Searching for Better Configurations:

A Rigorous Approach to Clone Evaluation

Tiantian Wang
Harbin Institute of Technology

Mark Harman
Yue Jia
Jens Krinke
CREST, University College London

Searching for Better Configurations:

A Rigorous Approach to Clone Evaluation

Tiantian Wang

Harbin Institute of Technology

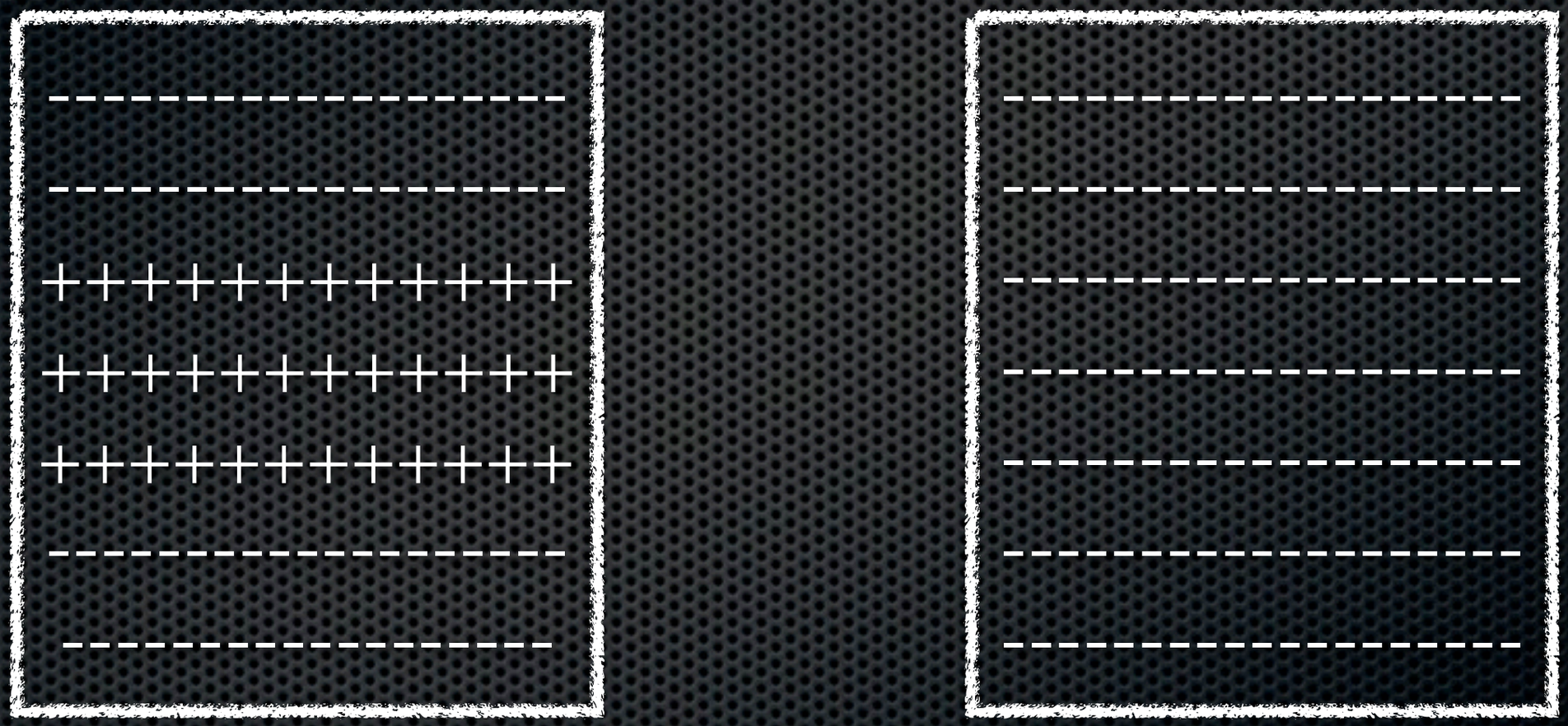
Mark Harman

Yue Jia

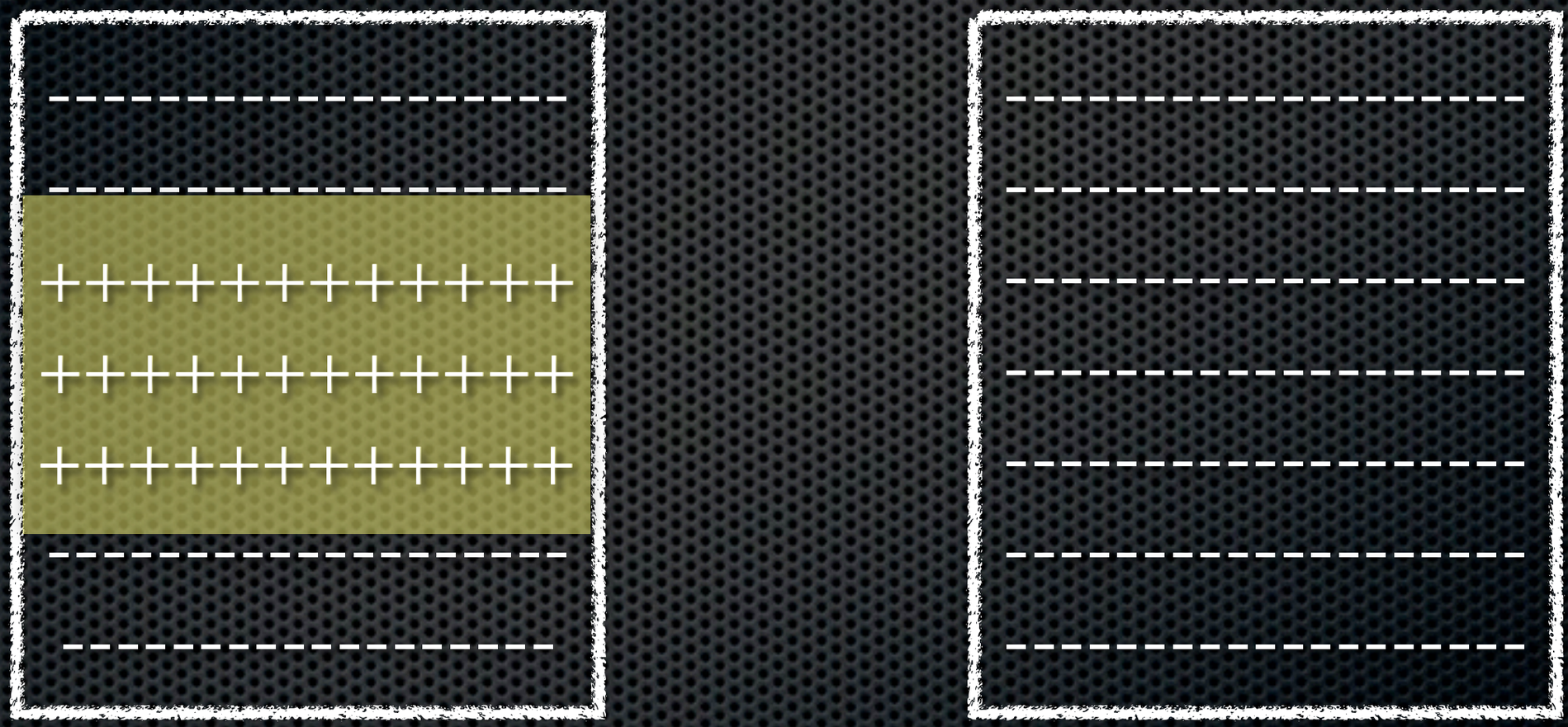
Jens Krinke

CREST, University College London

Software Clone



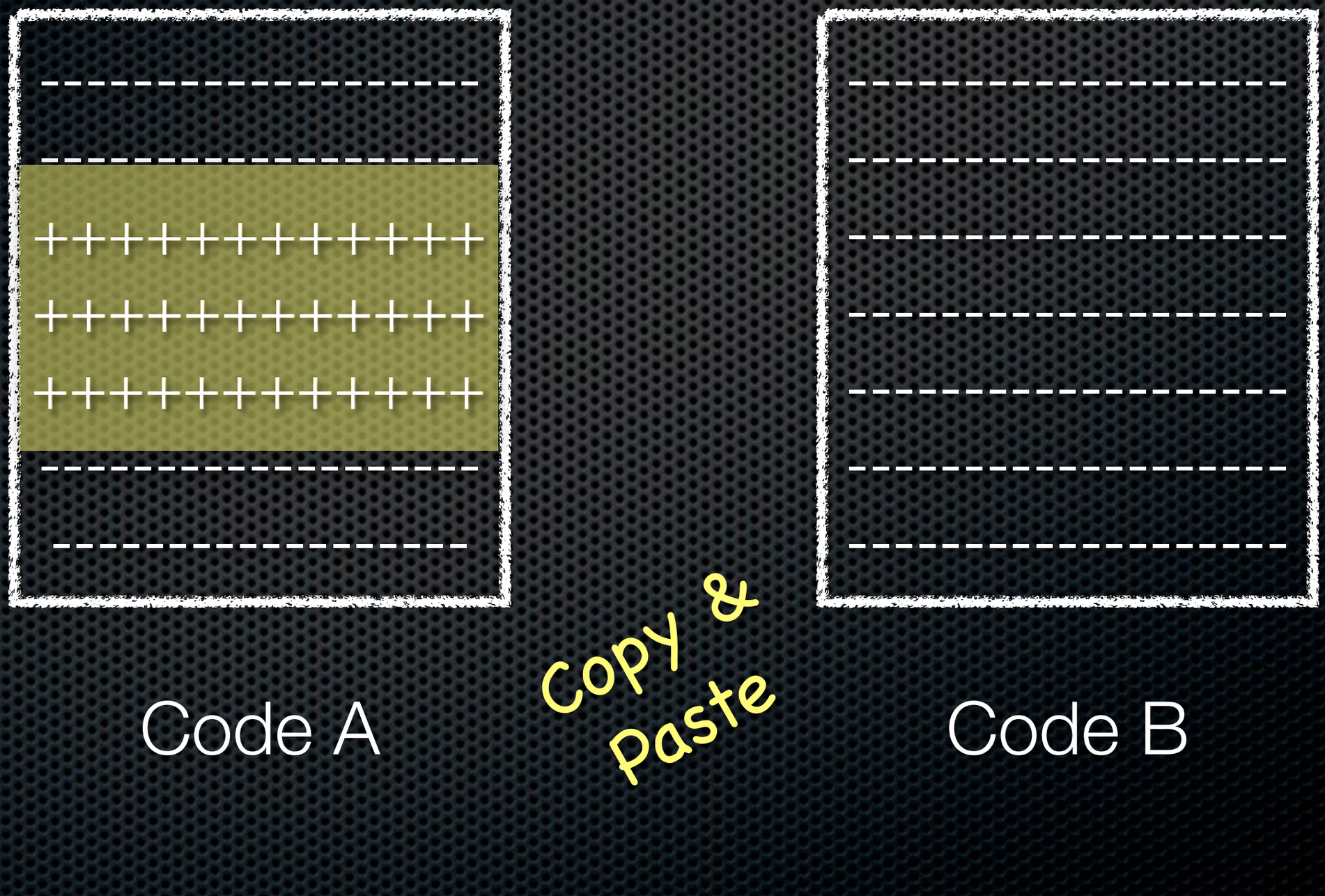
Software Clone



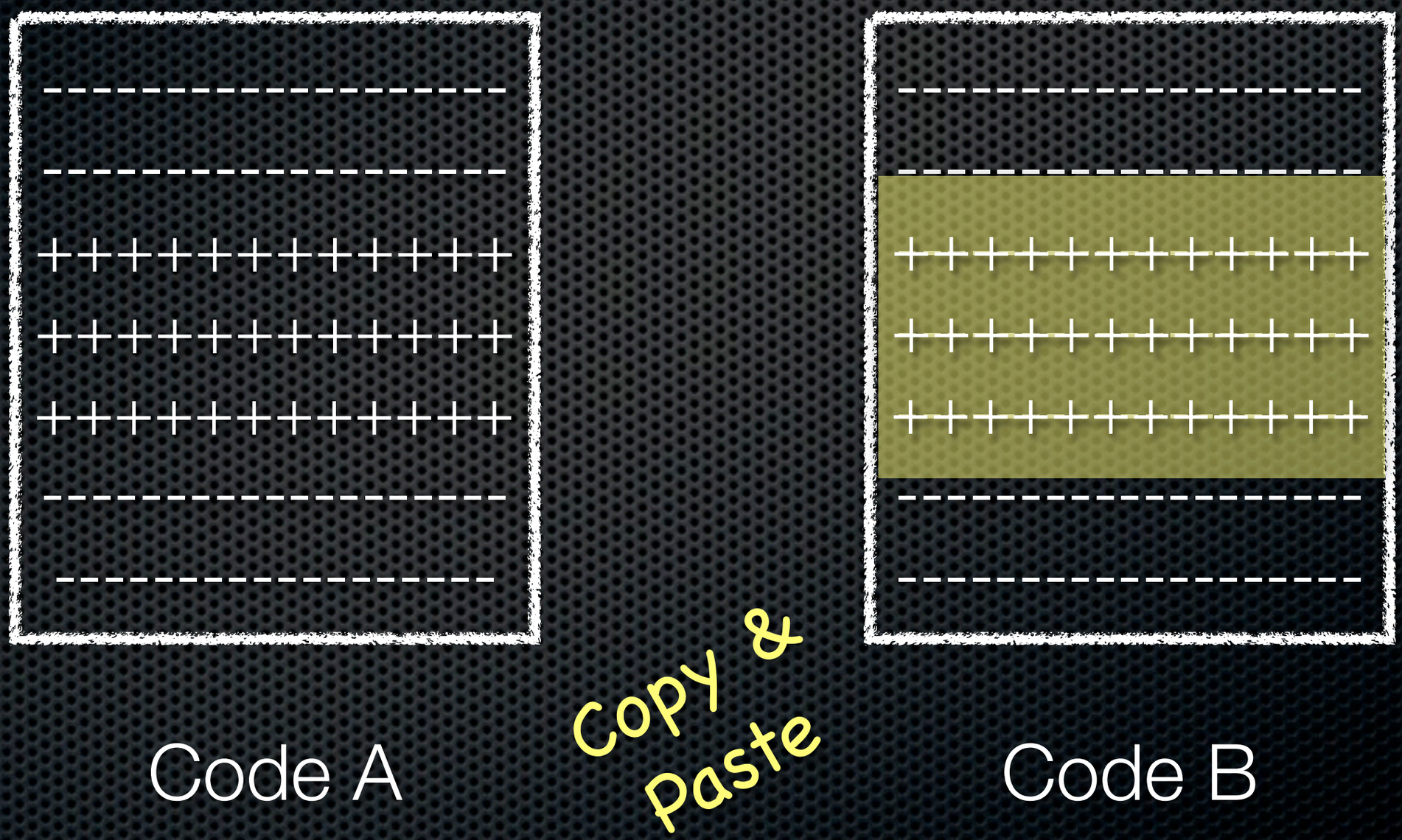
Code A

Code B

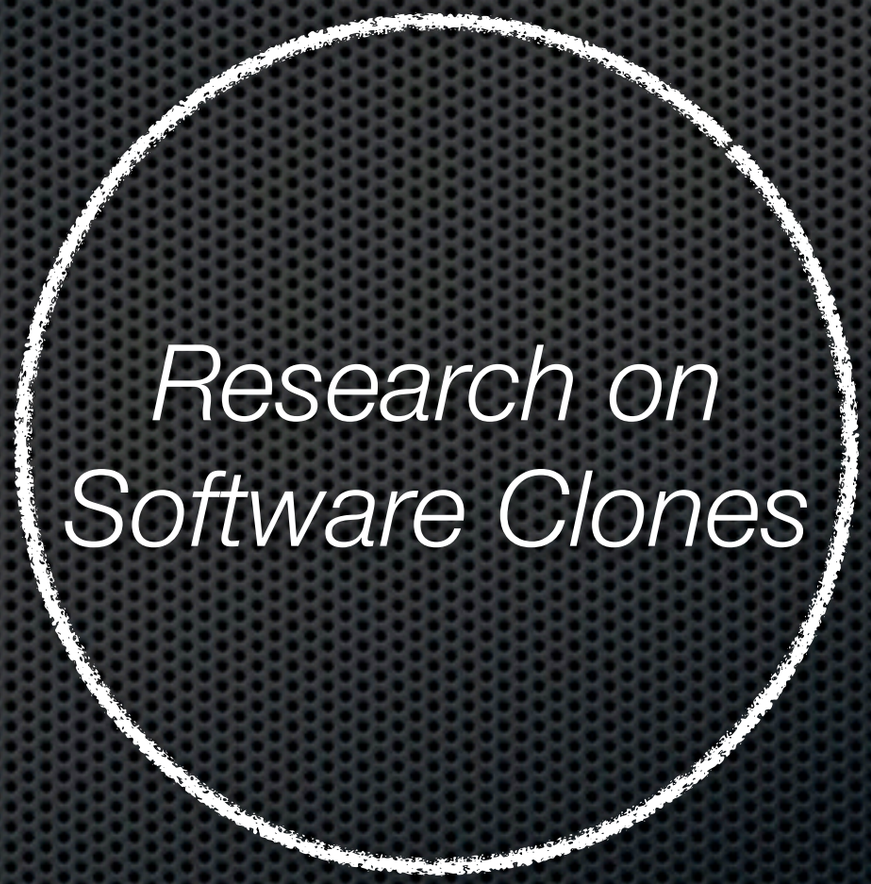
Software Clone



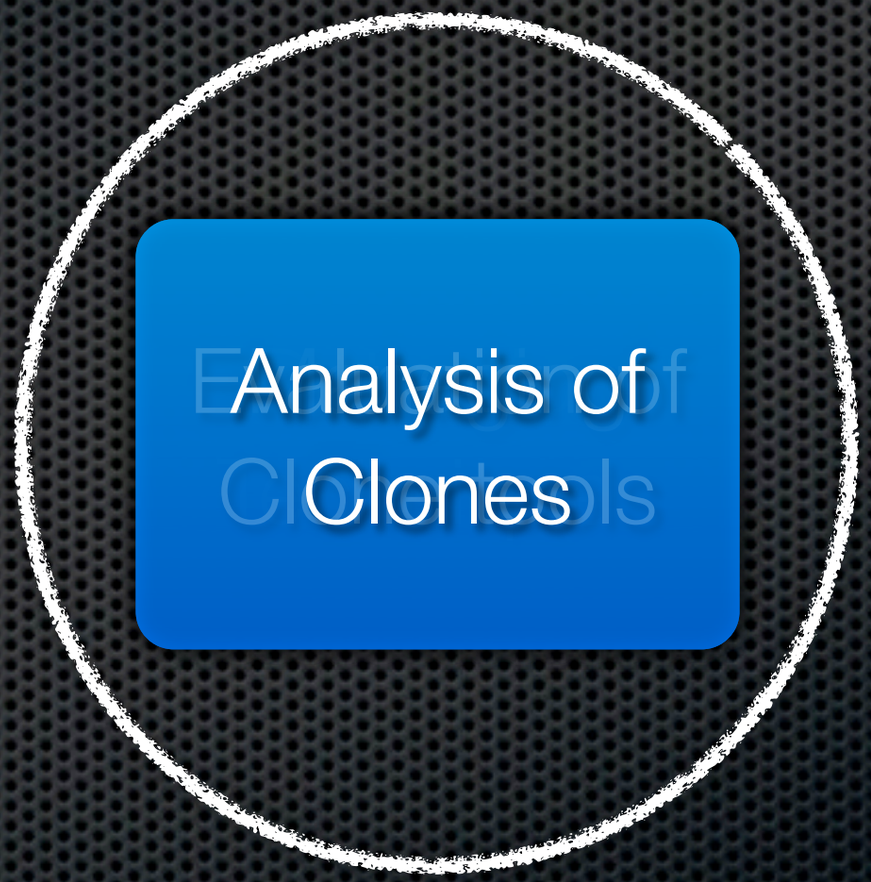
Software Clone



Software Clone



Software Clone



Software Clone

Detection Techniques

Analysis of Clones

Evaluation of Clone tools

Managing Clones

Software Clone

Detection Techniques

Analysis of Clones

Configuration Choice?

