# Using acceptance tests to predict merge conflict risk

Thaís Rocha (thais.burity@ufape.edu.br)
**Paulo Borba (phmb@cin.ufpe.br)**

Empirical Software Engineering 2023
**https://doi.org/10.1007/s10664-022-10266-8**

# Preventing or reducing merge conflicts would be good

- Conflicts might occur frequently, even with smarter (structured) merge tools
- Conflict resolution might be costly
- Conflicts might compromise system quality
- Practices for reducing conflicts might lead to other problems

# We could do that by avoiding parallel work on tasks with intersecting interfaces (changed files)

# Actual conflict in this case, the task interfaces intersect!

| Changed files by T175 |
| --- |
| app/models/question.rb |
| app/controllers/questions_controller.rb |
| … other files |

| Changed files by T176 |
| --- |
| app/models/question.rb |
| app/controllers/questions_controller.rb |
| … other files |

https://github.com/allourideas/allourideas.org

But developers might not be able to infer and specify task interfaces!

# We can infer task interfaces from tests, in a BDD context

```
Scenario: Lots of ideas
Given an idea marketplace exists with url 'test'
And idea marketplace 'test' has 100 ideas
When I go to the View Results page for 'test'
Then I should see "Idea #2"
And I should not see "Idea #50"
```



```
When /^idea marketplace '(.*)' has (\d*) ideas$/ do |url, num_ideas|
        ....
        (1..num_ideas.to_i).to_a.reverse.each do |n|
          the_params = {'auto' => 'test choices', :data => "Idea ##{n}", :question_id => @question.id}
            Choice.post(:create_from_abroad, :question_id => @question.id, :params => the_params)
        end

        unless prev_auto_activate
            @question.put(:set_autoactivate_ideas_from_abroad, :question => { :it_should_autoactivate_ideas => false})
        end
end
```

# TestI interfaces generated by our tool TAITIr

| TestI($T_{175}$) |
| --- |
| app/controllers/home_controller.rb |
| app/controllers/questions_controller.rb |
| app/models/choice.rb |
| app/models/earl.rb |
| app/models/item.rb |
| app/models/question.rb |
| app/views/abingo_dashboard/_experiment_row.html.haml |
| app/views/abingo_dashboard/index.html.haml |
| app/views/home/about.html.haml |
| app/views/home/index.html.haml |
| app/views/home/privacy.html.haml |
| app/views/questions/_idea.html.haml |
| app/views/questions/about.html.haml |
| app/views/questions/admin.html.haml |
| app/views/questions/new.html.haml |
| app/views/questions/results.html.haml |
| app/views/questions/voter_map.html.erb |
| app/views/questions/word_cloud.html.erb |
| app/views/shared/_google_jsapi.html.haml |
| app/views/shared/_header_vote.html.haml |
| app/views/shared/_highcharts_header.html.haml |

