Untangling Source Code Changes Using Program Slicing

Ward Muylaert (@wardmuylaert) and Coen De Roover (@oniroi)

BENEVOL 2017.

Software Languages Lab

VRIJE UNIVERSITEIT BRUSSEL

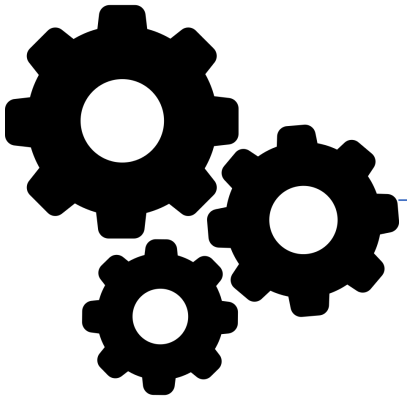
Untangling Source Code Changes Using Program Slicing

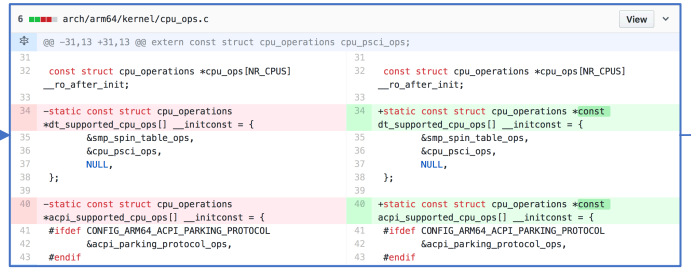
Ward Muylaert (@wardmuylaert) and Coen De Roover (@oniroi)

BENEVOL 2017.

Software Languages Lab

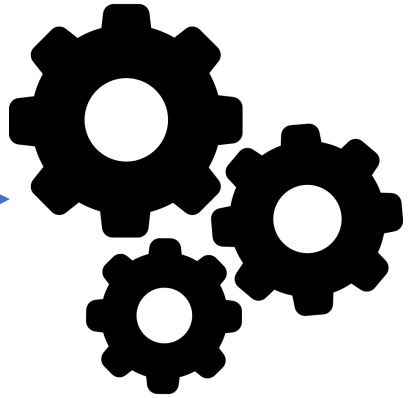
VRIJE UNIVERSITEIT BRUSSEL





6 arch/arm64/kernel/cpu_ops.c View

```
@@ -31,13 +31,13 @@ extern const struct cpu_operations cpu_psci_ops;
 const struct cpu_operations *cpu_ops[NR_CPUS] __ro_after_init;
-static const struct cpu_operations *dt_supported_cpu_ops[] __initconst = {
+static const struct cpu_operations *const dt_supported_cpu_ops[] __initconst = {
         &smp_spin_table_ops,
         &cpu_psci_ops,
         NULL,
 };
-static const struct cpu_operations *acpi_supported_cpu_ops[] __initconst = {
+static const struct cpu_operations *const acpi_supported_cpu_ops[] __initconst = {
 #ifdef CONFIG_ARM64_ACPI_PARKING_PROTOCOL
         &acpi_parking_protocol_ops,
 #endif
```



Version x

Version y

6 arch/arm64/kernel/cpu_ops.c View

```
@@ -31,13 +31,13 @@ extern const struct cpu_operations cpu_psci_ops;

 const struct cpu_operations *cpu_ops[NR_CPUS]
__ro_after_init;

-static const struct cpu_operations
*dt_supported_cpu_ops[] __initconst = {
+static const struct cpu_operations *const
dt_supported_cpu_ops[] __initconst = {
        &smp_spin_table_ops,
        &cpu_psci_ops,
        NULL,
 };

-static const struct cpu_operations
*acpi_supported_cpu_ops[] __initconst = {
+static const struct cpu_operations *const
acpi_supported_cpu_ops[] __initconst = {
 #ifdef CONFIG_ARM64_ACPI_PARKING_PROTOCOL
        &acpi_parking_protocol_ops,
 #endif
```

6 arch/arm64/kernel/cpu_ops.c View

```
@@ -31,13 +31,13 @@ extern const struct cpu_operations cpu_psci_ops;

 const struct cpu_operations *cpu_ops[NR_CPUS]
__ro_after_init;

-static const struct cpu_operations
*dt_supported_cpu_ops[] __initconst = {
+static const struct cpu_operations *const
dt_supported_cpu_ops[] __initconst = {
        &smp_spin_table_ops,
        &cpu_psci_ops,
        NULL,
 };

-static const struct cpu_operations
*acpi_supported_cpu_ops[] __initconst = {
+static const struct cpu_operations *const
acpi_supported_cpu_ops[] __initconst = {
 #ifdef CONFIG_ARM64_ACPI_PARKING_PROTOCOL
        &acpi_parking_protocol_ops,
 #endif
```