Evaluation of Clone tools

Managing Clones

The confounding configuration choice problem

How do we know that differences observed are related to the properties of clones, rather than the properties of configuration choices?

```
func1()
{
   int a, b, c;
   c = a + b;
   printf(“%d”, c);
}
```

```
func1()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func2()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func1()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func2()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func3()
{
    int a, b, d;
    d = a + b;
    printf("%d", d);
}
```

```
func1()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func2()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func3()
{
    int a, b, d;
    d = a + b;
    printf("%d", d);
}
```

```
func4()
{
    int a, b, d;
    d = a + b;
    d = d + 1;
    printf("%d", d);
}
```

```
func1()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func2()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func3()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func4()
{
    int a, b, d;
    d = a + b;
    printf("%d", d);
}
```

```
func5()
{
    int a, b, d;
    d = a + b;
    d = d + 1;
    printf("%d", d);
}
```

```
func1()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func2()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func3()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func4()
{
    int a, b, d;
    d = a + b;
    printf("%d", d);
}
```

```
func5()
{
    int a, b, d;
    d = a + b;
    d = d + 1;
    printf("%d", d);
}
```

...

```
func1()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func2()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func3()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func4()
{
    int a, b, d;
    d = a + b;
    printf("%d", d);
}
```

```
func5()
{
    int a, b, d;
    d = a + b;
    d = d + 1;
    ...
}
```

...

```
func1()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func2()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func3()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func4()
{
    int a, b, d;
    d = a + b;
    printf("%d", d);
}
```

```
func5()
{
    int a, b, d;
    d = a + b;
    d = d + 1;
    ...
}
```

...

Not a clone

```
func1()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func2()
{
    int a, b, c;
    c = a + b;
    printf("%d", c);
}
```

```
func3()
{
    int a, b, d;
    d = a + b;
    printf("%d", d);
}
```

```
func4()
{
    int a, b, d;
    d = a + b;
    d = d + 1;
    printf("%d", d);
}
```

```
func5()
{
    int a, b, d;
    d = a + b;
    d = d + 1;
    ...
}
```

Clone ?

...

Not a clone

The confounding configuration choice problem

The confounding configuration choice problem

Different configurations

Different

Numbers of clone

Lengths of clone

Types of clone

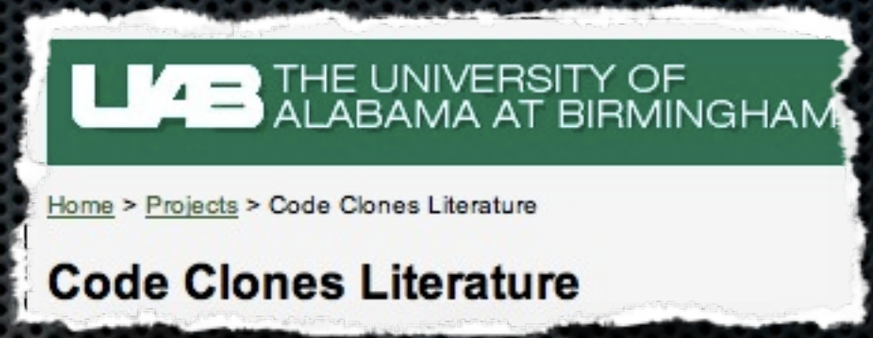
Execution times

...

The confounding configuration choice problem

- Two kinds of evidence
  - A detailed review of the literature
  - Empirical evidence
- A SBSE approach to search for better configurations

Analysis of the Clone Literature



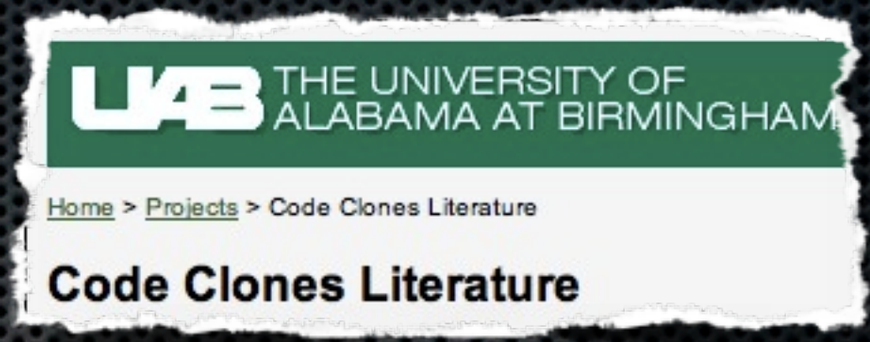
274 papers

Analysis of the Clone Literature

185 papers have empirical study

89 no empirical study

Analysis of the Clone Literature



113 papers aware that tool configuration may have affected the results in the paper

89 no empirical study

Analysis of the Clone Literature

185 papers have empirical study

89 no empirical study

Analysis of the Clone Literature

89 no empirical study

57 papers contain a ‘threats to validity’ section

Analysis of the Clone Literature

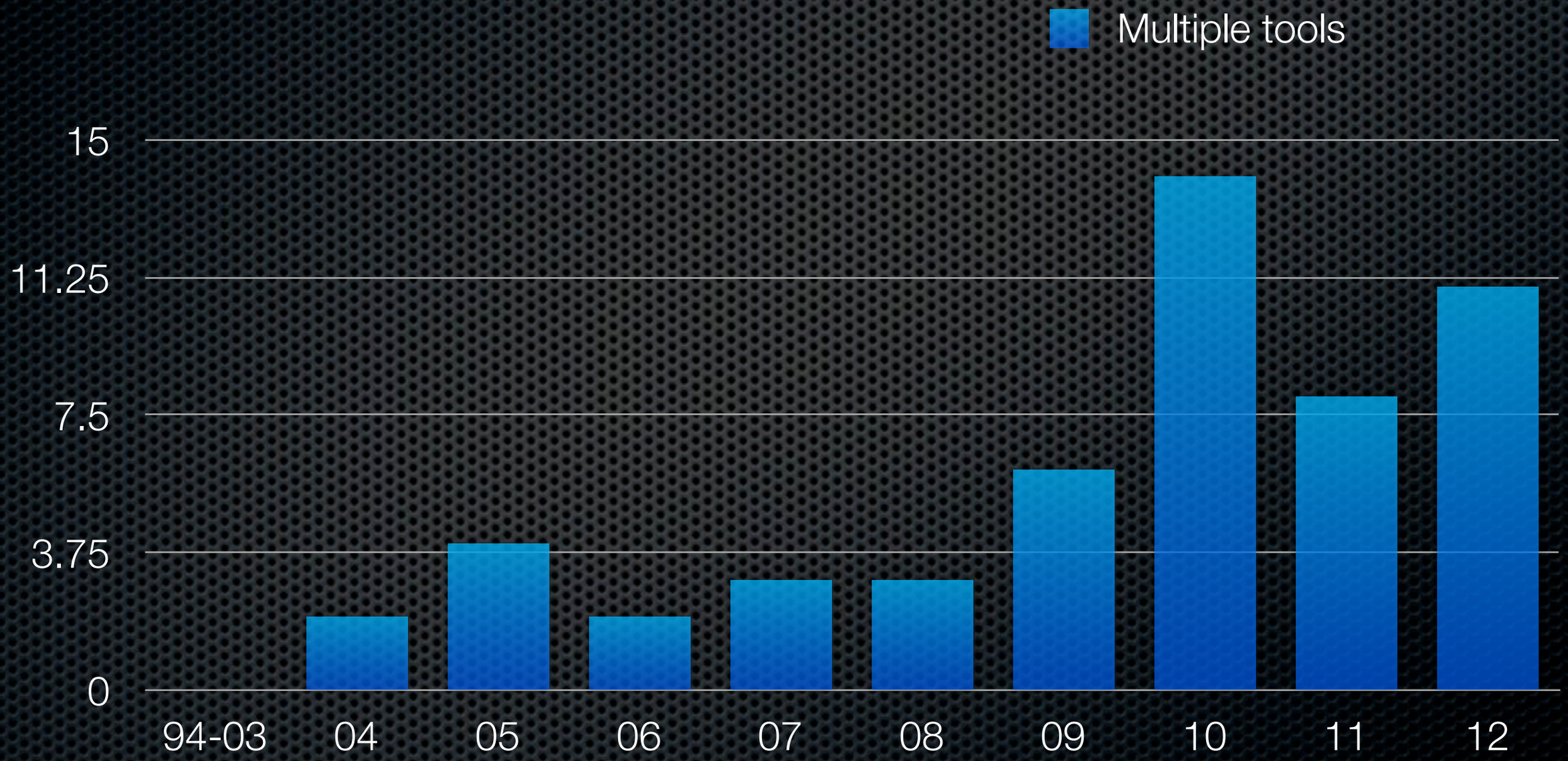
89 no empirical study

43 papers consider this is an important issue

Analysis of the Clone Literature

- Multiple tools

Analysis of the Clone Literature



Metrics for configuration choices

Varied: several different configurations are used to attempt to cater for confounding configuration effects.

Justified: configurations are reported, together with some explanation as to why they have been selected.

Default: the tools' default configurations are used.

Arbitrary: configurations are reported but with neither justification nor explanation.

Undefined: configurations are not reported

Metrics for configuration choices

Varied: several different configurations are used to attempt to cater for confounding configuration effects.

Justified: configurations are reported, together with some explanation as to why they have been selected.

Default: the tools' default configurations are used.

Arbitrary: configurations are reported but with neither justification nor explanation.

Undefined: configurations are not reported

Metrics for configuration choices

Varied: several different configurations are used to attempt to cater for confounding configuration effects.

Justified: configurations are reported, together with some explanation as to why they have been selected.

Default: the tools' default configurations are used.

Arbitrary: configurations are reported but with neither justification nor explanation.

Undefined: configurations are not reported

Metrics for configuration choices

Varied: several different configurations are used to attempt to cater for confounding configuration effects.

Justified: configurations are reported, together with some explanation as to why they have been selected.

Default: the tools' default configurations are used.

Arbitrary: configurations are reported but with neither justification nor explanation.

Undefined: configurations are not reported

Metrics for configuration choices

Varied: several different configurations are used to attempt to cater for confounding configuration effects.

Justified: configurations are reported, together with some explanation as to why they have been selected.

Default: the tools' default configurations are used.

Arbitrary: configurations are reported but with neither justification nor explanation.

Undefined: configurations are not reported

Metrics for configuration choices

Varied: several different configurations are used to attempt to cater for confounding configuration effects.

Justified: configurations are reported, together with some explanation as to why they have been selected.

Default: the tools' default configurations are used.

Arbitrary: configurations are reported but with neither justification nor explanation.

Undefined: configurations are not reported

Other tools

	Und	Abr	Def	Jus	Var	Total
Total	9	0	15	15	4	43

Other tools

Own tools	Und	Abr	Def	Jus	Var	Total
N/A						
Und						
Abr						
Def						
Jus						
Var						
Total	9	0	15	15	4	43

Other tools

Own tools	Und	Abr	Def	Jus	Var	Total
N/A						
Und						
Abr						
Def						
Jus						
Var						
Total	9	0	15	15	4	43

Other tools

Own tools	Und	Abr	Def	Jus	Var	Total
N/A	2					
Und	2					
Abr						
Def						
Jus						
Var					2	
Total	9	0	15	15	4	43

Other tools

Own tools	Und	Abr	Def	Jus	Var	Total
N/A	2					
Und	2					
Abr						
Def						
Jus						
Var					2	
Total	9	0	15	15	4	43

Other tools

Own tools	Und	Abr	Def	Jus	Var	Total
N/A	2					
Und	2					
Abr						
Def						
Jus						
Var					2	
Total	9	0	15	15	4	43

Other tools

Own tools	Und	Abr	Def	Jus	Var	Total
N/A	2					
Und	2					
Abr						
Def						
Jus						
Var					2	
Total	9	0	15	15	4	43

Other tools

Own tools	Und	Abr	Def	Jus	Var	Total
N/A	2					
Und	2					
Abr						
Def						
Jus						
Var					2	
Total	9	0	15	15	4	43

Other tools

Own tools	Und	Abr	Def	Jus	Var	Total
N/A	2	0	4	3	1	10
Und	2	0	0	0	0	2
Abr	2	0	0	0	0	2
Def	0	0	1	0	0	1
Jus	1	0	5	8	1	15
Var	2	0	5	4	2	13
Total	9	0	15	15	4	43

Other tools

Own tools	Und	Abr	Def	Jus	Var	Total
N/A	2	0	4	3	1	10
Und	2	0	0	0	0	2
Abr	2	0	0	0	0	2
Def	0	0	1	0	0	1
Jus	1	0	5	8	1	15
Var	2	0	5	4	2	13
Total	9	0	15	15	4	43

Other tools

Own tools	Und	Abr	Def	Jus	Var	Total
N/A	2	0	4	3	1	10
Und	2	0	0	0	0	2
Abr	2	0	0	0	0	2
Def	0	0	1	0	0	1
Jus	1	0	5	8	1	15
Var	2	0	5	4	2	13
Total	9	0	15	15	4	43

Other tools

Own tools	Und	Abr	Def	Jus	Var	Total
N/A	2	0	4	3	1	10
Und	2	0	0	0	0	2
Abr	2	0	0	0	0	2
Def	0	0	1	0	0	1
Jus	1	0	5	8	1	15
Var	2	0	5	4	2	13
Total	9	0	15	15	4	43

Other tools

Own tools	Und	Abr	Def	Jus	Var	Total
N/A	2	0	4	3	1	10
Und	2	0	0	0	0	2
Abr	2	0	0	0	0	2
Def	0	0	1	0	0	1
Jus	1	0	5	8	1	15
Var	2	0	5	4	2	13
Total	9	0	15	15	4	43

Other tools

Own tools	Und	Abr	Def	Jus	Var	Total
N/A	2	0	4	3	1	10
Und	2	0	0	0	0	2
Abr	2	0	0	0	0	2
Def	0	0	1	0	0	1
Jus	1	0	5	8	1	15
Var	2	0	5	4	2	13
Total	9	0	15	15	4	43

The confounding configuration choice problem

The confounding configuration choice problem

Tool Set TS = { T_1 , … , T_n }

T_1

T_2

T_3

T_4

The confounding configuration choice problem

Tool Set $TS = \{ T_1 , \dots , T_n \}$

Configuration Set $X = \{ X_1 , \dots , X_n \}$

Min Tokens

Min blocks

Min Lines

IgnoreLiterals

IgnoreIdentifiers

Ignore ...

T_1

T_2

T_3

T_4

The confounding configuration choice problem

Tool Set $TS = \{ T_1, \ldots, T_n \}$

Configuration Set $X = \{ X_1, \ldots, X_n \}$

Subject Set $SS = \{ S_1, \ldots, S_m \}$

The confounding configuration choice problem

Tool Set $TS = \{ T_1 , \ldots , T_n \}$

Configuration Set $X = \{ X_1 , \ldots , X_n \}$

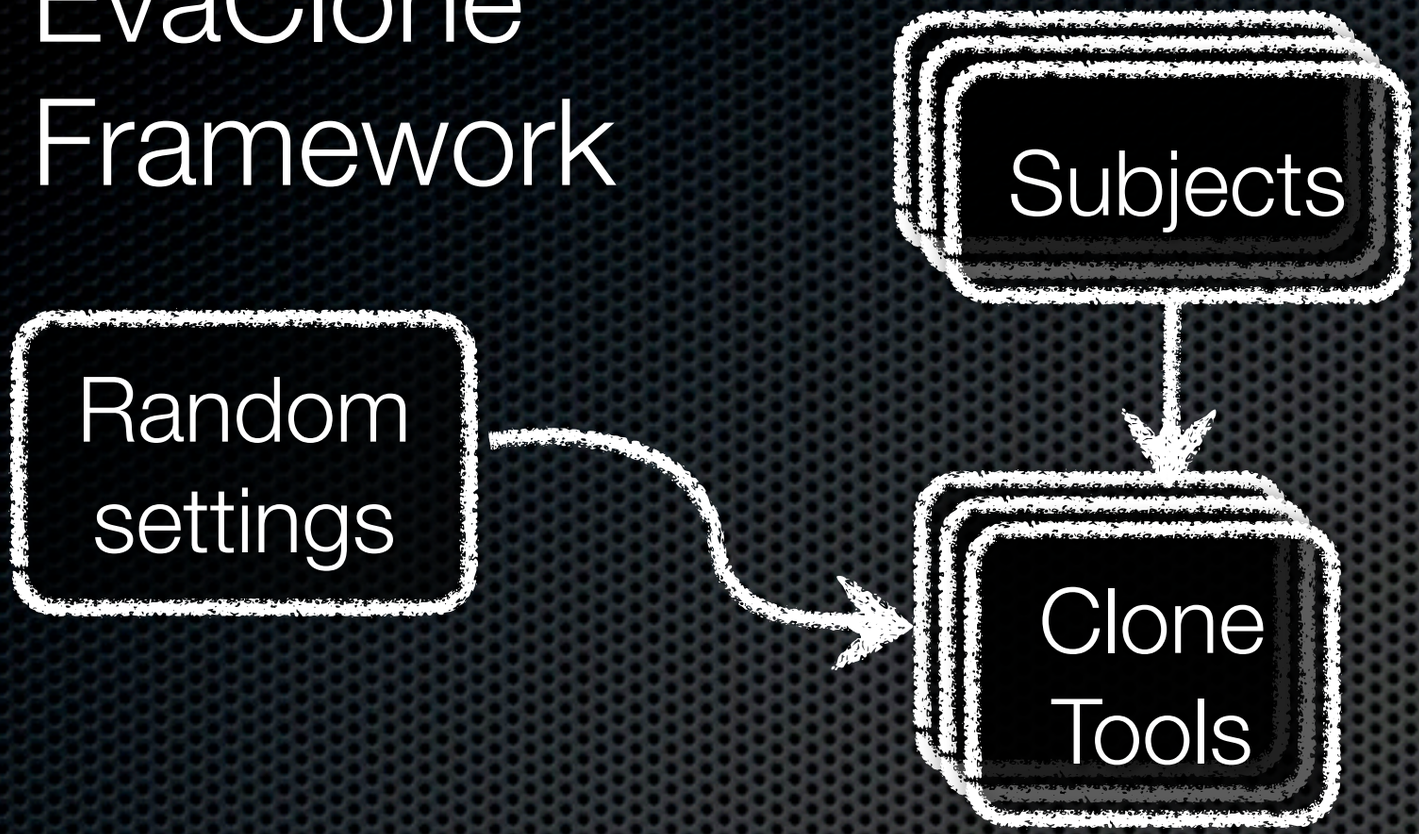
Subject Set $SS = \{ S_1 , \ldots , S_m \}$

- The clone detection tool configuration problem is to automatically search for configuration settings, *X*, for *TS* in the configuration search space Ω , subject to:

