



Decision Support in Software Project Management

Stefan Wagner

University of Stuttgart, Germany

9th International Advanced School of Empirical Software Engineering

Banff, September 2011



Decision making



**Decision
support**

Cross-functional tasks

Project management

Quality assurance

Configuration management

Requirements

Implementation

Maintenance

Design

Test

Project phases

Practitioners

Intra-company

Only what is necessary

Existing metrics and methods

Aiming at application

Academics (Consultants)

Inter-company

In-depth

Exploratory

Aiming at publication (next contract)

Outline

Empirical studies for decision support
by example

1) Academic studies

2) Hands-on: Design your own study

3) Practical studies



1) Academic studies

Methods

Analyse field data

Case studies

Experiments

Surveys, interviews

Literature reviews

Example: Code cloning

Decision:

Do I need to avoid and therefore control code cloning?

Questions:

Does cloning increase maintenance costs?

Does cloning lead to defects?

Code clone example

```
// Utilities for arrays of elements
public String showElements(ModelElement[] elements, String nomsg) {
    boolean found = false;
    StringBuffer res = new StringBuffer();
    if (elements != null) {
        Index.getInstance().setCurrentRenderer(
            FlatReferenceRenderer.getInstance());
        for (int i = 0; i < elements.length; i++) {
            ModelElement el = elements[i];
            res.append(showElementLink(el)).append(HTML.LINE_BREAK);
            found = true;
        }
        Index.getInstance().resetCurrentRenderer();
    }
    if (!found && nomsg != null && nomsg.length() > 0) {
        res.append(HTML.italics(nomsg));
    }
    return res.toString();
}
```

```
// Utilities for arrays of elements
public String showElements(ModelElement[] elements, String nomsg) {
    boolean found = false;
    StringBuffer res = new StringBuffer();
    if (elements != null) {
        Index.getInstance().setCurrentRenderer(
            FlatReferenceRenderer.getInstance());
        for (int i = 0; i < elements.length; i++) {
            ModelElement el = elements[i];
            res.append(showElementLink(el)).append(HTML.LINE_BREAK);
            found = true;
        }
        Index.getInstance().resetCurrentRenderer();
    }
    if (!found && nomsg.length() > 0) {
        res.append(HTML.italics(nomsg));
    }
    return res.toString();
}
```


Inconsistent clones

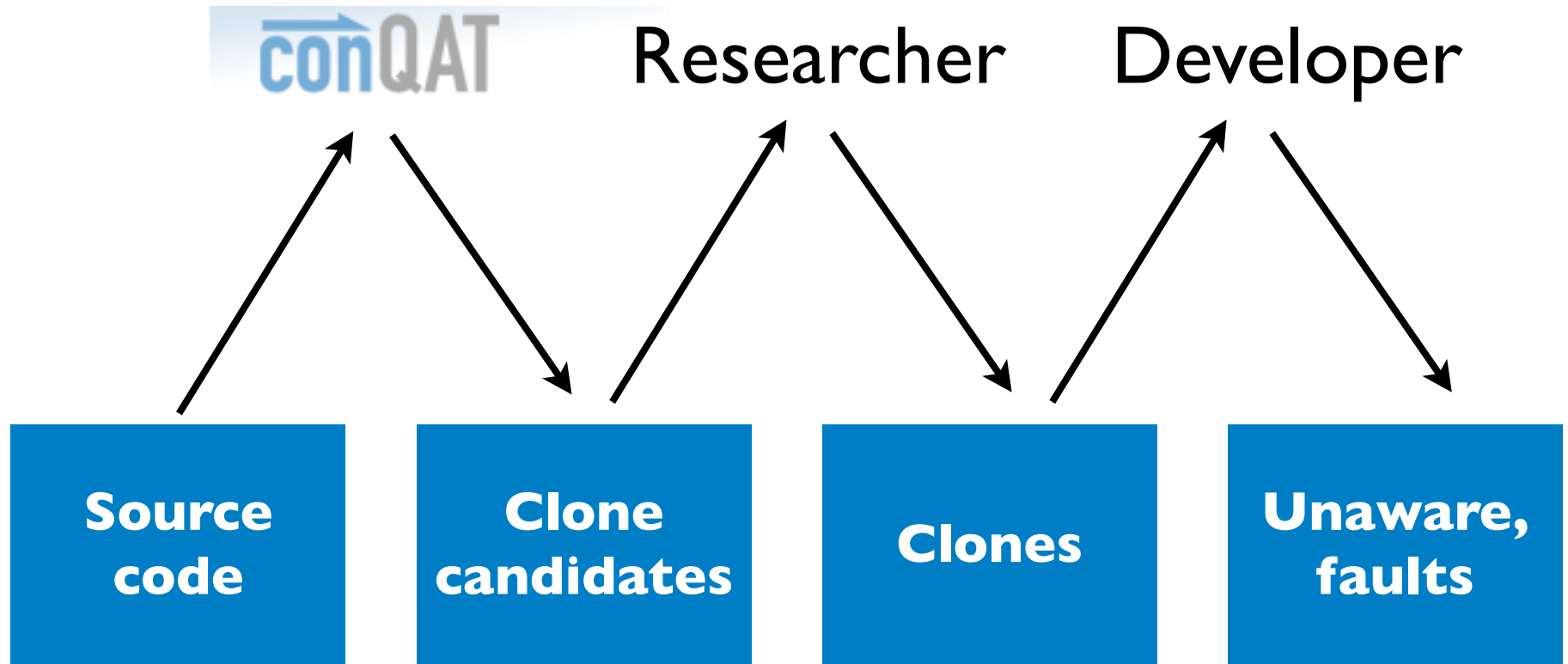
```
// Utilities for arrays of elements
public String showElements(ModelElement[] elements, String nomsg) {
    boolean found = false;
    StringBuffer res = new StringBuffer();
    if (elements != null) {
        Index.getInstance().setCurrentRenderer(
            FlatReferenceRenderer.getInstance());
        for (int i = 0; i < elements.length; i++) {
            ModelElement el = elements[i];
            res.append(showElementLink(el)).append(HTML.LINE_BREAK);
            found = true;
        }
        Index.getInstance().resetCurrentRenderer();
    }
    if (found && nomsg != null && nomsg.length() > 0) {
        res.append(HTML.italics(nomsg));
    }
    return res.toString();
}
```

```
// Utilities for arrays of elements
public String showElements(ModelElement[] elements, String nomsg) {
    boolean found = false;
    StringBuffer res = new StringBuffer();
    if (elements != null) {
        Index.getInstance().setCurrentRenderer(
            FlatReferenceRenderer.getInstance());
        for (int i = 0; i < elements.length; i++) {
            ModelElement el = elements[i];
            res.append(showElementLink(el)).append(HTML.LINE_BREAK);
            found = true;
        }
        Index.getInstance().resetCurrentRenderer();
    }
    if (!found && nomsg.length() > 0) {
        res.append(HTML.italics(nomsg));
    }
    return res.toString();
}
```

Study objects



Procedure



Example: Requirements cloning

Decision:

Do I also need to avoid and therefore control code cloning?

Questions:

**Does cloning exist in specifications?
Is it harmful there?**

"Redundancy [in requirements specifications] causes good engineers to suffer and the resulting systems will probably suffer, too."

–Matthias Weber,
Joachim Weisbrod



Modifiability generally requires a requirements specification to [...] not be redundant.

—IEEE 830-1998



Terms

Requirements specification

“specification for a particular software product, program, or set of programs that performs certain functions in a specific environment.” [IEEE 830-1998]

