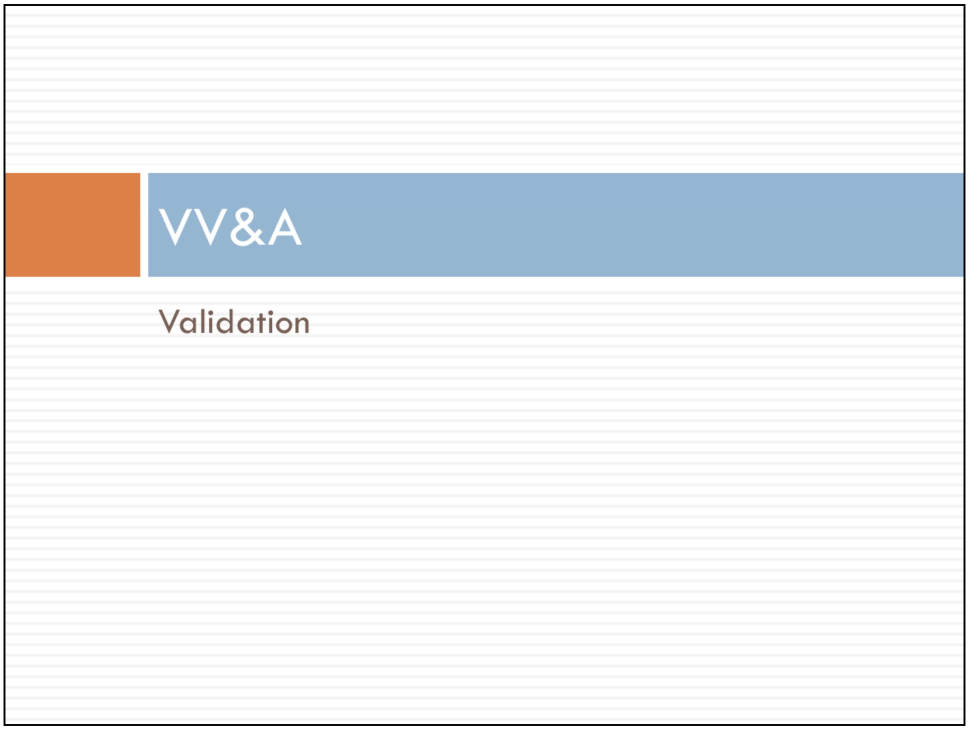


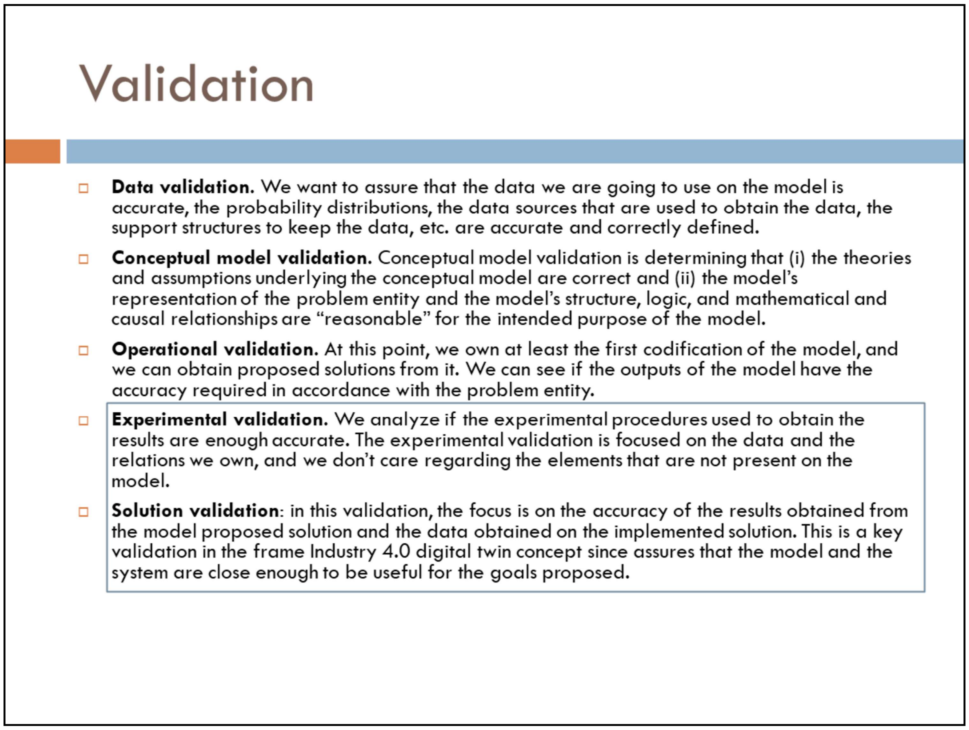
Let’s start with validation. Here we will describe the more common techniques one can apply to perform a Validation of a simulation model.

Validation

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

First it is needed to remark that the behavior of the model is correct as a consequence to the validation process. The Validation process often relies on the comparison of the model behavior with the behavior of the real system.

The question we want to answer is, we build the correct model?, or similarly, are the assumptions that build the model correctly used?



There are several aspects that can be validated.

First is data validation, where we will work with the data to be used on the model. The main concern is to assure that the data we are going to use on the model is accurate, accurate in different aspects. First that the probability distributions used fits well with the observation we have. Also to assure that the data sources that are used to obtain the data, that is the companies involved on the data acquisition and maintenance are working well, you have confident with them. Also to analyze that the support structures to keep the data, and to maintain the data are accurate and correctly defined. Here often is needed to define a time, when the data will no be correct, like a caducity time.

Second is the Conceptual model validation, that is determining, in one hand, if the theories and assumptions (simplification and systemic assumptions) underlying the conceptual model are correct and, in the other hand, if the model's representation of the problem entity (that is the problem we want to solve) and the model's structure, that is the logic and the mathematical and causal relationships that compose the model, are "reasonable" for the intended purpose of the model.

Third, we will be focused on Operational validation. At this point, we own at least the first iteration of the model codification and we can obtain proposed solutions from it. Since we have this codification of the modes, one can see if the outputs of the model have the accuracy required in accordance with the problem entity. Here well-known statistical tests can be applied to understand the model accuracy.

Next, we have two validations that often are being forgotten but are gaining more relevance because the general use of simulation in the Industry 4.0 context.

First is Experimental validation. On it 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.

Finally, the Solution validation where we will be focused on the accuracy of the results obtained from the model proposed solution and the data obtained on the implemented solution. Notice that we will be focused on the implementation, hence on the modifications we do on the system following our model recommendations. 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.

To perform this validation process we must follow something similar a template. First we will identify the aspects to validate, the data, the conceptual model, that encompasses the logical structure and the hypotheses of our model, and the Operational validity, that is the results we obtain from the model codification, understanding that this is focused in solve an specified problem entity.

To do all this validations it will be interesting to use some representation techniques. This will help looking the model and understanding that the behavior we see is similar to the behavior we see on the system.

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**.

To cope with all this validations, Naylor and Finger proposed an approach to perform the validation of a simulation model. First, we will start building a model that seems valid. On this model is not needed to include all the needed detail, remember that modeling is an iterative process, but we must enough to start the analysis. First, we review the conceptual model with the experts, to understand that the model assumptions seems correct. Once we do this, that is the validation of the conceptual model, we will analyze if the data we are using in our model is valid, that is the validity of the data.. Finally we can compare the model outputs with the system data, that is the Operational Validity. Remember that being an iterative process, we will continue doing this validations until we achieve a consensus with the stakeholders, that imply that we believe on the model we own.

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.

There are plenty of different techniques that can be used to do a validation. Here we have a subset of some of them. Please, refer to wintersim.org website to find papers that details some of this techniques.

Validity of the data

Ensure that the data of the model used correctly

Now we will be focused on the techniques we can use to perform the Validity of the data.

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

The Data Validation is determining that the data required for model building, validation and experimentation are sufficiently accurate. As one can imagine, and because without data the simulation models to do previsions can be at least risky, this applies to all aspects of the modelling process, since data are required at every stage of a simulation study. See Transaction 69 of the Journal Simulation by Stewart Robinson.

