Objectives

- Software Engineering at Google

1

Review: Collaboration Tools

- How do issue tracking, version control, and scrum boards fit together?

- Why is Coven the best-idea-*ever*-that-never-got-widely-adopted?

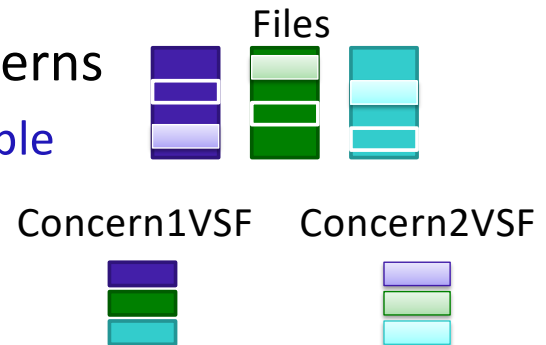- What are the tradeoffs in all-in-one solutions vs separate, integrated solutions?

2

Review: Coven: Fragments + Virtual Source Files

- Different developers can look at different parts of files
- Principle: Separation of Concerns
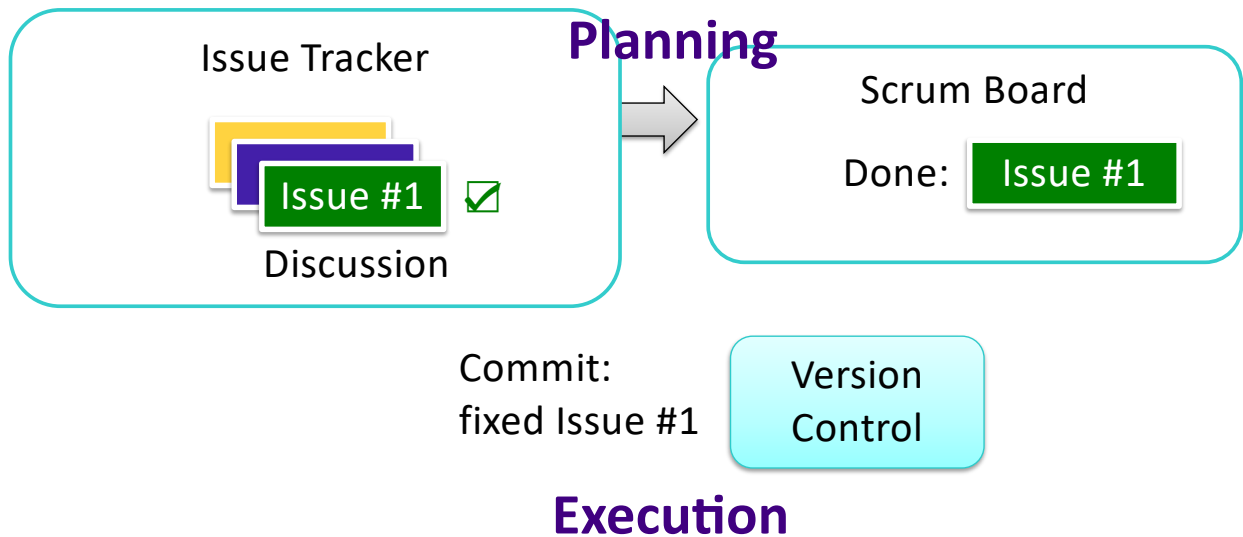 - ➢ A file can be made up of multiple concerns

Files

Concern1VSF   Concern2VSF

Mar 2, 2022   Sprenkle - CSCI397

3

Putting the Pieces Together

Planning

Issue Tracker

Issue #1 ☑

Discussion

Scrum Board

Done: Issue #1

Commit:
fixed Issue #1

Version
Control

Execution

Mar 2, 2022   Sprenkle - CSCI397

4

Tradeoffs

All-in-One Solutions

- Convenient!
  - Updates won't break individual pieces
- Its components might not (individually) be the best
- Do you need all of these

Integrating Solutions

- Separation of Concerns/Single Responsibility Principle
  - Each component is really good
  - Can easily* switch between components
- Integrations can be tricky
- Dependence on multiple things
  - Will upgrades break integrations?

5

How Should You Choose?

- Consider costs:
  - Purchasing software/service
  - Hosting software/service
  - Maintenance (number of people needed, upgrading, integrations)
  - Ease of use (in general, integrations)
- Do you need *all* those components in the all-in-one solution?
  - Can I add and remove components in the all-in-one?

6

Collaboration Tools Summary

- There are many different collaboration tools
  - ➢ More than what we've covered
- Find the right tools to help your team work the best together

7

SOFTWARE ENGINEERING AT GOOGLE

8

Discussion

- Pick your favorite point from the talk
  - ➢ Why was it your favorite?
  - ➢ What did you learn from it?
- How is SE at Google different from SE for you?
- What terms were introduced to you?

Mar 4, 2022   Sprenkle - CSCI397

9

Google's Concerns

- Availability
- How many bugs can be in your release?
- How many features should you support?
- Change over time

Mar 4, 2022   Sprenkle - CSCI397

10

9s of Availability

- Percentage of time a network component or service is accessible to a user in a given period, usually defined as a year

# of Nines	Unavailable Time in a Year
3 (99.9%)	~9 hours
4 (99.99%)	~52 minutes
5 (99.999%)	~5 minutes

Mar 4, 2022   Sprenkle - CSCI397

11

9s of Availability

Patrick Reynolds

piki

🔨 Computering

Nine fives of availability.

Mar 4, 2022   Sprenkle - CSCI397

12

"Defect Neutral"

- Question: Hardware is imperfect, OS are imperfect so you will never have a perfect software.
  Answer: True. The chapters on releasing in the book talk a lot about there never being a perfect release, it's more interesting to think of it as "defect neutral" or "value positive."
-

13

What does Defect Neutral Mean?

- Doesn't appear to be a "thing"
- Working definition: Few enough bugs that the software is worth releasing to users.
  - ➤ That's definitely context dependent!
  - ➤ Braking-control software in a car needs to be pretty close to defect-free, while lots of websites are worth publishing even if half their pages are broken.

14

What Does Defect Neutral Mean?

- Revised definition: A lack of bugs that would prevent a release or be more important than new feature development
  - ➢ Zero net *new* bugs

Sprenkle - CSCI397

15

Technical Debt

- Result of prioritizing speedy delivery over perfect code
  - ➢ Need to refactor/fix (e.g., efficiency) later

Sprenkle - CSCI397

16

How Many Bugs Can (Should?) Be in Your Backlog?

17

Prioritizing Work

- What should we work on next, bug fixes or new features?
- Which one delivers more short-term customer value?
- Which one keeps longer-term maintenance costs (tech debt) under control?
- It's always a trade-off, and the absolute number of bugs in the bug tracker isn't generally the right metric

18

How Many Features Should You Support?

Miscellaneous

- Trunk-based Development
- Upgrading
  - ➤ Elastic Search
    - 7.13 → 7.15 – broke my code
    - Needed 7.16 for security vulnerability fix

Google SWE "Secrets"

It's programming if "clever" is a compliment.

It's software engineering if "clever" is an accusation.

21