TU WIEN TECHNISCHE UNIVERSITÄT WIEN Vienna University of Technology

uff Universidade Federal Fluminense

# Improving Model Inspection Processes with Crowdsourcing: Findings from a Controlled Experiment

Dietmar Winkler[1], Marta Sabou[1], Sanja Petrovic[1], Gisele Caneiro[2], Marcos Kalinowksi[2], Stefan Biffl[1]

[1] Vienna University of Technology, Institute of Software Technology and Interactive Systems, Vienna, Austria

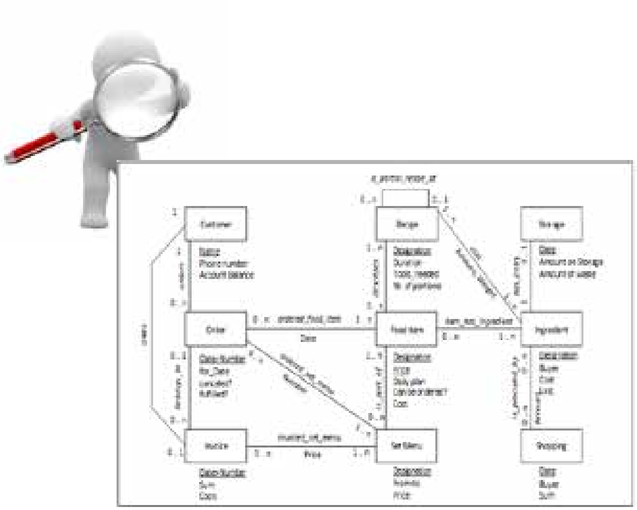
[2] Fluminese Federal University, Graduate Program in Computing, Niteroi, RJ, Brazil

qse.ifs.tuwien.ac.at

# Motivation & Goals

## Context

- Large scale software engineering models for describing the system structure and behavior.
- Established software inspection for early and efficient defect detection (model vs. reference documents) with limited resources.
- Crowdsourcing mechanism can help to distribute the work load among a group of experts.



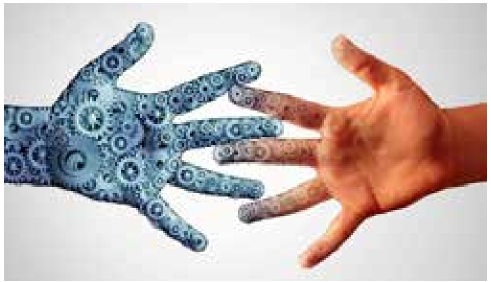
Model Inspection

## Key questions

- How to handle large-scale engineering models with limited resources?
- How to better coordinate inspection tasks for inspecting large-scale artefacts within an inspection team?
- How to provide appropriate tool support for inspection handling?

## Goal of this Presentation

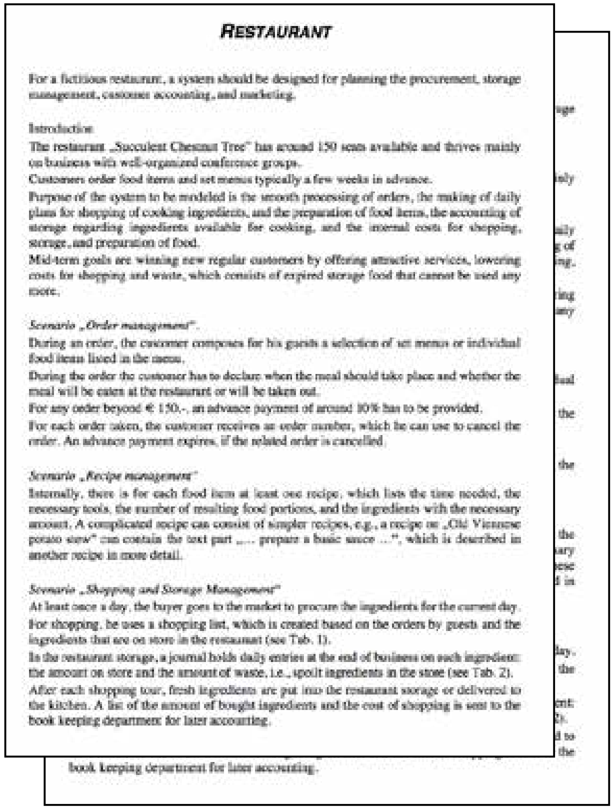
- Definition and evaluation of a Crowdsourced Inspection (CSI) Process with tool support.



