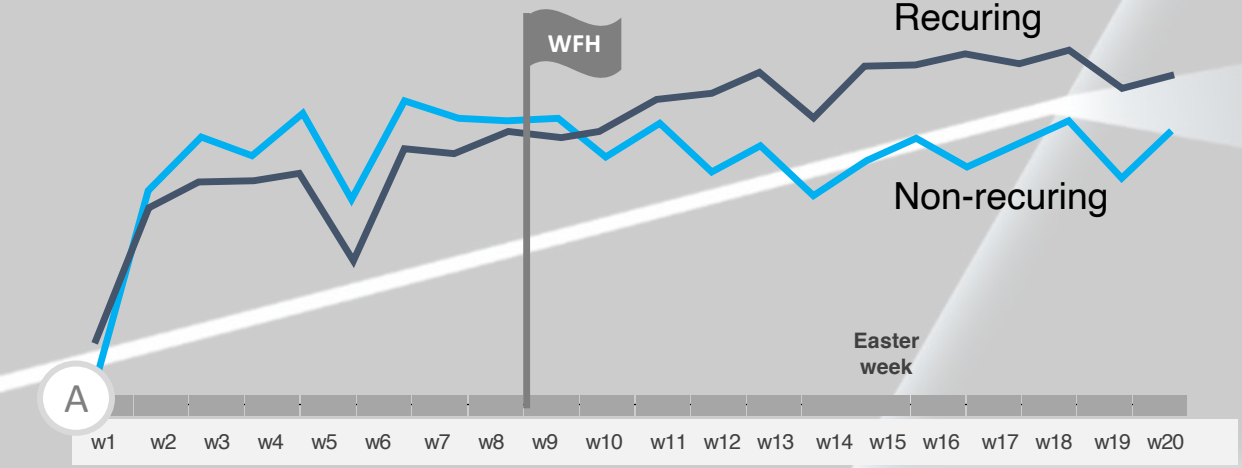
Work done together with Darja Šmithe, Erik Klotnins, and Nils Broede Moe https://arxiv.org/abs/2101.08315

