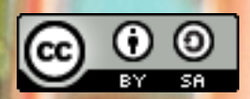
2. experimentation



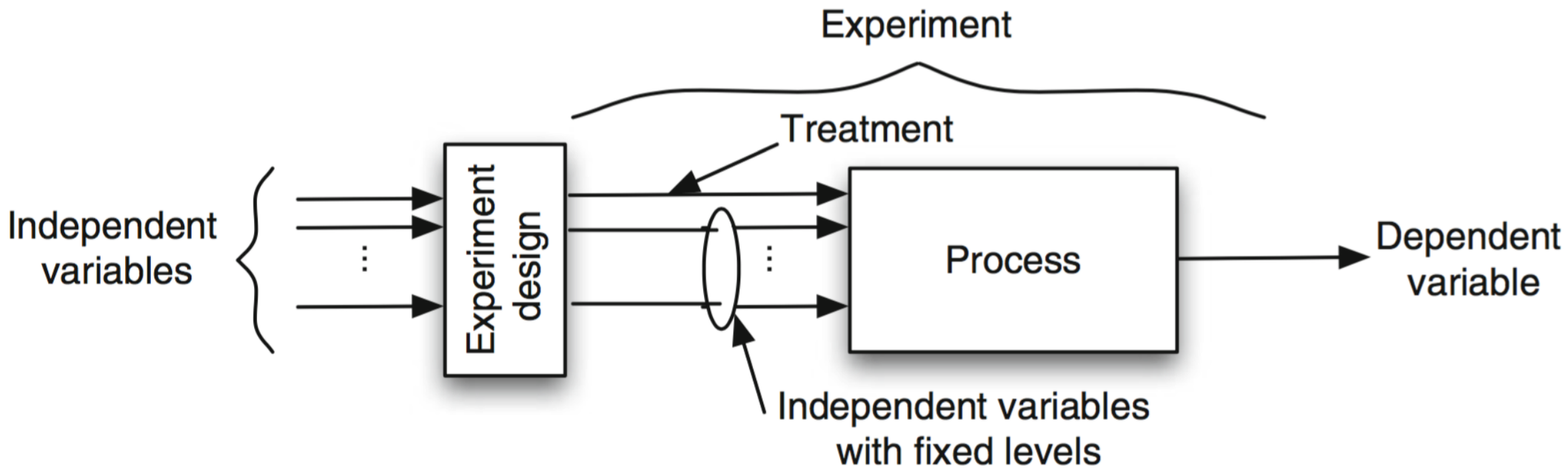
disclaimer!

- attention to details, note-taking and complete intellectual honesty are necessary
- statistics
  - is valuable in testing an hypothesis, not in initiating a discovery
- logic and common sense have first to be satisfied

why is exp. important?

- substantiate claims
- strengthen of falsify hypothesis
- evaluate and improve/revise/reject models
- gain additional insights, stimulate creativity

vocabulary



Source: Experimentation in Software Engineering, Springer, 2012

example

Investigating the effect of using Python instead of Java on the productivity of the developers.

- **dependent var**: {productivity}
- **independent vars**: {programming language, experience, GUI used, environment}
- **factor**: {programming language}
- **treatments**: {Python, Java}