# Illustrative Example .. the starting point

## Inspection Task

- Input: Reference document, e.g., requirements specifications.
- Task: Identify defects in (large-scale) models early, effective, and efficient.
- Output: True defects in the model.



### RESTAURANT

For a fictitious restaurant, a system should be designed for planning the procurement, storage management, customer accounting, and marketing.

Introduction

The restaurant „Succulent Chestnut Tree" has around 150 seats available and thrives mainly on business with well-organized conference groups.

Customers order food items and set menus typically a few weeks in advance.

Purpose of the system to be modeled is the smooth processing of orders, the making of daily plans for shopping of cooking ingredients, and the preparation of food items, the accounting of storage regarding ingredients available for cooking, and the internal costs for shopping, storage, and preparation of food.

Mid-term goals are winning new regular customers by offering attractive services, lowering costs for shopping and waste, which consists of expired storage food that cannot be used any more.

*Scenario „Order management"*.

During an order, the customer composes for his guests a selection of set menus or individual food items listed in the menu.

During the order the customer has to declare when the meal should take place and whether the meal will be eaten at the restaurant or will be taken out.

For any order beyond € 150,-, an advance payment of around 10% has to be provided.

For each order taken, the customer receives an order number, which he can use to cancel the order. An advance payment expires, if the related order is cancelled.

*Scenario „Recipe management"*

Internally, there is for each food item at least one recipe, which lists the time needed, the necessary tools, the number of resulting food portions, and the ingredients with the necessary amount. A complicated recipe can consist of simpler recipes, e.g., a recipe on „Old Viennese potato stew" can contain the text part „... prepare a basic sauce ...", which is described in another recipe in more detail.

*Scenario „Shopping and Storage Management"*

At least once a day, the buyer goes to the market to procure the ingredients for the current day.

For shopping, he uses a shopping list, which is created based on the orders by guests and the ingredients that are on store in the restaurant (see Tab. 1).

In the restaurant storage, a journal holds daily entries at the end of business on each ingredient: the amount on store and the amount of waste, i.e., spoilt ingredients in the store (see Tab. 2).

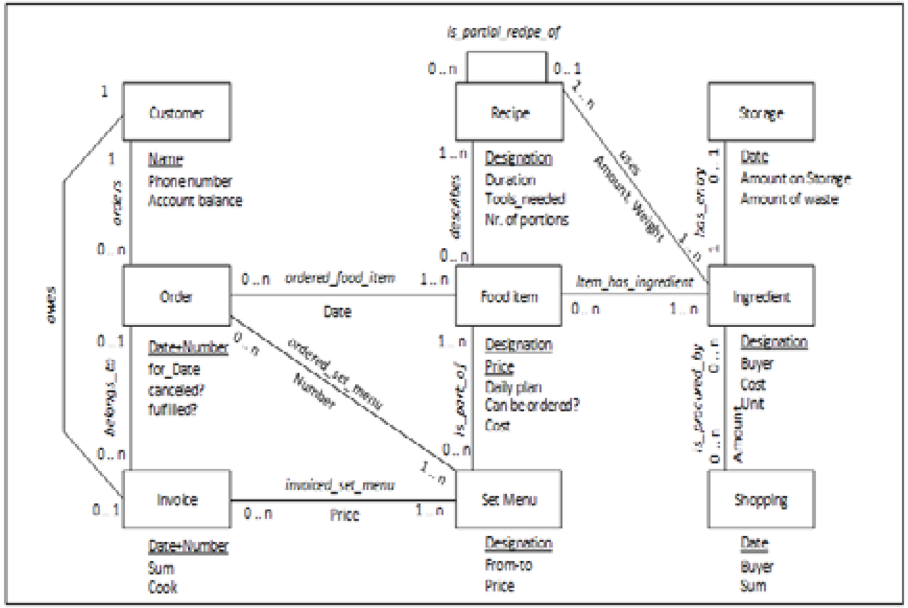
After each shopping tour, fresh ingredients are put into the restaurant storage or delivered to the kitchen. A list of the amount of bought ingredients and the cost of shopping is sent to the book keeping department for later accounting.

book keeping department for later accounting.

Requirements Specification

Does the model completely and correctly represent the specification?