Meetings

20 shades of blue graph © by

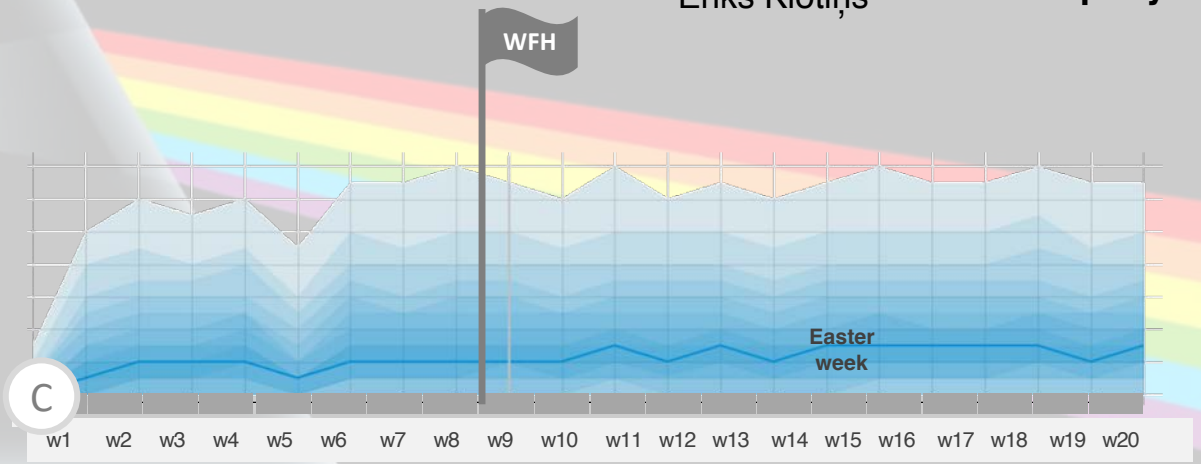
Ēriks Klotiņš

Company A



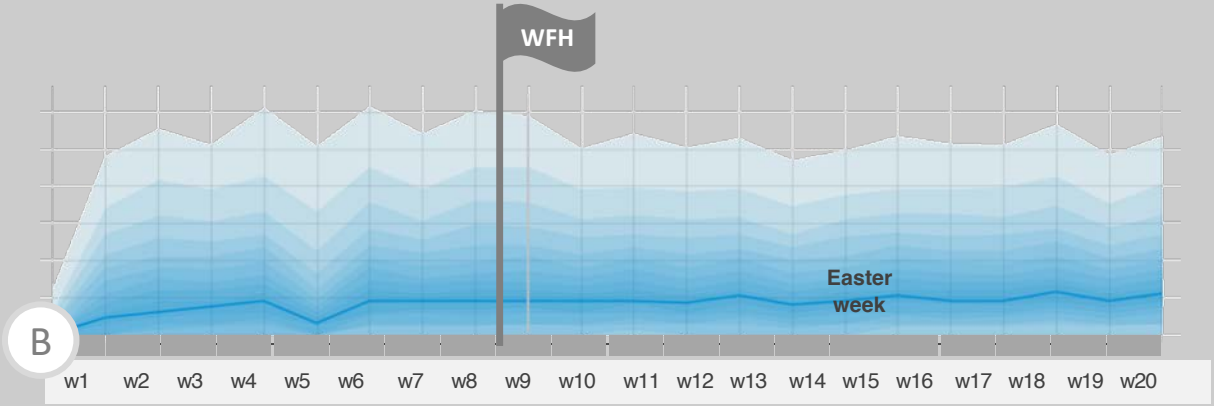
Recuring versus non-recuring meetings

Spontaneous meetings are now scheduled on a regular basis



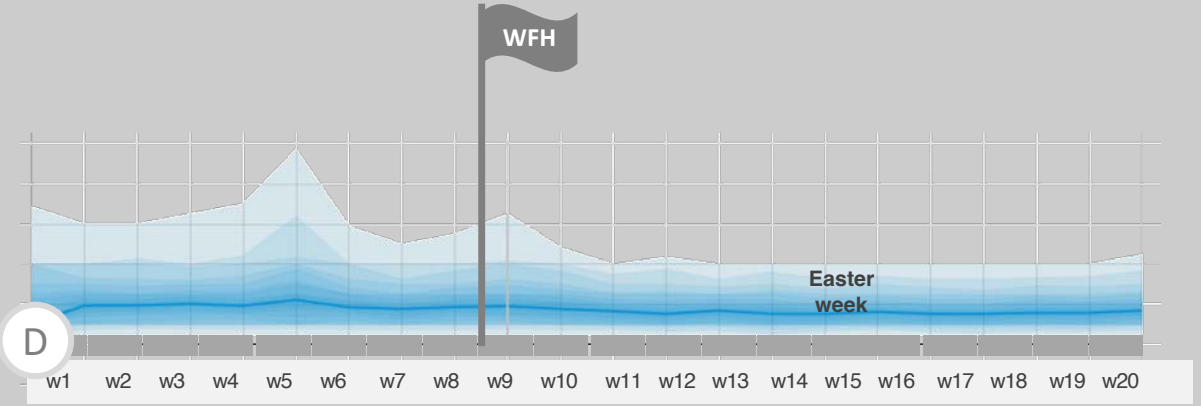
Number of meetings per week

More meetings



Minutes spent in meetings per week

Less time spent in meetings due to a decreased amount of long meetings



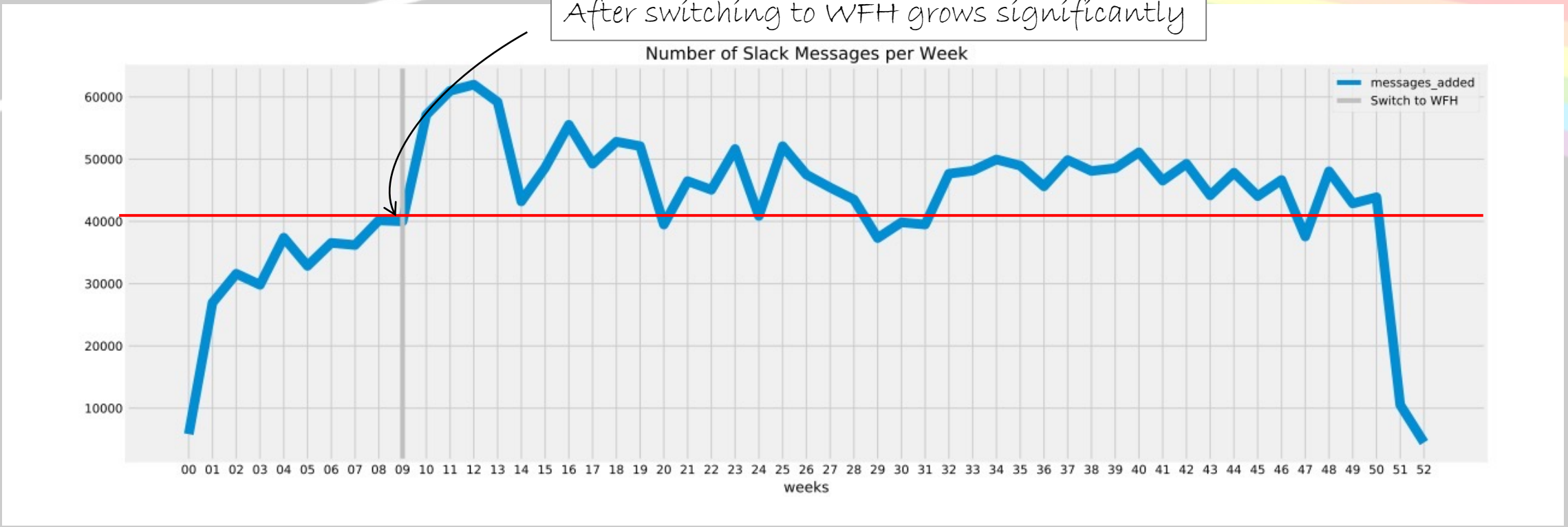
Meeting duration in minutes

Slightly shorter meetings

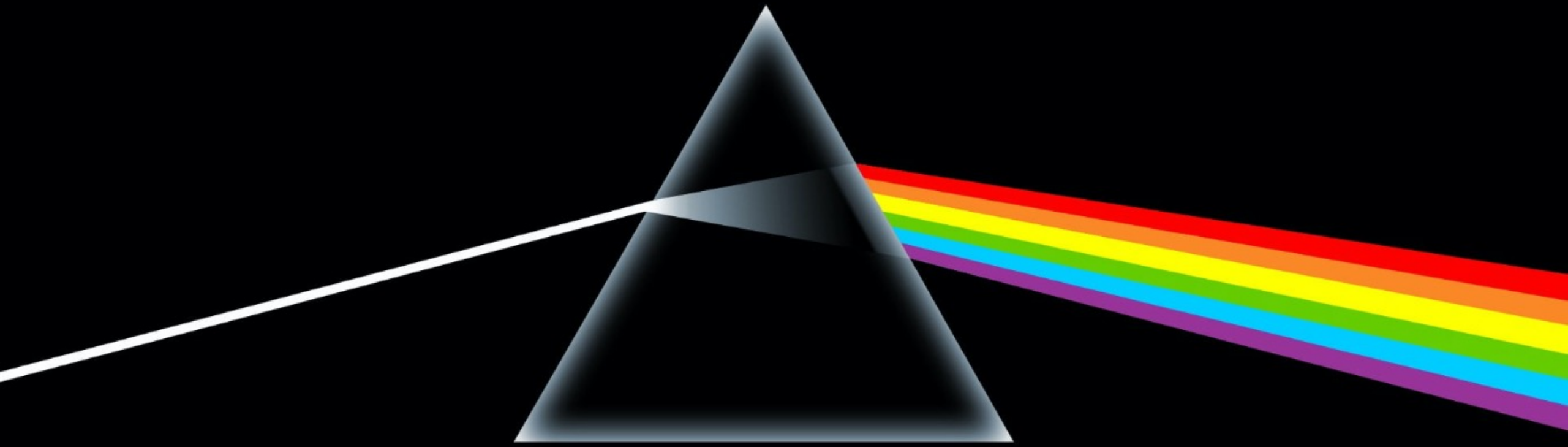
SERL Sweden
LEADING SOFTWARE ENGINEERING

Work done together with Darja Šmithe, Erik Klotnins, and Nils Broede Moe https://arxiv.org/abs/2101.08315

Slack Communication



Work done together with Darja Šmithe, Erik Klotnins, and Nils Broede Moe https://arxiv.org/abs/2101.08315



The uncharted side...

WFH and its impact on the accumulation of TD

SERL Sweden
LEADING SOFTWARE ENGINEERING

Companies, data and analysis

Company C

Scope: whole company data (>800 employees)

Locations: Sweden

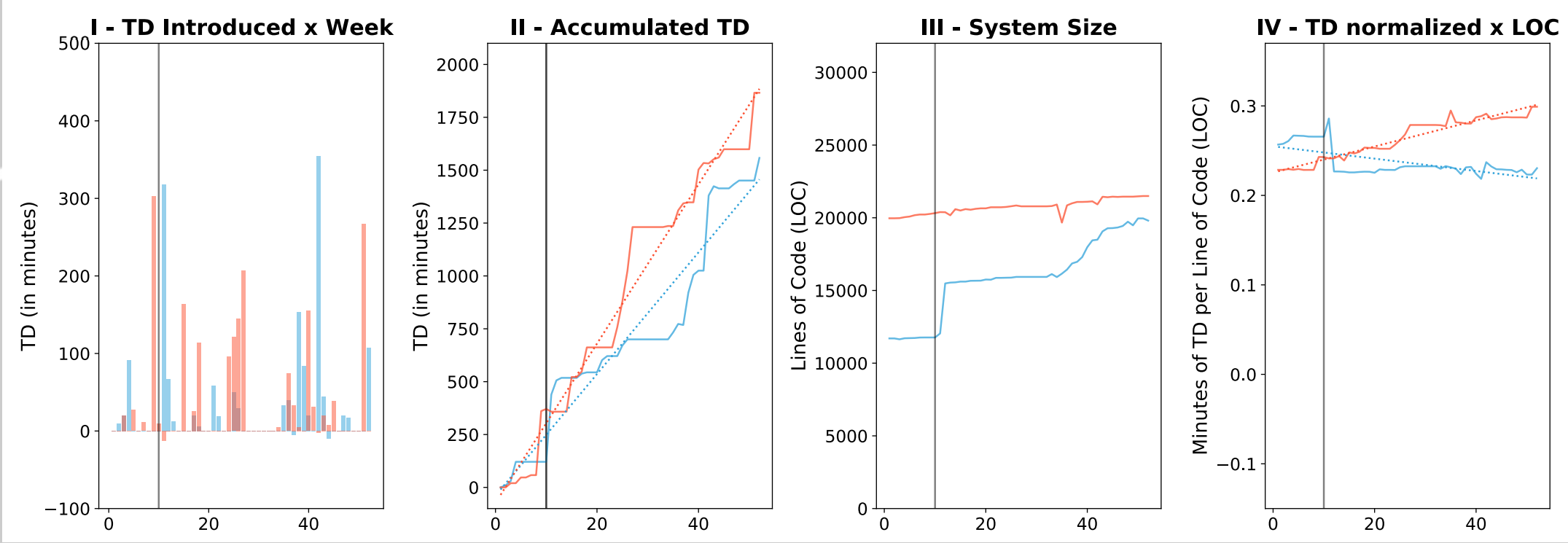
Quantitative analysis:

- SonarQube Data
- Code Review
- Commit Data

WFH and the accumulation of TD

Ehsan Zabardast, Javier Gonzalez-Huerta, Francis Palma: The Impact of Forced Working-From-Home on Technical Debt: An Industrial Case Study, Submitted to TechDebt2022 (under-review) preprint available

Results (summarizing): Pattern 1

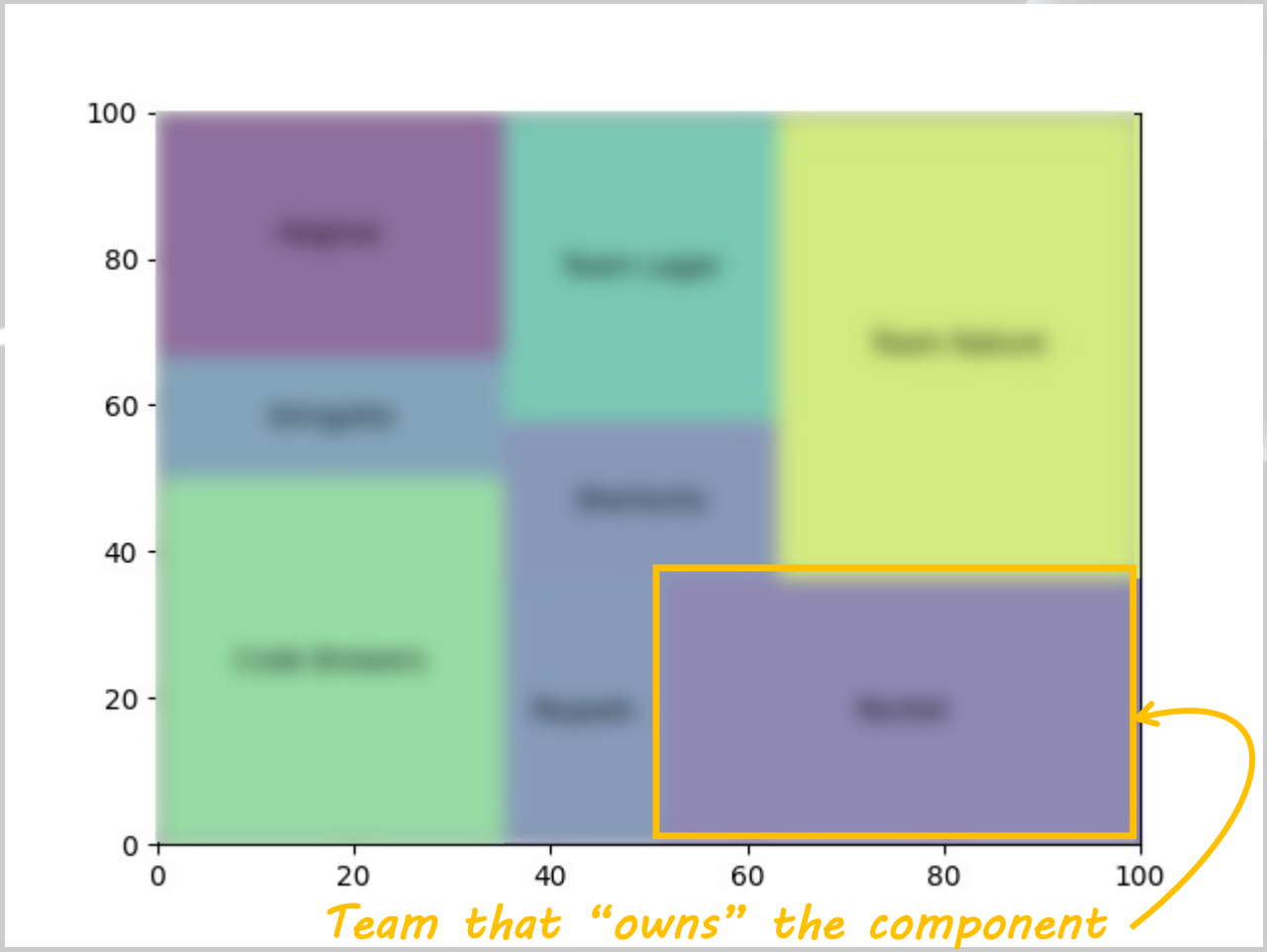


sonarqube git Gerrit

Ehsan Zabardast, Javier Gonzalez-Huerta, Francis Palma: The Impact of Forced Working-From-Home on Technical Debt: An Industrial Case Study, Submitted to TechDebt2022 (under-review) preprint available