We will be focused on checking that the data transformations are correct; it means to take care of the data cleansing we can do and of the possible modifications to the data structure to avoid to lose of modify accidentally any information.

Other interesting aspect that often is forgotten is the data expiration. The data often owns a time when will become invalid, we must analyze this to assure that the analysis are not using outdated datasets.

Also see related to this

Type of data

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

We must also consider the type of the data we will use that can be different depending on the status of the model development, often the data for the model construction the data for testing, and the data to conduct the experiments will be different, hence we need to assure the validity of the data for each one of these scenarios.

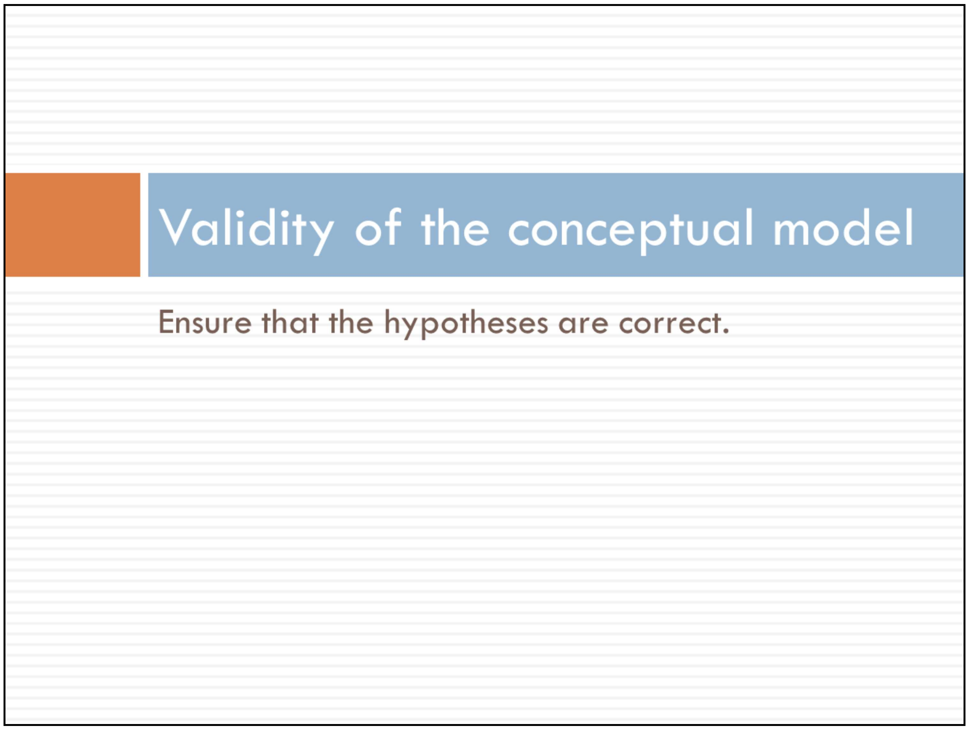
Methods

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

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

Regarding the methods to do the data validation, there are strongly related to the statistical methods one can know, first will be needed to assure that the methods we are using to obtain the data are correct. We must test the data the internal consistency using statistical techniques. We must also consider the procedures needed to keep the data correct enough for the experimentation, also what will be the structure of the databases to be used.

Finally, as we mention, detect when the data is going to become outdated. As an example in a wildfire, every years, at least, we must update the database that refers to the soil composition, since this soil can contain material that will be susceptible to start a wildfire, and this soil changes from year to year.



Now we will review the validity of the conceptual model.

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?

Conceptual Model Validation is focused on determining that the scope and level of detail of the proposed model is sufficient for the purpose of the problem entity, and that any assumptions are correct. The question being asked is: does the conceptual model contain all the necessary details to meet the objectives of the simulation study?, See Transactions 69 by Stewart Robinson

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.

Data assumptions should be based on the collection of reliable data and correct statistical analysis of the data, while the conceptual model validation must rely on the analysis of the model assumptions, how the system operates, and the tacit and implicit and explicit knowledge of the experts on the system.

At this stage, we must evaluate each sub model regarding the structure logic and the causal relationships detail versus the aggregation level we apply to our analysis.

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).

One of the validation techniques one can use here is the face validity, where the experts will be asked about the behavior of the system, and to acknowledge that the model behavior is reasonable in comparison with this system behavior understanding.

Validity of the conceptual 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.
- The analysis of the Systemic Structural assumptions are based on the formal definition of the model.

The analysis of Systemic Data assumptions will consist of three steps:

1. Identifying the appropriate probability distribution

2. Estimating the parameters of the hypothesized distribution

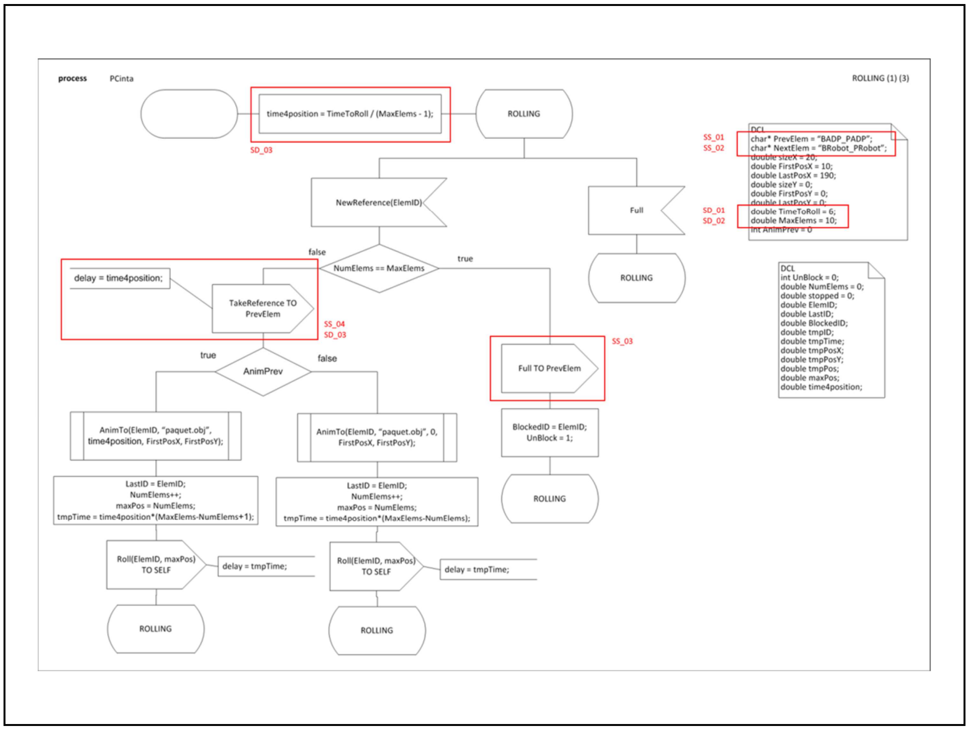
3. Validating the assumed statistical model by a goodness-of fit test, such as the chi-square or Kolmogorov-Smirnov test, and by graphical methods.

The analysis of the Systemic Structural assumptions are based on the formal definition of the model.

