VERIFICATION, VALIDATION AND ACCREDITATION

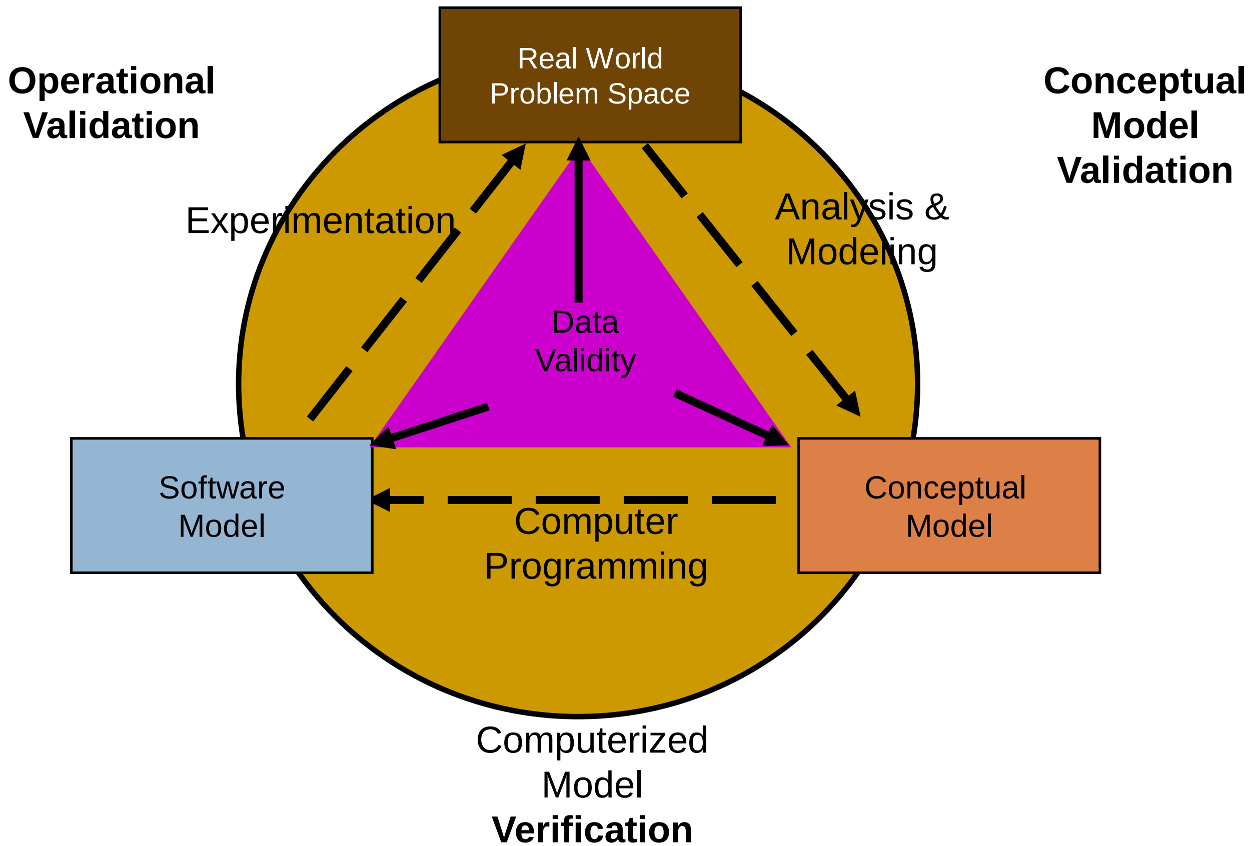
Pau Fonseca i Casas; pau@fib.upc.edu
Herman D. Hughes; hughes@cse.msu.edu

Validation?

- The rationalist view: "Rationalism holds that a model is simply a system of **logical deductions made from a set of premises of unquestionable truth**, which may or may not themselves be subject to empirical or objective testing. In its strictest sense, the premises are what Immanuel Kant termed synthetic a priori premises."
- The empiricist view: "The empiricist holds that **if any of the postulates or assumptions used in a model cannot be independently verified by experiment**, or analysis of experimental data, **then the model cannot be considered valid**. In its strictest sense, empiricism states that models should be developed only using proven or verifiable facts, not assumptions." (Naylor and Finger 1967)
- The positivist view: "The positivist states that **a model is valid only if it is capable of accurate predictions**, regardless of its internal structure or underlying logic. Positivism, therefore, shifts the emphasis away from model building to model utility."

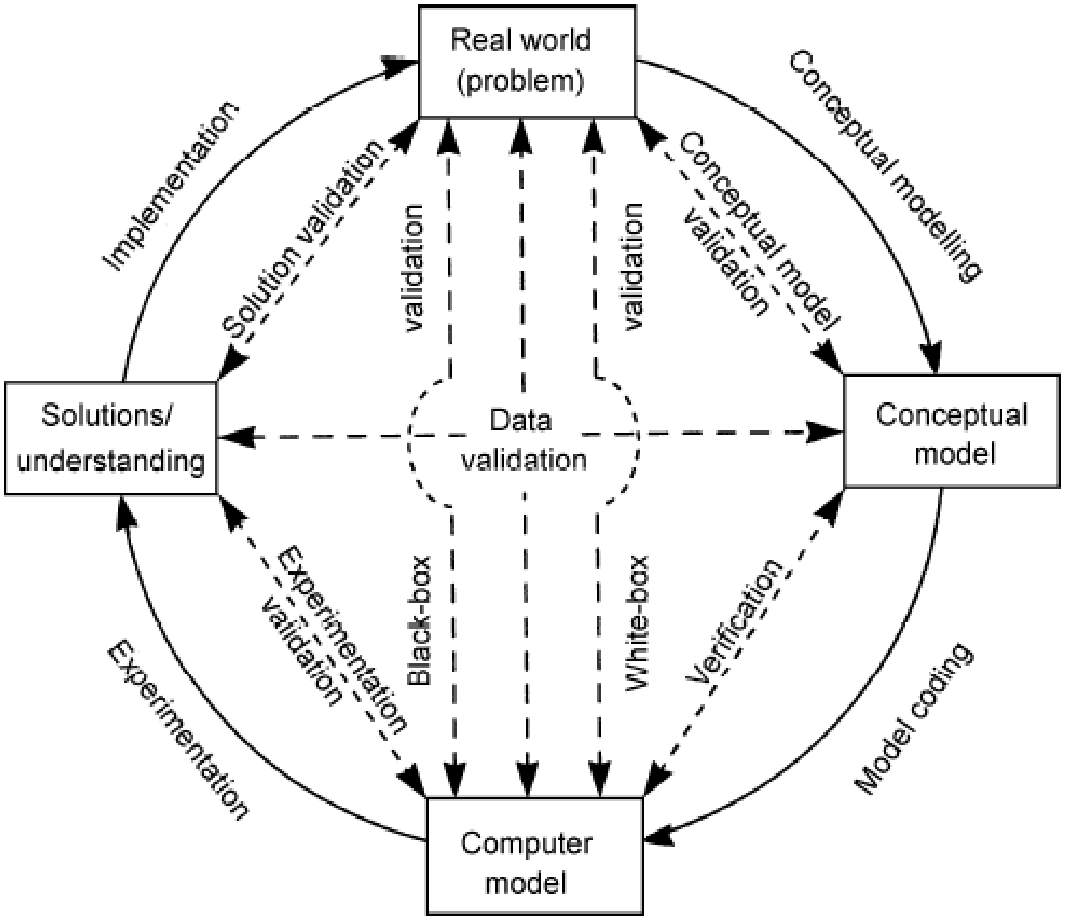
The approach we follow on Boston Marathon Simulator is the positivist approach.

