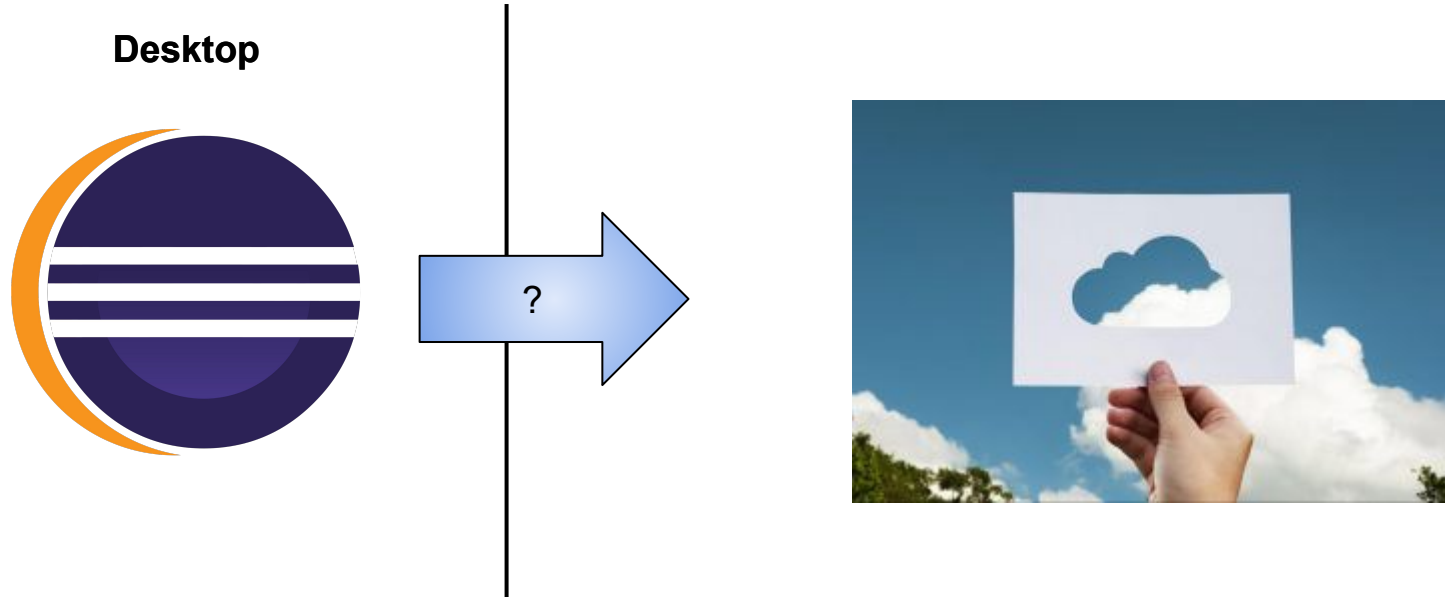


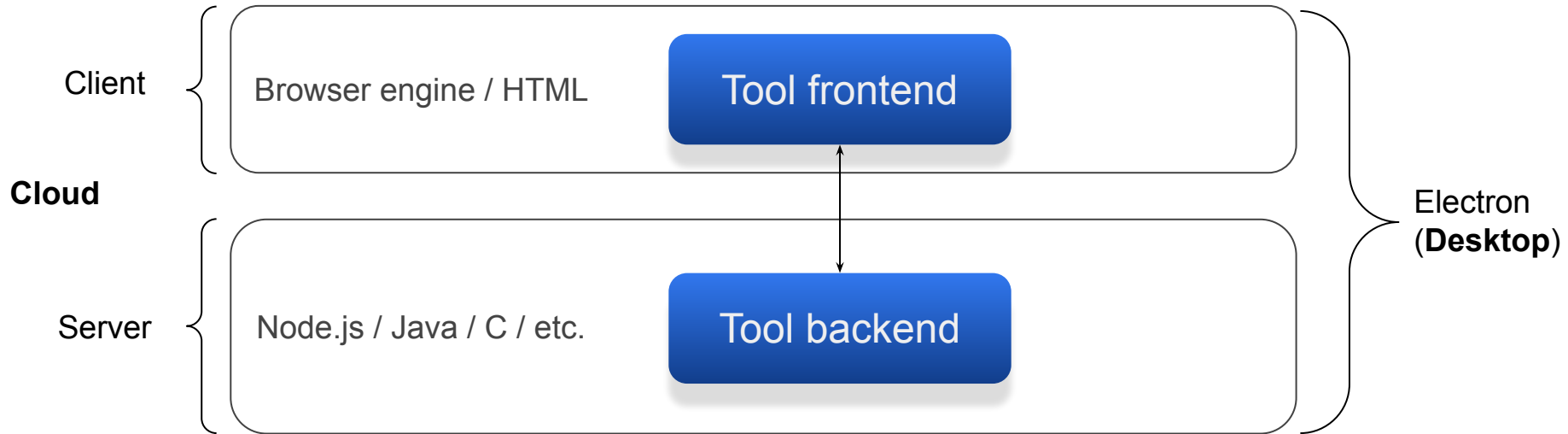
Migrating Eclipse-based Tools/Plugins to Eclipse Theia or VSCode

