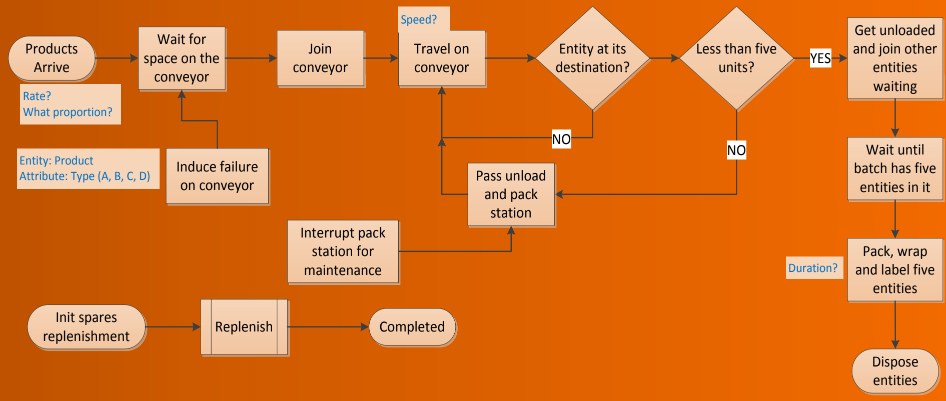
Determine the problem and the objectives
Project planning
Define the boundaries for the study
Formulate a concept model
Preliminary experiment
Determine input data required, obtain the data and process it
Translate the model to a computer simulation language
Verify the model
Validate the model
Rework model
Execute the production runs
Perform the statistical analysis
Model modification and scenario analysis
Document the study concurrently
Implement the results of the study
Monitoring and maintenance



2. Steps in a simulation study

A SIMULATION study is initiated by a problem in an existing system or while designing a new system. The analyst should however carefully consider the nature of the problem: is simulation really the tool to be used for investigation, or is any other modelling or problem solving tool more suitable? Simulation is one of several tools available to the engineer for problem solving purposes. The other alternatives usually provide quicker results and are not as expensive, but do not always provide suitable solutions to the problems at hand.

Once the analyst has established that simulation is indeed the preferred analysis tool, the following steps are suggested to model the problem. The steps should not be followed as a recipe, but may be tailored to suit the problem. Some are mandatory, while others may be reduced to the essential. Some steps may also be executed concurrently. The steps are discussed next.

2.1 Determine the problem and the objectives

This step is divided into three sub-steps: define the problem, *i.e.* compile a problem statement, do orientation and finally state the desired objective(s) with the study (Chung, 2004).

It often happens that a problem is analysed without really understanding what the real problem is (or what the root causes are), and often only the side effects or symptoms of a problem are investigated. Einstein stated that the formulation of a problem is even more essential than its solution, and in defining a problem one gets a better understanding of it. (The Japanese are well known for their belief that when a problem is properly defined, the solution is already 50% in sight). All members of a simulation project team, as well as the client (management, colleagues or an outside company) should therefore collaborate to define and agree on the problem definition/formulation.

To assist with the problem formulation, one could use techniques like the Fishbone and Pareto chart (Chung, 2004). It is important to state the expectations of all stakeholders during this phase to avoid scope creep of the study. It is further essential to clearly define the goals – what are the purposes of the study, why is it done, what is expected and what can be expected within the limitations of available resource allocation? The expectations must be stated for the short, medium and long term. It is also important that all parties agree on the goals, and that conflicting objectives be resolved. One or more of the following, depending on the real-world system, can drive the purpose of the study (Law and Kelton, 2000):

Evaluation	Comparing a system when evaluated against specific criteria.
Comparison	Comparing competing systems of similar functionality and comparing proposed alternatives.
Prediction	Estimate system performance under different operating conditions.
Sensitivity analysis	Determine which of one or several factors influence system performance.
Systems Improvement	Determine which of one or several factors influence system performance.
Optimisation	Determine which combination of factor levels results in best overall system performance.
Simulation optimisation	Integrating conventional simulation techniques with optimisation techniques to produce an optimal solution.
Functional relations	Determine the nature of relationships and the effects on the system's performance.
Other	Confirmation of proposals.

