



UNIVERSITÀ DEGLI STUDI DI SALERNO

Dipartimento di Informatica

Corso di Laurea Triennale in Informatica

**INTERVENTI DI MANUTENZIONE ED EVOLUZIONE DEL SOFTWARE:
IMPATTO DELLE METRICHE DI RIUSO**

Relatore

Prof. Carmine Gravino

Prof. Fabio Palomba

Dott. Giammaria Giordano

Università degli Studi di Salerno

Candidato

Gerardo Festa

Matricola: 0512105908

Anno Accademico 2020-2021

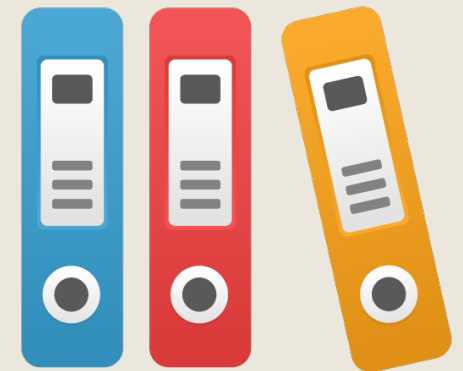
Introduzione e background



Il Riuso

Pro

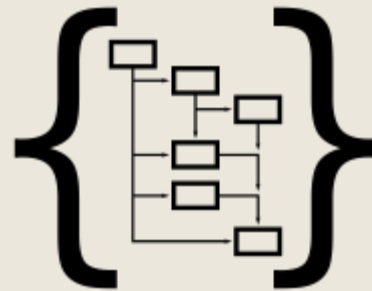
- Time Saving
- Semplificazione della documentazione