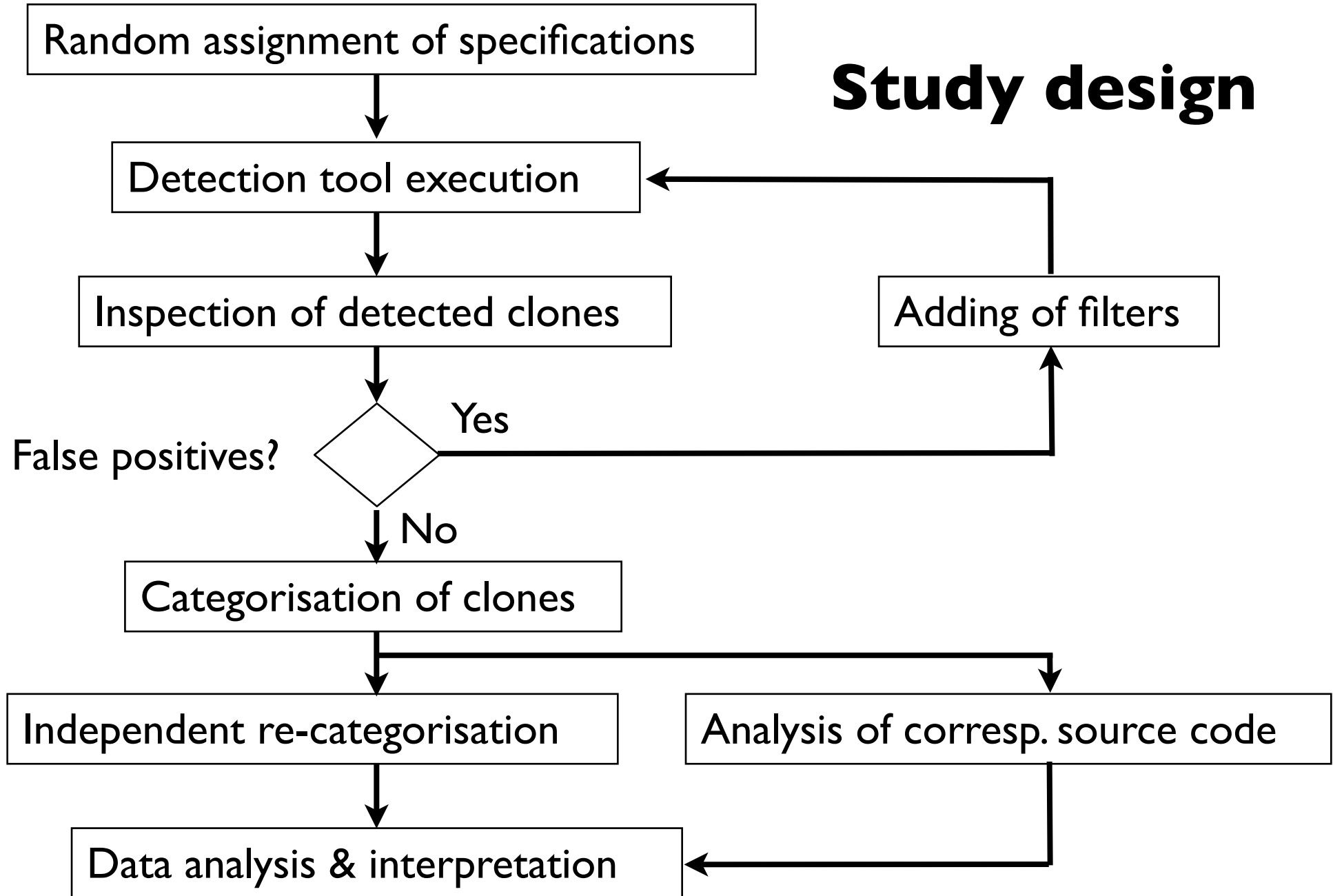
Clone

- Duplicated specification text of at least 20 words
- Small differences (e.g., declination) are tolerated
- Must refer to specified system
- False positives: e.g., page footers with copyright information

Research questions

1. How much cloning do real-world requirements specifications contain?
2. What kind of information is cloned in requirements specifications?
3. What consequences does cloning in requirements specifications have?
4. Can cloning in requirements specifications be detected accurately using existing clone detectors?