Fix issue #31415

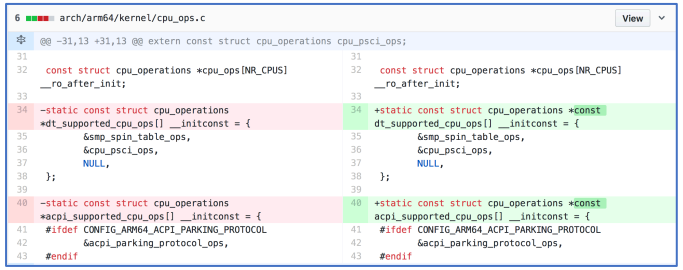
ward committed 7 days ago ✔

Add timetraveling feature

ward committed 7 days ago ✔

Refactor benevol presentation

ward committed 7 days ago ✔



Fix issue #31415

ward committed 7 days ago ✔

Add timetraveling feature

ward committed 7 days ago ✔

Refactor benevol presentation

ward committed 7 days ago ✔

Fix issue #271828 and some refactoring to do so

bad-ward committed 7 days ago ✔

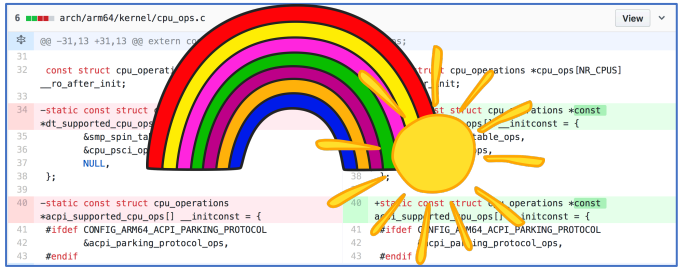
Add timetravel, remove locations

bad-ward committed 7 days ago ✔

Change 50 different things for procrastination

bad-ward committed 7 days ago ✔

Atomic commits



Fix issue #31415

ward committed 7 days ago ✔

Add timetraveling feature

ward committed 7 days ago ✔

Refactor benevol presentation

ward committed 7 days ago ✔

Fix issue #271828 and some refactoring to do so

bad-ward committed 7 days ago ✔

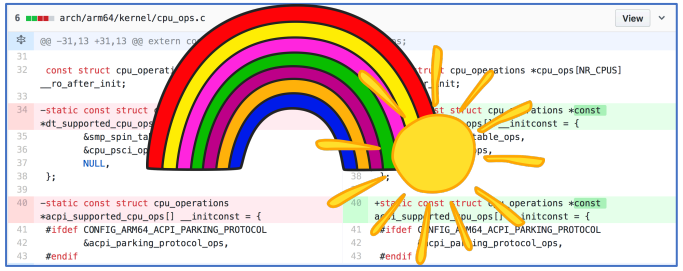
Add timetravel, remove locations

bad-ward committed 7 days ago ✔

Change 50 different things for procrastination

bad-ward committed 7 days ago ✔

Atomic commits



Fix issue #31415

ward committed 7 days ago ✔

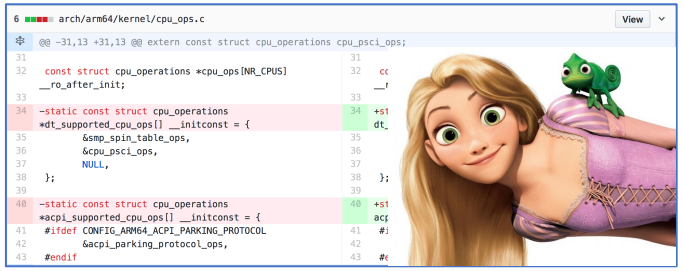
Add timetraveling feature

ward committed 7 days ago ✔

Refactor benevol presentation

ward committed 7 days ago ✔

Tangled commits



Fix issue #271828 and some refactoring to do so

bad-ward committed 7 days ago ✔

Add timetravel, remove locations

bad-ward committed 7 days ago ✔

Change 50 different things for procrastination

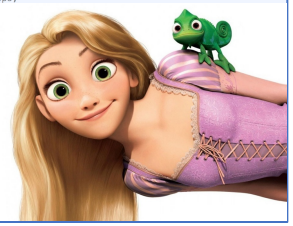
bad-ward committed 7 days ago ✔

Integrate

Revert

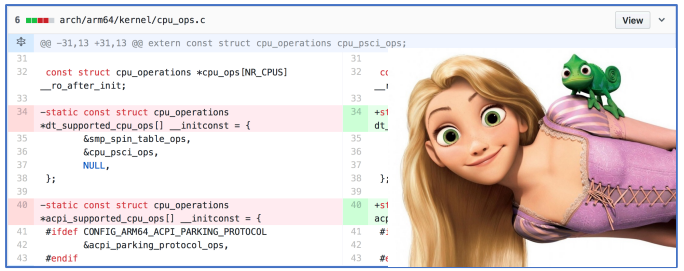
6 arch/arm64/kernel/cpu_ops.c View

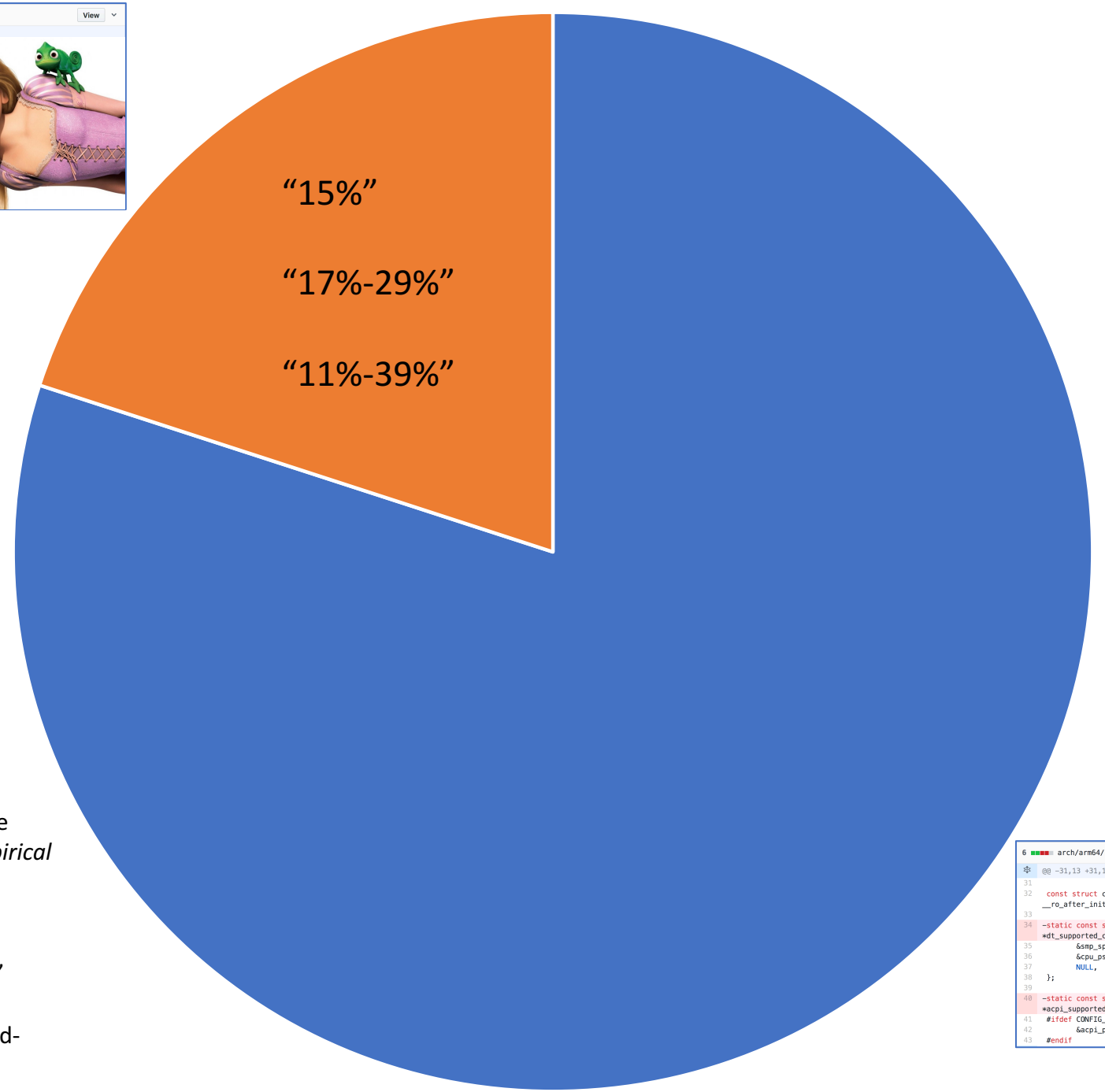
```
@@ -31,13 +31,13 @@ extern const struct cpu_operations cpu_psci_ops;
 const struct cpu_operations *cpu_ops[NR_CPUS]
__ro_after_init;

-static const struct cpu_operations
*dt_supported_cpu_ops[] __initconst = {
        &smp_spin_table_ops,
        &cpu_psci_ops,
        NULL,
};

-static const struct cpu_operations
*acpi_supported_cpu_ops[] __initconst = {
#ifdef CONFIG_ARM64_ACPI_PARKING_PROTOCOL
        &acpi_parking_protocol_ops,
#endif
```