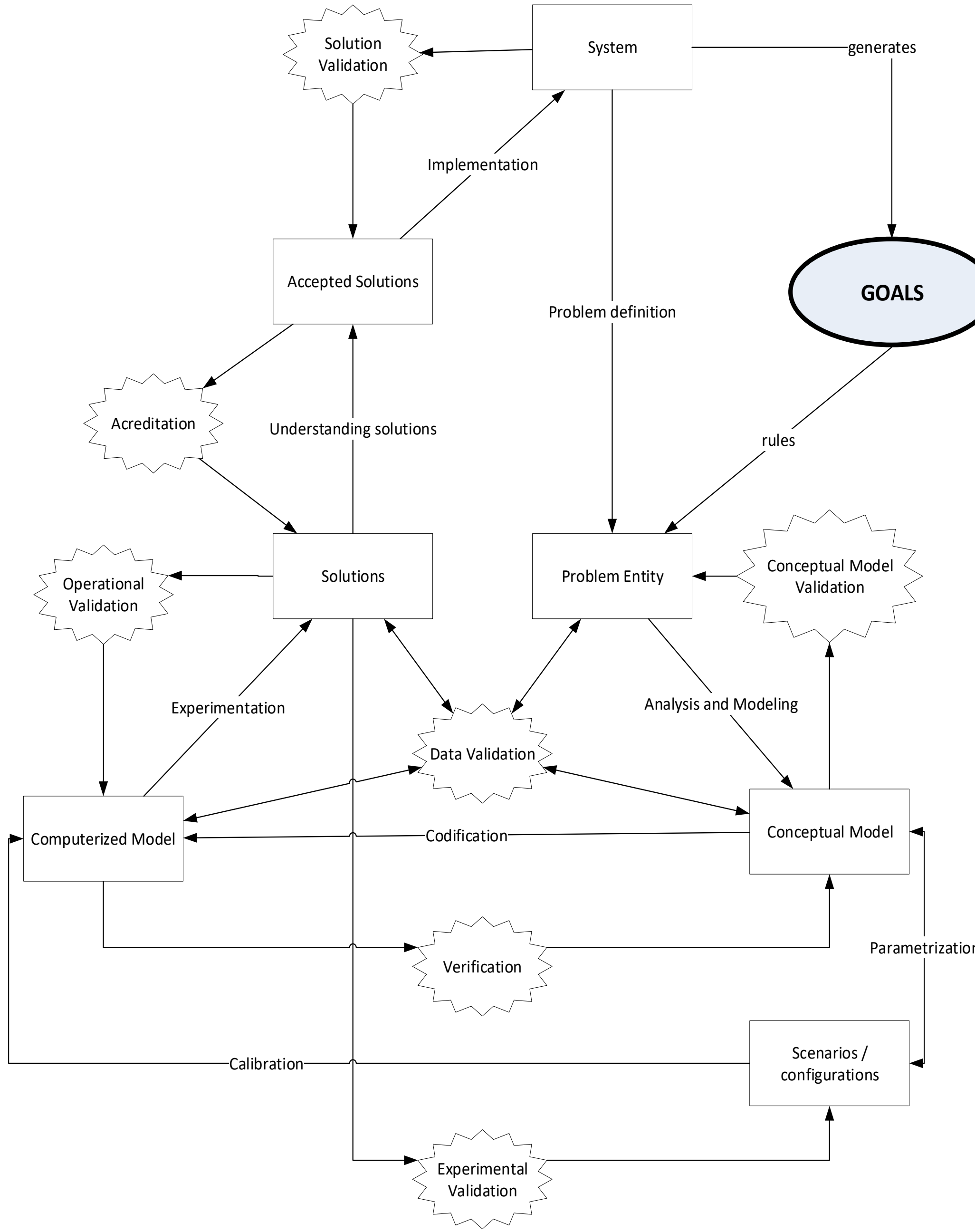
Introduction



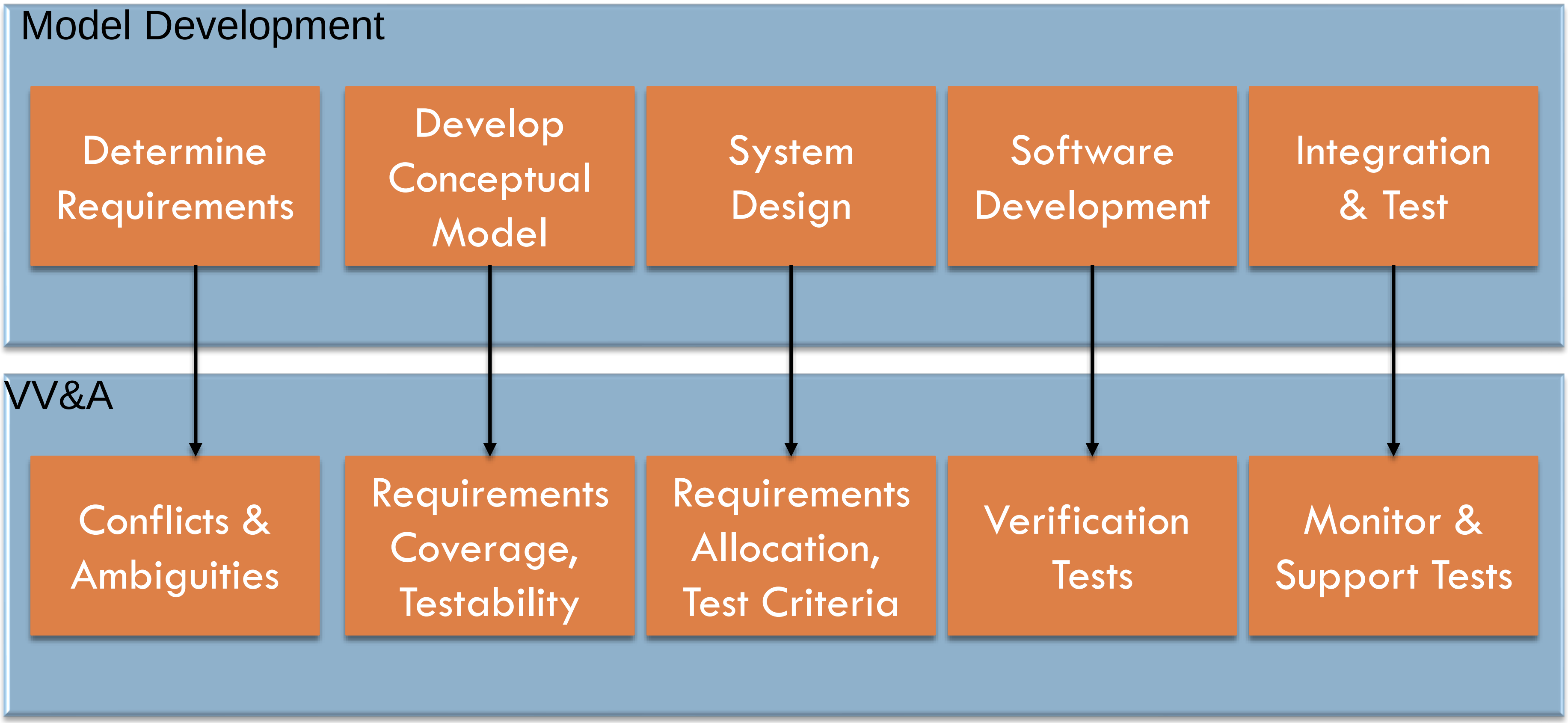
Sargent R. Verification and validation of simulation models. In: Rossetti MD, Hill RR, Johansson B, Dunkin A, Ingalls RG, editors. Proceedings of the 2009 Winter Simulation Conference (WSC) [Internet]. 2009 [cited 2014 Oct 8]. p. 66–66. Available from: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5235461

