

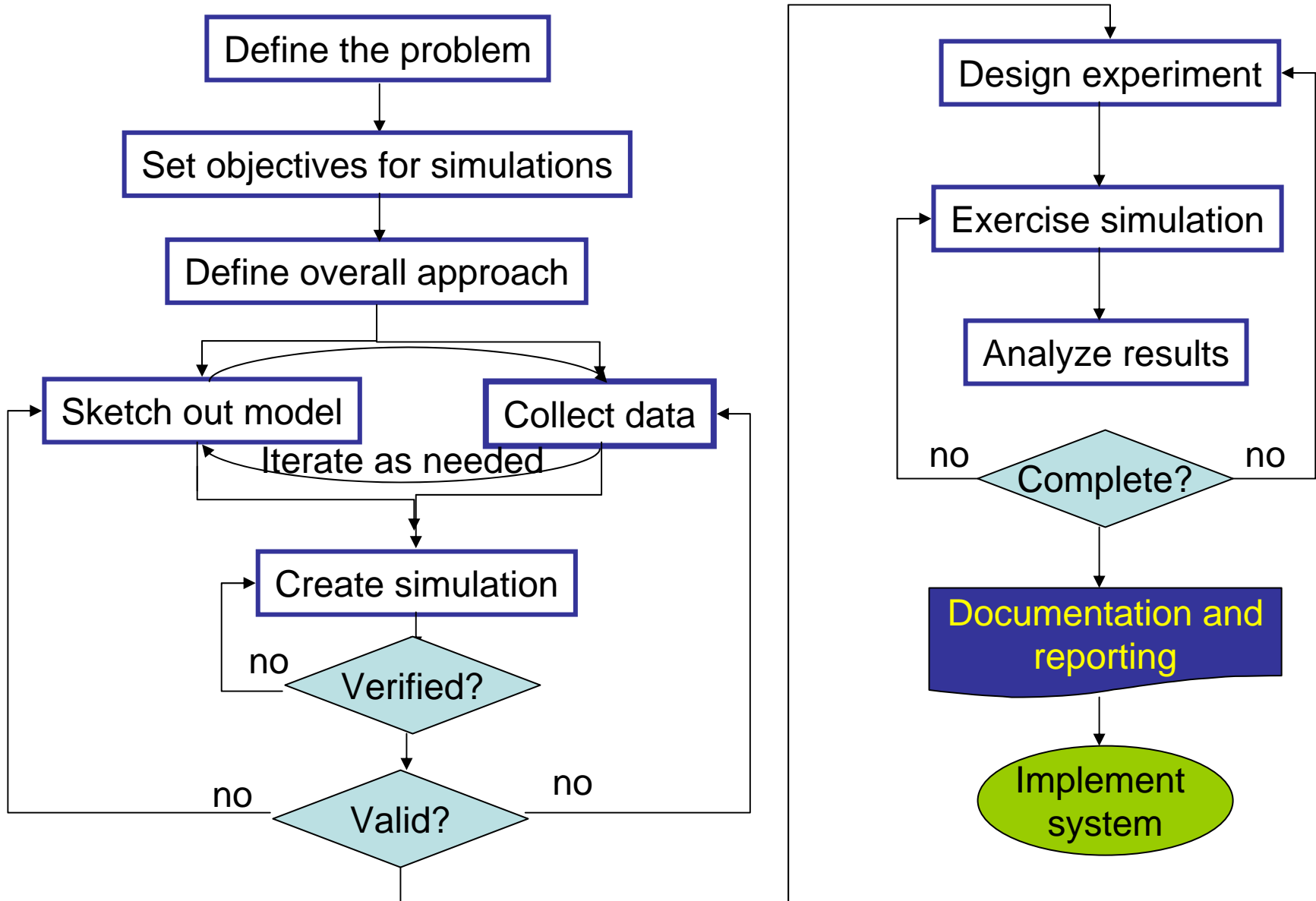
CPE 345: Modeling and Simulation

Lecture 11

Today's topics

- Verification and validation of your simulation

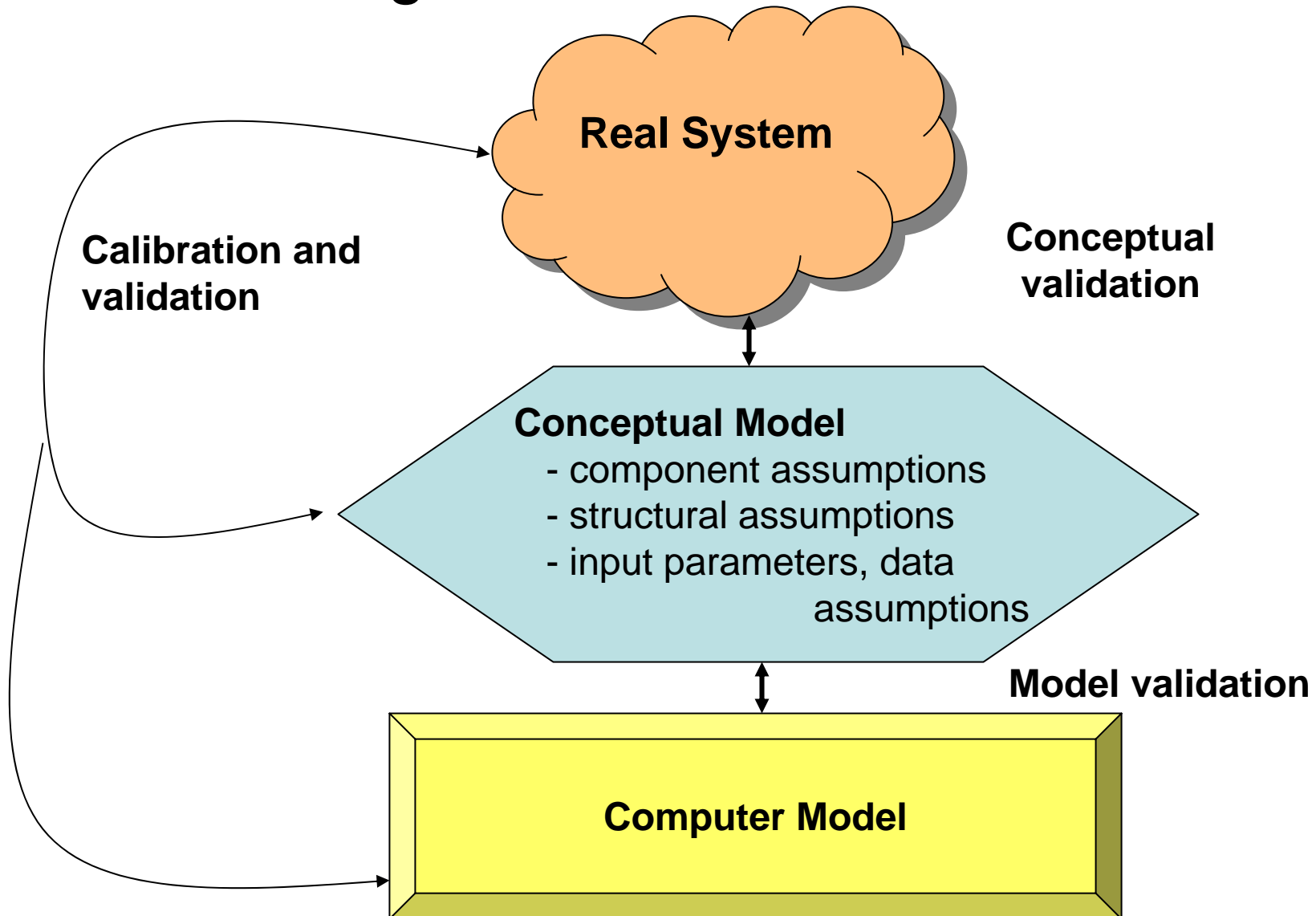
Recall: Steps in a simulation study



Verification versus validation

- We always simulate a model of the real system
- When building the simulation there are two questions to be asked
 - Is the model correct (captures the desired characteristics of the real system) ?
 - **Validation**
 - Is the model built correctly (no logical errors in your simulation)
 - **Verification**
 - use debugging tools
- **Validation**
 - Face validity
 - Model assumptions
 - Input-output transformation
 - Reliability and credibility