Study design



Random assignment of specifications

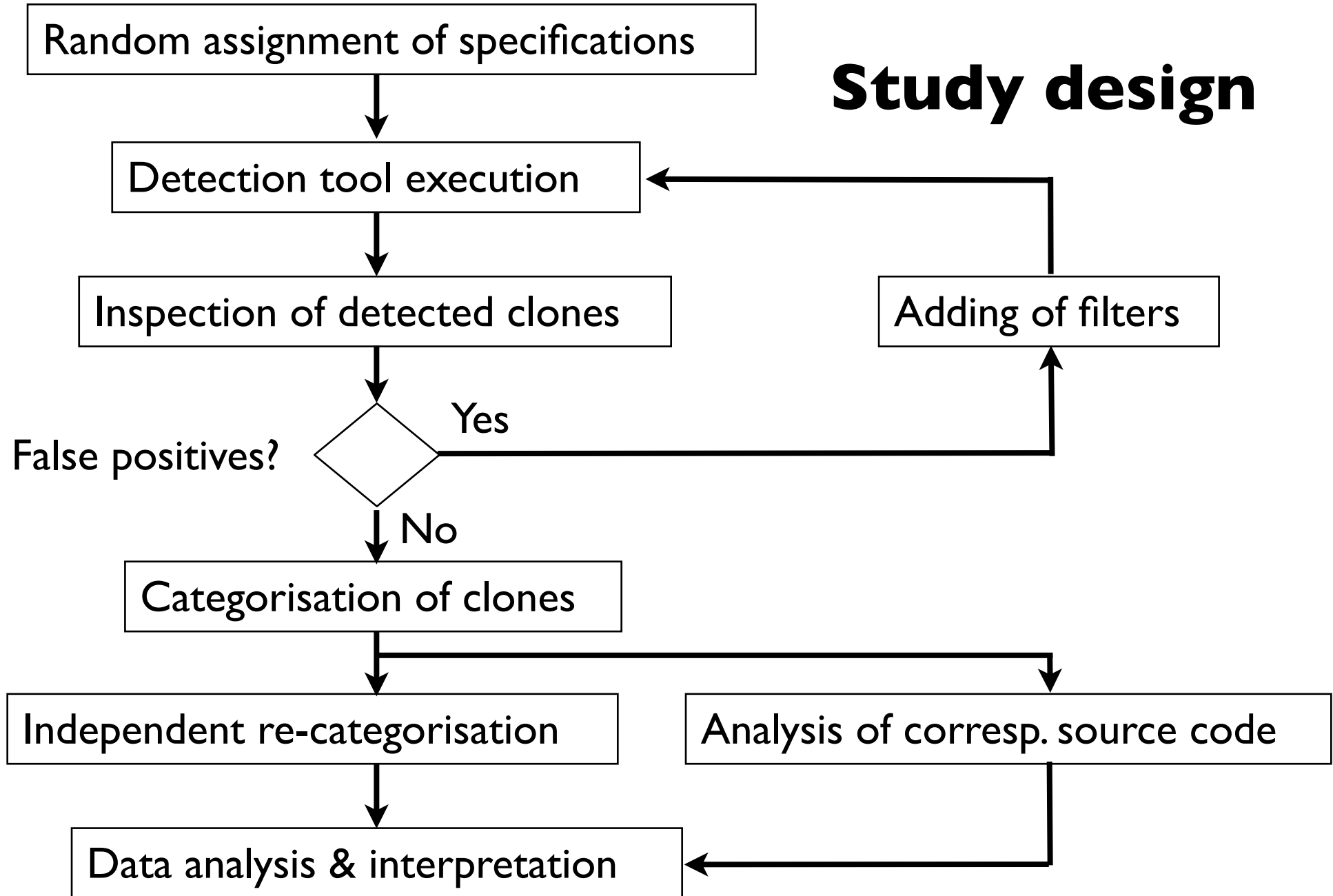
8 participants

4 pairs

1 room

1 day

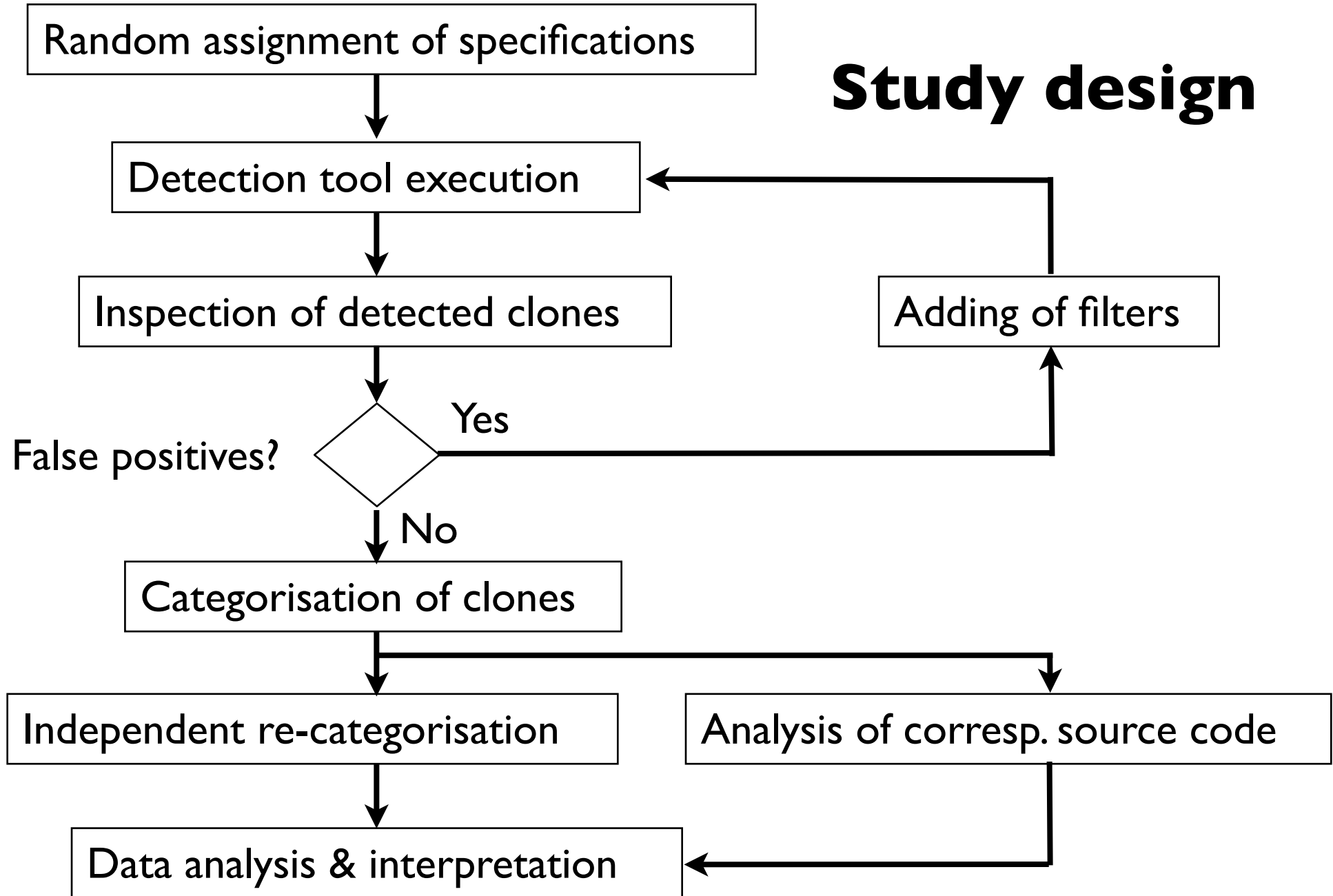
Study design



Detection tool execution

Time measurement
High number of clones

Study design



Inspection of detected clones

Clone Compare

MOSTSpecification_3V0.txt

the operation is related to the respective element in the record.
Even if the record does not contain an Array, the position consists of two bytes, but the set is not used in this case.

Data represents data according to the structure of the record and the specifications by position.

If a record contains an Array, the Array must be the last element of the record.

NElements Unsigned Byte Number of elements in Record

IntDescX are the interface descriptions of the single elements. Depending on the data type, interface descriptions defined for the respective class can be inserted. Please note that here elements, parameter Flags is not available. For parameter OPTypes only Set, Get, SetGet, SetIncrement, Decrement, and Error can be used.

Note: IntDesc only represents a group of parameters. No referencing of other functions and interface descriptions is done here!

Below, IntDesc is displayed with respect to the basic classes:

Class IntDesc
Switch Class, OPTypes, Name
Number Class, OPTypes, Name, Units, DataType, Exponent, Min, Max, Step
Text Class, OPTypes, Name, MaxSize
Enumeration Class, OPTypes, Name, Size, Name1, Name2, ...
BoolField Class, OPTypes, Name, DataType, NElements, BitName, BitSize, BitName, BitSize, ...

Specification Document © Copyright 1999 - 2008 MOST Cooperation

MOSTSpecification_3V0.txt

Signed Byte
Unsigned Word
Signed Word
Unsigned Long
Signed Long
String
Stream1
Classified Stream
Short Stream
NElements: Unsigned Byte Number of elements in function class Sequence Property

IntDescX are the interface descriptions of the single elements. Depending on the data type, interface descriptions defined for the respective class can be inserted. Please note that here elements, parameter Flags is not available. For parameter OPTypes only Set, Get, SetGet, SetIncrement, Decrement, and Error can be used.

1 If data type Stream is used, the corresponding parameter must be the last parameter.

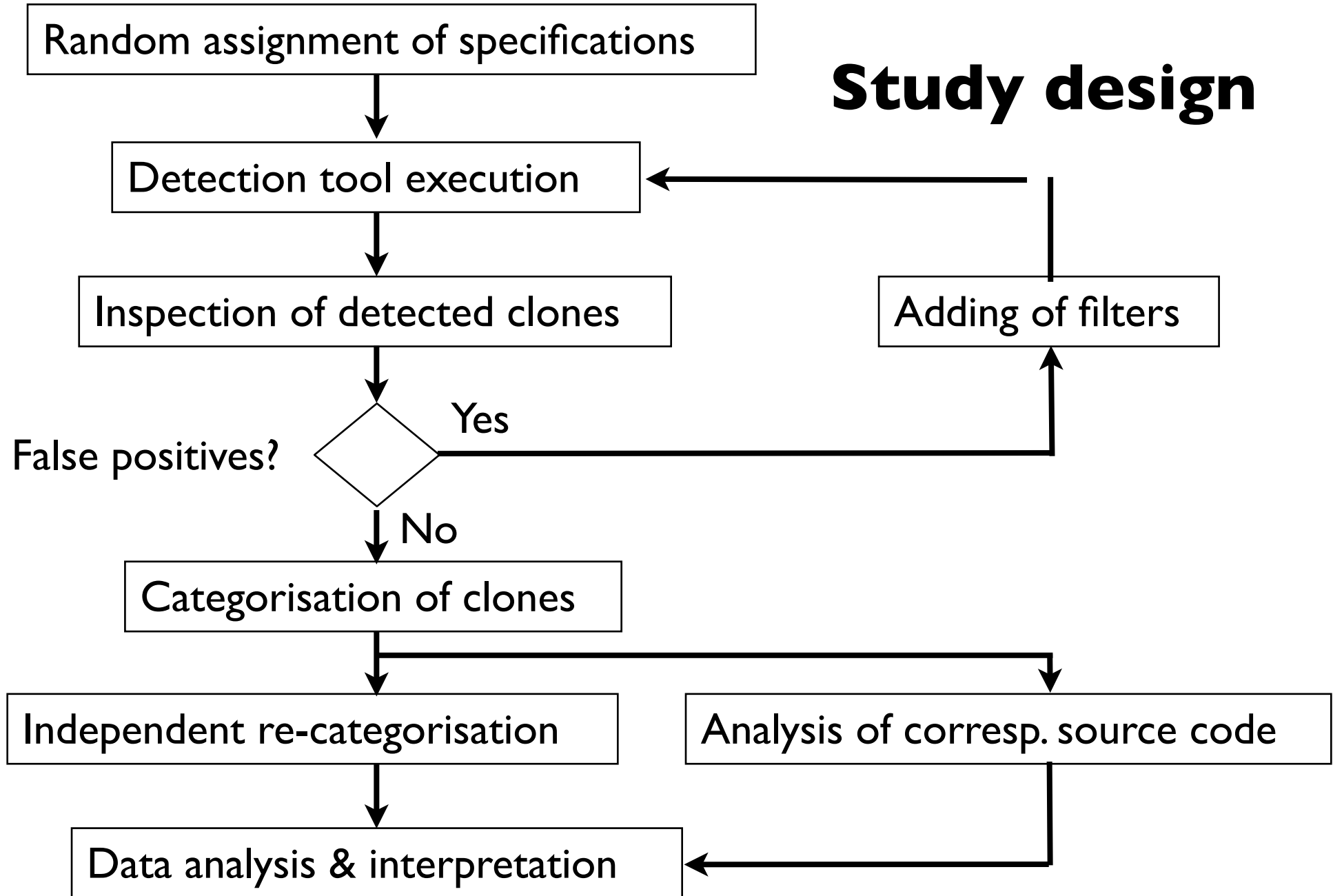
Specification Document © Copyright 1999 - 2008 MOST Cooperation

