

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 Commands¹

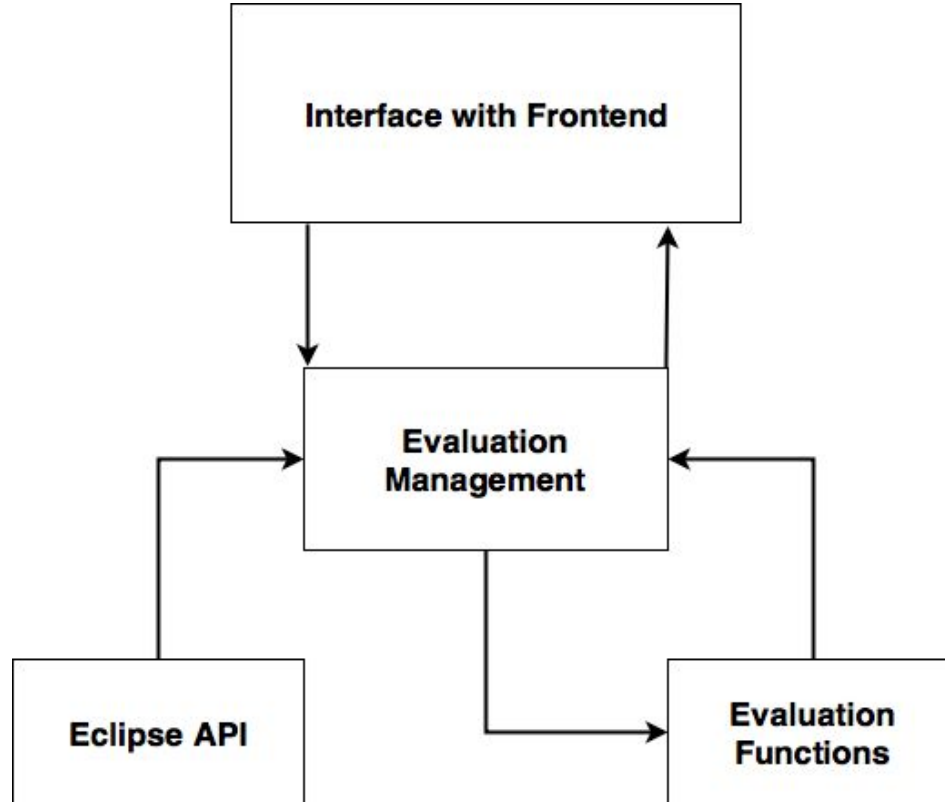
- Recommendations based on sequences of commands
- Doesn't work for general development process

Intellij Idea Feature Suggester

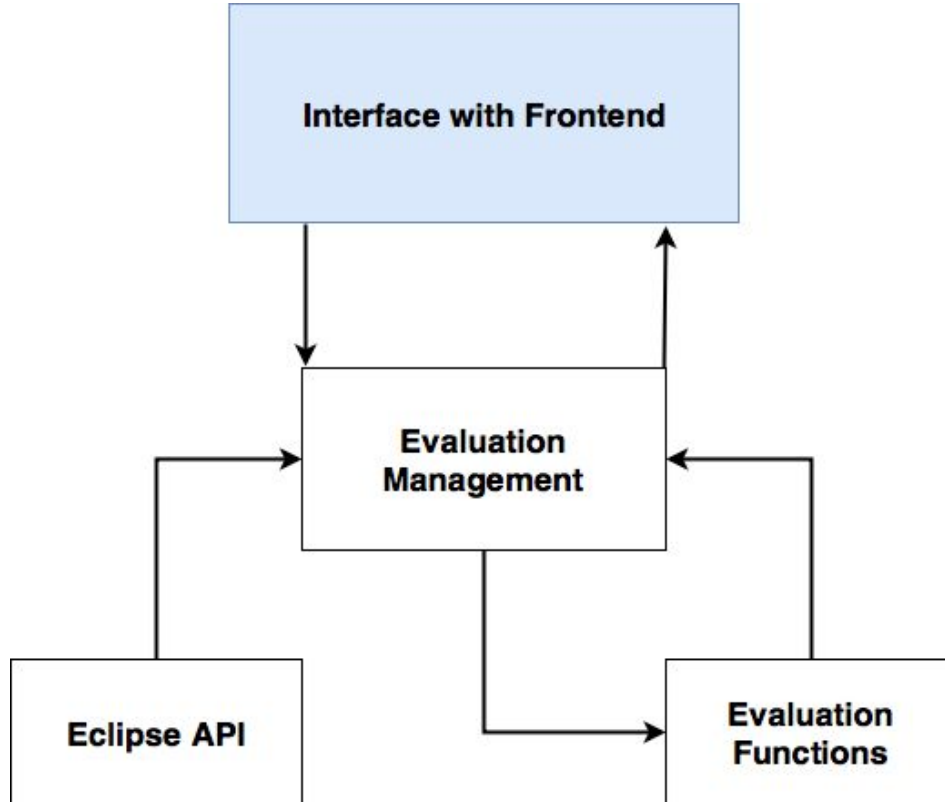
- Similar idea to IDE-IT
- Does not support an arbitrary frontend
- Not available for Eclipse