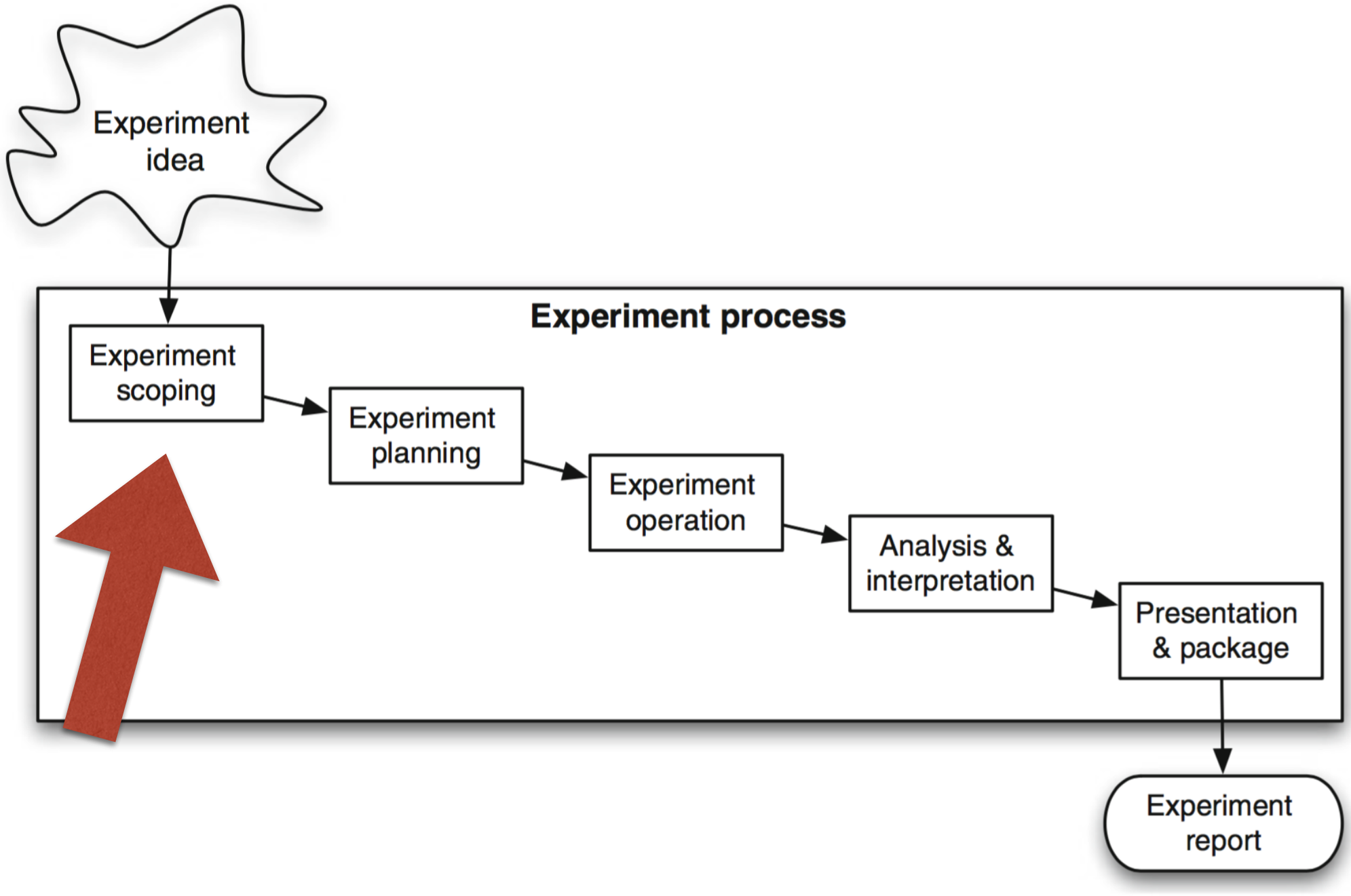
example

Investigating the effect of using Python instead of Java on the productivity of the developers.

- **objects**: {software to be developed}
- **subjects**: {developers}
- **tests**: developer_d uses treatment_t for developing object_o

typical CS scenario

- a particular task needs to be solved by a software
- it's already solved by a pre-existing one (baseline)
- you propose a new, in your opinion, better (S)
- you argue why your S is better than baseline
- you support your arguments by providing evidence