Results (summarizing): Pattern 1

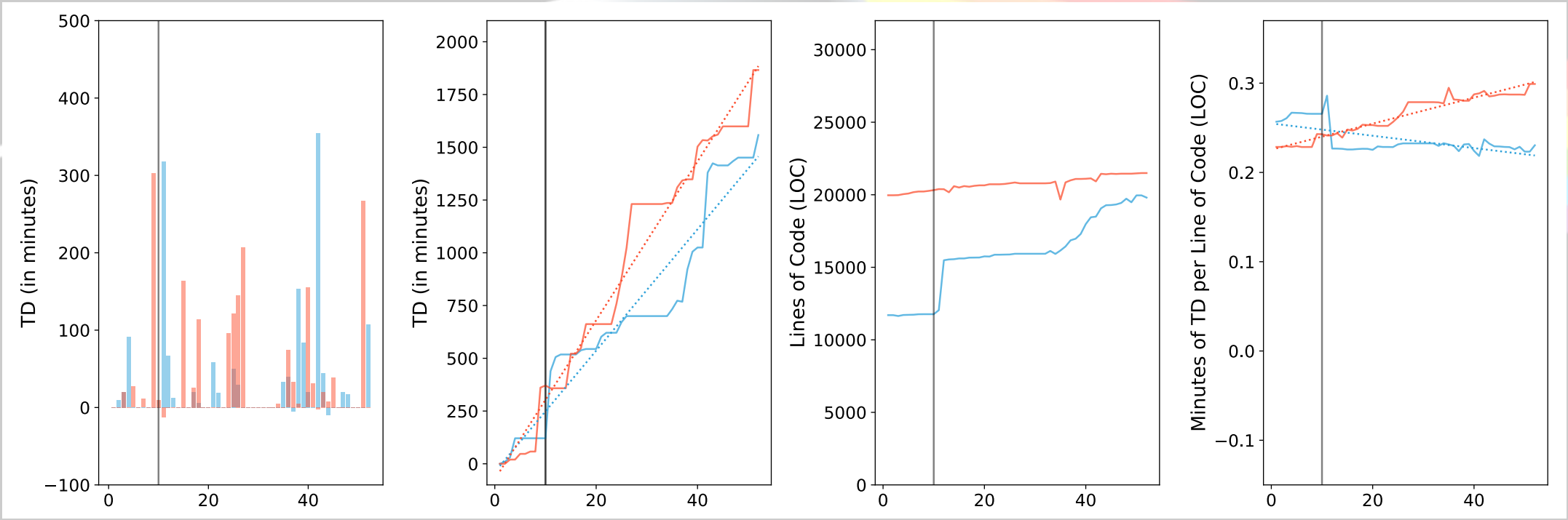
2019
2020
Trend 2019
Trend 2020
switch to WFH (in 2020)



git Gerrit

Focus-Group Observation: The team who owns the component is not by far its main contributor. They are only ”gate keepers” for the code reviews

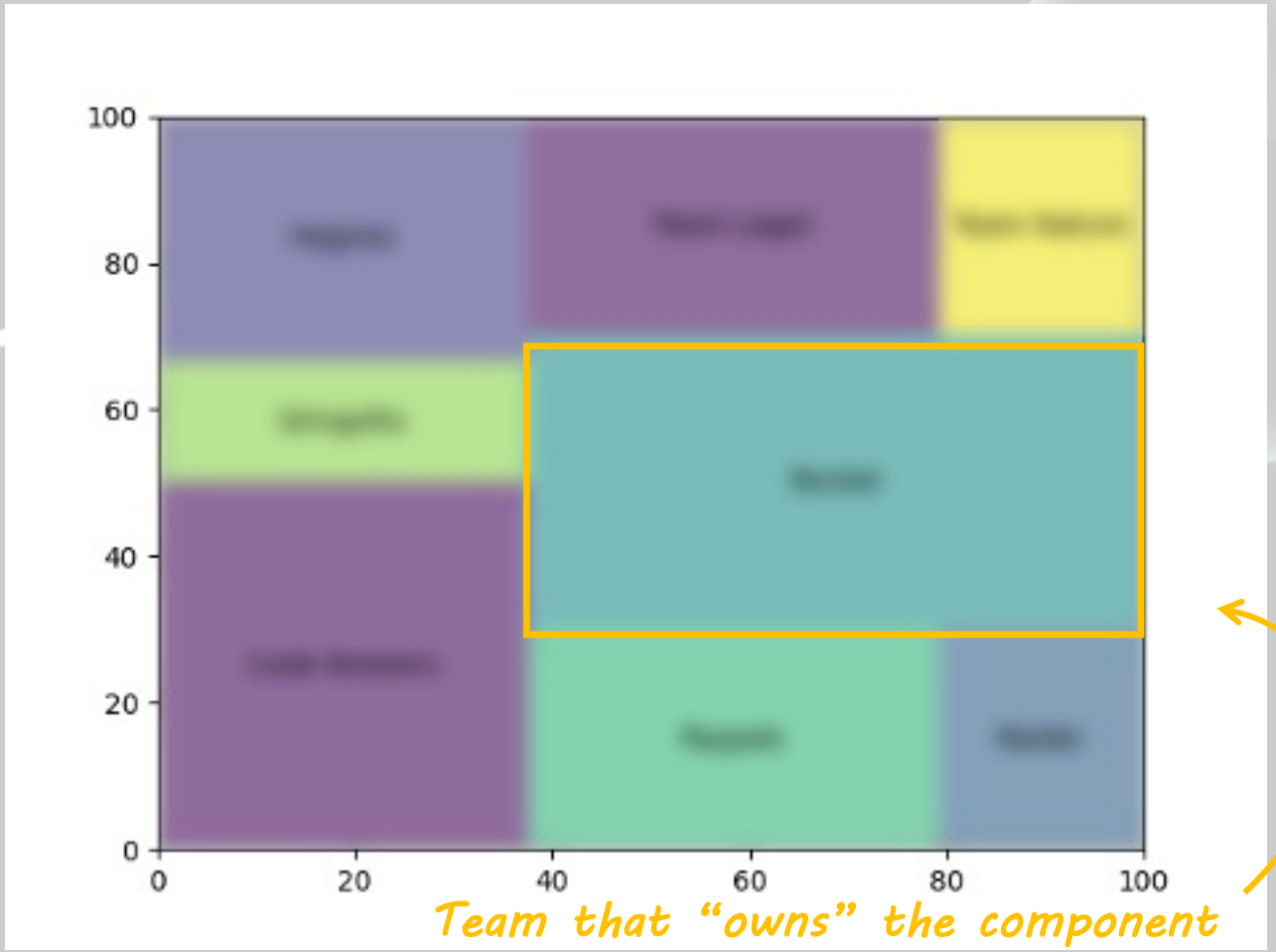
Results (summarizing): Pattern 2

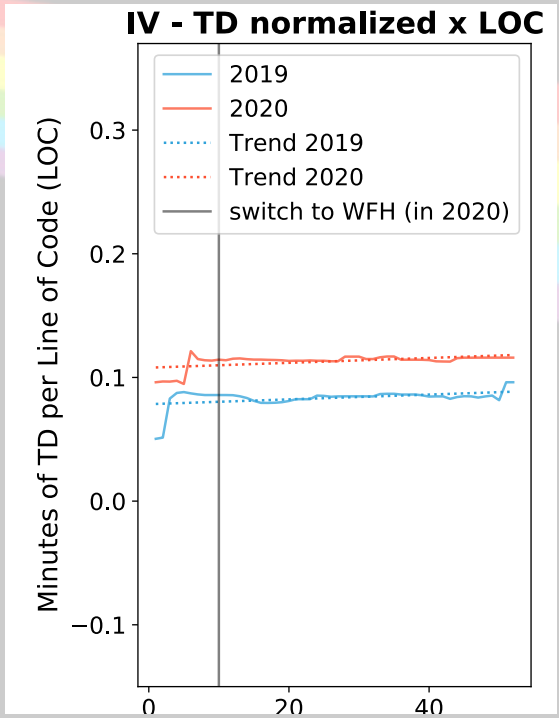


sonarqube git Gerrit

Ehsan Zabardast, Javier Gonzalez-Huerta, Francis Palma: The Impact of Forced Working-From-Home on Technical Debt: An Industrial Case Study, Submitted to TechDebt2022 (under-review) preprint available

Results (summarizing): Pattern 2





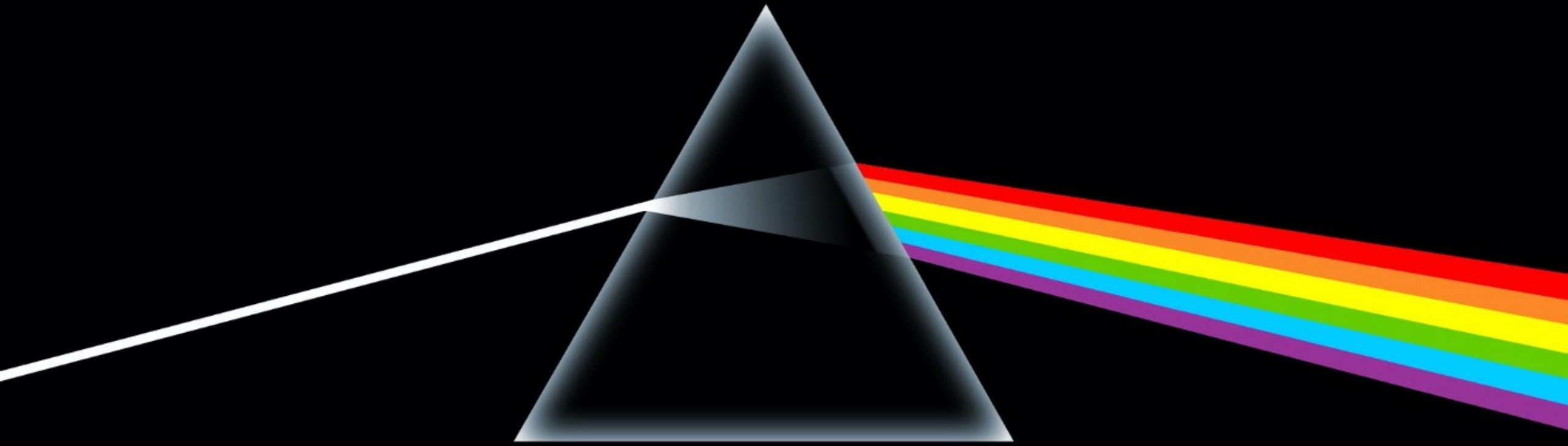
sonarqube git Gerrit

Ehsan Zabardast, Javier Gonzalez-Huerta, Francis Palma: The Impact of Forced Working-From-Home on Technical Debt: An Industrial Case Study, Submitted to TechDebt2022 (under-review) preprint available

Restults in a Nutshell

- When looking at individual commits, we cannot find statistically significant changes
  - Different commit frequencies WFH vs Office work
- We observe differences on the accumulation of TD while WFH vs Office work
  - These changes are statistically significant (for an extended version of the paper)
- One of the findings from the focus group is that ownership diffusion can be an explanation for faster accumulation of TD while WFH

Ehsan Zabardast, Javier Gonzalez-Huerta, Francis Palma: The Impact of Forced Working-From-Home on Technical Debt: An Industrial Case Study, Submitted to TechDebt2022 (under-review) preprint available



The uncharted side...