Page 99

MOST Specification
Rev 3.0 05/2008

MOSTMOST

Study design



Adding of filters

Specification Document

© Copyright 1999 - 2008 MOST Cooperation

Page 3

MOST Specification
Rev 3.0 05/2008

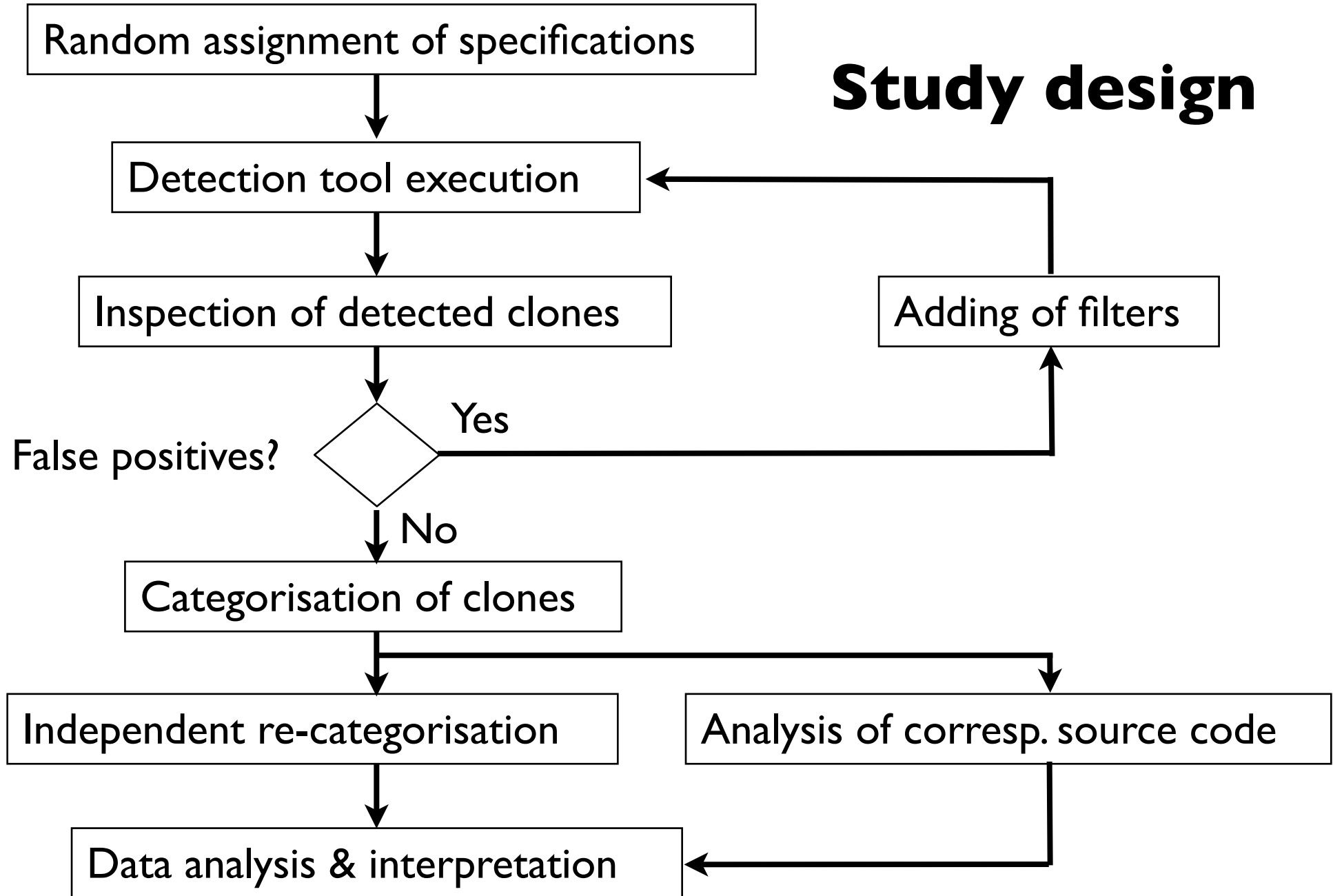
Regular expressions

Removal of clones

Improvement in precision

Categorisation of the types of false positives

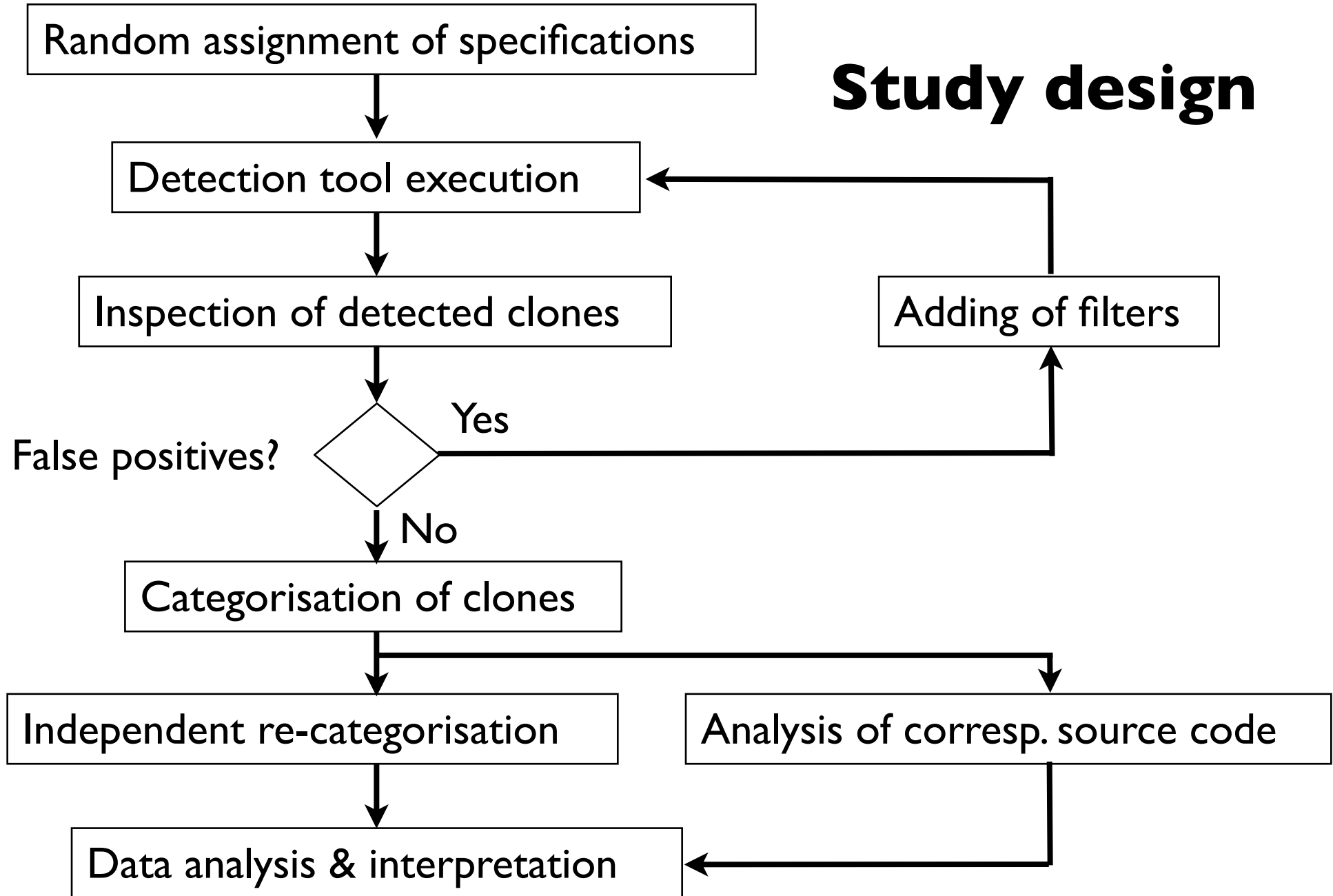
Study design



Categorisation of clones

- Qualitative analysis: content analysis
- Sample is categorised
- Mix of theory-based and Grounded Theory
- 4+8 categories
- Documentation of additional information (mostly inconsistencies between clones)