(1) 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>

IDE-IT Backend Approach

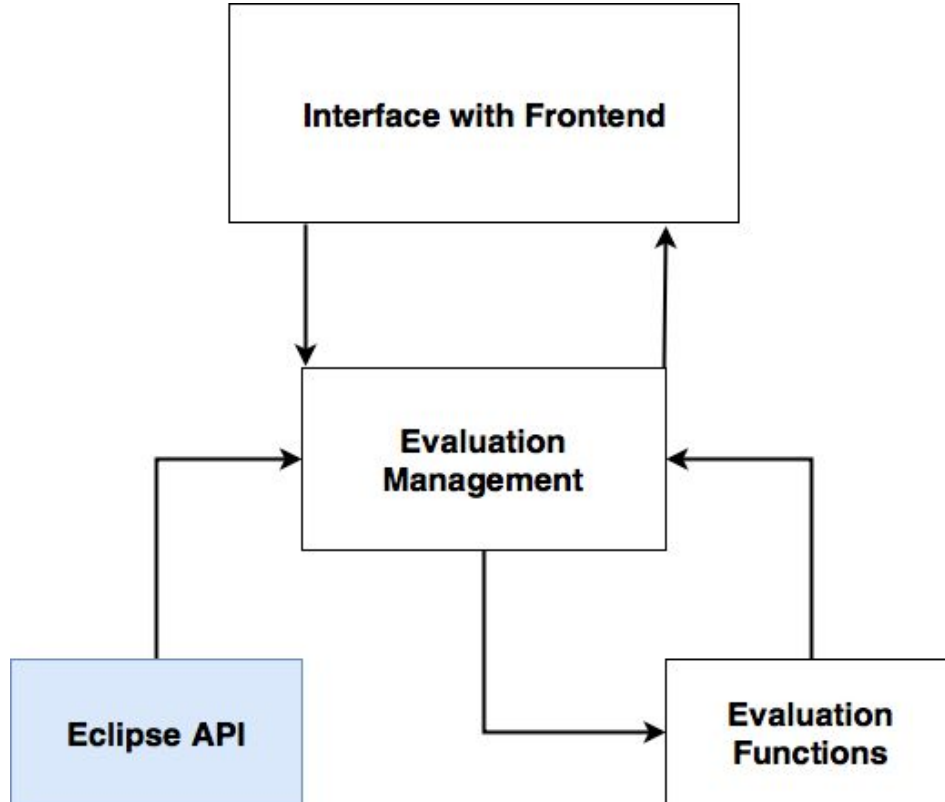


IDE-IT Backend Approach



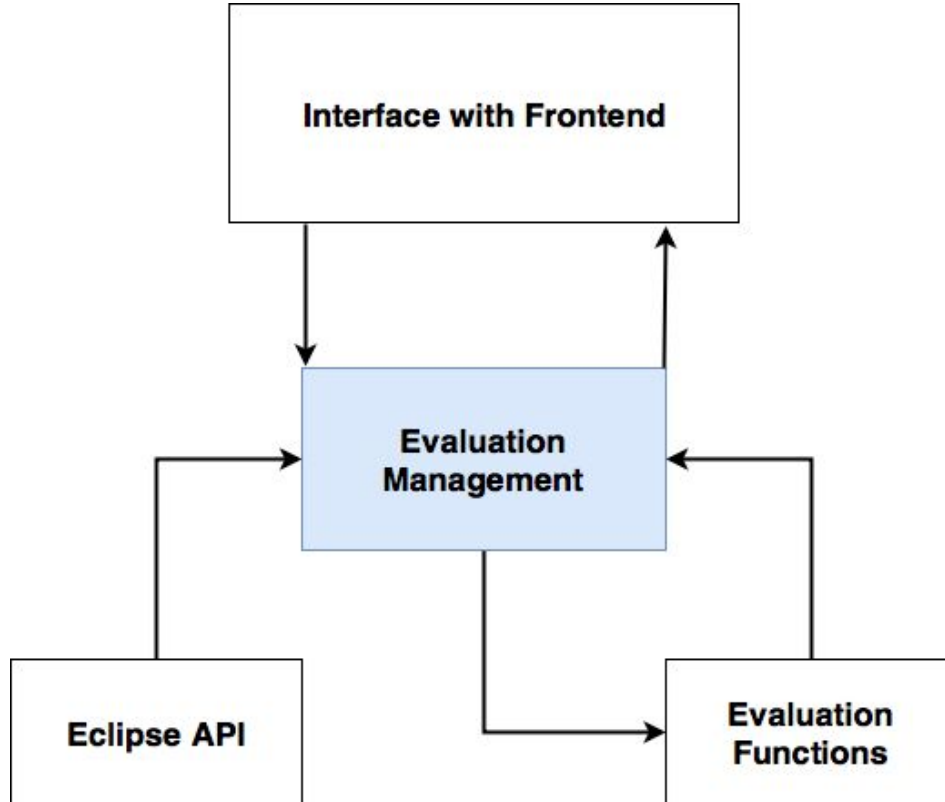
- Allows arbitrary frontend client to use backend functionality
- Frontend client creates an observer, registers it with the interface