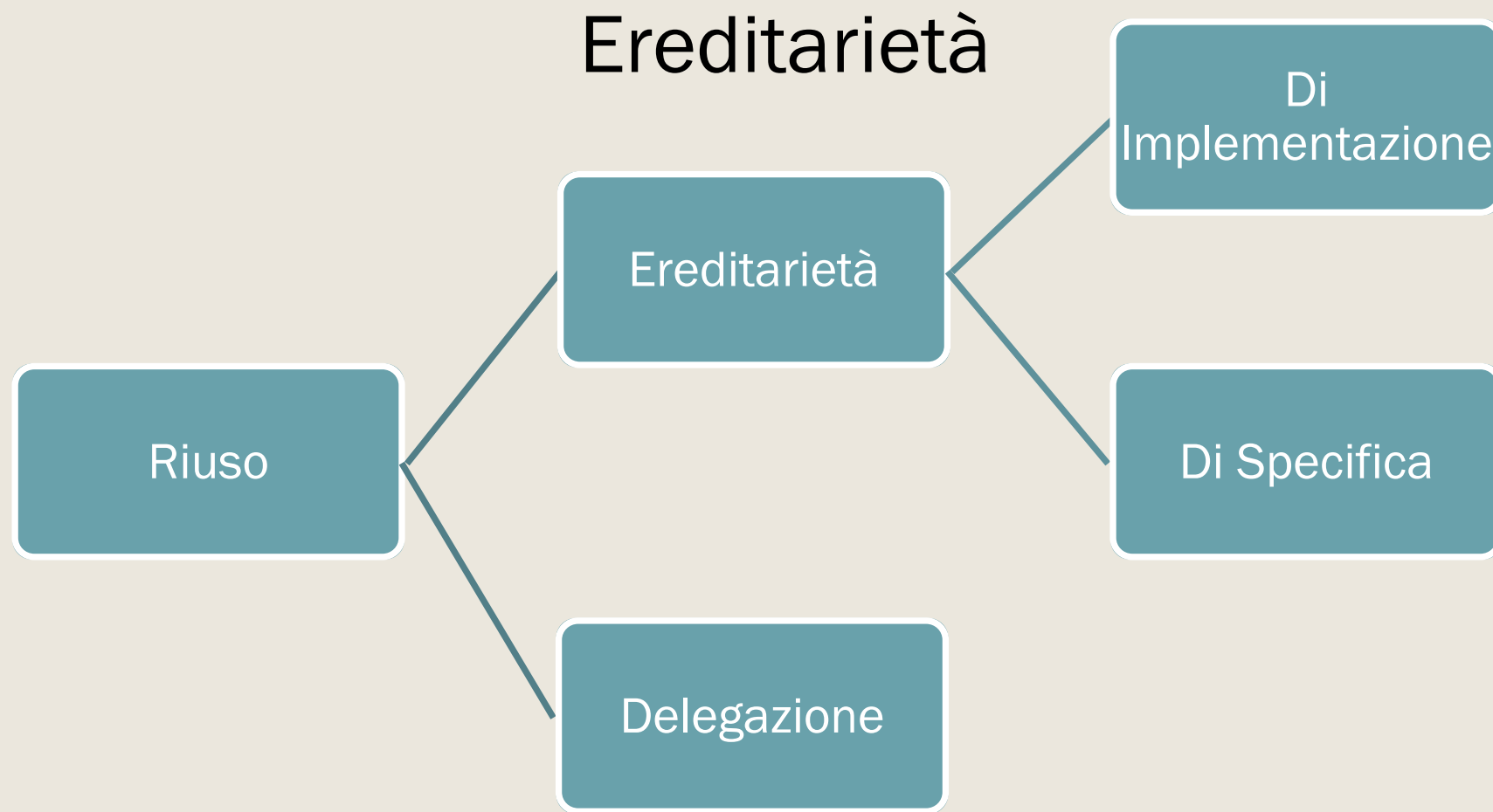
Il Riuso

Contro

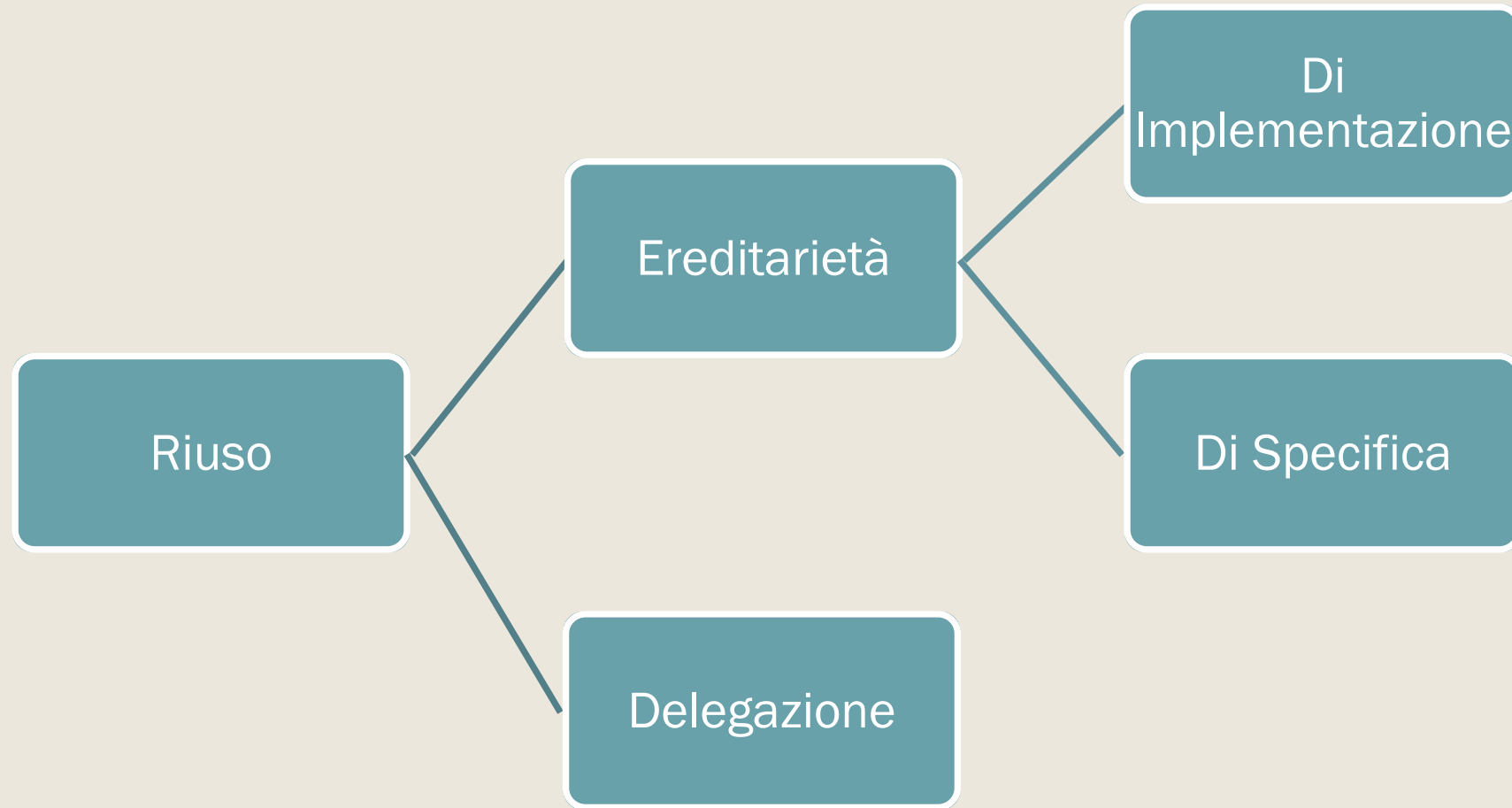
Aumento della complessità



Il Riuso nei linguaggi Object Oriented



Il Riuso nei linguaggi Object Oriented



Il Riuso nei linguaggi Object Oriented

Delegazione



Lavori Correlati

Studi Precedenti

- Assenza di distinzione dei meccanismi di ereditarietà
- Pochi studi sull'evoluzione

Studio *Giordano et al.*

