Modelling of Computer Systems

Computer Science is about building and using hardware and software entities which we call **systems**. From an industrial perspective, these systems must satisfy a number of postulates:

- **It works**. A system that does not work might sell (cf. early Windows) but is unacceptable in the long run. Other courses cover software and hardware correctness/reliability; in this course we falsely assume that if it is called a **system**, it works.
- **It is efficient**. Modelling is a prime method of evaluating system efficiency ("performance") and this is what this course deals with, albeit only at an introductory level.
- **It is cost–effective**. This extremely important topic is not covered here.

This course introduces various ways of modelling computer systems in order to be able to assess some aspects of the performance of these systems. While many modelling methods are in use, **simulation** will be the prime method discussed here, with some coverage of analytical techniques (i.e. mathematical).

Statistics will be used heavily; lab meetings will help those students whose grasp of introductory Statistics is dubious but nobody will help those whose knowledge is non–existent.

System

What is a system? Anything that is self–enclosed and communicates with the rest of the world by accepting input and producing output.

In the Computer Science domain, systems come in two varieties:

Software	Hardware
Algorithm	Circuit
Program	Device
	Communication Channel
Software	Computer system

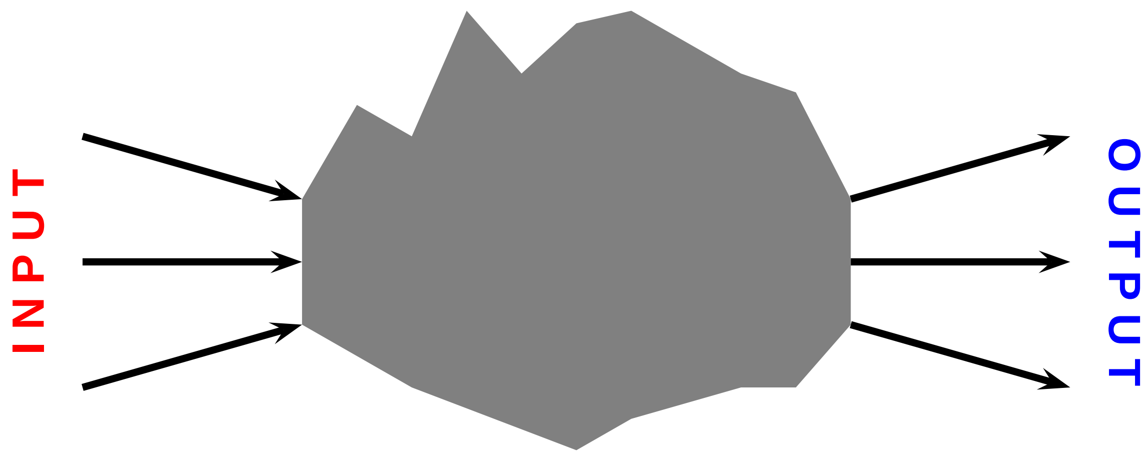
This table illustrates the fact that systems may be built of sub–systems, as a program may contain several algorithms; the performance of each of these algorithms influences the performance of the whole program, but not to the same degree. The same applies to hardware systems.

Software is a category that includes products such as Operating Systems, DB Systems, etc. Such products seldom **work** hence it is difficult to talk about their efficiency; besides, their complexity makes modelling them hard (too hard for CIS2460).
Salesmen simplify their task by reducing the performance of a software system to a single statistic, such as "mega transactions per second" (for DB systems).

Computer Systems are not doing anything without software systems driving them; hence, it is pointless to study the performance of a "computer system" in isolation of its software (and user) environment. The usual performance measures given for computer systems are descriptions of the performance of individual devices forming the system (e.g. CPU clock rate as a faulty measure of **speed**, whatever speed means).