- When a feature is suggested, frontend notified via the observer

IDE-IT Backend Approach



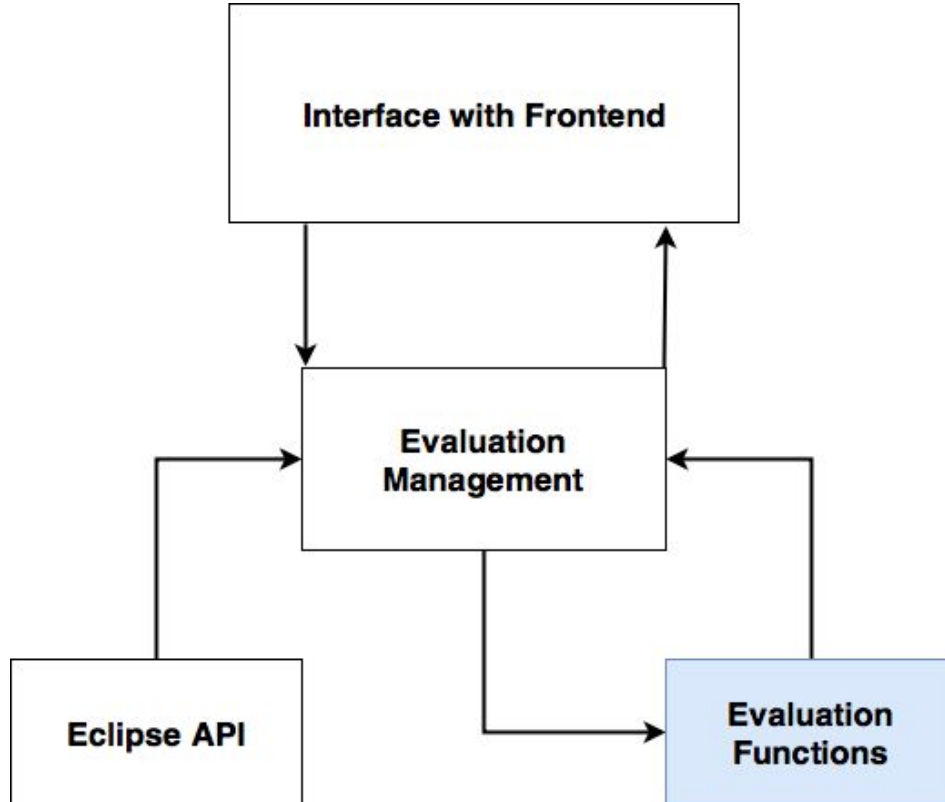
- Listens for:
 - New editor windows
 - Document changes
 - Annotation changes
 - Document saves
- Sends event information to evaluation management