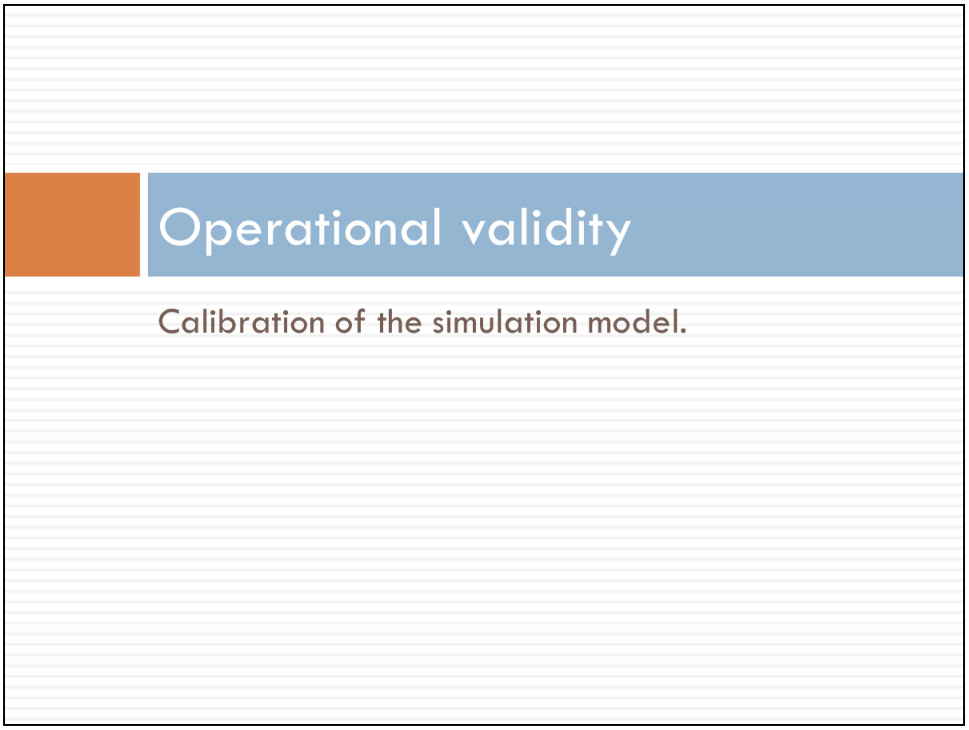
Validity of the conceptual model (Example)

- 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 $\frac{TimeToRoll}{(MaxElems-1)}$.

This is an example of a conceptual model validation. On the slide are presented the assumptions on the hypotheses document that defines how the system works. This document must be assumed by the experts on the system and the Stakeholders. Notice that each one of the assumptions owns a code, to identify faster them. Also, the code depends on the type of the assumption, where SS means Systemic Structural, SD, systemic Data assumption. A simplification assumption can be represented by SM.



All model assumptions will be represented explicitly, or implicitly in the case of simplification assumptions, in the model diagrams. In this case we present a model represented using Specification and Description Language, and we remark where the assumptions of the previous slide are located and represented.



Operation validity can be applied once we have a codification of the simulation model, here we will be focused on understand if the model outputs are coherent with our assumptions.

Operational validity (Calibration)

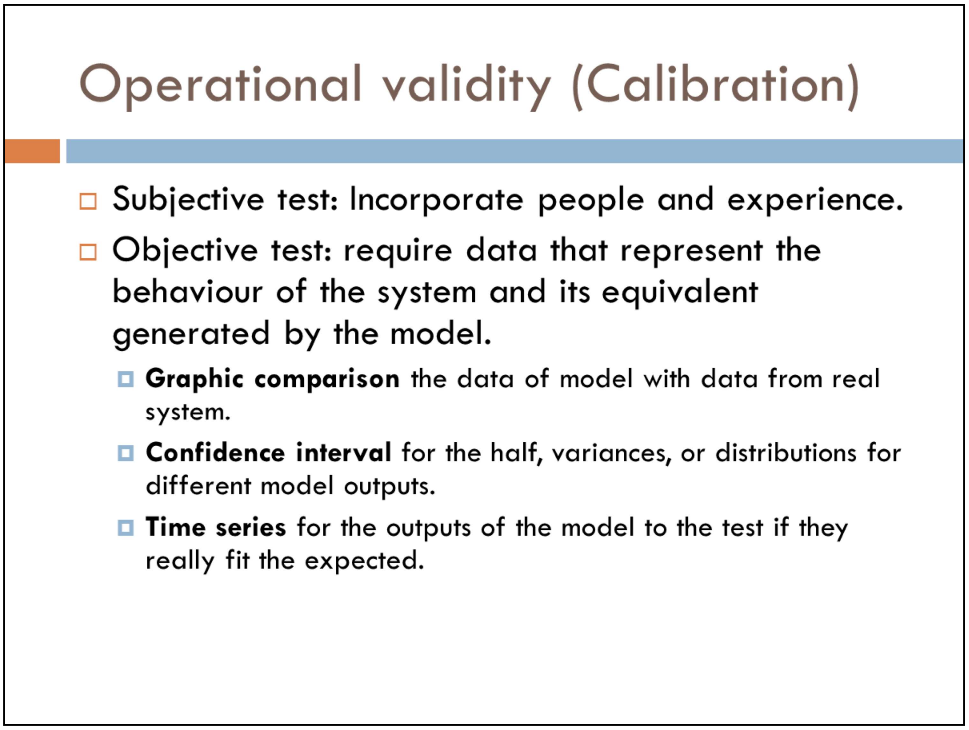
- The objective of the test is to confirm the ability of the model to predict the behaviour of the real system.
- Iterative process of comparing the model and the real system: make adjustments in the model and compare the new model revised.
- Must collect over a set of system data.
- Trade-offs: cost/time/effort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

An operational validation the goal is to confirm the ability of the model to predict the behavior of real system, in this case we are going to use the data that is obtained through the simulation execution. At this point it is needed to own codification of the model. This is an iterative process; we will compare the model outputs with the data that we will obtain from the real System. To do so we must collect real data from the System. Notice also that as any other validation processes, this is a trade off between the time needed to do the actions, and the time we have to perform all the validations on the model

Operational validity

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

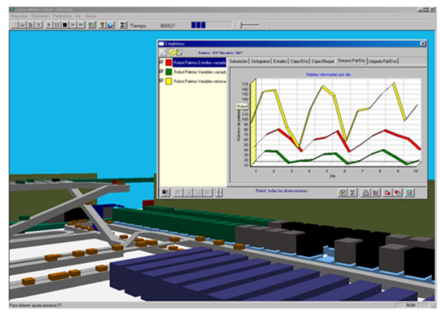
On operational validity there is also a lot of different techniques that can be Applied. the selection of the technique depends on the model and on the goal that we are looking for. Hence, the experience will guide what are the techniques that we will apply in order to increase the confidence on the model.



One of the test typologies that one can use are the subjective tests. On this type of test we will be focused on the knowledge that the personnel specialized on the system have.. We will incorporate people and experience on the validation process.

Other type of tests that we can apply are the Objective test. On these typology of tests we require data, that represent the behavior of the system, and its equivalent generated by the model. With these two datasets we can do a graphic comparison of the data of model with data from real system. We can calculate confidence intervals for the half, variances, or distributions for different model outputs. And see if these intervals fits with the real data, and we can analyze time series for the outputs of the model to the test if they really fit the expected.

Operational validity (Calibration)
Graphic comparison

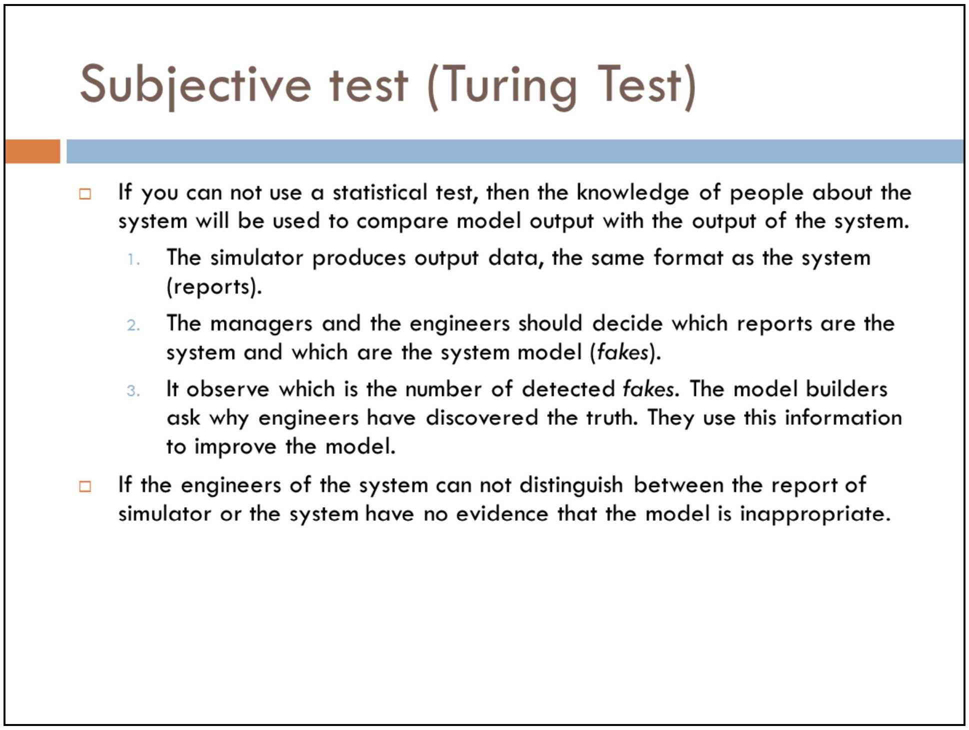
System	Model
	

On this slide we can see a graphical comparison example. On the left side is represented the real system, with the conveyors that moves the different boxes along the factory, on the right side, the simulation model, that using a virtual representation allows to perform this graphical comparison. The graphical comparison doe not need a virtual reality of the simulation model but needs a representation good enough to see if the model behaves as expected.