Jonas Helming
EclipseSource

Migrating tool to the web/cloud?



Typical architecture of a web/cloud-based tool



Desktop vs. Online/Cloud

Desktop	Online

Note that you can single-source most tools features between cloud and desktop
=> Many vendors follow a dual strategy: Web-based on the desktop first, add cloud offering as a second step

The underlying tool platform

Theia vs. VS Code (see also [this article](#))

Conceptional assessment of your Eclipse tool/plugin

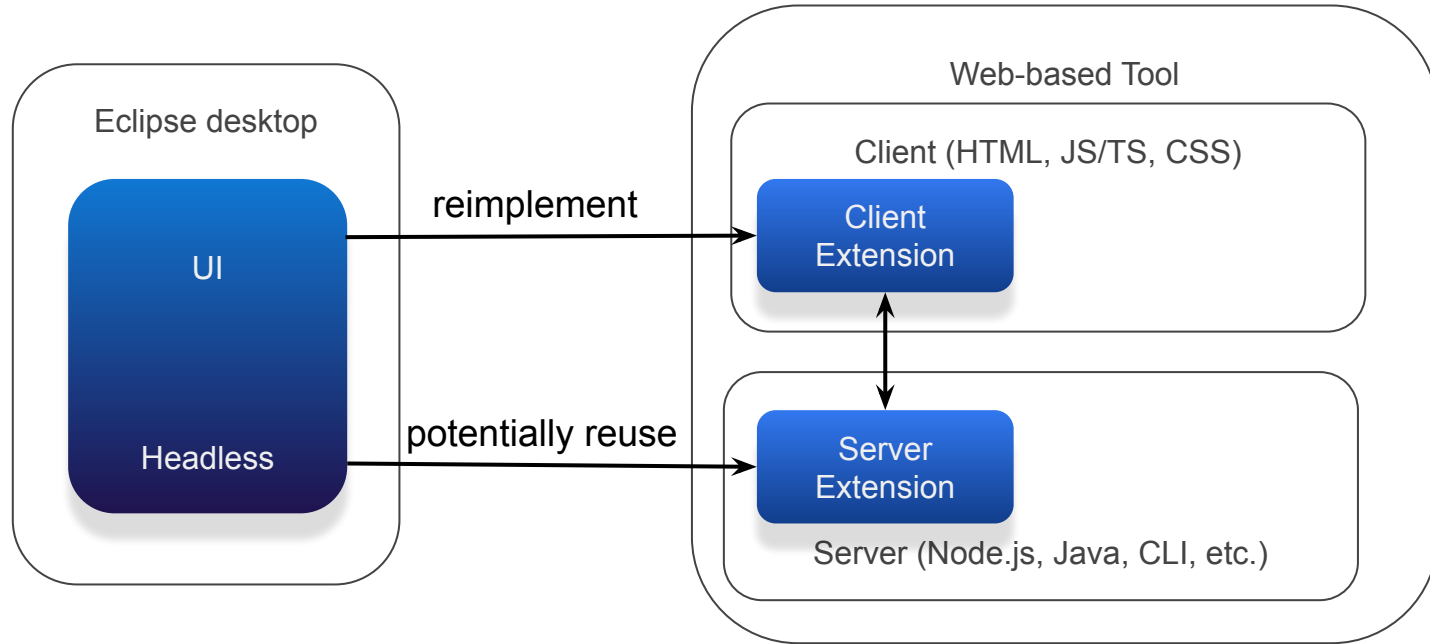
Conceptual assessment

- Redesign your workflows
- Remove unnecessary features
- Redesign your UX



Technical assessment of your Eclipse plugin

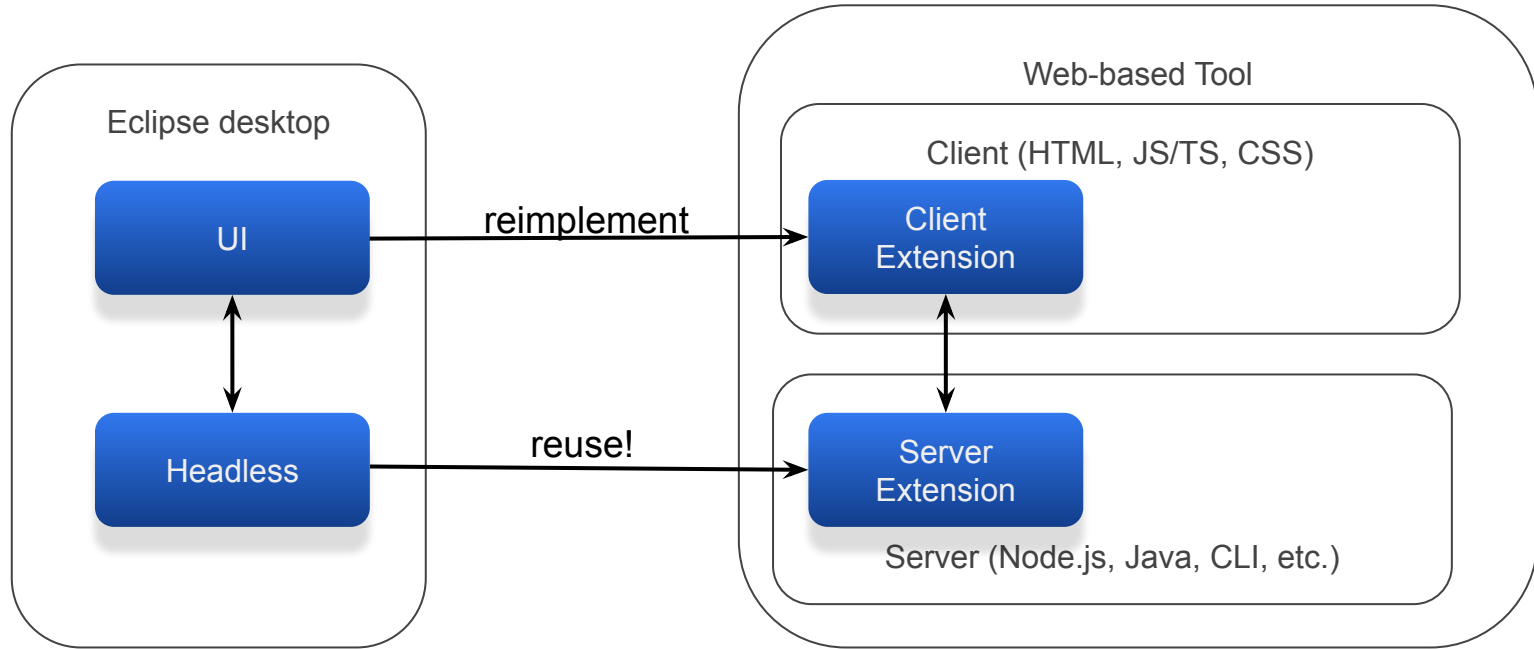
Technical assessment



Technical assessment - examples

- A code generator
- A custom DSL (e.g. with XText)
- A diagram editor
- Dedicated process / CLI
- Language Server Protocol (LSP)
- Eclipse GLSP + Custom server interfaces (e.g. REST or Sockets)

