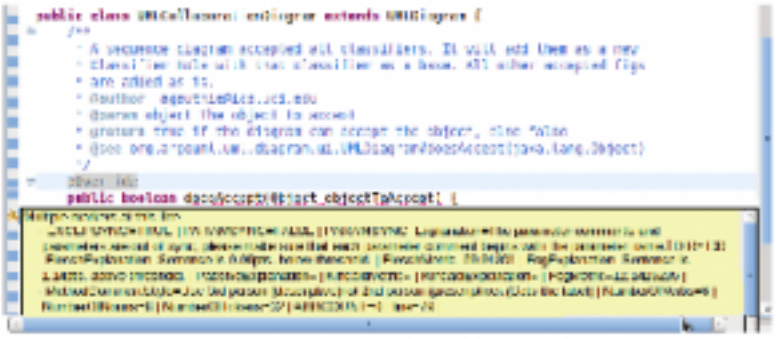
SA-Flyer: Eclipse

Semantic Assistants: Eclipse Plug-In

The Eclipse plug-in for the Semantic Assistants architecture provides a user interface for calling text analysis services directly from within the development environment. In particular, when using Eclipse as a software development environment, novel domain-specific semantic services, such as named entity detection or quality analysis of source code comments can be offered to software developers.



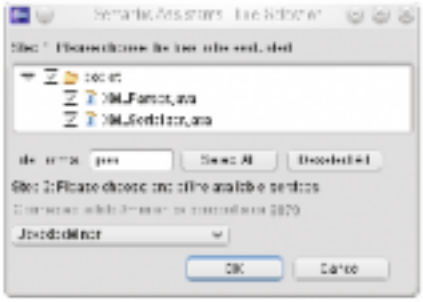
Background

Software engineers need to be able to create, modify, and analyze knowledge stored in software artifacts. A significant amount of these artifacts contain natural language, like version control commit messages, source code comments, or bug reports. Many widely-used IDEs are only concerned with syntactic aspects and some basic forms of analysis (e.g., spell-checking) of software artifacts – they do not offer support for analyzing unstructured natural language and relating this knowledge with the source code.

Our Semantic Assistants plug-in for the Eclipse platform allows users to access any NLP analysis service deployed in the Semantic Assistants architecture. Example services that can help managing large amounts of textual content include information extraction, index generation and automatic traceability of software artifacts.

Design

The Semantic Assistants plug-in adds a new menu entry to the Eclipse native toolbar, from which users can inquire about and invoke available services. Service results are displayed depending on the type of the server response: either as a new text file, as annotations on the existing file, or by opening an external viewer (e.g., a Web browser for HTML files). In the figure above, a Java source code comment quality analysis service has been run a on Java method that detected inconsistencies between the source code and its Javadoc comments.



Features

The plug-in is able to perform analysis on text documents, source code files and even complete projects. It also offers the possibility to customize NLP services by specifying runtime parameters, e.g., a threshold for text readability metrics. Integration of new services does not require any changes on the client side–any new NLP service deployed by a language engineer is dynamically discovered by the architecture and so becomes immediately available. Analysis results are consistent with the workow and visualization paradigm in a software IDE; e.g., detected NL "defects" are mapped to the corresponding line of code in the editor, similar to code warnings displayed in the same view.

More Information, Software Download, and Contact

Semantic Software Lab, Concordia University, Montréal; Contact: Prof. René Witte <witte@semanticsoftware.info>
Open Source Software Download (AGPL3): http://www.semanticsoftware.info/semantic-assistants-eclipse-plugin