- Campionamento delle release
- Qualità del software come code smells

Studio in esame

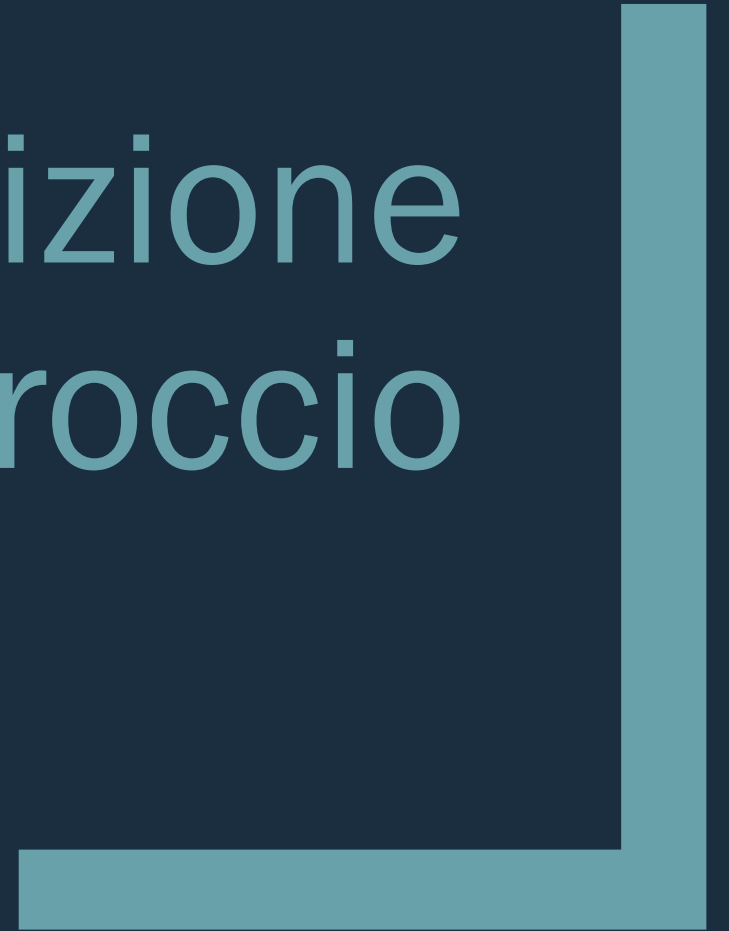
- Distinzione tra ereditarietà di specifica, di implementazione e delegazione
- Studio sull'evoluzione dei meccanismi di riuso
- Relazione tra riuso e qualità del software
- Analisi completa della storia dei progetti
- Qualità del software come presenza di difetti
- Analisi dell'effort per la correzione dei difetti