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.

On this table we can see when to use of the different validation techniques depending on the model nature, if the model is observable or not. An unobservable system can be as an example a system that does not yet exists.

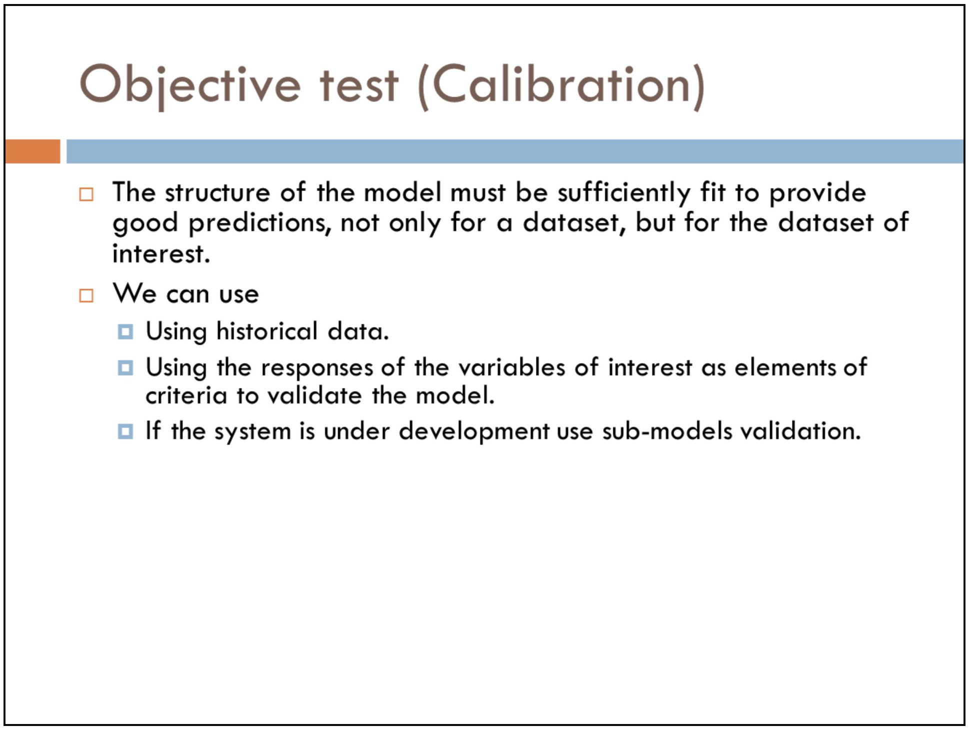


The main idea of the Turing test is to see if the experts are able to detect if the data is generated by the system or by the model.

Then, the simulator produces output data, following the same format as the system (reports). With this information the managers and the engineers should decide which reports are the system and which are the system model (fakes).

If the specialists detects a high number of model reports implies that the model builders discovers the model behavior that is different from the system. We can use this information to improve the model.

On the other hand, 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 Objective tests we will use data to understand if the model behaves correctly. No subjective evaluation can be done on this approach.

The structure of the model must be sufficiently fit to provide good predictions, not only for a dataset, but for the dataset of interest, hence remember that the dataset used to model, and the dataset used to this the model must differ.

At this stage, the model is treated as a black or while box that accepts values of input parameters and transforms them into outputs..

Some techniques that can be used are

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, we must use other types of validation, for example, analyze their subsystems and perform a 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?

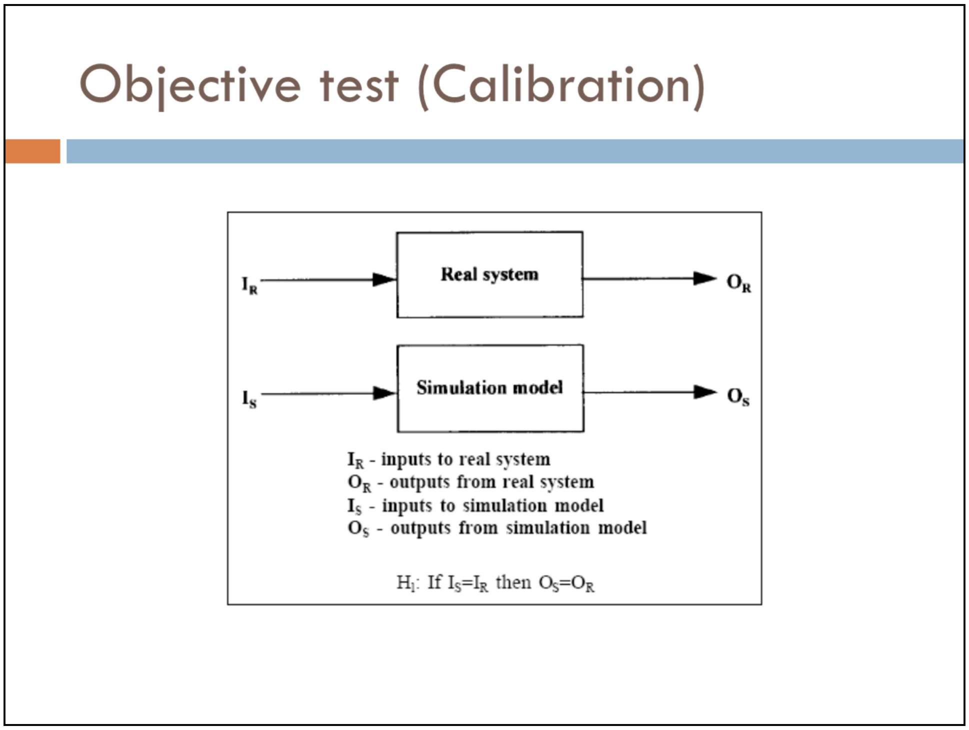
White-Box Validation will be determining that the constituent parts of the computer model represent the corresponding real-world elements with sufficient accuracy. This is a detailed, or micro, check of the model, we analyze any part of the model component, in which the question is asked: does each part of the model represent the real world with sufficient accuracy?

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 in which the question is asked: The model provides with enough precision for representing the system?

On Black-Box Validation we will determine that model, as a whole, represents the system with enough accuracy.

This is a global test, or macro, focused in the form of the model operates, in which the question that is asked is: The model provides with enough precision for representing the system?



On a black box, the input and the outputs of the model are the only elements we know. We will use the outputs to compare with the system data, depending on the scenario defined on the inputs.

Objective test (Calibration)
Using historical data

- Do not use the GNA, using historical data.
- We hope that the model duplicates the 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.

We can use historical data to perform this comparison,. In that case can help do not use the GNA, using historical data also as an input.

Doing so we expect that the model duplicates the important events that took place in the real system.

To do so It is important that all input data and the answers of the system we want to validate, have been collected during the same period.

Due to the nature of this technique, it will be difficult to implement for large systems due to the huge and the diversity of the data that will be involved during the validation process.