Ownership vs Contribution (OCAM): Detecting the Misalignment

SERL Sweden
LEADING SOFTWARE ENGINEERING

Companies, data and analysis

Company C

Scope: whole company data (>800 employees)

Locations: Sweden

Quantitative analysis:

- Commit
- Code Review Data
- Tickets
- Ownership Data

Currently Replicating in Company A

Scope: whole company data (2,026 engineers)

Locations: Sweden, the UK, USA

Quantitative analysis:

- Code Review
- Commit Data
- Ownership Data

Ehsan Zabardast, Javier Gonzalez-Huerta, Binish Tanveer: Ownership vs Contribution: Investigating the Alignment Between Ownership and Contribution. ICSA 2022 (Accepted, to appear) Preprint Available

Architecture and Organization

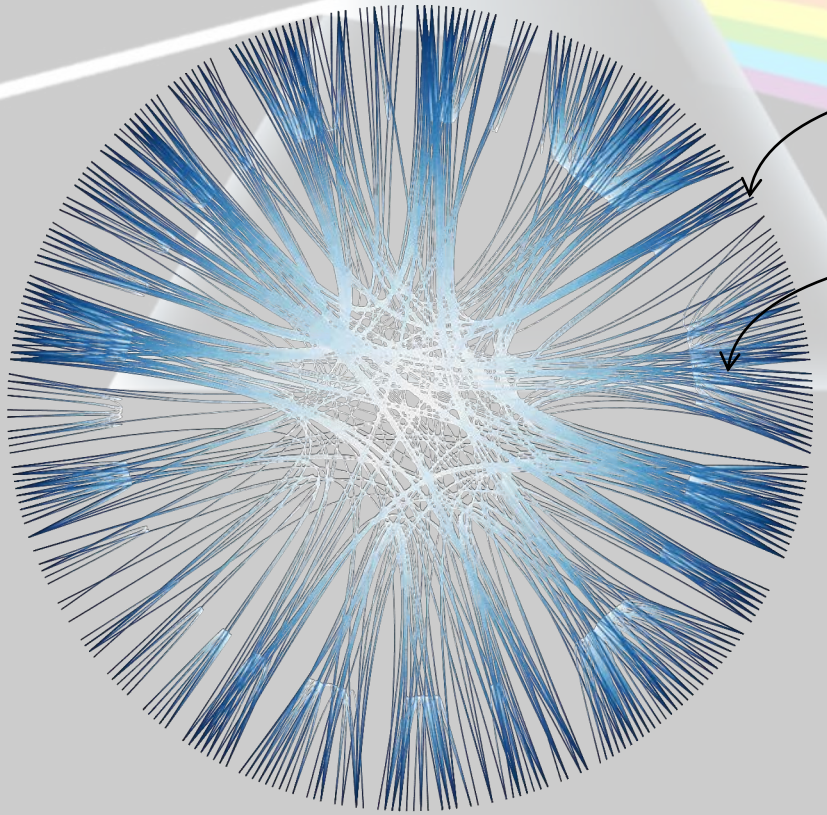
- Connected Repositories with Jira Tickets
- The Network
  - Nodes as repositories
  - Jira tickets as node size
  - Number of connections as edge strength

Company C

git JIRA

Company A - Task Dependencies: Who reviews what?

Gerrit



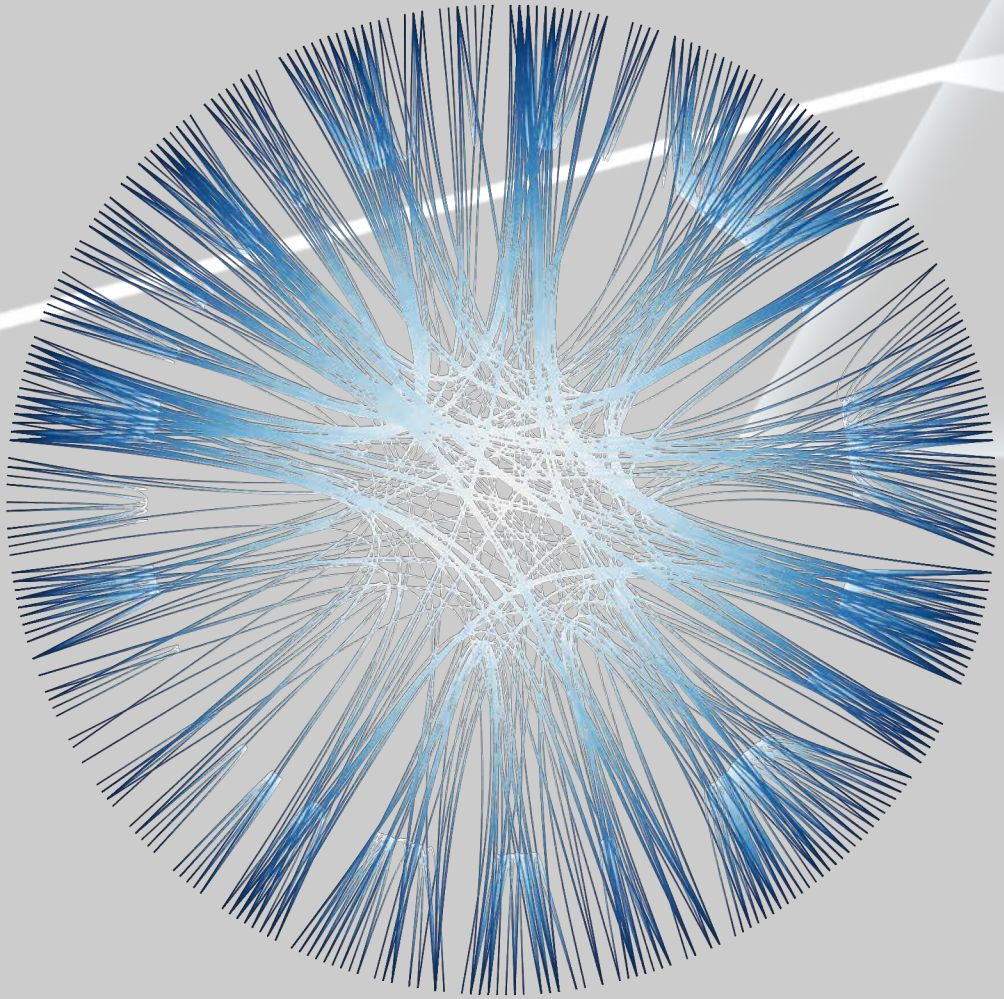
Each node is a team

Each group of
Close nodes is a sub-organization

Company A

Code Review Data: Who reviews what?