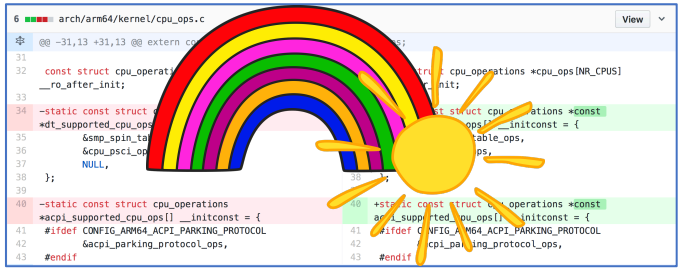


Research

Understand





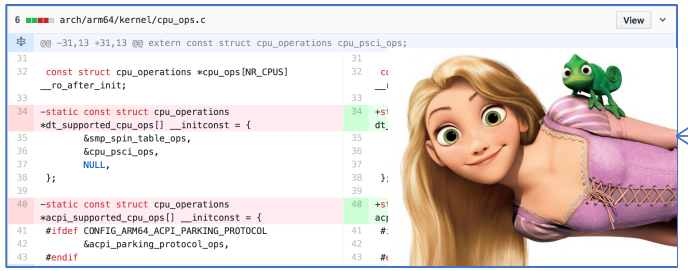


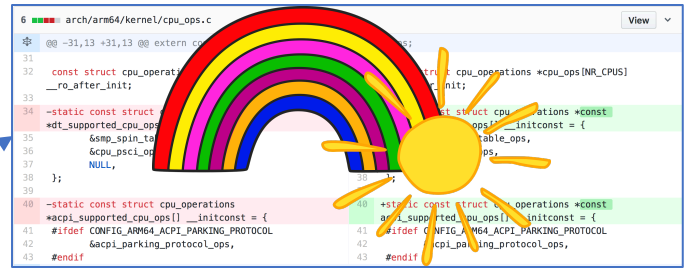
K. Herzig et al., “The impact of tangled code changes on defect prediction models,” *Empirical Software Engineering*, 2015.

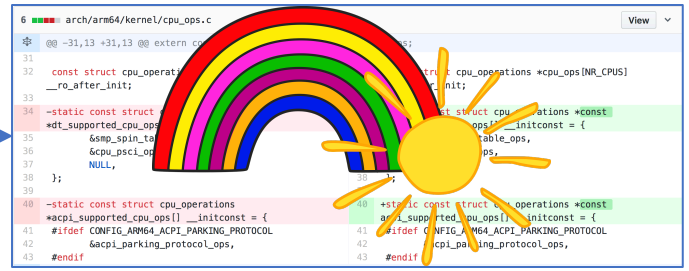
Y. Tao et al., “Partitioning Composite Code Changes to Facilitate Code Review,” in *MSR*, 2015.

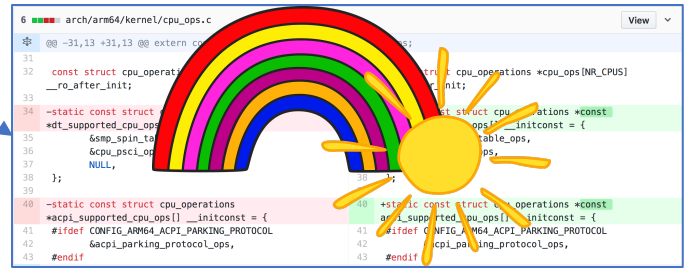
H. A. Nguyen et al., “Filtering Noise in Mixed-Purpose Fixing Commits to Improve Defect Prediction and Localization,” in *ISSRE*, 2013.

Goal


6 arch/arm64/kernel/cpu_ops.c
View
@@ -31,13 +31,13 @@ extern const struct cpu_operations cpu_psci_ops;
const struct cpu_operations *cpu_ops[NR_CPUS] __ro_after_init;
-static const struct cpu_operations *dt_supported_cpu_ops[] __initconst = {
+static const struct cpu_operations *const dt_supported_cpu_ops[] __initconst = {
&smp_spin_table_ops,
&cpu_psci_ops,
NULL,
};
-static const struct cpu_operations *acpi_supported_cpu_ops[] __initconst = {
+static const struct cpu_operations *const acpi_supported_cpu_ops[] __initconst = {
#ifdef CONFIG_ARM64_ACPI_PARKING_PROTOCOL
&acpi_parking_protocol_ops,
#endif


6 arch/arm64/kernel/cpu_ops.c
View


6 arch/arm64/kernel/cpu_ops.c
View

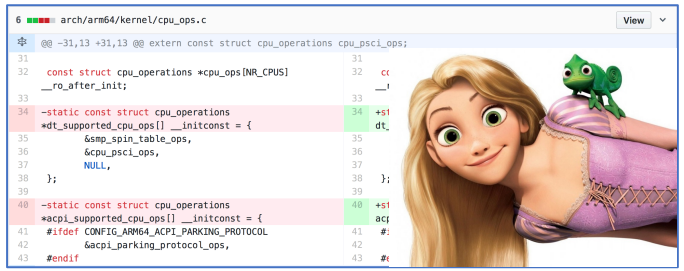

6 arch/arm64/kernel/cpu_ops.c
View

Hypothesis

Related changes are part of the same program slice.
A commit may be untangled using the program slices.