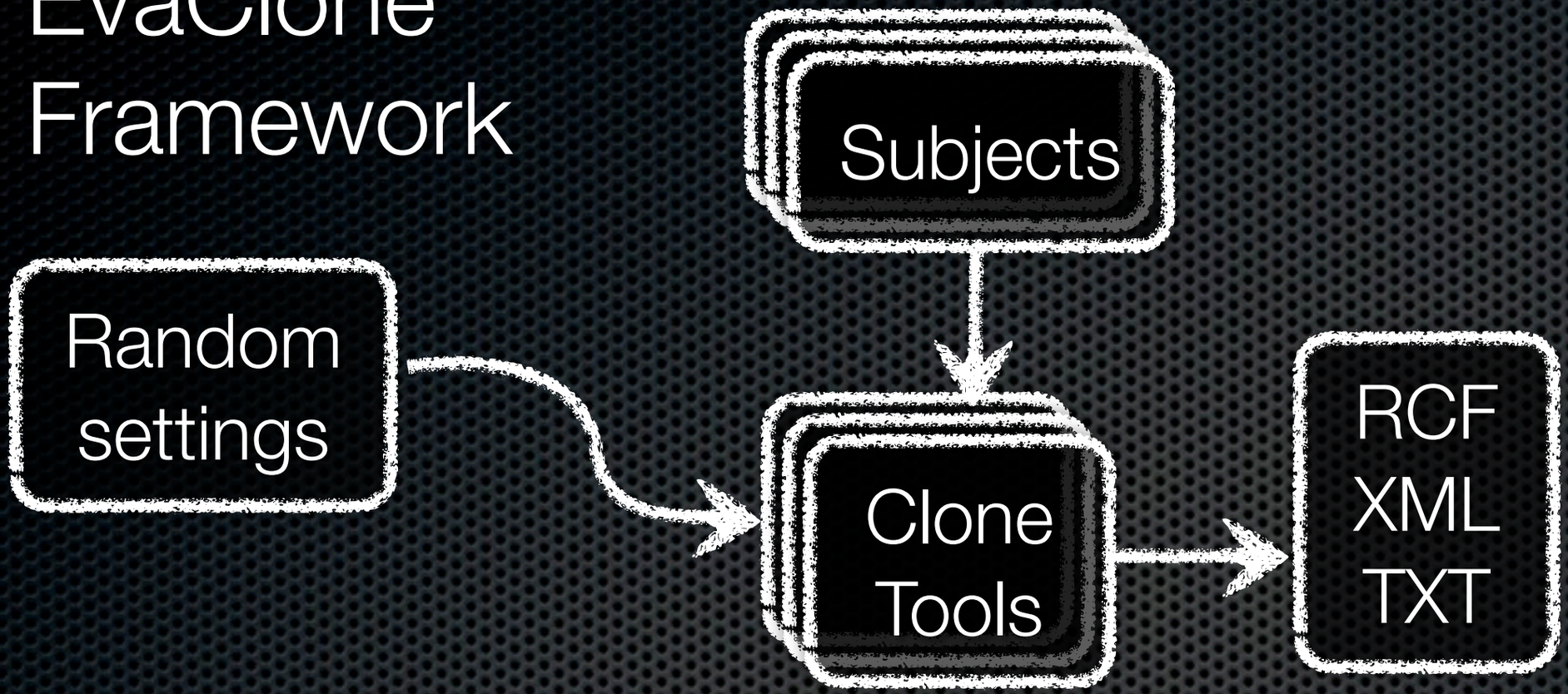
maximise $f (TS(X) , SS)$

EvaClone Framework

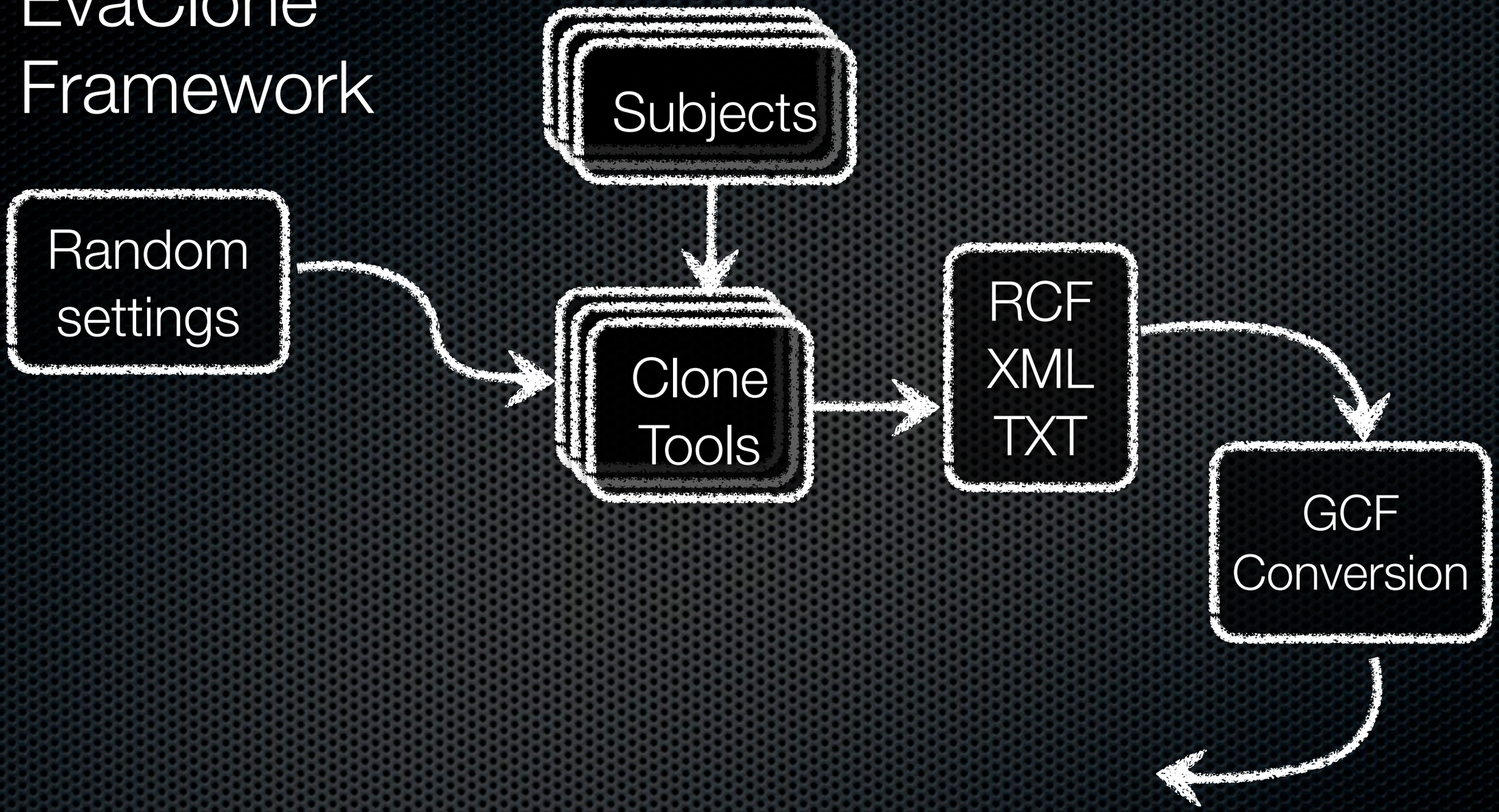
EvaClone Framework



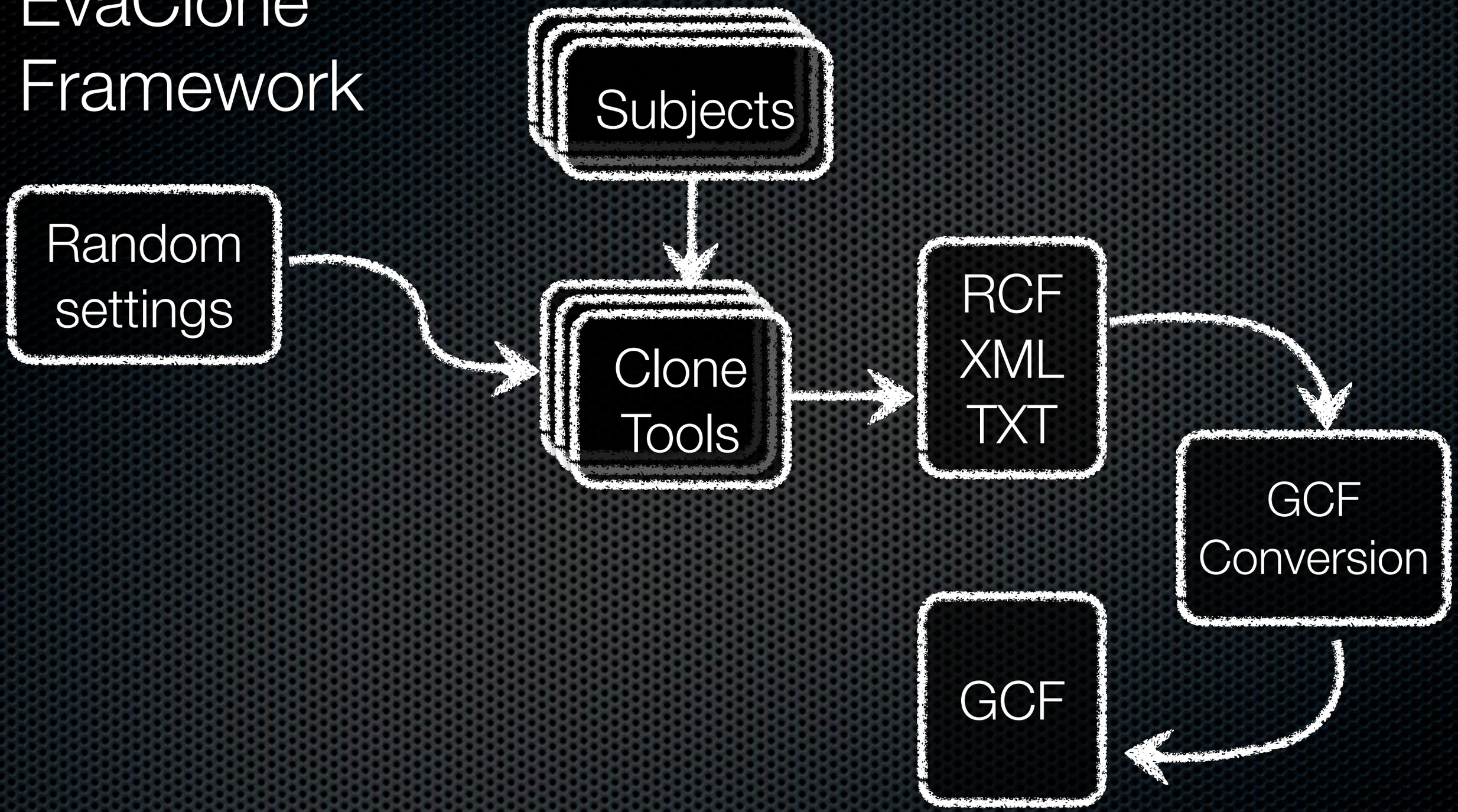
EvaClone Framework



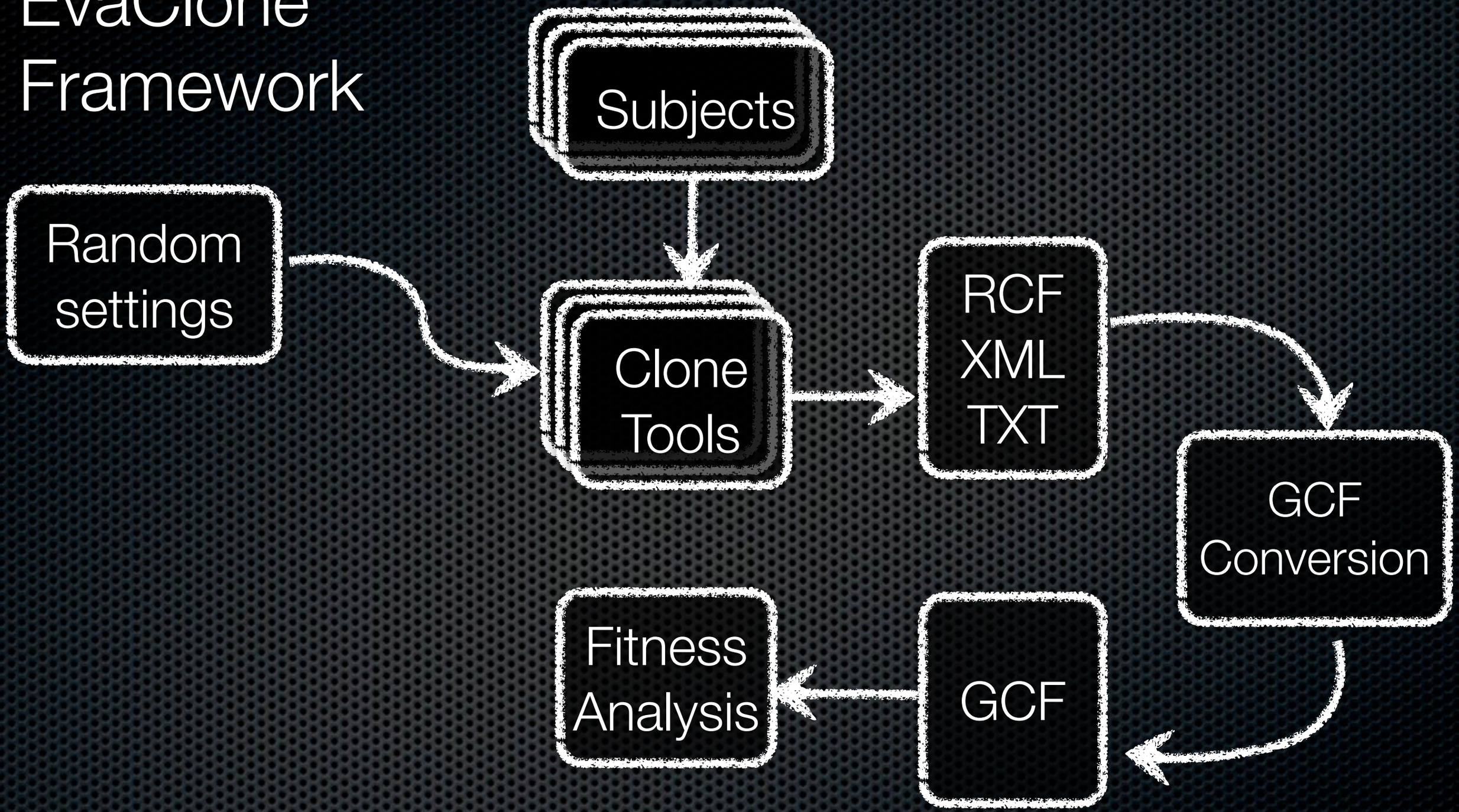
EvaClone Framework



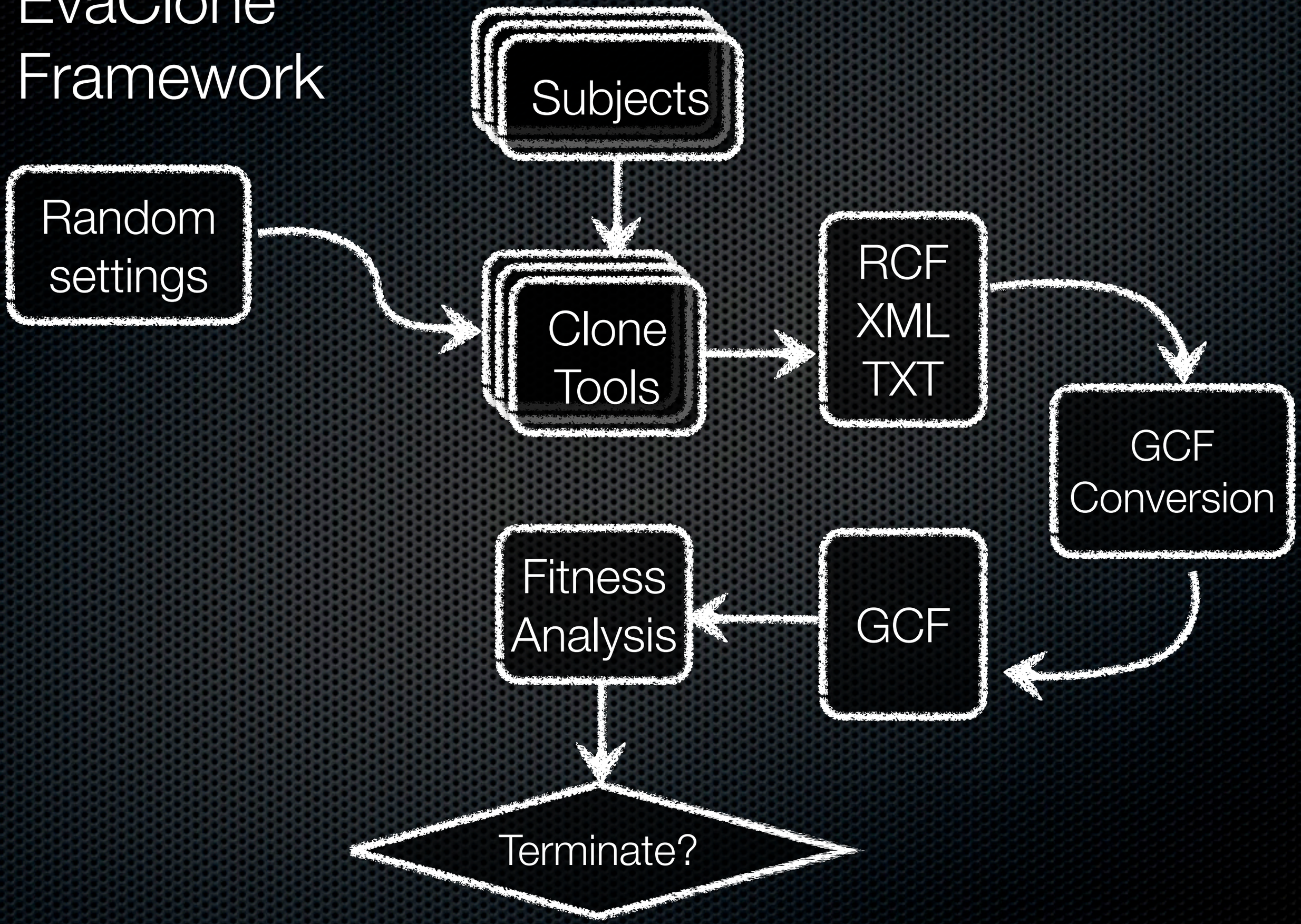
EvaClone Framework



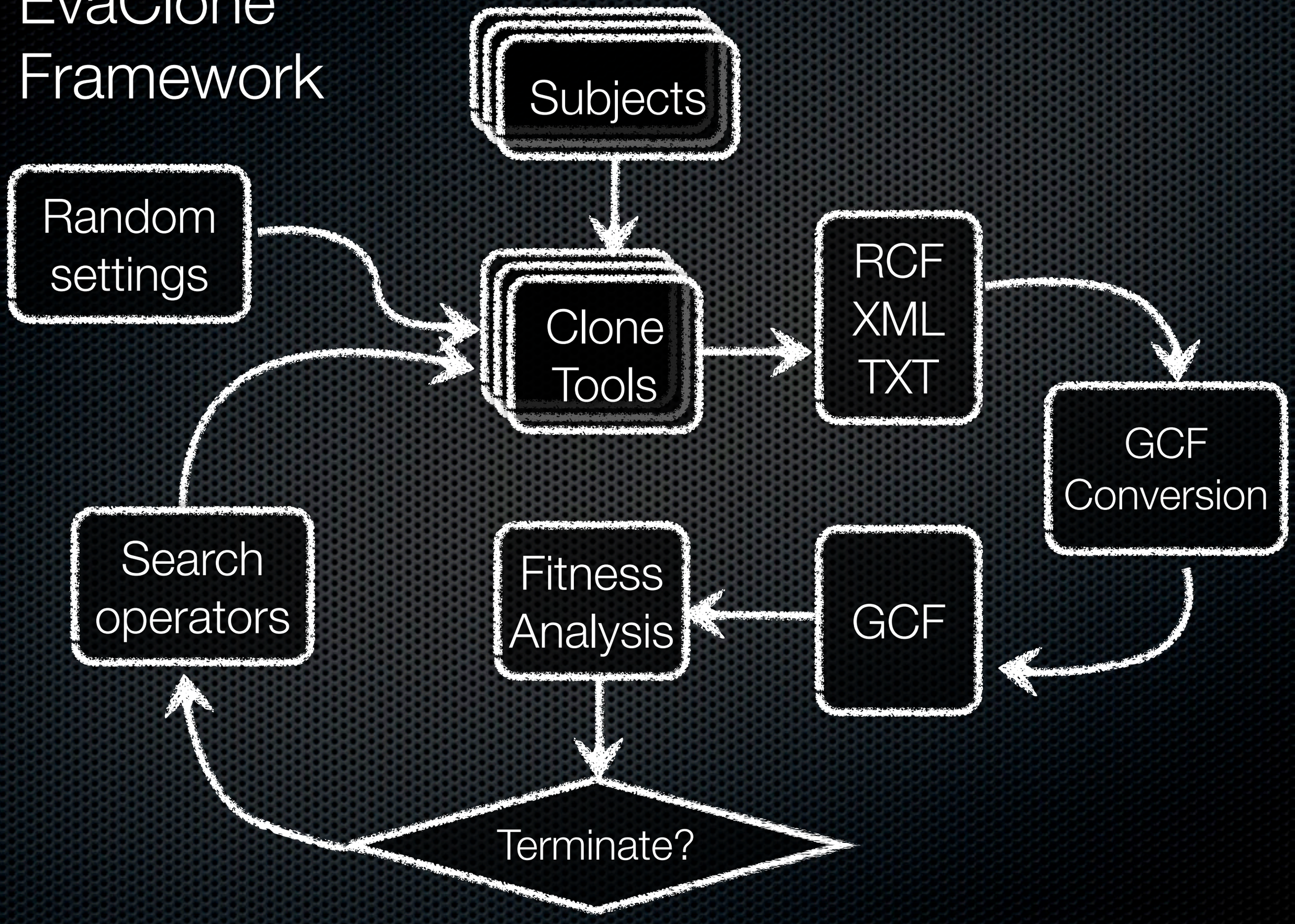
EvaClone Framework



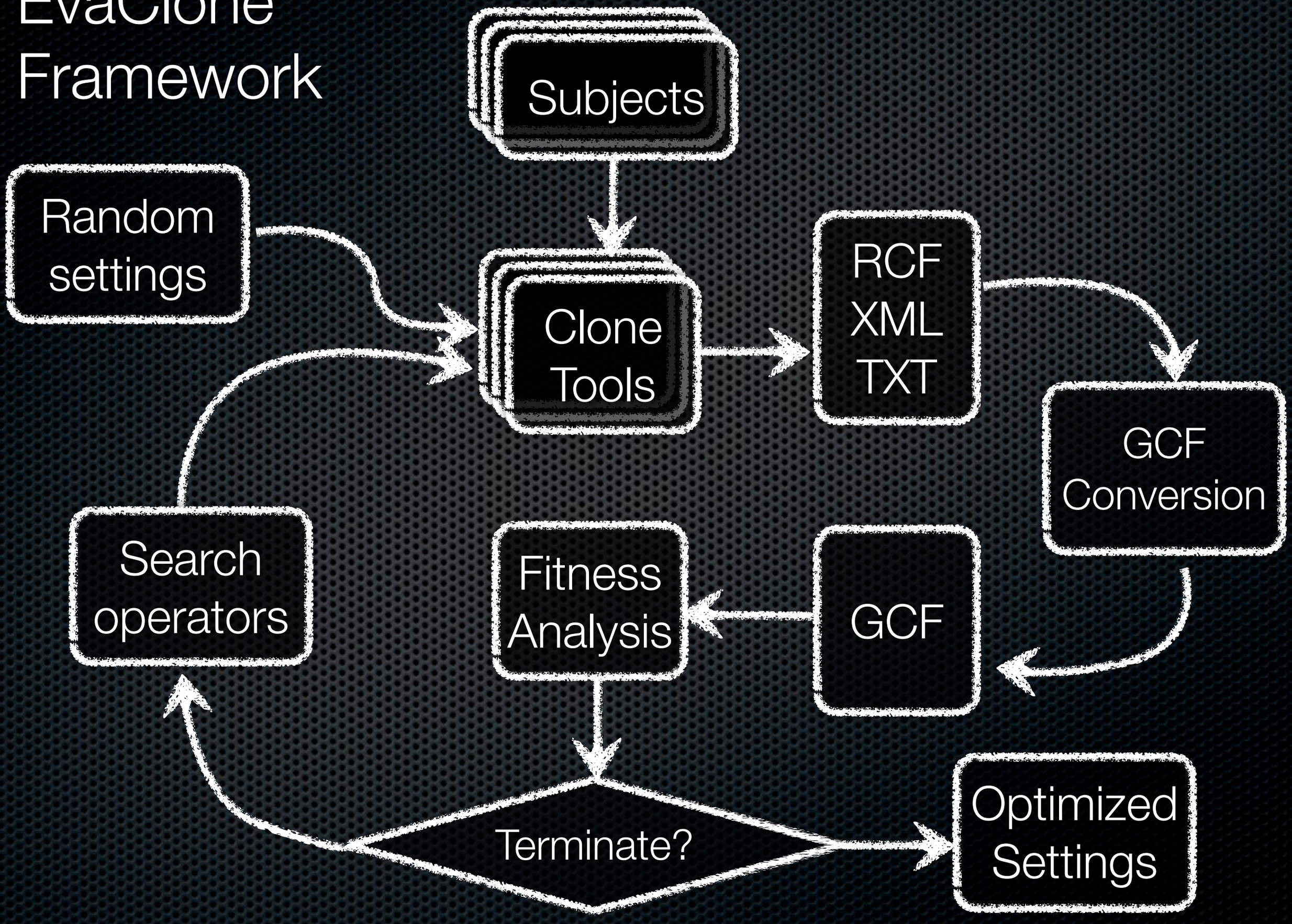
EvaClone Framework



EvaClone Framework



EvaClone Framework



EvaClone Framework

Subjects

Random settings

Search operators

Fitness Evaluation Unit

Clone Tools

RCF XML TXT

GCF Conversion

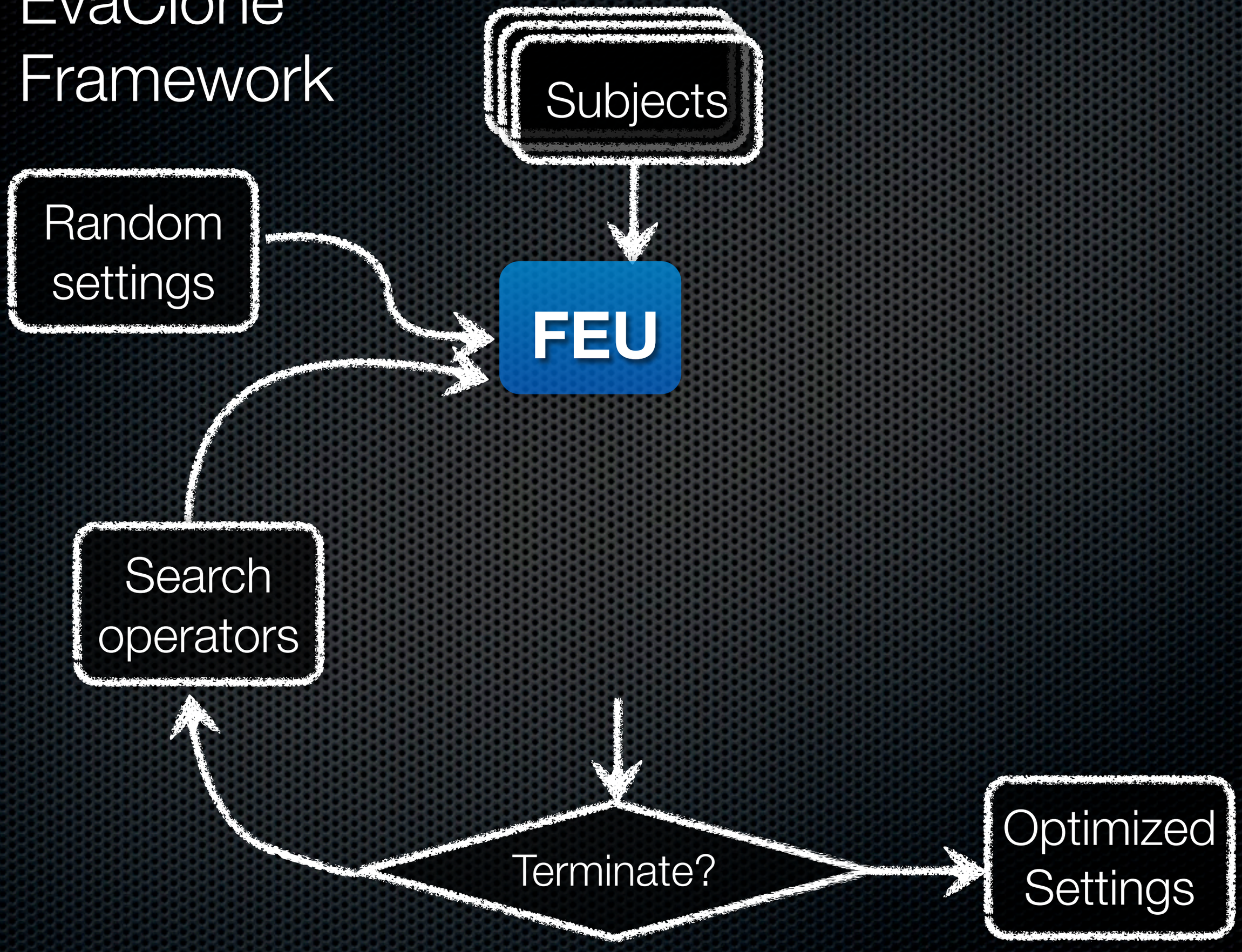
Fitness Analysis

GCF

Terminate?

Optimized Settings

EvaClone Framework



Cloud EvaClone Framework

Subjects

Random settings

FEU FEU FEU
FEU FEU FEU
FEU FEU

X 1000

amazon web services™

Search operators

Terminate?

Optimized Settings

Subject Tools

Table 4: The Clone Detection Tools Used

Tool	Approach	Support Language	Type
PMD's CPD 5.0 [6]	Token	C, C++, C#, Java	1,2
IClones 0.1 [10]	Token	Java, C, C++, ADA	1,2,3
CCFinder 10.2.7.4 [18]	Token	C, C++, Java, COBOL	1,2,3
ConQAT 2011.9 [17]	Token	independent	1,2
Simian 1.5.0.13 [14]	Text	independent	1,2
NiCAD 3.2 [33]	Parser	C, C#, Java, Python	1,2,3

Subject Systems

The Bellon Suite of Eight Benchmark Subjects

C Sets

- Weltab 10 k loc
- Cook 80 k loc
- Snns 120 k loc
- Psql 230 k loc

Java Sets

- Javadoc 14 k loc
- Ant 34 k loc
- Jdtcore 140 k loc
- Swing 200 k loc

Fitness Function

Maximise tool agreement

Multiple Clone Tools (53 papers)

Agreement / Disagreement (80%)

Multiple Clone Tools (53 papers)

Agreement / Disagreement (80%)

- Use the other tools to evaluate the given clone detection tool (58%)

Authors' tool

