

Davide Rendina

Estensione di un tool per il riconoscimento di nuovi architectural smell

Relatore Prof.ssa Francesca Arcelli Fontana

Co-relatore Dott.ssa Ilaria Pigazzini

Anno Accademico 2019-2020



Obiettivo

Estendere le capacità di un tool esistente per riconoscere tre nuove tipologie di problemi nell'ambito delle architetture software



Lavoro svolto

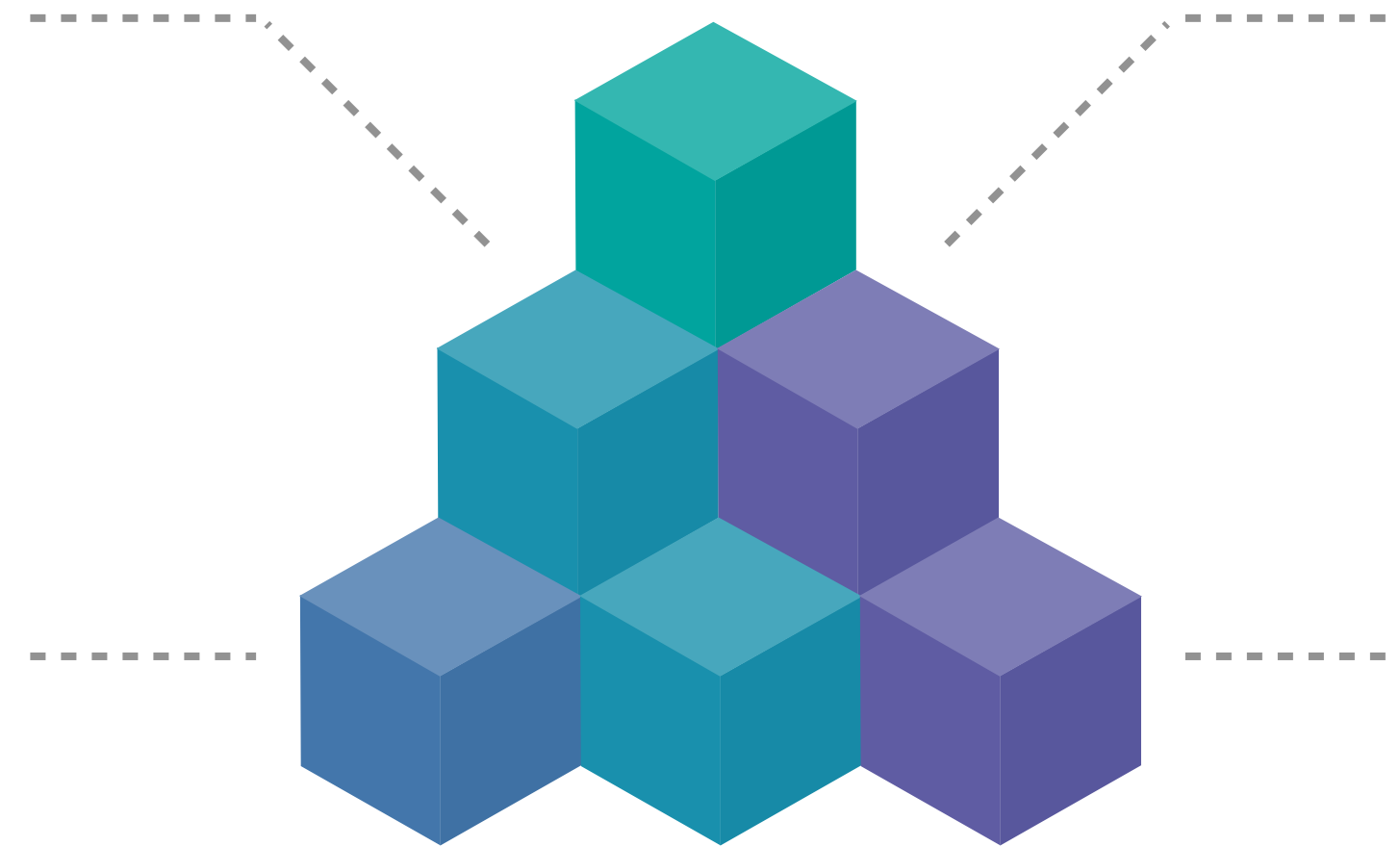
1. Modifiche alle strategie e strutture dati del tool
2. Implementazione nuovi algoritmi di detection
3. Validazione degli algoritmi sviluppati

