IDE-IT (Backend)

Eric Wahlquist, John Barcellos, David Thien

Motivation

IDEs are great with many features to speed up the development process, however...

Discoverability of IDE features is extremely poor

- Difficult to know what features exist
- Current methods don't give you features at relevant times

IDE-IT

Integrated Development Environment - Intelligent Tutorials

Eclipse plugin

- Tracks user input and provides suggestions for using IDE features

IDE-IT Back end

- Responsible for detecting user input and evaluating which IDE features the user might be interested in
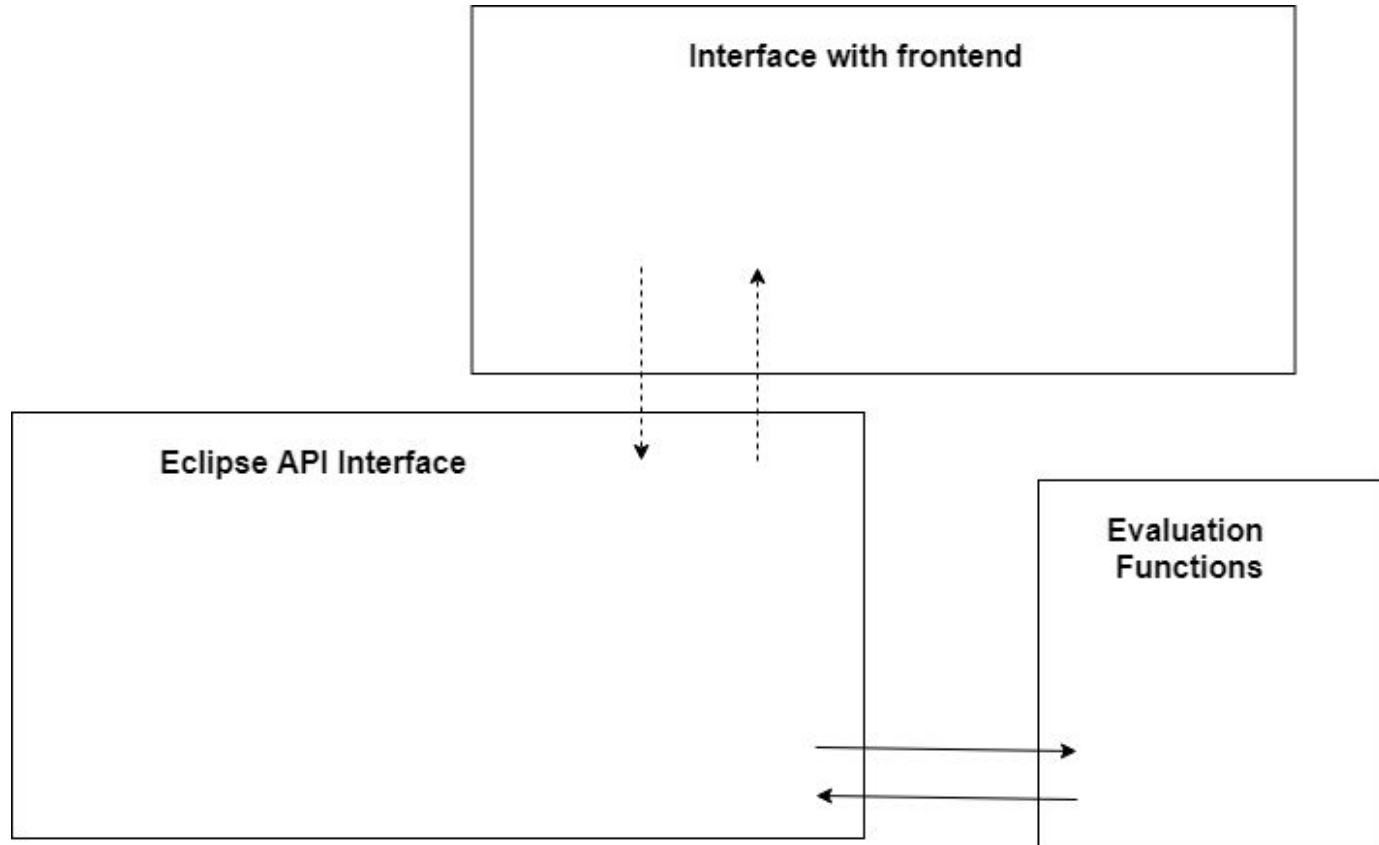
IDE-IT Vs. Previous approaches

Improving Software Developers' Fluency by Recommending Development
Environment Commands