Study design



Independent re-categorisation

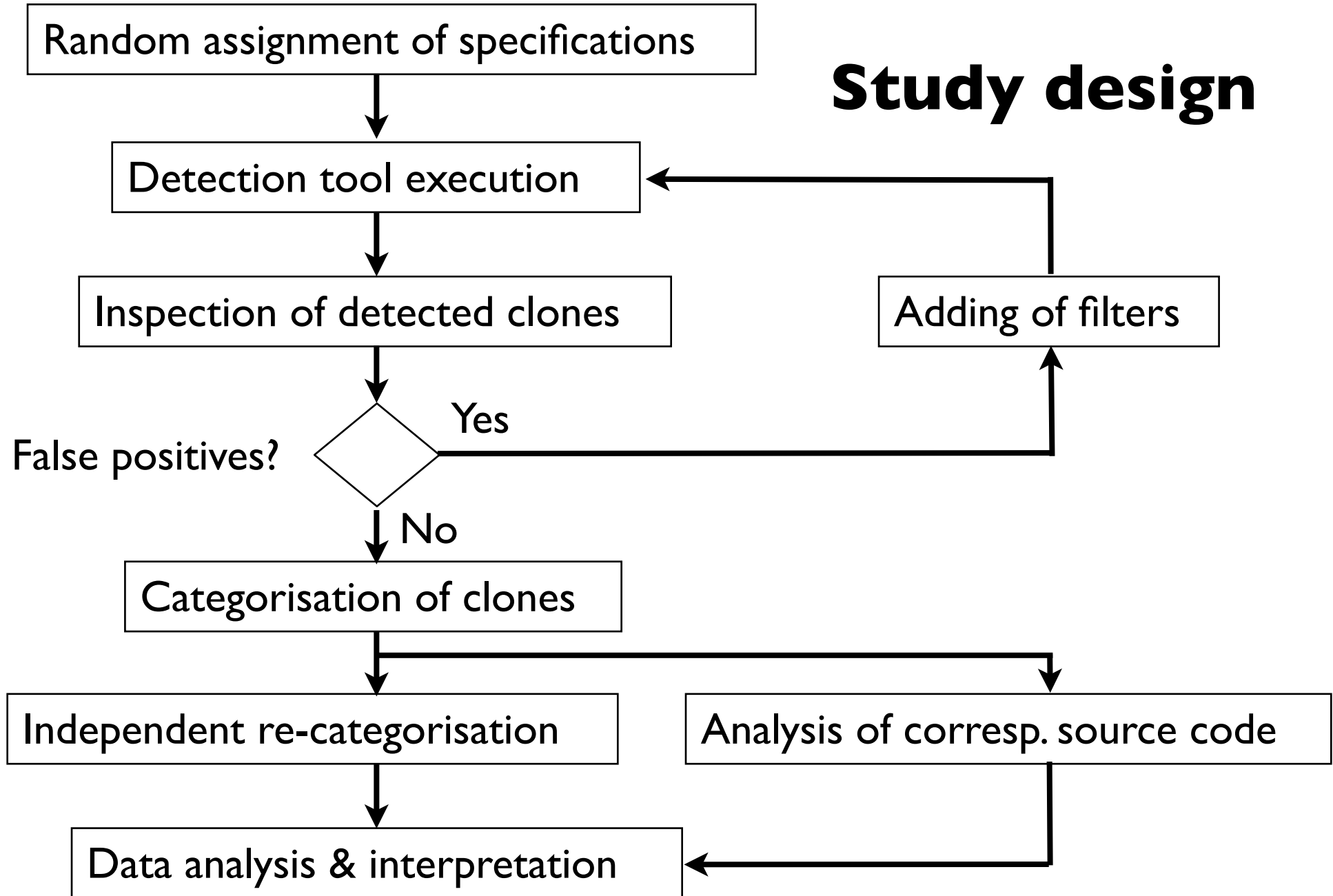
2 raters

Sample: 5 specifications

Sample: 5 clone groups

Analysis of inter rater agreement

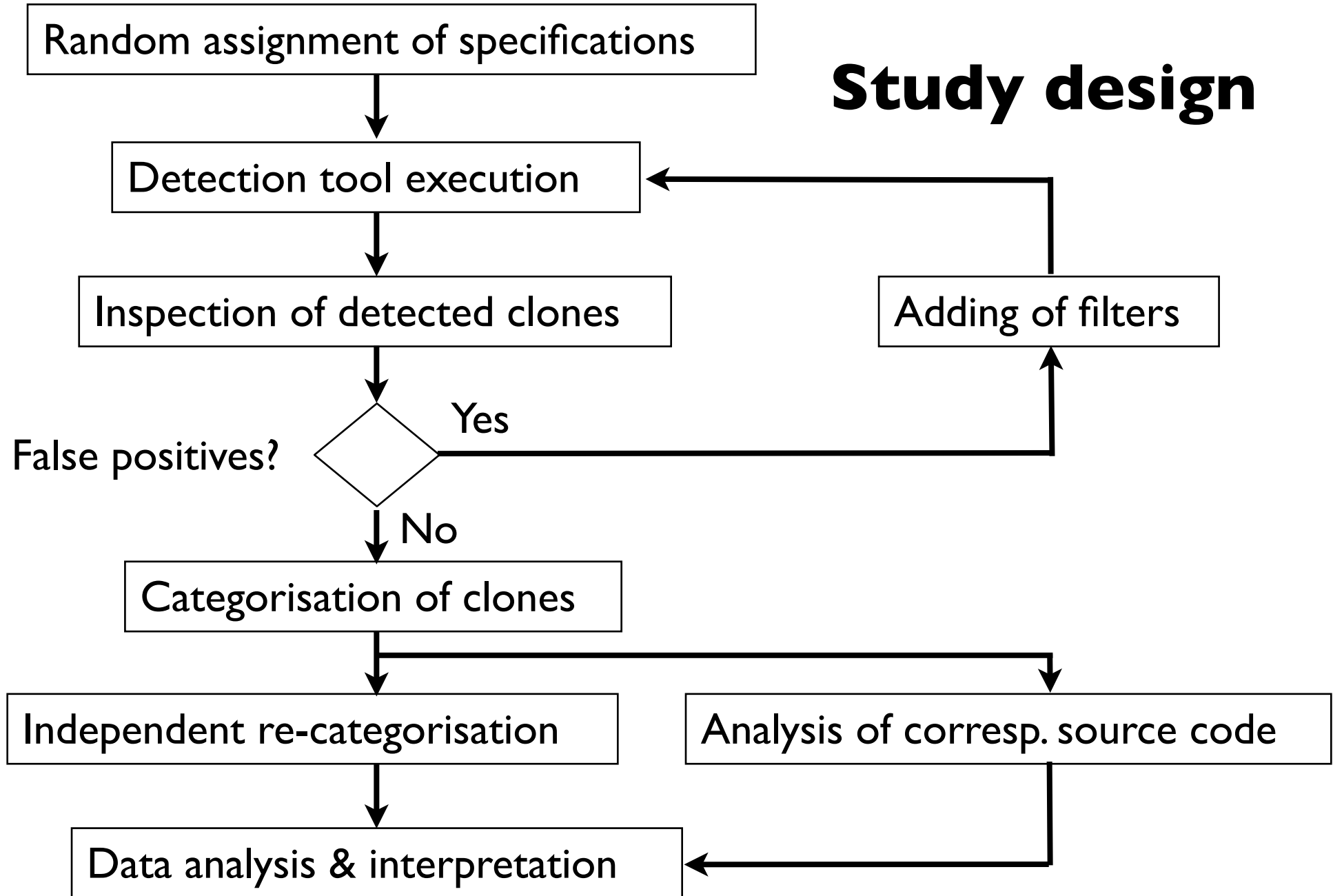
Study design



Analysis of corresp. source code

- Qualitative analysis: content analysis
- Explorative

Study design



Study objects

28 specifications
11 organisations
8,667 pages
over 1.2 Mio. words
English & German

Domains:
automotive
avionics
finance
telecommunication
transport

Typical Clones

- Entire use cases copied (create / edit XY)
- Similar combinations of pre and post conditions copied
- Descriptions of terms or roles copied

Example*

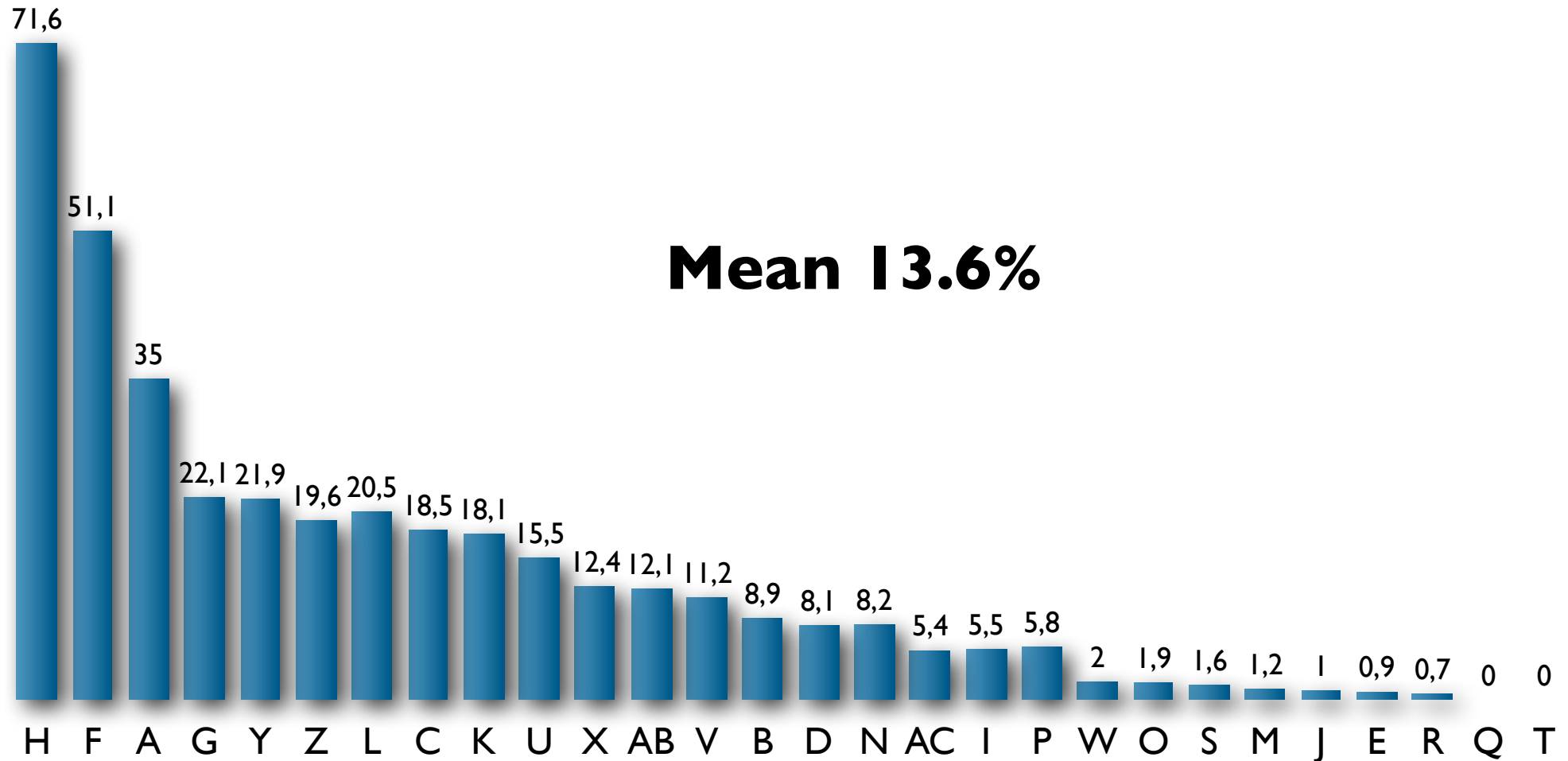
42 instances (61 words, 13 instances with > 100 words)