ARCHITETTURA

Divisione e organizzazione del sistema in differenti componenti in relazione tra loro

Supporta l'intero **ciclo di vita** del software

Influenza le **risorse** utilizzate dal team



Architettura

La sua **qualità** condiziona quella dell'intero progetto

Può presentare diversi problemi tra cui gli **Architectural Smell**

ARCHITECTURAL SMELL

Violazioni di design principles o decisioni progettuali che diminuiscono la qualità del sistema



Comprensione

Facilità di comprensione del sistema e della sua struttura



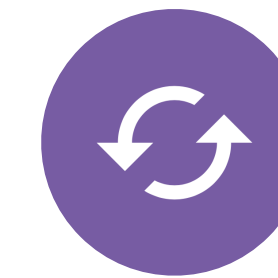
Affidabilità

Corretta realizzazione delle funzionalità da parte dei componenti e prevenzione degli errori



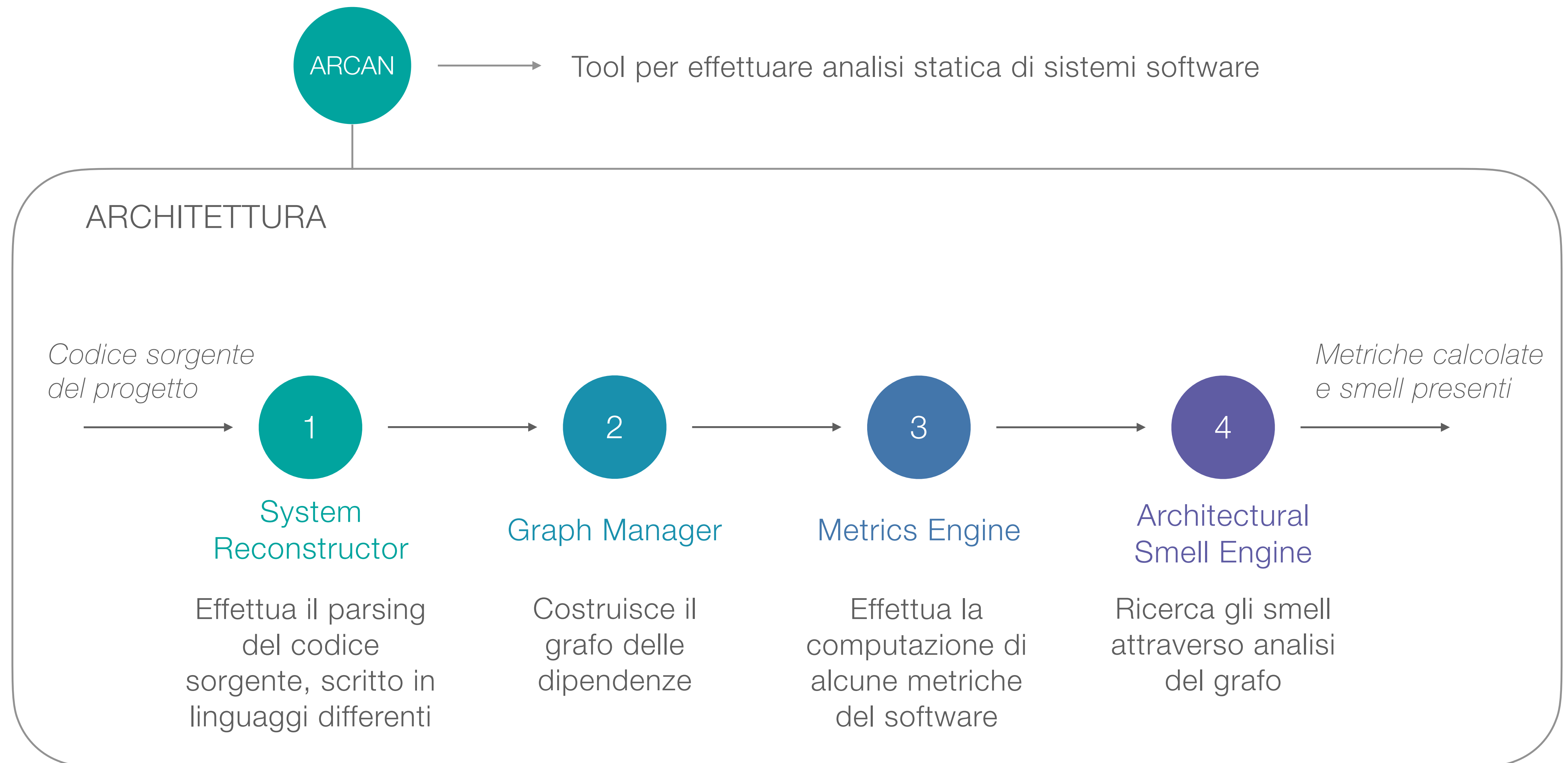
Manutenzione

Semplicità nello svolgimento delle attività di manutenzione del sistema



Riutilizzo

Possibilità di riutilizzare i componenti del codice in contesti differenti



Subclasses Do Not Redefine Methods

Si presenta quando, in una relazione gerarchica, la sottoclasse non ridefinisce alcun metodo della sua superclasse



Unutilized Abstraction

Si manifesta quando una classe o interfaccia viene lasciata inutilizzata, cioè non direttamente usata o non raggiungibile



Unnecessary Abstraction

Si verifica quando una classe o interfaccia non necessaria, e che quindi potrebbe essere evitata, viene introdotta nel design



MODIFICHE AL PARSER E AL GRAFO DELLE DIPENDENZE

Necessità di rappresentare nel grafo nuovi elementi per la detection degli architecture smell