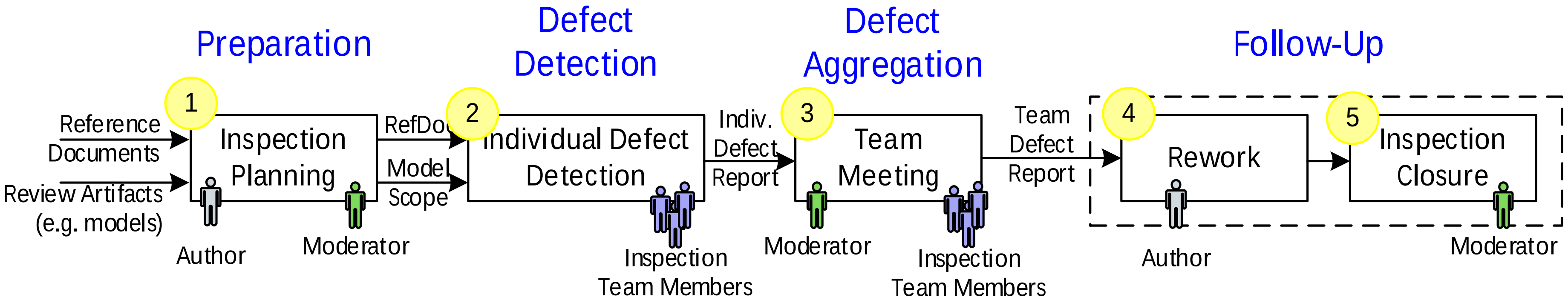
Are there defects in the model?



System EER Diagram Model

# Software Reviews / Inspections

## Related Work



**Benefits:**

§ Formal and structured process approach (five inspection phases) to identify defect early and efficient in engineering artifacts.

§ Well established and investigated process approach.

§ Guidelines and reading techniques support defect detection, e.g., perspectives or scenarios.

**Limitations**:

§ Typically (expensive) experts are part of the inspection team.

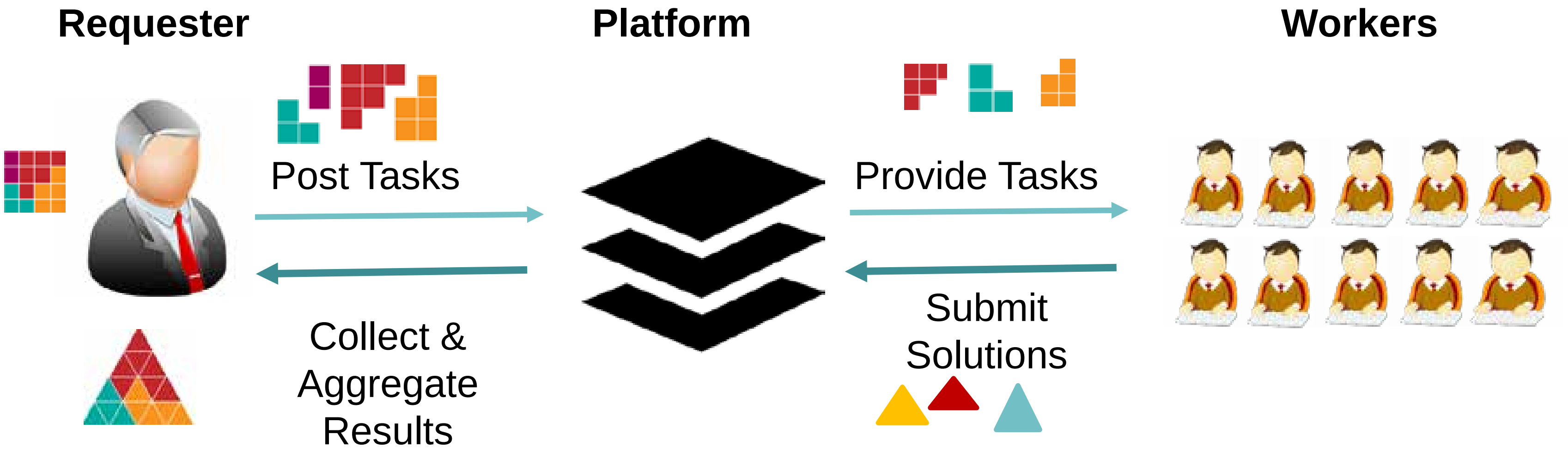
§ Limited resources (e.g., 2h of inspection) à for large-scale documents need for several inspection cycles and coordination.

§ Limited tool support.

# Crowdsourcing in Software Engineering

## Related Work

“The act of undertaking any external software engineering tasks by an undefined, potentially large group of online workers in an open call format.” (Mao et al., 2016)



Requester

Platform

Workers

Post Tasks

Collect & Aggregate Results

Provide Tasks

Submit Solutions

- Crowdsourcing (CS) mechanism has been applied in software engineering planning and analysis, implementation, maintenance, and testing ..
- .. but very limited in the area of Software Quality Assurance or Software Inspection.