Experimentation validation

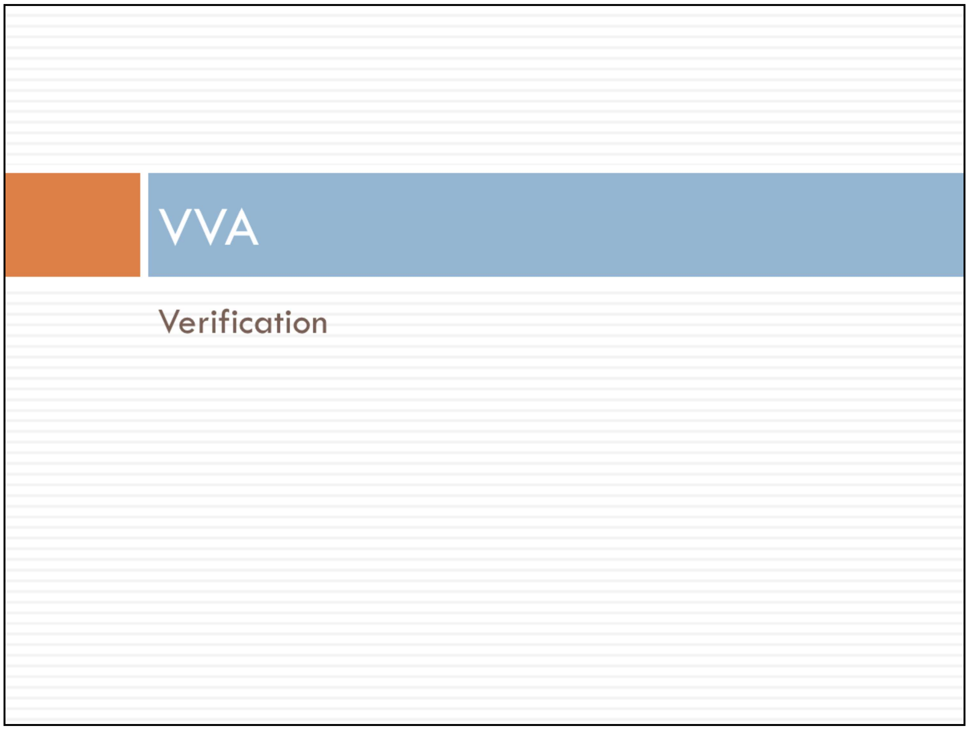
- **Experimentation** Validation: determining that the experimental procedures adopted are providing results that are sufficiently accurate.
- The important aspects to consider are:
  - the requirements for the load period.
  - the length of the executions.
  - the numbers of replications.
  - the experimental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

Finally just to mention that Experimentation Validation will be : determining that the experimental procedures adopted are providing results that are sufficiently accurate. Key issues are the requirements for warm-up, run length, replications, experimental design and sensitivity analysis to assure the accuracy of the results.

Solution validation

- **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



No we are going to review verification and the techniques that we can apply.

Verification

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

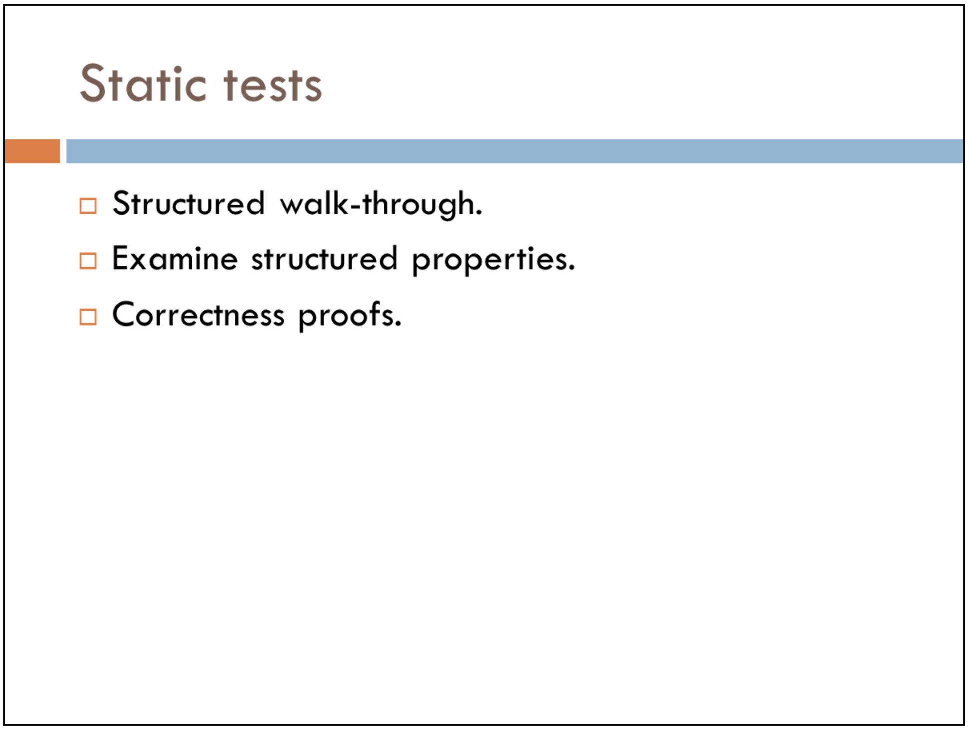
Verification will be the process to compare the program that we are obtaining through the codification of our model, with the data that we can obtain from the real System. the question that we're going to answer is, are we doing the codification correctly?. As you might imagine this process will be related to debug.

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.

Here you can use common software engineering techniques to the deck is the code that we develop is correct or not. You can use top down approaches, you can use modularity to enforce the verification of each one with different individual components, or any other so in general you can apply any technique that help you to assure that the codification has been done currently.

More specifically, we can apply static tests, where we are going to look the structural properties of the code, To detect if this codification has been done correctly, or we can apply dynamic test, where we are going to execute the program that we obtain throw the codification to see if it behaves as expected. in that case we are going to compare the results we will obtain with the results we expect to find.

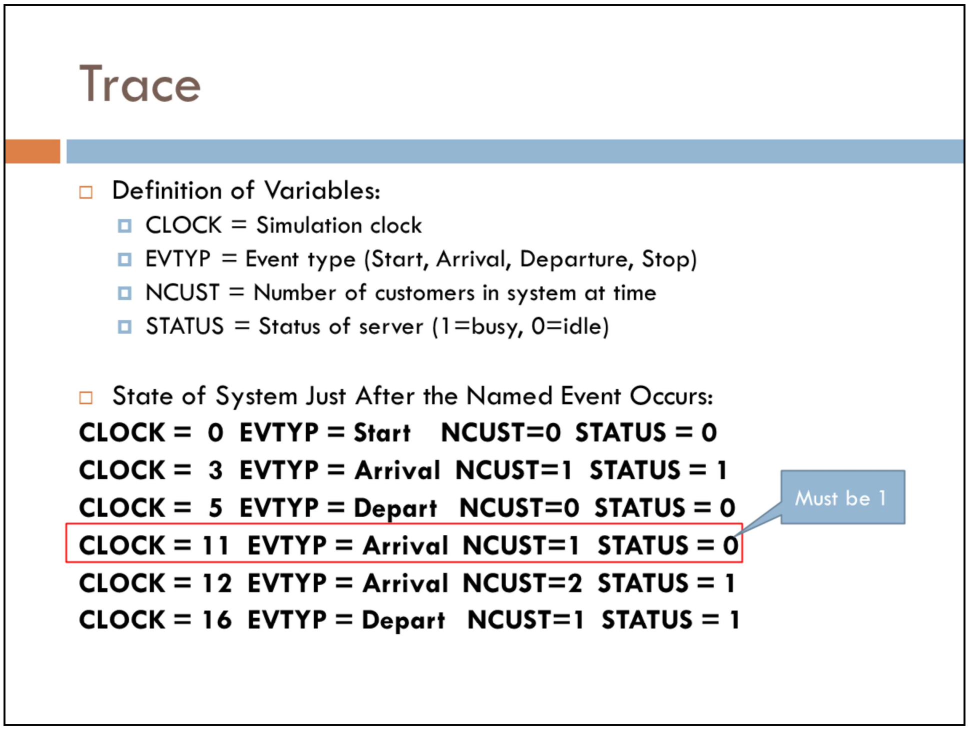


There are several techniques that can be applied force that test like structured walk-through, we can examine the structural properties of the code and also, we can apply correctness proof.