Introduction

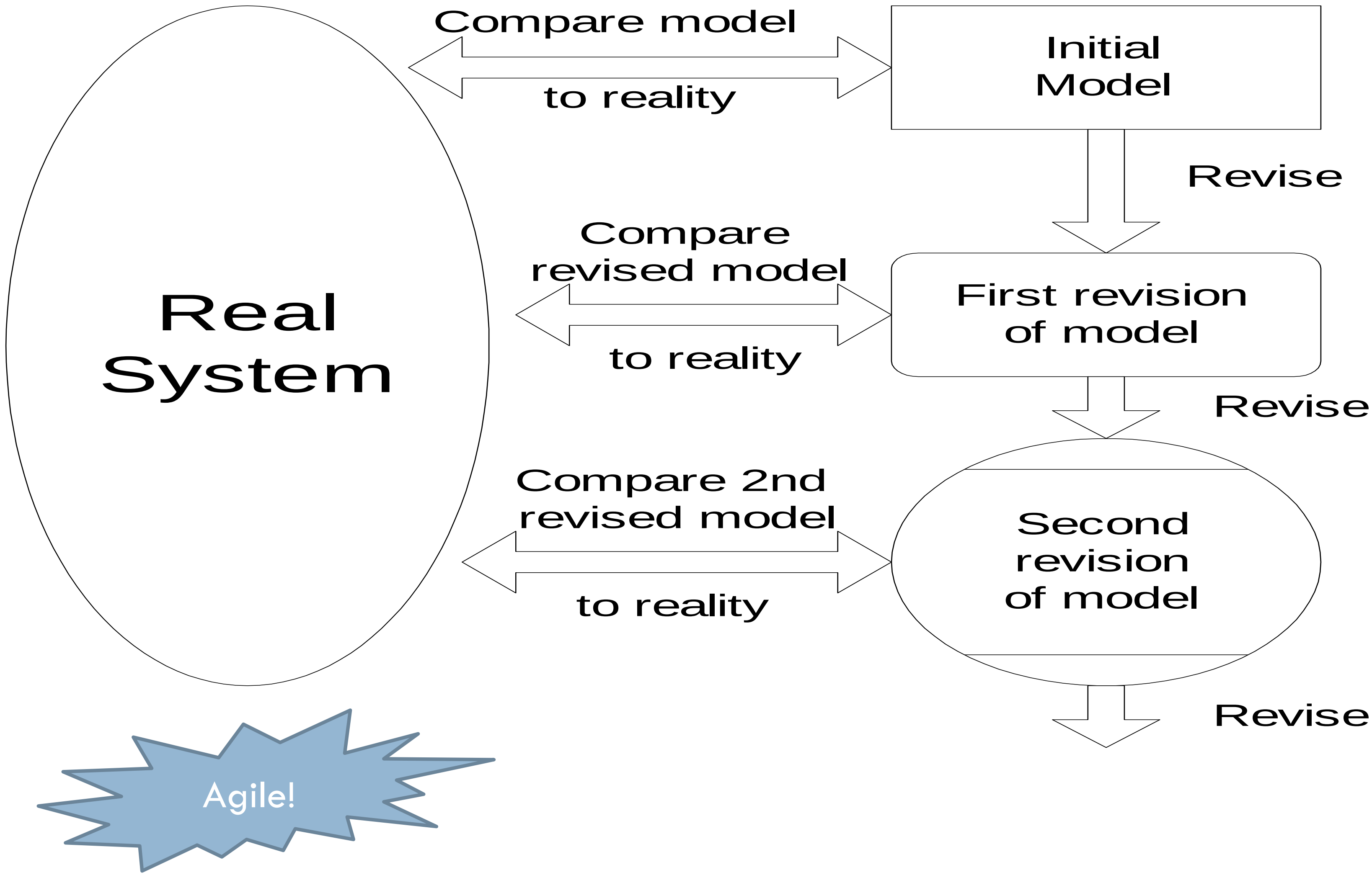



System
generates
Solution Validation
Implementation
GOALS
Accepted Solutions
Problem definition
Acreditation
Understanding solutions
rules
Operational Validation
Solutions
Problem Entity
Conceptual Model Validation
Experimentation
Analysis and Modeling
Data Validation
Computerized Model
Codification
Conceptual Model
Verification
Parametrization
Scenarios / configurations
Calibration
Experimental Validation

Collaborative VV&A Process



Iterative VV Process



Aspects to consider

- **Agreement** between the modeller and the client (approach used to validate, experiments, information and documentation to use).
- **Agreement** between (modeller and client) and other possible parts.

To achieve the agreement

- **Purpose** of the model: A model is developed with a specific objective.
- **Validity** of the model: range of precision, can only prove that a model is false, you must do test.
- **Iterations** of the model, before reaching a satisfactory model.
- Is needed to separate the VV&A processes?
  - Development of the model: team one
  - Validation, Verification and Accreditation: team two.

Who performed?

- The **same** team that develops always carries out the V & V. (Sargent).
- IV&V (**Independent** validation and verification team), usually for large projects. (Sargent, Law).

Errors associated with the VV&A

- Correct decision
- Incorrect decision:
  - Error type I
    - Rejecting a valid model.
    - Risk of the modeller.
  - Error type II
    - Accepting an invalid model.
    - Risk of the customer.
    - Dangerous.

Type of errors

- Type I error: "rejecting the null hypothesis when it is true".
- Type II error: "accepting the null hypothesis when it is false".
- Type III error: “solving the wrong problem [representation]“.
- Type IV error: "the incorrect interpretation of a correctly rejected hypothesis”.

Dirty Rotten Strategies: How We Trick Ourselves and Others into Solving the Wrong Problems Precisely (High Reliability and Crisis Management), Ian I. Mitroff, Abraham Silvers. ISBN-13: 978-0804759960

Objectives of the VV&A

- Produce a model that represents the system behavior **as close as** possible to make it useful.
- Increase the credibility of the model so that it can be used for management and for prediction.
- The validation:
  - We have built the correct model
  - Is the appropriate model to represent the real system?
- The verification should ask:
  - We have built the model correctly?

To bearing in mind

- Must be an integral part of the development of the model, not an isolated part.
- Is an iterative task, starts the day 0.

Techniques of VV&A

- Informal techniques
- Formal techniques
- Static techniques
- Dynamic techniques

Informal techniques

- Every system contains an operation of inherent logic that is known to **experts.**
  - These people know the system works perfectly.
  - Are the best suited to determine whether the model fits or not whose it believed appropriate.
  - Must preserve maximum independence of the group guarantor in order to ensure their objectivity.

In Boston Marathon, the experts are those who understand the system (Marathon and runners).

Formal techniques

- For example, the **calculation of the predicates** guarantees completely the correctness of the model.
- However these techniques tend to over-complicate the understanding of the model and tend to be complicating to implement for some complex models.

Static techniques

- Evaluate the static model design and the code used for its implementation.
- Using this methodology should put special emphasis on two aspects:
  - The **formal construction** of the simulation model, based on an appropriate methodology for establishing a good communication channel between all members of the simulation team and experts of the system.
  - Set the method for, **from formalism, implement** in the computer, **the simulation model**. There are simulation systems that allow doing this step automatically, thus guaranteeing this way at this point.

You can enter at http://sdlps.com/ to review an approach based on SDL and C++.

Dynamic techniques

- Analyze the **results** provided by the simulator.
- Used common statistical techniques to assess whether the data that the simulator provides conform to reality or not.

Difficulties of the VV&A

- No exist something called general validation:
  - A model is only valid according to their purpose.
  - A model may be valid for one purpose and invalid for another.

- *"All models are wrong, but some models are useful."*

Professor George Box (18 October 1919 – 28 March 2013)

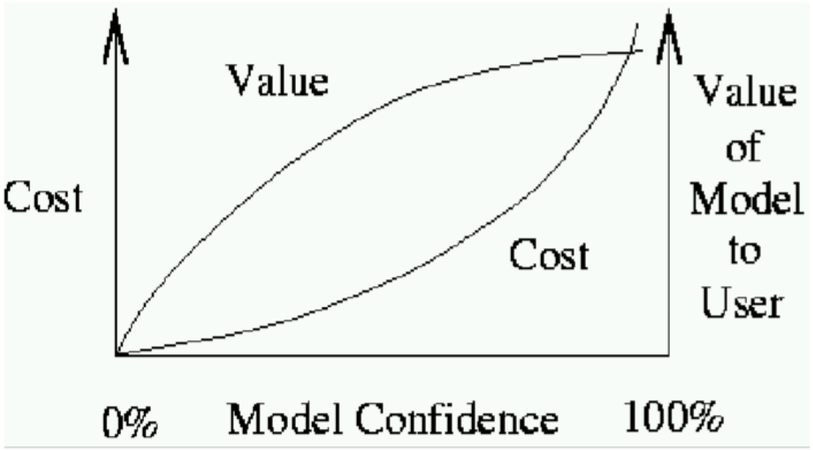
http://www.engr.wisc.edu/ie/faculty/box_george.html

Difficulties of the VV&A

- It is possible that a “real world” does not exist to compare with the model:
  - Often the models are created to evaluate alternatives exist.
- What is the “real world”?:
  - Different roles have different visions of the system. The interpretations and therefore the real world vary.

Difficulties of the VV&A

- Often the system data are not adequate:
  - Maybe the data don't exist.
  - You might not represent all possibilities.
- The time:
  - No time to validate and verify everything.



Sargent R.

Difficulties of the VV&A

- Only can **demonstrate** that the model is **wrong**:
  - As more tests are done which can not demonstrate that the model is incorrect, the confidence interval of the model grows.
  - The objective of V&V is to increase this confidence interval.

The logic of scientific research (1935), Karl Popper

Difficulties of the VV&A

- A valid model is not necessarily credible, and inverse.
- A simulation model and its results have credibility if the contracting parties believe their correct results.

Robert G. Sargent
http://www.informatik.uni-trier.de/~ley/db/indices/a-tree/s/Sargent:Robert_G=.html

Stewart Robinson
http://www.stewartrobinson.co.uk/

Averill M. Law
http://www.averill-law.com/

VV&A

Validation

Validation

- The validation is the process of comparing the **behavior of the model** and the **behavior of the real system.**
- Build the **correct** model.

Validation

- **Data validation**. We want to assure that the data we are going to use on the model is accurate, the probability distributions, the data sources that are used to obtain the data, the support structures to keep the data, etc. are accurate and correctly defined.
- **Conceptual model validation**. Conceptual model validation is determining that (i) the theories and assumptions underlying the conceptual model are correct and (ii) the model's representation of the problem entity and the model's structure, logic, and mathematical and causal relationships are "reasonable" for the intended purpose of the model.
- **Operational validation**. At this point, we own at least the first codification of the model, and we can obtain proposed solutions from it. We can see if the outputs of the model have the accuracy required in accordance with the problem entity.
- **Experimental validation**. We analyze if the experimental procedures used to obtain the results are enough accurate. The experimental validation is focused on the data and the relations we own, and we don't care regarding the elements that are not present on the model.
- **Solution validation**: in this validation, the focus is on the accuracy of the results obtained from the model proposed solution and the data obtained on the implemented solution. This is a key validation in the frame Industry 4.0 digital twin concept since assures that the model and the system are close enough to be useful for the goals proposed.