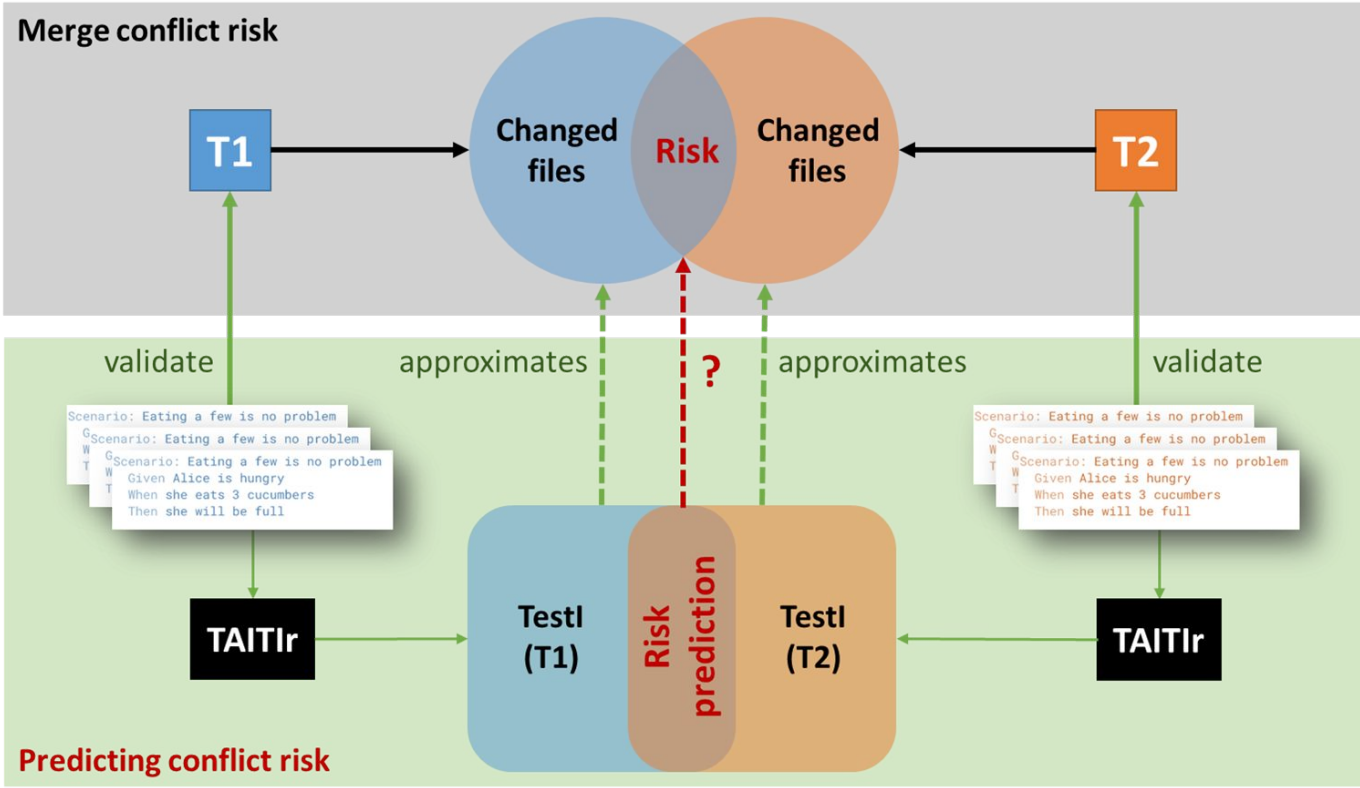
| TestI($T_{176}$) |
| --- |
| app\controllers\questions_controller.rb |
| app\models\choice.rb |
| app\models\earl.rb |
| app\models\question.rb |
| app\views\abingo_dashboard_experiment_row.html.haml |
| app\views\abingo_dashboard\index.html.haml |
| app\views\questions\new.html.haml |

| Legend | |
| --- | --- |
| | Intersection between interfaces |
| | Conflicting files |

**Can the generated *test interfaces* be used to predict the corresponding *task interfaces*?**

# Evaluation

- 990 tasks
- 19 Rails projects
- 6,360 task pairs




*https://thaisabr.github.io/conflict-risk-prediction-study-site/*

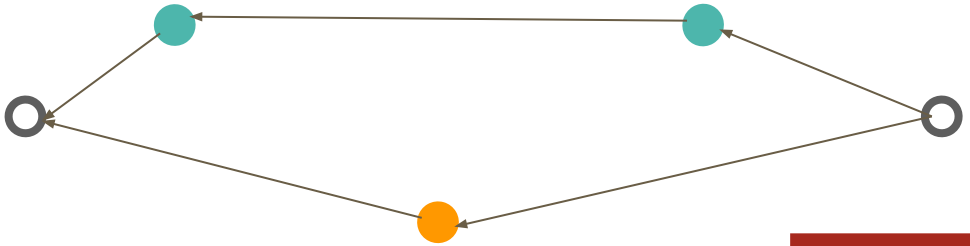
# Collecting tasks

**Task 175**

Refactorings

**Task 176**

Bug fix and enhancement of GUI layout



**merge commit**

https://github.com/allourideas/allourideas.org

RQ1: Are tasks with non-disjoint TestI interfaces associated with higher merge conflict risk?