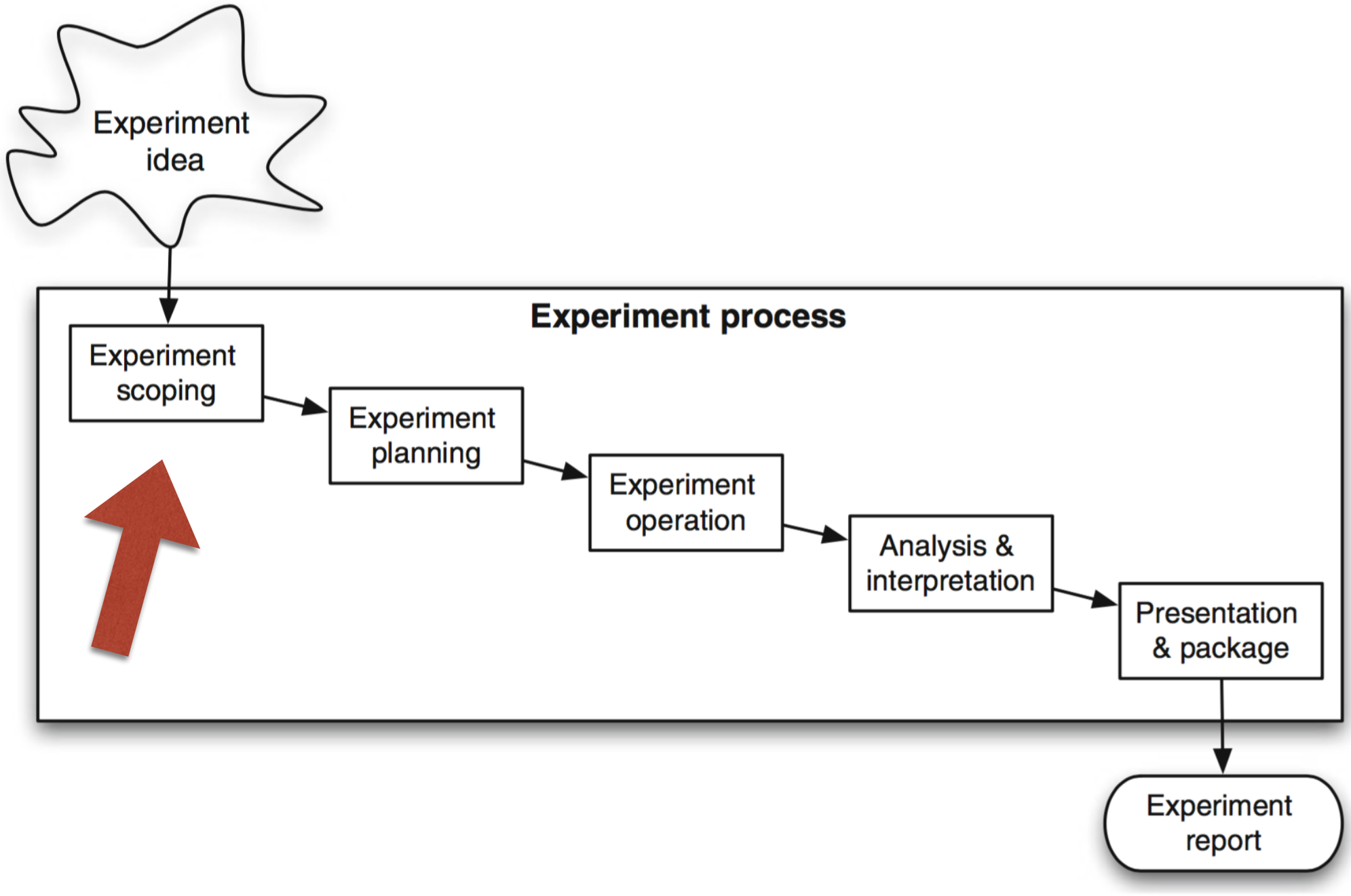
Source: Experimentation in Software Engineering, Springer, 2012

statement

I want to analyse <OBJECT> for the purpose of <PURPOSE> with respect to their <QUALITY FOCUS> from the point of view of the <PERSPECTIVE> in the context of <CONTEXT>.

Object	Purpose	Quality focus	Perspective	Context
Product	Characterize	Effectiveness	Developer	Subjects
Process	Monitor	Cost	Modifier	Objects
Model	Evaluate	Reliability	Maintainer	
Metric	Predict	Maintainability	Project manager	
Theory	Control	Portability	Corporate manager	
	Change		Customer	
			User	
			Researcher	

Example from software engineering



Source: Experimentation in Software Engineering, Springer, 2012

context selection

	risk	off-line	on-line
	cost	students	professionals
	data	toy	real
	scope	specific	general

hypothesis formulation

- null (H_0)
  - there are no real underlying trends in the experiment; the only reasons for differences in our observation are coincidental.

- alternative (H_1)
  - the hypothesis in favour of which H_0 is rejected.

an example

- null (H_0)
  - a new algorithm makes on average the same number of mistakes as the old one.
- alternative (H_1)
  - a new algorithm **does not** make on average the same number of mistakes as the old one.

test risks

WHAT WE ACTUALLY DO	REALITY: H_0 TRUE	REALITY: H_0 FALSE
reject H_0	Type I error (α) (false positive)	-
do not reject H_0	-	Type II error (β) (false negative)

power of a statistical test

- probability that the test will reveal a true pattern if H_0 is false.

