



Empirical Evaluation of  
Impact, Efficiency & Effectiveness of  
Defect-detection Techniques in  
the Event-driven Software Domain:  
***Code Reading  
and  
Functional Testing***

Department of Informatics,  
System, and Production, DISP

Experimental Informatics  
Group, URM2 ESEG

# ***Experiment Process Report***

**ØExperiment Definition**

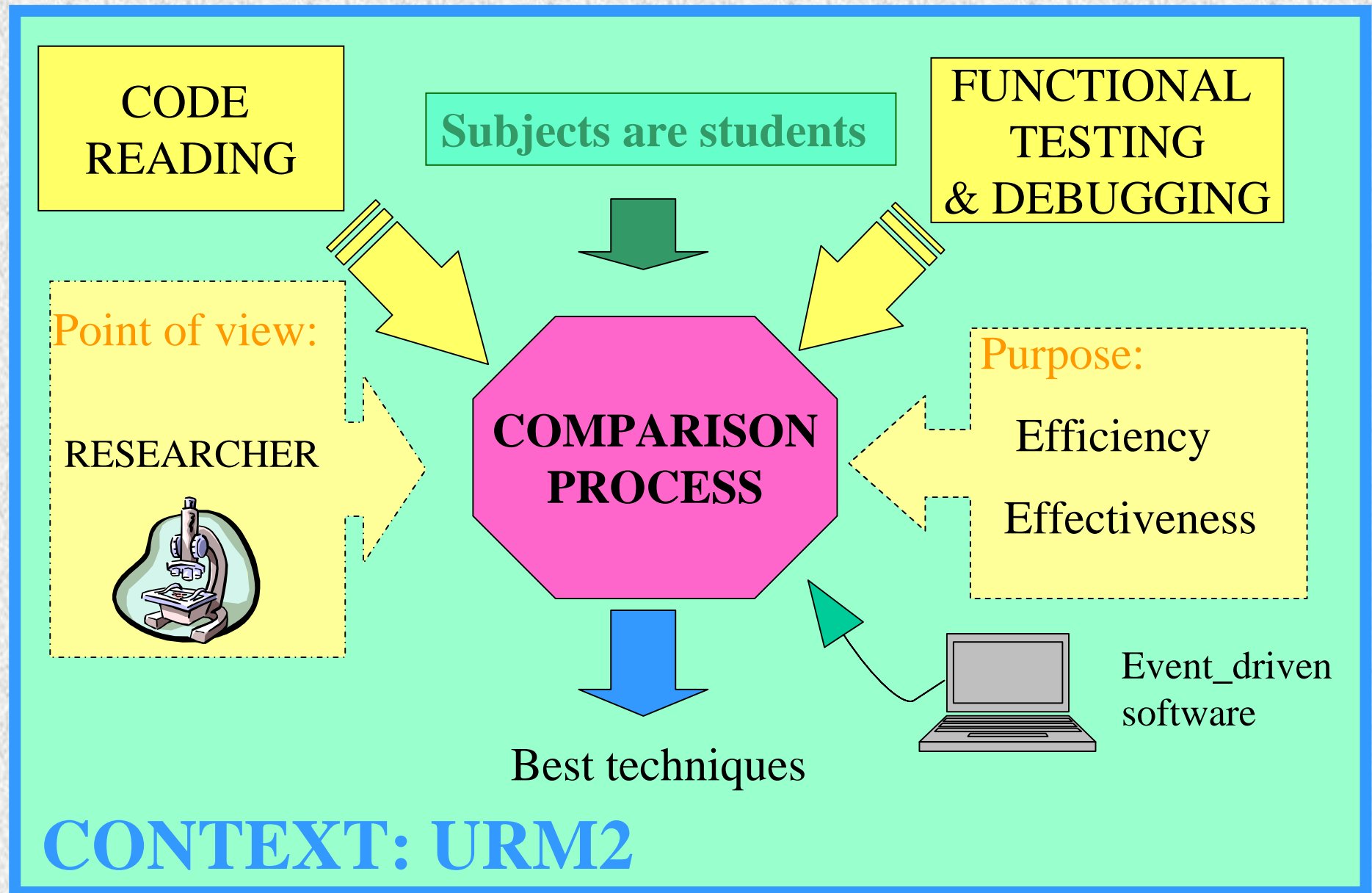
**ØExperiment Planning**

**ØOperation**

**ØStatistical Analysis Approach**

**ØSummary and Future Planes**

# Experiment Definition





# Experiment planning

Retrospect,  
initial  
HYPOTHESES

**H0: There is no significantly difference between Code Reading and, Functional Testing & Debugging**

**H1: There is significantly difference between Code Reading and, Functional Testing & Debugging**

## *Independent variables*

Elements:

Code Reading  
Functional Testing  
& Debugging

Factors

Limited time  
Faults seeded

**COMPARISON  
PROCESS**

## *Dependent variables*

Efficiency

Effectiveness

**Low level experience**

# Experiment planning

---

## Independent Variables:

### 1. Faults spread according to previously defined categories

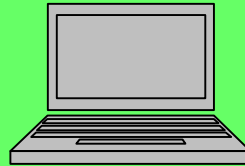
<i>fault category</i>	<i>examples</i>	<i># seed</i>
Initialization	wrong initialization of attributes	7
Computing	wrong computations of variables	6
Control	wrong definitions of logic variables	5
B&U complex structures of data	elements unrelated	5
Graphical interface	wrong settings of interface windows	6
Functionality	wrong realization	6
Events managing	wrong management	1
Exceptions handling	unforeseen produced exception	2

**Total**  
**38**

# Experiment planning

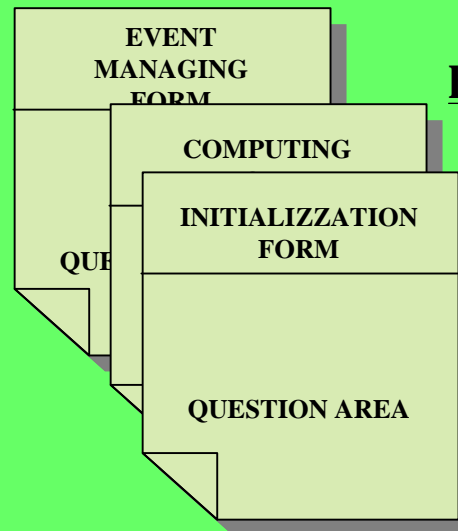
## Measurement Instruments

**Application:**

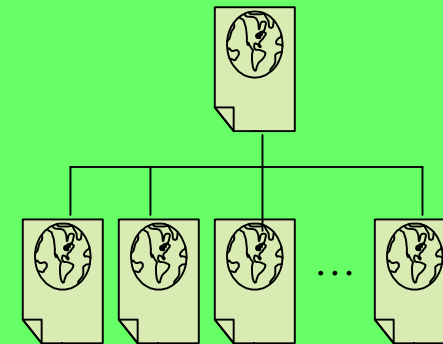


**Event\_driven software**

**Form  
&  
Web-Site:**



**<http://160.80.100.250/Esperimento/>**



**Training**

# Automated Data Collection System

---



Why automating data collection?