Validation

- Aspects to validate:
  1. Validation of **data.**
  2. Validation of the **conceptual model**: logical structure and hypothesis.
  3. **Operational validity**: In this step, see if the outputs of the model have the accuracy required in accordance with the problem.
- At this point the representation techniques can be extremely useful to visually check whether the behavior of the model is appropriate.

Validation

- **Naylor and Finger** formulated an approach based on 3 steps:

1. Build a model that **seems** valid.
   - If the model is reasonable for users and experts.
2. Validate the **assumptions** : how the system operates?
   - Structural hypotheses : how the system operates? VALIDITY OF THE **CONCEPTUAL MODEL**.
   - Data hypotheses: collection of reliable data and correct statistical analysis of data. **VALIDITY OF DATA**.
3. Compare the changes of the inputs and outputs in the model with corresponding inputs and outputs of the real system. **OPERATIONAL VALIDITY**.

Validation techniques (mainly informal and dynamic)

- Historical methods (rationalism, empiricism, positive economy.)
- Validation of multistage.
- Compare with other models.
- Tests degenerative.
- Validation for events.
- Time of extreme conditions.
- Validation “Face”.
- Fixed values.
- Validation with historical data.
- Internal validation.
- Animations.
- Variability of the parameters, sensitivity analysis.
- Predictive validation: is based on predictions with data system.
- Traces.
- Turing tests.
- Test chi, Kolmogorov, etc.

Validity of the data

Ensure that the data of the model used correctly

Validity of the data

- **Data Validation:** determining that the data required for model building, validation and experimentation are sufficiently accurate. (…), this applies to all aspects of the modelling process, since data are required at every stage of a simulation study. - (Transaction 69) **Stewart Robinson**
  - Checking that the data transformations are correct.
  - This applies to all aspects of the modeling process, since the data are necessary at each stage of the simulation study.
  - Data expiration.

Combining formal definition of a simulation model with heuristics to improve building sustainability
https://www.informs-sim.org/wsc18papers/includes/files/203.pdf

Type of data

- Data for model construction.
- To test.
- To experience the model validated.

Methods

- Good methods for obtaining the data.
- Test the data (internal consistency, statistical techniques).
- Procedures for keeping the data.
- Good databases.

- Define when the data is going to be outdated.
  - See the wildfire example.

Validity of the conceptual model

Ensure that the hypotheses are correct.

Validity of the conceptual model

- Determine that the scope and detail of the proposed model is enough for the purpose and that all assumptions are correct.
- The question to be answered is: Contains the conceptual model all the details necessary to cover the objectives of the simulation study?

Validity of the conceptual model

- Structural hypotheses : how the system operates.
- The hypotheses about the data should be based on a collection of reliable data and a proper statistical analysis of data.
- Evaluate each sub model regarding: Structure logic, causal relationships, detail versus aggregation.

Techniques

- Face validity: is asking people knowledgeable about the system whether the model and/or its behavior are reasonable. This technique can be used in determining if the logic in the conceptual model is correct and if a model's input-output relationships are reasonable. (Sargent – WSC 1998)
- Traces.

On Boston Marathon one can analyze the traces of a group to detect that the behavior of the model is as expected.

Trace

- Definition of Variables:
  - CLOCK = Simulation clock
  - EVTYP = Event type (Start, Arrival, Departure, Stop)
  - NCUST = Number of customers in system at time
  - STATUS = Status of server (1=busy, 0=idle)

- State of System Just After the Named Event Occurs:

CLOCK = 0 EVTYP = Start NCUST=0 STATUS = 0

CLOCK = 3 EVTYP = Arrival NCUST=1 STATUS = 1

CLOCK = 5 EVTYP = Depart NCUST=0 STATUS = 0

CLOCK = 11 EVTYP = Arrival NCUST=1 STATUS = 0

Must be 1

CLOCK = 12 EVTYP = Arrival NCUST=2 STATUS = 1

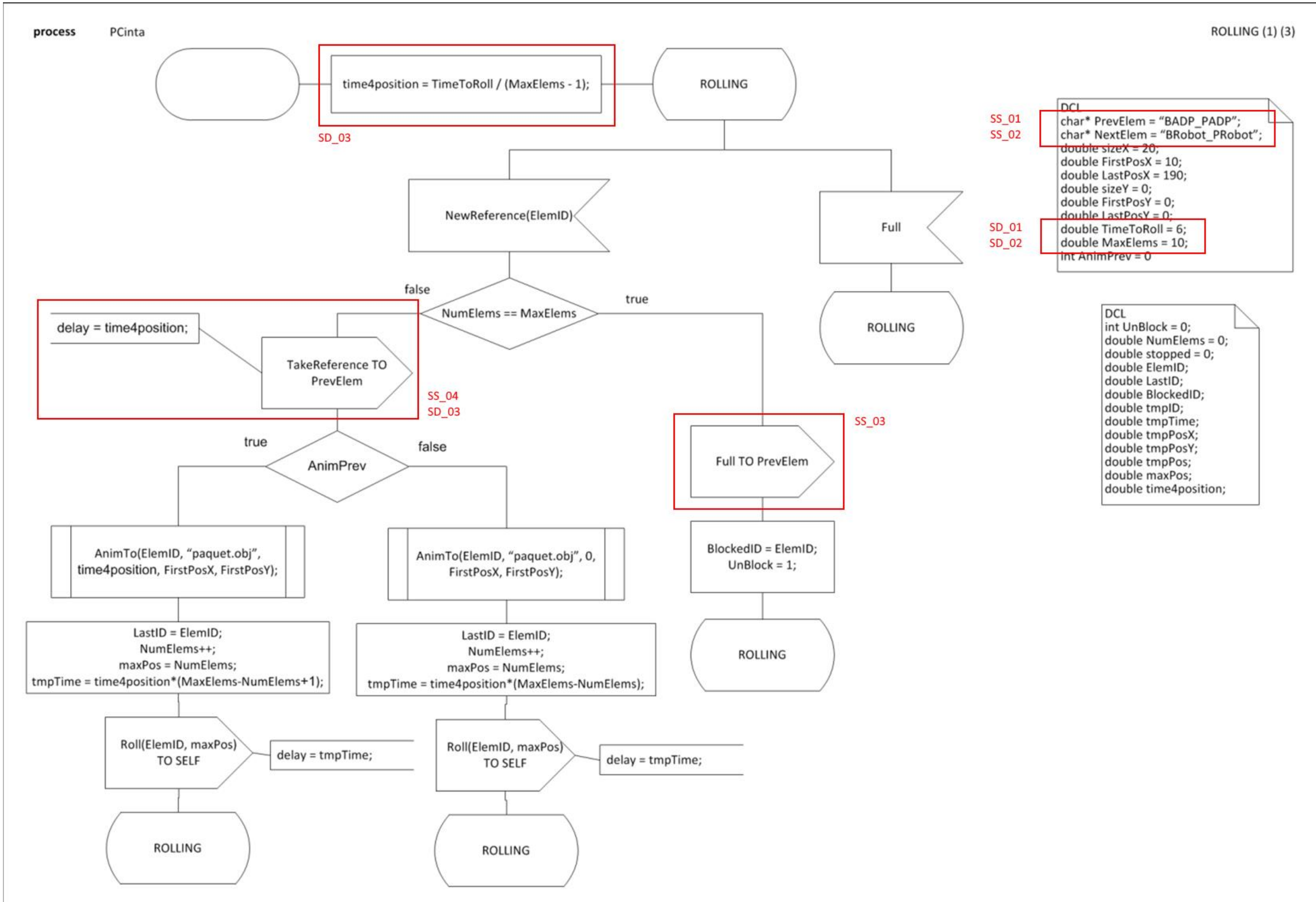
CLOCK = 16 EVTYP = Depart NCUST=1 STATUS = 1

Validity of the conceptual model

- The analysis of the Systemic Structural assumptions are based on the formal definition of the model.
- The analysis of Systemic Data assumptions consists of three steps:
  - Identifying the appropriate probability distribution.
  - Estimating the parameters of the hypothesized distribution.
  - Validating the assumed statistical model by a goodness-of fit test, such as the Chi-square or Kolmogorov-Smirnov test, and by graphical methods.