From modeling to simulation: model building, verification and validation



From modeling to simulation: model building, verification and validation

- Steps
 - Observe and understand the real system's operation and behavior
 - Create a conceptual model
 - Create a computer model

Verification

- Suppose that you do have the conceptual model of the system
- How to ensure that your computer implementation is accurate?
 - Logical errors – hardest to debug

Some hints:

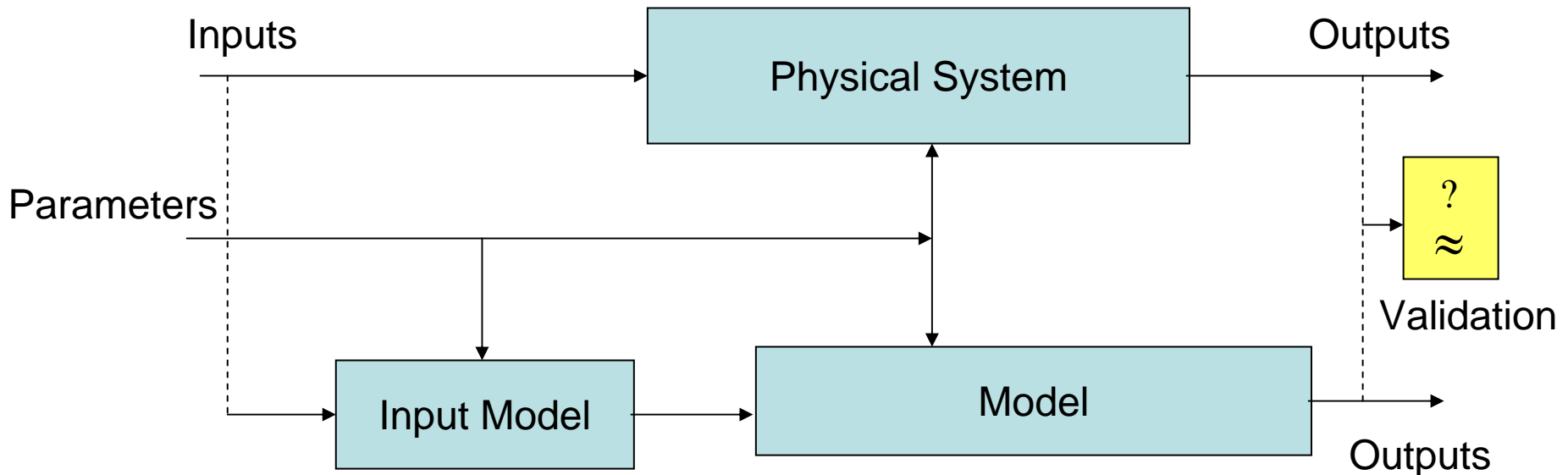
- Use graphical representation whenever possible
 - Animation
 - Track evolution of internal variables (object watches in OMNET++)
 - Plot traces with the system evolution (output vector in OMNET++)
 - Output variables graphs (use Plove to plot output vectors and apply various filters to the data)

Verification – more hints

- Test the simulation model for reasonableness under extreme value conditions
- Test the simulation model for reasonableness under variations in parameters
 - Does the response follow the direction of change?
- Examine the sequence of events
- Whenever possible, compare output results with analytical derived values
 - It might be useful to run a simplified simulation that has an analytical solution, just for verification