To run the experiment  
we just need machines  
connected to the internet

Avoid data collection  
mistakes and boredom

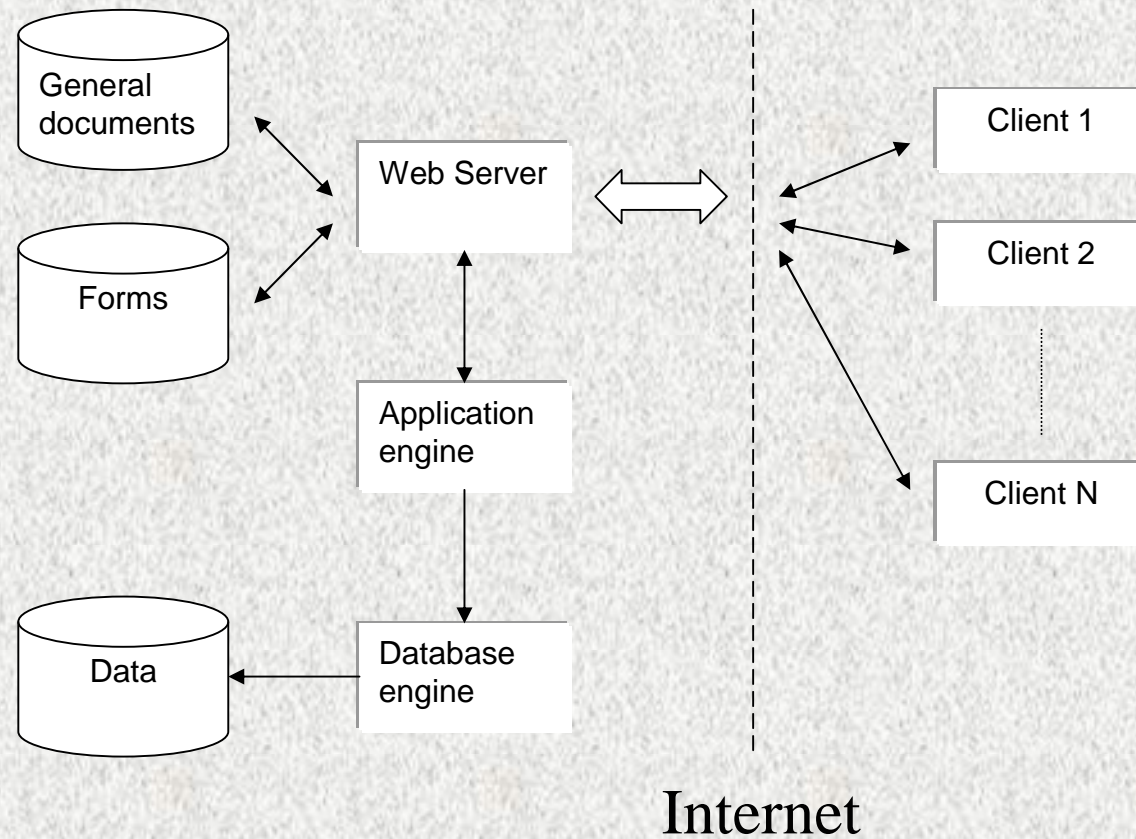


# Automated Data Collection System

---



## System implementation





# Automated Data Collection System

---

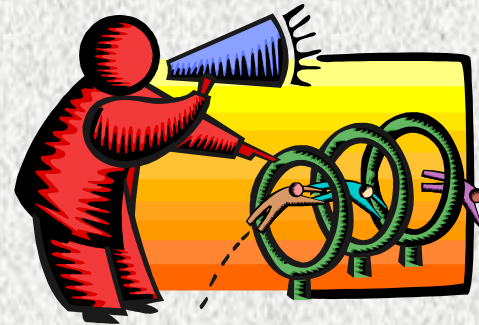
## Summary

- Ø Automated data collection guarantees correct data and site independency
- Ø Data collected are stored directly into a database, ready for processing
- Ø Easy experiment replication

