ICS 500: Research Methods and Experiment Design in Computing

Lecture

Experiment Process

The slides are prepared by: Dr. Mahmood Niazi, Dr. Malak Baslyman, Dr. Hamoud Aljamaan & Dr. Mohammad Alshayeb

Lecture Objectives

- ✓ Experiments Fundamentals
- ✓ Experiment process

Experiments Fundamentals

- Experimentation is not simple; we have to ***prepare***, ***conduct*** and ***analyze*** experiments properly.

- One of the main advantages of an experiment is the ***control*** of, for example, subjects (e.g., people), objects (e.g., tools) and instrumentation (e.g., questions).

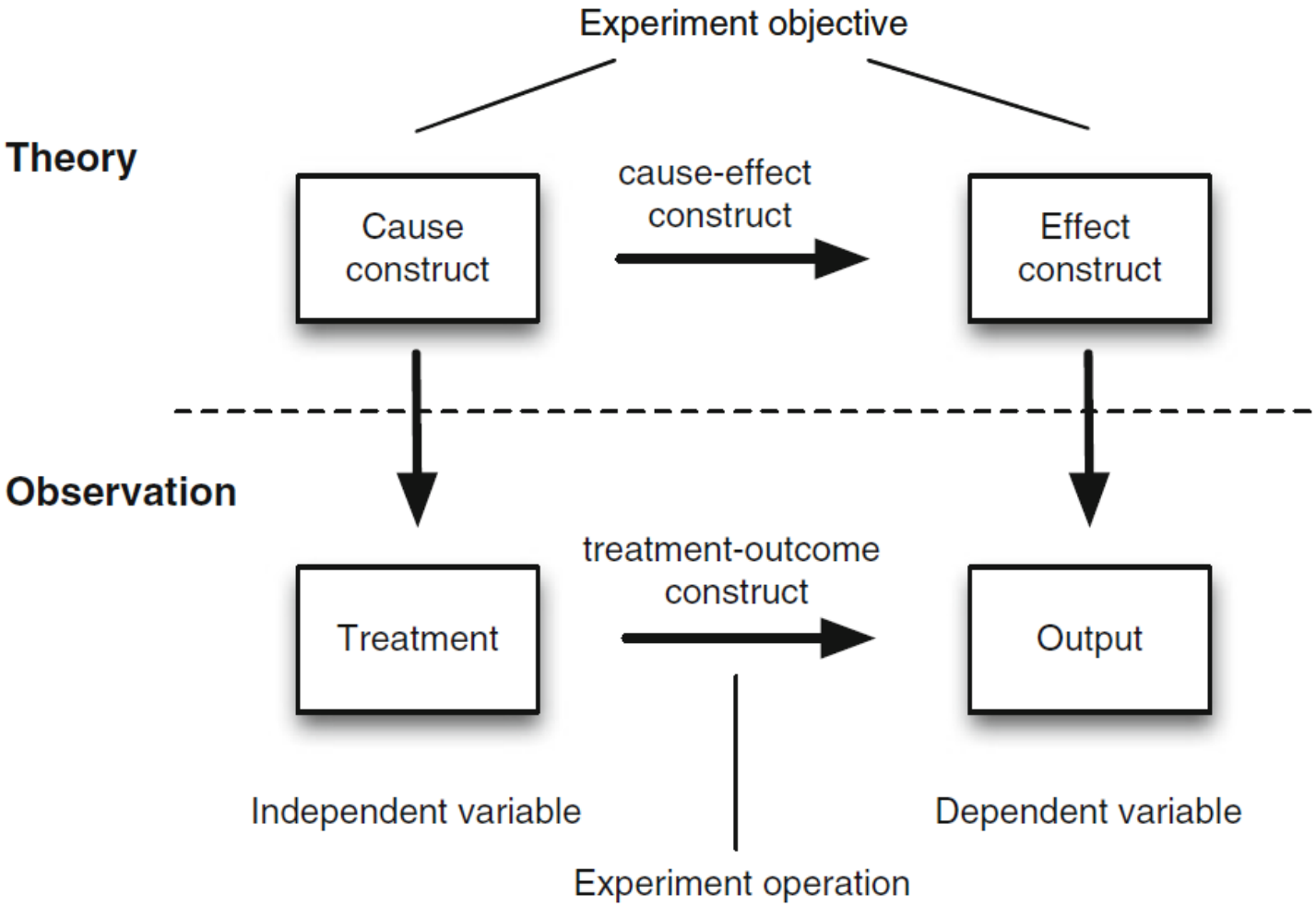
- Other advantages include ability to perform **statistical analysis** using **hypothesis testing** methods and opportunities for **replication**.