- we want to be able to reject an erroneous hypoth.

- power = P(reject H_0 I H_0 false) = 1 - P(Type II error)
- which statistical test do we choose?
  - the one with the highest power

variable selection

- independent
  - variables we can control and have some effects on the dependent ones.
- dependent
  - variables on which we measure the effect of the treatment
  - typically it’s just one

examples

- independent
  - room temperature
  - genre of music
  - level of noise
  - quantity of oxygen
- dependent
  - focus

- independent
  - caffeine
  - time on Facebook
  - # of news read
  - # papers read
- dependent
  - PhD productivity

population sampling

- probability sampling
  - simple random
  - systematic
  - stratified random

- non-probability sampling
  - convenience
  - quota

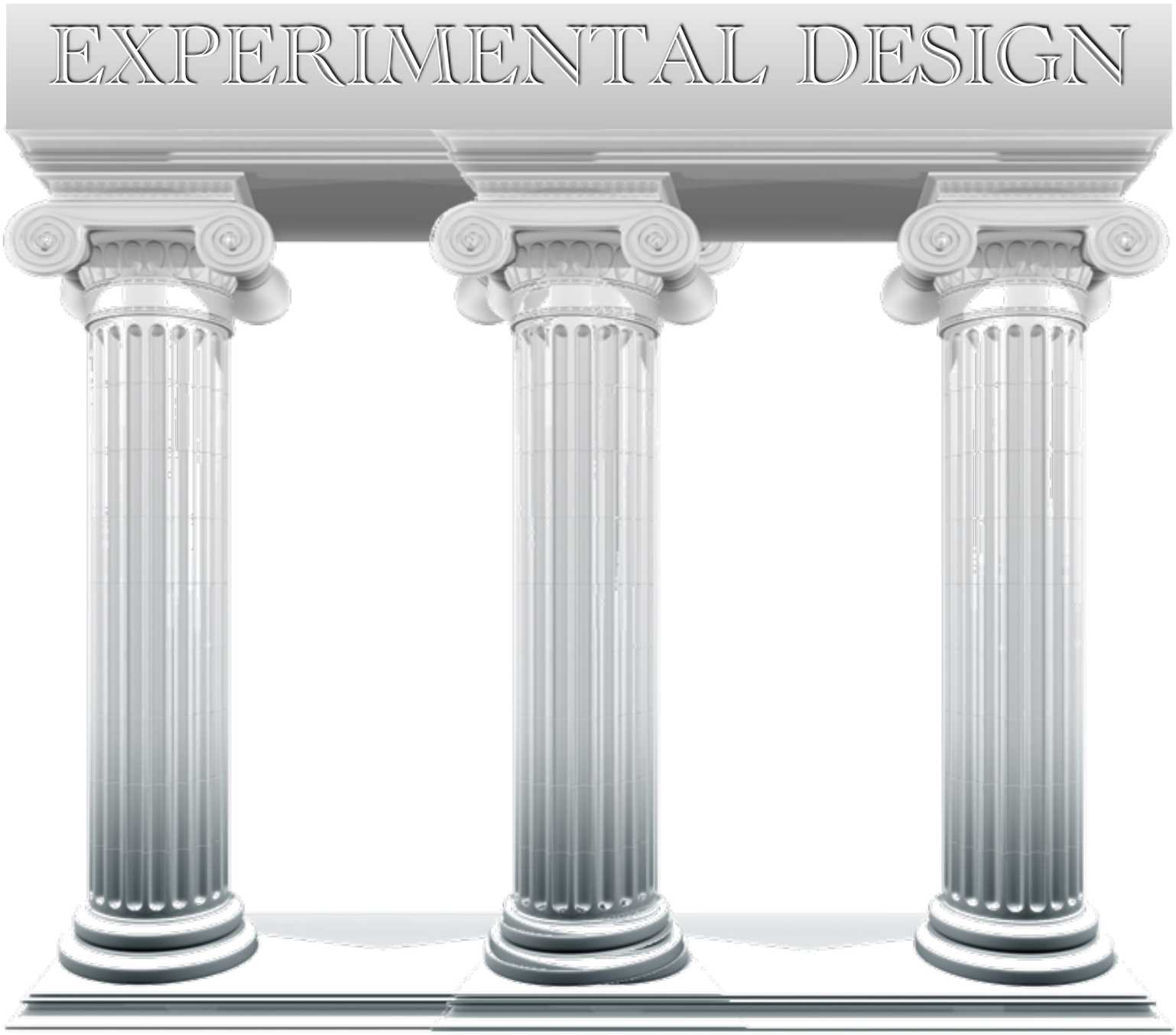
Remember: Make sure you save the seed numbers!