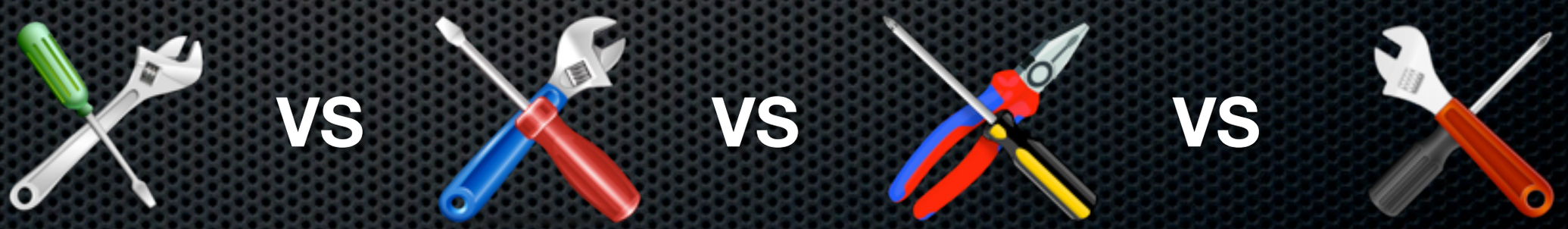
vs

Other tools

Multiple Clone Tools (53 papers)

Agreement / Disagreement (80%)

- Use the other tools to evaluate the given clone detection tool (58%)
- Compare how results are different from detection tools (19%)
- Select the best tool for the analysis task (3%)



Fitness Function

- Individual Task

$$f_I(TS(X), S_k) = \frac{\sum_{i=1}^{n} (i \times \text{AgreedLOC}[i])}{n \times \sum_{i=1}^{n} \text{AgreedLOC}[i]}$$

Fitness Function

- Individual Task

$$f_I(TS(X), S_k) = \frac{\sum_{i=1}^{n}(i \times \mathrm{AgreedLOC}[i])}{n \times \sum_{i=1}^{n} \mathrm{AgreedLOC}[i]}$$

Fitness Function

- Individual Task

$$f_I(TS(X), S_k) = \frac{\sum_{i=1}^{n}(i \times \text{AgreedLOC}[i])}{n \times \sum_{i=1}^{n} \text{AgreedLOC}[i]}$$

e.g. AgreedLOC[3] returns the number of lines of code on which 3 tools agree that they are cloned.

Fitness Function

- Individual Task

$$f_I(TS(X), S_k) = \frac{\sum_{i=1}^{n} i \times \text{AgreedLOC}[i])}{n \times \sum_{i=1}^{n} \text{AgreedLOC}[i]}$$

e.g. AgreedLOC[3] returns the number of lines of code on which 3 tools agree that they are cloned.

Fitness Function

- Individual Task

$$f_I(TS(X), S_k) = \frac{\sum_{i=1}^{n} (i \times \text{AgreedLOC}[i])}{n \times \sum_{i=1}^{n} \text{AgreedLOC}[i]}$$

- General Task

$$f_G(TS(X), SS) = \frac{1}{m} \sum_{S_k \in SS} f_I(TS(X), S_k)$$

Fitness Function

- Individual Task

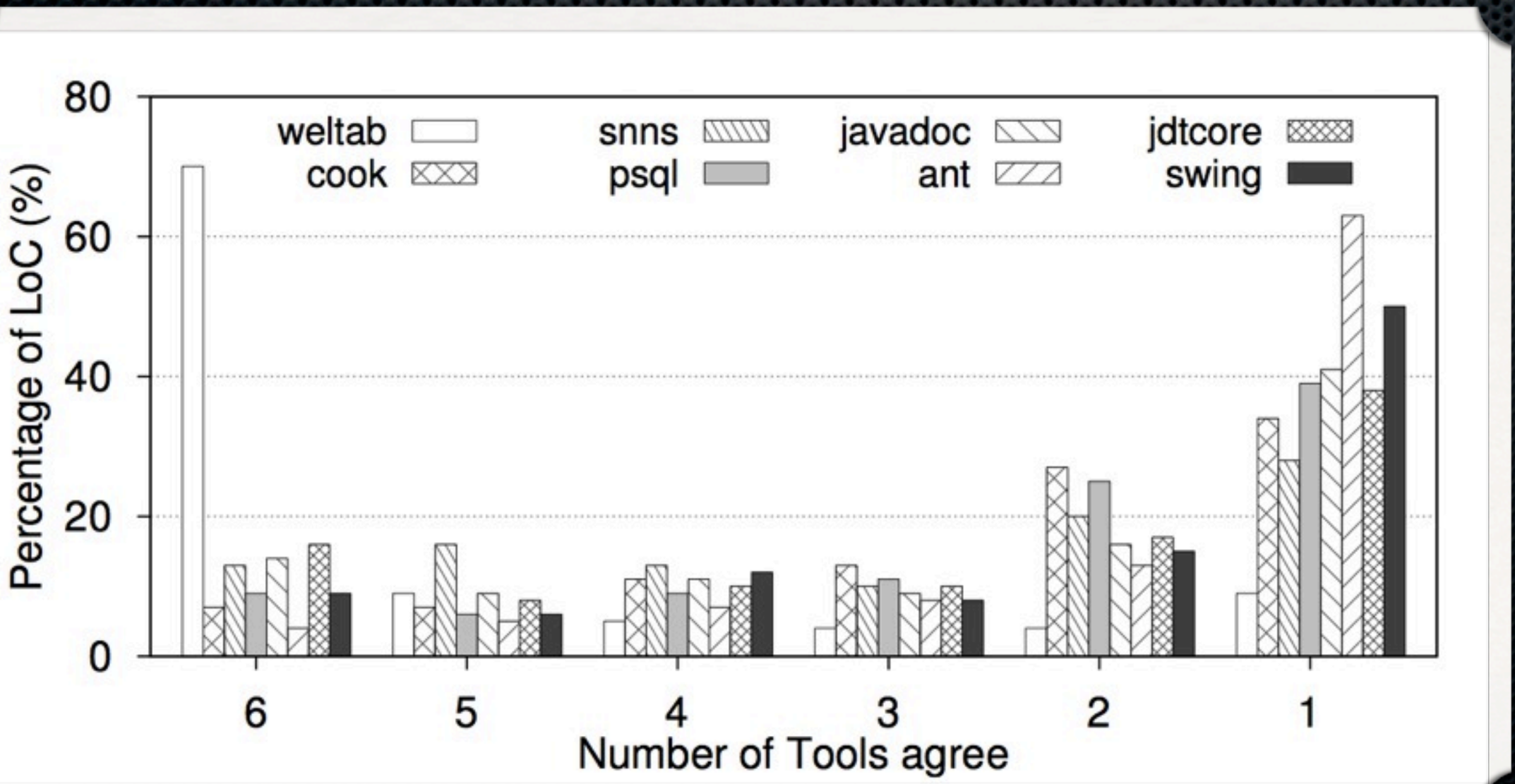
$$f_I(TS(X), S_k) = \frac{\sum_{i=1}^{n}(i \times \text{AgreedLOC}[i])}{n \times \sum_{i=1}^{n} \text{AgreedLOC}[i]}$$

- General Task

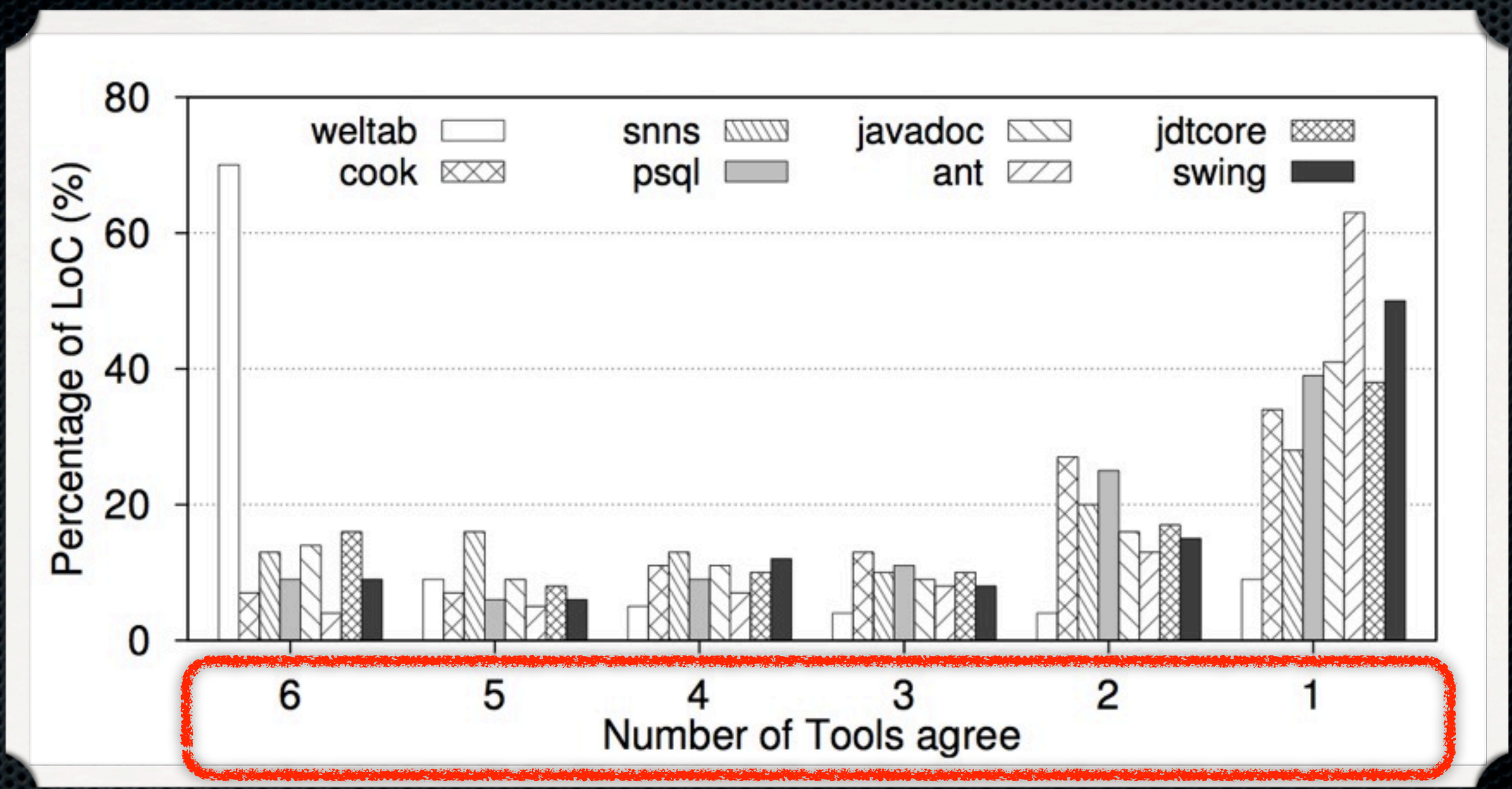
$$f_G(TS(X), SS) = \frac{1}{m} \sum_{S_k \in SS} f_I(TS(X), S_k)$$

RQ1: How much agreement can be obtained using the default configuration of clone detection tools?