- To ensure that we make use of the advantages, we need a process supporting us in our objectives in doing experiments.

Experiment Principles

- The starting point is that we have an idea of a ***cause-and-effect*** relationship
  - We are able to formulate a hypothesis - i.e., Pair programming has effect on code quality.
- In order to test hypothesis, we may use an experiment.
- In the design of the experiment, we have a number of treatments (values that the studied variable can take, e.g., application of pair programming) over which we have control.
- The experiment is performed, and we are able to test the relationship between the **treatment** and the **outcome**.
  - If the experiment is properly set up, we should be able to draw conclusions about the relationship between the cause and the effect for which we stated a hypothesis.

Experiment Principles



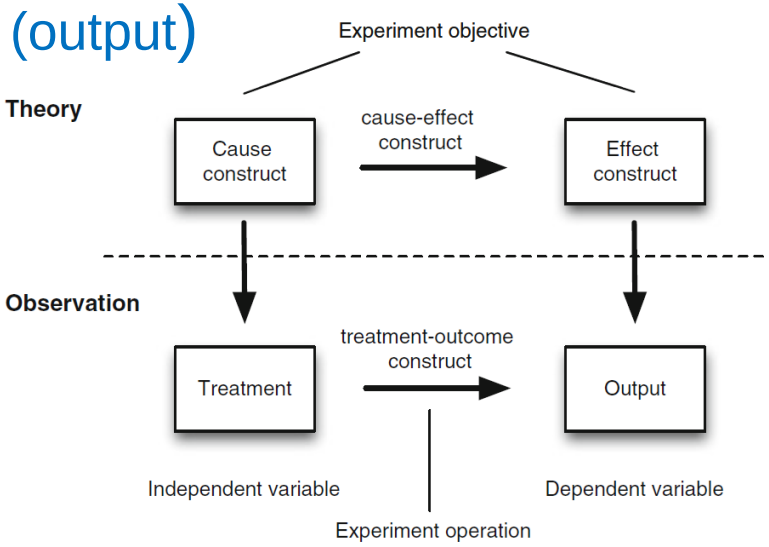
Experiment Principles - example

The starting point is that we have an idea of a cause and effect relationship, i.e. we believe that there is a relationship between a cause construct and an effect construct.

- Theory (hypothesis) = Pair programming (cause) has effect on code quality (effect).

In order to evaluate our beliefs, we may use an experiment.

- Observation = Apply pair programming (treatment) to observe the effect (good or bad) on code quality (output)



Variables/Treatments/Objects/Subjects

- When conducting a formal experiment, we want to study the outcome when we vary some of the input variables to a process.
- There are two kinds of variables in an experiment.
  - All variables in a process that are manipulated and controlled are called *independent variables*.
  - Those variables that we want to study to see the effect of the changes in the independent variables are called *dependent variables (response variables)*

- An experiment studies the effect of changing one or more independent variables.
  - Those variables are called factors.
  - A treatment is one particular value of a factor.

Variables/Treatments/Objects/Subjects

- Example 1: To determine the effect of Pair programming on code quality.
  - Pair programming (independent variable) as we can control it
  - Effect on code quality (dependent variable), as we want to study this variable and see if changes in the independent variable can have effect on this variable.
- Example 2: To determine the effect of aerobic training on Percentage of Body Fat
  - The independent variable is aerobic training intensity, i.e., high, medium and low (can be controlled)
  - The dependent variable is percentage of body fat
- Subjects: People, Objects: code quality and body fat
- Treatments: pair programming and aerobic training intensity

Tests (Trials)

- An experiment consists of a set of tests (sometimes called trials) where each test is a combination of **treatment**, **subject** and **object**.