RQ1. Come varia l'utilizzo dei meccanismi di riuso nell'evoluzione del software?

RQ2. Esiste una correlazione tra l'utilizzo dei meccanismi di riuso e la presenza di difetti nel codice?

RQ3. È possibile predire l'effort necessario per la correzione dei difetti?

Descrizione dell'approccio



Contesto Applicativo

Defects4j come dataset di bugs

Mining repository GitHub per
analizzare l'evoluzione dei software



Mining della repository



Mining delle metriche di riuso

Chidamber & Kemerer Object-Oriented Metrics Suite

WMC	DIT*	NOC*	CBO	RFC	LCOM
Complessità	Profondità	Ampiezza	Accoppiamento	Risposta della classe	Coesione

* Strettamente legate al concetto di ereditarietà

Ereditarietà di implementazione (implnh):

Numero delle classi che fanno uso della keyword 'extends'

Ereditarietà di specifica (intlnh):

Numero delle classi che fanno uso della keyword 'implements'

Delegazione (delegations):

Somma del numero di chiamate a metodi esterni effettuate da tutte le classi

Mining della repository

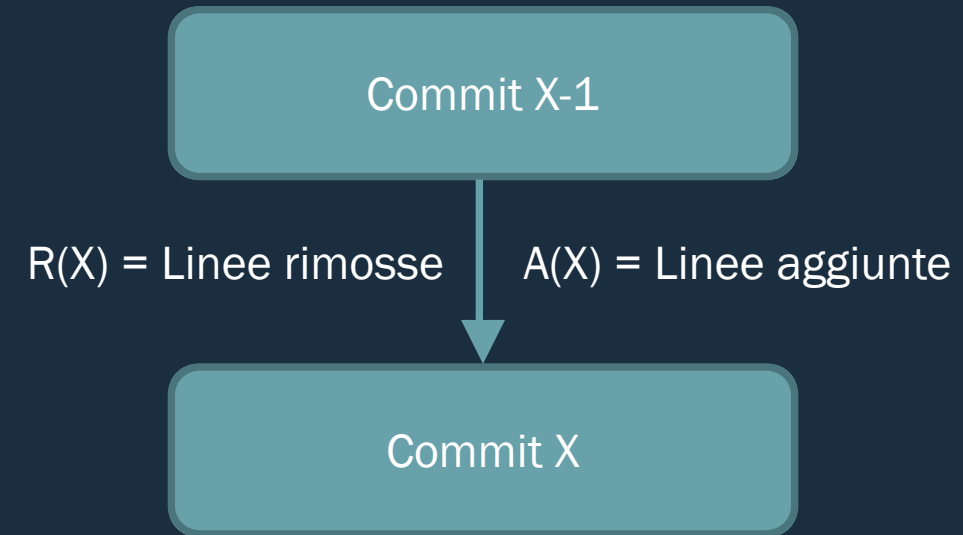


Mining dell'effort e dei bug

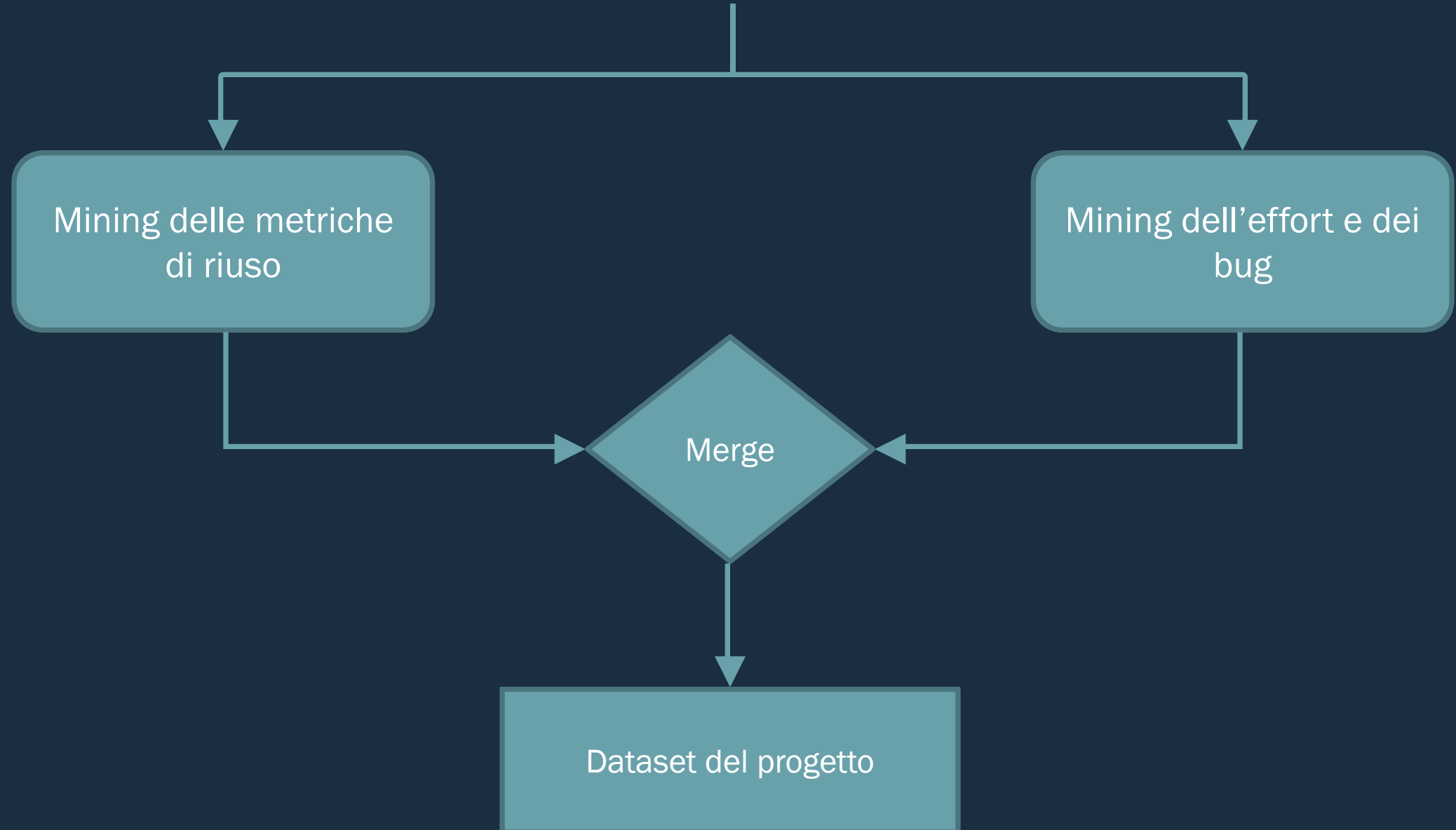
Effort espresso come **code churns**, ovvero la somma delle linee aggiunte e rimosse da un commit