known facts about sampling

- large variability in the population
  - needs larger sample size
- the larger the sample size
  - the lower the generalisation error
- sample size and power of the stat. test are related
- the analysis of data may influence the sample size

Michele Filannino

EXPERIMENTAL DESIGN



MANCHESTER 1824
The University of Manchester

design principles

- **randomisation**: for subjects, objects, order of tests & everywhere else
  - averages out the effect of a possible factor
- **blocking**: eliminate undesired effects
  - increases the precision of the experiment
- **balancing**: equal # of subjects per treatment
  - simplifies and strengthens the statistical analysis

standard design types

- one factor with two treatments
- one factor with more than two treatments
- two factors with two treatments
- more than two factors each with two treatments

1 factor with 2 treatments

completely randomised

subjects	treat. 1	treat. 2
1	x	
2		x
3		x
4	x	
5	x	
6		x

- H_0: $\mu_1 = \mu_2$
- H_1: $\mu_1 \neq \mu_2$, $\mu_1 < \mu_2$ or $\mu_1 > \mu_2$
- Tests: t-test, Mann-Whitney

paired comparison

subjects	treat. 1	treat. 2
1	2	1
2	1	2
3	1	2
4	2	1
5	2	1
6	1	2

- H_0: $\mu_d = 0$ (where $d_j = y_{1j} - y_{2j}$)
- H_1: $\mu_d \neq 0$, $\mu_d < 0$ or $\mu_d > 0$
- Tests: paired t-test, Signed-test, Wilcoxon

Example: Investigate the F1-score when using Naïve Bayes and Random Forest.

1 factor with 2+ treatments

completely randomised

subjects	treat. 1	treat. 2	treat. 3
1		x	
2			x
3	x		
4	x		
5			x
6		x	

- H_0: $\mu_1 = \mu_2 = \ldots = \mu_a$
- H_1: $\mu_i \neq \mu_j$ for at least 1 pair (i,j)
- Tests: ANOVA, Kruskal-Wallis

randomised complete block

subjects	treat. 1	treat. 2	treat. 3
1	1	3	2
2	3	1	2
3	2	3	1
4	2	1	3
5	3	2	1
6	1	2	3

- H_0: $\mu_1 = \mu_2 = \ldots = \mu_a$
- H_1: $\mu_i \neq \mu_j$ for at least 1 pair (i,j)
- Tests: ANOVA, Kruskal-Wallis

