Copilot Workspace Informal

Don Syme
GitHub Next

What is GitHub Next?

What

An applied R&D group attached to GitHub, reports to Thomas

Mission

Transform the practice of software development

Mode of Operation

***Build**, **Release**, **Learn**, **Co-operate**.*

Who

~15 applied LLM/ML experts (many ex-Copilot), UX experts, CS experts

Why this is the right way to run innovative applied R&D

Operates at the Goldilocks distance!

Copilot => Copilot Workspace

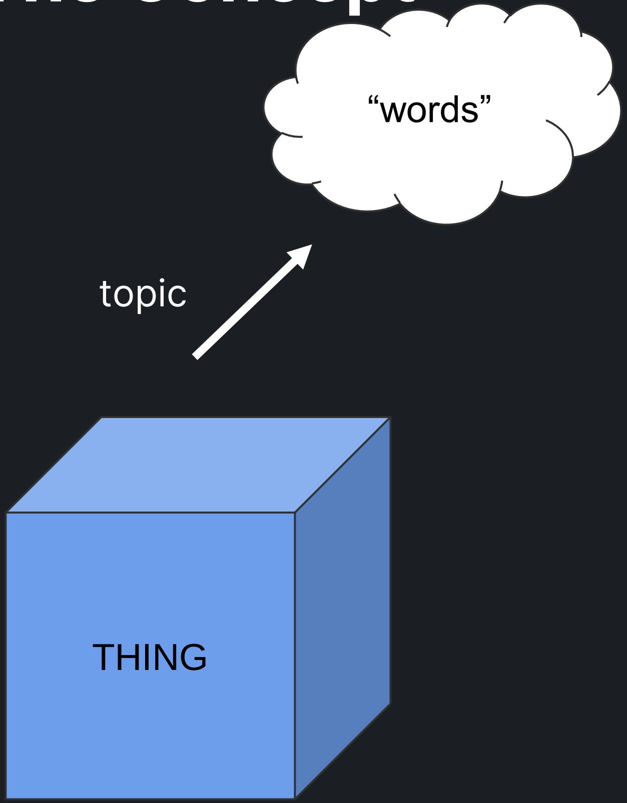
- The original GitHub Copilot completes code in your editor
  - You have to accept small bits
  - You have to move your cursor around
  - You still have to type quite a lot
  - Copilot has to infer what you are trying to do

- Can we switch POV to **whole task, whole change, whole repository**
  - While keeping the key learning of Copilot: *repeated AI-user alignment*

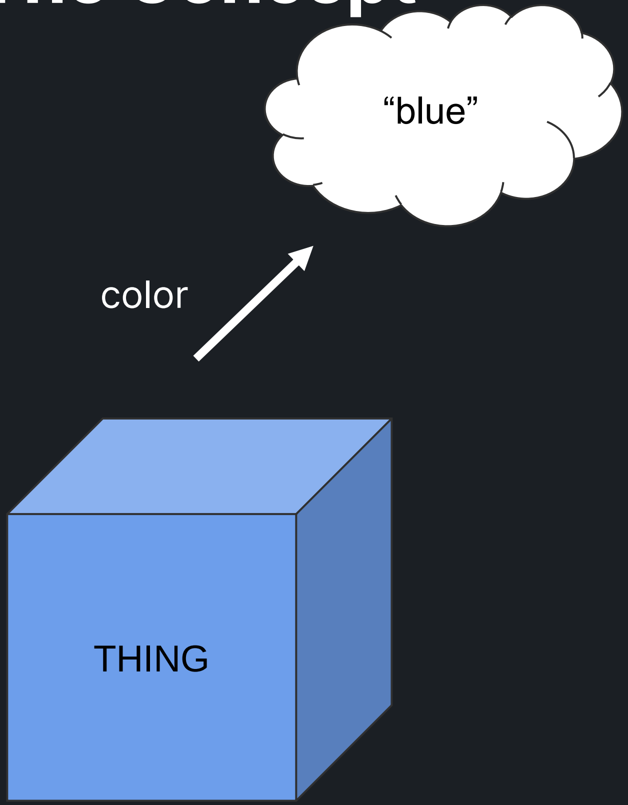
Completions => Tasks

- Consider a task that a developer might work on
  - “Add a refresh button to the main page”
  - “Create end-to-end tests using Playwright”
  - “Set up continuous integration using GitHub Actions”
  - “Set up production resources in Azure using Terraform”
- Copilot Workspace operates at this ***task*** granularity rather than completions