“The contracts with the clients describe the conditions regarding obligatory liabilities that the clients have agreed on with X. The liabilities are calculated from the exposures from Y and the contract conditions from X. The liability-relevant parts of the contracts thus need to be managed in system Z.”

...
regarding
n X. The
the
of the

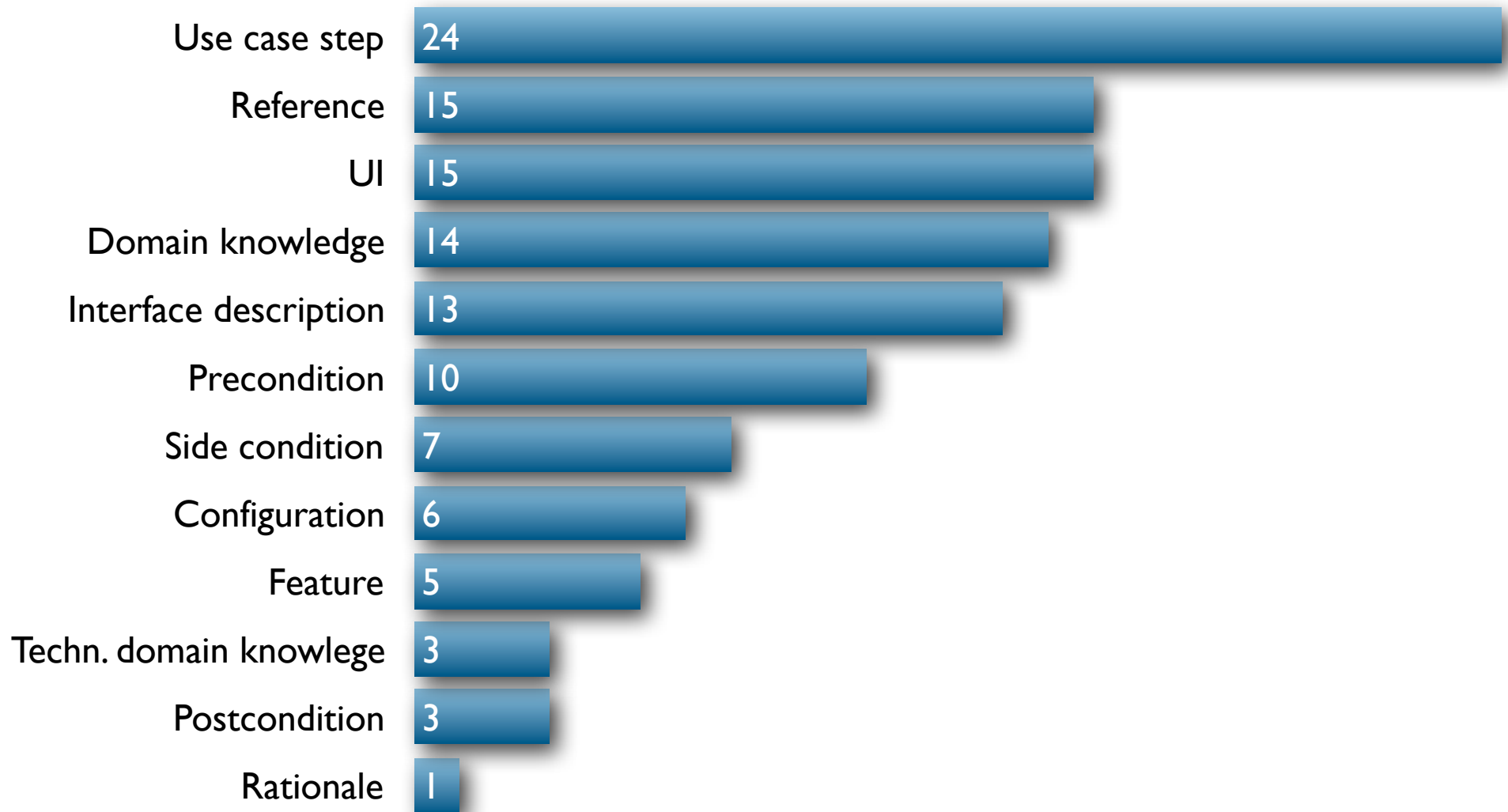
I. How much cloning do real-world requirements specifications contain?

Clone coverage in percentage



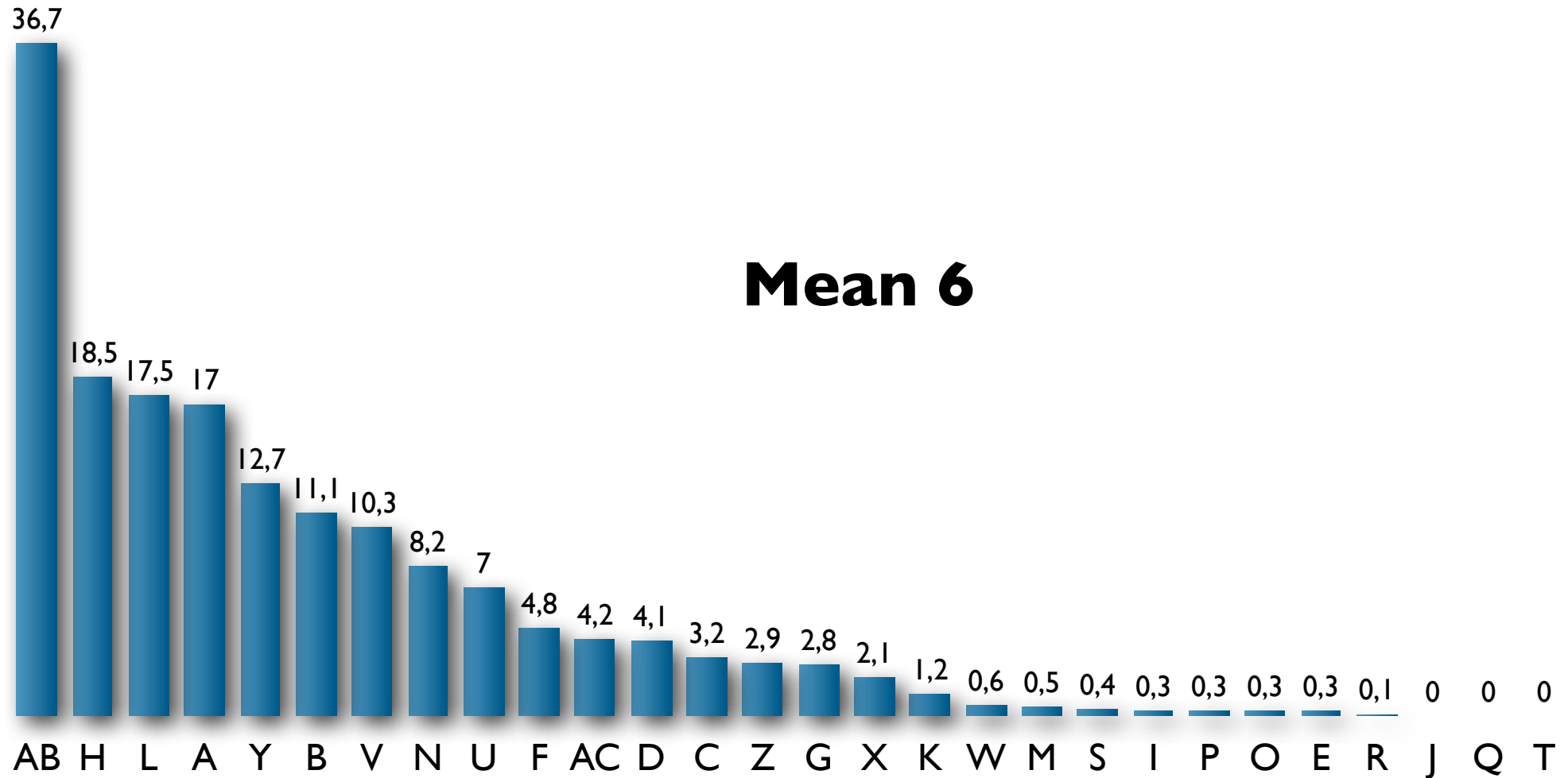
2. What kind of information is cloned?