Dynamic tests

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

Regarding the dynamic test we can focus several approaches, like bottom up top down, we can combine both approaches, we can use also a lot of different techniques here like traces, input and output relations, we can analyze the direction of the change of the results that we are obtaining and also, we can see the amount of change that we are obtaining through the modification of the different factors that we are using in our models.



At this point one can use simulation traces to compare the results, and to analyze if the logic of the events are coherent with the understanding the experts have of the system.

As an example, here we have a simulation trace of a simulation model.

Notice at level eleven we are expecting that the status of the server is one, because the server is doing something, but in that case, we are looking at the value is zero. It means that there is a problem with the codification on this model.

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.

There's some things that we can follow to simplify the verification process the suggestions are mainly the same suggestion so we can provide to a programmer that is going to debug a computer program. It is really interesting that someone different to the programmer test the model, is quite common that for you the program is working well but when someone order is trying to execute the program it crashes. Other interesting recommendation is to create flowcharts, that include every different possible action that the system is going to do; following this logic is going to provide you information regarding if the program is following the correct paths. as you might imagine at this point the formal representation of the model is going to help largely on these verification process.

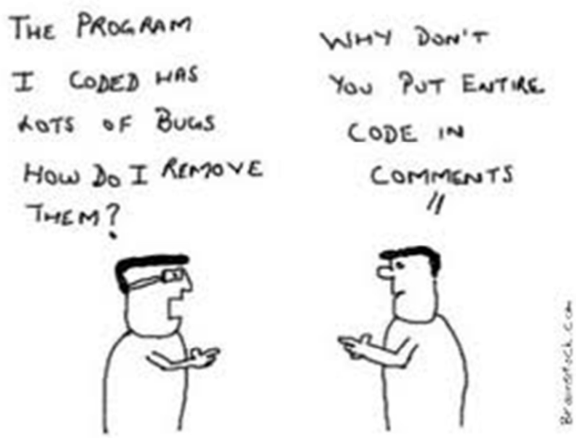
Verification of simulation models

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.

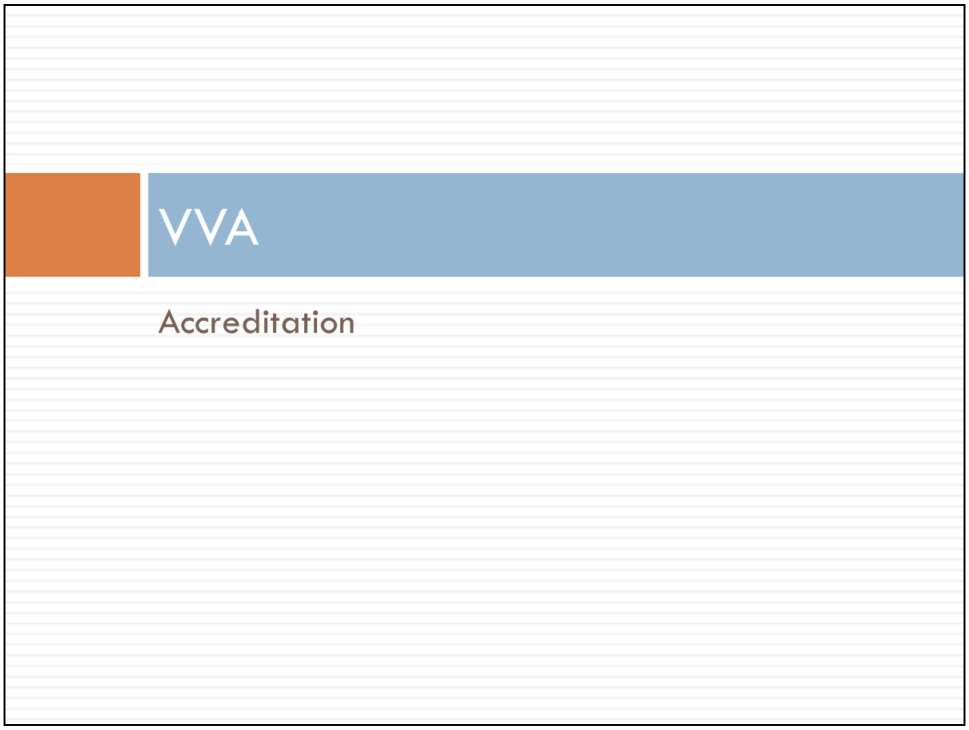
Other recommendations that you might follow is to examine in detail the output of the model for a well-known set of input parameters. You will be able to test if resource that you're obtaining are querying with the expected results. do the show it will be needed to have pieces of code to print the different preliminary statistical information that you will obtain.

other interesting good practice, is to print the parameters at the end of each Simulation. This will help to understand and to compare the different outputs. finally it could be really important to make the code self documented this will provide precision formation of the different variables and elements that you are using, and also remember to add a general description of the propose of each major section of code.