Hypothesis

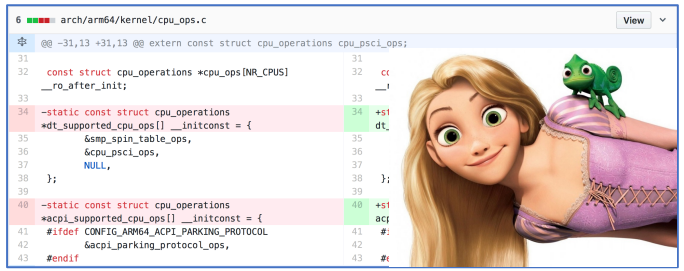
Related changes are part of the same program slice.
A commit may be untangled using the program slices.

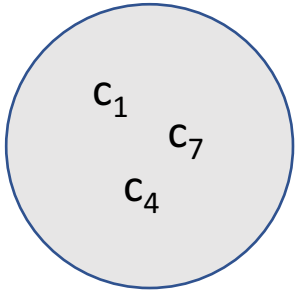


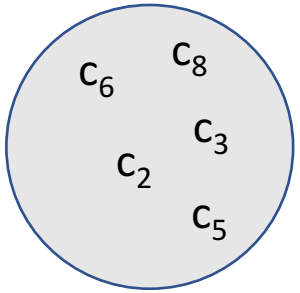
C_1 C_2 C_3 C_4 C_5 C_6 C_7 C_8

Hypothesis

Related changes are part of the same program slice.
A commit may be untangled using the program slices.



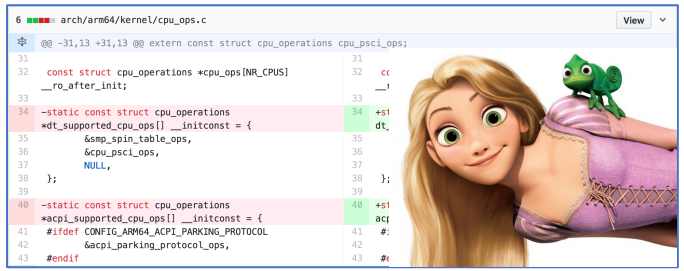


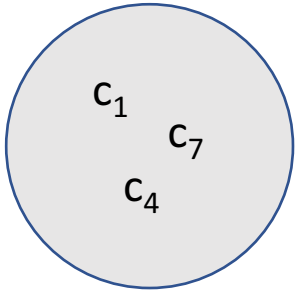


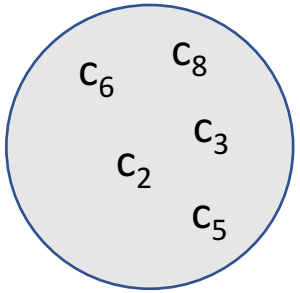
c_1 c_2 c_3 c_4 c_5 c_6 c_7 c_8

Hypothesis

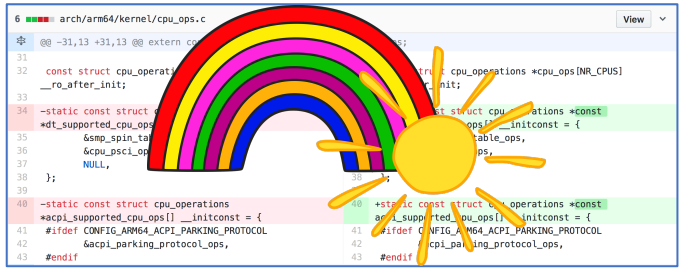
Related changes are part of the same program slice.
A commit may be untangled using the program slices.