Effort del Commit X \longrightarrow $\text{Churns}(X) = A(X) + R(X)$

Commit che introducono e risolvono bug vengono individuati grazie all'architettura di Defects4j



Mining della repository

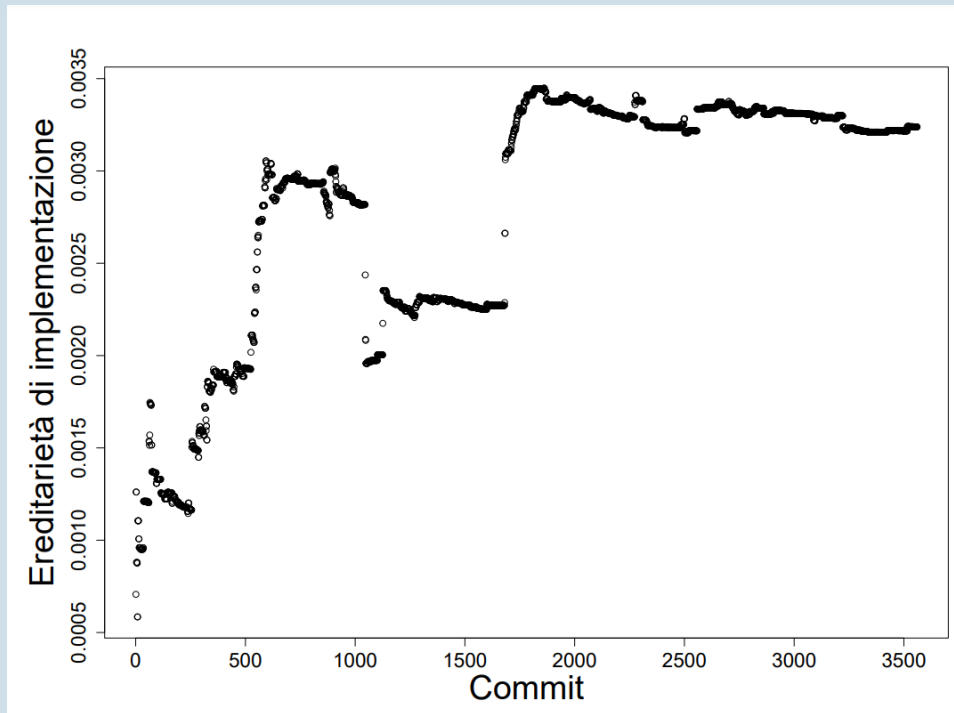


Studio Empirico

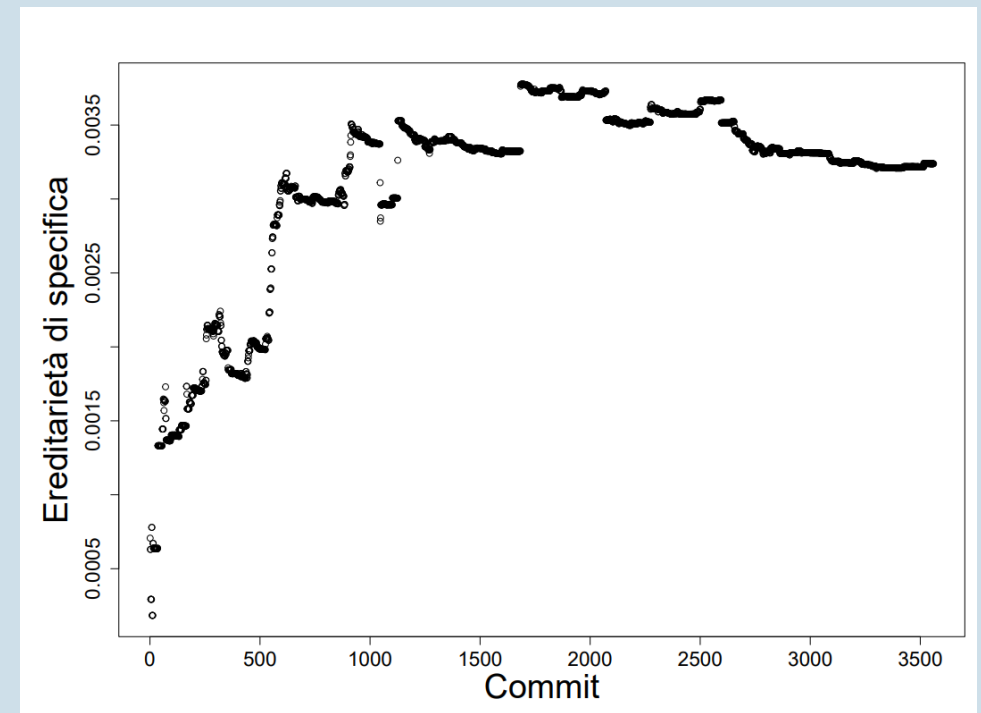


RQ1. Come varia l'utilizzo dei meccanismi di riuso nell'evoluzione del software?

- Normalizzazione delle metriche sulla dimensione del progetto
 - Plot delle metriche normalizzate dal primo all'ultimo commit
- Evoluzione della frequenza relativa del riuso



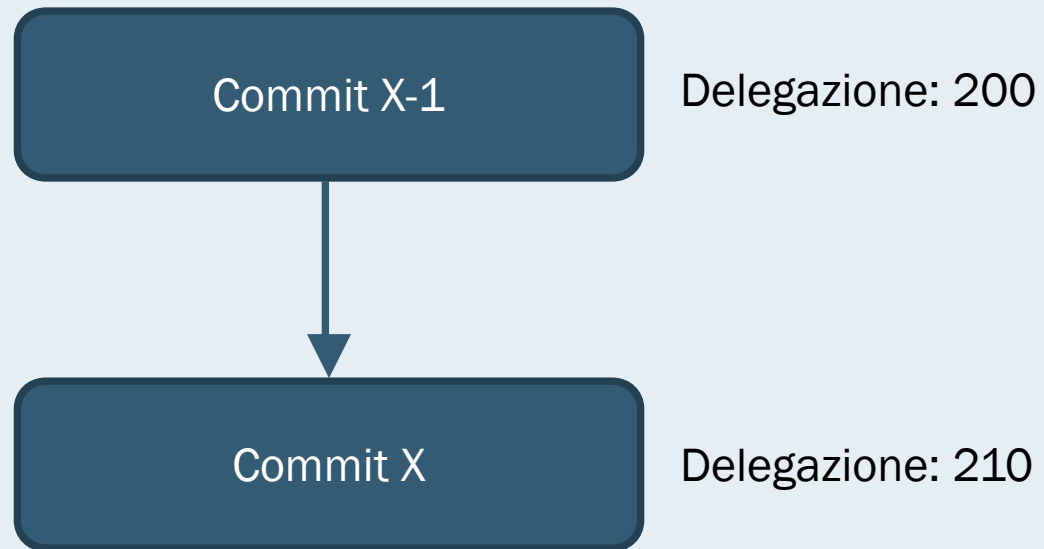
CommonsCollections – Ereditarietà di implementazione



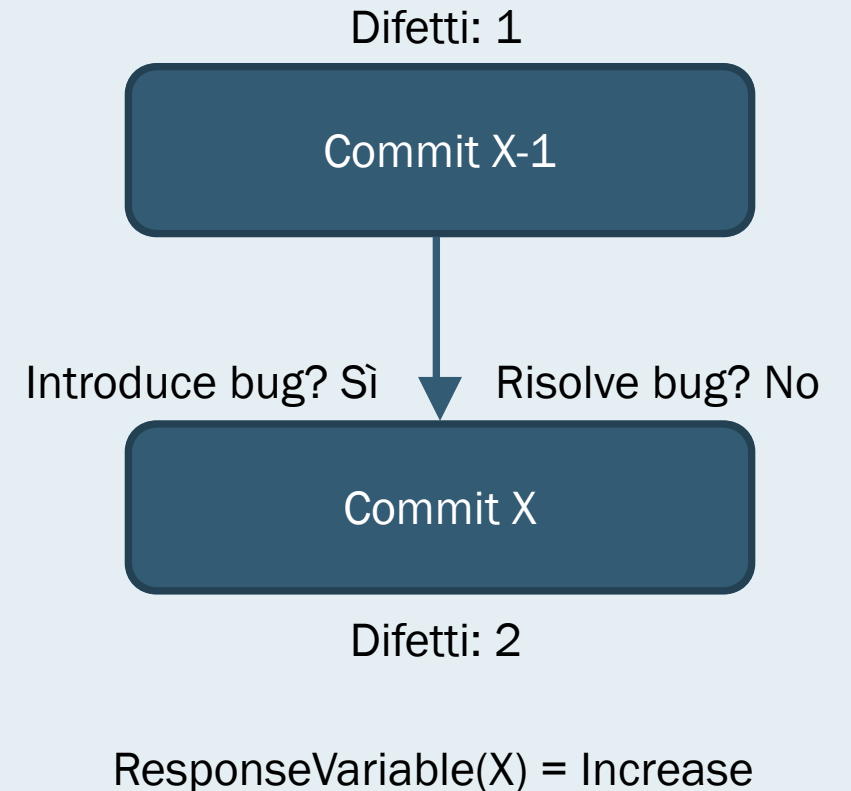
CommonsCollections – Ereditarietà di specifica

RQ2. Esiste una correlazione tra l'utilizzo dei meccanismi di riuso e la presenza di difetti nel codice?