K. Mao, L. Capra, M. Harman, Y. Jia. A survey of the use of crowdsourcing in software engineering. Journal of Systems and Software, 2016.

# Research Issues

### Goal

§ Support of software inspection tasks with crowdsourcing techniques.

§ Key Elements:

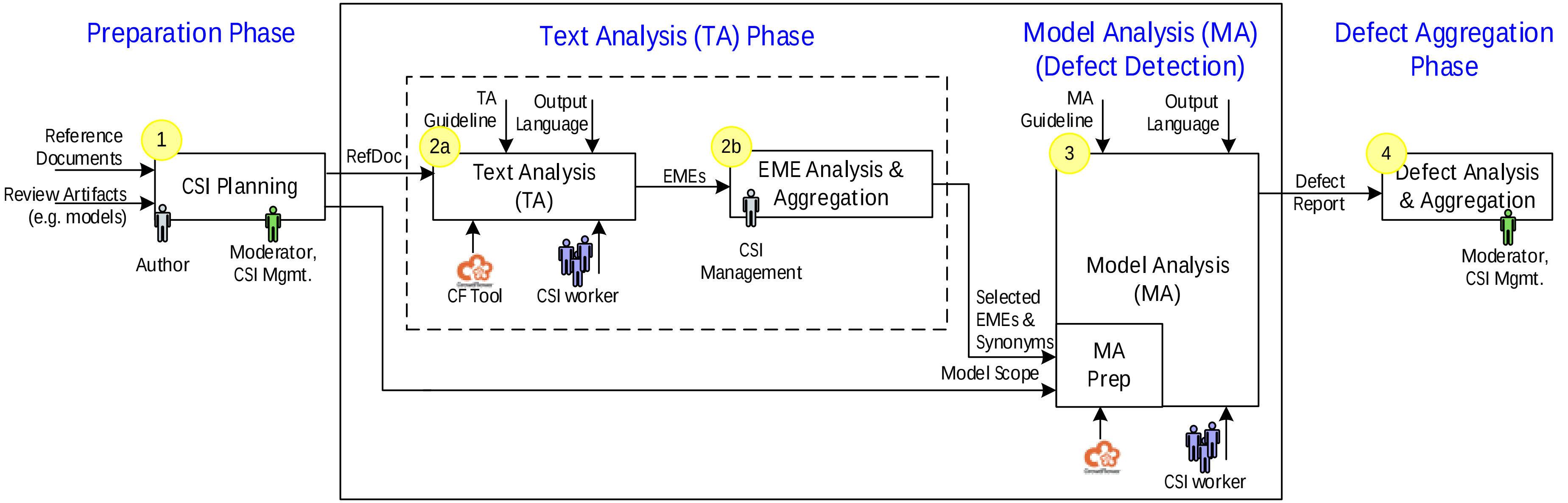
- Splitting up inspection tasks (for large models) into small pieces of work,
- Distributing inspection work load to a crowd of workers and/or experts within an organization,
- Improving inspection control due to feedback cycles.
- Providing tool support.

### Questions

§ How to design an inspection process with crowdsourcing mechanisms?
à Approach: Crowdsourced Software Inspection (CSI) Process.

§ What are effects of the CSI process approach compared to traditional inspections?
à Approach: Controlled experiment for evaluation.