IDE-IT Backend Approach



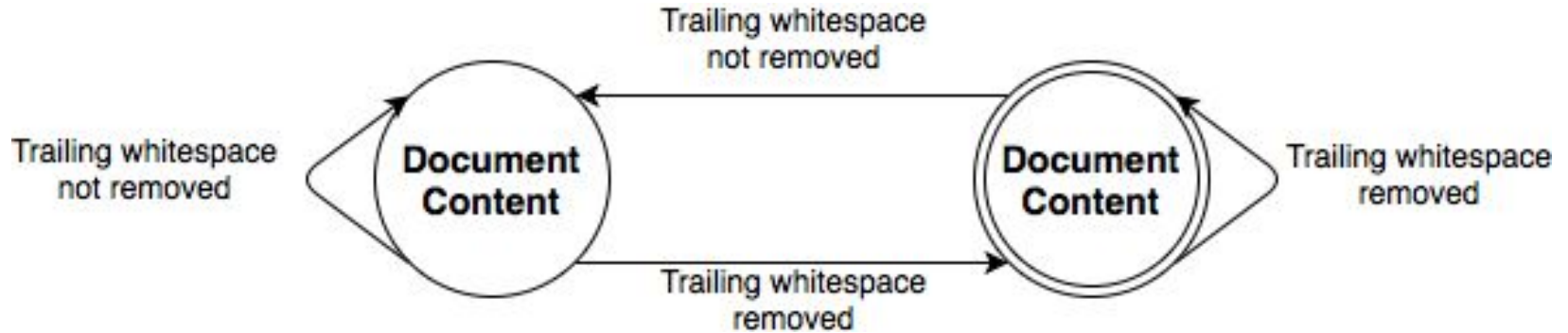
- Attaches evaluators to document editor windows
- Evaluators notified of change events via the Eclipse API
- Evaluators process change events through evaluation functions
- Notifies frontend interface of any suggested features