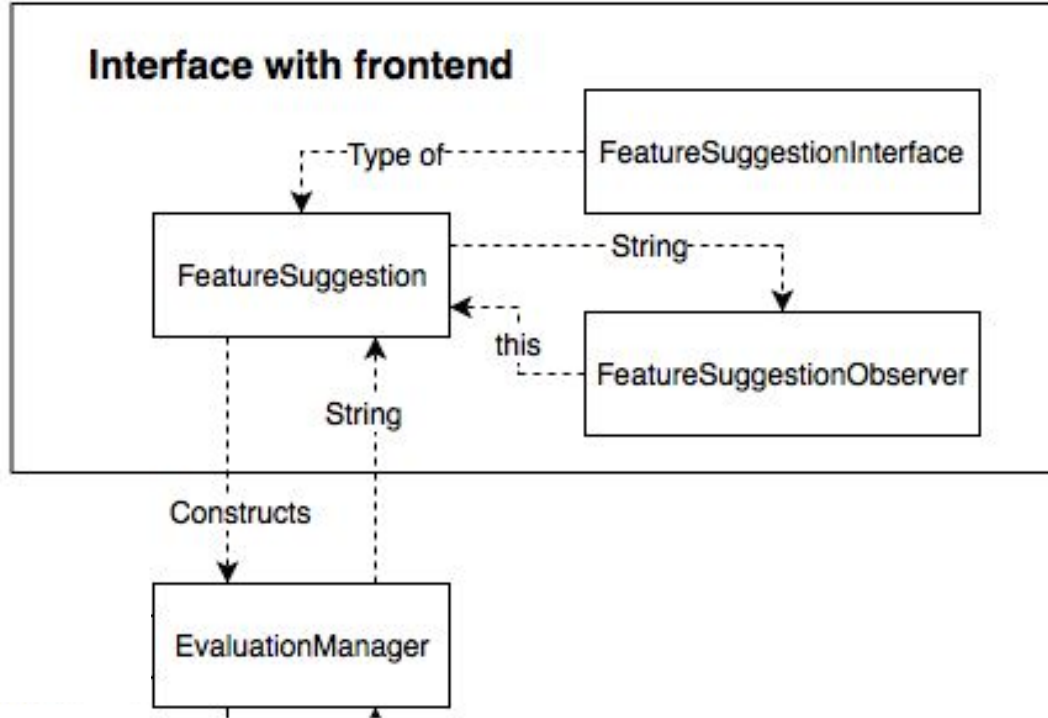-   Recommendations based on sequences of commands

Intellij Idea Feature Suggester

-   Recommendations based on changes to files
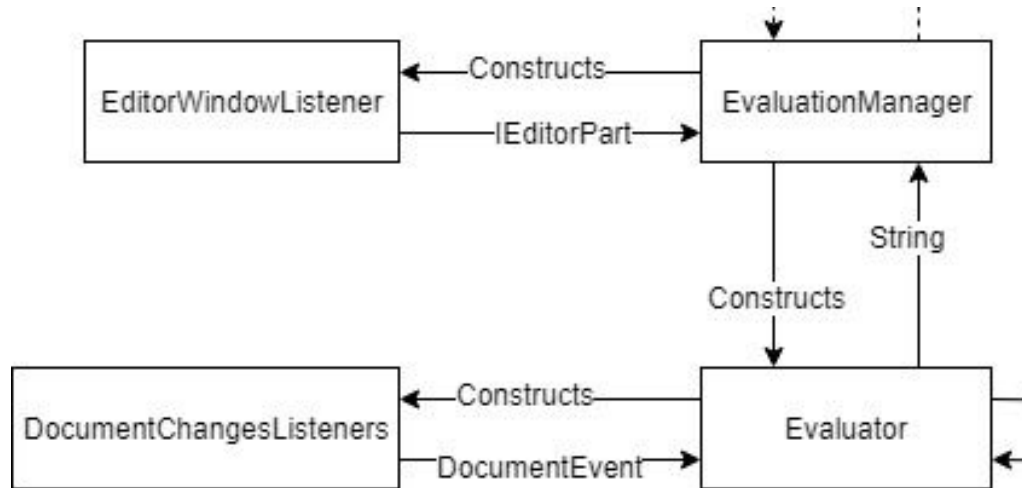-   Similar idea, but buggy and no interface for frontend
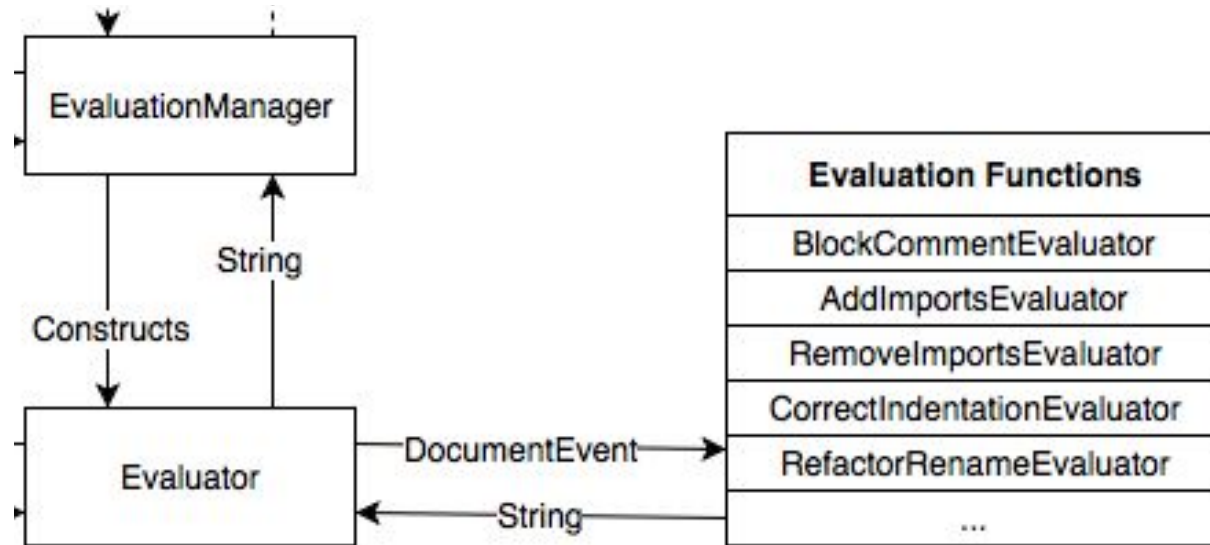
IDE-IT Backend Approach