The industrial engineer should also remember their six Ms during problem formulation. The Ms tell us what we work with, and the efficiency of one or more Ms in a system could be improved using simulation. The Ms are summarised in Table 2.1.

Table 2.1: The six Ms guiding industrial engineers

M	Explanation
Man	The human being is part of our systems: it could be clients, personnel (operators, nursing staff), stakeholders, colleagues, contractors and competitors. Questions: 1. How many construction workers do we need for a given task? 2. What is the best schedule for a number of nursing staff? 3. How do we motivate employees?
Machine	The non-human resources doing work in the system, for example manufacturing machines, trucks, fork lifts, conveyors, robots, computers. Questions: 1. How many of a resource is needed in a process, for example, how many drills are needed in a job shop? 2. When must resources be taken out of production for maintenance? 3. When must a resource be salvaged? 4. Which is the best capacity of a potential new resource?
Material	This is processed or transformed into something useful: Steel, wood, fresh produce/food. Questions: 1. What is the best inventory level for a material to minimise inventory cost but maximise availability? 2. How can the cost of material handling be reduced (or contained)? 3. When must a material be acquired, and how much of it?
Money	Finances are always paramount in a capitalistic world. Costs of the other Ms and profits are considered, usually over time. Questions: 1. Which resources are the cheapest to acquire, but will give the best service? 2. What is the cost of a personnel schedule? 3. What is the expected profit of a given combination of personnel, resources and materials?

Continued on next page

M	Explanation
Method	"Work smart, not hard!" Work methods, procedures and processes can be improved using simulation. Questions: 1. How can a process, consisting of a series of activities be streamlined without losing its essential outcome? 2. How can the duration of an activity be shortened? 3. What is the best layout for a planned workstation?
Mother Nature	Always find ways to cause the least harm to the planet. When solving problems, keep the environment in mind. Questions: 1. How can the energy consumption of resources be minimised? 2. What is the best operating procedure in a process to minimise carbon emission? 3. What are good recycling policies for my company?

During orientation, the analysts familiarise themself with the system that will be studied. This activity comprises an initial orientation visit to the problem area, during which high-level information regarding operations is collected. A second visit focuses on details, for example, resources, buffers, business rules, exceptions, and performance measures or *key performance indicators* (KPIs). There are two main types of performance measures. The first type is *qualitative*, like customer satisfaction, employee morale or operator fatigue. The second type is *quantitative* and examples of these are shown in Table 2.2 (adapted from Pooch and Wall (2000)). A third visit serves as a review opportunity during which the analyst will confirm previous observations and finalise outstanding issues (Chung, 2004). Knowing and understanding the

Table 2.2: Examples of simulation performance measures

Measure	Example(s)	Where, typically?
Throughput (rate)	Number of Car model X manufactured (per day)	Car assembly plant
	Number of freight containers loaded by a crane (per day)	Sea port
	Number of patients treated (per 24h)	Doctors' rooms
	Volume/mass of table grape packed (per shift)	Farm packhouse
Performance	Percentage of time a resource works/idles	Factory, construction site
	Percentage of time a resource is available/down	Factory
	Proportion of calls an operator accepts	Call centre
	Proportion of calls of a specific type	Call centre
Finance	Measure of cost over time	Supply chain
	Measure of income over time	Factory, warehouse
	Measure of net profit over time	Factory, transport fleet
Operations	Number of products delayed per time	Factory
	Buffer occupation by parts at a machine	Factory
	Number of patients waiting in emergency	Trauma unit
	Duration of patient waiting time in emergency	Trauma unit
	Duration of caller waiting time	Call centre
	Service level	Call centre, bank
	Number of customers balking (what is this term?)	Retail store
Time measures	Makespan of a simulated schedule	Job shop
	Average overtime worked per 24h	Factory, rail network
	Average lateness of orders	Online store
	Time to produce a specified number of products	Factory

process(es) and the (sub)system to be studied is also important for the development of a concept model as discussed in Section 2.4 on p. 25.