- **Example.** A test can be that person N (subject) uses the new development method Y (treatment) for developing program A (object). This is trial -1

- **Example.** A test can be that person X (subject) uses the new development method Y (treatment) for developing program A (object). This is trial -2

Human-Oriented Vs Technology-Oriented

- In **human-oriented** experiments, humans apply different treatments to objects
  - For example, two inspection methods (treatments) are applied (by human) to two pieces of code (objects).
- In **technology-oriented** experiments, typically different tools are applied to different objects,
  - For example, two test case generation tools (treatments) are applied to the same programs (objects).
- The human-oriented experiment has less control than the technology-oriented one, since humans behave differently at different occasions, while tools (mostly) are deterministic.
  - Due to learning effects, a human subject cannot apply two methods to the same piece of code (as after first method human will learn and his application of second method will be different), which two tools can do without bias

Limitations in Controlling Human-Oriented Experiments

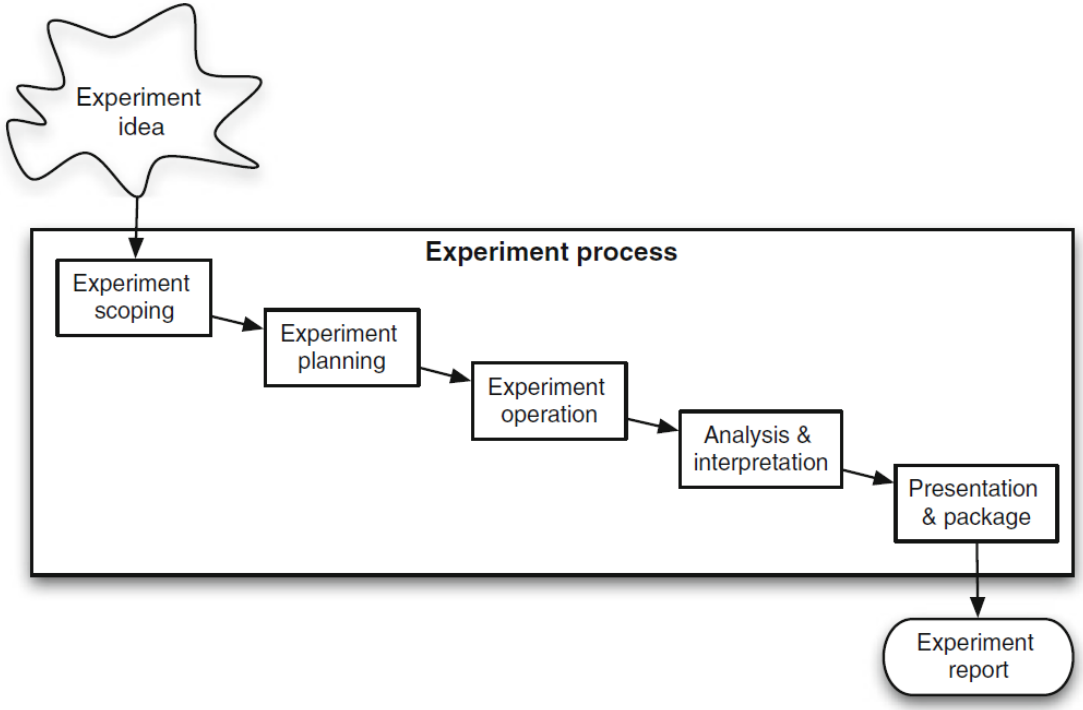
- This implies several limitations to the control of the experiment.
  - Humans have different skills and abilities, which in itself may be an independent variable.
  - Humans learn over time, which means that if one subject applies two methods, the order of application of the methods may matter, and also the same object cannot be used for both occasions.
  - Human-oriented experiments are impacted by all sorts of influences and threats, due to the subject's ability to guess what the experimenter expects, their motivation for doing the tasks etc.

- It is critical for the outcome of the experiment how subjects are selected and treated.

Experiment Process

- Processes are important as they can be used as checklists and guidelines of what to do and how to do it.

- Experiments is **fixed design** studies, primarily execute the steps once.