Interface with frontend

Eclipse API Interface

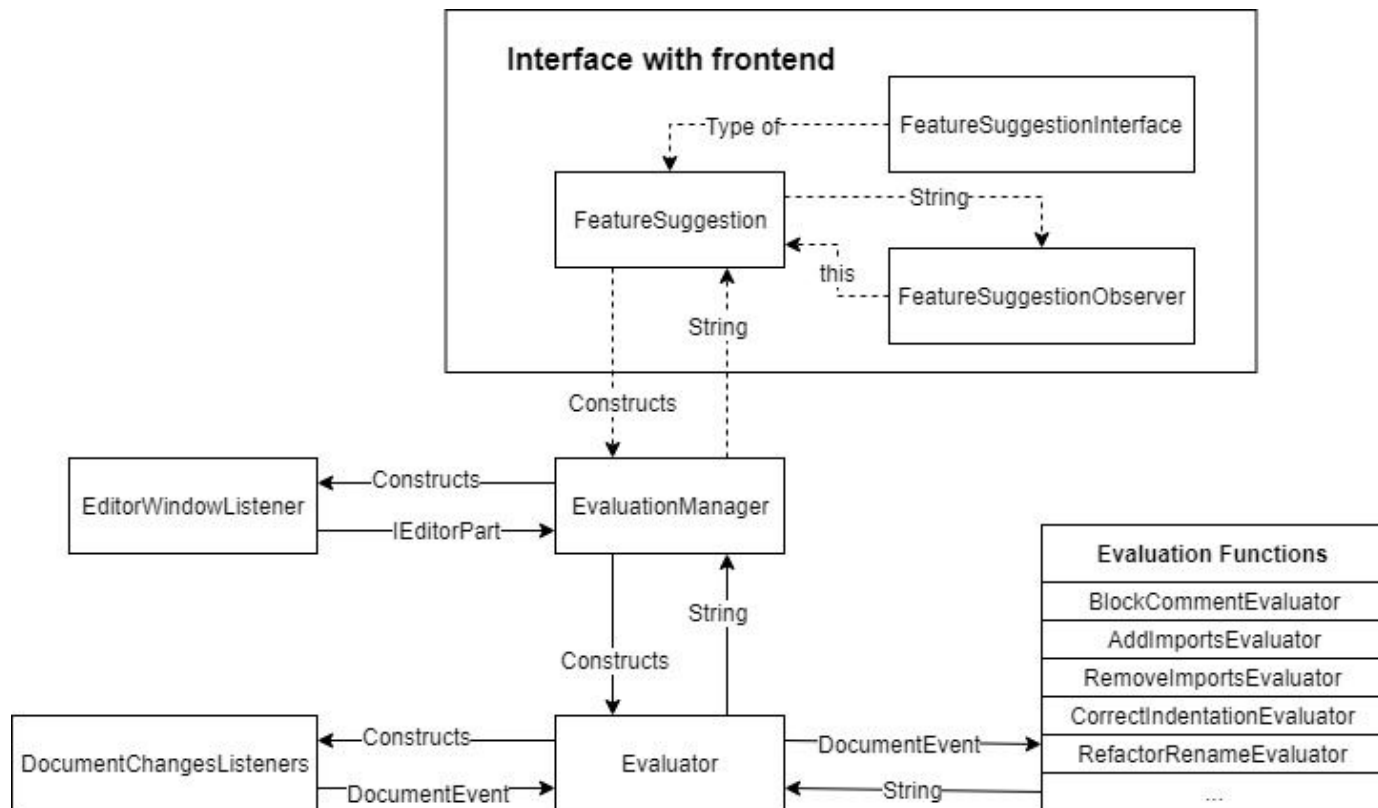Evaluation Functions

IDE-IT Backend/Frontend Interface

Eclipse API Interface

Evaluation Functions

IDE-IT Backend Approach
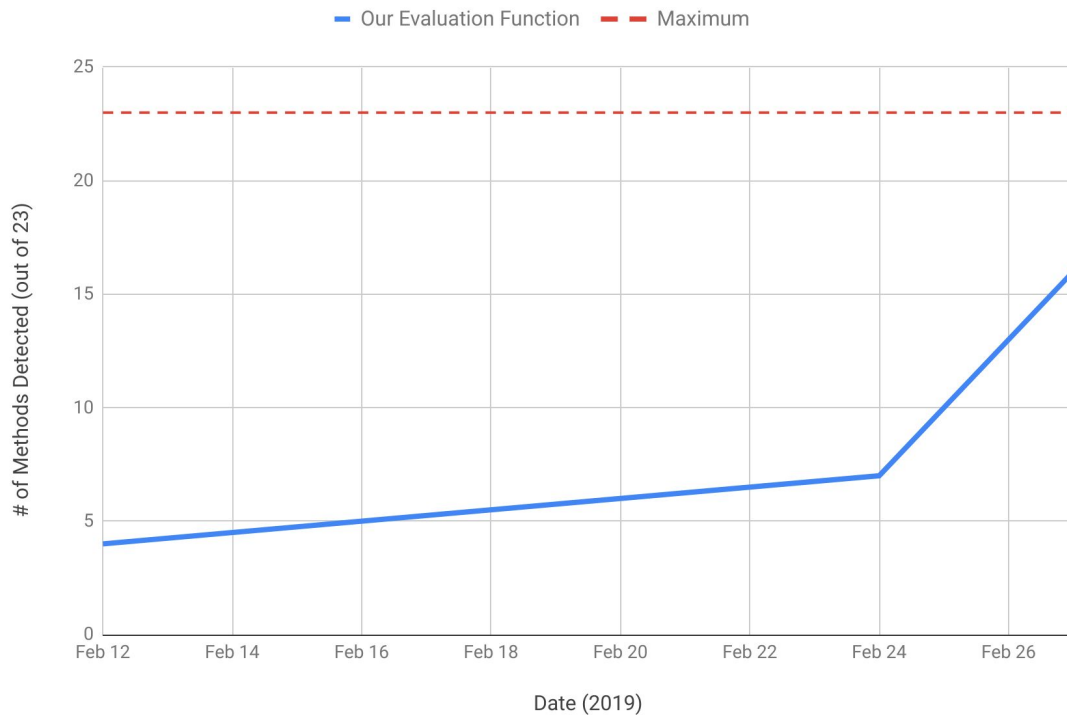
Demo

Project Success Metrics

Our focus:

- Minimize false negative evaluation triggers
- Keep false positives low, but less of a concern

Evaluate against a list of possible user actions

- Comment out lines with //, /* * */, or single lines with ctrl + /
- Comment out lines in different orderings

Initial Results

BlockCommentEvaluator Successful Activation Methods

Estimated # for 1st point

Citations

- Emerson Murphy-Hill, Rahul Jiresal, and Gail C. Murphy. 2012. Improving software developers' fluency by recommending development environment commands. In Proceedings of the ACM SIGSOFT 20th International Symposium on the Foundations of Software Engineering (FSE '12). ACM, New York, NY, USA, Article 42, 11 pages. DOI: https://doi.org/10.1145/2393596.2393645

-