Gerrit

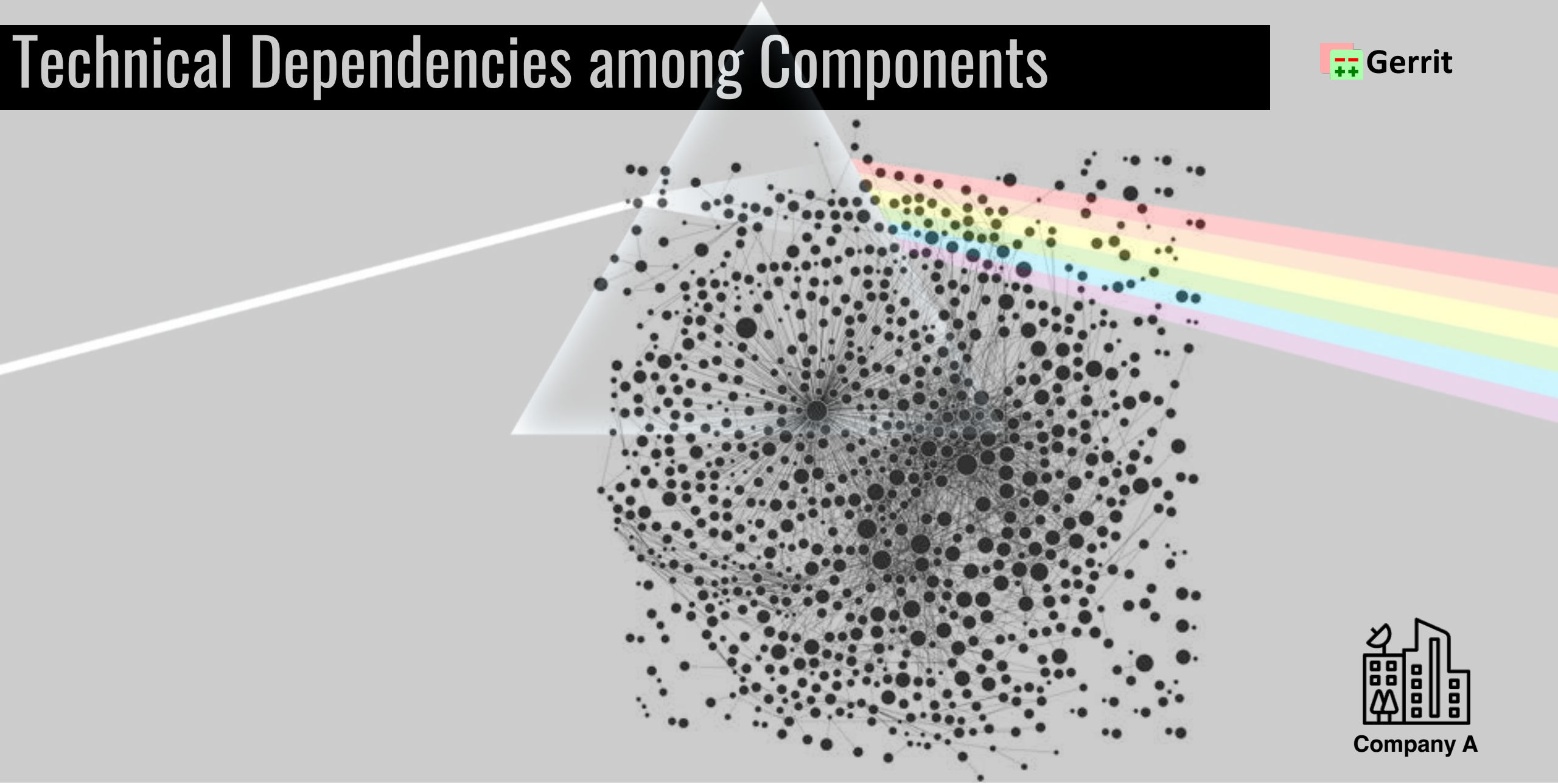


- The graphs shows strong interaction between suborganizations
- Here we only show one perspective (author vs reviewer)
- This gets even more complex since we are removing spurious interactions (consider more than 5 code reviews between teams)

Company A

Technical Dependencies among Components

Gerrit

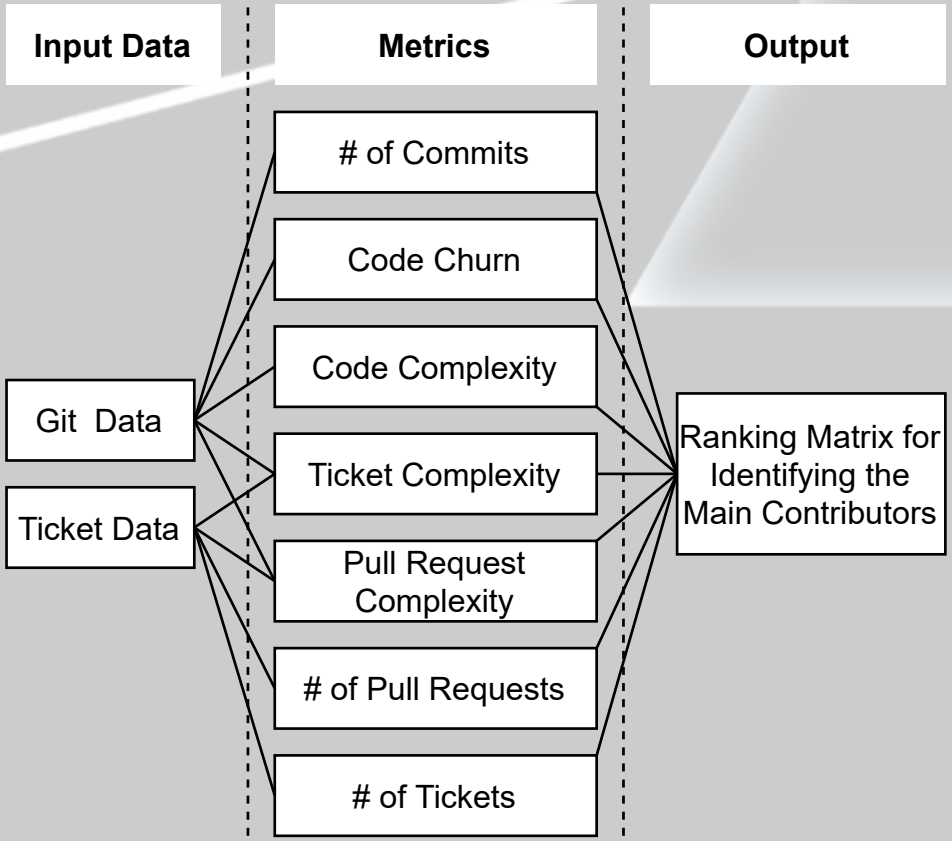


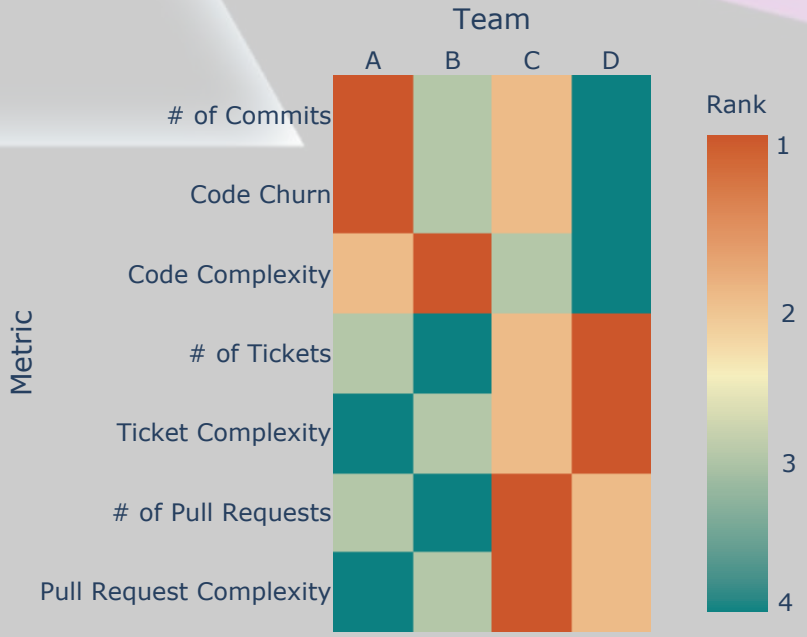
Company A

Ownership vs Contribution

Ehsan Zabardast, Javier Gonzalez-Huerta, Binish Tanveer: Ownership vs Contribution: Investigating the Alignment Between Ownership and Contribution. ICSA 2022 (Accepted, to appear) Preprint Available

What we did: OCAM Model





Ehsan Zabardast, Javier Gonzalez-Huerta, Binish Tanveer: Ownership vs Contribution: Investigating the Alignment Between Ownership and Contribution. ICSA 2022 (Accepted, to appear) Preprint Available

What we did

- Analyzed 267 components (repos)
- Find 193 missalignments
- We validated the results by doing a focus-group with members of two teams.
  - Inspected the results for 10 repos
  - All of them were correctly identified

Company C

Ehsan Zabardast, Javier Gonzalez-Huerta, Binish Tanveer: Ownership vs Contribution: Investigating the Alignment Between Ownership and Contribution. ICSA 2022 (Accepted, to appear) Preprint Available

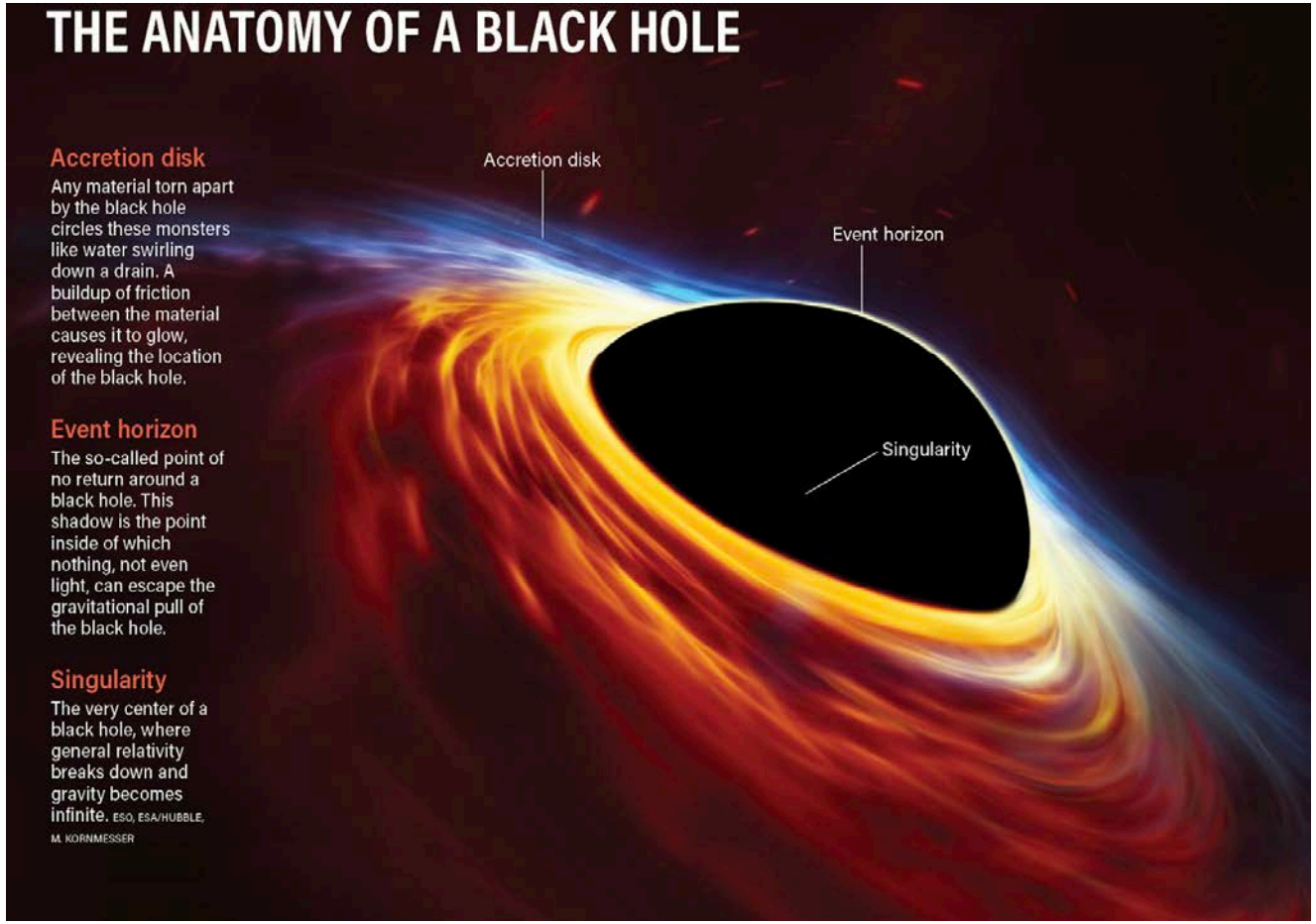
Where are we now?