Finally, the following maxims from *The Handbook of Simulation* (Banks (1998), pp. 724 – 726) are worth considering:

- *A strong finish begins with an effective start.*
- *Fuzzy objectives lead to unclear successes.*
- *It is easier to correct an expectation now than to change a belief later. (Perception often becomes fact.)*
- *Focus on the problem more than on the model.*

Another good guideline for successful practice of simulation is given by Sturrock (2013).

2.2 Project planning

Project planning is a crucial element in ensuring the success of a simulation project, as it is the integrator of the efforts performed by the different resources of the simulation project. The major components of a project are

- Personnel
  - expertise
  - availability
- Hardware and software
- Funds
- Time

From a simulation project point of view, the personnel required to execute the project include the analysts, members of management and the system's operating staff. The analysts may be divided into subgroups executing parts of the project, and may include consultants from one or more external parties.

Simulation software often requires additions to standard available hardware to operate efficiently. If hardware and/or software are not available, they must be obtained, incurring costs and time delays that include training because the required skills and expertise are not always readily available.

Secondary costs may be incurred by travelling and the labour hours consumed by personnel who have to supply information for the study.

A complete, agreed project plan must be compiled, with detailed documentation of all the activities listed. Management must support the project and make it known to all concerned. *It is better to work with many intermediate milestones than with one absolute deadline* – maxim from The Handbook of Simulation (Banks (1998), p. 733).

2.3 Define the boundaries for the study

We have seen before that a system is a collection of interrelated objects that function towards one or more stated goals. The system can be viewed from different perspectives: functional, data flow, physical, *etc*.

The simulation analyst needs to view a part of the real world (the problem) as a system in order to define boundaries for the study as well as the level of detail. The boundaries define what will be included in and excluded from the model. Two boundaries are of concern (Law and Kelton, 2000):

- The boundary that separates the system (problem) from the rest of the universe it belongs to (the physical boundary).

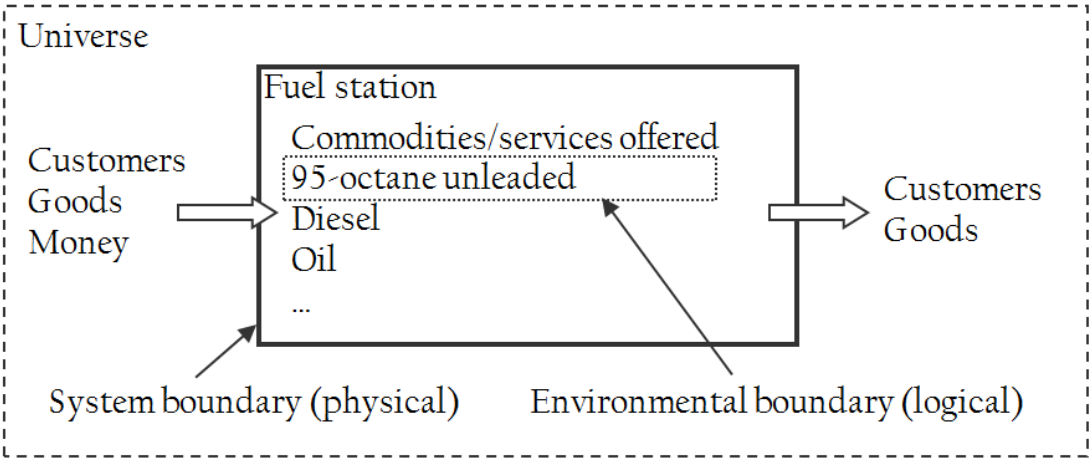


Figure 2.1: Example of a system boundary

- The boundary that separates the system defined from the environment in which it operates (the business boundary).

It is important not to exclude the interactions of the total system from the bounded system, unless they are insignificant. Consider a fuel station as a system. The fuel station sells several commodities and services, but the manager wants to determine the optimum reorder point for 95-octane unleaded petrol. The two boundaries are shown in Figure 2.1.