MODIFICHE AD ARCAN

Modifiche alle strategie di parsing e ai componenti del grafo delle dipendenze

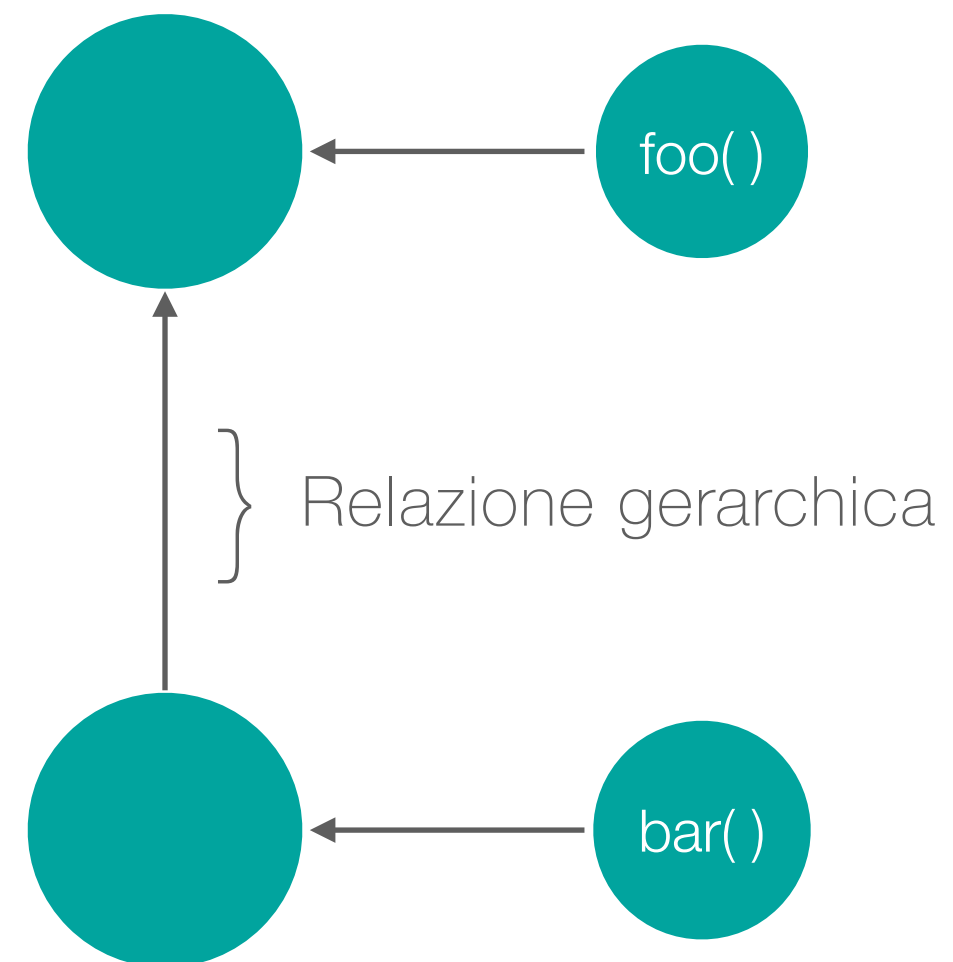
Rappresentazione
di attributi

Necessaria per il
riconoscimento di
Unnecessary
Abstraction

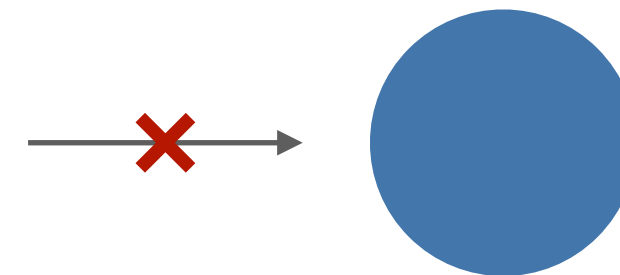
Rappresentazione
di funzioni

Necessaria per il riconoscimento di
Subclasses Do Not Redefine Methods,
Unnecessary Abstraction,
Unutilized Abstraction