Validity of the conceptual model (Example)

- SS_01: The previous element of the conveyor is the machine named “BADP_PADP”.
- SS_02: The next element to the conveyor will be the robot “BRobot_PRobot”.
- SS_03: When the conveyor is full, it sends a message to the previous element in order to stop this machine.
- SS_04: When the conveyor has empty spaces, the previous element put a new box in the conveyor and starts its movement to the position that lasts a time defined in SD_03.
- SD_01: The time needed to cross the conveyor is 6 seconds, defined on *TimeToRoll* variable.
- SD_02: The maximum number of elements on the conveyor is 10, defined on the *MaxElems* variable.
- SD_03: The time needed for the element to reach its position on the conveyor depends is represented by the expression $\frac{TimeToRoll}{(MaxElems-1)}$.


process PCinta
ROLLING (1) (3)
time4position = TimeToRoll / (MaxElems - 1);
SD_03
ROLLING
SS_01
SS_02
DCL
char* PrevElem = "BADP_PADP";
char* NextElem = "BRobot_PRobot";
double sizeX = 20;
double FirstPosX = 10;
double LastPosX = 190;
double sizeY = 0;
double FirstPosY = 0;
double LastPosY = 0;
SD_01
SD_02
double TimeToRoll = 6;
double MaxElems = 10;
int AnimPrev = 0
NewReference(ElemID)
Full
ROLLING
false
NumElems == MaxElems
true
delay = time4position;
TakeReference TO PrevElem
SS_04
SD_03
SS_03
Full TO PrevElem
DCL
int UnBlock = 0;
double NumElems = 0;
double stopped = 0;
double ElemID;
double LastID;
double BlockedID;
double tmpID;
double tmpTime;
double tmpPosX;
double tmpPosY;
double tmpPos;
double maxPos;
double time4position;
true
AnimPrev
false
AnimTo(ElemID, "paquet.obj", time4position, FirstPosX, FirstPosY);
AnimTo(ElemID, "paquet.obj", 0, FirstPosX, FirstPosY);
BlockedID = ElemID;
UnBlock = 1;
LastID = ElemID;
NumElems++;
maxPos = NumElems;
tmpTime = time4position*(MaxElems-NumElems+1);
LastID = ElemID;
NumElems++;
maxPos = NumElems;
tmpTime = time4position*(MaxElems-NumElems);
ROLLING
Roll(ElemID, maxPos) TO SELF
delay = tmpTime;
Roll(ElemID, maxPos) TO SELF
delay = tmpTime;
ROLLING
ROLLING

Operational validity

Calibration of the simulation model.

Operational validity (Calibration)

- The objective of the test is to confirm the ability of the model to predict the behaviour of the real system.
- Iterative process of comparing the model and the real system: make adjustments in the model and compare the new model revised.
- Must collect over a set of system data.
- Trade-offs: cost/time/effort versus detail.

Operational validity

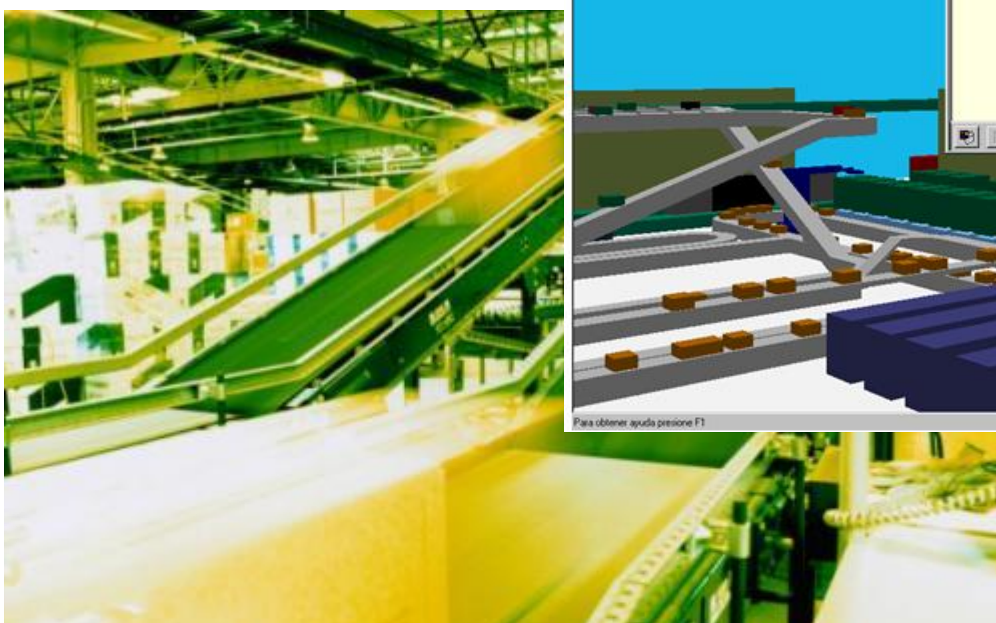
- Variety of techniques.
- There isn't an algorithm to select the techniques to use.
  - Depend on the problem, the system model.

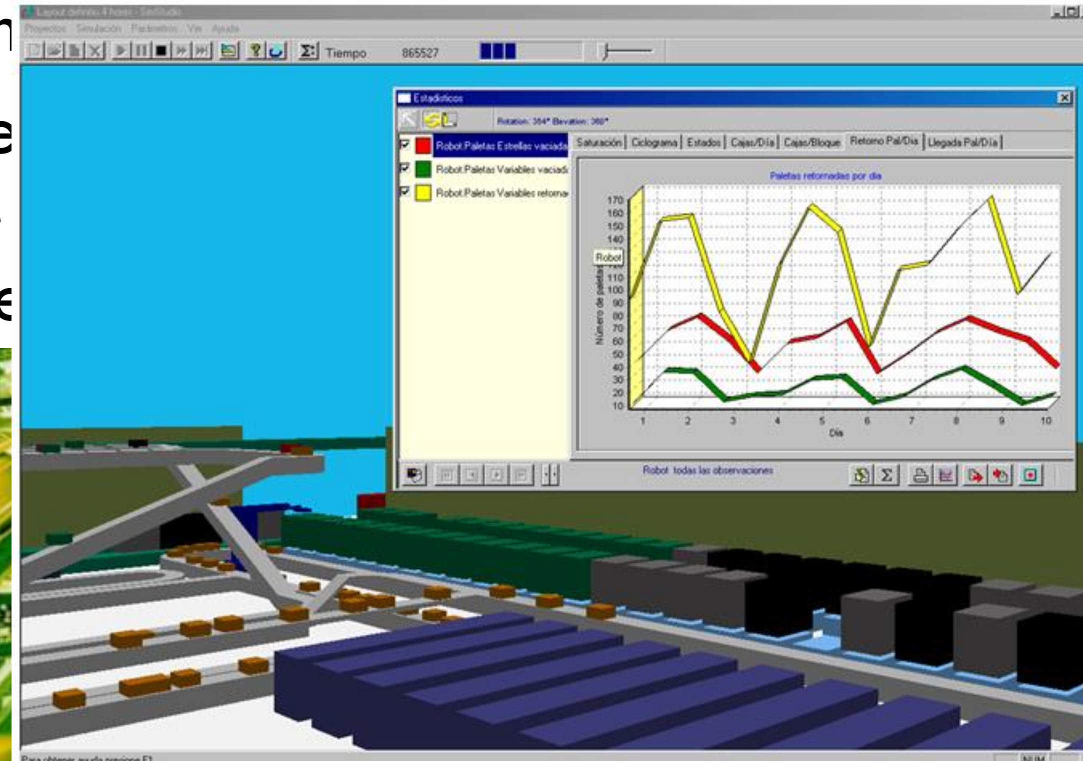
Operational validity (Calibration)

- Subjective test: Incorporate people and experience.
- Objective test: require data that represent the behaviour of the system and its equivalent generated by the model.
  - **Graphic comparison** the data of model with data from real system.
  - **Confidence interval** for the half, variances, or distributions for different model outputs.
  - **Time series** for the outputs of the model to the test if they really fit the expected.

Operational validity (Calibration)

- Subjective test: In
- Objective test: re behaviour of the generated by the





Operational validity

	Observable system	Unobservable system
Subjective test	Comparison with the data. Graphic comparison.	Explore the model.
Objective test	Comparison based on statistical studies. Graphic comparison.	Explore statistics.

Subjective test (Turing Test)

- If you can not use a statistical test, then the knowledge of people about the system will be used to compare model output with the output of the system.
    1. The simulator produces output data, the same format as the system (reports).
    2. The managers and the engineers should decide which reports are the system and which are the system model (*fakes*).
    3. It observe which is the number of detected *fakes*. The model builders ask why engineers have discovered the truth. They use this information to improve the model.
- If the engineers of the system can not distinguish between the report of simulator or the system have no evidence that the model is inappropriate.

On Boston Marathon the simulator can generate a new report of the marathon from the simulated dataset for the selected groups. One expect that no differences exist on the reports generated.