Percentage of clones, more than one category possible



3. What consequences does cloning have?

Additional effort in hours per inspector



Modification

- Multiple inconsistent specification clones identified
- Differences suspected to be unintentional

⇒ Indication that inconsistent updates happen in practice

Implementation

Traced specification clone groups to implementation. 3 cases:

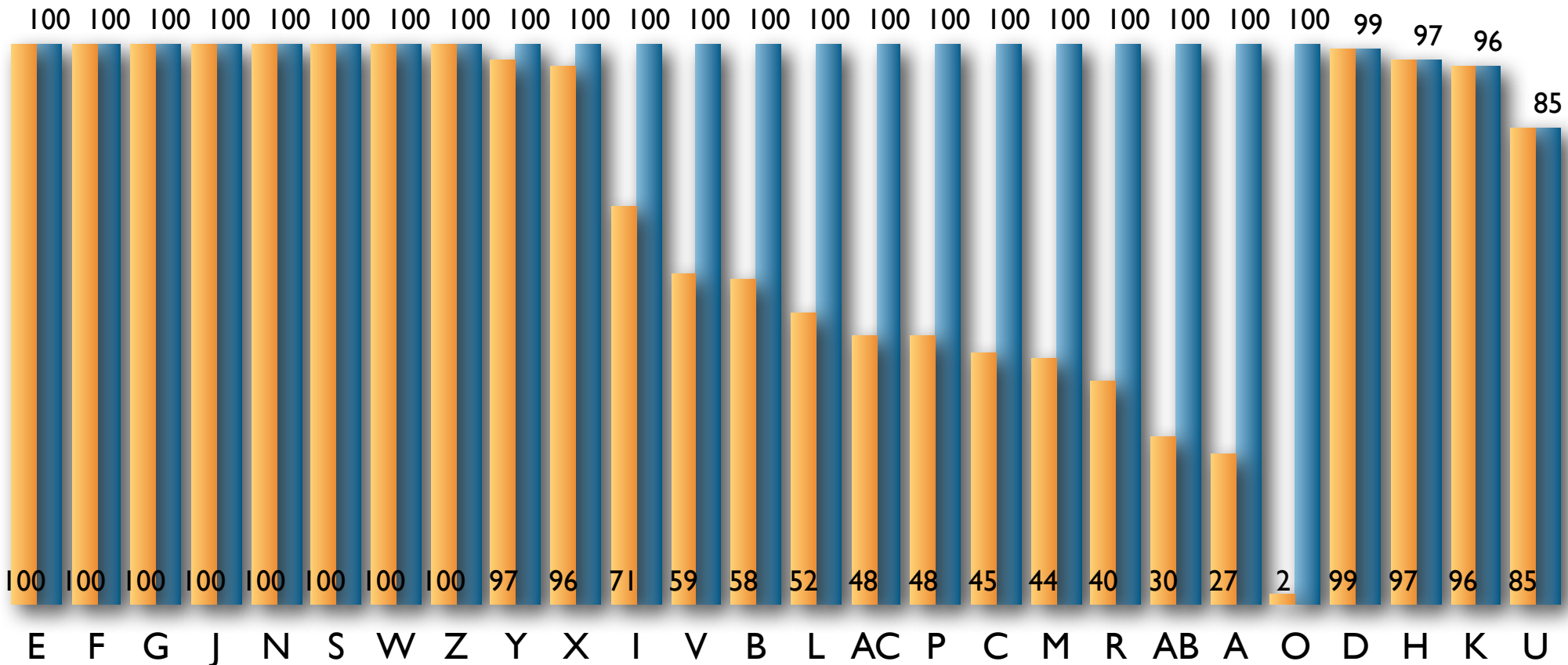
- Shared abstraction
- Cloned code
- Independent reimplementations of similar functionality

⇒ Indication that spec. cloning causes redundancy in implementation

4. Can cloning be detected accurately using existing clone detectors?

Precision in percentage

Before tailoring
After tailoring



Threats to validity

Internal

- Pairs of researchers to reduce errors during manual steps
- Reading speeds for cloned vs non-cloned text? Assumed similar. Further research required
- Recall unclear. But: does not affect study results

External

- Substantial differences between requirements specifications (format, organization, language, ...)

But: large amount of study objects from different companies, domains

Conclusion

Lessons Learned

- Many specs contain cloning
- Negative impact on reading and inspection effort
- Indication for corresponding redundancy in source code
- Cloning not necessary – many specs contain none
- Tailoring required but feasible: effort small w.r.t. inspection overhead

Future Work

- How can cloning be avoided or removed?
- What are the causes for cloning? Different than for code clones?
- Further studies on consequences for implementation



2) Hands on

Design your own study!

Form groups of 3–5 people

Find a relevant SE decision

Derive questions for a study

Design the study

Prepare a short summary to present


15 Minutes

Example decisions

How many people do I need in which role?

Do I need to improve the maintainability of my system?

Will my project be finished on time?



3) Practical studies

Methods

Usually only collecting and analysing data
from projects

Sometimes surveys

Example: When to stop testing?

Decision:

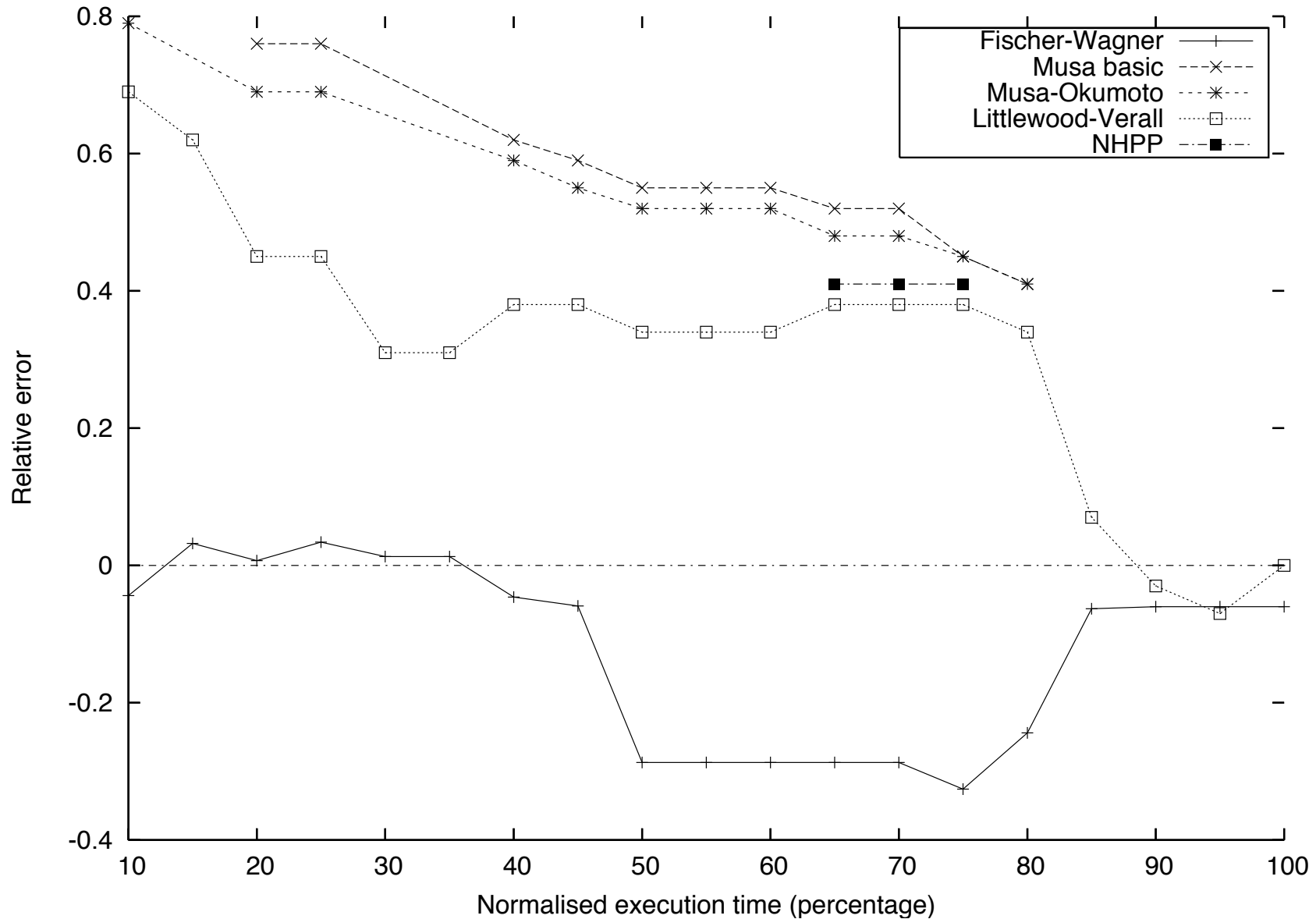
How long does it make sense to run a field test?