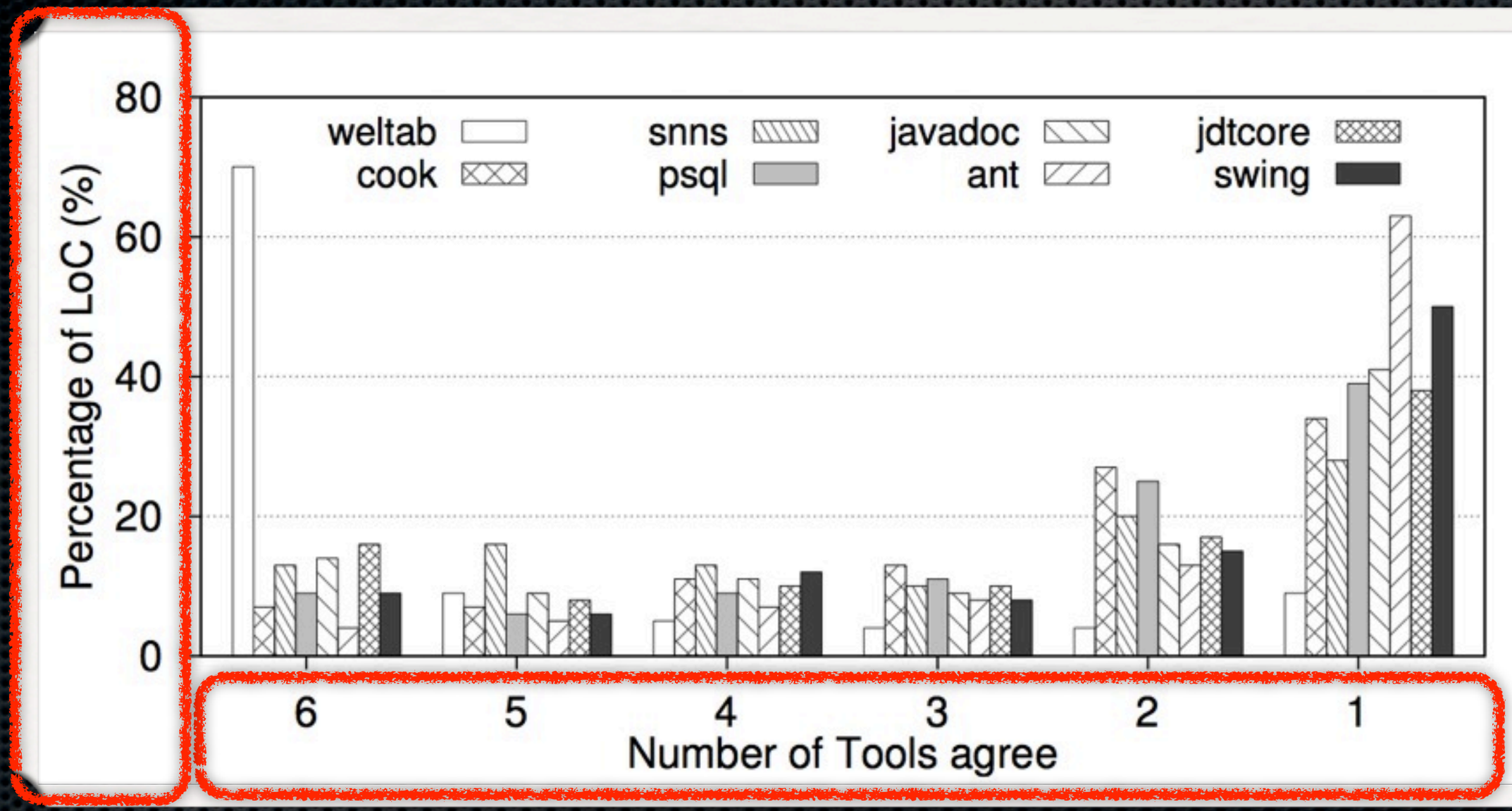
RQ1: Default Agreement Baseline



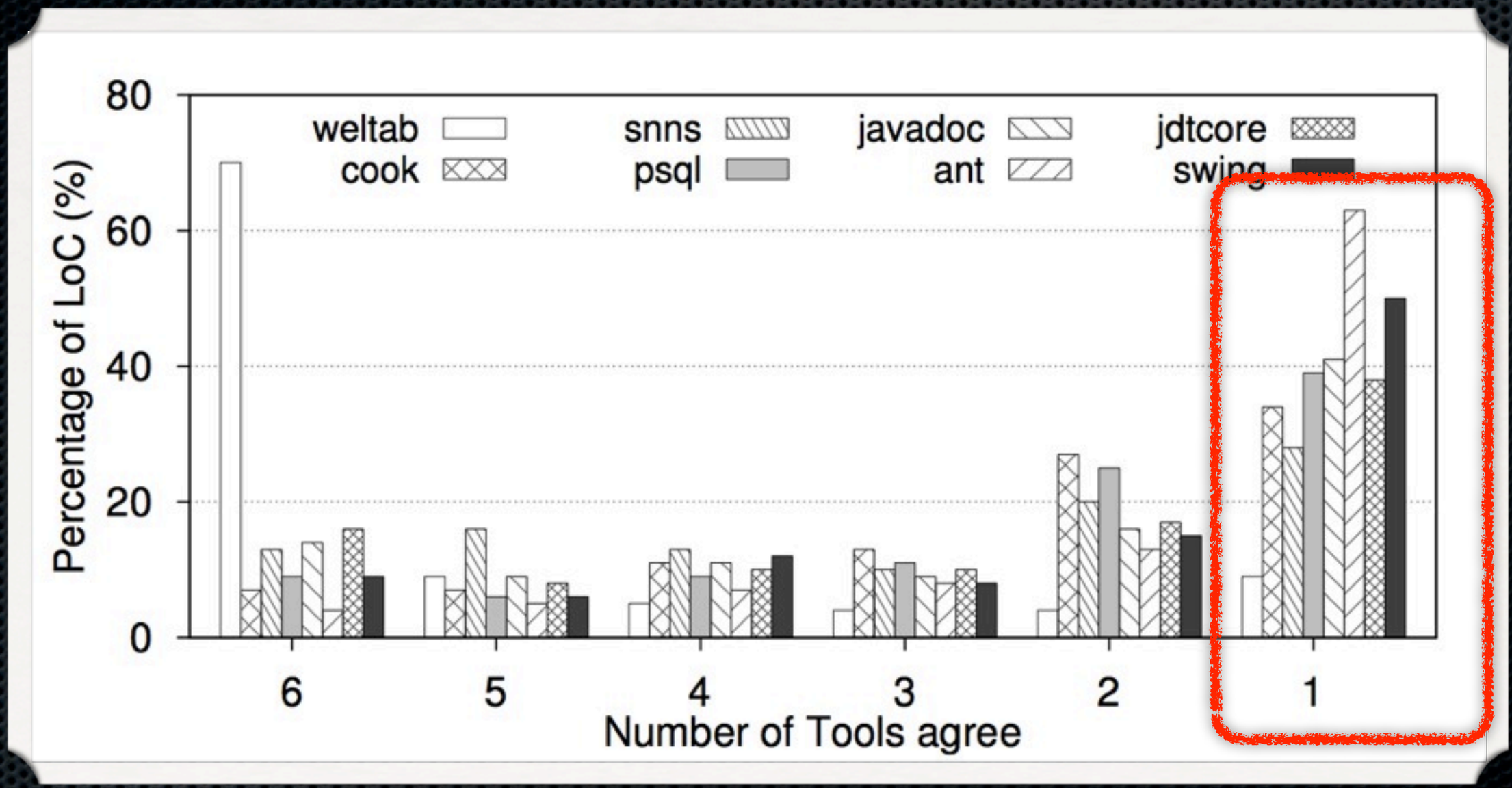
RQ1: Default Agreement Baseline



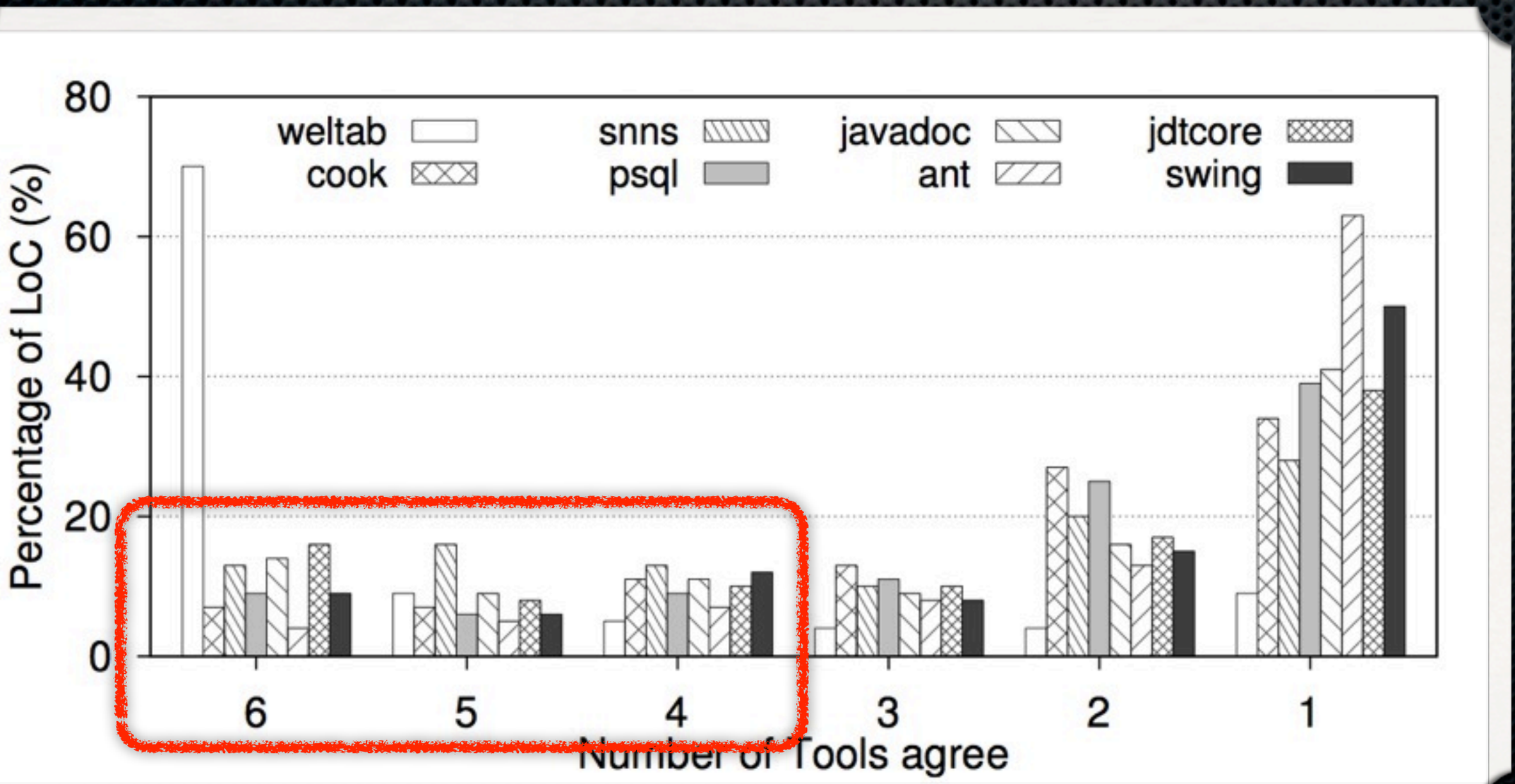
RQ1: Default Agreement Baseline



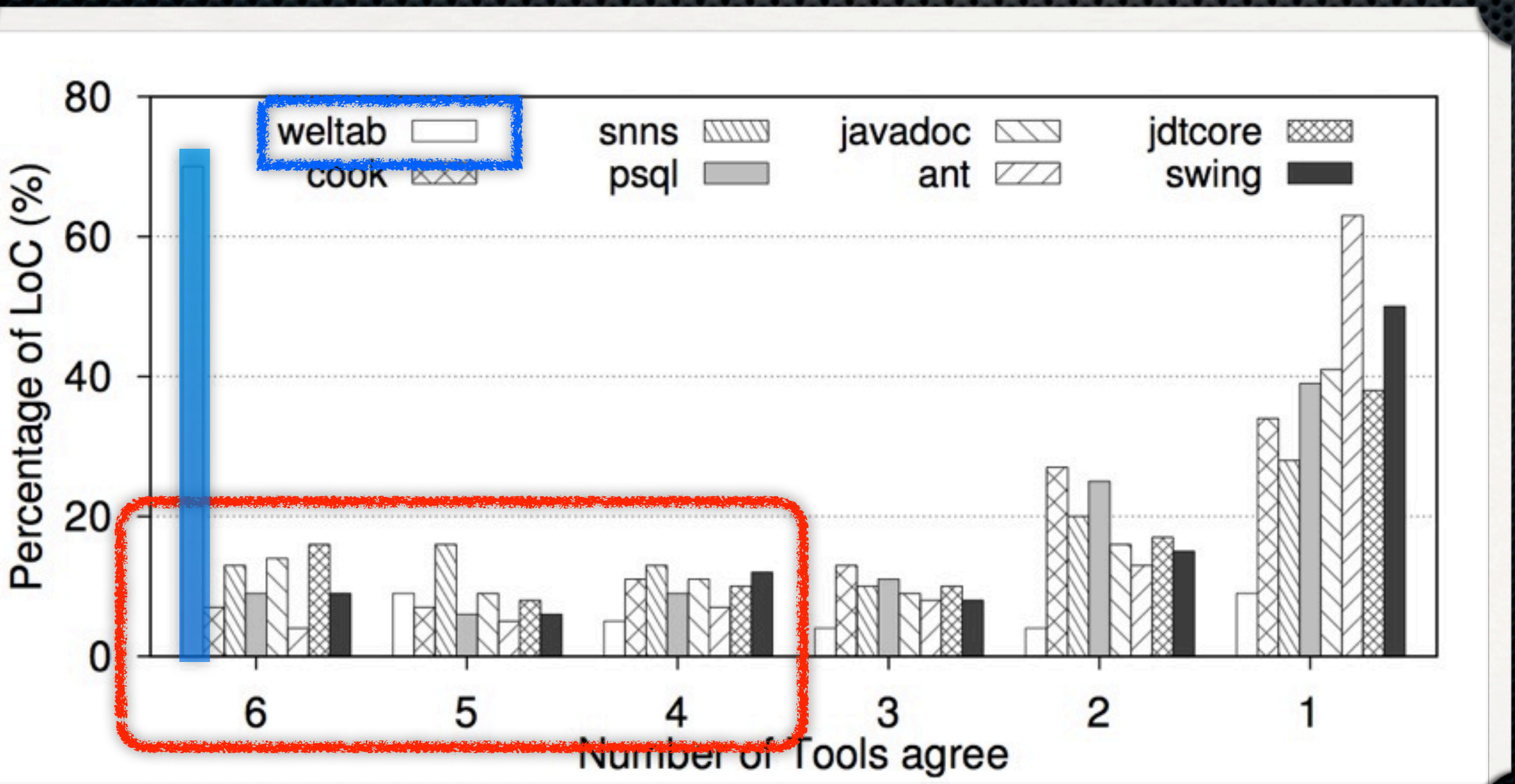
RQ1: Default Agreement Baseline



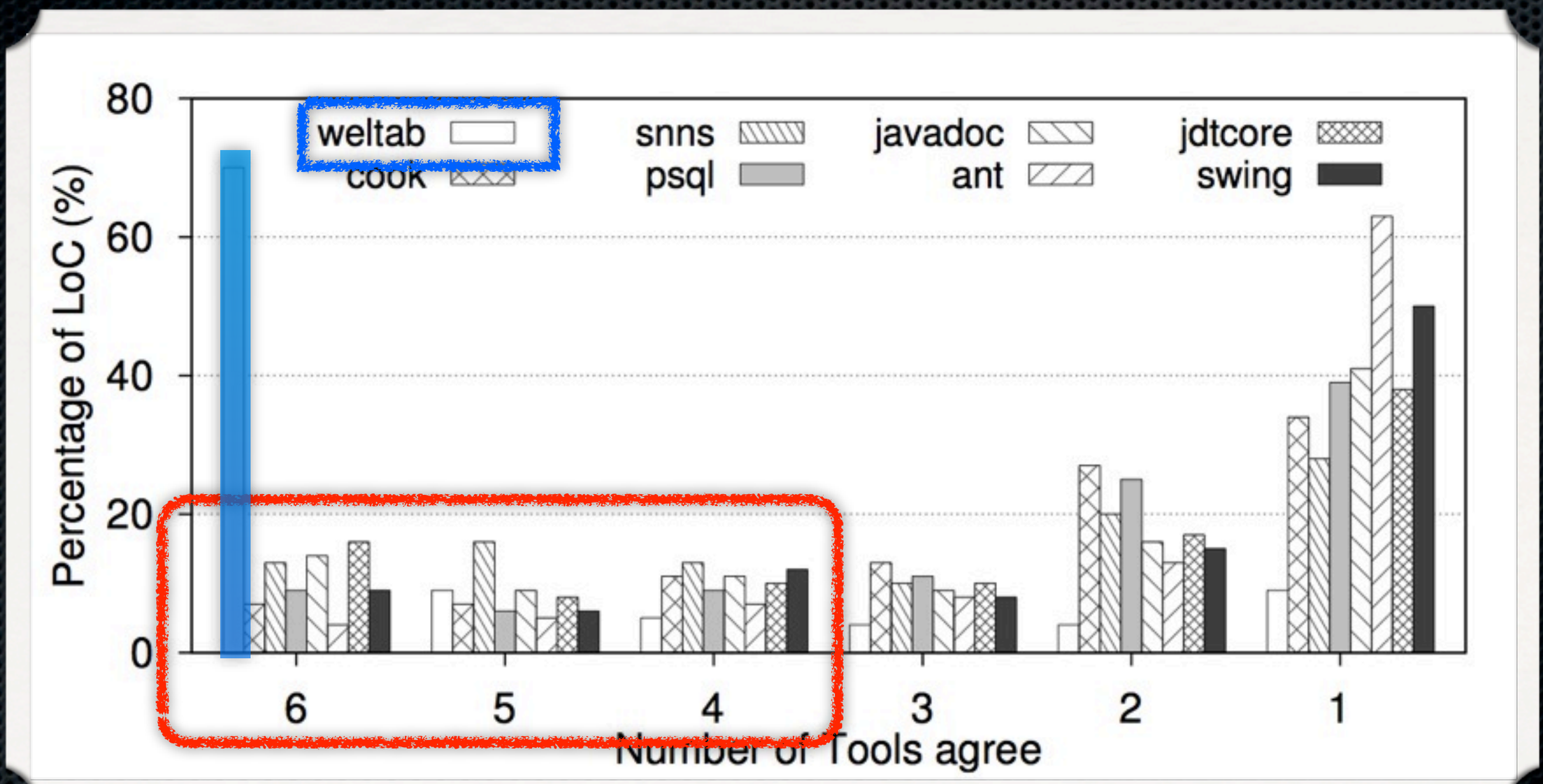
RQ1: Default Agreement Baseline



RQ1: Default Agreement Baseline



RQ1: Default Agreement Baseline



In their default configurations, clone detection tools have a low agreement on which lines are cloned.

RQ1: Default Agreement Baseline

- However, perhaps the low agreement observed is due to the fundamentally different clone detection techniques the tools implement?

RQ1: Default Agreement Baseline

- However, perhaps the low agreement observed is due to the fundamentally different clone detection techniques the tools implement?