Objective test (Calibration)

- The structure of the model must be sufficiently fit to provide good predictions, not only for a dataset, but for the dataset of interest.
- At this stage, the model is treated as a black box that accepts values of input parameters and transforms them into outputs..
  - Using historical data.
  - Using the responses of the variables of interest as elements of criteria to validate the model.
  - If the system is under development must use other types of validation, for example, if their subsystems will need to use partial validation of input and output data with that sub models.

Objective test (Calibration)

- **White-Box** Validation: determining that the constituent parts of the computer model represent the corresponding real world elements with sufficient accuracy.
- This is a detailed test, or micro, check of the model, in which the question is asked: Does each part of the model represent the real world with sufficient accuracy?

On Boston Marathon analyzing if all the functions that represents the time needed for a runner to cross a segment can be considered as a linear model, hence passes Normality, homoscedasticity and Independence.

Objective test (Calibration)

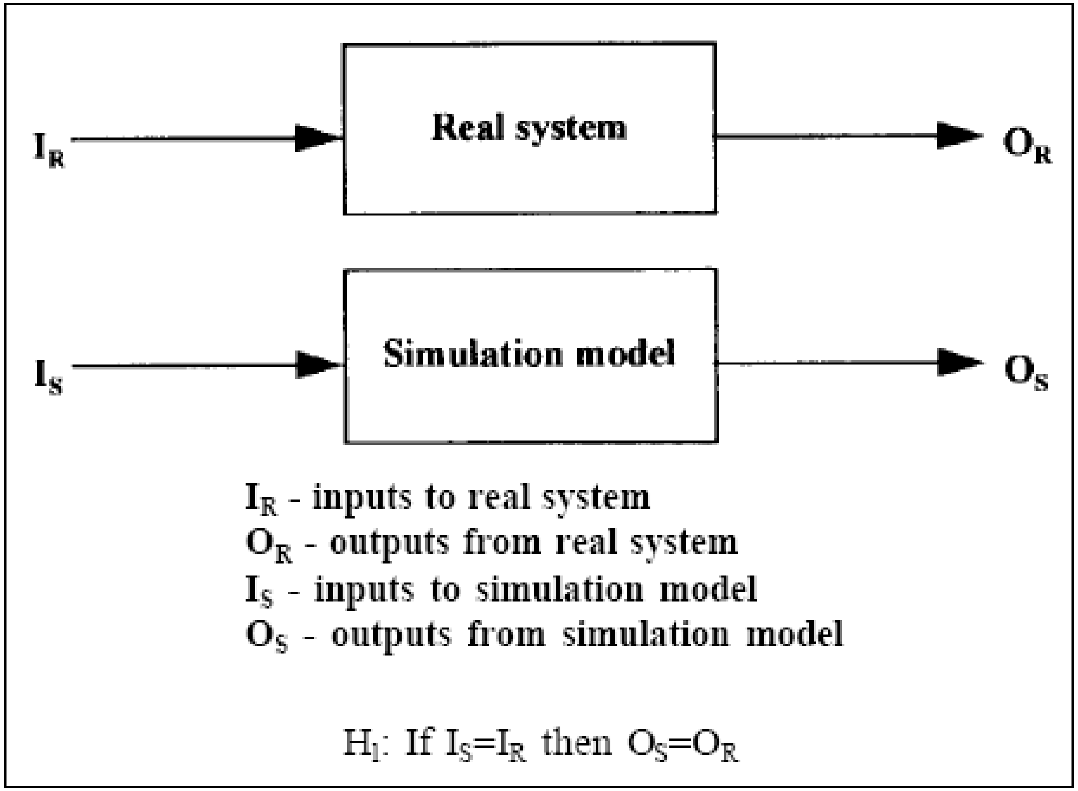
- **Experimentation** Validation: determining that the experimental procedures adopted are providing results that are sufficiently accurate.
- The important aspects to consider are:
  - the requirements for the load period.
  - the length of the executions.
  - the numbers of replications.
  - the experimental design.
  - the sensitivity analysis to assure the accuracy of the results.

As an example, on Boston Marathon this validation implies a clear justification that the experimental design selected answers the problem detected.

Objective test (Calibration)

- **Black-Box** Validation: determining that the set of the model represents the system with enough accuracy.
- This is a global test, or macro, of the form of operate of model, in which the question is asked: The model provides with enough precision for representing the system?

Objective test (Calibration)



Objective test (Calibration) Using historical data

- Do not use the GNA, using historical data.
- We hope that the model duplicates of important events that took place in the real system.
- It is important that all input data and the answers of the system have been collected during the same period.
- This technique is difficult to implement for large systems.

Objective test (Calibration)

- **Solution Validation**: determining that the results obtained from the model of the proposed solution are sufficiently accurate.
- This is similar to black-box validation in that it entails a comparison with the real world. It is different in that it only compares the final model of the proposed solution to the implemented solution.
  - The solution validation can only take place post-implementation.
  - Unlike the other forms of validation, it is not intrinsic to the simulation study itself.
  - It has no value in giving assurance to the user, but it does provide some feedback to the modeller.

Fonseca i Casas, P., Fonseca i Casas, A., Garrido-Soriano, N., Godoy, A., Pujols, W.C., Garcia, J.: Solution validation for a double façade prototype. Energies. 10, (2017). https://doi.org/10.3390/en10122013

Black-box validation example

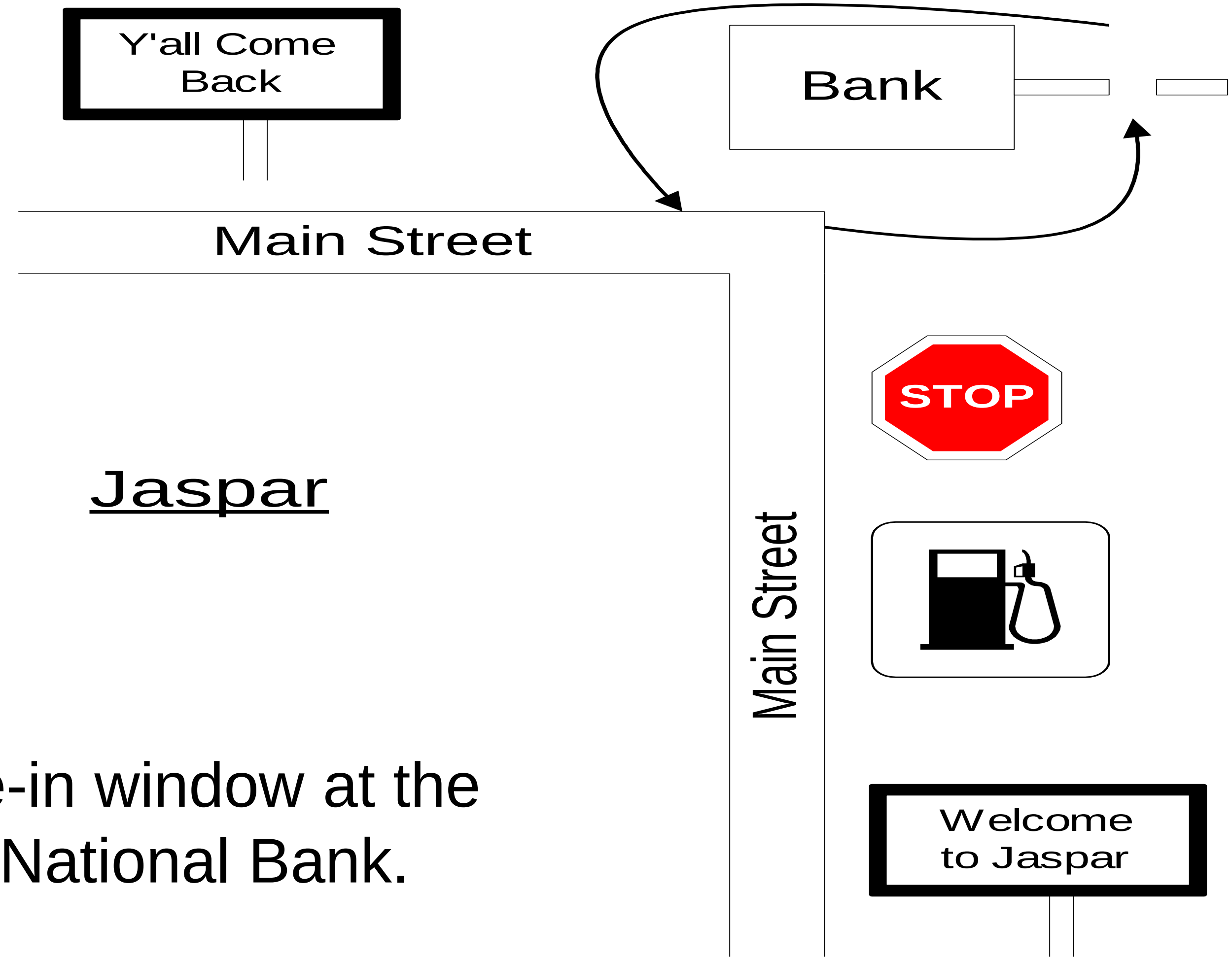
Example of objective test, using the output data.