Experiment Process

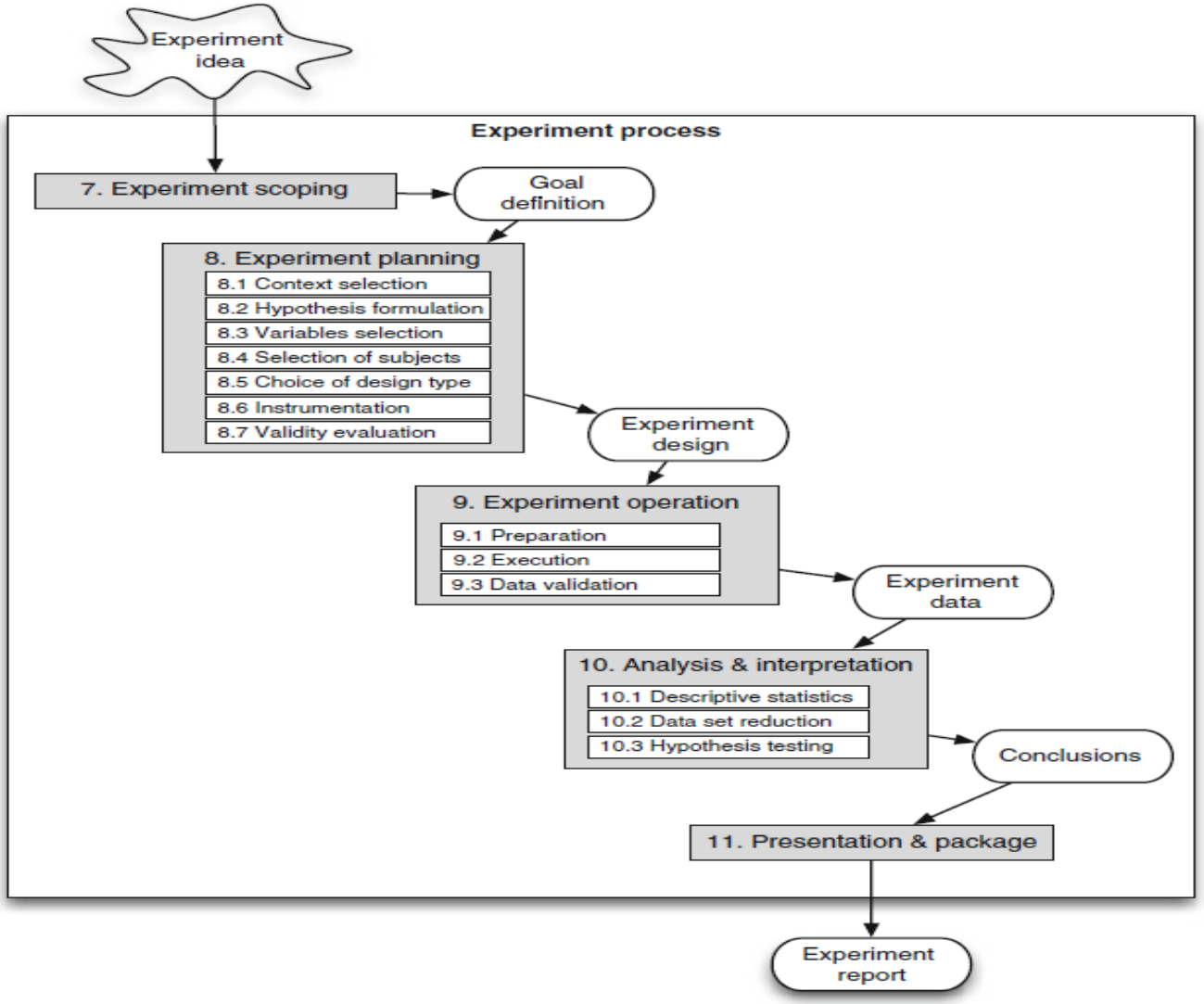
- The process is not supposed to be a ‘true’ waterfall model; it is not assumed that an activity is necessarily finished prior to that the next activity is started.

- The order of activities in the process primarily indicates the starting order of the activities.
  - The exception is when the operation of the experiment has started, then it is not possible to go back to the scoping and planning of the experiment.

Experiment Process

- When experimenting, subjects are assigned to different treatments at random.
- In cases where it is impossible to randomly assign treatments to subjects, we may use quasi-experiments.
- **Quasi-experiment** is an empirical enquiry similar to an experiment, where the assignment of treatments to subjects cannot be based on randomization but emerges from the characteristics of the subjects or objects themselves, who is using one method and who is using another method.

