

Evaluating Code Clone Genealogies at Release Level: An Empirical Study

(リリースレベルでのコードクローン系譜の評価：
実証的研究)

Ripon K. Saha, Muhammad Asaduzzaman, Minhaz F. Zibran,
Chanchal K. Roy, and Kevin A. Schneider

Proceedings of the 10th Source Code Analysis and
Manipulation (SCAM), pp 87–96

クローン系譜

- クローンの系譜とは、コードクローンの変更の履歴のこと
- クローンの系譜から、コードクローンがいつ発生したか、またクローンセットにいつコードクローンが追加されたか、といった情報を得ることが出来る

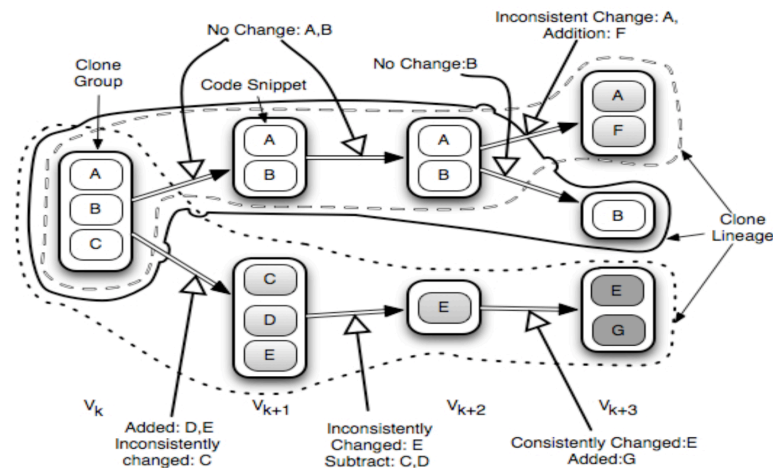


Figure 1. Clone genealogy

背景

- 先行研究では2つのプロジェクトだけ対象にした
- 異なる言語または異なるプログラムサイズを対象にし, クローン系譜を調べる研究はない
- 実証的な評価を行う必要がある

目的

- 17のオープンソースシステムにおけるクローン系譜の評価に関する詳細な実証的研究を実施する
- コードクローンが各リリースでどのように進化するかをクローン系譜の観点から理解するため

メソッド

- 4つの異なるプログラミング言語, C, C++, Java, C# を使用する17のオープンソースソフトウェアシステムを対象
- 各リリースのコードクローンを検出するために, AIST CCFinderXが使用された

Table 1. Subject systems

Lang	Subject System	SLOC	Duration	No. of Releases
Java	JUnit	2,179-8,785	2003-05-12 to 2009-12-08	20
	CAROL	2,812-11,694	2002-11-12 to 2005-04-13	10
	dnsjava	11,025-23,334	2001-03-29 to 2009-11-21	22
	JabRef	11,352-74,104	2003-11-30 to 2010-04-14	33
	iText	51,860-82,164	2002-03-07 to 2008-01-25	49
C++	KeePass	14,789-43,644	2003-11-17 to 2006-10-14	35
	Notepad++	26,937-81,980	2003-11-25 to 2007-02-04	30
	7-Zip	71,638-100,823	2003-12-11 to 2009-02-03	45
	eMule	6,803-203,780	2002-07-07 to 2010-04-07	73
C	Wget	14,209-40,021	1998-09-23 to 2009-09-22	17
	Conky	7,029-42,060	2005-07-20 to 2010-03-30	70
	ZABBIX	12,468-70,890	2004-03-23 to 2010-01-27	28
	Claws Mail	126,247-203,783	2005-03-19 to 2010-01-31	47
C#	NAnt	686-52,533	2001-07-19 to 2007-12-08	22
	iTextSharp	33,545-163,890	2003-02-04 to 2007-03-08	26
	Process Hacker	10,349-123,878	2008-10-17 to 2010-01-23	38
	ZedGraph	2,439-26,433	2004-08-02 to 2008-12-12	28

リサーチクエスチョン

- **RQ1**：異なる言語やサイズのオープンソースソフトウェアで, クローンの系譜はどのように違うか
- **RQ2**：リリースレベルでのクローン系譜は, 共通の特徴が持っているか. また, 他の系譜より長命を示す系譜はあるか

結果

- i. クローン系譜の種類
- ii. 一貫性を持って変更された系譜 (CCG)
- iii. 生存している系譜 (AG)
- iv. 構文的に類似している系譜 (SSG)
- v. 途絶えた系譜 (DG) と短命なクローン