100 valid random settings

$$\text{fitness ratio} \equiv \frac{f(\text{random}) - f(\text{default})}{f(\text{default})}$$

RQ1: Default Agreement Baseline

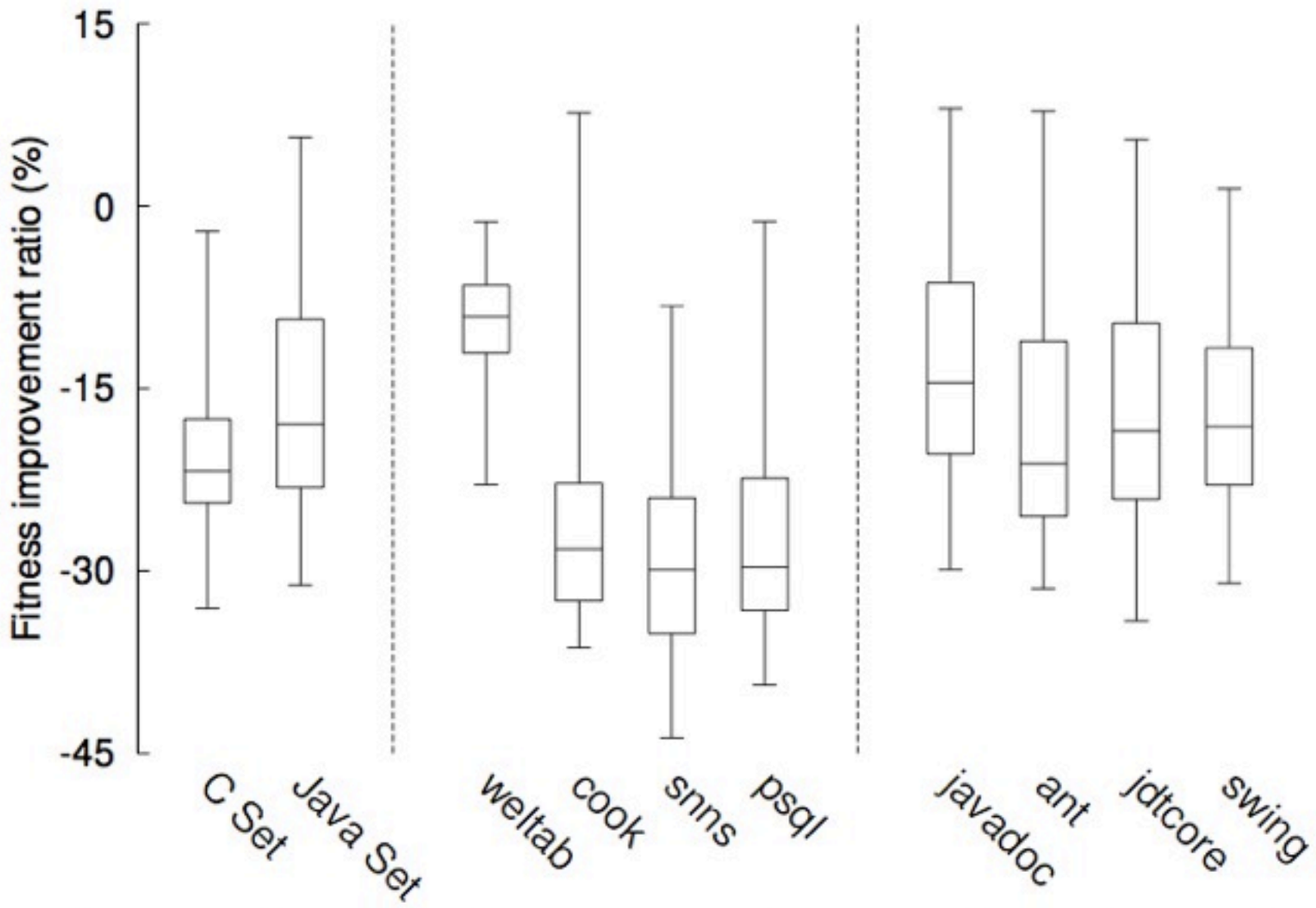


Figure 5: Default Configurations Perform Poorly

RQ1: Default Agreement Baseline

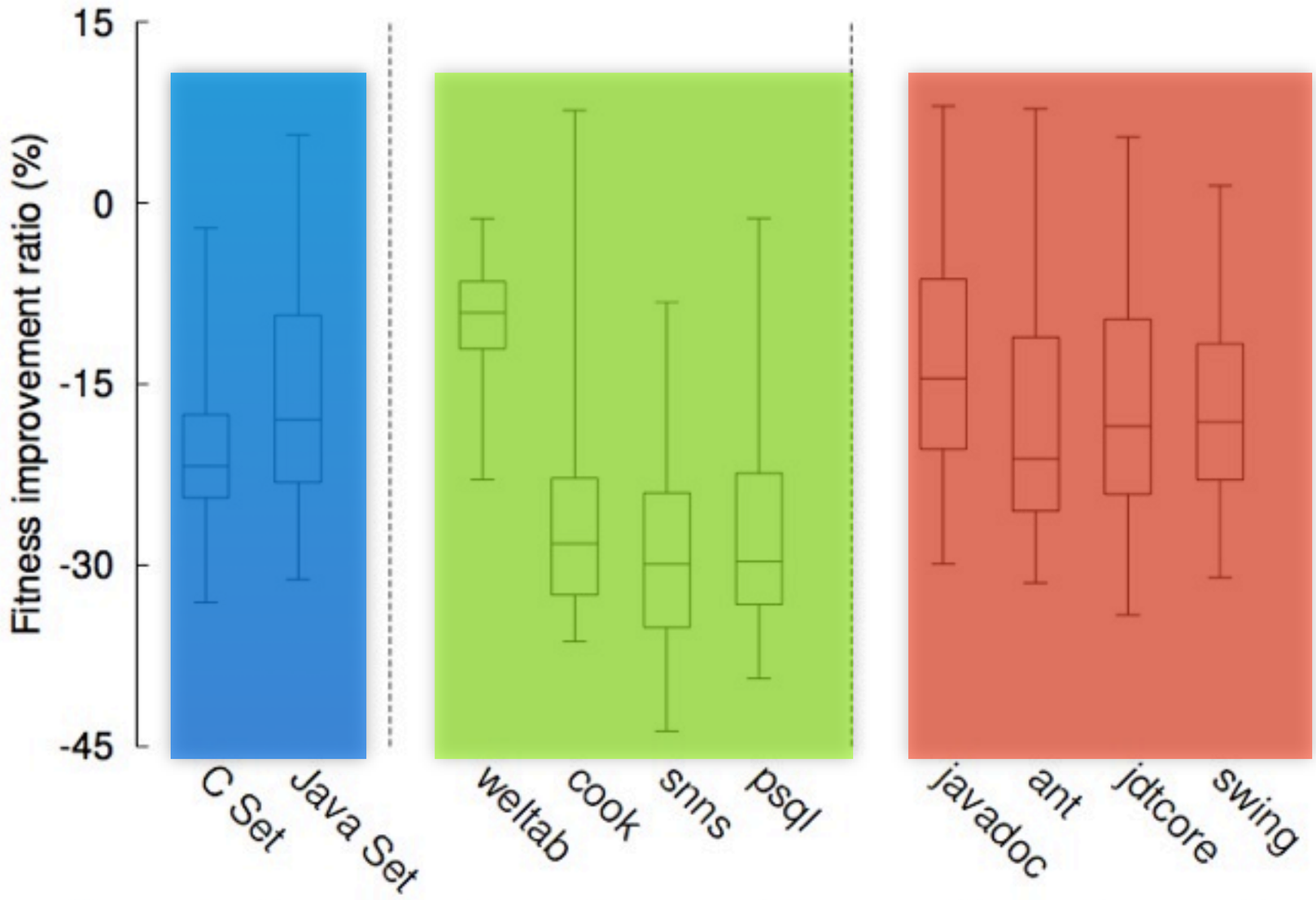


Figure 5: Default Configurations Perform Poorly

RQ1: Default Agreement Baseline

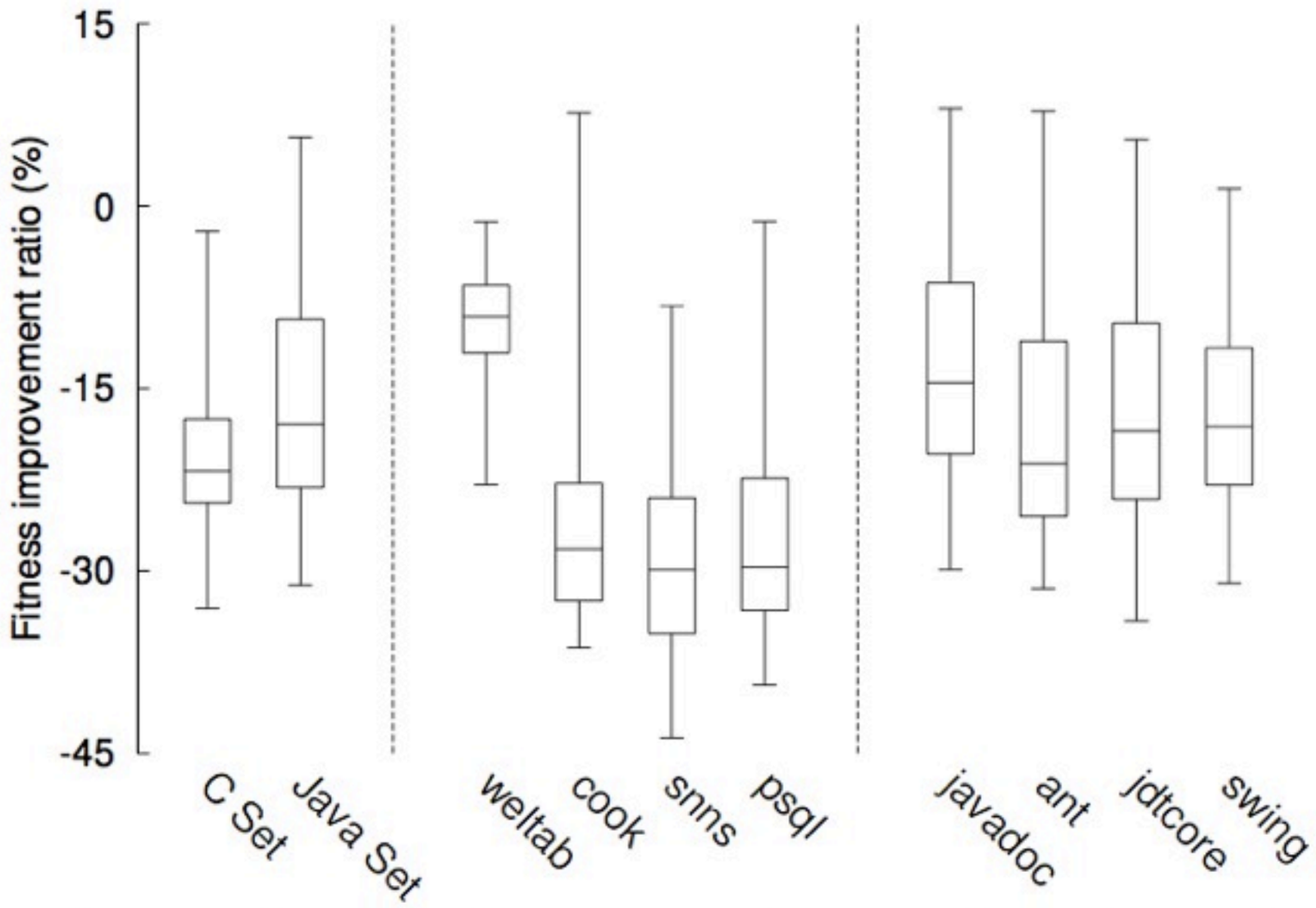


Figure 5: Default Configurations Perform Poorly

RQ1: Default Agreement Baseline

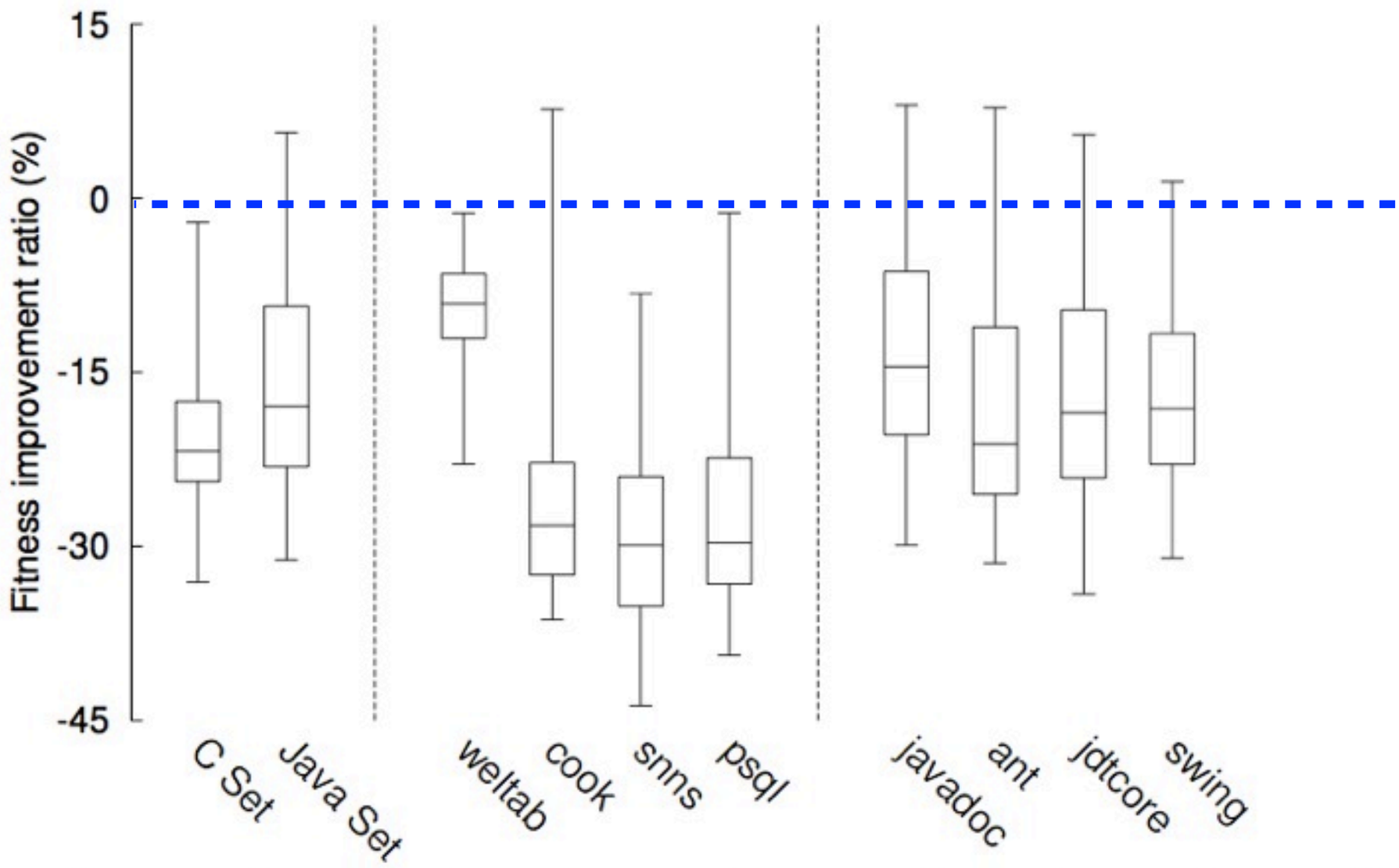


Figure 5: Default Configurations Perform Poorly

RQ1: Default Agreement Baseline

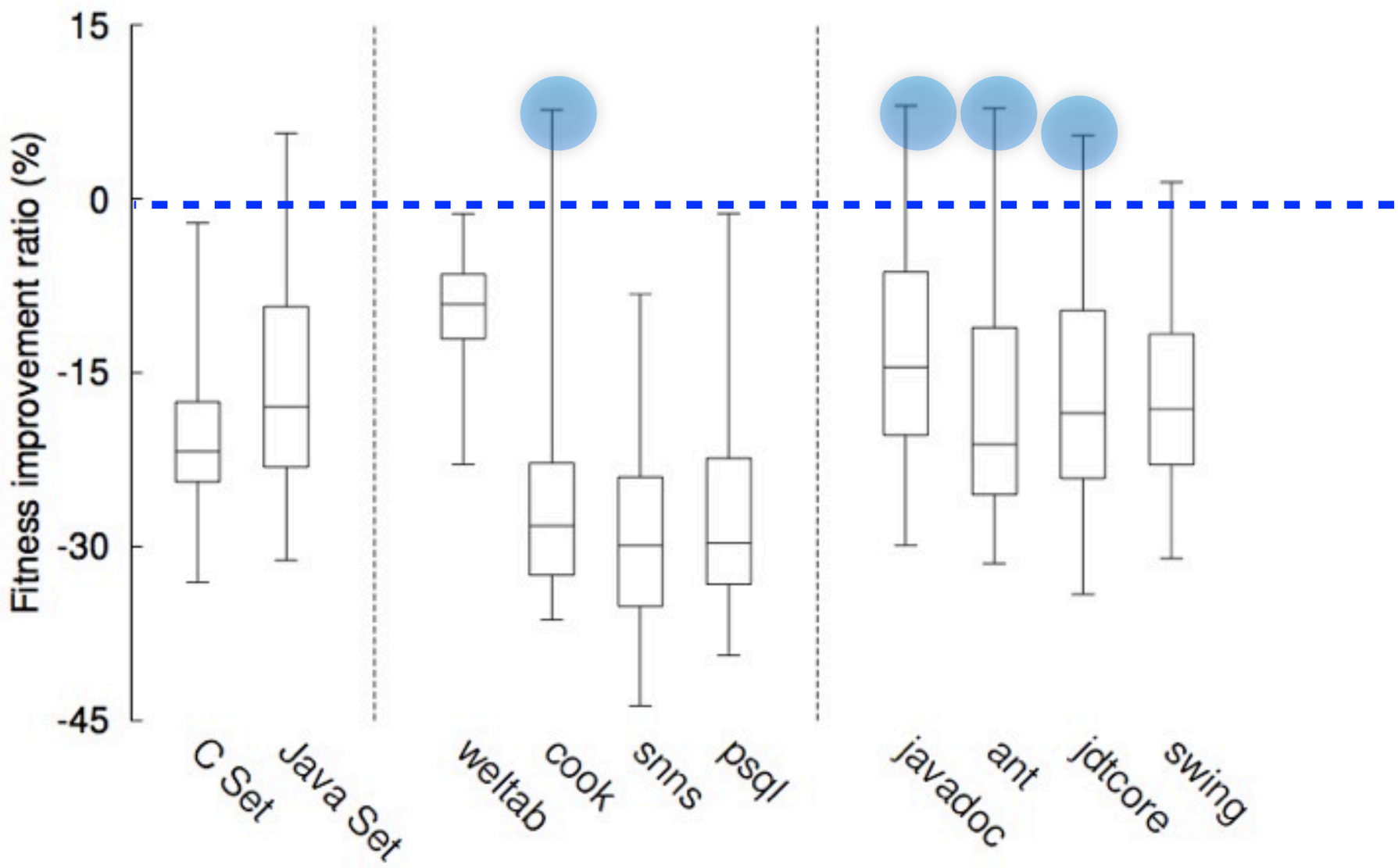


Figure 5: Default Configurations Perform Poorly

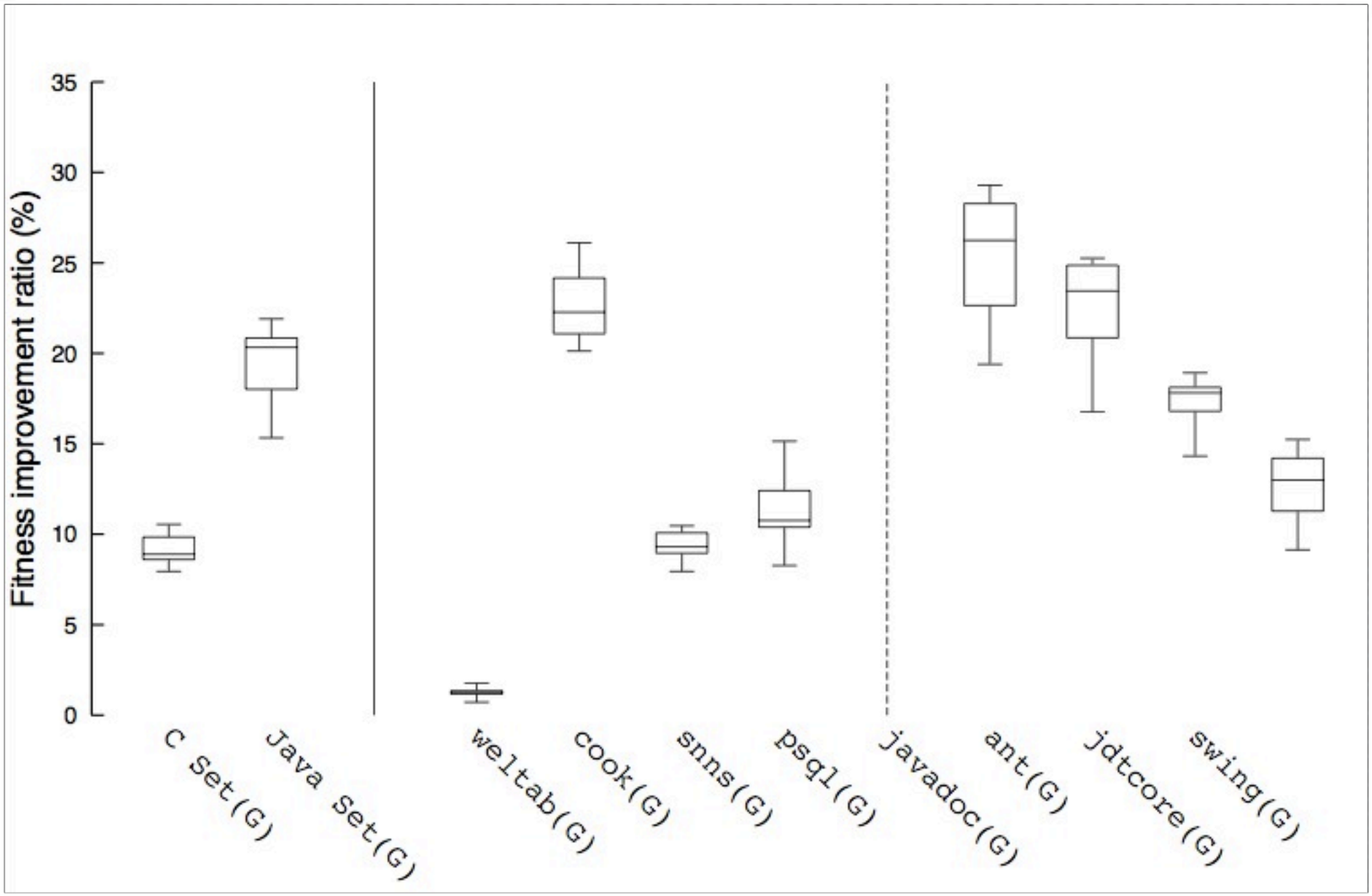
RQ2: How much agreement can our approach find among all tools using the general fitness function, which seeks to find agreement on all subject systems?