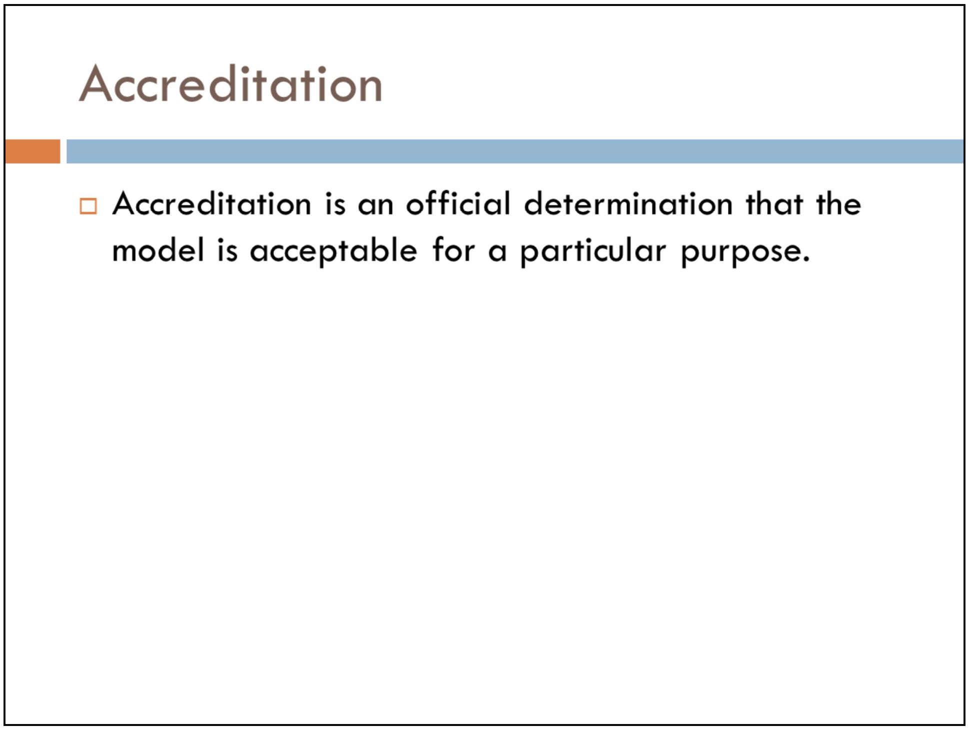
Remember to comment the code



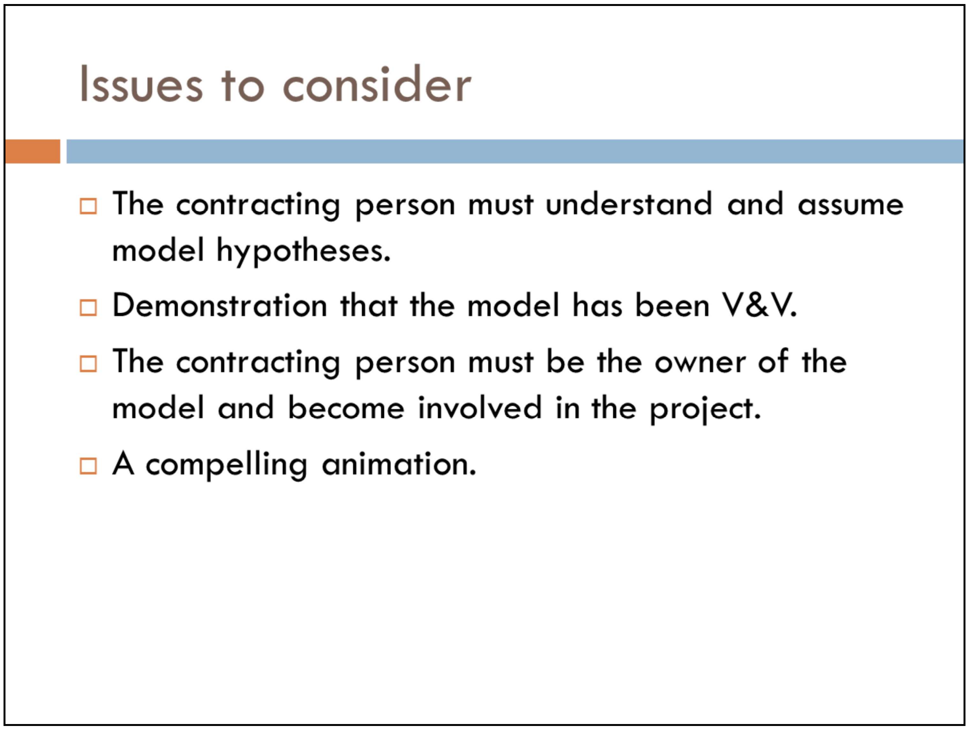
And as a final recommendation remember always the comments on your code.



Finally we will discuss regarding the last steep a successfully simulation project must achieve, the Accreditation.



Accreditation means that someone believes on the simulation model, hence this model can be used for decision making purposes. As one can imagine, without this accreditation the model is not useless, since in the process to define the model you generate knowledge regarding the system, and maybe the first goal of define your simulation model is this, understanding how the system behaves. But it is clear that without the Accreditation, the simulation model is not going to be used to obtain previsions regarding how the system will behave that will be credible enough to imply some implementations on the system.

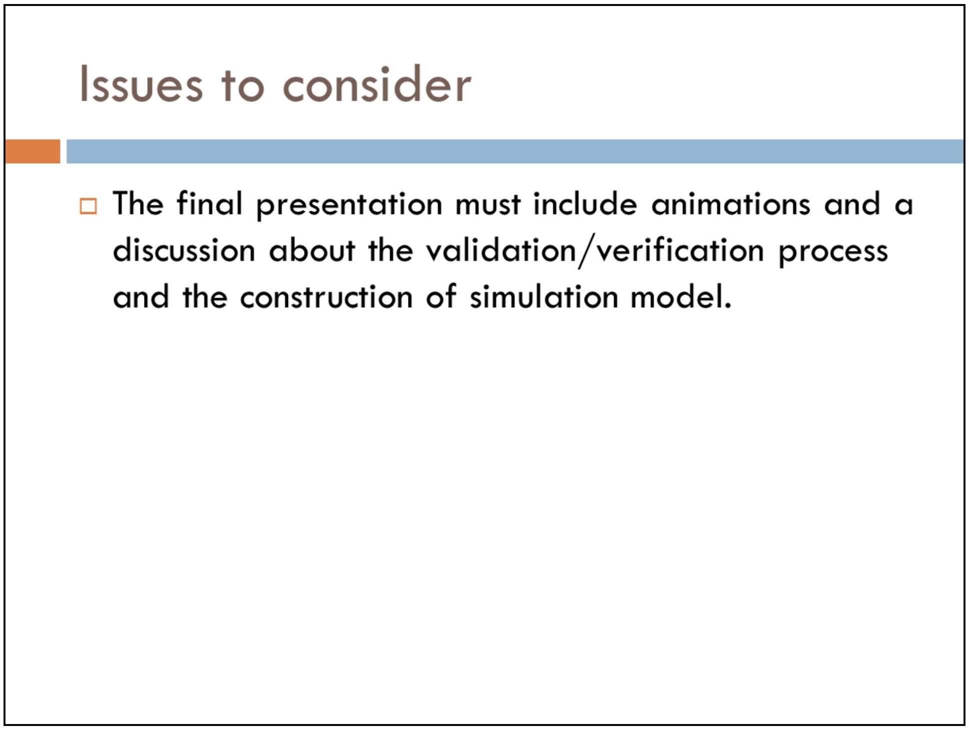


There are some issues one must consider, first is that the contracting person, usually the Stakeholders, must understand and assume model hypotheses. Without this, they can sink the model from its basis, the assumptions.

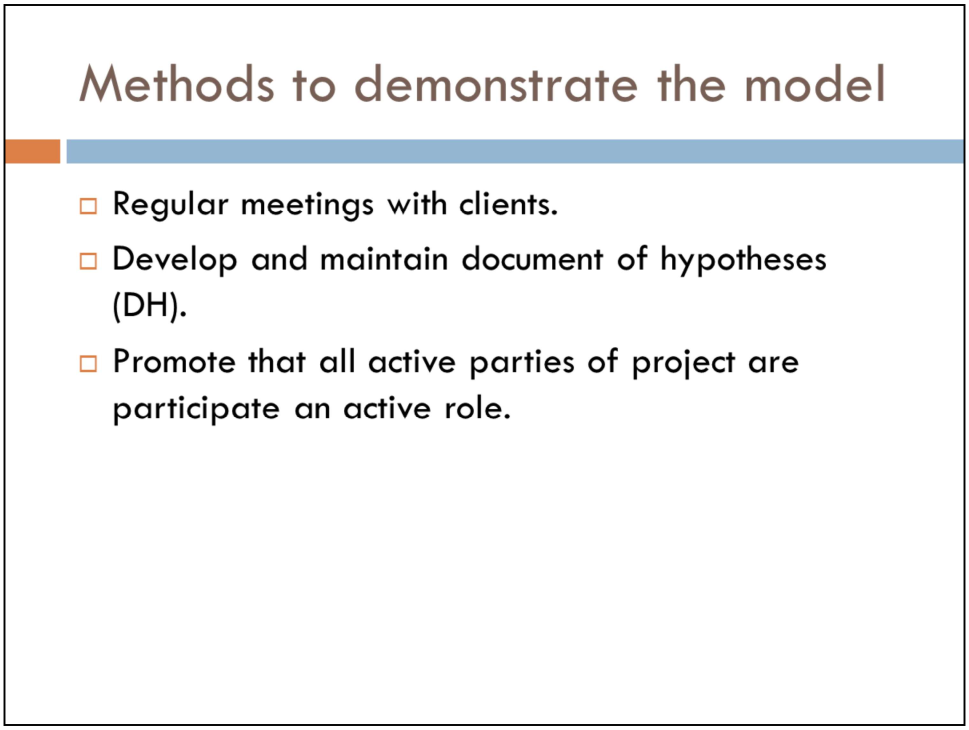
It is needed to assure doing demonstrations, that the model has been Validated and Verified, to increase the Stakeholder confidence on the model.

Also, it is important to detect if the contracting person is be the owner of the model and become involved in the project. If not you need to detect who will be the final owner, o include them on the Validation and Verification processes.

Finally just to remark that a compelling animation is going to help to increase the confidence on the model.