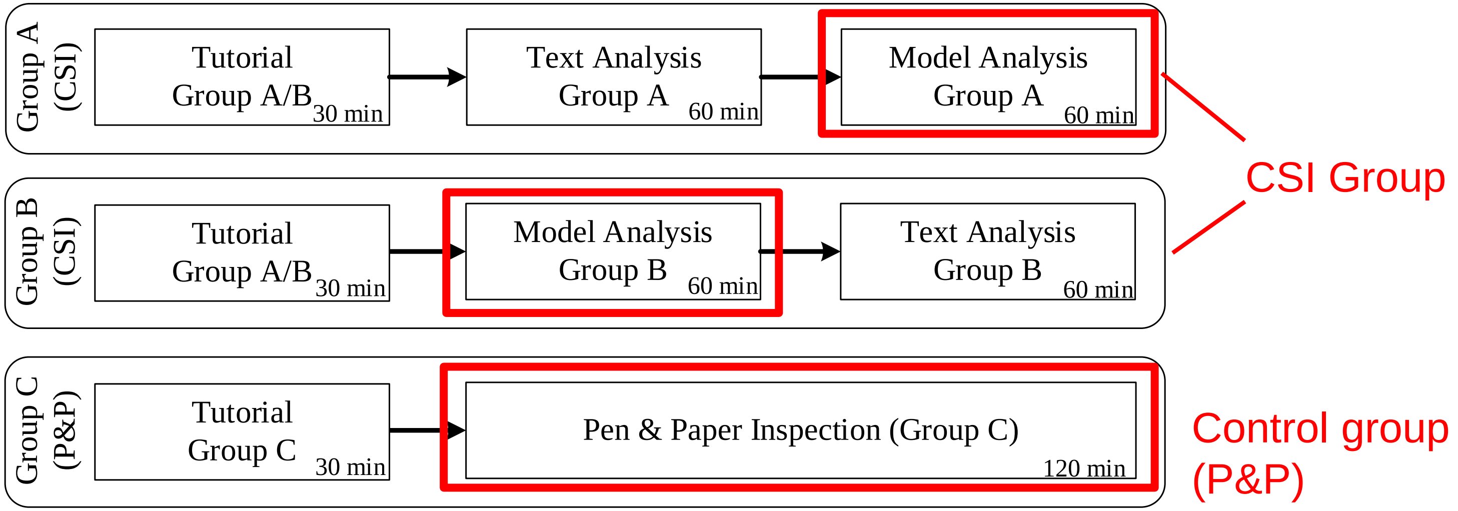
# Crowdsourced Software Inspection (CSI)



- **Planning & Preparation (step 1)**
- **Text Analysis (step 2)**
  - 2a. Identification of *Expected Model Elements*, e.g., entities, attributes, relationships.
  - 2b. Aggregation of individual EME results.
- **Model Analysis (step 3)**
  - 3. Model analysis based on *Expected Model Elements* (EMEs) to identify candidate defects.
- **Aggregation of individual candidate defects (step 4)**

# Experimental Study

## Study Design



Group A (CSI)
Tutorial Group A/B 30 min
Text Analysis Group A 60 min
Model Analysis Group A 60 min
Group B (CSI)
Tutorial Group A/B 30 min
Model Analysis Group B 60 min
Text Analysis Group B 60 min
Group C (P&P)
Tutorial Group C 30 min
Pen & Paper Inspection (Group C) 120 min

CSI Group

Control group (P&P)

- Study Type: Controlled Experiment
- CSI process vs. Traditional Best-Practice Inspection (control group) with cross-over design.
- 75 participants in academic course in 4 sessions (63 crowd workers; 12 inspectors).
- Study Material:
  - Design Specification: 3 pages, 7 scenarios and 110 EMEs.
  - EER Diagram: 9 entities, 13 relationships, 32 attributes; 33 seeded defects.
  - Questionnaires (experience and feedback), guidelines for task execution.
  - Tool: *Crowdflower*[1] application and configuration.

[1] Crowdflower: www.crowdflower.com

# Study Results: Effectiveness

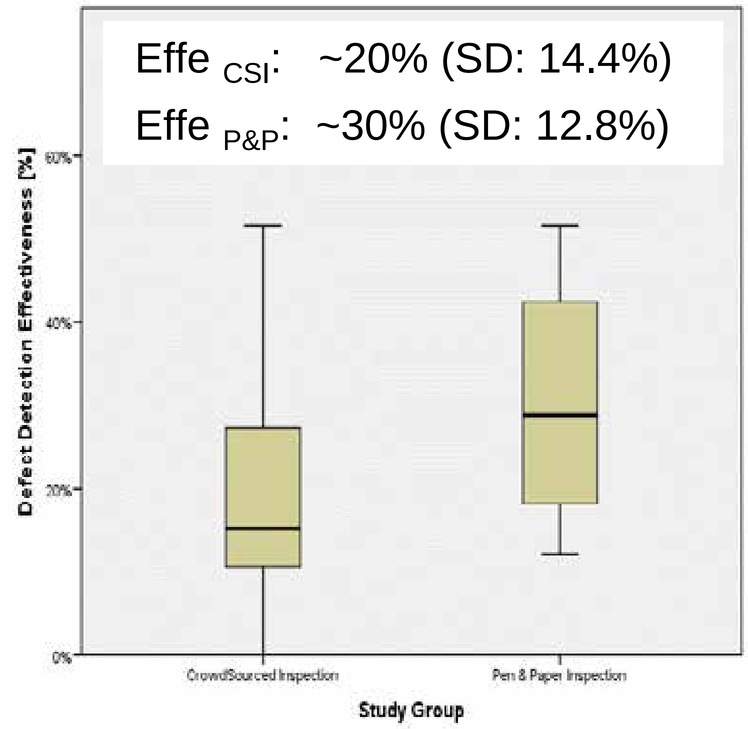
- Effectiveness is defined as share of identified true defects and seeded defects.
  - 33 seeded defects represent typical defects in the domain.