The Fifth National Bank of Jaspar.

Black-box validation example

- The Fifth National Bank of Jaspar, is planning to expand its drive-in service at the corner of Main Street.
- Currently, there is one drive-in window serviced by one teller. Only one or two transactions are allowed at the drive-in window.
- It was assumed that each service time was a random sample from some underlying population.

Black-box validation example



Drive-in window at the Fifth National Bank.

Black-box validation example

- Service times $\{S_i, i = 1, 2, \ldots 90\}$ and interarrival times $\{A_i, i = 1, 2, \ldots 90\}$ were collected for the 90 customers who arrived between 11:00 A.M. and 1:00 P.M. on a Friday.
- This time slot was selected for data collection after consultation with management and the teller because it was felt to be representative of a typical rush hour.

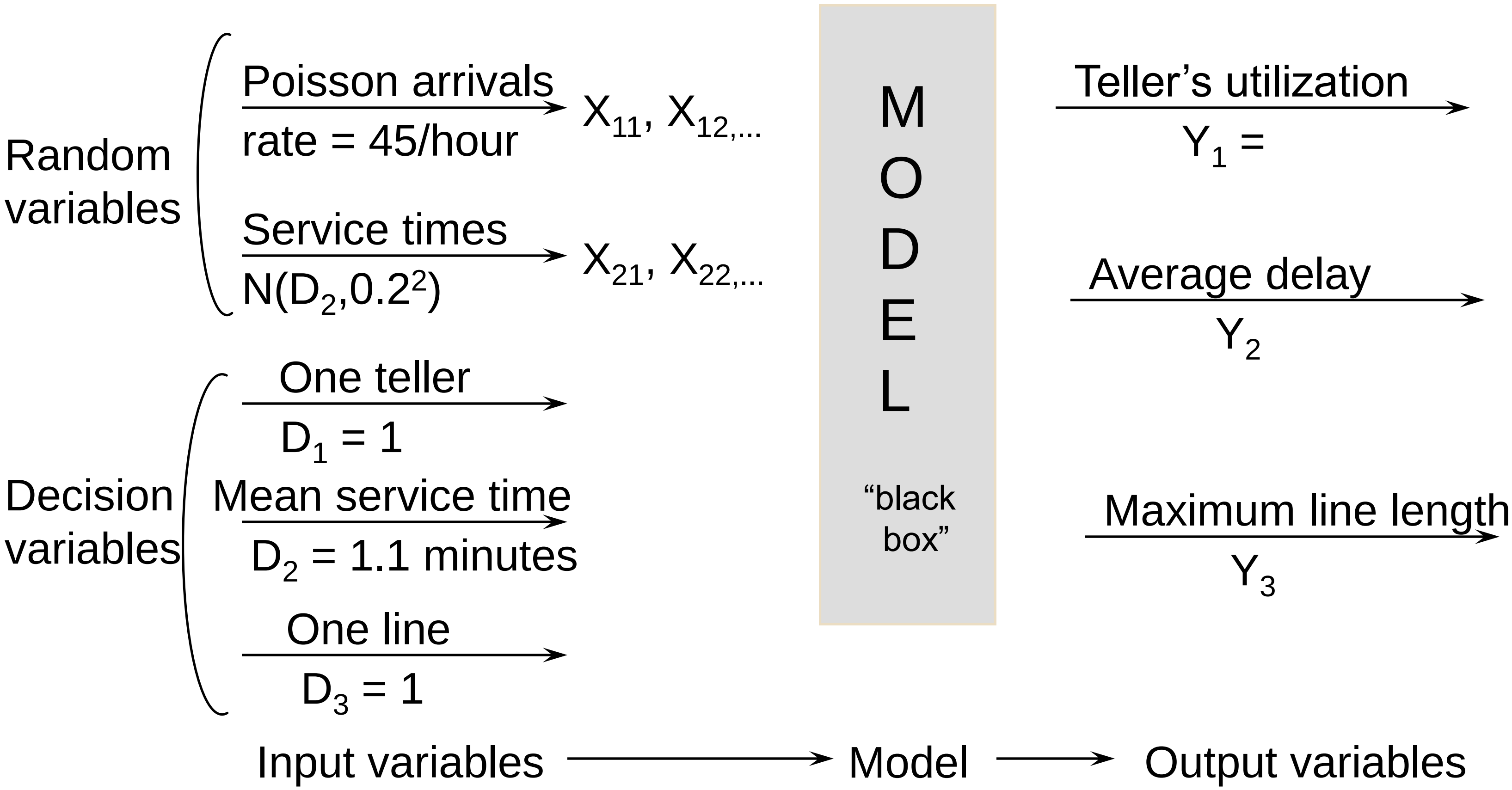
Black-box validation example

- Data analysis led to the conclusion that the arrival process could be modelled as a Poisson process with an arrival rate of 45 customers per hour; and that service times were approximately normally distributed with mean 1.1 minutes and standard deviation 0.2 minute.
- Thus, the model has two input variables:
  1. Interarrival times, exponentially distributed (i.e. a Poisson arrival process) at rate $\lambda = 45$ per hour.
  2. Service times, assumed to be $N(1.1, (0.2)^2)$.

Black-box validation example

- The uncontrollable input variables are denoted by X, the decision variables by D, and the output variables by Y.
- From the "black box" point of view, the model takes the inputs X and D and produces the outputs Y, namely
  - $(X, D) \xrightarrow{f} Y$
  - $f(X, D) = Y$

Black-box validation example



Model input-output transformation

Black-box validation example

Input variables	*Output variables, Y*
D = decision variables (interest) D_1 = 1 (a teller) D_2 = 1.1 min D_3 = 1 (a queue) X = Other variables Rate of arrivals Poisson= 45 / hour Service time: $N(D_2, 0.2^2)$	Primary variables of interest (Y_1, Y_2, Y_3) Y_1 = use of the teller Y_2 = average waiting time Y_3 = maximum length of queue Y_4 = observed rate of arrivals Y_5 = average time of service Y_6 = average time of service of sample Y_7 = mean size of the queue

Input and Output variables for model of current bank operation.

Black-box validation example

Statistical Terminology	Simulation Terminology	*Associated risk*
Type I : reject H_0 when H_0 is true.	Reject a valid model.	α
Type II : do not reject H_0 when H_0 is false.	Do not reject an invalid model.	β

Error type in the validation of a model

If the sample is fixed, the needs to reduce error of type II increases α and decreases β and inverse.
Once α has been determined, the only way to decrease β is increasing the sample.

Black-box validation example

Replicas	*Y_4 = Inputs (hour)*	*Y_5 (Minutes)*	*Y_2 =average delay (Minutes)*
1	51	1.07	2.79
2	40	1.12	1.12
3	45.5	1.06	2.24
4	50.5	1.10	3.45
5	53	1.09	3.13
6	49	1.07	2.38

Average: 2.51
Deviation : 0.82
Results of six replicas of the model bank

Black-box validation example

- Delay observed in the system $Z_2 = 4.3$ minutes.
- Delay of the model Y_2.
- We propose a statistical test of null hypothesis
  - $H_0 : E(Y_2) = 4.3$ minutes
  - $H_1 : E(Y_2) \neq 4.3$ minutes
- If H_0 is rejected following the rules of this test, there is no reason to consider the model invalid.
- If H_0 is rejected, the current version of the model can be rejected, and the modeler is forced to seek ways to improve the model.

Black-box validation example

- The appropriate statistical test is *t*, which is conducted as follows:
  - ***Step 1***. Select the level of significance a, sample e and size n. For the bank model:
    - a = 0.05, n = 6
  - ***Step 2***. Calculate the mean of Y2 and standard deviation S on these *n* replicas.

$$Y_2 = \frac{1}{n}\left(\sum_{i=1}^{n} Y_{2i}\right) = 2.37 \quad S = \{(Y_{2i} - Y_2 / (n-1)\}^{1/2} = 0.82$$

  - Where Y2i, i = 1, .., 6, are shown in the above table.

Black-box validation example

- ***Step 3***. Getting the critical value t of the table.
  - For a test of two queues, must use $t_{\alpha/2,\, n-1}$; for a test of one queue must use $t_{\alpha,\, n-1}$ or $-t_{\alpha,\, n-1}$.
  - n -1 are the degrees of freedom.
  - From the table $t_{0.025,5} = 2.571$ for a test of two tails.