The purpose of boundary definition is therefore primarily to *simplify* the simulation study by reducing the amount of detail included. It is recommended that the smallest and least detailed model that provides the required information be used. The author has learnt through experience that large models should be developed from the top down and in two or more levels of detail. The first 'sweep' is usually performed at a rather high level of abstraction thus providing the least detailed model. This model is then embellished during a second and subsequent sweeps to a more detailed model. Verification and validation are also generally easier and faster for each sweep. The level of detail is generally reduced by aggregating related interdependent processes into 'black boxes' with a single input and output, and should vary in proportion to the impact it has on the results.

2.4 Formulate a concept model

During this step, the digital model is planned, and a proposed model is laid out 'on paper' in pseudo-code or graphical format, which we refer to as the *concept model*. A concept model describes the model logic in programming language-agnostic fashion so that a common reference for stakeholders from different backgrounds can be established. The real-world process or system is thus translated into the concept model from which the digital model is derived. The idea is shown in Figure 2.2, where detail is removed by making assumptions. The real-world is detailed and complex, but an approximate model is created, which must still be useful, but with less detail.

Boundary definition as discussed under the previous step, and model abstraction (developing a concept model) also include making assumptions. These should be *thoroughly and unambiguously* documented and referred to during the study. The list of assumptions is dynamic and should be updated during model development and must be included in the final report. The output results should also be evaluated against the assumptions made, and the project team and client(s) should all agree on the assumptions.

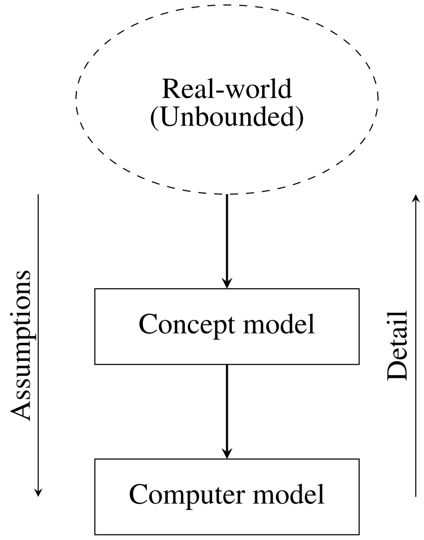


Figure 2.2: Real-world process via concept model to digital model

A typical relationship between the utility and complexity of a model is shown in Figure 2.3. The usefulness of a model can be the same for various levels of complexity, as shown by the red lines and blue circles. A less complex model (shown with a '1' on the vertical axis) representing the essentials of a system is usually a sufficient imitation of the real world. A more complex model (shown with a '2') requires more effort and cost but yields the same utility.

There are several advantages to the concept model, some are

1. The analyst(s) will quickly realise how well the system under study is understood.
2. Lack of information on system operations will be identified, which can be supplemented during a second round of interviews with those involved in the operations.
3. The model's logic is established (at least on a first order level).
4. Input data requirements are identified.
5. Assumptions are verified.
6. The need for more/less assumptions is identified.
7. First order validation can be done with those involved in the system's operations. This has the secondary advantage that these people do not need any computer knowledge.
8. A thoroughly designed concept is easier (and usually faster) to code.
9. The various levels of detail become apparent and can be adjusted.

This step can be viewed as a form of planning the computer model, and is usually neglected, because we tend to try to start coding the model as soon as possible, and often sooner than possible! We diagram the flow of a typical entity through the process logic. Along the way, we consider

- The entity type and possible transformations into other entity types.
- Entity attribute changes.
- Resources needed.
- Buffer/queue areas needed.
- Business rules to obey.
- Input data required.
- Performance measures required.

When constructing the concept model, consider the following maxims from *The Handbook of Simulation* (Banks (1998), pp. 733–737):

- Add detail, do not start with it.

- Let the model work for you.
- If it does not make sense, check it out.
- People decide, models do not.

A worked example of a concept model follows below. A narrative is usually obtained from one or more stakeholders and is used to develop the concept model. The model is usually presented in a graphical format. An informal set of symbols is used in the example, but the simulation analyst may use formal languages like activity diagrams from UML or Business Process Modelling Notation (BPMN, see `www.bpmn.org`). The business rules, decisions, storage, buffering, processing, and entity interaction are modelled.