Architecture and Organization

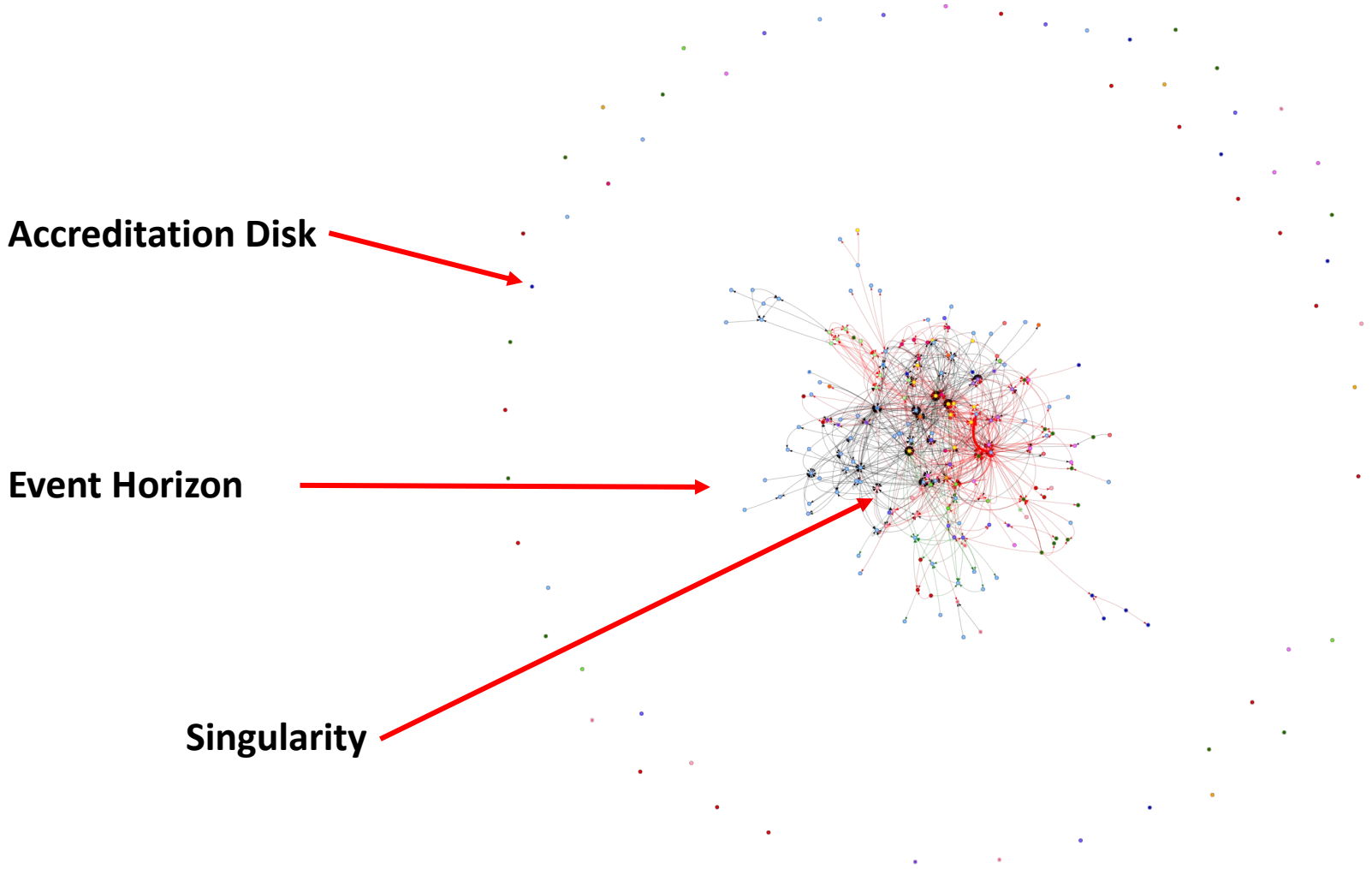
- Connected Repositories with Jira Tickets
- The Network
  - Nodes as repositories
  - Jira tickets as node size
  - Number of connections as edge strength

Architecture and Organization: Putting Everything Together

Supermassive Black Hole



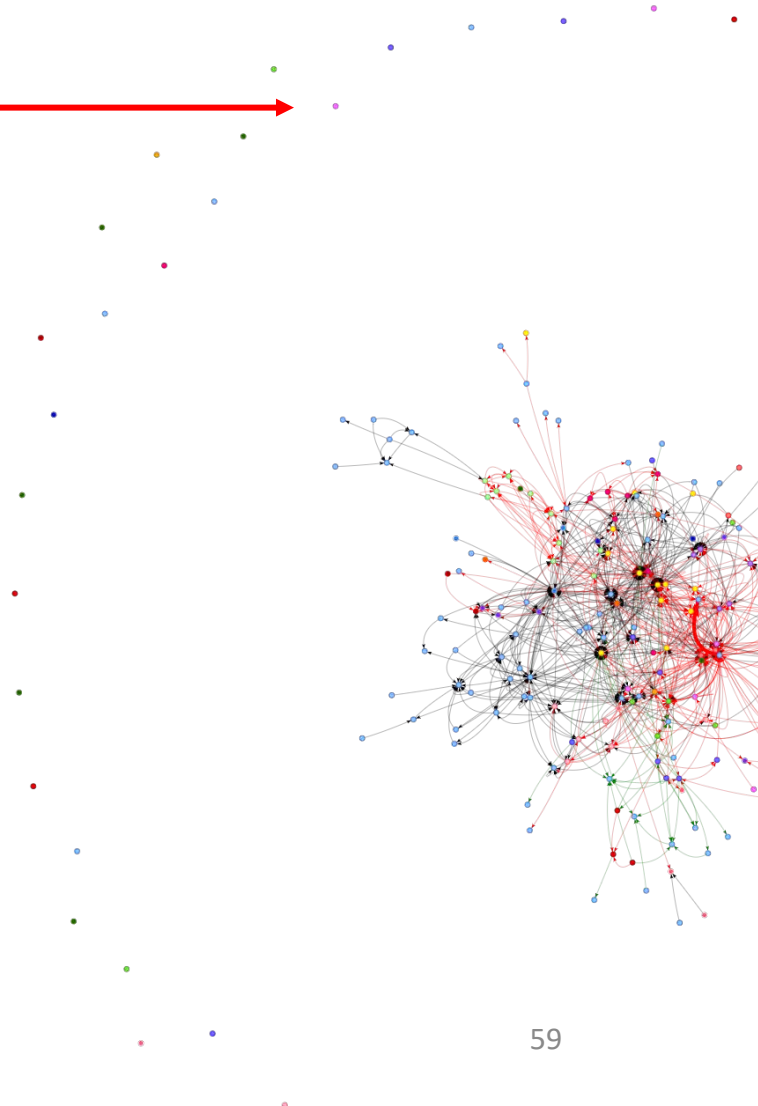
Architecture and Organization: Putting Everything Together



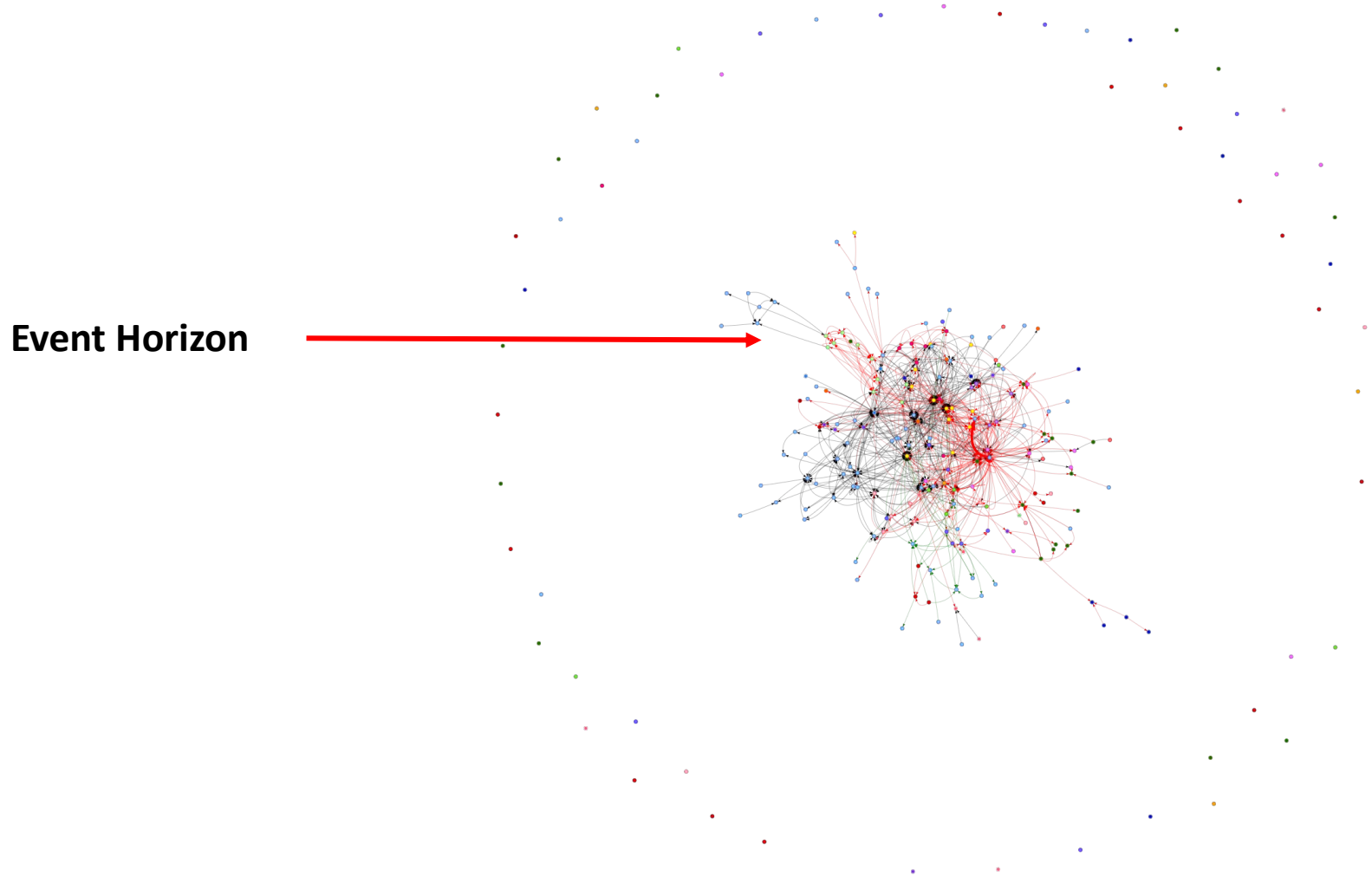
Architecture and Organization: Putting Everything Together