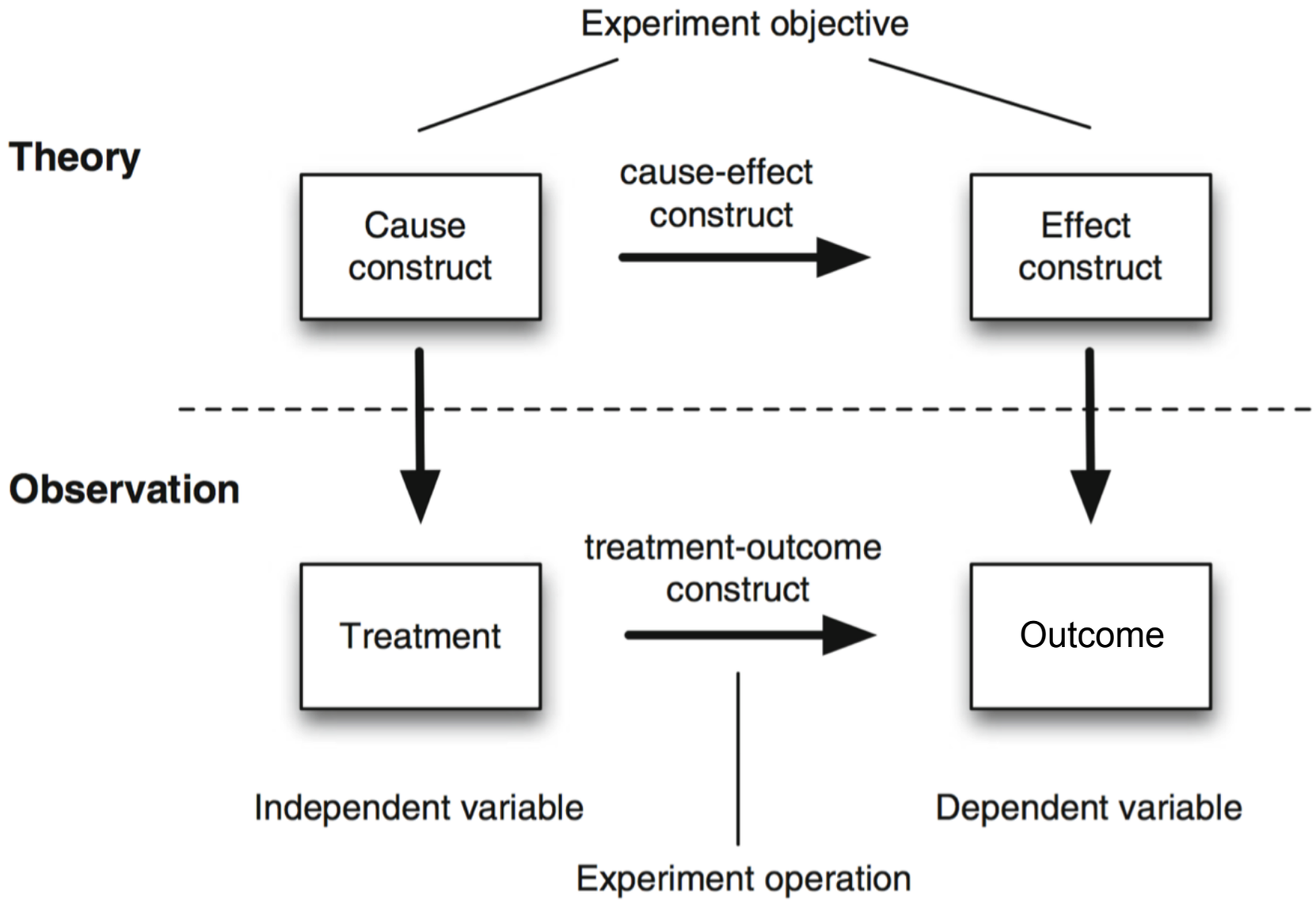
Example: Investigate the quality of the software when using C, C++ or Java.

other designs

- two factors
  - 2*2 factorial design
  - Two-stage nested design
- more than two factors
  - 2^k factorial design
  - 2^k fractional factorial design
  - 1/2 fractional factorial design of 2^k factorial design
  - 1/4 fractional factorial design of 2^k factorial design

Look at: Experimentation in Software Engineering, Springer, 2012, pages 95-101

how valid the results are?



Source: Experimentation in Software Engineering, Springer, 2012

conclusion validity

- relation between treatment and outcome
- is there a statistically significant relationship?
- threats
  - **low statistical power**, **violated assumption of statistical tests**, **fishing** and the error rate, reliability of treatment implementation, irrelevancies in experimental setting, random heterogeneity of subjects

internal validity

- does the treatment really cause the outcome?
- are there any other possible reasons for the outcome besides the reason I want it to be?

- threats
  - history, maturation, testing, instrumentation, statistical regression, selection, mortality, ambiguity about direction of causal influence, interactions with selection, diffusion of imitation of treatments, etc…

construct validity

- relation between theory and observation
- does the treatment reflect the construct of the cause?
- does the outcome reflect the construct of the effect?
- threats
  - **mono-operation bias**, **mono-method bias**, confounding constructs and levels of constructs, interaction of different treatments, interaction of testing and treatment, hypothesis guessing, **experimenter's expectancies**.