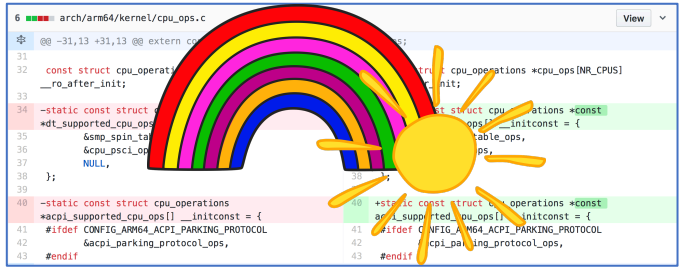




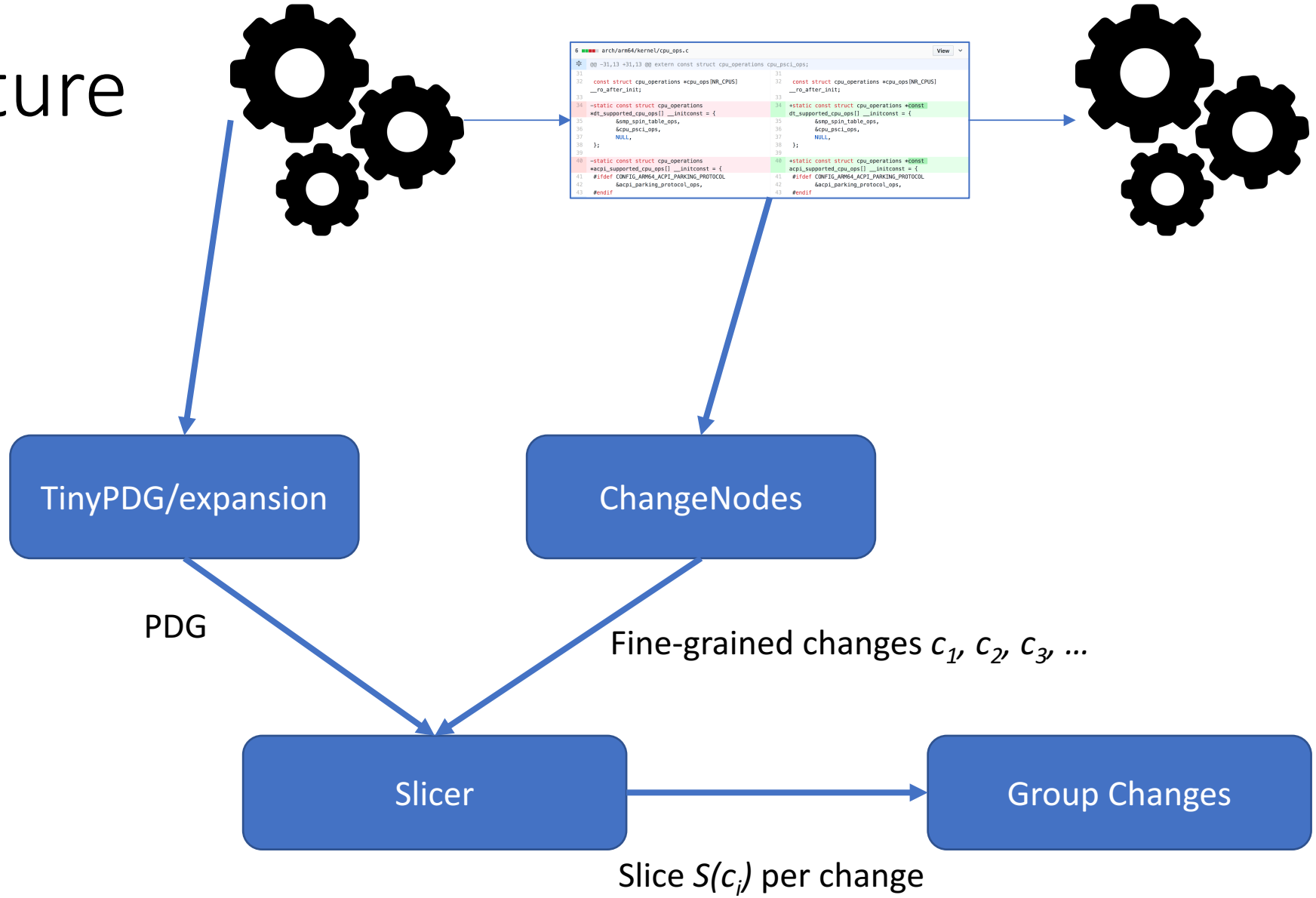


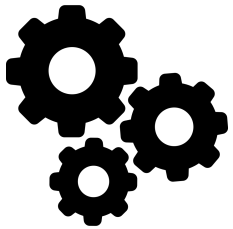
c_1 c_2 c_3 c_4 c_5 c_6 c_7 c_8





Architecture





TinyPDG
- Java
- Eclipse AST
- Intra-procedural

TinyPDG fork WIP
- Java
- Eclipse AST
- Inter-procedural

PDG

https://github.com/YoshikiHigo/TinyPDG

6 arch/arm64/kernel/cpu_ops.c View

```
@@ -31,13 +31,13 @@ extern const struct cpu_operations cpu_psci_ops;

 const struct cpu_operations *cpu_ops[NR_CPUS]
 __ro_after_init;

-static const struct cpu_operations
*dt_supported_cpu_ops[] __initconst = {
+static const struct cpu_operations *const
dt_supported_cpu_ops[] __initconst = {
        &smp_spin_table_ops,
        &cpu_psci_ops,
        NULL,
 };

-static const struct cpu_operations
*acpi_supported_cpu_ops[] __initconst = {
+static const struct cpu_operations *const
acpi_supported_cpu_ops[] __initconst = {
 #ifdef CONFIG_ARM64_ACPI_PARKING_PROTOCOL
        &acpi_parking_protocol_ops,
 #endif
```

ChangeNodes
- Java
- Eclipse AST

Fine-grained changes c_1, c_2, c_3, …

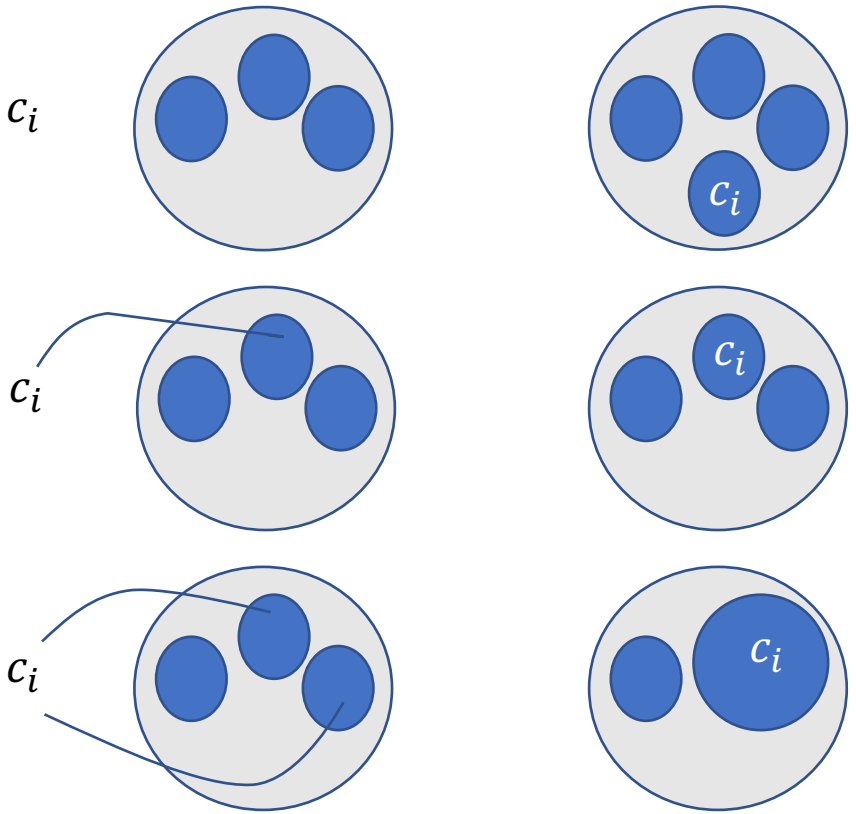
- Insert
- Update
- Move
- Delete

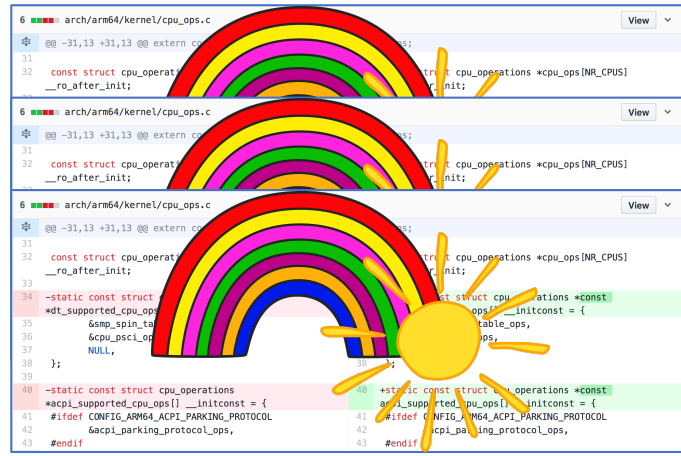
R. Stevens et al., “Extracting Executable Transformations from Distilled Code Changes,” in *SANER*, 2017.