- Presenza dei difetti come variazione del numero di bug
- Variazione del numero di bug come Response Variable del modello
- Variazione delle variabili indipendenti (metriche di riuso e CK metrics)



$$\text{diffDelegazione}(X) = 10$$



RQ2. Esiste una correlazione tra l'utilizzo dei meccanismi di riuso e la presenza di difetti nel codice?

Multinomial Log-Linear Model

- Variabile dipendente categorica
- Variabili indipendenti continue
- Valutato l'impatto delle variabili indipendenti sulla probabilità di introdurre un difetto e sulla probabilità di rimuovere un difetto
- Valutazione delle variabili statisticamente significative

	<i>Dependent variable:</i>	
	Decrease (1)	Increase (2)
diffWMC	-5.248*** (0.033)	-1.903 (4.387)
diffNOC	10.188*** (0.002)	-5.653*** (0.051)
diffLCOM	-0.066 (0.335)	0.046 (0.125)
diffDIT	12.511*** (0.002)	-5.151*** (0.125)
diffCBO	-4.163*** (0.021)	1.467 (2.717)
diffLOC	0.149* (0.090)	0.024 (0.104)
diffdelegations	0.013 (0.017)	-0.003 (0.013)
diffintInh	-0.187 (0.862)	-0.148 (0.637)
diffimpInh	-0.337 (0.488)	0.154 (0.383)
churns	-0.003 (0.002)	-0.0003 (0.001)
Constant	-4.236*** (0.160)	-4.327*** (0.159)
Akaike Inf. Crit.	948.921	948.921
<i>Note:</i> * p<0.1; ** p<0.05; *** p<0.01		

RQ3. È possibile predire l'effort necessario per la correzione dei difetti?

- Effort come code churns
- I churns rappresentano variazioni intrinseche dello stato della repository
- Variazione delle variabili indipendenti (presenza di difetti, metriche di riuso e CK metrics)

Generalized Linear Model

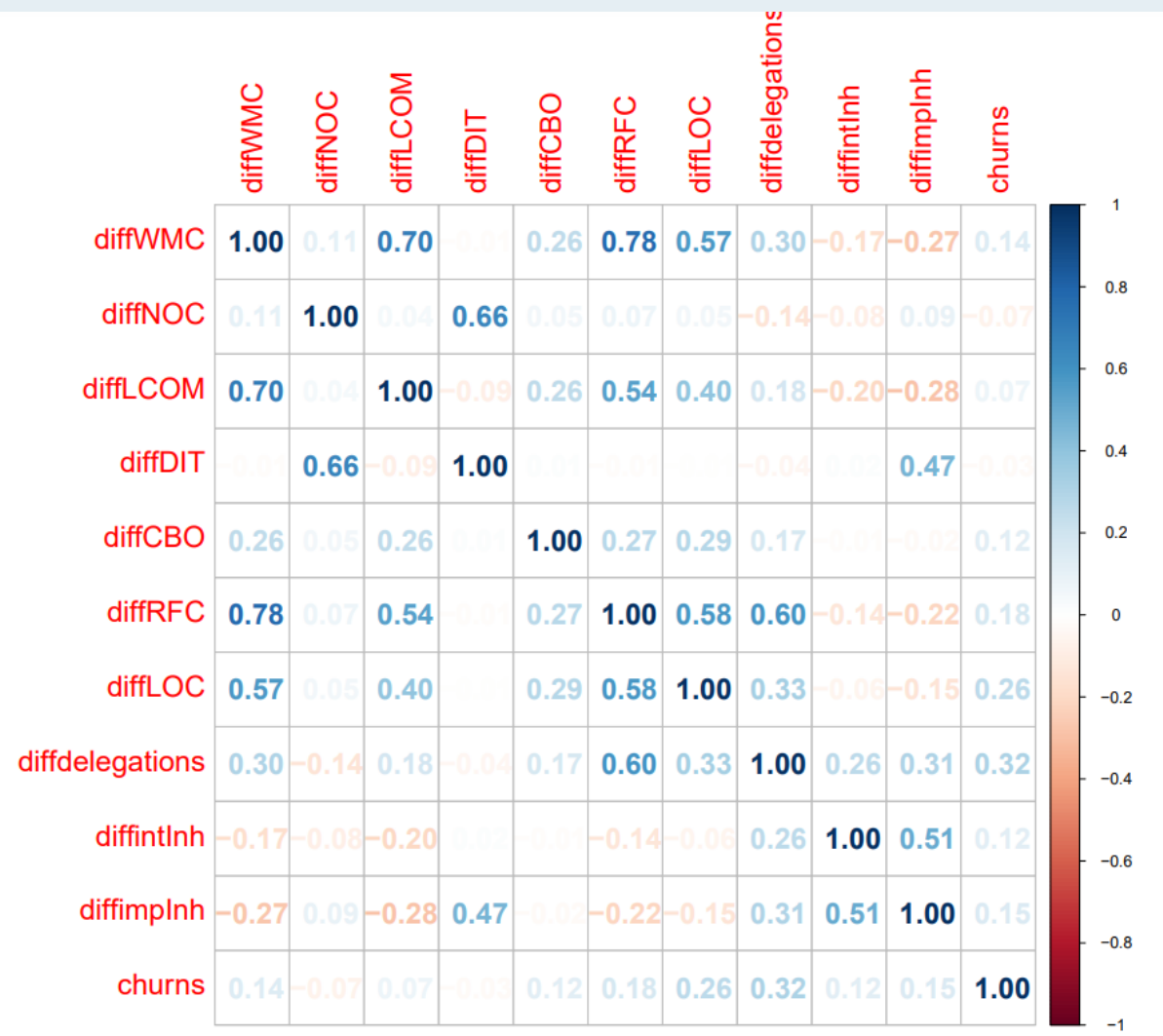
- Variabile dipendente continua
- Variabili indipendenti continue e categoriche
- Valutato l'impatto delle variabili indipendenti sulle linee di codice aggiunte e rimosse
- Valutazione delle variabili statisticamente significative