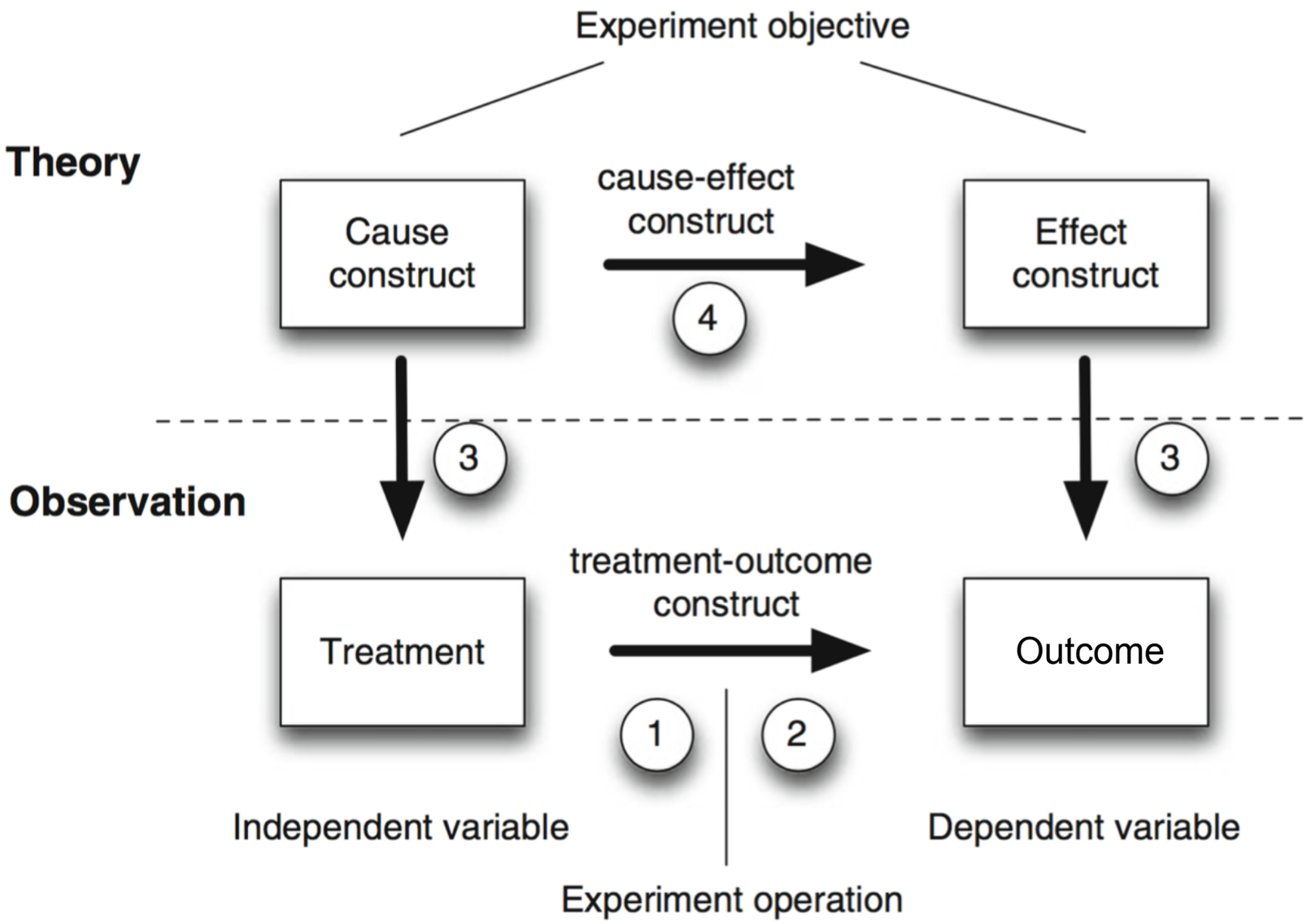
Example: Is Python better than Java?

external validity

- extent to which results can be generalised
- can the result of the study be generalised outside the scope of our study?
- threats
  - wrong participants as subjects
  - **wrong experimental environment**
  - **timing affecting the results**

Example 1: Conducting experiments on toy samples.
Example 2: Counting visits to the e-commerce portal during hot periods.

experiment principles



Source: Experimentation in Software Engineering, Springer, 2012

tensions among threats

- undergraduate students rather than professionals
  - larger study groups, reduce heterogeneity within groups, reliable treatment implementation
  - higher conclusion validity, lower external validity
- subjects measure factors and fill in forms
  - no bias from experimenters, tasks are more tedious for the subjects -> more errors
  - higher construct validity, lower conclusion validity