



Lile Hattori
lile.hattori@lu.unisi.ch

An Environment for

Synchronous Software Development



Michele Lanza
michele.lanza@unisi.ch

Collaboration?

Collaboration is essential for the development of complex software systems.

When a team of developers is not co-located and face-to-face communication is lost, workspace awareness help developers to keep up-to-date about the others' activities.

However, workspace awareness is usually based on information obtained from SCM systems, thus introducing a latency in the propagation of changes: only when a developer decides to commit the changes to the repository they will become accessible to others.

Syde.

Syde, our Eclipse plug-in, promotes workspace awareness by propagating changes to developers in real time.

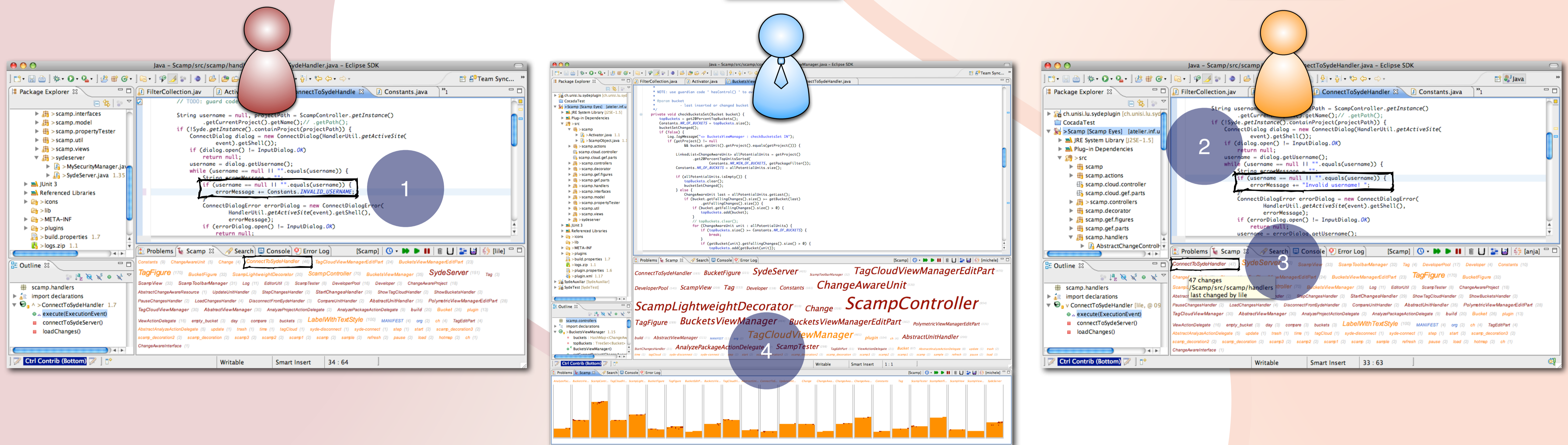
Its fine-grained change tracking mechanism instantaneously notifies any developer working on the same system about changes as other developers implement them.

Syde offers a synchronous environment where developers can abandon the observer role and start taking actions to communicate with each other. For example, spot specialists and seek for help, and monitor others' actions and offer help to those who need it.

Lile

Michele

Anja



Imagine...

Lile (👤) and Anja (👤) are both working on the Scamp project, while Michele (👤) is the stern (mind the tie) manager of their work. All three have Syde installed in their Eclipse IDE.

1 Lile decides to refactor the code of `connectToSydeHandler`, while Anja is working on a class that depends on it. Lile changes the string "invalid username!" to the constant `Constants.INVALID_USERNAME`.

2 As Lile saves the changes, Anja is notified by Syde (the tag cloud refreshes itself and reorders the tags from the most to the least recent changes).

3 Anja sees the change history of `connectToSydeHandler`, which shows that Lile was the last one to edit it. Anja, then, gets concerned about whether Lile's changes will affect the class she is coding on now. Luckily she finds it out at an early stage and can take preventive actions.

4 Michele uses Syde's views to analyze the evolution of Scamp. He notices that Lile has been barely touching it compared to Anja's work. He asks Lile, who explains that she is changing Scamp in the context of its integration with Syde.

See!

Syde currently captures new changes every time a developer saves and compiles his code. This information is immediately broadcast and appears as different views, or annotations on the Eclipse workbench of all team members.

A developer can decide to what extent he wants to be notified by choosing which views he wants to see, and can request or compare his local version to the latest version saved on the server. Syde is free and open source software. Try it out!

www.inf.unisi.ch/phd/hattori/syde.html

References

Mining the History of Synchronous Changes to Refine Code Ownership
Lile Hattori, Michele Lanza
In Proceedings of MSR 2009 (6th IEEE Working Conference on Mining Software Repositories), IEEE CS Press, 2009.

An Environment for Synchronous Software Development
Lile Hattori, Michele Lanza
In Proceedings of ICSE 2009 (31st ACM International Conference on Software Engineering - New Ideas and Emerging Results Track), IEEE CS Press, 2009.