Rimozione di un bug

	Dependent variable:
	churns
diffNOC	333.786 (509.649)
diffLCOM	-8.269*** (0.992)
diffCBO	239.229*** (19.892)
diffLOC	8.568** (0.946)
delegations	-0.096** (0.044)
intInh	-2.847 (2.324)
impInh	3.001** (1.408)
Bug2Decrease	19.831 (23.738)
Bug2Increase	-20.624 (26.005)
Constant	57.931*** (9.413)
Observations	1,128
Akaike Inf. Crit.	12,367.870
Note:	* p<0.1; ** p<0.05; *** p<0.01

Multicollinearità

- Prima di procedere con i modelli di RQ2 ed RQ3, viene esaminata la matrice di correlazione delle variabili indipendenti per ogni progetto.
- Le relazioni problematiche sono quelle con coefficiente >0.75



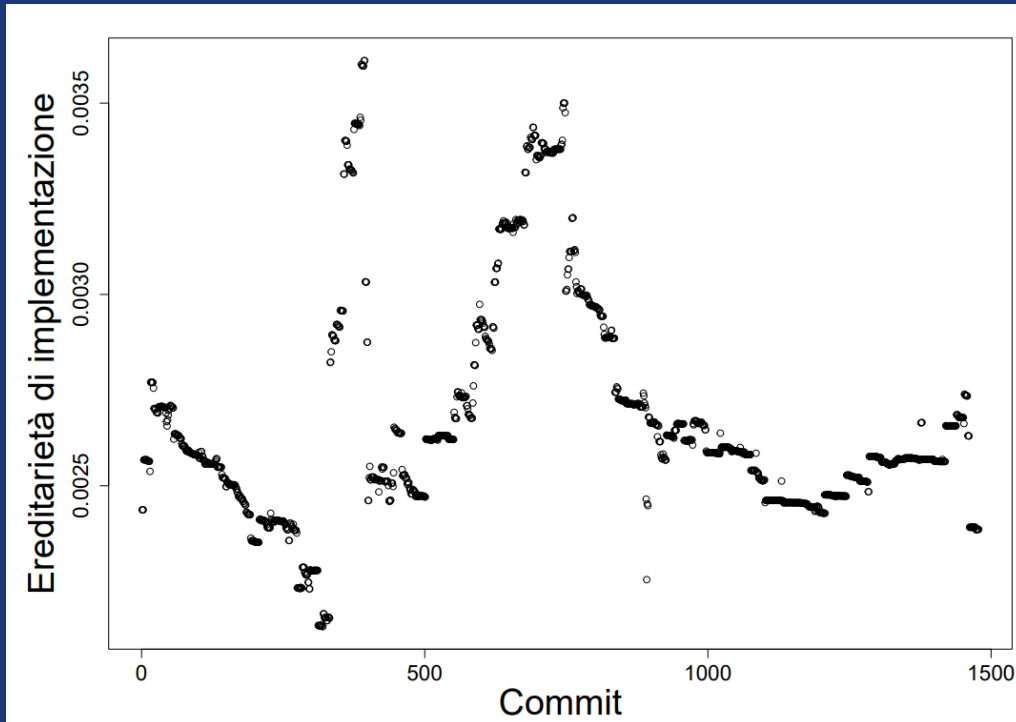
CommonsCollections – Correlation Plot

Risultati e conclusioni

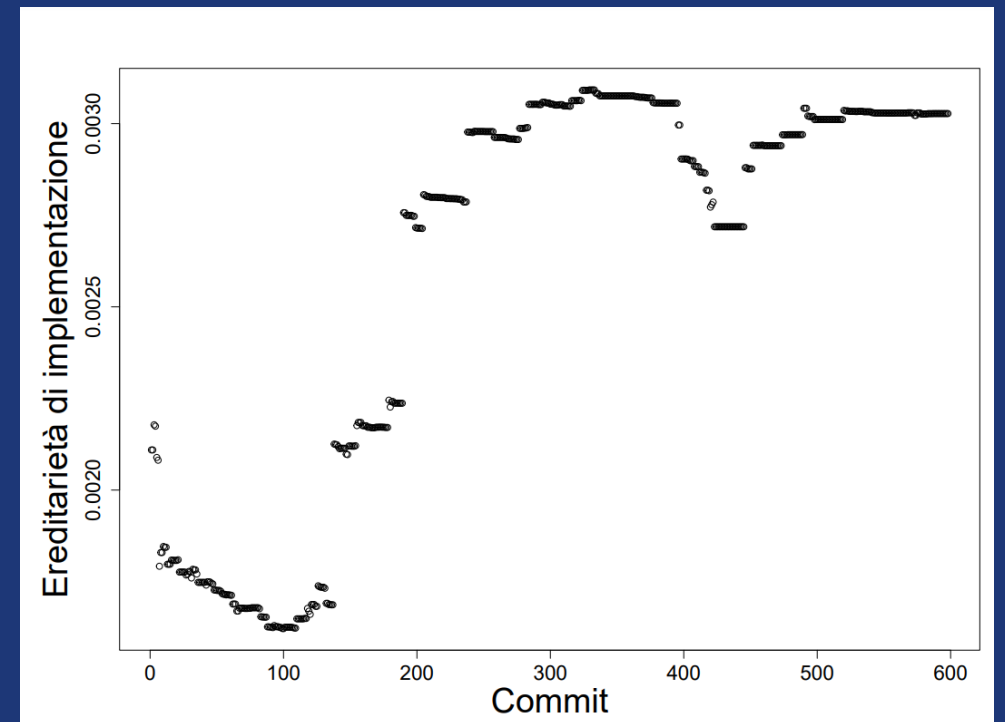


Risultati RQ1

Il riuso evolve, ma i risultati non sono generalizzabili



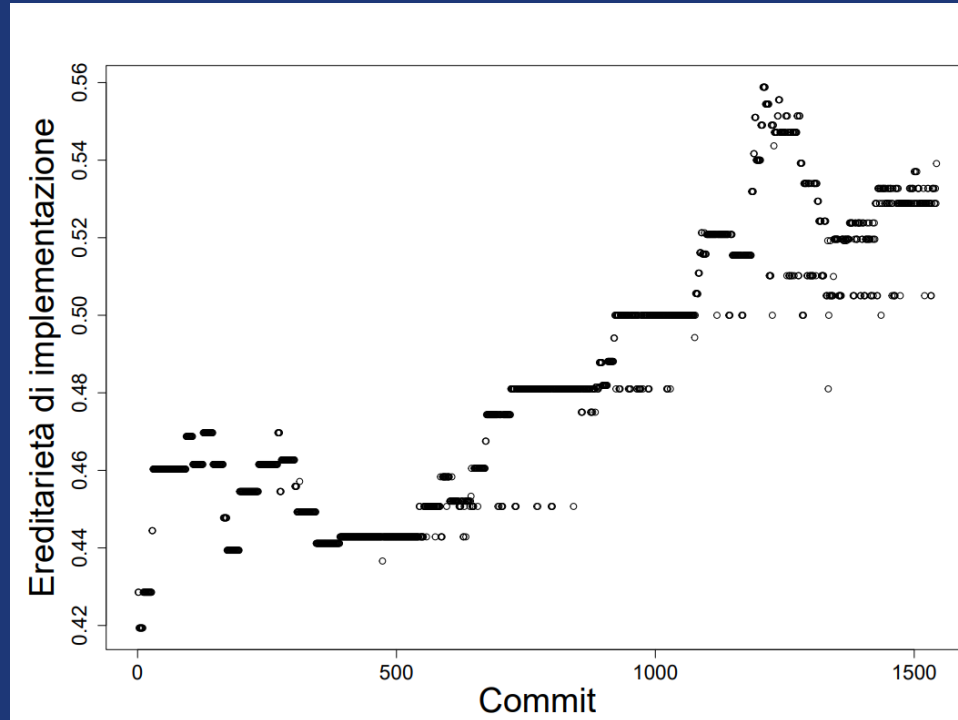
Gson – Ereditarietà di implementazione



JXPath – Ereditarietà di implementazione

Risultati RQ1

Il riuso evolve, ma i risultati non sono generalizzabili



JacksonCore – Ereditarietà di implementazione

La gestione della repository gioca un ruolo fondamentale

Risultati RQ2

Il riuso impatta solo in pochi casi sulla presenza di difetti, con effetti talvolta opposti

Impatto non generalizzabile



	<i>Dependent variable:</i>	
	Decrease (1)	Increase (2)
diffWMC	-2.330 (1.546)	3.344** (1.619)
diffNOC	-58.836*** (0.037)	-152.598*** (0.043)
diffLCOM	0.155*** (0.049)	0.179*** (0.045)
diffDIT	-70.763*** (0.028)	-124.104*** (0.029)
diffCBO	1.062 (0.979)	2.261* (1.194)
diffLOC	-0.045 (0.147)	-0.013 (0.141)
diffdelegations	-0.005 (0.003)	-0.010*** (0.003)
diffintInh	0.109 (0.095)	-0.065 (0.100)
