

Mining Billions of AST Nodes to Study Actual and Potential Usage of Java Language Features[^]

PROBLEM STATEMENT

- Programming languages constantly evolve
- New features added in Java JLS2/JLS3/JLS4
- *When are features adopted by developers?*
- *Are features living up to their full potential?*
- *Do developers refactor to adopt features?*

APPROACH

- Studied open-source Java projects on SourceForge
 - Both small (novice developers) and large projects
 - Azureus/Vuze, Weka, Hibernate, JHotDraw, JUnit, etc
- Utilized Boa language and infrastructure to mine data
- Analyzed CVS and Subversion repositories
 - Over 10 years of commit history

Studied Dataset: September 2013 (SF.net)

Projects	31,432
Revisions	4,298,309
Java Files	9,093,216
Java File Snapshots	28,747,948
AST Nodes	18,323,905,323

1. Do projects use language features before release?

Result: Yes! (more details in paper...)

2. How frequently is each language feature used?

	Annotation Assert	Annotation Declaration	Annotation Use*	Enhanced For Loop*	Enums	Generic Variable*	Generic Method	Generic Type	Extends Wildcard	Super Wildcard	Other Wildcard	Varargs	Binary Literals	Diamond	Multi Catch	Safe Varargs	Try w/ Resources	Underscore Literals
Uses	408,802	29,415	11,692,911	2,666,411	162,445	2,473,581	257,921	214,012	411,940	84,602	936,546	221,322	90	22,473	1,920	192	1,597	889
Files	1.04%	0.28%	21.98%	8.4%	1.47%	9.24%	0.9%	1.89%	1.37%	0.15%	2.45%	0.95%	0%	0.08%	0.01%	0%	0.01%	0%
Projects	12.72%	6.35%	57.07%	48.61%	29.42%	57.15%	13.52%	19.87%	15.67%	2.71%	22.52%	15.43%	0.02%	0.4%	0.27%	0.06%	0.21%	0.02%

* Some features are much more popular

3. How Did Committers Adopt and Use Language Features?

- Features are not adopted by teams as a whole, but rather by a few members (confirms [Parnin et al. MSR'11])
- Very few committers have adopted all new language features

4. Were there missed opportunities to use Language Features? **Result: Millions of missed opportunities!**

Actual Uses (% projects)

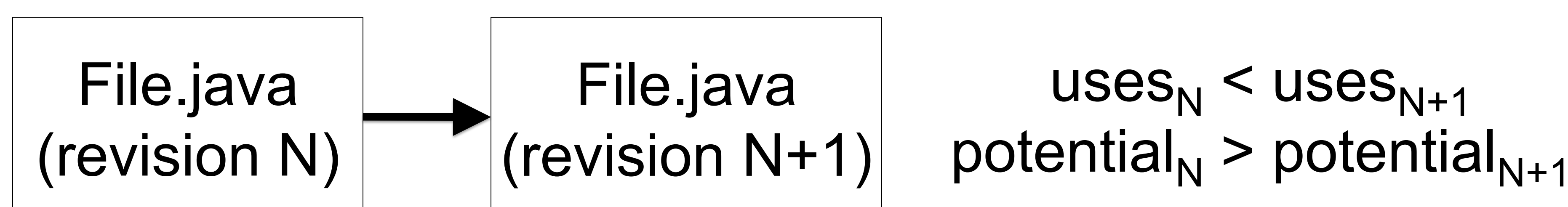
Binary Literals	Diamond	Multi-Catch	Try w/ Resources	Underscore Literals
90 (0.02%)	22K (0.4%)	2K (0.27%)	1.6K (0.21%)	900 (0.02%)

Potential Uses (increase % projects)

Binary Literals	Diamond	Multi-Catch	Try w/ Resources	Underscore Literals
61K (↑5.88%)	3.7M (↑58.68%)	365K (↑49.48%)	522K (↑37.06%)	24.5M (↑88.84%)

5. Was Old Code Converted to Use New Language Features?

Result: Developers convert existing code to new language features!



	Assert	Varargs	Diamond	MultiCatch	Try w/ Resources	Underscore Literals
Files	105	1.6K	3.8K	125	99	1
Projects	37	488	72	23	17	1

RESEARCH QUESTIONS