Accreditation Disk

- Mostly isolated repos
- The dependencies are mostly technical
- The problems can be address easily
- Limited to No propagation of problems



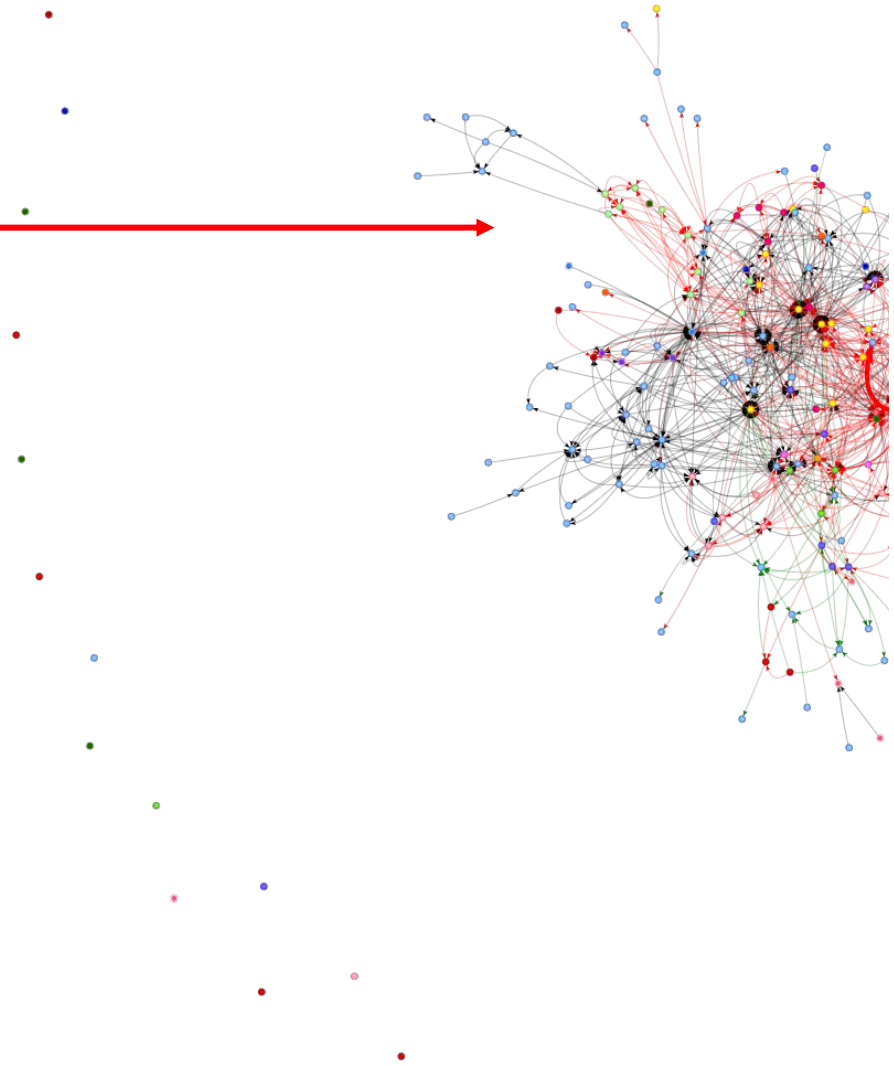
Architecture and Organization: Putting Everything Together



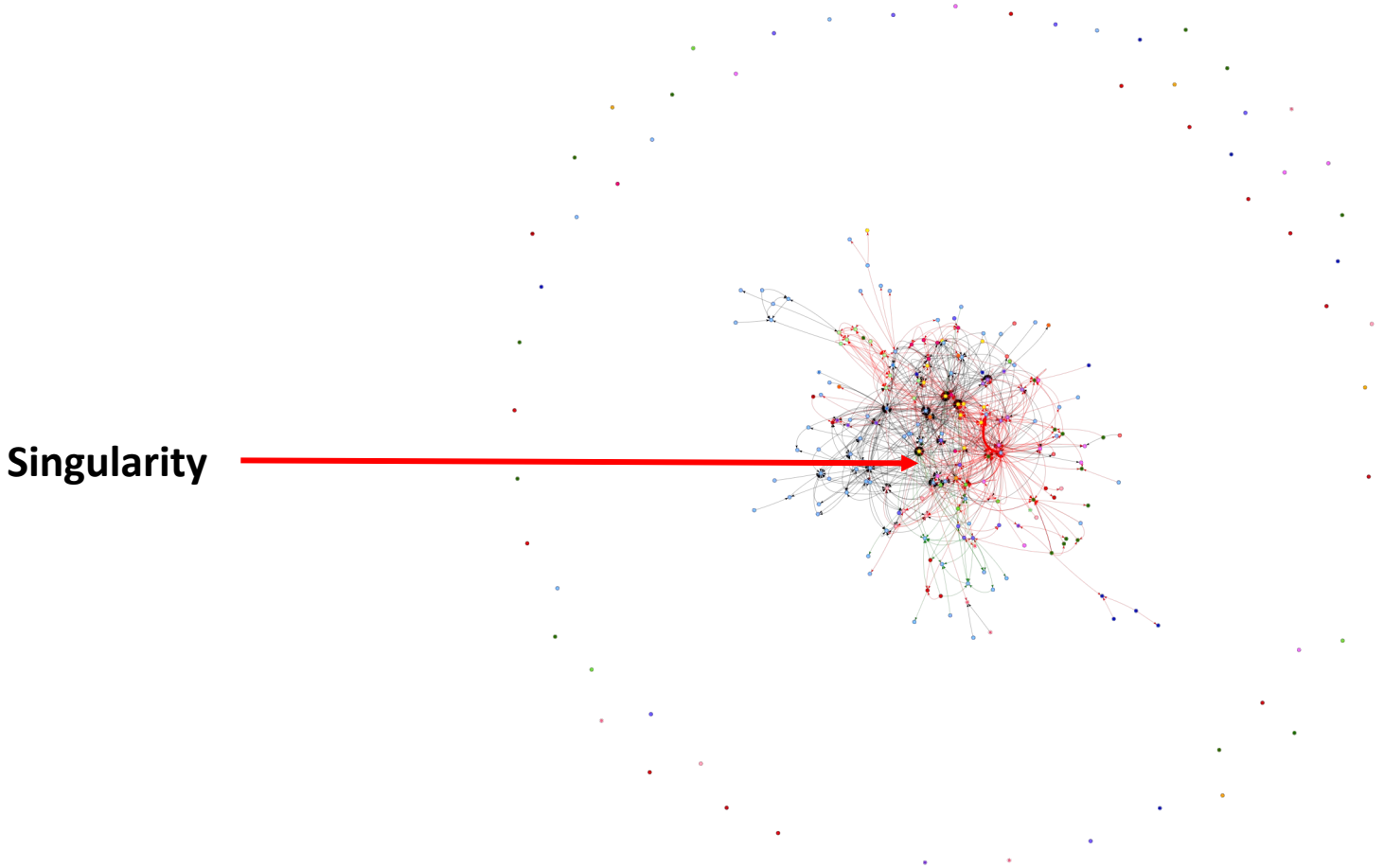
Architecture and Organization: Putting Everything Together

Event Horizon

- Repos with smaller dependencies
- Mostly owned by the same team
- Small propagation of problems
- The problems can fixed with small effort