PDG
Fine-grained change c_i
Slicer
- Backward
- Forward
Slice $S(c_i)$ per change

$$c_i \in S(c_j) \vee c_j \in S(c_i)$$

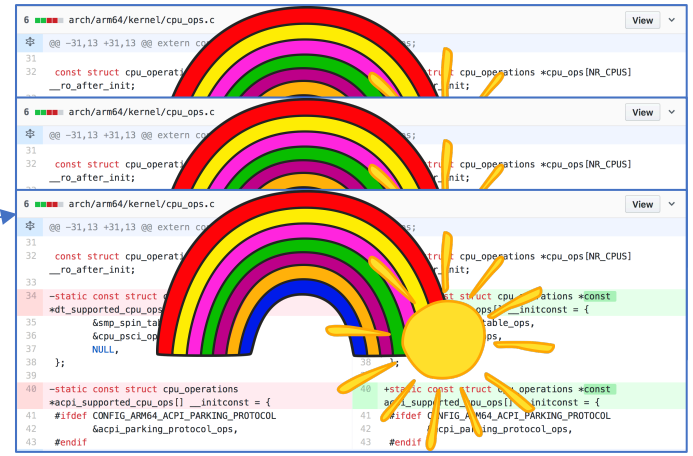
Slices $S(c_1)$, $S(c_2)$, $S(c_3)$, …





Evaluation Dataset

ArgoUML,
GWT, Jaxen,
JRuby, Xstream



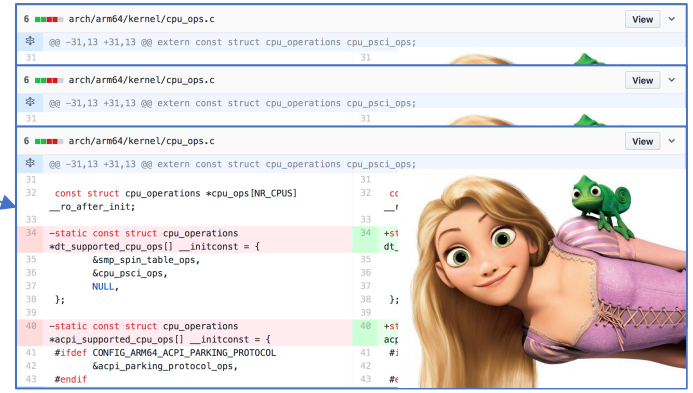


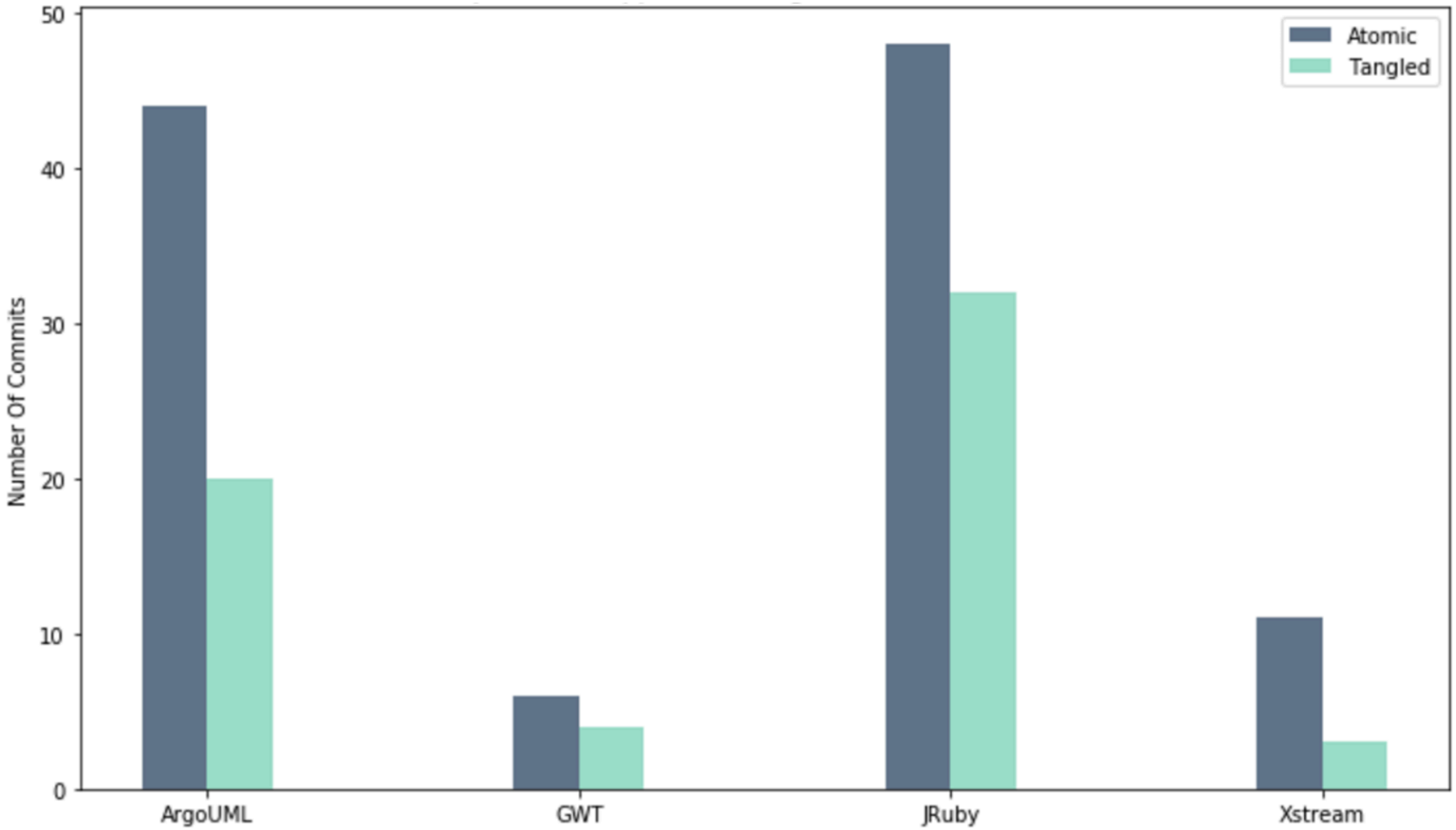
Table III
DETAILS OF PROJECTS USED DURING EXPERIMENTS.

