Integrating COTS Search Engines into Eclipse: Google Desktop Case Study

Denys Poshyvanyk, Maksym Petrenko, Andrian Marcus

SEVERE Group @

WAYNE STATE UNIVERSITY



Issues in concept location

- Simple and efficient methods -> poor precision and recall
- Complex methods -> higher precision and recall at the cost of efficiency





Evolution REsearch GROUF

IR based concept location

- Uses Latent Semantic Indexing to index and search the source code
- Problems with the prototype
 - slow
 - inefficient re-indexing
 - no resources to support and adoptable version

Solutions for adoption

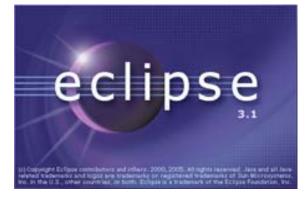
- Industry grants
 - not always fits our research agenda
- Component based approach with industrial strength COTS













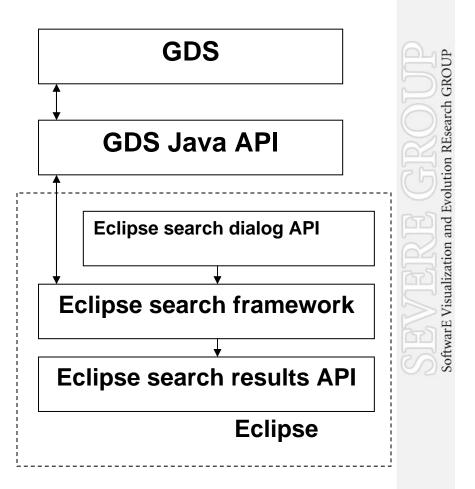
Advantages

- Support of natural language queries
- Scalability, efficiency, and reliability
- Sorting results by relevance (date)
- Search in intra- and extra-net



Integrating GES into Eclipse

- GDS Java API
- org.eclipse.search extension points:
- searchPages
- searchResultViewPages





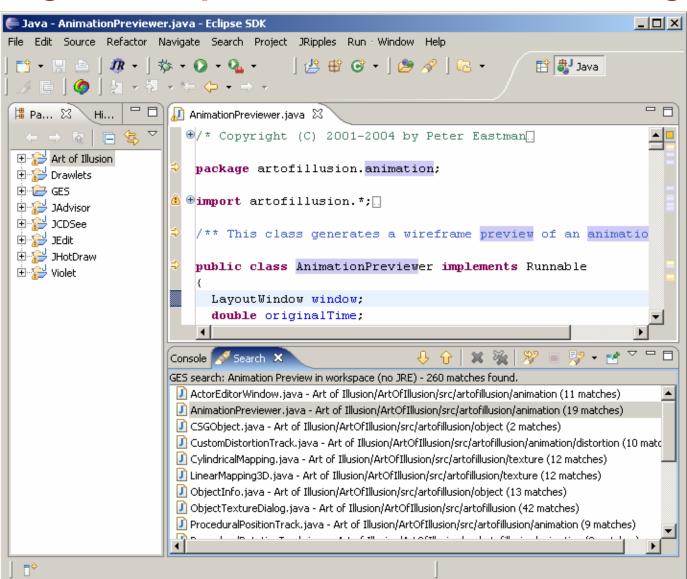
SoftwarE Visualization and Evolution REsearch GROUP

Google Eclipse Search (GES) Plug-in

nter the search str Animation Preview			
Highlight mate	ches		
Scope			
Workspace	Selected resources	Enclosing proje	ects
) Working set:			Choose

Wayne State University

Google Eclipse Search (GES) Plug-in



SIEVIERE GIROUT SoftwarE Visualization and Evolution REsearch GROUP

Advantages of COTS GDS

- GES improves as GDS does
- E.g., GDS v4 removes deleted files from the search results
- Modifiers to restrict scope of the search
 - previously had to tweak undocumented Win registry keys

Full Price of GDS

- Indexes everything (pdf, mail, app data)
- Re-indexing is done while PC is idle
- GDS v3 feature 'share across computers'
 - indexed files are copied to the server
 - security and privacy issues
- "Lock Search" feature

Availability

- Open-source
- <u>http://ges.sourceforge.net/</u>
- Researchers extended and compared GES with other tools:
 - Shepherd et al [AOSD'07]



WAYNE STAT

Future Work

- Searching based on developer profiles
 Visited/edited/deleted files
- Hybrid search engines (COTS-based)
 - LSI + GDS

Googling public source code from IDE

org.openide.util.Helputx;					
org.openide.util.NbBu ⁻¹ org.openide.util.acti	Go To		•		
org.openide.acti.acti	Select in		•		
final class GoogleCod	Find Usages	Alt+F7			
	Show Javadoc	Alt+F1			
tected void performAc	Refactor		•		
EditorCookie c = (Ed	Google Co <mark>d</mark> e Search		Cook		
JTextComponent edito	Reformat 🖄 de	Ctrl+Shift+F	ent ()		
String selection = e	Fix Imports	Alt+Shift+F	ene ()		
porting beredoton - e	Surround With true optob	ALL CHIEF W			