Preparing the architecture of your Eclipse plugin

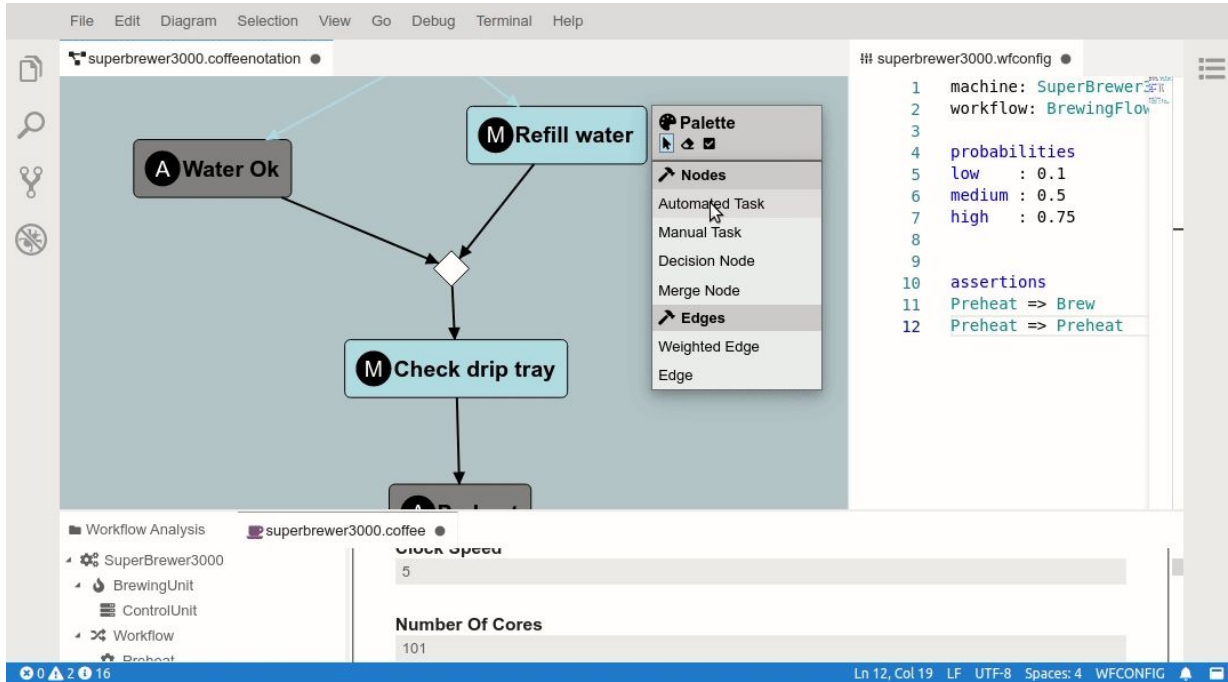


Technology selection for the web-based version

Technical assessment - examples

Eclipse desktop	Eclipse Theia / VS Code

The coffee editor



The screenshot shows the Eclipse IDE interface for the coffee editor. The main workspace displays a workflow diagram with the following nodes and connections:

- Water Ok** (Automated Task, 'A') and **Refill water** (Manual Task, 'M') both point to a central **Decision Node** (diamond shape).
- The **Decision Node** points to **Check drip tray** (Manual Task, 'M').
- Check drip tray** points to a partially visible **Automated Task** node at the bottom.

A **Palette** is open on the right, showing the following categories and items:

- Nodes**
 - Automated Task
 - Manual Task
 - Decision Node
 - Merge Node
- Edges**
 - Weighted Edge
 - Edge

The bottom panel shows the **superbrewer3000.wfconfig** editor with the following configuration:

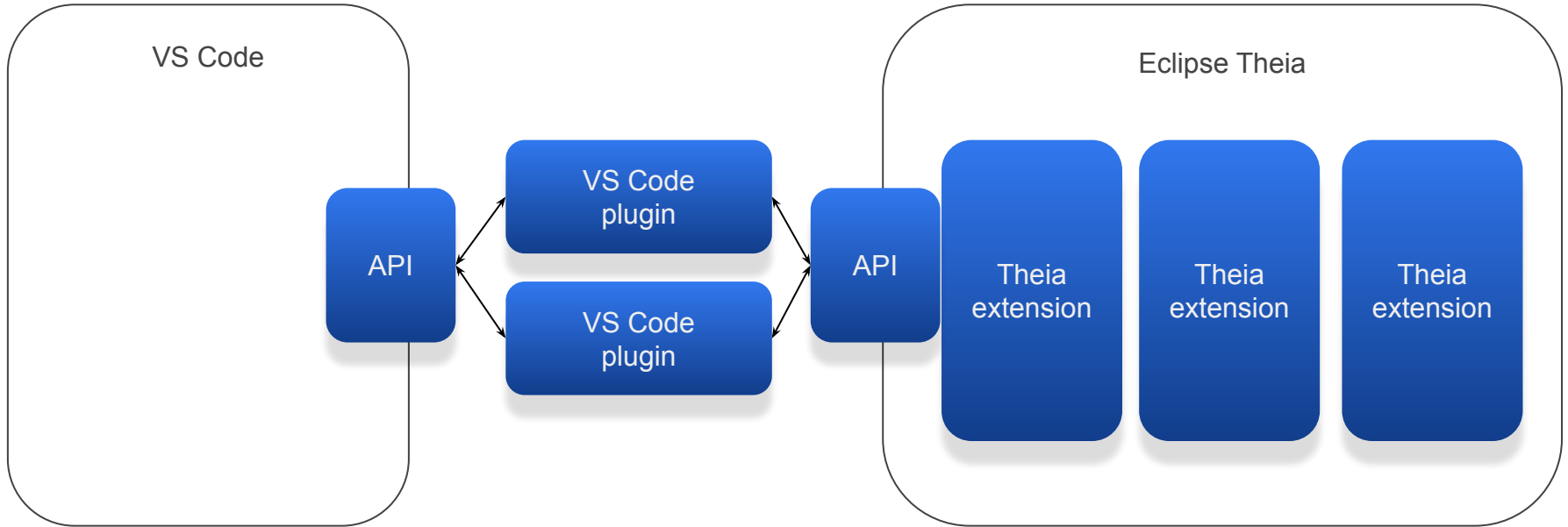
```

1 machine: SuperBrewer3000
2 workflow: BrewingFlow
3
4 probabilities
5 low : 0.1
6 medium : 0.5
7 high : 0.75
8
9
10 assertions
11 Preheat => Brew
12 Preheat => Preheat
  
```

The bottom status bar indicates the current position: Ln 12, Col 19, LF, UTF-8, Spaces: 4, WFCOFIG.

What extension mechanism to use?

VS Code extensions vs. Theia extensions



Define a Strategy

Strategies for migrating your tool

- Evaluate technology choices
- Build PoCs
- Define MVPs
- Go Iterative



Questions?

Today 17:00: Spotlight session

Wed 27.10., 18:15, [BoF - Building web-based tools with Eclipse](#)

[EclipseSource Booth](#)

Wed 27.10., 13:50, [CDT.cloud? C/C++ tooling in the web](#)

Wed 27.10., 16:10, [Model validation, diffing and more with EMF.cloud](#)

Thu 28.10., 16:10, [Papyrus UML - the first stage of a journey to the cloud](#)

Evaluate ~~the~~ Sessions

- Please help by leaving feedback on the sessions you attend!
- To rate a session, you must be registered for it in Swapcard BEFORE the talk starts.
- Swapcard will prompt you to leave feedback after the end of each session.
- You may also rate a session by clicking on the session from the “Agenda” or “My Event” buttons on the Event Home page, click on the session and look for the “Give your feedback” box.



ECLIPSE
2021 CON

Backup slides

Web-based vs. cloud-based tools

- Web-based:
 - Use web technologies, especially in the UI, such as: HTML, CSS, JavaScript/TypeScript
 - Use a “browser” for UI rendering (real browser or Electron)
 - Usually split UI and a “backend” server for the business logic, even when deployed on the desktop
- Cloud-based:
 - Backend is deployed remotely (in the cloud)
 - UI is accessed via: Browser or Desktop tool connecting to the backend
- Cloud-based tools are usually web-based, web-based tools are usually prepared for the cloud!