Example of a concept model

Question: Develop a concept model for the following description:

Cars arrive at a parking area in Bitterfontein. An operator at the only access/exit gate regulates entries. If a parking space is available, the currently arriving car is given a ticket with the time stamped on it, and the car is allowed into the parking area. If the area is full, any cars arriving are denied entry. Drivers and their cars wishing to leave the parking area must first pay the operator at the gate a fee that depends on the duration of the parking time. A car waiting to pay before leaving the parking lot is considered to be taking up parking space. That space is thus only released when the payment transaction has been completed. All cars are the same size, so one car occupies one parking space. Ignore the driving of the cars in the parking area and assume the operator does not get tired. Also, assume that the technology does not fail.

Answer: How should we approach the concept model development? Look for (an) object(s) that flow(s) through a process, as the documented narrative mentions. If the absence of this object causes the process to cease, then it must be an entity. Identify objects (resources) for which the entities compete, for example, transporters, personnel, or machines. Look out for business rules which dictate the flow of entities. Also, identify holding areas or buffer spaces. Points along the flow of the entities will require input data, for example, the duration of a service by a resource, while some KPI values could be observed and registered along the way. It is possible that an entity can be split into other entities (an order becomes an order and a picking slip), or entities

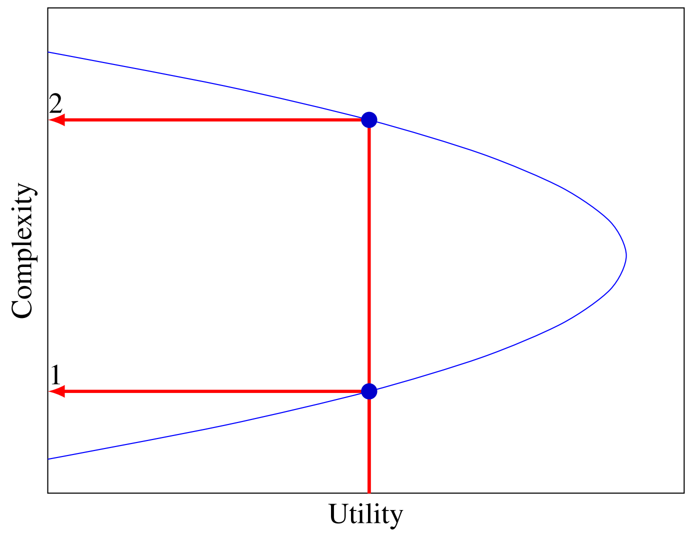


Figure 2.3: Relationship between complexity and utility (Banks, 1998)