| | | Reported Defects | | True Defects | |
|---|---|---|---|---|---|
| Group | No. part. | Mean | SD | Mean | SD |
| CSI | 63 | 15 | 6.5 | 7 | 4.9 |
| P&P | 12 | 21 | 5.7 | 10 | 4.6 |

- In the study context, traditional P&P inspection performs significantly better compared to the CSI MA approach.

- However …
  - P&P spent more time on defect detection.
  - CSI focuses on certain parts of the system.

- Conclusion: More detailed investigations on the scope of the defect detection part is required.



Effe $_{CSI}$: ~20% (SD: 14.4%)

Effe $_{P\&P}$: ~30% (SD: 12.8%)

# Study Results: False Positives

- False Positives are defined as wrongly reported defects, i.e., reported candidate defects that cannot be mapped to seeded defects.

- Goal: low number of false positive because of additional and high analysis and aggregation effort.

| Group | No. Part. | Mean | SD | Min | Max |
|---|---|---|---|---|---|
| CSI | 63 | 8 | 5.0 | 1 | 18 |
| P&P | 12 | 11 | 4.7 | 5 | 22 |

- In the study context,
  CSI performs better (but not significantly) compared to traditional P&P inspection.

- Conclusion:
  - Model Analysis guidance by Expected Model Elements (EMEs) can keep the inspection focused and can lead to a lower number of false positives.

# Study Results: Efficiency

- Efficiency is defined as identified true defects per time interval (e.g., calculated per hour).

- **Defect Detection Efficiency with focus on Defect Detection Tasks**
  - Defect detection based on a given set of EMEs.
  - Focus on MA.
  - In the study context,
    CSI performs better (but not significantly) compared to traditional P&P inspection.

| Group | No. Part. | Mean | SD | Min | Max |
|---|---|---|---|---|---|
| CSI | 63 | 7.5 | 5.29 | 0 | 23 |
| P&P | 12 | 5.7 | 2.17 | 2 | 9 |

- **Defect Detection Efficiency for the overall CSI process (i.e., TA + MA)**
  - Identifying EMEs (TA) is part of the CSI process approach and need to be considered.
  - Overall effort increases and efficiency for CSI decreases.

| Group | No. Part. | Mean | SD | Min | Max |
|---|---|---|---|---|---|
| CSI | 63 | 3.5 | 2.46 | 0 | 11 |
| P&P | 12 | 5.7 | 2.17 | 2 | 9 |

- Conclusions:
  - Given EMEs can help to increase defect detection efficiency.
  - Natural language processing approaches can be used for EME identification as foundation for model analysis.

# Summary and Future Work

### Summary

- The Crowdsourced Software Inspection (CSI) process approach can support defect detection in large software models with tool support.
- Results of a controlled experiment showed promising result for defect detection performance, i.e., effectiveness, false positives, and efficiency.

### Current Limitations of the CSI approach

- Focus on a small software model (EER) in context of this study.
- Tool support needs considerable human effort for configuration.

### Future work

- Detailed and further analysis of study data needed.
- Further improvement of the CSI process.
  - Automation supported EME identification.
  - Extended and improved tool support.
- Establishing a family of experiments, that focuses on
  - Different model types (e.g., behavioral models)
  - Different model sizes (towards large-scale engineering models)
- Field study with industry models and industry people as expert “crowd”.

Ongoing Research

# Thank you ...

## Improving Model Inspection Processes with Crowdsourcing: Findings from a Controlled Experiment

Dietmar Winkler[1], Marta Sabou[1], Sanja Petrovic[1],
Gisele Caneiro[2], Marcos Kalinowksi[2], Stefan Biffl[1]

[1] Vienna University of Technology,
Institute of Software Technology and Interactive Systems, Vienna, Austria
dietmar.winkler@tuwien.ac.at

[2] Fluminese Federal University,
Graduate Program in Computing, Niteroi, RJ, Brazil