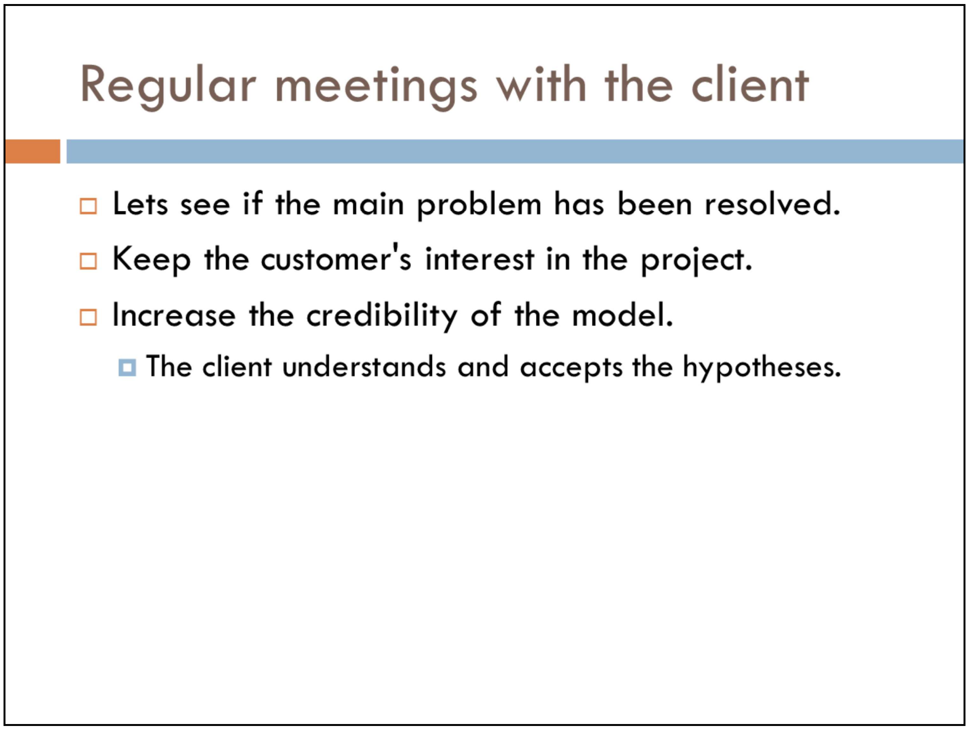
To do so, the final presentation must include animations and a discussion regarding the validation and verification processes that has been done during the construction of the simulation model.



There are several methods tool increase the confidence of the final stakeholders with the model. First, keep regular meetings with the clients. don't wait to present a final model. Present different prototypes as they are evolving on the model description and codification.

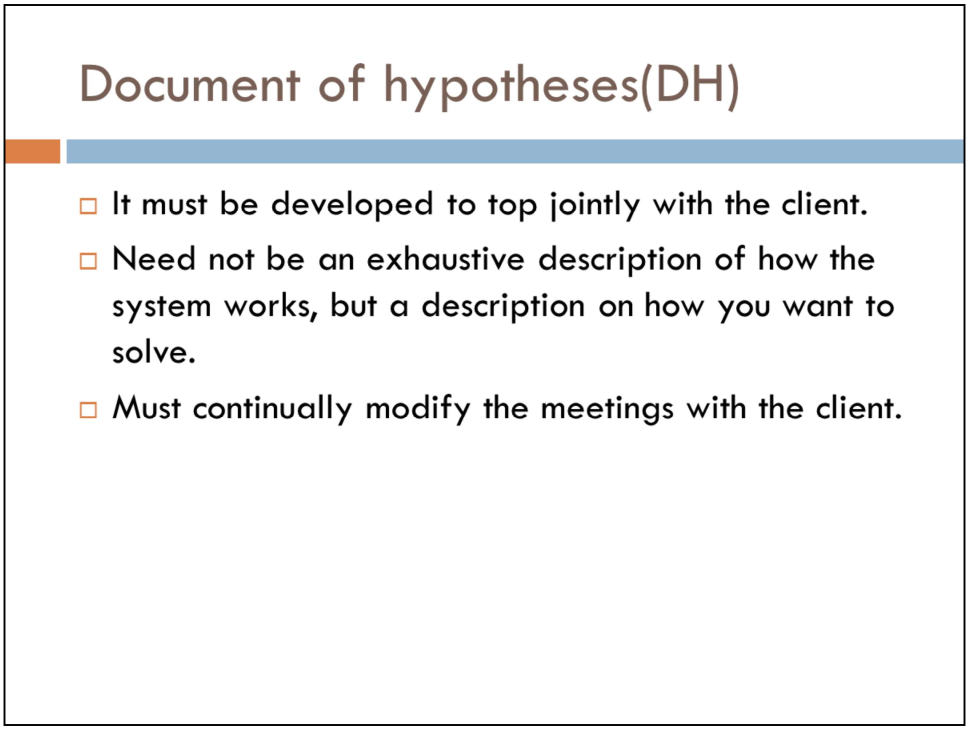
Develop and maintain the hypotheses document, It is a key document since it contains all the assumptions that must be assumed as true by all involved parties of the project.

Finally, promote that all active parties of project are participate an active role, if someone is discarded, problems can arise from its side.



The regular meetings with the client must be focused on detect if the problem that we want to solve has been resolved. When this happens the project will finish.

This meetings keeps the Stakeholders and customers interest on the project evolution and allows to detect any change on its interests. Also, increases the model credibility, since the Stakeholders assumes as true the assumptions.



The hypotheses document must be developed with the client and contains the relevant assumptions to be considered on the model. Hence, is not needed that be an exhaustive document, but relevant to understand the key assumptions to be considered on the modeling process. Later on the conceptual model all the assumptions will be formalized. Is a live document, hence can change through a consensus.

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.

The components of the document that we will present on the regular meetings can have this different sections:

The Objectives, problems and performance measures we will use to see if we achieve the goals.

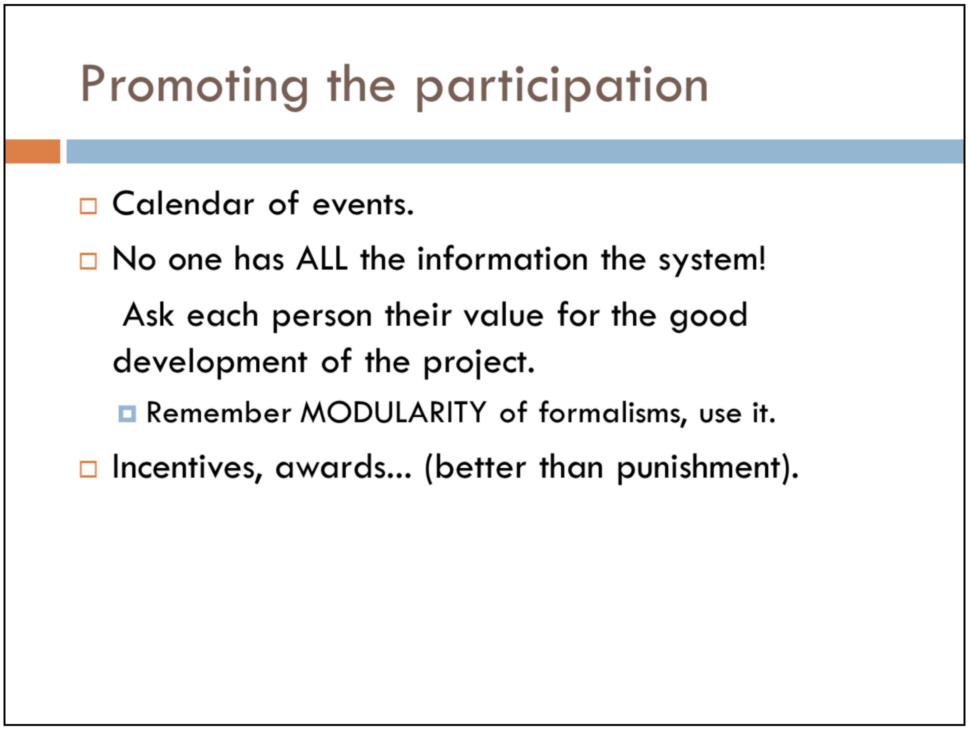
The interaction of the different subsystems that compose the model.

Hypotheses, through the hypotheses document.

The limitations of the model at the current development state, and the limitation we know due to the knowledge we have of the system or the technology we can use.

The data the model use.

And the sources of information related to the project.



To promote the participation on the project, it is needed to define a calendar of events.

Remember that no one owns all the information of the system, then each person can introduce some components to the model, modularity can help here dividing the model indifferent sub-parts that can be discussed separately.

And remember that is better to use incentives and awards instead of punishment to make that the projects evolves satisfactorily.

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/

Finally, since the accreditation must be headed by a different third team, different to the team that is doing the Validations and Verifications or the team that is doing the modeling process (in the case of Independent Validation and Verification), there are some big organizations that develops clear directives to do so.

At this point it could be of your interest to review what is done in the frame of the USA ministry of defense in this aspect.