diffimpInh	0.018 (0.099)	0.341*** (0.090)
churns	-0.001 (0.001)	-0.004** (0.002)

	<i>Dependent variable:</i>	
	Decrease (1)	Increase (2)
diffWMC	-14.452*** (2.914)	41.577*** (3.503)
diffNOC	4.203*** (0.113)	-1.798*** (0.040)
diffLCOM	0.217 (1.161)	-1.167 (1.509)
diffCBO	-11.521 (12.928)	-28.867*** (9.482)
diffRFC	8.152 (7.483)	-47.303*** (8.690)
diffLOC	-3.004* (1.736)	5.674*** (1.428)
diffdelegations	0.201* (0.104)	0.399*** (0.100)
diffintInh	-4.686* (2.742)	0.161 (1.719)
diffimpInh	-2.314 (1.718)	-15.482*** (4.800)
churns	-0.015* (0.008)	-0.026*** (0.007)

JacksonDatabind – Modello Multinomiale

JXPath – Modello Multinomiale

Risultati RQ3

La procedura di eliminazione dei difetti non è una discriminante sufficientemente significativa da far variare l'effort

Sul dataset di Defects4j non è possibile stimare l'effort per la correzione dei difetti

<i>Dependent variable:</i>	
churns	
diffWMC	853.627*** (71.347)
diffNOC	22,830.430*** (1,564.318)
diffLCOM	-28.862*** (1.208)
diffDIT	50,782.460*** (1,712.723)
diffCBO	3,147.449*** (74.440)
diffLOC	-8.353*** (3.029)
delegations	0.010* (0.006)
intInh	-1.281*** (0.305)
impInh	-0.140 (0.499)
Bug2Decrease	28.236 (19.492)
Bug2Increase	-7.043 (20.359)

JacksonDatabind – GLM

Mai
significativo

<i>Dependent variable:</i>	
churns	
diffWMC	-889.089 (1,208.343)
diffNOC	24,786.920*** (5,760.288)
diffLCOM	21.501*** (5.505)
diffCBO	3,504.462*** (363.997)
diffRFC	1,358.532*** (363.735)
diffLOC	-158.255*** (12.875)
delegations	-0.598** (0.294)
intInh	6.773 (14.133)
impInh	0.942 (3.559)
Bug2Decrease	-41.147 (149.800)
Bug2Increase	14.086 (149.626)

JXPath – GLM

Sviluppi futuri

Standardizzazione e
analisi qualitative



Grazie per
l'attenzione