Questions:

What is the failure distribution?

Can we predict reliability?

Reliability growth model



Example: Inspection efficiency

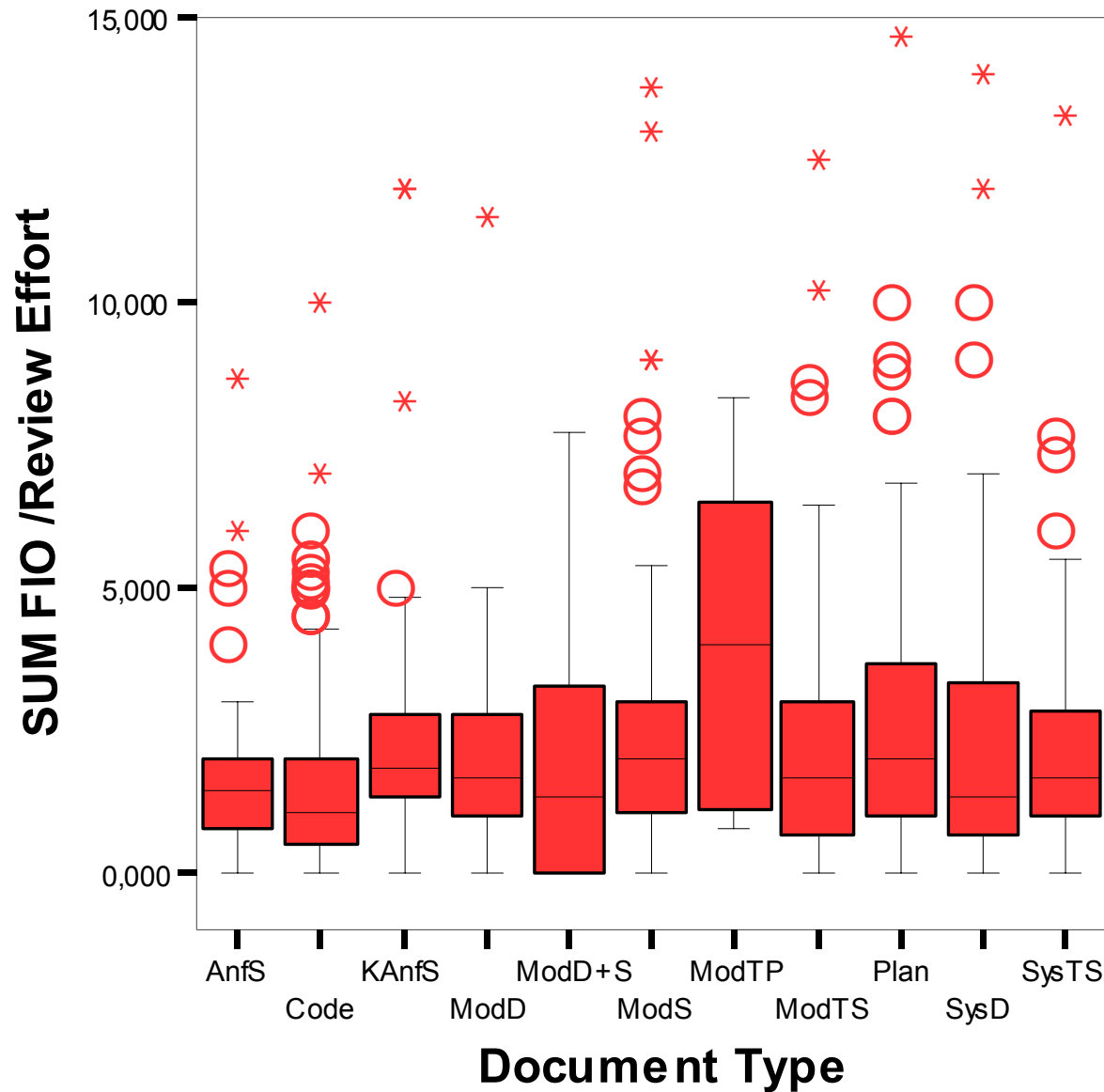
Decision:

Should I use inspections and if yes, for which documents?

Question:

How efficient are inspections on different documents?

Inspection data analysis



Example: Maintenance types

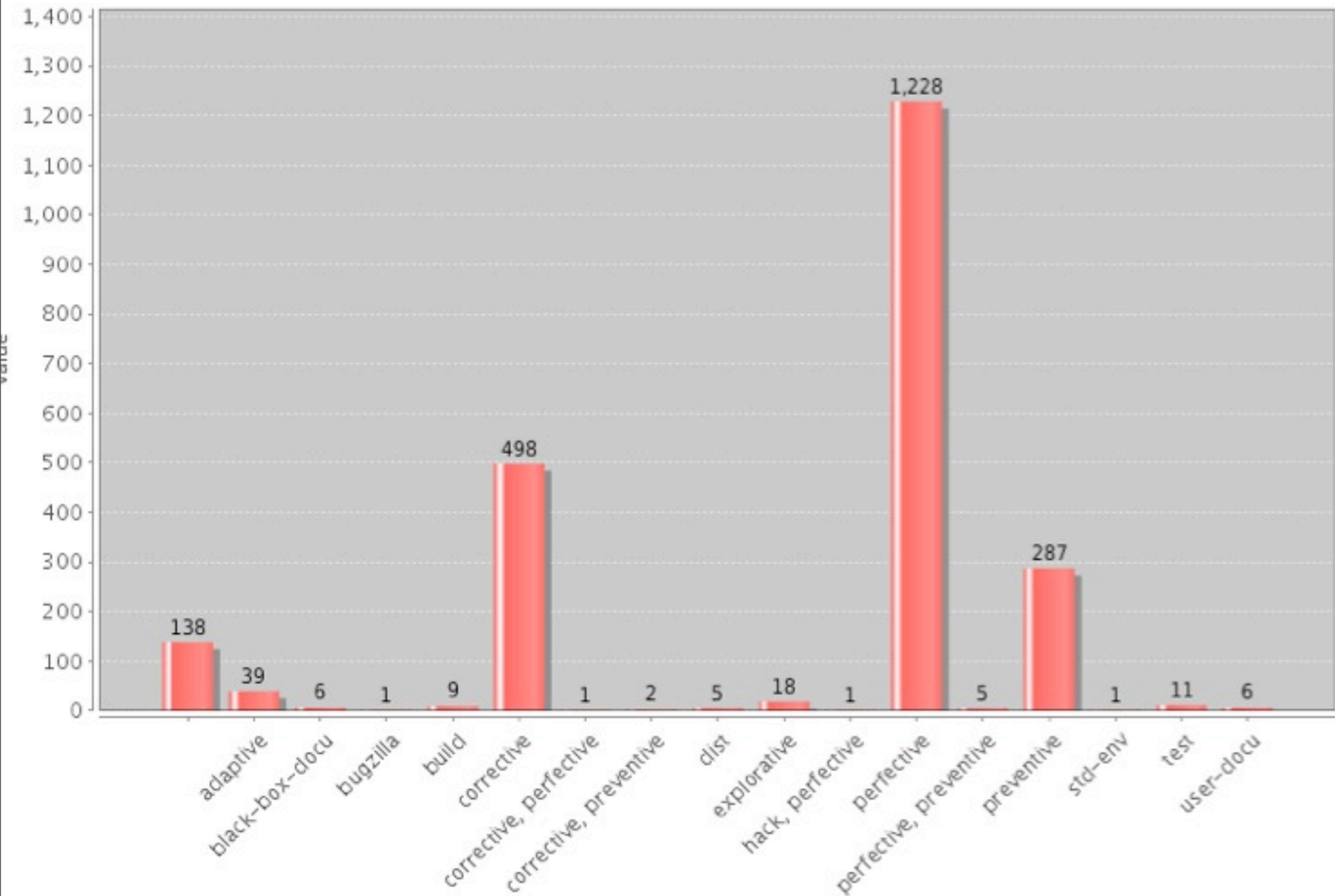
Decision:

Is my maintenance budget well spent?

Question:

What kind of CRs have we had?

CR Type



Outline

Empirical studies for decision support
by example

1) Academic studies

2) Hands-on: Design your own study

3) Practical studies

A good decision
is based on knowledge
and not on numbers.



Plato

**In my house I'm the boss,
my wife is just
the decision maker.**

Woody Allen