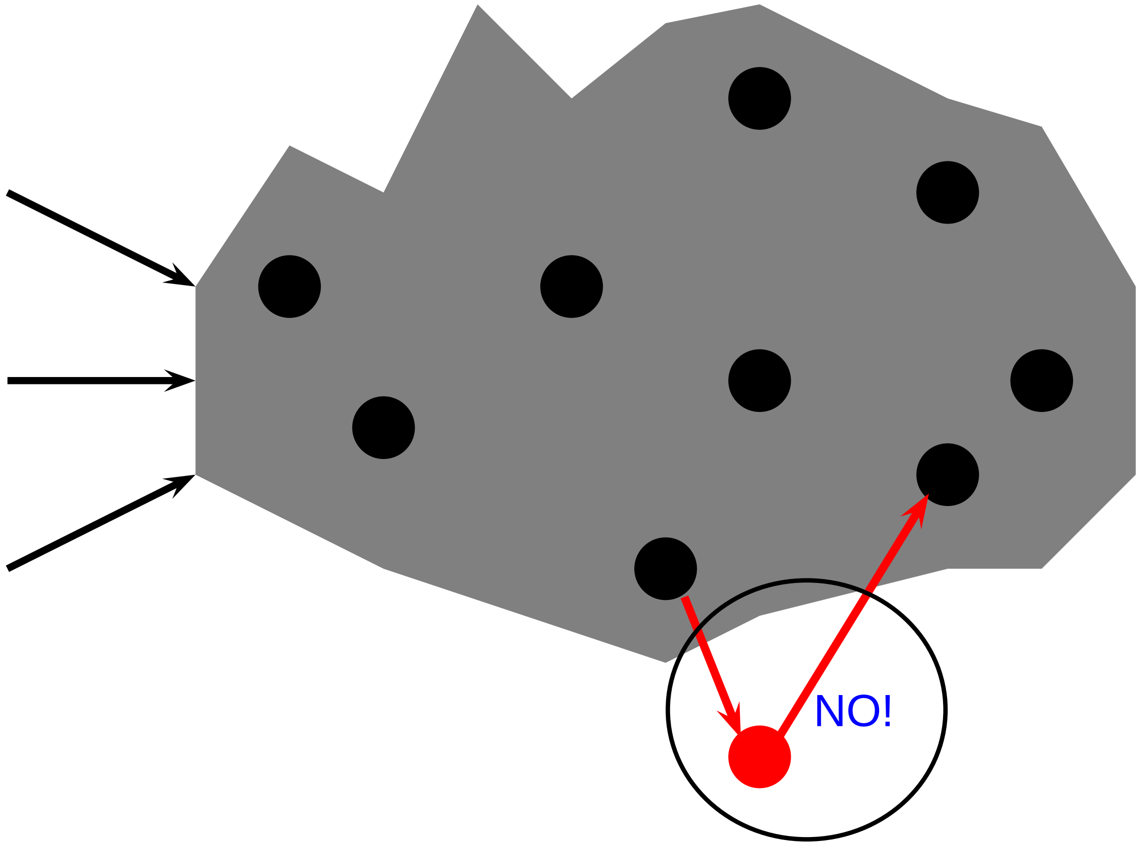
System

The most important property of a system is that it is **self–enclosed** so that it does not depend on **feedback** from anything that lies outside the system.



A system is made of components which may also be self–enclosed, forming sub–systems. The **black box** approach is often used in dealing with complex subsystems.

System components



Modelling

The key part of modelling is developing a **model** of the system under investigation.

A **model** usually is a simplification of the modelled system, but one that retains the critical properties of that system. A model can be formulated as a set of:

Mathematical, logical or operational relationships

among the **objects** (components) forming the system.

What for?

Trivial systems can be handled using a mathematical methods called **Queuing Theory**. Complex models are studied using computer simulation.

In either case, the main interest is in seeing how the behaviour of the system (as modelled) changes under different conditions (such as the rate of arrival of customers) or studying the impact of modifications to the model on the performance (the famous **“what if?”** question).

Is modelling a scientific method?

Every model distorts reality and thus modelling can only be considered a scientific tool to the same degree as **laboratory tests** in other sciences.

We use modelling only because no other options are known. We must, however, keep in mind that modelling never **proves** anything; it only indicates that some behaviour is likely, and even that is true only to the extent that the **model is faithful**.

System vs. model