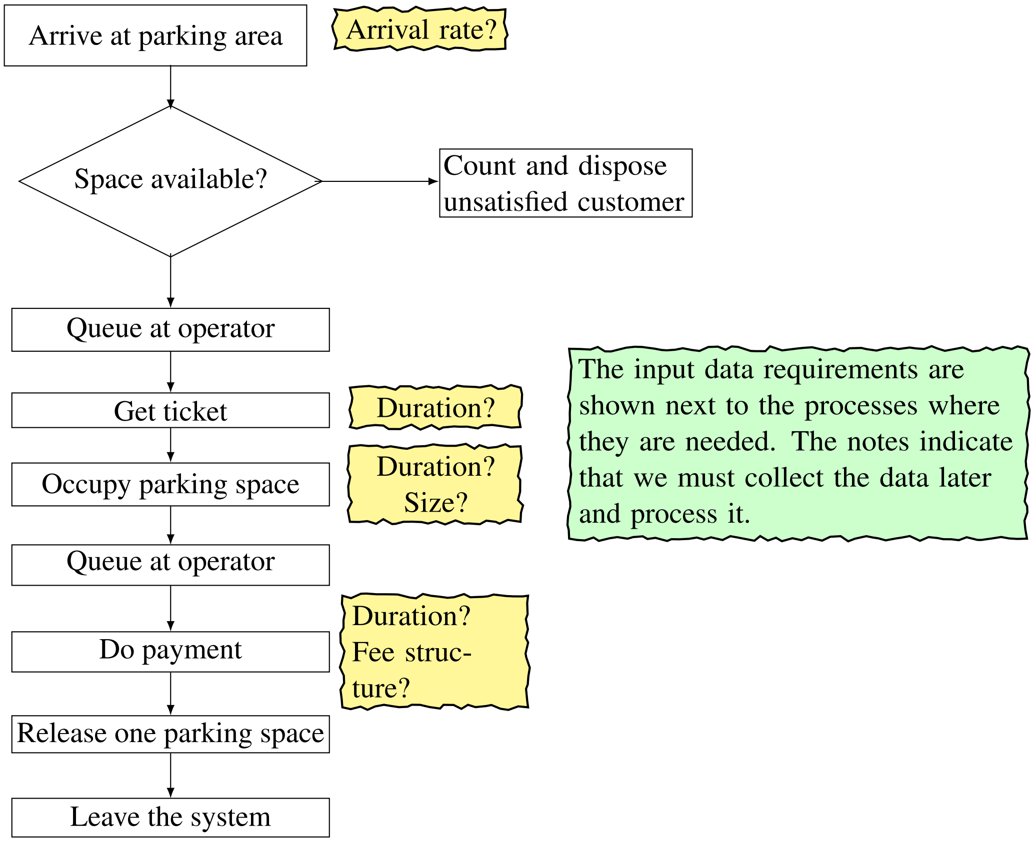


Figure 2.4: An example concept model

can be merged (a truck entity and load entity become a dispatched entity). The description of the example shows that the model entity represents arriving cars. The cashier is a resource, while the parking area is a holding area where each entity (car) will be held until it starts to compete for the cashier again (on exit). An example answer of the model is shown in Figure 2.4.

Once the concept model has been developed, the preliminary experiment must be defined, as is subsequently discussed.

2.5 Preliminary experiment

This step comprises the establishment of

- Level of confidence for the confidence intervals (usually 95%).
- Model time span, for example 8 hours per day. This determines if we deal with a terminating or non-terminating system.
- Input variables to be varied during the study: These are the *decision variables* over which the simulation analyst takes control.
- Limits of the decision variables: The ranges of the decision variables have an impact on the simulation effort. Many decision variables with large ranges may cause long simulation times because many scenarios must be simulated. Suppose we have two discrete variables X and Y, and we vary each from 1 to 10, then the decision space has $10 \times 10 = 100$ scenarios.
- Identify the model output parameters (also known as *key performance indicators* (KPIs))that will be studied to supply the desired information from which conclusions of the system's performance will be made. These will comprise the *output parameters* on which output

statistical analysis will be done, and they will drive the required number of replications and/or the length of the simulation run. These parameters are identified and translated from the simulation study objectives to variables, *e.g.* the variable Throughput will describe the output parameter *Products completed per day*. An objective of the study could be to determine the scenario with the highest number of products produced per day.

- Definition of the entity or the entities, and possible conversions of entities into other entity types.
- Attributes per entity where applicable.
- Scale of units of measurement: Time and other units to be used, *e.g.* m/s vs. km/h *etc*. Tecnomatix uses SI units of measurement.
- Resources in the model.

2.6 Determine input data required, obtain the data and process it

Collect the data required for the model as identified by the concept model definition and the preliminary experimental design. This is usually a time-consuming process. The details of this step are discussed in Chapter 4.

2.7 Translate the model to a computer simulation language

The computer model is developed from the concept model as defined in Section 2.4.

2.8 Verify the model (debugging)

This is a detailed step in the simulation study and is simulation software orientated, and we ask the question “Did we build the model right?”. The main action here is debugging to verify that the computerised model works correctly. Other actions include:

- Syntax errors are corrected.
- Logic is checked.
- Compiler (if applicable) errors are corrected.
- A secondary source of errors is *Run-time errors* that only manifest during program execution, which must be corrected.