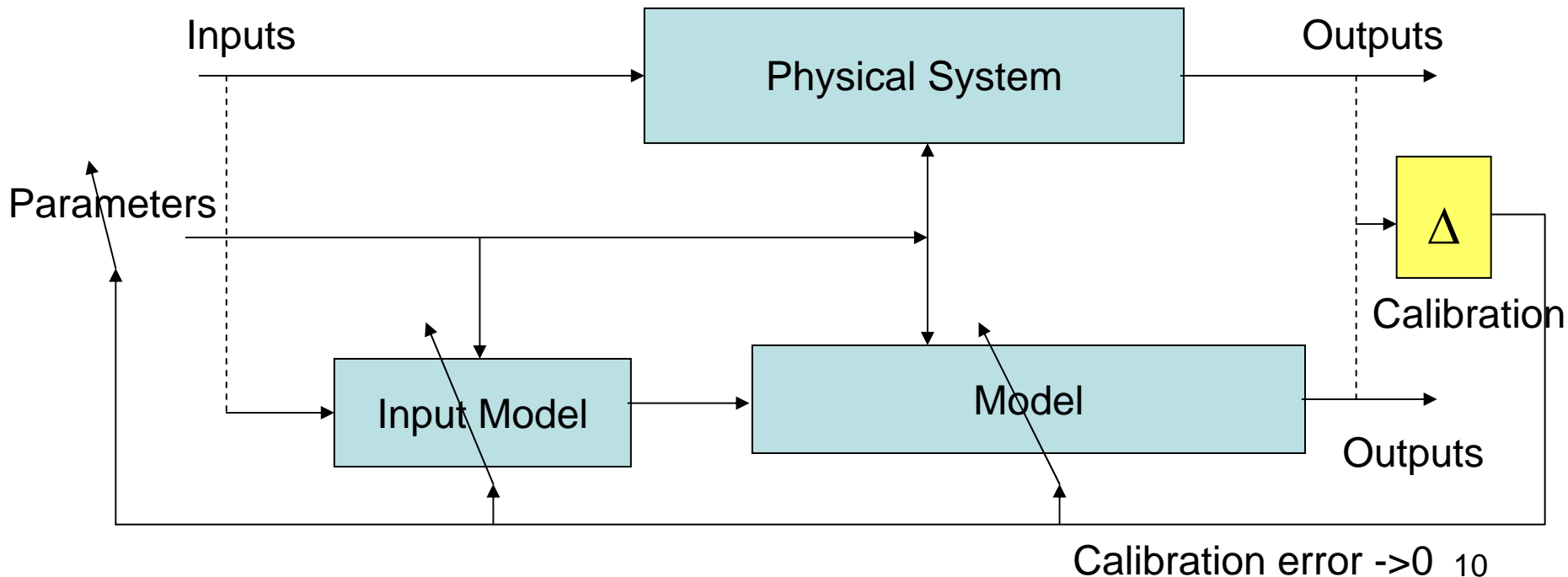
Calibration and Validation of the Model

- Calibration and validation – distinct processes, but performed simultaneously and interactively
 - Validation → does the behavior of the simulated system match that of the real system ?



Calibration and Validation of the Model

- Calibration – adjust models and parameters to fit real system behavior
 - Feedback control system → adjust model to minimize the error
 - Calibration/Validation stops when error is less than some threshold



Some validation techniques: face validity

- Is the model reasonable on its face?
 - without deep inspection or analysis
- Does model behavior change in expected ways with modification of parameters?

Face validity: example

- We know that the server utilization for an M/M/c queue is

$$\rho = \frac{\lambda}{c\mu}$$

- What happens if we assume in our model that the server utilization is

$$\rho = \frac{\mu}{c\lambda}$$

- Is this a valid model ?
- Let's do a face validity test
 - Is the model reasonable on its face?
 - Verify units: utilization is a dimensionless quantity and λ , μ are rates, and c is dimensionless \rightarrow this passes the first test

Face validity example – cont.

- Does model behavior change in expected ways with modification of parameters?
- What happens to the server utilization as the arrival rate λ , or service rate μ increases, or as the number of servers, c , changes?
 - Utilization
 - Increases with increasing service rate ???
 - Decreases with increasing arrival rates ???
 - Decreases with the increase in the number of servers
 - This model is invalid on its face

Another example for face validity

- Pareto distribution is used to model the internet traffic. Assume that one student wanted to compute some delay properties in a computer network using internet traffic and he did a “Google” search to find out the CDF for Pareto. The formula he found is

- $$D(x) = 1 - \left(\frac{b}{x}\right)^a, x \geq b$$

- Is this formula valid on its face?
- The student also found a web site that posted a formula for the delay experienced in a single server queue with exponential service times (service rate μ) and Pareto distributed arrival process (with mean μ_a and variance $\sigma_a^2 \neq 0$):

$$w = \frac{\mu_a^2}{\mu_s^2}$$

Solution

- Verify CDF properties:
$$\begin{cases} \lim_{x \rightarrow \infty} D(x) = 1 \\ \lim_{x \rightarrow -\infty} D(x) = 0 \\ x_1 \leq x_2 \Rightarrow D(x_1) \leq D(x_2) \end{cases}$$