クローン系譜の種類

1) 生存している系譜 (AG)

- 少なくとも1つのクローングループが最終リリースに含まれている

2) 途絶えた系譜 (DG)

- クローングループは最終リリースに含まれていない

3) 構文的に同様の系譜 (SSG)

- バージョン間に、クローングループには変更がないまたは
フォーマットと識別子だけ変更された

4) 一貫性を持って変更された系譜 (CCG)

- すべてのクローングループには、少なくとも1つの一貫性を持って
変更されたパターン（追加など）がある

一貫性を持って変更された系譜(CCG)

- CCGは10.43%~38.30%
- CCGの平均数は、次の点で異なる：
 - プログラムサイズ：17.71%から24.6%
 - 実装言語：16.84%から26.18%
- CCGの数はそれほど多くない
(平均24.28%)

Table 2. Clone genealogies

System	Total # of Gen.	AG (%)	DG (%)	SSG (%)	CCG (%)
JUnit	127	78.74	21.26	81.89	15.75
CAROL	141	44.68	55.32	56.73	38.30
dnsjava	417	82.97	17.03	85.37	12.23
JabRef	1132	73.41	26.59	66.25	26.06
iText	1568	68.75	31.25	74.62	20.22
Avg. of Java Systems		71.43	28.57	72.67	21.77
KeePass	790	70.76	29.24	73.54	20.63
Notepad++	977	81.99	18.01	73.69	19.86
7-Zip	1427	65.38	34.62	64.62	24.46
eMule	3547	66.08	33.92	59.57	29.86
Avg. of C++ Systems		68.79	31.21	64.32	26.18
Wget	206	57.77	42.23	60.68	28.64
Conky	1328	53.69	46.31	82.45	11.97
ZABBIX	1026	65.79	34.21	49.31	28.65
Claws Mail	2363	85.57	14.43	63.26	27.08
Avg. of C Systems		71.68	28.32	65.43	23.40
NAnt	625	76.96	23.04	55.20	34.08
iTextSharp	2666	77.16	22.84	86.38	10.43
Process Hacker	950	71.79	28.21	73.26	20.32
ZedGraph	374	76.74	23.26	62.83	24.87
Avg. of C# Systems		76.00	24.00	77.55	16.84
Avg. of all Systems		70.33	29.67	66.56	24.28

Table 3. Distribution of genealogies by program size

Program Size	AG (%)	DG (%)	SSG (%)	CCG (%)
<50K	64.65	33.35	76.15	17.71
50K-100K	71.04	28.96	65.32	24.60
>100K	74.58	25.42	69.36	22.78

生存している系譜 (AG)

- 系譜の割合がかなり生存している
(平均**70.33%**)
- 最終リリース前のほんの数回のリリースでかなりクローングループが作成されたため

Table 2. Clone genealogies

System	Total # of Gen.	AG (%)	DG (%)	SSG (%)	CCG (%)
JUnit	127	78.74	21.26	81.89	15.75
CAROL	141	44.68	55.32	56.73	38.30
dnsjava	417	82.97	17.03	85.37	12.23
JabRef	1132	73.41	26.59	66.25	26.06
iText	1568	68.75	31.25	74.62	20.22
Avg. of Java Systems		71.43	28.57	72.67	21.77
KeePass	790	70.76	29.24	73.54	20.63
Notepad++	977	81.99	18.01	73.69	19.86
7-Zip	1427	65.38	34.62	64.62	24.46
eMule	3547	66.08	33.92	59.57	29.86
Avg. of C++ Systems		68.79	31.21	64.32	26.18
Wget	206	57.77	42.23	60.68	28.64
Conky	1328	53.69	46.31	82.45	11.97
ZABBIX	1026	65.79	34.21	49.31	28.65
Claws Mail	2363	85.57	14.43	63.26	27.08
Avg. of C Systems		71.68	28.32	65.43	23.40
NAnt	625	76.96	23.04	55.20	34.08
iTextSharp	2666	77.16	22.84	86.38	10.43
Process Hacker	950	71.79	28.21	73.26	20.32
ZedGraph	374	76.74	23.26	62.83	24.87
Avg. of C# Systems		76.00	24.00	77.55	16.84
Avg. of all Systems		70.33	29.67	66.56	24.28

Table 3. Distribution of genealogies by program size

Program Size	AG (%)	DG (%)	SSG (%)	CCG (%)
<50K	64.65	33.35	76.15	17.71
50K-100K	71.04	28.96	65.32	24.60
>100K	74.58	25.42	69.36	22.78

生存している系譜 (AG)