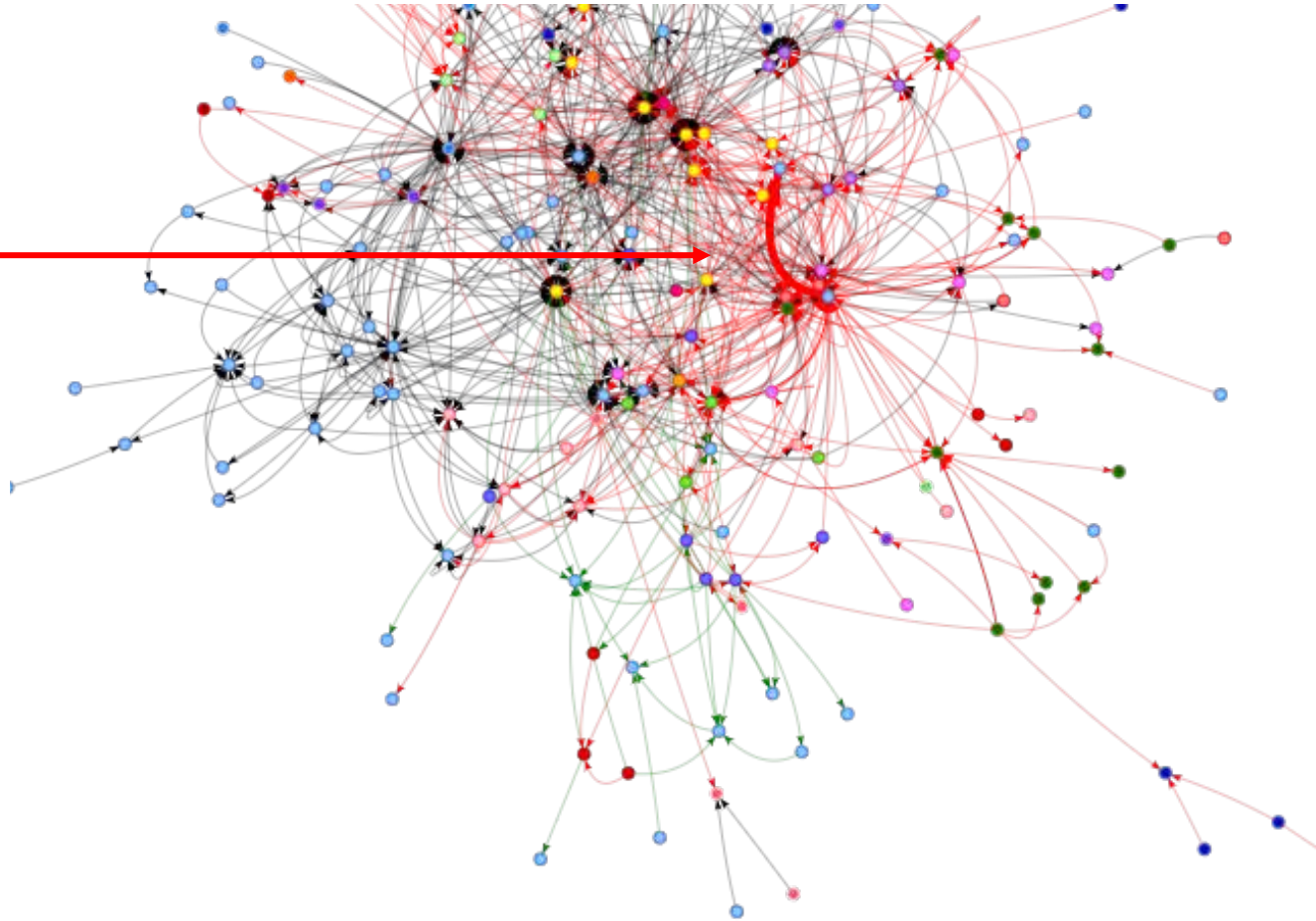
Architecture and Organization: Putting Everything Together



Architecture and Organization: Putting Everything Together

Singularity

- Dependency magnets
  - Regardless of the type
- High propagation of problems
- Problems need much attention



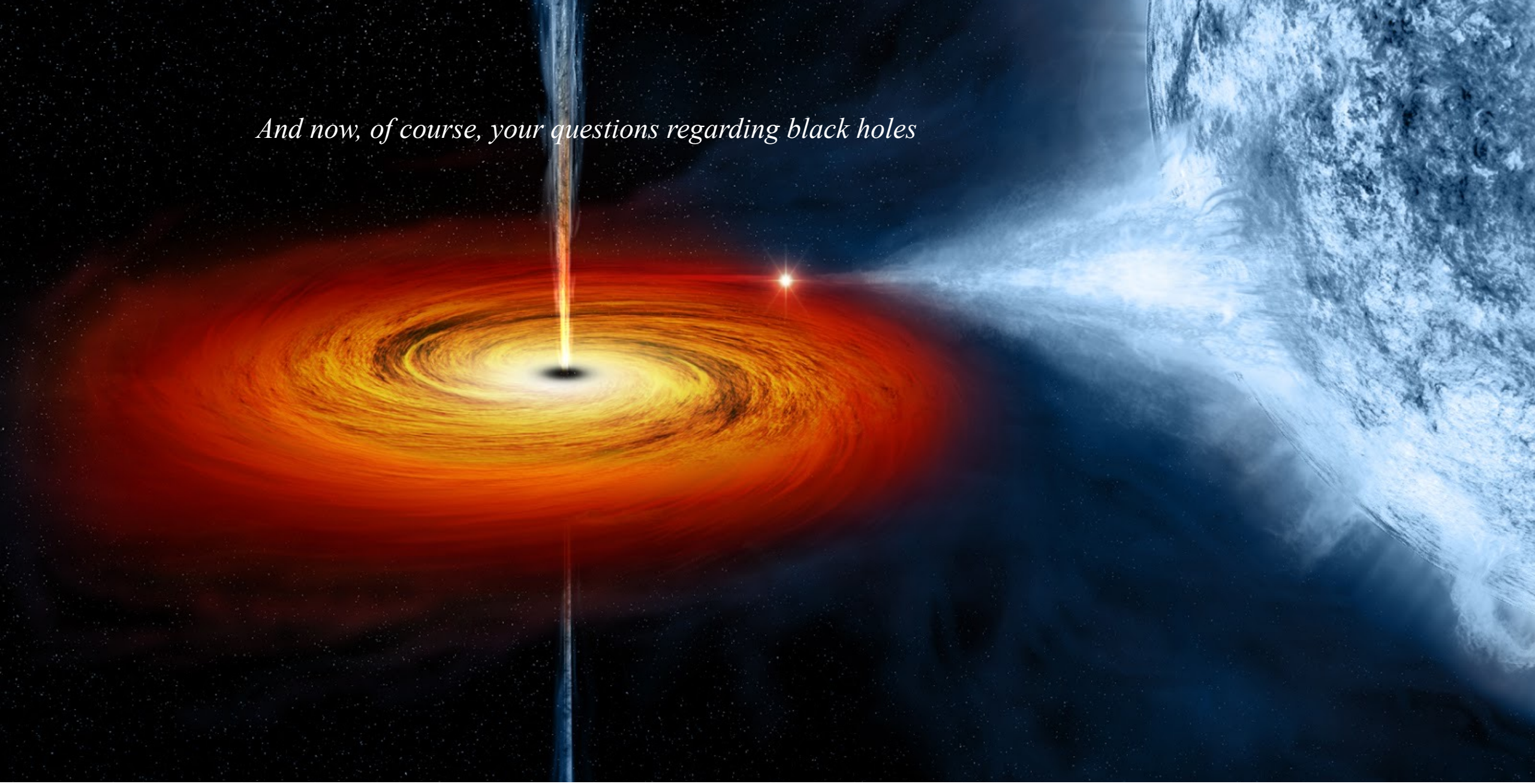
Architecture & Organization - Scenarios

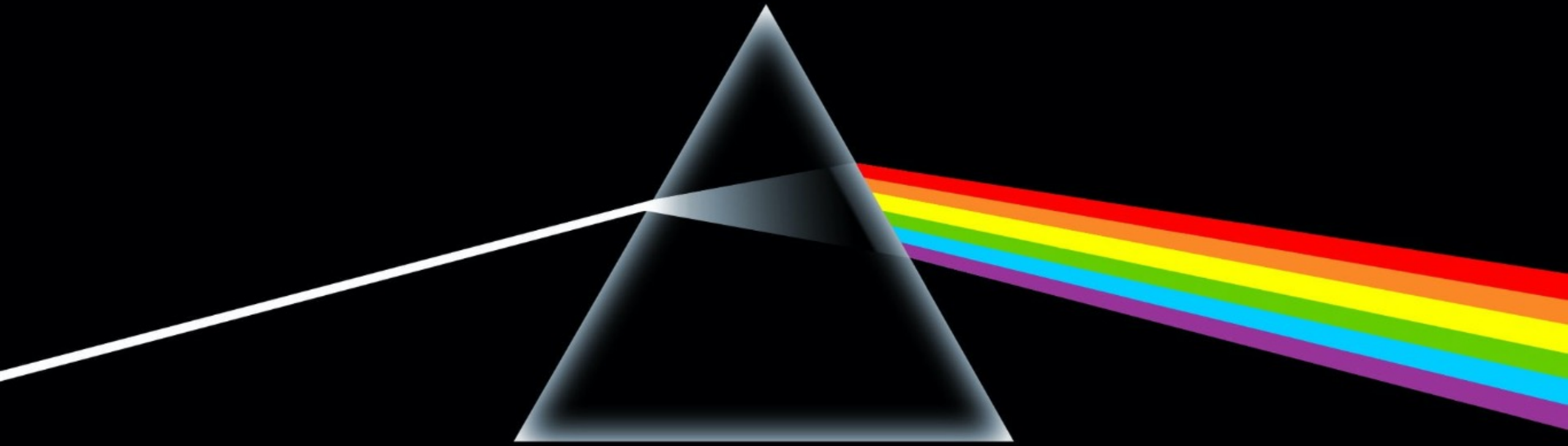
- **Scenario A:** Technical Dependency between A and B + Organizational Dependency (Jira Tickets)
  - **Problematic case:** If there are ownership problems, we (might) add overhead, there might be a third team involved
- **Scenario B:** Technical Dependency between Repo A and Repo B - No Organizational Dependency (Shared Jira Tickets)
- **Scenario C:** No Technical Dependency between Repo A and Repo B - But Organizational Dependency (Shared Jira Tickets)
  - **Problematic case:** If the repos are owned by different teams we are introducing communication overhead

So what?

So what?

- Prioritize TD repayment in Repos in Singularity Area
  - Higher number of Technical or Task Dependencies
- Analyze whether missalignment explains lead time, productivity…


And now, of course, your questions regarding black holes



The uncharted side of your data

Important things you can learn from it

Javier Gonzalez Huerta
Associate Professor
Software Engineering Department, Blekinge Tekniska Högskola, Sweden

SERL Sweden
LEADING SOFTWARE ENGINEERING

February 16th, 2022