Black-box validation example

- ***Step 4.*** Calculate the statistic
  - $t_0 = (Y_2 - \mu_0) / \{S / \sqrt{n}\}$
  - on μ_0 is the specific value of the null hypothesis
  - H_0 . Where $\mu_0 = 4.3$ minutes, so
    - $t_0 = (2.51 - 4.3) / \{0.82 / \sqrt{6}\} = - 5.34$
- ***Step 5.*** For a test of two queues:
  - if $|t_0| > t_{\alpha/2,\, n-1}$, reject H_0 .
  - Otherwise do not reject H_0.
  - [For a test of one queue with H_1: $E(Y_2) > \mu_0$,
    - reject H_0 if $t > t_{\alpha,\, n-1}$; with H_1 : $E(Y_2) < \mu_0$,
    - reject H_0 if $t < -t_{\alpha,\, n-1}$]

Black-box validation example

- Since $|\ t\ | = 5.34 > t_{0.025,5} = 2.571$, must reject H_0 and conclude that the model is not suitable in their prediction for the average delay for a client.
- Note that when you are making a hypothesis test, reject H_0 is a strong conclusion, so
  - $P(\text{reject } H_0 \mid H_0 \text{ is true}) = \alpha$

Black-box validation example

Replicas	*Y_4 = Inputs(hour)*	*Y_5 (Minutes)*	*Y_2 =average delay (Minutes)*
1	51	1.07	5.37
2	40	1.12	1.98
3	45.5	1.06	5.29
4	50.5	1.10	3.82
5	53	1.09	6.74
6	49	1.07	5.49

Average: 4.468
Deviation: 1.66
Results of six replicas of the model bank

Black-box validation example

- Step 1. Select $\alpha = 0.05$ and $n = 6$ (sample size).
- Step 2. Calculate $Y_2 = 4.468$ minutes, $S = 1.66$ minutes.
- Step 3. Calculate the critical value of t.
- $t_{0.025,5} = 2.571$.
- Step 4. Calculate the statistic

  $$t_0 = (Y_2 - \mu_0) / \{S / \sqrt{n}\} = 0.247$$
- Step 5. Since $| t | < t_{0.025,5} = 2.571$, cannot reject H_0, and can "tentatively" **accept the model as a valid.**

Objective test (calibration)
If the system does not exist

- The model can be used to represent the behaviour of systems that do not exist:
  - Not yet been built.
  - Alternative of system design.
- If some version of the system is operational and has been validated, the validity of the model system that does not exist can be evaluated from a model of the old system.
  - The responses of the two models under similar entries can be used as criteria for comparison.

Objective test (calibration) If the system does not exist

- If the proposed system is a modification of the existing system, changes that can be made are:
  - Minor changes in numerical parameters: # of servers.
  - Minor changes in probability distributions: service time.
  - Major changes in the logical structure : schedules.
  - Major changes including different designs of the new system.

As an example, on Boston Marathon one can change the number of servers and define a model that is not going to work to assure that the structural properties of the model works.

VVA

Verification

Verification

- Verification is the process of comparing the program with the model and its behavior with the real system.
- Constructing the model correctly.
- Debugger.

Verification

- Common engineering techniques of software, in particular:
- **Static tests**: It looks at the structural properties of the code to evaluate whether really correct.
- **Dynamic tests**: The program runs under different initial conditions to see if it really works as expected. The results obtained are used to determine if the implementation is correct or not.

Static tests

- Structured walk-through.
- Examine structured properties.
- Correctness proofs.

Dynamic tests

- Approaches: Bottom-up, top-down, combined.
- Techniques: Traces, input and output relations, directions of change, amount of change.
- Large numbers.

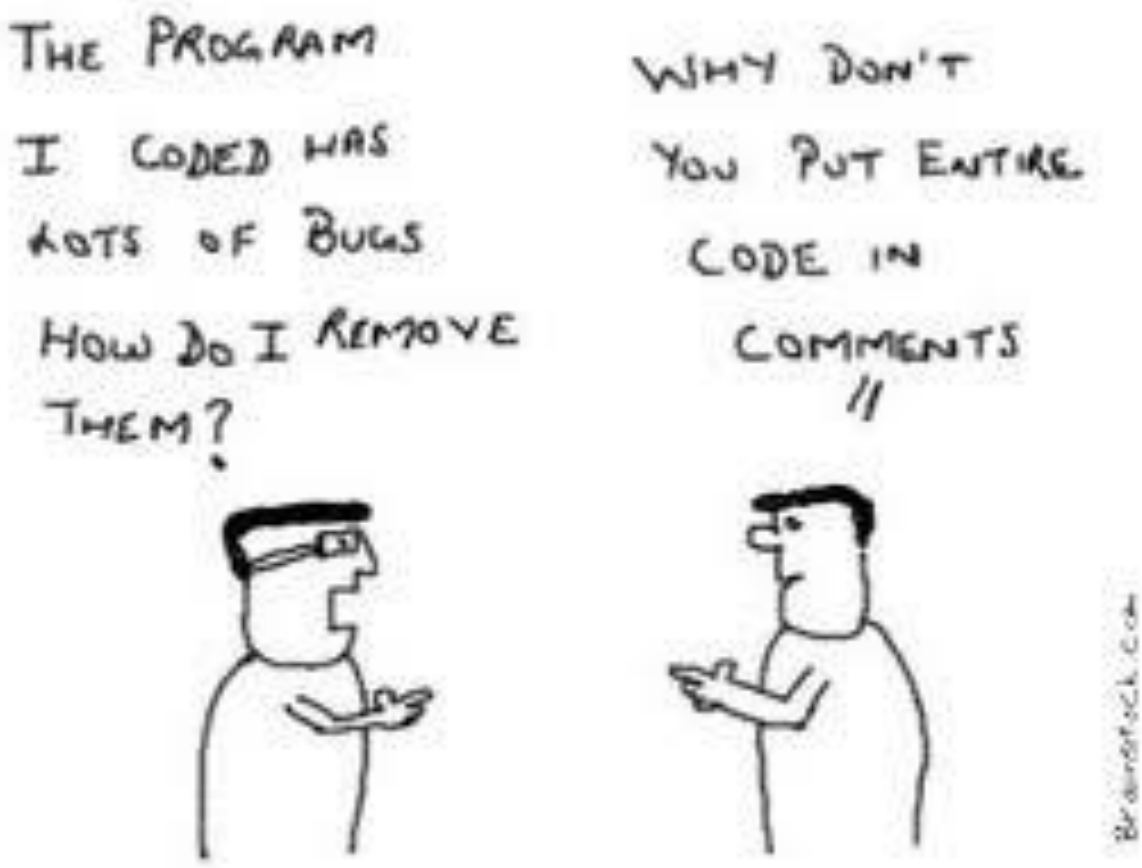
Verification of simulation models

- Tips to follow to simplify the verification process (These suggestions are basically the same as any programmer must follow in order to debug a computer program):
  1. That is someone different than the programmer who validates the model.
  2. Creating flow charts that include every possible action that the system can take before an event. Following the logic of the model for each share of each type of event.

Verification of simulation models

1. Examining in detail the output model for a reasonable set of input parameters. Having the code to print a different set of statistics.
2. Allowing the printing of the parameters at the end of the simulation, ensure that these parameters have not changed inadvertently.
3. Make the code self-documented. It provides a precise definition of each variable used and a general description of the purpose of each major section of code.

Remember to comment the code



VVA

Accreditation

Accreditation

- Accreditation is an official determination that the model is acceptable for a particular purpose.

Issues to consider

- The contracting person must understand and assume model hypotheses.
- Demonstration that the model has been V&V.
- The contracting person must be the owner of the model and become involved in the project.
- A compelling animation (Sargent).

Issues to consider

- The final presentation must include animations and a discussion about the validation/verification process and the construction of simulation model.

Methods to demonstrate the model

- Regular meetings with clients.
- Develop and maintain document of hypotheses (DH).
- Promote that all active parties of project are participate an active role.

Regular meetings with the client

- Lets see if the main problem has been resolved.
- Keep the customer's interest in the project.
- Increase the credibility of the model.
  - The client understands and accepts the hypotheses.

Document of hypotheses(DH)

- It must be developed to top jointly with the client.
- Need not be an exhaustive description of how the system works, but a description on how you want to solve.
- Must continually modify the meetings with the client.

Components of the document (DH)

- Objectives, problems, performance measures.
- Interaction of subsystems.
- Hypotheses.
- Limitations of the model.
- Data.
- Sources of information related to the project.

Promoting the participation

- Calendar of events.
- No one has ALL the information the system!

  Ask each person their value for the good development of the project.
  - Remember MODULARITY of formalisms, use it.
- Incentives, awards… (better than punishment).

Finally

- The accreditation must be headed by a different third team of the contracting team of simulation and the team responsible for developing the simulation.
  - The client has been involved in the developing.
- More information https://www.msco.mil/