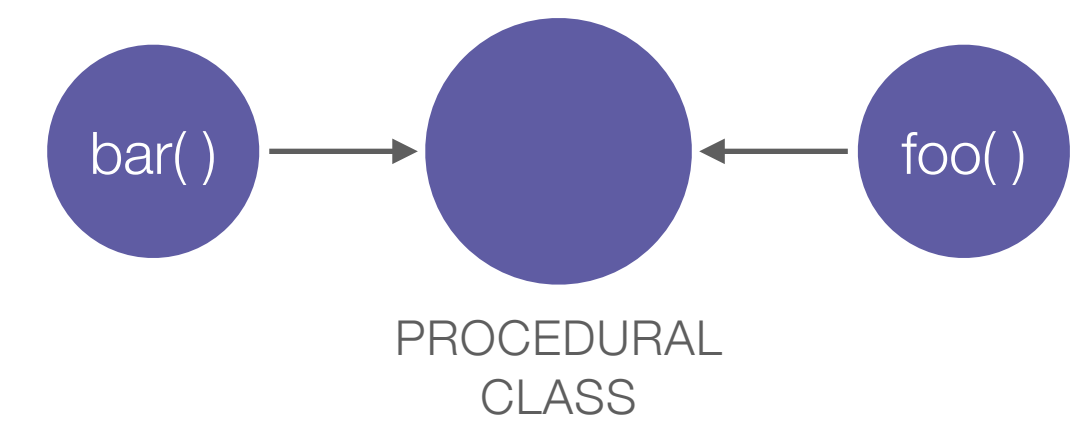
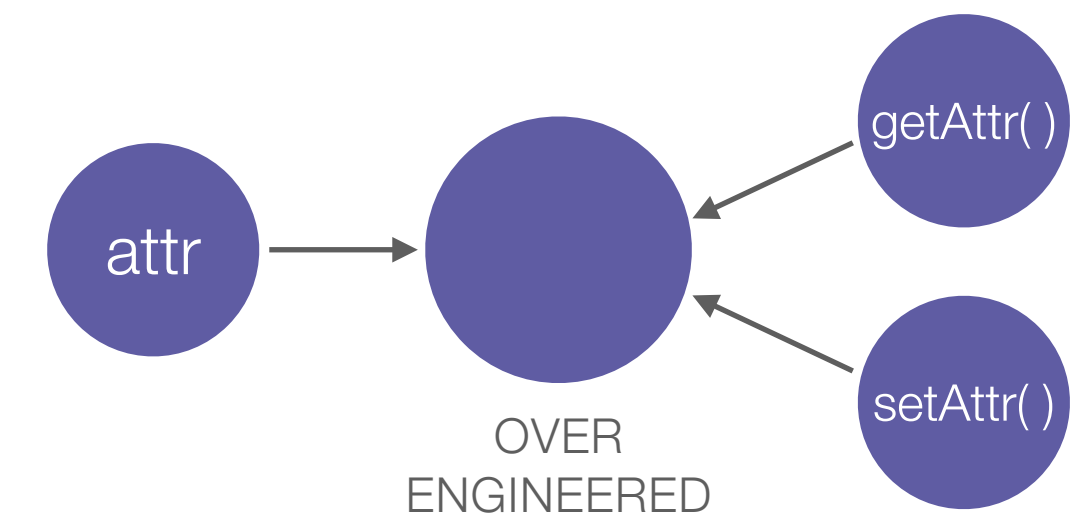
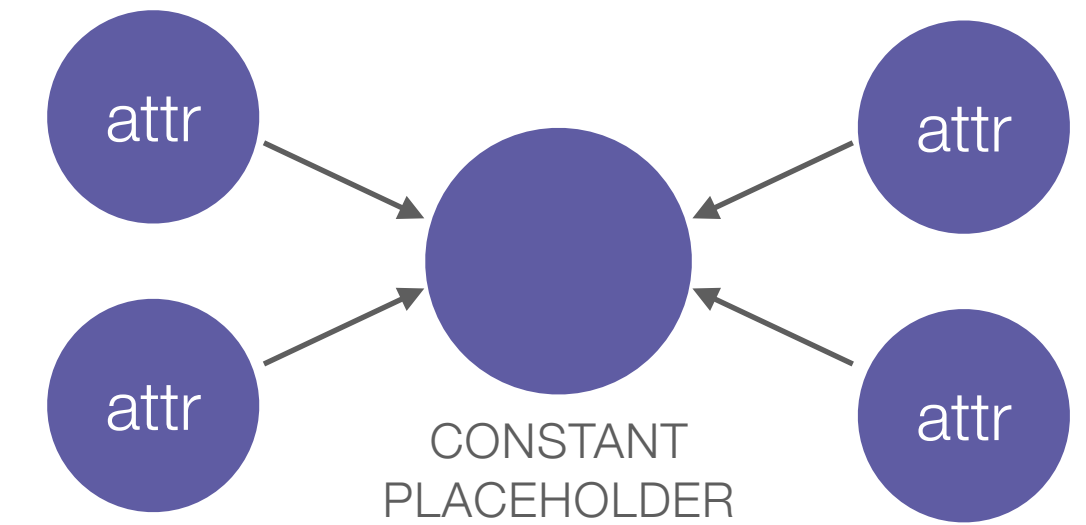
Subclasses Do Not Redefine Methods



Unutilized Abstraction



Unnecessary Abstraction





PROGETTO	LOC	N° CLASSI	N° PACKAGE	PROGETTO	LOC	N° CLASSI	N° PACKAGE
Accumulo	43 682 818	3 789	212	Flink	148 600	915	49
Beam	52 414	201	8	Geode	2 236 679	4 293	190
Bookkeeper	1 495 659	2 036	252	Kafka	575 967	2 389	138
Cassandra	33 968 645	3 910	116	Skywalking	141 674	979	526
Druid	369 487	736	63	Zookeeper	269 378	755	52

RISULTATI DETECTION E VALIDATION

Subclasses Do Not Redefine Methods

708
ISTANZE TOTALI

400
ISTANZE VALIDATE

89.75%
PRECISION

Unutilized Abstraction

2033
ISTANZE TOTALI

460
ISTANZE VALIDATE

95.43%
PRECISION

Unnecessary Abstraction

2491
ISTANZE TOTALI

448
ISTANZE VALIDATE

73.66%
PRECISION

Sono state presentate le attività svolte al fine di poter riconoscere gli smell Subclasses Do Not Redefine Methods, Unutilized Abstraction e Unnecessary Abstraction.

Alcuni sviluppi futuri per il progetto di Arcan possono essere:



Estensione della
capacità del tool



Miglioramento
algoritmi introdotti



Refactoring automatico
degli smell

Grazie per l'attenzione

Davide Rendina