If no errors occur, it does not mean that the model is error-free, the errors have simply not manifested with a given data set. Because the absence of errors does not guarantee that the model is correct, our attitude must be *destructive*, *i.e.* we must focus on *finding* and correcting errors rather than demonstrating model correctness. Several approaches can be followed to find errors:

- Establish a doubtful frame of mind and try to ‘crash’ the model by supplying various data sets. A successful model run with a given data set must be viewed as a failure to locate errors.
- Outside ‘doubters’ may be incorporated. They should understand the model’s intention but prior involvement in the model development is not a prerequisite for such collaborators.
- Conducting ‘walk through’ sessions by manually emulating the execution of the model.
- Execute the model with a single entity created, and follow it through the model with a tracing facility. The latter is a feature included in most simulation packages. Note that a single entity may never follow certain routes through the model due to logic, so that these excluded paths are never followed. The logic may be modified for the moment to force the model execution through those routes. A complete model execution may also be traced, but it becomes difficult to keep track of all entities in a complex model because of the parallel execution of the model.

- The model can be executed with certain parameters set to certain values, so that the effect on the model can be anticipated.
- Animation can be used to verify a model's execution.

2.9 Validate the model

Verification allows us to confirm that we have built the model right. In contrast, validation allows us to confirm that we have built the *right model* for the particular objective(s) of the study (Law, 2015). A verified model has code that executes correctly, given the tests it has been subjected to. There may be latent errors in the model code not identified by verification tests. A valid model is an approximate but adequate representation of the real-world process being modelled.

Verification and validation are interrelated: when verifying, the model is partially validated in terms of correct output. Verification is first done, and some validation is often required to demonstrate that a model is verified. For example, a model gives the correct output when all input is made deterministic, and the output can be calculated. The relationship can be visualised with a Venn diagram as shown in Figure 2.5.

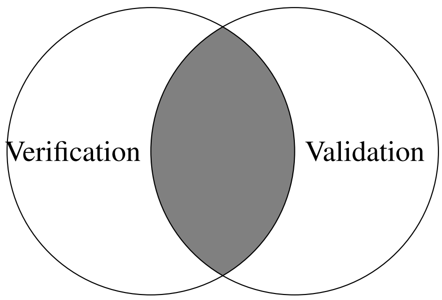


Figure 2.5: Verification and validation

Three questions must be considered during validation (Law and Kelton, 2000):

Conceptual validity:	Does the model represent the real world system adequately?
Operational validity:	Can the model's generated data be associated with the real world system's behavioural data?
Credibility:	Does the model's end-user have confidence in the model's results?

From this it follows that validation takes place from three different perspectives (Law and Kelton, 2000):

The analyst	Design and conduct confidence-building tests.
The technical evaluator	Reviews the technical data and information provided by the modeller.
The user and/or decision-maker	Usually an outsider who does not understand the validation tests executed; they only require that the model maps onto the real world.

The Handbook of Simulation (Banks, 1998) presents 75 verification, validation and testing techniques. Of these, the informal techniques like auditing, desk checking, face validation, walk-through sessions, inspections and reviews are arguably the most intuitive, easiest and efficient in terms of detecting most errors/discrepancies in models.

Formal statistical techniques like the paired t-test require that some assumptions be taken into account, which are not always practical. An example is the assumption of normality of the test sets. One can thus not compare simulation output of the model with that of the real world unless there is reason to believe that the two data-sets are normally distributed. In cases where this is not possible, the non-parametric tests like the Wilcoxon rank-sum test can be used. In this case, the equality of the means of two non-normal continuous distributions of which samples are independent, is tested (Walpole and Myers (1993), pp. 634–635).

A model must exhibit reasonableness, measured against the following factors:

Continuity: If small changes are made to input parameters, consequent small and appropriate changes should be reflected by the model's output and variables.

Consistency: Similar runs of the model should reflect similar results – the output should not be influenced to a large extent because a random number generator seed has changed. (Tecnomatix has several random number streams; the modeller may select the stream number and seed.)

Degeneracy: The model should reflect the removal of one or more features of the model. If, for example, there are two instances of a certain resource and one is removed, we expect less output from the set of resources, and a possible overload of the remaining resource.