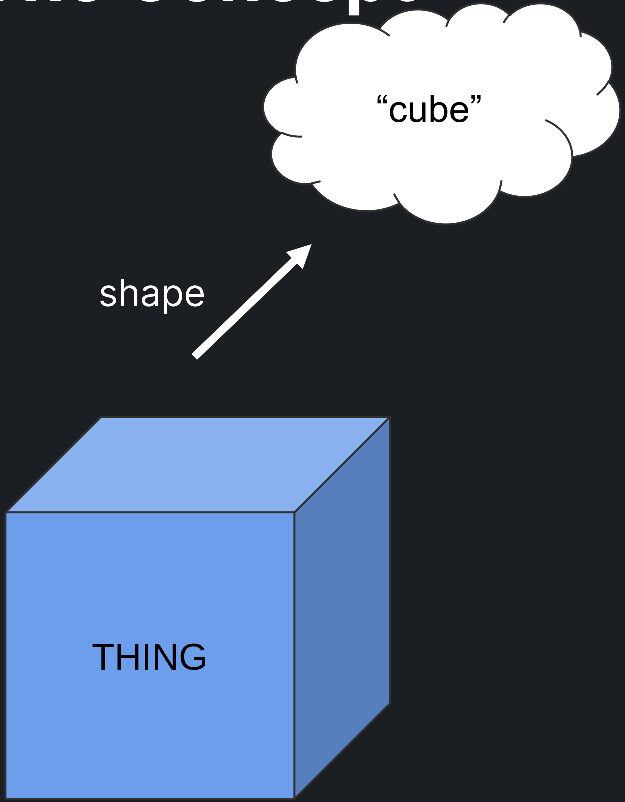
The Concept



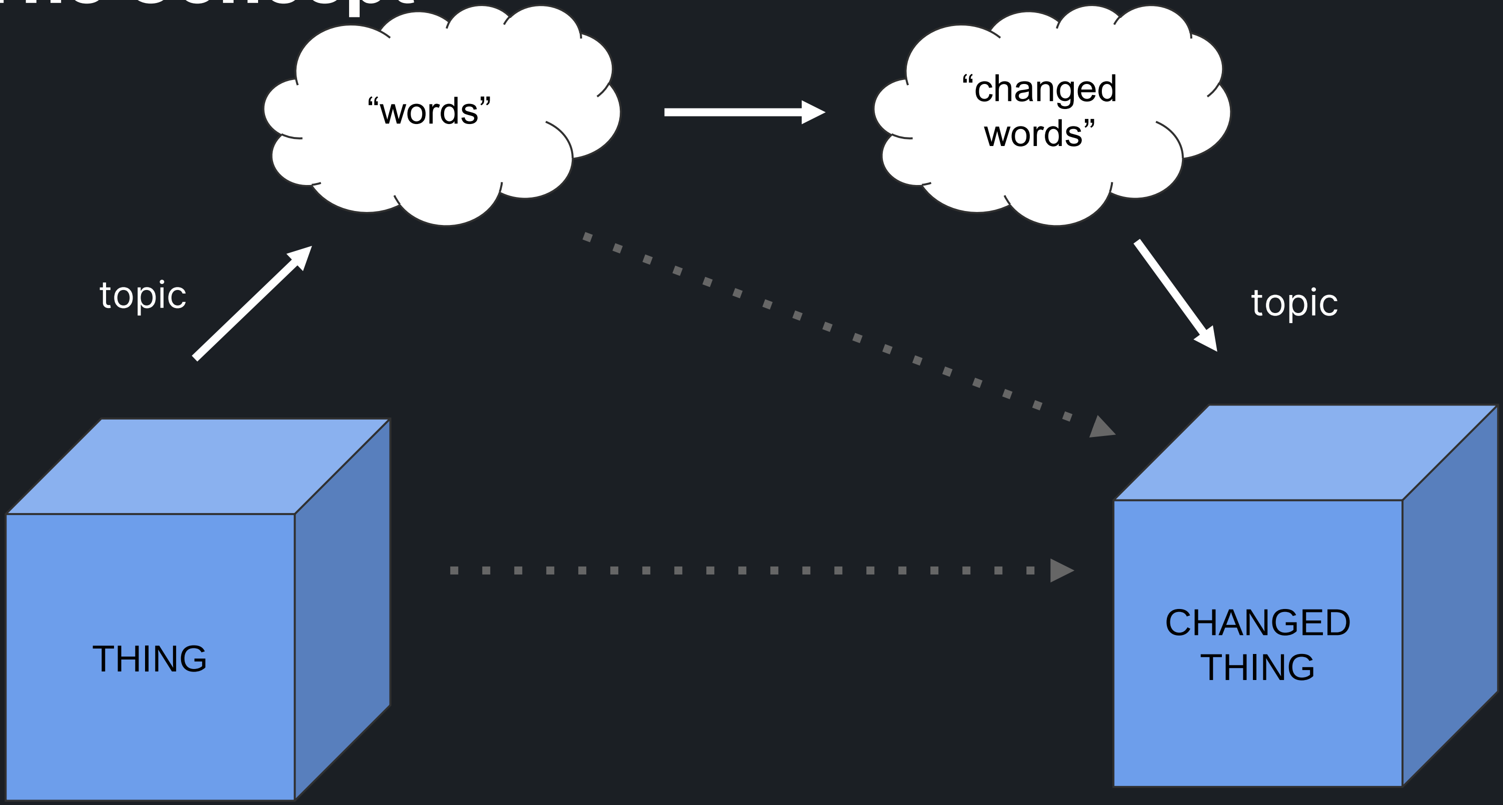
The Concept



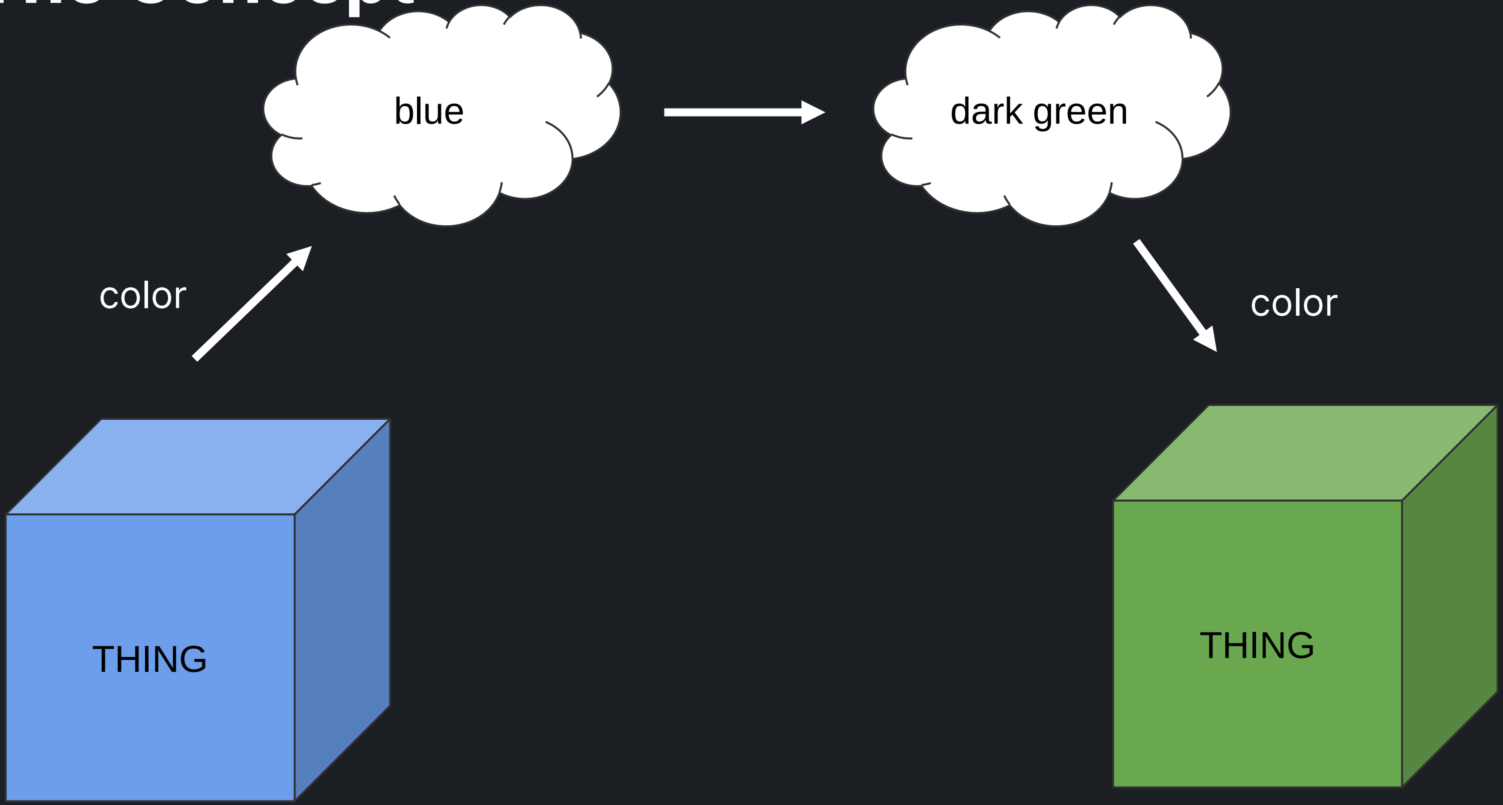
The Concept



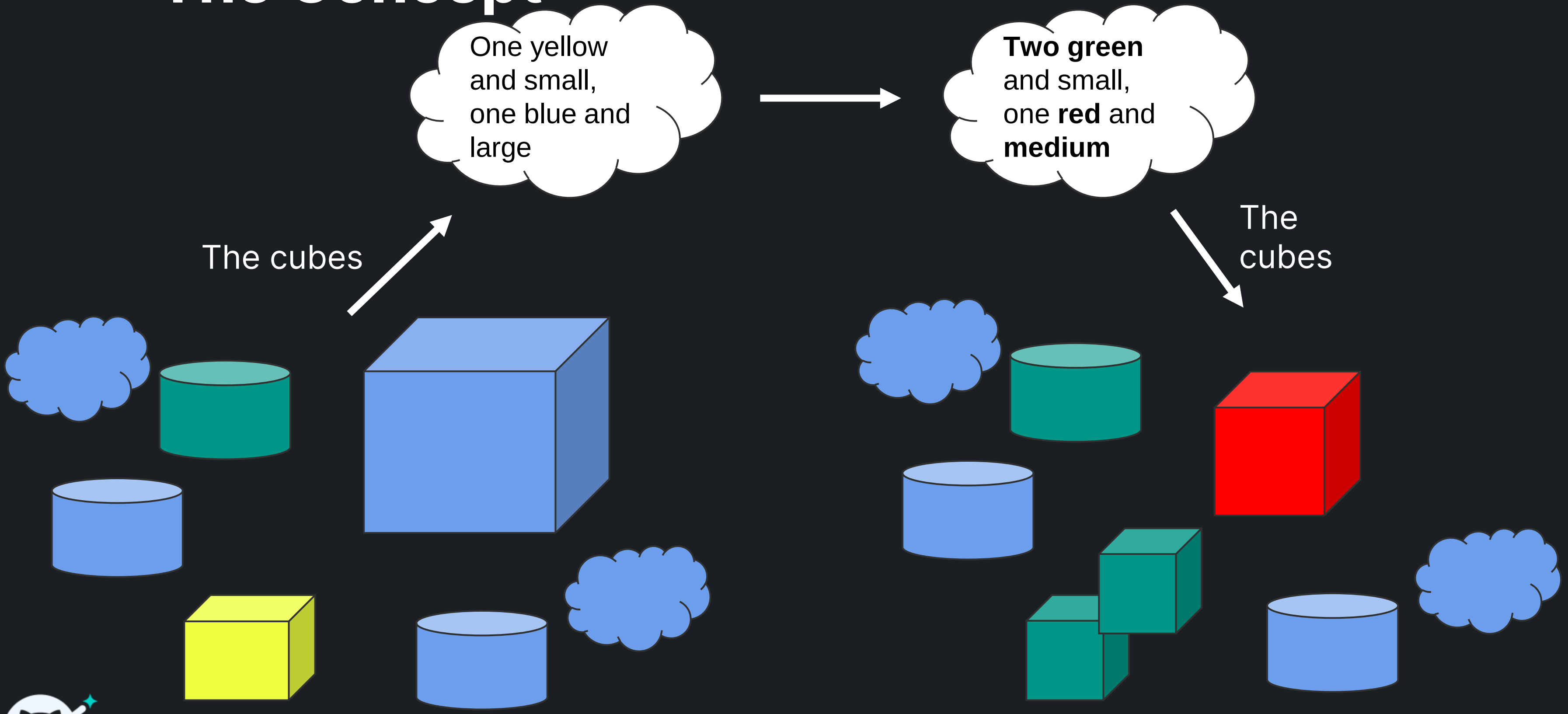
The Concept



The Concept



The Concept



One yellow and small, one blue and large

Two green and small, one **red** and **medium**

The cubes

The cubes

Principles

- Change is described in **language**
- **Specifications** are **ephemeral**
- There are **many possible specifications**, one for each topic
- Embrace the **fluidity** and **power** of **words**
- The model both **hides** and **fills in** the details

Rationale

- Edit > Create-from-scratch
  - (also, create-from-scratch is a special case of editing)
- People lack the **terms of discourse** to describe the change they want
  - The **topic of focus** pins us down to a particular slice
  - The **extraction** describes the thing as it is
  - The **edit** describes how we want it to be

Notes

- Summarization the inverse of generation
  - “Extract” is summarization w.r.t. topic
  - “Apply” is generation-of-change w.r.t. topic
- “Extract” seems independently useful