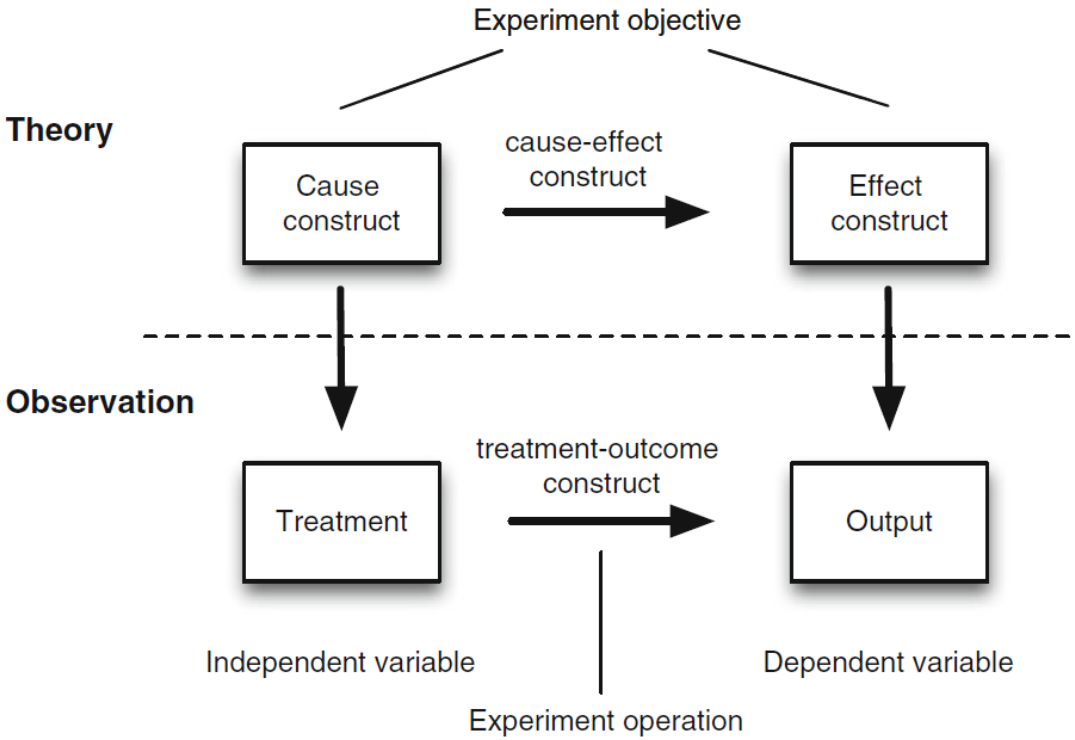
Experiment Process



Cause-effect construct examples

- While using email account, on entering valid email, the system accepts it but, when you enter invalid email, it throws an error message. In this technique, the input conditions are assigned with causes and the result of these input conditions with effects
- The lightning struck the tree (cause), The tree caught on fire (effect).
- Eating too much fast food without any physical activity leads to weight gain. Here eating without any physical activity is the "cause" and weight gain is the "effect."

Class activity



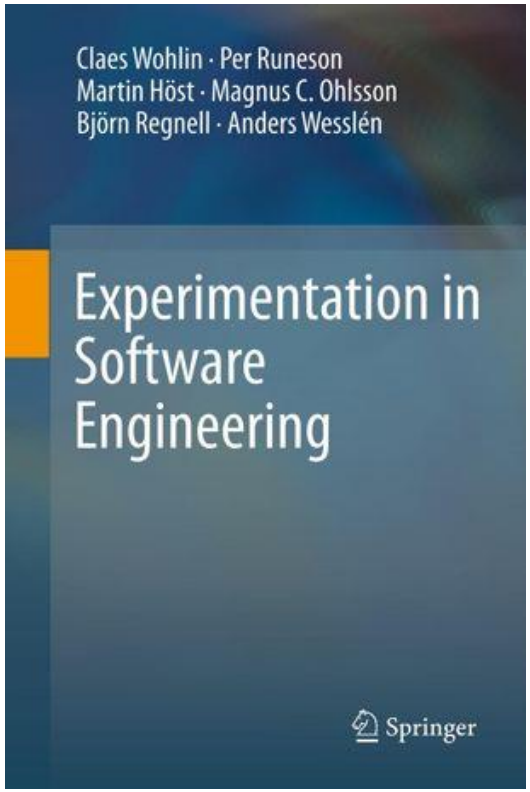
Using the above diagram, design a cause-effect construct for your term paper.

Pair programming (cause) has effect on code quality (effect).

Summary

- Experiments Fundamentals
- Experiment process

Reference



Chapter 6