# Operation

---

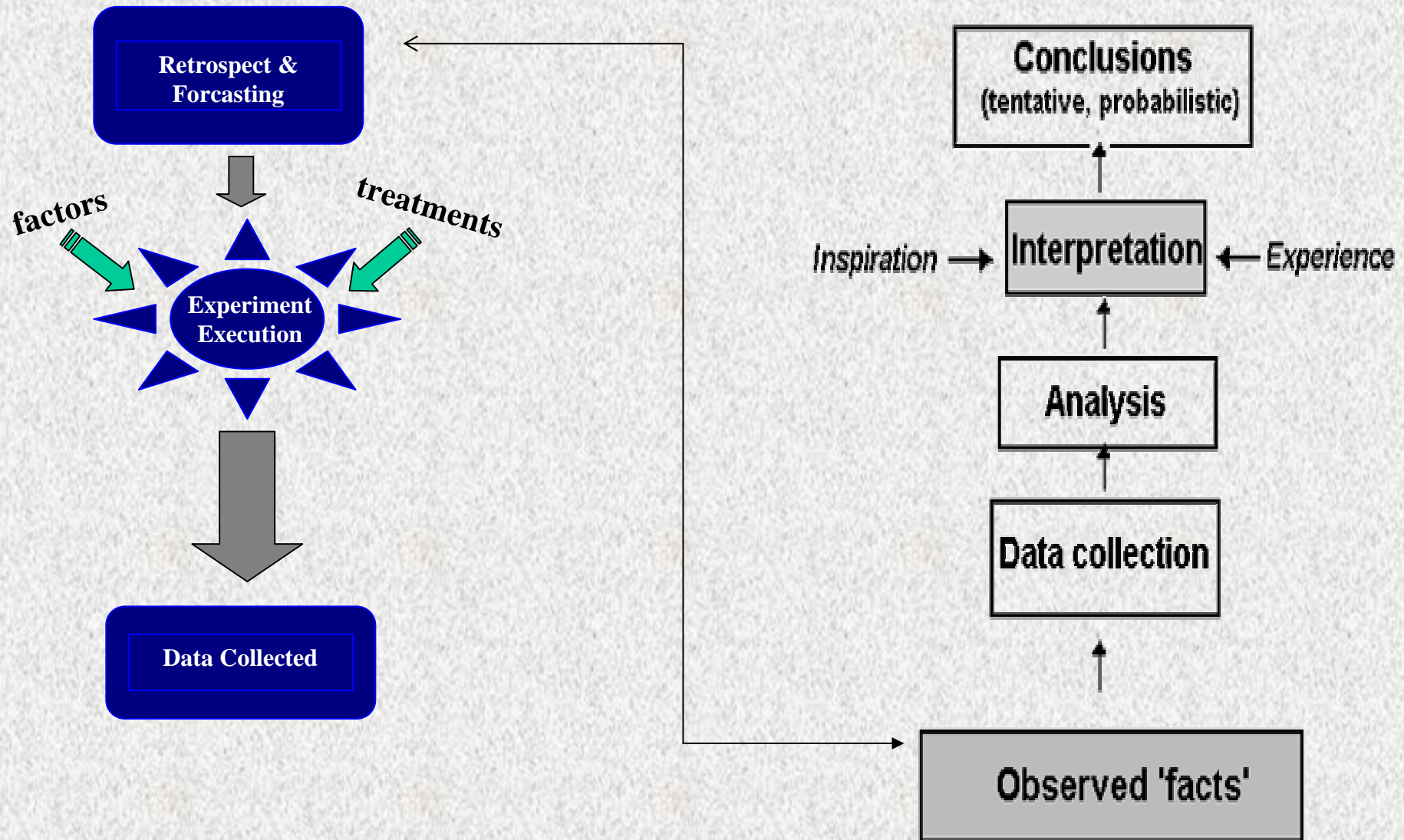
Preparation



Execution

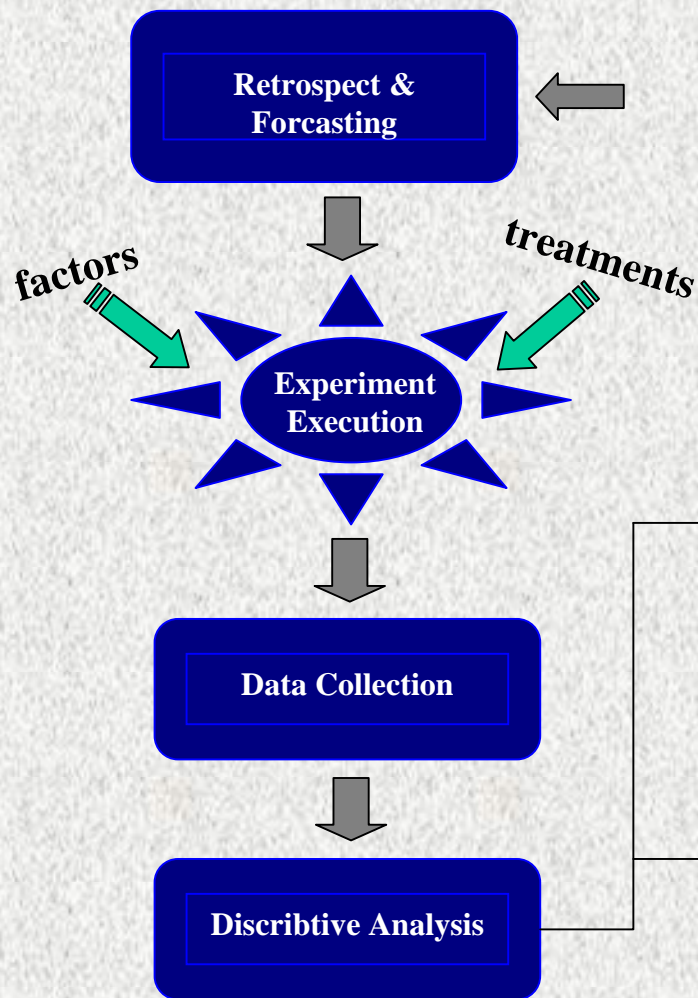


# *strategies of statistical analysis process*

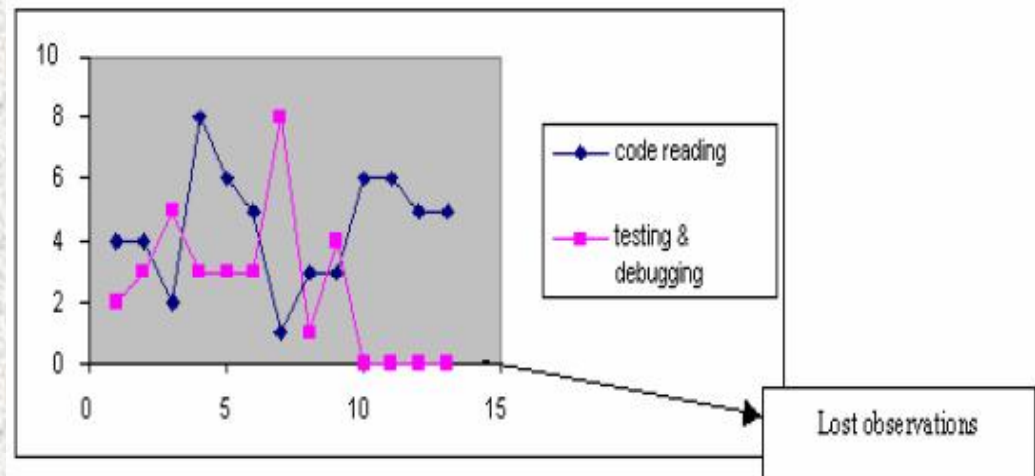




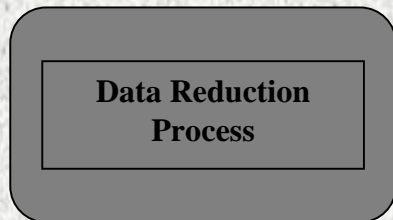
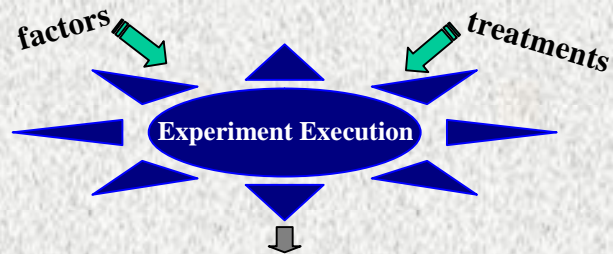
# *Descriptive statistics*



Code reading		Testing & debugging	
Average of total errors detected	Average time related Per minute	Average of total errors detected	Average time related Per minute
4	92 min	2	56 min



# *Data reduction & enhancement*



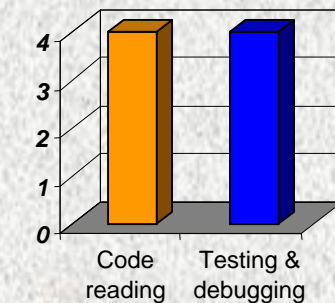
Code reading		Testing & debugging	
Average of total errors detected	Average time related Per minute	Average of total errors detected	Average time related Per minute
4	92 min	2	56 min

Code reading		Testing & debugging	
Average of total errors detected	Average time related Per minute	Average of total errors detected	Average time related Per minute
4	92 min	4	79 min

Average number of error detected



Average number of error detected



# Data Reclassification

Data Collected



Average of detecting faults

*Code reading*

**EQUAL**

*Testing & Debugging*

???



Matching &  
validation Process