RQ2: How much agreement can our approach find among all tools using the general fitness function, which seeks to find agreement on all subject systems?

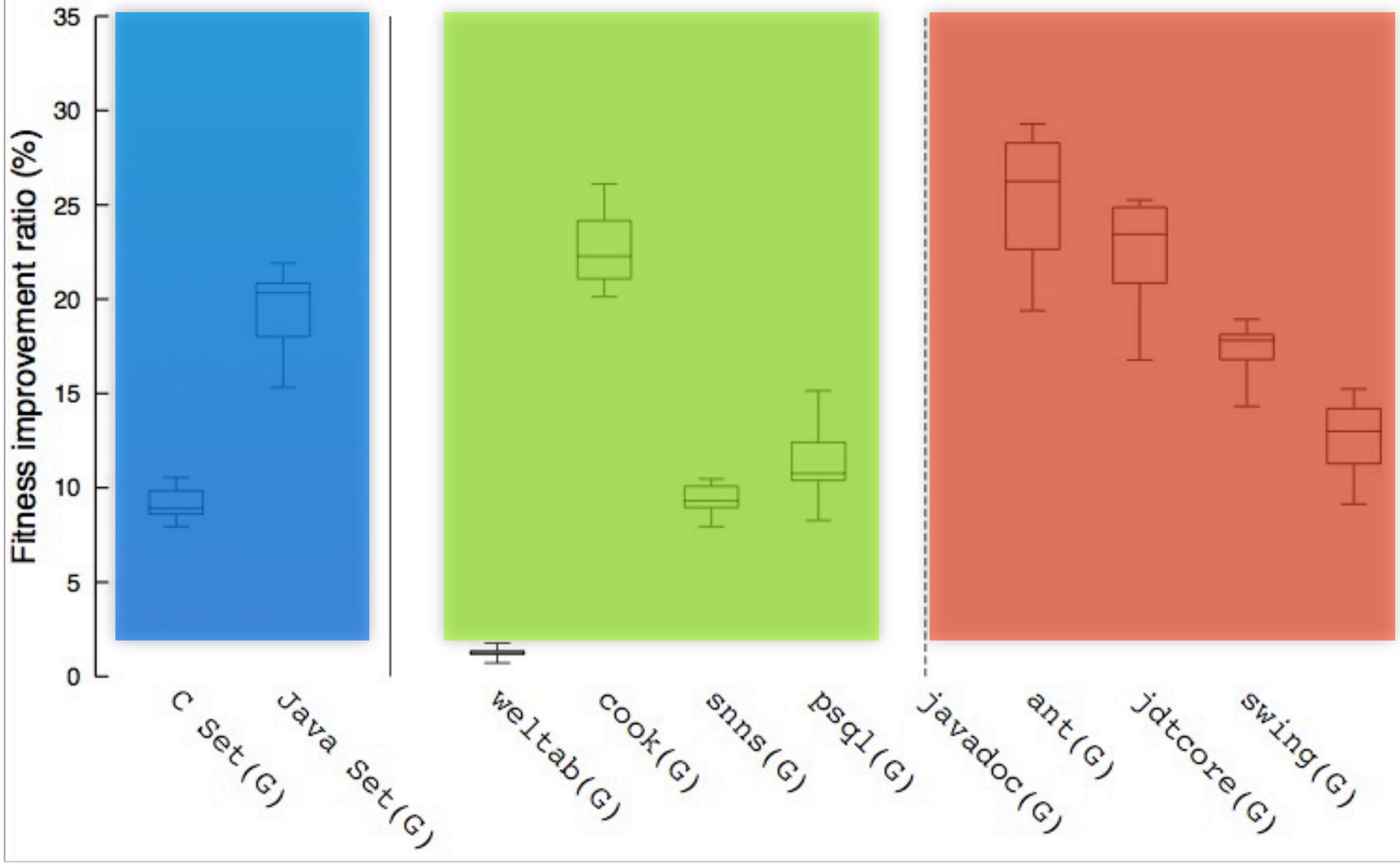
20 Runs (Search for better general settings)

$$\text{fitness ratio} \equiv \frac{f(\text{general}) - f(\text{default})}{f(\text{default})}$$

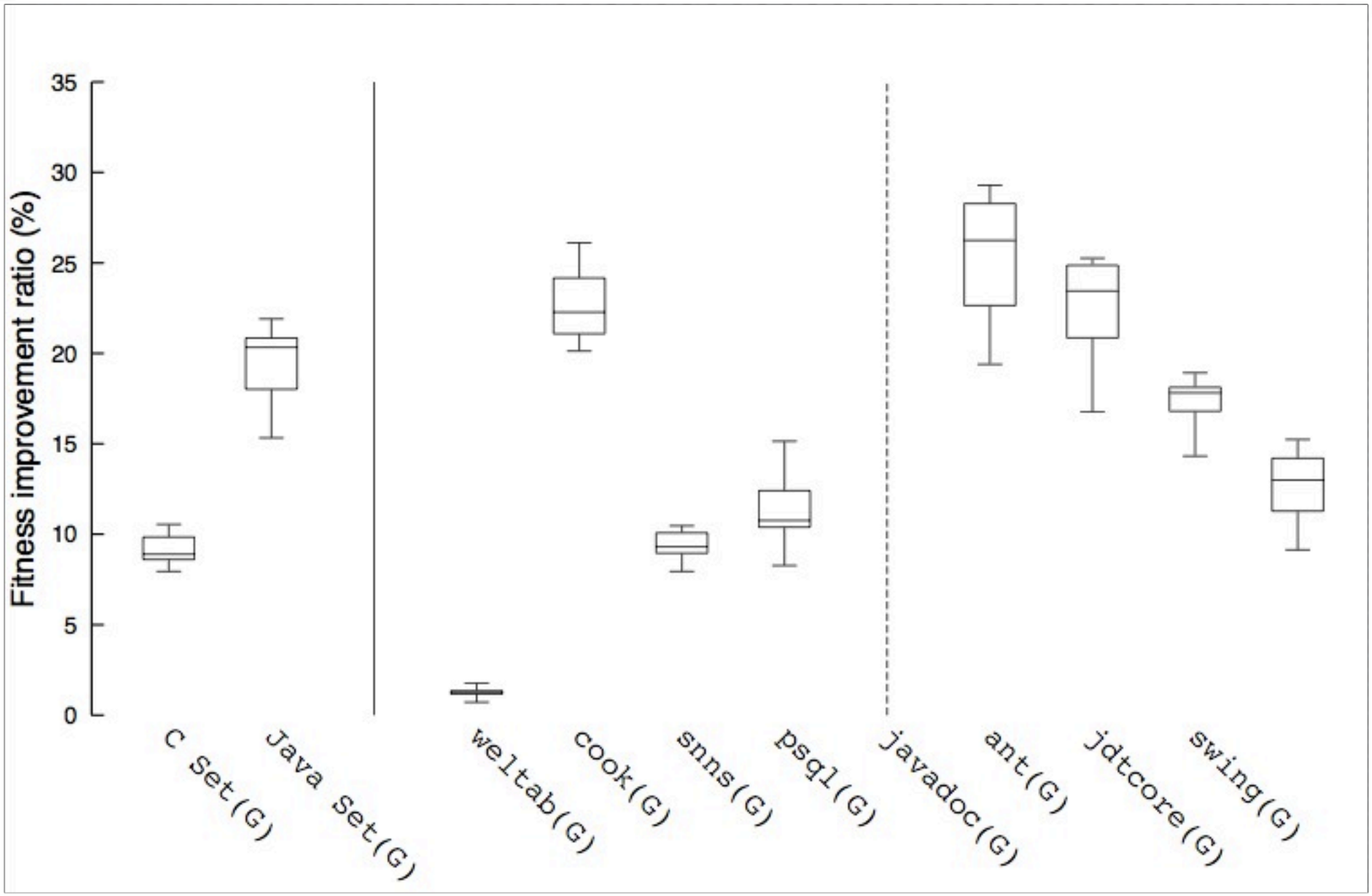
RQ2: Optimised General Agreement



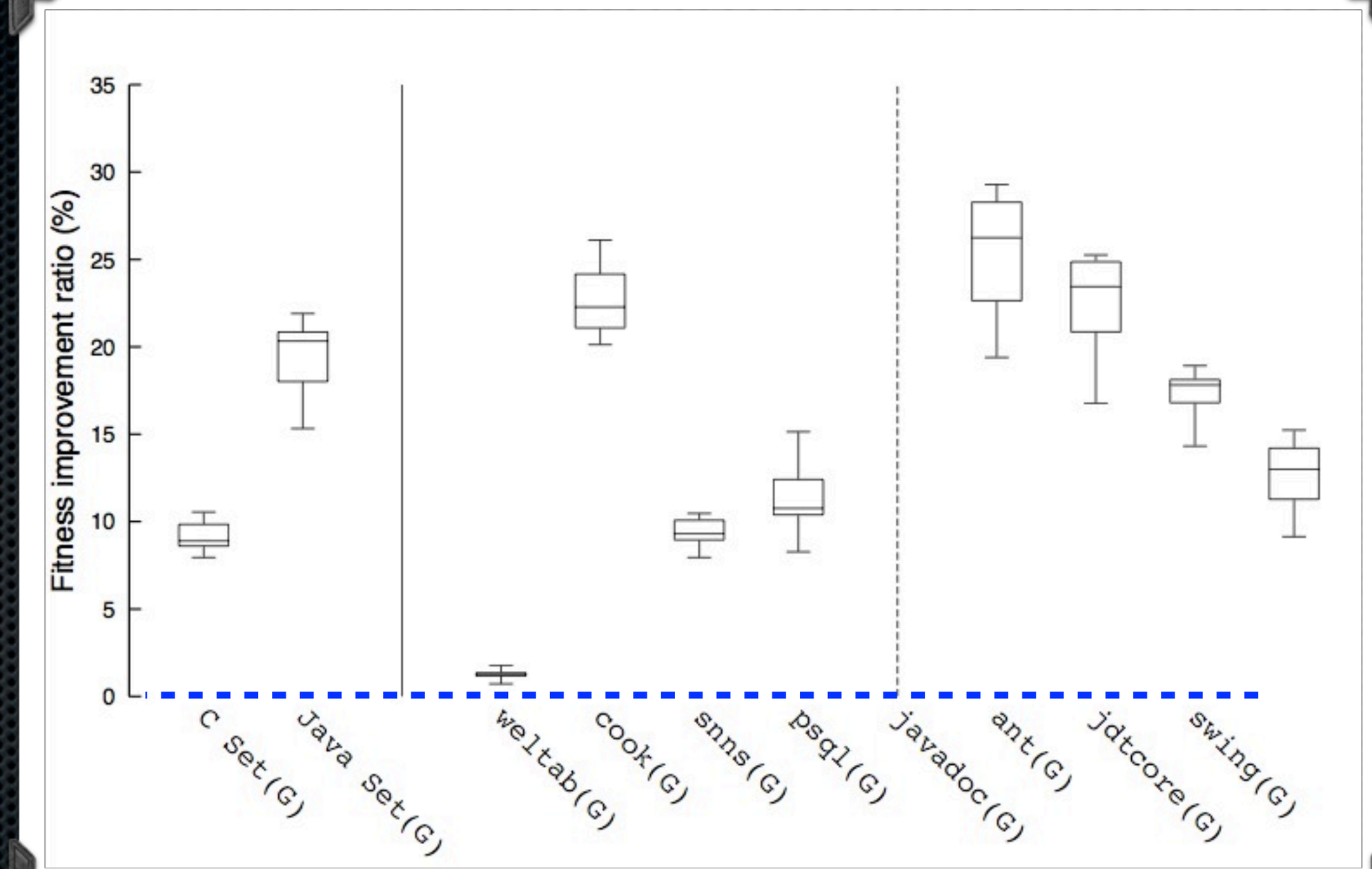
RQ2: Optimised General Agreement



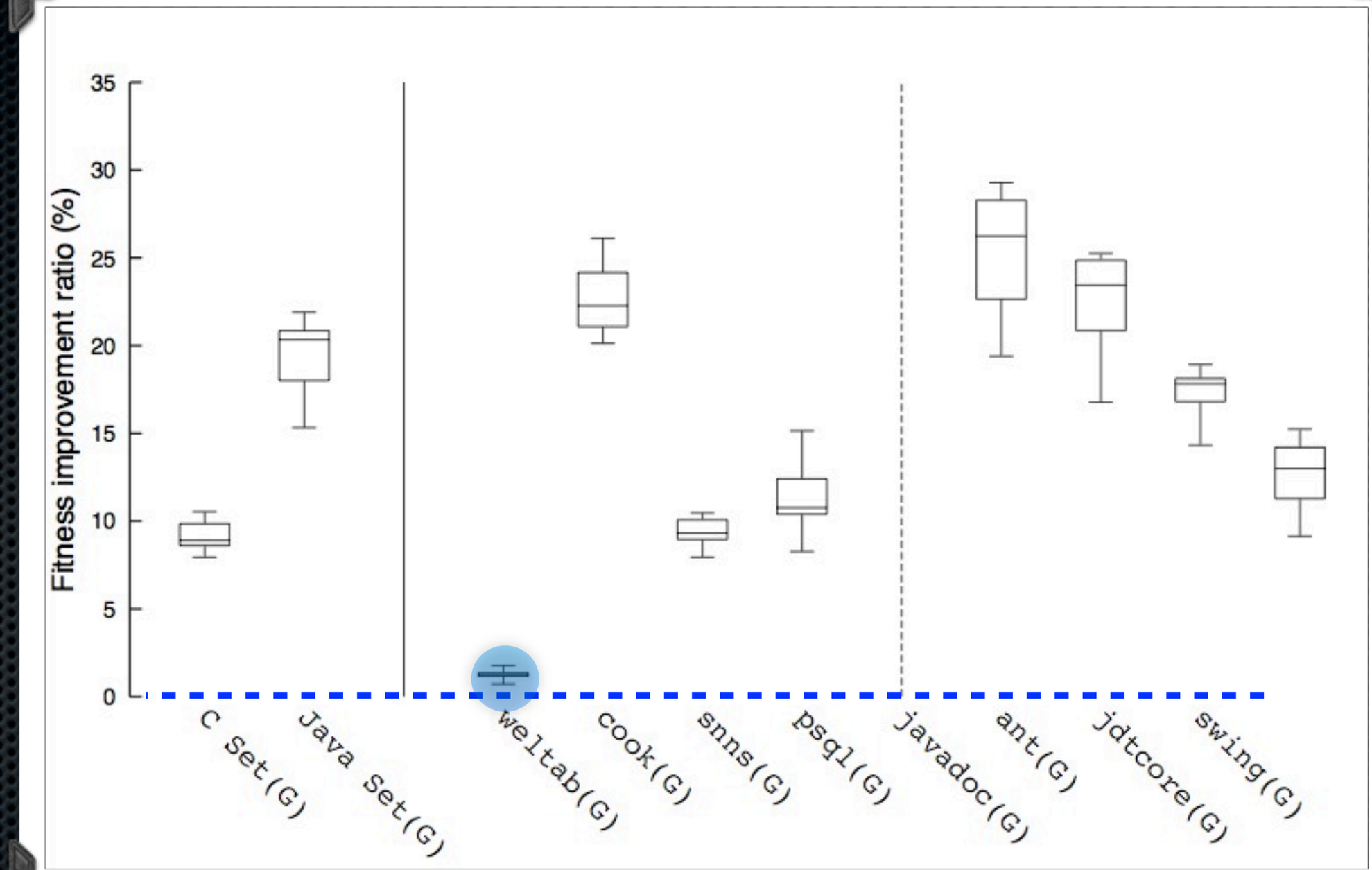
RQ2: Optimised General Agreement



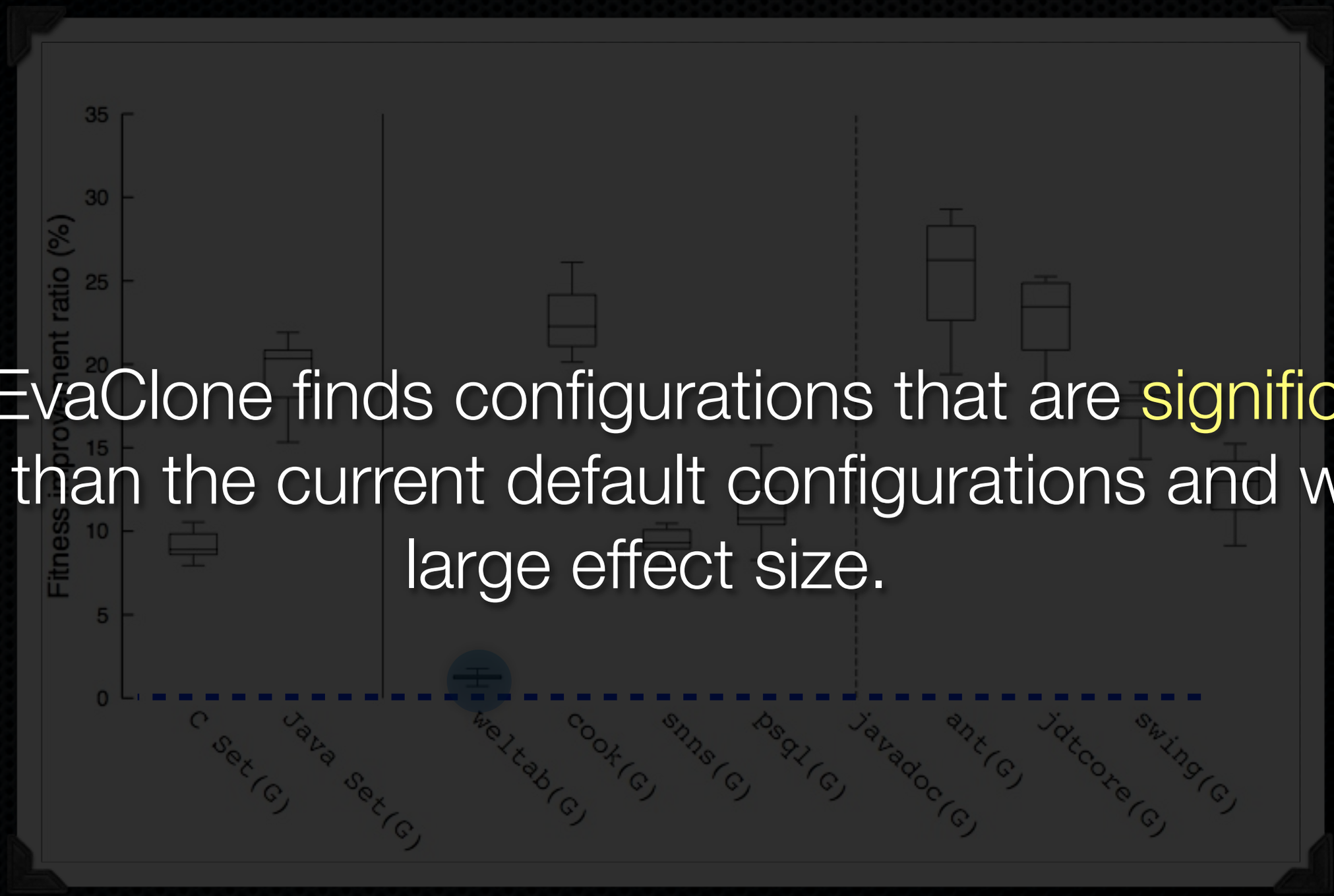
RQ2: Optimised General Agreement



RQ2: Optimised General Agreement



RQ2: Optimised General Agreement



CloudEvaClone finds configurations that are significantly better than the current default configurations and with a large effect size.