- 最後の5つのリリース内に作成された系譜とリリース履歴の半分以上存続した系譜を表示
- 最近作成され, 生存している系譜は無視できない (平均23.11%)
- 生存している系譜の大部分は, リリース履歴の半分以上 (平均47.57%) 存続する
- この結果から, リファクタリングを適用する代わりに, クローンの管理を支援する技術を組み込んだ方が重要

Table 4. Alive genealogies

System	AG	AG created within recent five releases	AG that survive more than half of release histories
JUnit	100	31 (31%)	68 (68%)
dnsjava	346	15 (4.34%)	82 (23%)
CAROL	63	47 (74.60%)	17 (26.98%)
JabRef	831	72 (8.66%)	400 (48.13%)
iText	1078	490 (45.45%)	765 (70.96%)
KeePass	559	163 (29.16%)	241(43.11%)
Notepad	801	59 (7.36%)	587 (73.3%)
7-Zip	933	34 (3.64%)	678 (72.66%)
eMule	2344	385 (16.42%)	365 (15.57%)
Wget	119	16 (13.44%)	74 (62.18%)
Conky	713	490 (68.74%)	136(19.07%)
ZABBIX	675	11 (1.62%)	467 (69.18%)
Claws Mail	2022	46 (2.29%)	1335(66.02%)
NAnt	481	275 (57.17%)	62 (12.89%)
iTextSharp	2057	864 (42%)	1094 (53.18%)
Process Hacker	682	236 (34.60%)	158 (33.17%)
ZedGraph	287	23(8.01%)	174 (60.63%)
Avg. of all systems		23.11%	47.57%

構文的に類似している系譜 (SSG)

- クローン系譜の大部分が構文的に類似していることが見られた
(平均**66.56%**)
- CおよびC++のSSGの数は, JavaおよびC#よりも少ない
 - Java : 72.67%
 - C# : 77.55%
 - C : 64.32%
 - C++ : 65.43%
- サイズが50K~100Kのシステムでは, 他の2つのサイズ範囲のシステムと比較して, SSGが少ない

Table 2. Clone genealogies

System	Total # of Gen.	AG (%)	DG (%)	SSG (%)	CCG (%)
JUnit	127	78.74	21.26	81.89	15.75
CAROL	141	44.68	55.32	56.73	38.30
dnsjava	417	82.97	17.03	85.37	12.23
JabRef	1132	73.41	26.59	66.25	26.06
iText	1568	68.75	31.25	74.62	20.22
Avg. of Java Systems		71.43	28.57	72.67	21.77
KeePass	790	70.76	29.24	73.54	20.63
Notepad++	977	81.99	18.01	73.69	19.86
7-Zip	1427	65.38	34.62	64.62	24.46
eMule	3547	66.08	33.92	59.57	29.86
Avg. of C++ Systems		68.79	31.21	64.32	26.18
Wget	206	57.77	42.23	60.68	28.64
Conky	1328	53.69	46.31	82.45	11.97
ZABBIX	1026	65.79	34.21	49.31	28.65
Claws Mail	2363	85.57	14.43	63.26	27.08
Avg. of C Systems		71.68	28.32	65.43	23.40
NAnt	625	76.96	23.04	55.20	34.08
iTextSharp	2666	77.16	22.84	86.38	10.43
Process Hacker	950	71.79	28.21	73.26	20.32
ZedGraph	374	76.74	23.26	62.83	24.87
Avg. of C# Systems		76.00	24.00	77.55	16.84
Avg. of all Systems		70.33	29.67	66.56	24.28

Table 3. Distribution of genealogies by program size

Program Size	AG (%)	DG (%)	SSG (%)	CCG (%)
<50K	64.65	33.35	76.15	17.71
50K-100K	71.04	28.96	65.32	24.60
>100K	74.58	25.42	69.36	22.78

構文的に類似している系譜 (SSG)

- SSGとAGの関係を表示
- 平均的に, 69.04%SSGが対象システムの最終リリースに達した
- 一方, 平均的に66.61%の生存している系譜は, 構文的に変化しなかった
- 構文的に変化がないクローングループは、ソフトウェアシステムの開発中に除去される可能性が低い
- プログラムのサイズが大きくなると, 合計SSGに対する生存しているSSGの割合も増加する