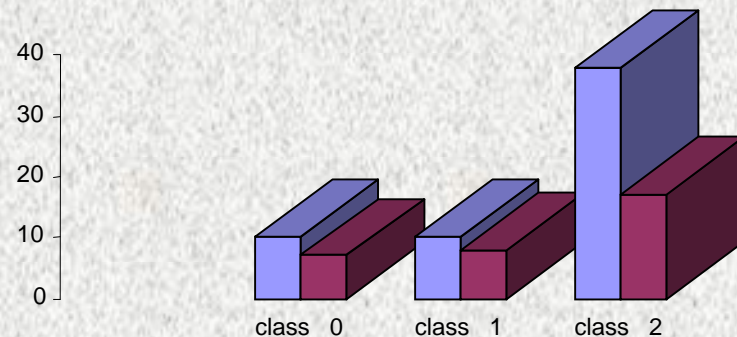
Reclassification

Classification type	Errors detected using Code reading	Percentage %	Percentage out of 38 seeded errors	Errors detected using Testing & debugging	Percentage %	Percentage out of 35 seeded errors
0	10	17 %	26 %	7	22 %	19 %
1	10	17 %	26 %	8	25 %	21 %
2	38	66 %	100 %	17	53 %	45 %
Total	58	100 %		32	100 %	

Class 0  
data

Class 1  
data

Class 2  
data



■ Faults detected using Code reading  
■ Faults detected using Testing & debugging



# ***Analysis and statistical approaches***

## Q hypothesis testing

Ü1 - Difference between code reading and, testing & debugging

Ø According to total faults detected

**No Significant Difference**

Ø According to right type 2 error detected

**No Significant Difference**

Ü2 – Relationships between faults detected and time effectiveness for both techniques  
(According to right type 2 error detected)

Ø Code Reading Technique

**Strong Significant Relation**

Ø Testing & Debugging Technique

**Weak Significant Relation**

## Q Conclusion

Ü Experiment results effected directly by Experiment subjects experience

Ü More experiments should be organized

# Conclusions & Future Plans