  - Has similarity with Chat
- “Apply” utilises hallucination well!
  - “Fill in the details, I’ll check what you do”

Demo

The Fundamental Theorem of Chat

For any *context* and *question* there is a most-natural* *answer*

We let the AI work this out

**(in reality a whole manifold of them)*

The Fundamental Theorem of Change

For any *thing* and *change-intent*, there is a most-natural* *changed-thing*

We let the AI work this out

**(in reality a whole manifold of them)*

Foundational techniques and challenges

- Extracting initial change intent
- Iterative lowering of change intent
- Partial repository selection
- Partial repository rewriting

Foundational techniques and challenges

- **Extracting initial change intent**
    - Multiple possible ways to extract change intent
    - User and AI in co-operation, iteratively
- **Iterative lowering of change intent**
- **Partial repository selection**
- **Partial repository rewriting**

Foundational techniques and challenges

- **Extracting initial change intent**
- **Iterative lowering of change intent**
    - Change Intent → Plan → Changes
    - Iteration and clarification
    - User and AI in co-operation
- **Partial repository selection**
- **Partial repository rewriting**

Foundational techniques and challenges

- Extracting initial change intent
- Iterative lowering of change intent
- Partial repository selection
  - Crucial for almost every AI feature
  - Based on topic, plan etc, manual selection as backup
  - Challenging to scale to truly massive repositories
- Partial repository rewriting

Foundational techniques and challenges

- **Extracting initial change intent**
- **Iterative lowering of change intent**
- **Partial repository selection**
- **Partial repository rewriting**
  - The key performance bottleneck, will definitely limit uses
  - Very challenging to scale
  - Currently single model invocation ~24K prompt, ~8K reply
  - Currently selects files, and rewrites some of them
  - The ideal: selectively rewrite parts of files and fold back

Towards a Semantic Workspace: Aspirations

- Allow engagement with software at the conceptual, semantic level
  - Iterate and lower to concrete changes
  - Ideally allow continual iteration with a dev/test/run loop
  - Ideally multiplayer
  - Ideally automatically scalable
- Attempt to escape the gravity of free-form chat
  - A definite object of study (a repository)
  - A definite outcome (lowered change intent)
  - If we allow chat, it will focus on change intent and planning

Towards a Semantic Workspace: Key User Tasks

Experienced developers on github.com sketching things that are unfamiliar or routine and involve multiple changes

- “I didn’t know where to begin with this”
- Changes in unfamiliar repositories
- Feature sketching
- Issue solving
- Unit-test creation, esp legacy code
- Documentation generation

Towards a Semantic Workspace: Key Challenges

- Getting Closer to the Dev Loop
  - **Web delivery** for generality and simplicity
  - Eventually **IDE delivery** to get closer to the dev loop
  - **Issue-first** workflows
  - Utilise **codespaces** as a general runtime for dev/test loop
- Scaling
  - Automatic file selection is key but ours is weak

Give it a go!

- Send us your github handle, we'll get you added
    - Everything may/will change
- https://copilot-workspace-dev.githubnext.com/

GitHub Next

Researching the future of
software development

githubnext.com