IDE-IT Backend Approach

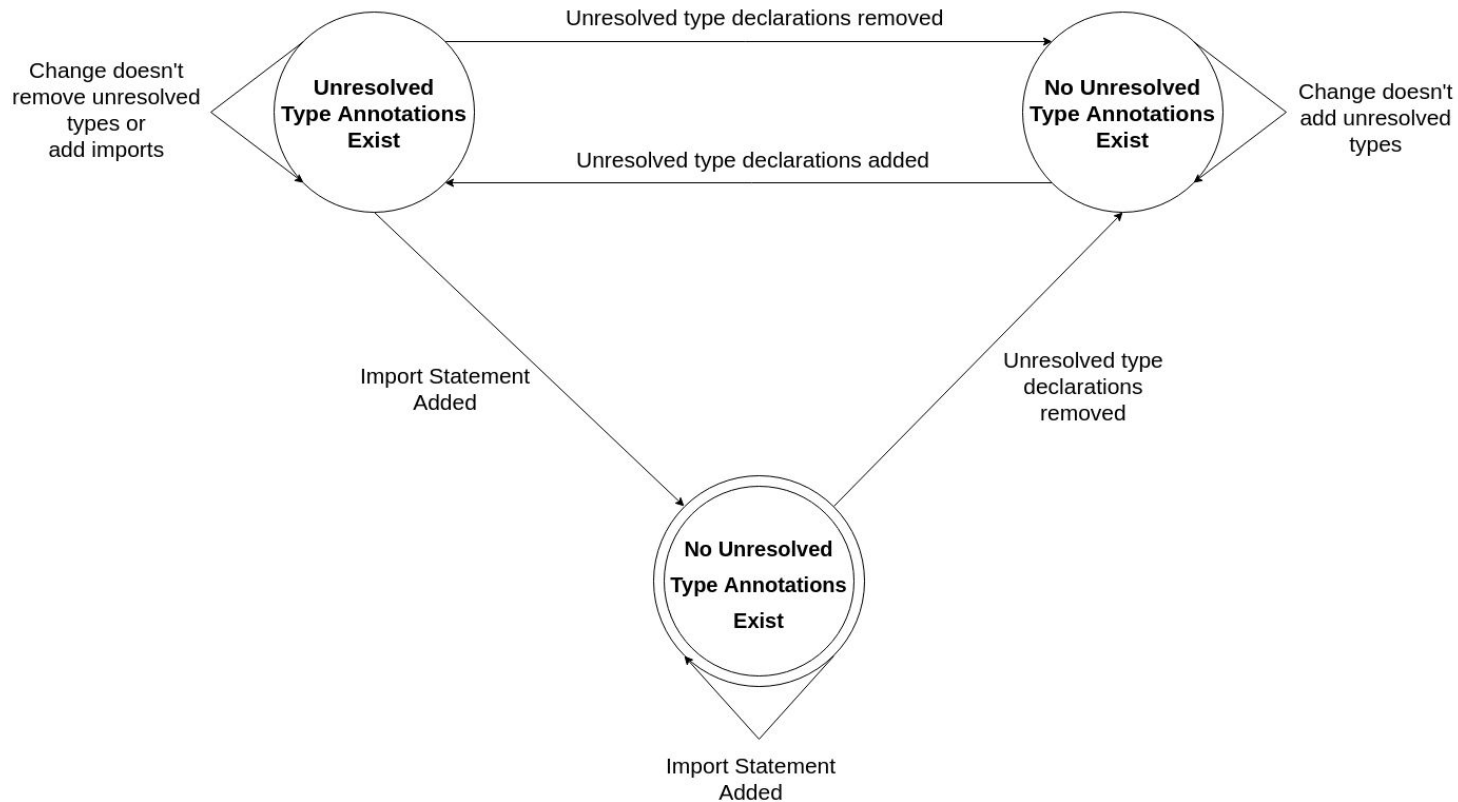


- Gets change event information from evaluation management
- Determines if change event means a feature should be suggested
- Notifies evaluation management if a feature should be suggested

Remove trailing whitespaces on save



Add Imports Evaluator



Demo

Project Success Metrics

Our focus:

- Maximize true positives (minimize false negatives)
- Keep false positives low, but less of a concern

Evaluate against a list of possible user actions for each feature

- Ask users to find all ways they can to circumvent using a feature
- Compile results into a list of “cases”
- Compare # of cases our evaluators cover to the total # of cases

Evaluator Function Testing

Tests of true positive cases (cases where the feature evaluator should trigger)	
FeatureEvaluator	Percentage Covered
AddImport Evaluator	100%
BlockComment Evaluator	72%
CorrectIndentation Evaluator	94%
GetterSetter Evaluator	66%
RemoveImport Evaluator	100%
TrailingWhiteSpace Evaluator	100%

Conclusions

- IDE-IT has proven to be a successful technique for discoverability of IDE tools
- Easily extensible with high modularity
- Available for download and use from github