RQ2: How often does TestI predict conflict risk between two tasks?

# Tasks with non-disjoint TestI interfaces more likely modify files in common

They are **2.07 times** more likely to change a file in common

# A minimal intersection is the best predictor

| Intersection lower bound | Predicted positive condition rate (%) | Precision | Recall | $F_2$ |
|---|---|---|---|---|
| 1 | 96.70 | 0.59 | 0.98 | 0.86 |
| 2 | 92.41 | 0.60 | 0.95 | 0.85 |
| 3 | 88.07 | 0.60 | 0.91 | 0.83 |
| 4 | 78.90 | 0.62 | 0.84 | 0.78 |
| 5 | 71.07 | 0.63 | 0.77 | 0.74 |

**58.4% task pairs are risky**

# Discarding test setup when computing TestI interfaces might benefit precision

```
Scenario: User deactivates an active choice
Given an idea marketplace exists with admin 'test@test.com'
and url 'test' and 4 ideas
And I sign in as test@test.com/password
And I am on the Admin page for 'test'
When I click on the toggle link for the first choice
Then the first choice should be Deactivated
```

**Filtering out files accessed by Given steps**

| Intersection lower bound | Precision | Recall | F2 |
|---|---|---|---|
| 1 | +8.3% | -9.4% | -4.9% |
| 4 | +7.6% | -21.9% | -16.1% |

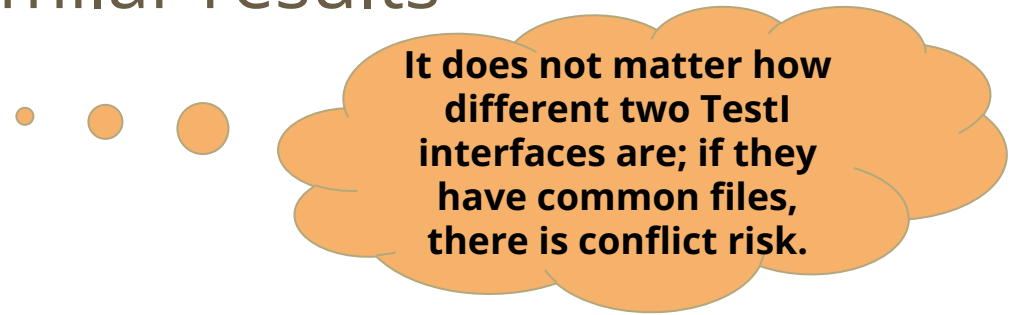
## RQ3: Is the intersection size between two TestI interfaces proportional to the number of files changed in common by the corresponding tasks?

## The larger the intersection between TestI, the higher the risk of merge conflict between tasks

Small correlation - Cohen's assignment of effect size's relative strength

# A predictor based on TestI intersection outperforms alternative predictors (RQ4 and RQ5)

- **TextI**: Improves precision and reduces recall
- **TestI similarity**: similar results



Test-first context
&
significant test coverage

=>

Test-based task interfaces as an *additional* factor to consider for scheduling programming tasks

Smaller overlapping task interfaces

=>

reduced chances of merge conflict when independently performing these tasks

When picking a new task, developers might minimize conflict risk by computing task interfaces and risk rate (of tasks being currently executed)

# Using acceptance tests to predict merge conflict risk

Thaís Rocha (thais.burity@ufape.edu.br)
**Paulo Borba (phmb@cin.ufpe.br)**

Empirical Software Engineering 2023
**https://doi.org/10.1007/s10664-022-10266-8**