RQ3: How much agreement can our approach find among all tools using the individual fitness function, which seeks to find agreement on each individual subject system in isolation?

RQ3: Optimised Individual Agreement

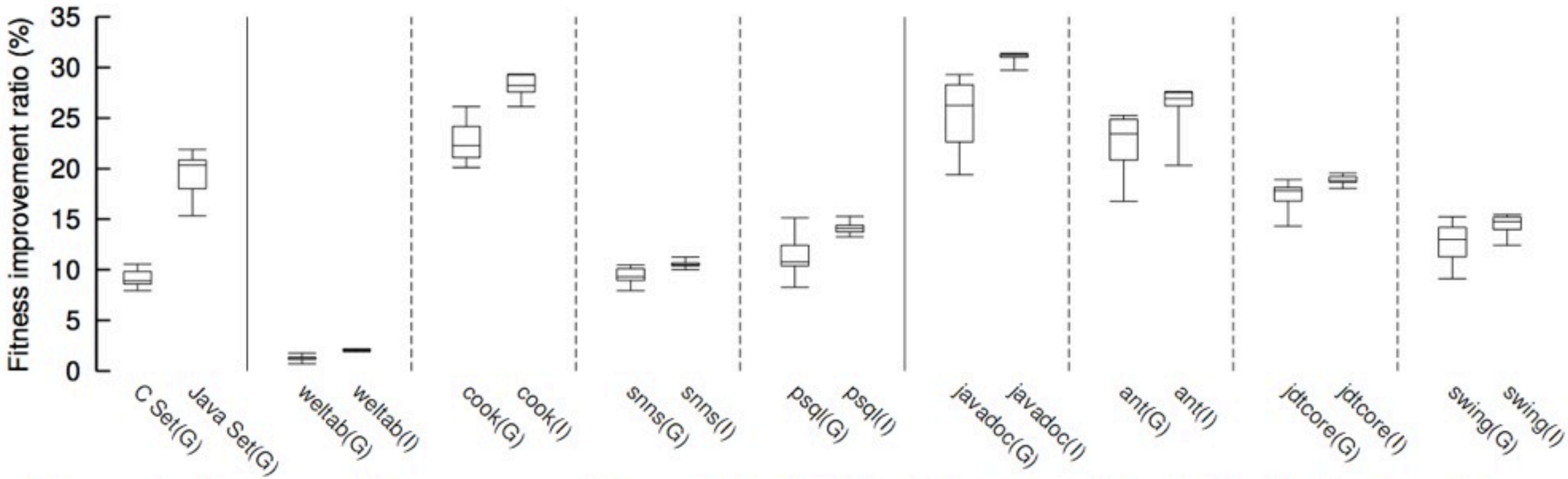


Figure 6: Agreement Improvement (over Defaults) for General and Individual Configurations

RQ3: Optimised Individual Agreement

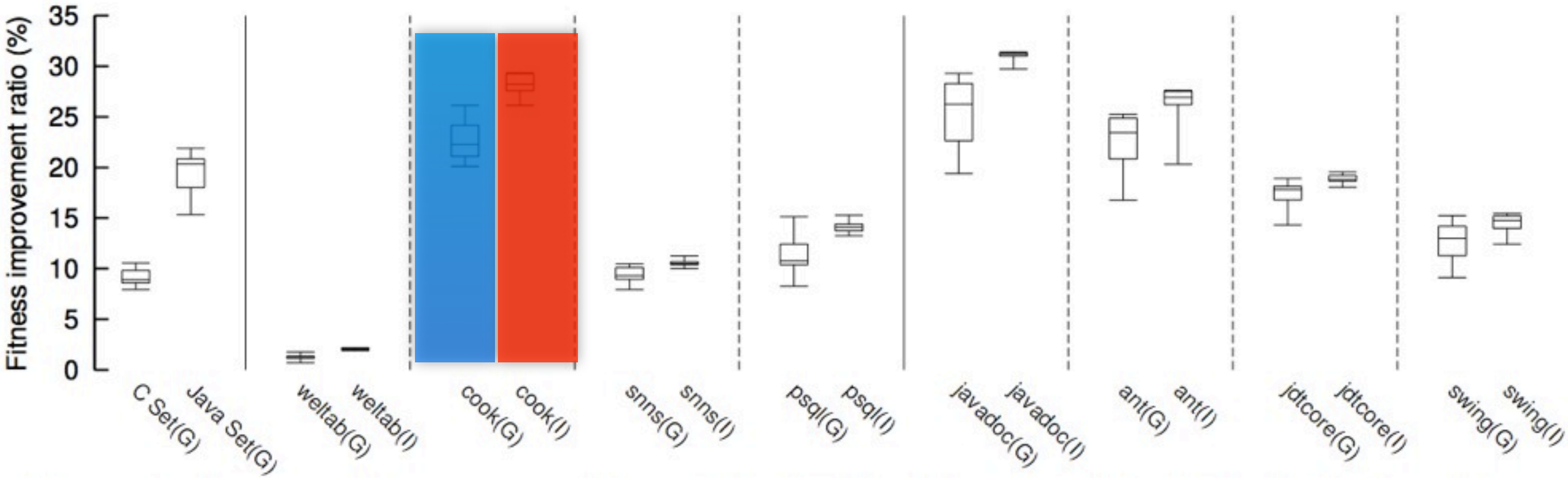


Figure 6: Agreement Improvement (over Defaults) for General and Individual Configurations

RQ3: Optimised Individual Agreement

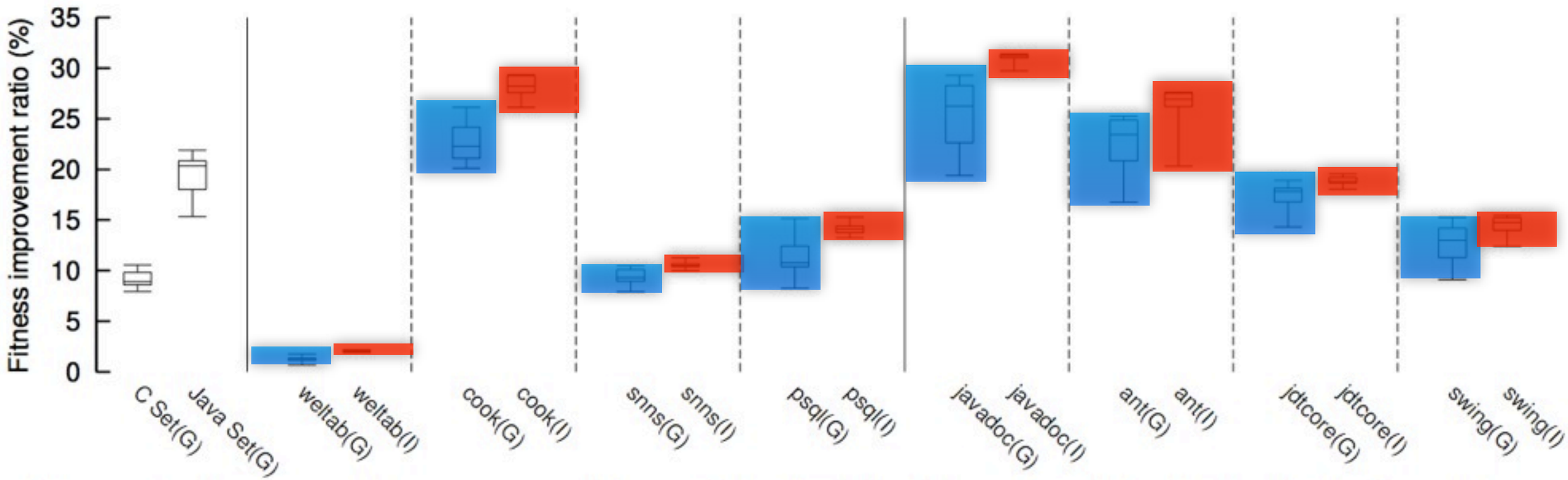


Figure 6: Agreement Improvement (over Defaults) for General and Individual Configurations

RQ3: Optimised Individual Agreement

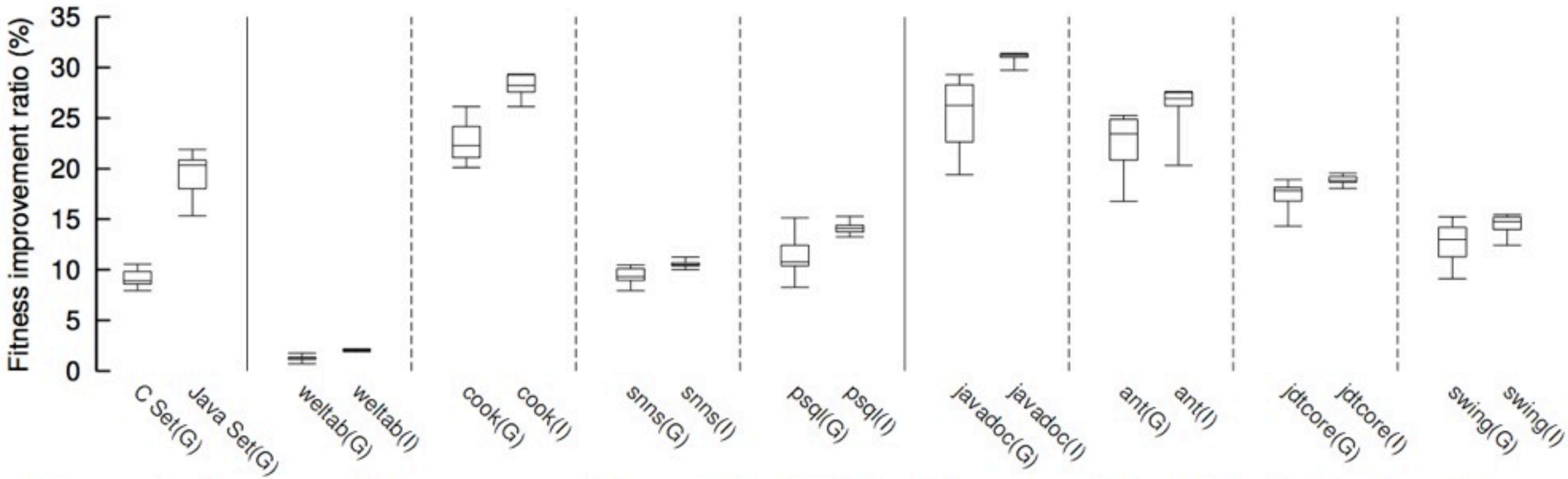


Figure 6: Agreement Improvement (over Defaults) for General and Individual Configurations

CloudEvaClone can find even greater agreement using the individual fitness function applied to each subject system in isolation.

RQ4: How much will recall and precision change when the optimised configurations are used?

RQ4: How much will recall and precision change when the optimised configurations are used?

Evaluated using Bellon's benchmark (OK match)

- If CloudEvaClone is used to maximise agreement between clone detectors, recall will be favoured over precision and more candidates will be reported.

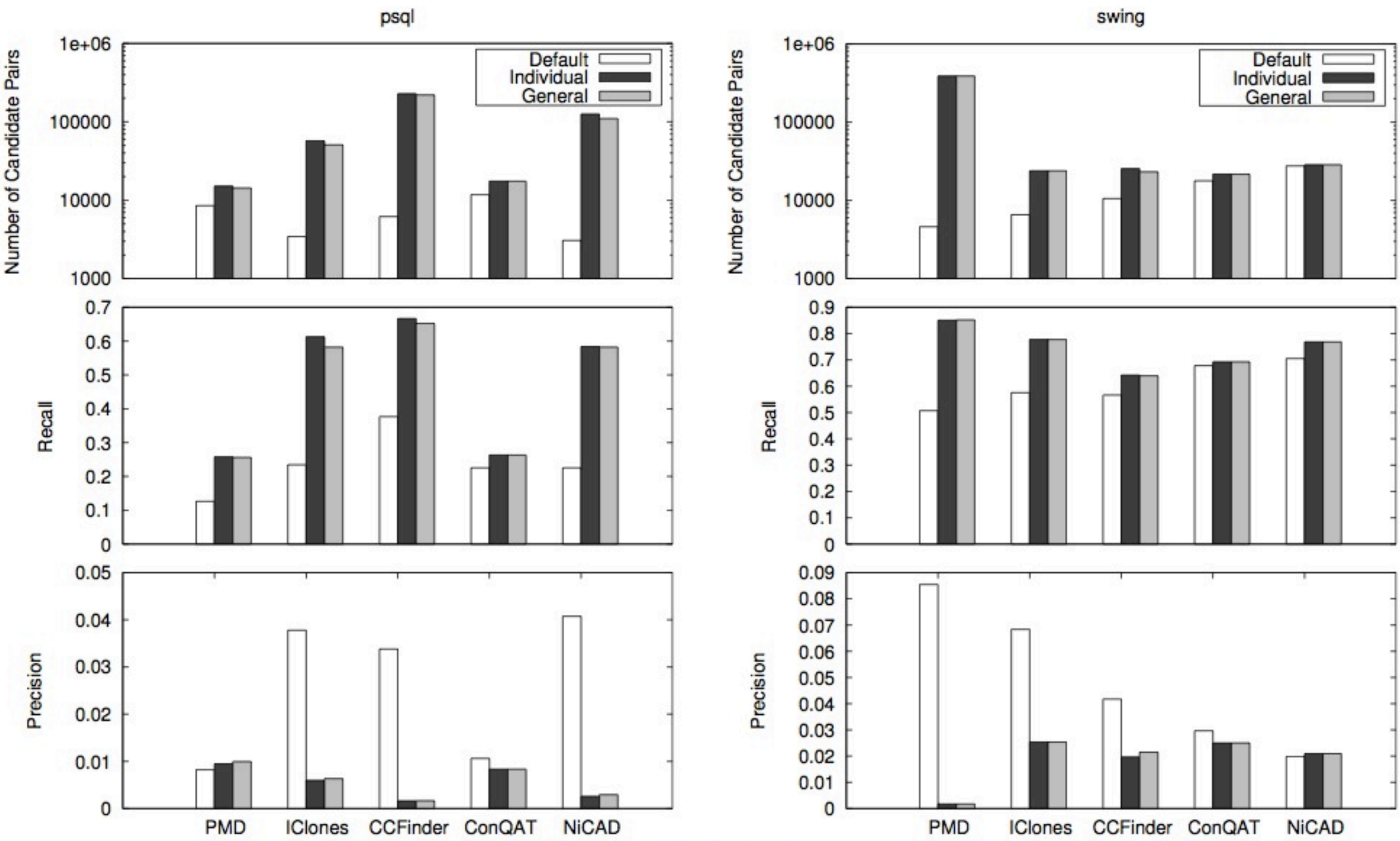
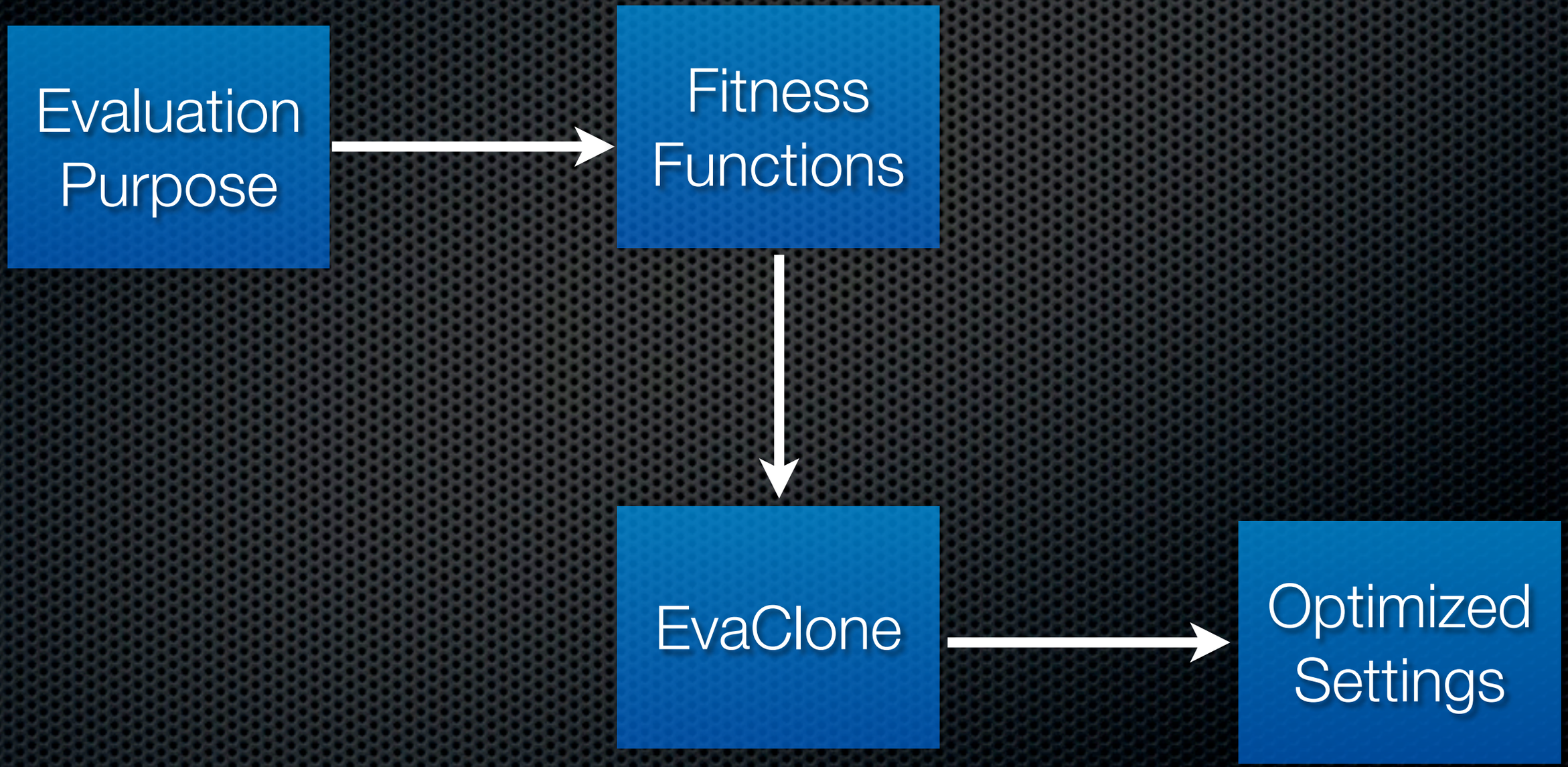


Figure 8: Results from applying Bellon's framework on *psql* and *swing*

A Rigorous Approach to Clone Evaluation



Summary

- The confounding configuration choice problem
  - A detailed review of the literature
  - Empirical evidence
  - EVA Clone Framework (A SBSE approach)

http://www.cs.ucl.ac.uk/staff/Y.Jia/projects/eva_clone/