– Formula is valid on its face

- Verify units for the delay formula

$$w = \frac{\mu_a^2}{\mu_s^2}$$

– Formula gives a dimensionless quantity \rightarrow it is not valid on its face (delay is measured in time units)

Validation of Model Assumptions

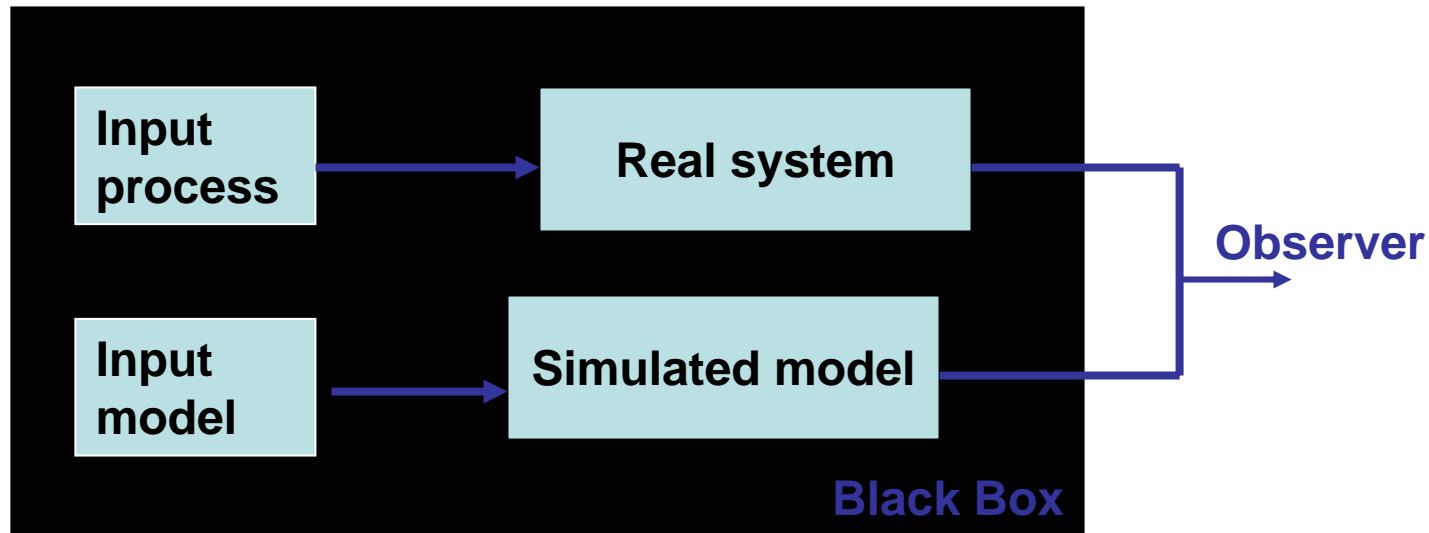
Type of model assumptions	Based on
Structural assumptions <ul style="list-style-type: none">- system operation- simplifying assumptions- abstractions of reality	Observation of the system
Data assumptions <ul style="list-style-type: none">- Homogeneity- Independence- Distribution	Collection of reliable data Correct statistical analysis

Validating input-output transformations

- We can see the outputs of the system as being a functional transform of the inputs, based on parameter settings
- Instead of validating the model input-output transformations by predicting the future → may use past historical data which have been reserved for validation purposes only
 - Separate data sets should be used for modeling/calibration and final validation, respectively

Input-output validation: the Turing test

- Consider an observer
 - Knows nothing about the real system or model



If the observer cannot distinguish the output for the real and the simulated system with any consistency → validity of the simulation is not rejected

If the observer consistently identifies simulation – improve model, based on the observer's critical observations

In summary

- The quality of your simulation is only as good as your system model is
 - Validation – very important

Possible validation techniques:

- Test for face validity
- Statistical tests on input
- Turing test
- Compare model and real-system output using a statistical test
- Collect data and use it for validation. Also use historical data

Homework (not to be handed in)

- In the context of your projects, what models are you using for the system and the input data?
- How do you think of validating these models?
- What validation techniques will you use for your project?