Table 5. Syntactically similar genealogies

System	Alive SSG	% of alive SSG of total SSG	% of alive SSG of total AG
JUnit	85	81.73	85.00
CAROL	34	42.50	53.97
dnsjava	310	87.08	89.60
JabRef	530	70.67	63.78
iText	811	69.32	75.23
Avg. of Java Systems		71.95	73.20
KeePass	402	69.19	71.91
Notepad++	568	78.89	70.91
7-Zip	586	63.56	62.8
eMule	1262	59.73	53.83
Avg. of C++ Systems		64.99	60.77
Wget	60	48.00	50.42
Conky	590	53.88	82.74
ZABBIX	279	55.13	41.33
Claws Mail	1220	81.60	60.33
Avg. of C Systems		66.72	60.90
NAnt	235	68.12	48.86
iTextSharp	1734	75.29	84.30
Process Hacker	505	72.56	74.04
ZedGraph	175	74.47	60.98
Avg. of C# Systems		74.02	75.53
Avg. of all Systems		69.04	66.61

Table 6. Syntactically similar genealogies by program size

Program Size	% of alive SSG of total SSG	% of alive SSG of total AG
<50K	64.29	75.72
50K-100K	68.18	62.70
>100K	71.45	66.44

途絶えた系譜（DG）と短命なクローン

- リリース数のシステムで途絶えた系譜がどのくらい存続したかを知りたい
- K-volatile genealogy : Kバージョンの内に途絶えた系譜

途絶えた系譜 (DG) と短命なクローン

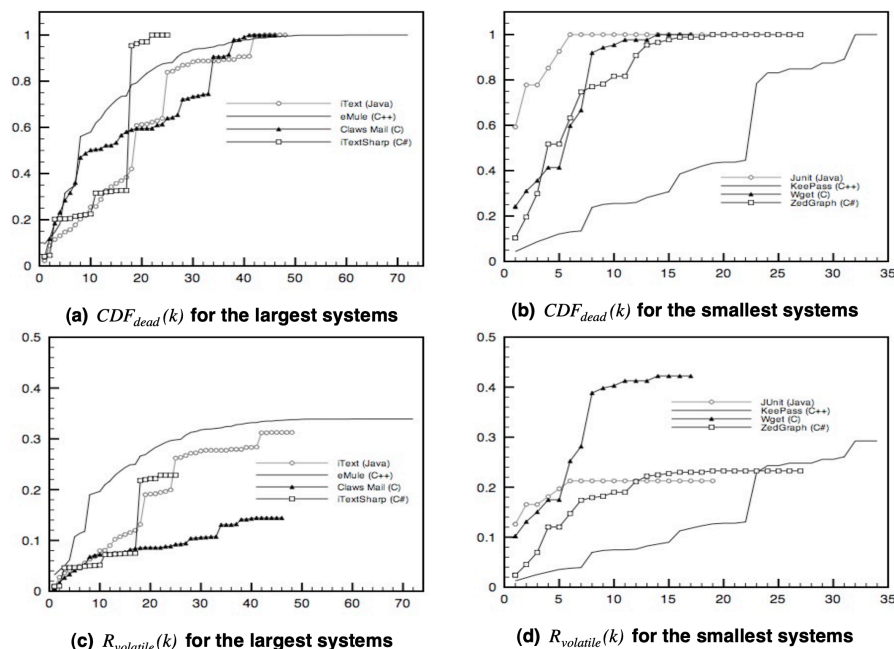


Figure 2. $CDF_{dead}(k)$ and $R_{volatile}(k)$ for the largest and smallest systems of each language category

各言語で最大のシステム

- iText : 6リリース以内に途絶えた系譜の16%が消えた
- Claws Mail : 5リリース以内に28%
- eMule : 5リリース以内に33%

各言語で最小のシステム

- Junit : 6リリース以内に100%
- KeePass Password Safe : 5リリース以内に12%
- Wget : 6リリース以内に60%
- ZedGraph : 5リリース以内に52%

途絶えた系譜（DG）と短命なクローン

- リリースレベルでも, 短命なクローンの数は
無視できない
- 積極的なリファクタリングを行うのはおそらく
あまり価値がない

まとめ

- 4つの異なるプログラミング言語を使用する17のオープンソースソフトウェアシステムを対象にして, コードクローンの系譜を評価するための実証的研究が行われた
- 多くの系譜は最終のリリースまで生存しており, 長命
- 生存している系譜の67%には, 行の追加や削除, 識別子の名前変更が含まれていなかった
- 一貫性を持って変更されたクローン系譜は約11%から38%が観察された
- 途絶えた系譜のうち, 多くの場合は数回のリリースの後で除去された (短命)
- 開発言語やプロジェクトサイズは, クローンの進化にあまり影響がない

所感

- 系譜のことがもっと理解できた
- リファクタリングのことがあまり分からなく, もっと勉強が必要だと思う