	ArgoUML	Google Web Toolkit	Jaxen	JRuby	XStream
lines of code	164,851	266,115	20,997	101,799	22,021
#history months	150	54	114	105	90
#developers	50	120	20	67	12
Total #change sets	16481	5326	1353	11134	1756
#bug fixes	2,945	809	105	2,977	312
#atomic bug fixes	125 (4.2%)	44 (5.4%)	32 (30.5%)	200 (6.7%)	40 (12.8%)

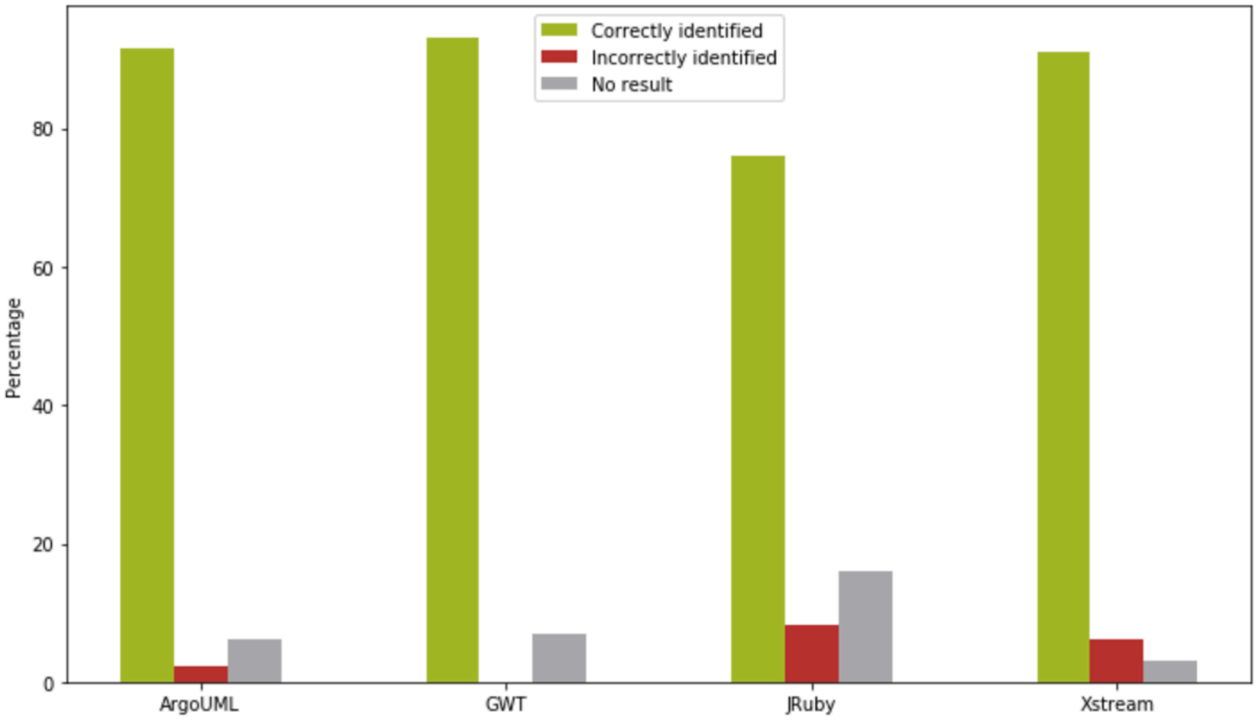
K. Herzig et al., “The impact of tangled code changes on defect prediction models,” *Empirical Software Engineering*, 2015.

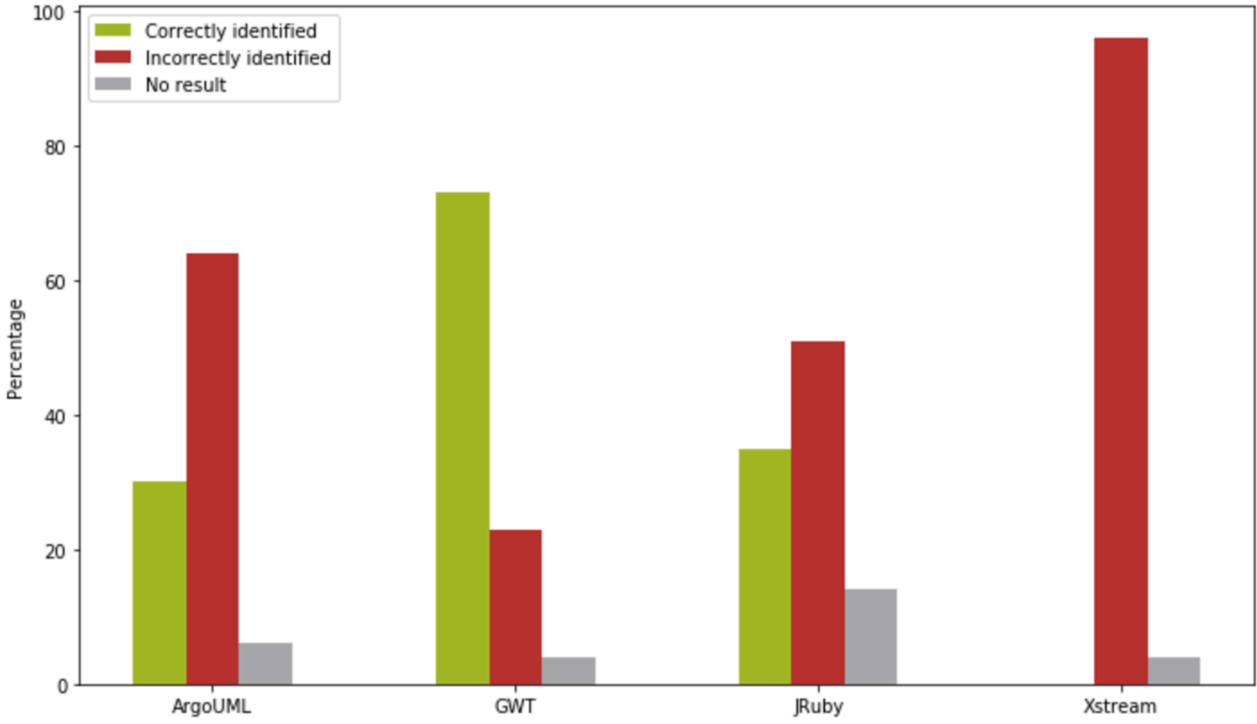
K. Herzig et al., “Untangling Changes,” Saarland University, 2011. [Tech Report]