Absurd conditions: If absurd conditions are introduced, the model should not necessarily produce equally absurd outputs. Absurd conditions should also not arise during model execution, for example negative mass, negative times. Variables should always be in their range where applicable – if we know that the start-up time for a machine is at least x minutes, then the variable should never be less than this value.

A model should also be tested at its extremes, that is at the minimum and maximum limits that certain variables may assume.

Validation is done at various stages of the simulation project. It starts when the concept model is developed, it is done while and after the computer model is developed, and leads to credibility once the model's results are accepted.

2.10 Rework the model where necessary

It will almost always be necessary to rework the model after verification and validation. After the changes have been made, the model must again be verified and validated. This is an iterative process, which is repeated until all parties concerned deem the model to be acceptable.

A word of advice may be added: the author has experienced that it is well worth the effort and time to ensure that the model is widely accepted and 'correct' before commencing with subsequent steps in the project. These steps are time-consuming and labour intensive. If any one or more of them are executed and it is found that the model is not working properly, a great deal of time and money are wasted in reworking the model and regenerating output data.

2.11 Execute the production runs

Refer to Section 5.1 (p. 59), Section 5.3 (p. 66) and Section 5.4 (p. 69). Note that to do validation, preliminary production runs are needed.

2.12 Perform the statistical analysis

Refer to Section 5.1 (p. 59), Section 5.3 (p. 66) and Section 5.4 (p. 69).

2.13 Model modification and scenario analysis

Modify the model for alternatives where required and repeat Steps 2.8 to 2.12. The output analysis must be repeated for a modified model.

2.14 Document the study concurrently

Although this step is listed as one of the last steps in the simulation process, the study must be documented right from the start. Ideally, only final touches should be required at this point, apart from adding the results and their interpretation, conclusions and recommendations. Experience has taught that it is good practice to update the simulation report after each step is completed while the actions involved in the particular step are fresh in memory. It is very difficult to recall decisions and their rationale after several months of other simulation activities.

2.15 Implement the results of the study

This step can be the responsibility of the client only, or it can be in co-operation with the modeller(s). Third party consultants may also be used for certain aspects of the implementation, *i.e.* human resources and/or issues with unions.

2.16 Maintenance, feedback, monitoring and refining

The modeller(s) should monitor that the implementation is maintained and must obtain feedback from the client to evaluate the success of the study. This normally leads to refining the model where possible and to subsequent improvement in their level of experience.

What proportion of budgeted project time should be spent on each of the steps above? Various sources recommend different time allocations to the steps above. The '40-20-40' rule is suggested by Pegden, R.E. Shannon, and R. Sadowski (1995), which states that 40% of the overall project time should be devoted to steps 1 to 7, 20% to step 8 and 40% to steps 9 to 19. R. Shannon (1975) suggests a different allocation, but the central message is that model translation (computer model development) is not the major effort of a simulation study, there are more important issues that deserve more time.

The steps above set the most formal requirement for a simulation study; the analyst should however judge to what extent each should be executed by tailoring where necessary. It will hardly be possible to follow the steps in strict sequential order – some steps may be conducted concurrently, especially when working in a project team. Some members may, for example, collect input data while others may start developing a concept model, which may impose new and/or different requirements for the input data. A study may often have a false start, in which case much of the work in some steps must be abandoned, or some rework must be done.

These are some realities that the analyst may face, and one should try to plan a simulation study in terms of these potential problems: the study would usually take longer than planned, cost more and require more effort.

P Point to ponder: Suppose you lead a team doing a simulation study for a large enterprise, spanning over a few months. How would you coordinate each team member?

Survival kit:

Ensure you have the following in your assessment pack:

1. Know the steps in a simulation study in broad terms.
2. Know the objectives drivers of a simulation study (Section 2.1).
3. Know the six Ms guiding industrial engineers (Table 2.1).

4 Know the simulation performance measure types and give examples (Table 2.2).
5 Know the types of system boundaries (Section 2.3).
6 Create and draw a concept model (Section 2.4).
7 Verification and validation are very important (Sections 2.8 and 2.9).