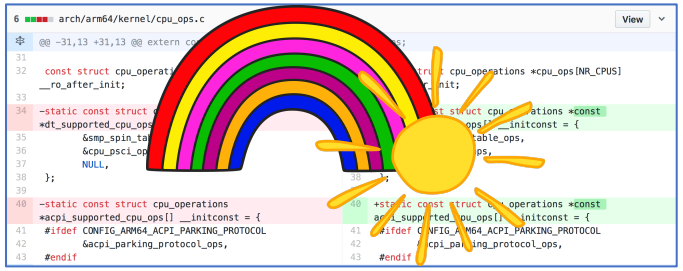
Preliminary Intra-procedural Results

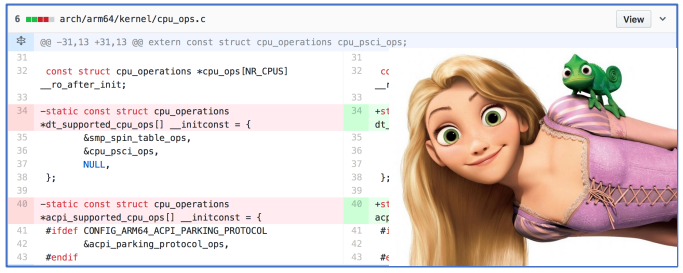


Preliminary Intra-procedural Results





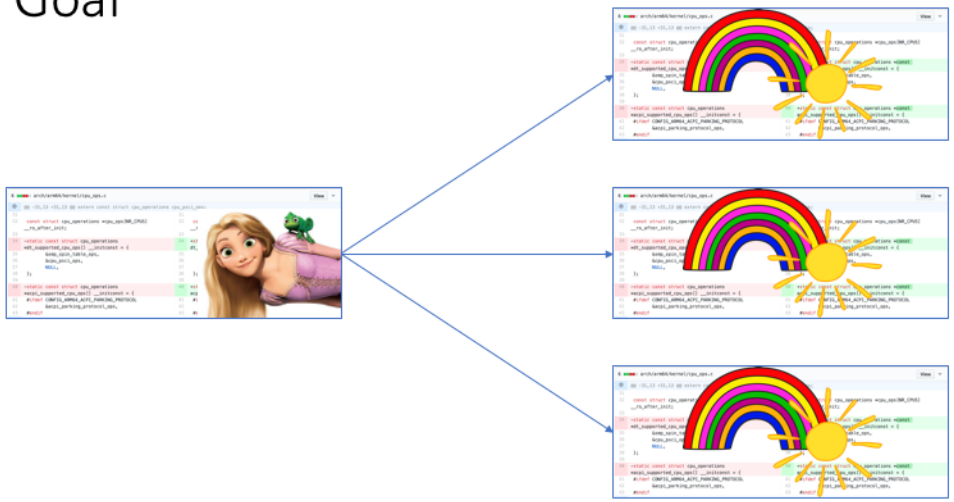




Current and Future Work

- Inter-procedural dependence graphs and slicing
- Perform full evaluation

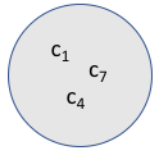
Goal

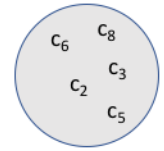


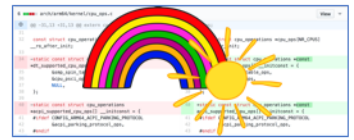
Hypothesis

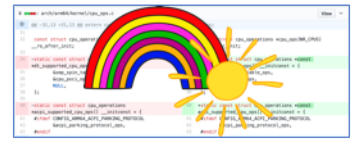
Related changes are part of the same program slice.
A commit may be untangled using the program slices.

c_1 c_2 c_3 c_4 c_5 c_6 c_7 c_8

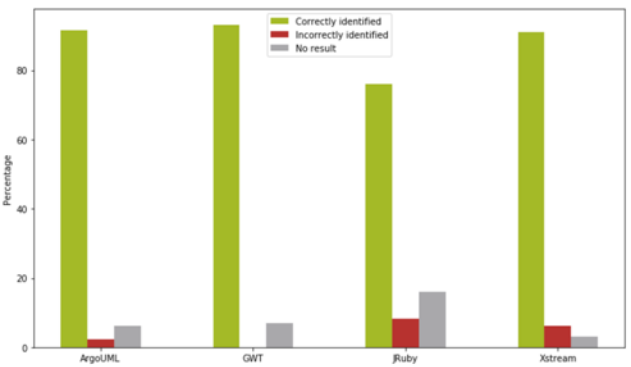


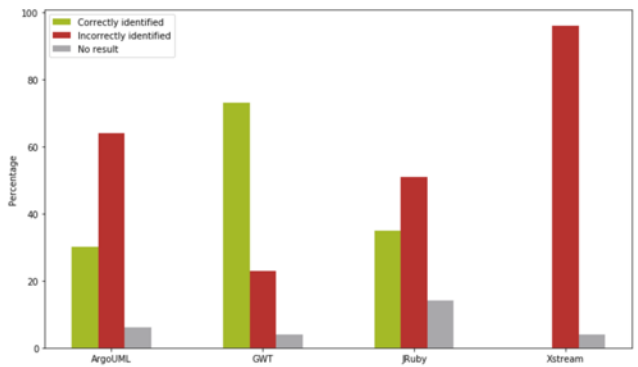






Preliminary Intra-procedural Results



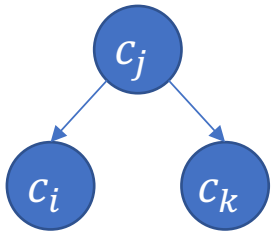


Current and Future Work

- Inter-procedural dependence graphs and slicing
- Perform full evaluation

Backup slide: reflexive, symmetric, transitive

$$c_i R c_j \Leftrightarrow c_i \in S(c_j) \vee c_j \in S(c_i)$$



$$c_i R c_j$$
$$c_j R c_k$$
$$\neg(c_i R c_k)$$

$$c_i R' c_j \Leftrightarrow \exists c_{t_1}, \ldots, c_{t_n} : c_i R c_{t_1}, \ldots, c_{t_n} R c_j$$

$$c_i R' c_j$$
$$c_j R' c_k$$
$$c_i R' c_k$$