Implied Logic

Two-hour tutorial demo

Robin BAILEY

August 2022 ■ Cambridge, UK

DOC01 20100914\0032

August 2022

Two-hour tutorial demo

This is a basic introduction to the *STEM visual software for the reliable modelling of business* which covers all the most commonly-used features, and alludes to plenty more. It is intended as a follow-up to the minimal *One-hour tutorial demo* which provides an end-to-end overview in less time, but obviously with less detail. The two tutorials share certain elements, so time invested in the first will speed progress with the second. However, both are scripted to suit a newcomer, so the first step is not essential.

This *Two-hour tutorial demo* is framed around the provision of broadband connectivity in a metropolitan area. Tutorial examples explore essential techniques that would be used for many more revenue and cost headings in a detailed, commercial model.

The 25 exercises combine the definition and scaling of cost drivers with the calculation of a cost per customer, and hence profit margin for a given tariff, plus further iterations to compare business-case dynamics within a scenario framework. The result is a model that connects **technical credibility with reliable financial impact** within a fabric that is easily visualised and can be reliably and consistently extended to great scale.

Please contact sales@impliedlogic.com if you would like advice on working with STEM in your business.

Watch the accompanying video and share with colleagues

This tutorial is also available online, and thus readily shared. Each exercise is accompanied by a video extract from a live run-through, preserved, as-is, for the sake of spontaneity. A very few deviations from the text are noted on-screen to avoid confusion.

https://help.stem.impliedlogic.com/Training/Two-hour-tutorial-demo/

Contents

1. The business-modelling context

As a next step up from the wholly generic *One-hour tutorial demo*, we are going to model the provision of broadband connectivity in a metropolitan area. This will be a simplified approach, using just a few key elements as tutorial examples of the approach that would be used for many more revenue and cost headings in a detailed, commercial model.

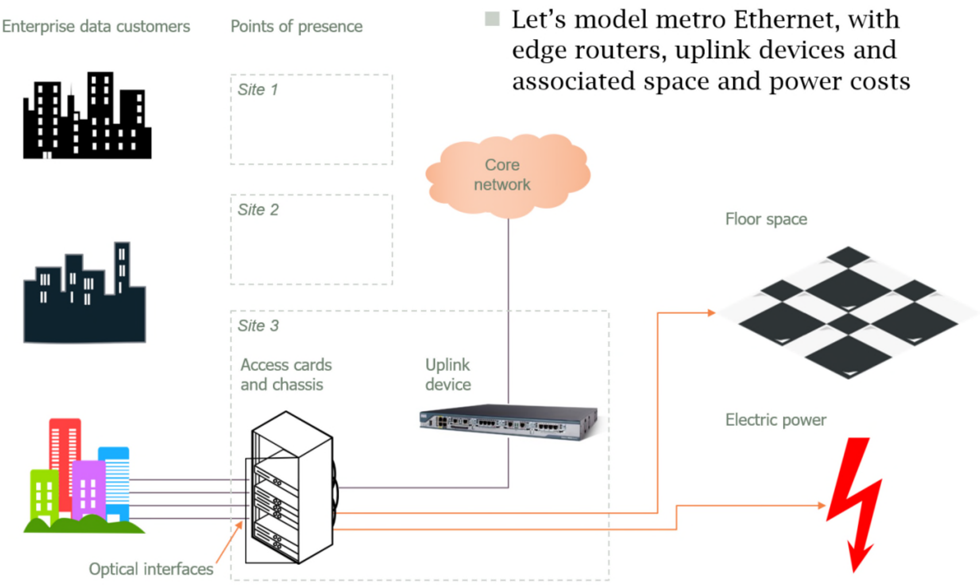


Figure 1: A 'real life' modelling context

The area in question has 25 central office locations where access equipment will be sited, and we will not consider the ring infrastructure required to connect these buildings. Our focus is the service and local access provision. We wish to offer a 100 Mbit/s service to a potential market of 1000 subscribers in the vicinity of these 25 locations. The business plan will assume a market penetration of 2% by the end of Y1 and 15% by the end of Y2. We will also assume an optimistic 10:1 contention ratio in the access equipment, meaning that the subscribers' traffic will be sporadic rather than sustained.

The access equipment comprises an access chassis which accommodates up to 5 access cards, each of which has 16 subscriber-facing ports. A per-subscriber optical interface (plug-in) is required for each active port. An uplink device connects the chassis to the assumed ring infrastructure, and this has a maximum bitrate of 1 Gbit/s.

Each chassis requires 0.5 sq m of floor space in the central office and has a power consumption of 200W. The corresponding overhead costs and the costs and physical lifetimes associated with the access equipment are summarised in the figure below, together with all of the other assumptions.

Element	*Capacity*	*Lifetime (yrs)*	*Capital cost (USD)*
Optical interface	1 customer	10	500
Access card	16 ports	5	5,000
Access chassis	5 cards	10	5,000
Uplink	1 Gbit/s	10	10,000

- 1000 potential customers:
  - 2% penetration in Y1 (of 10)
  - 15% penetration in Y2
  - 100 Mbit/s service
  - 10:1 contention in central office
  - tariff = $150 p.c.m.
- 25 central-office locations
- Maintenance = 5% of capex
- Floor space:
  - 0.5 sq m per chassis
  - 100 USD p.c.m. / sq m
- Electric power:
  - 200W per chassis
  - 15c per kWh

Figure 2: Some specific assumptions for the model

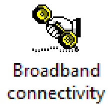
Note: an otherwise-blank model with notes of all these assumptions is available as a free download alongside the online version of this document on our website.

By creating this model, we hope to establish what will be a reasonable tariff to charge for this service; and then, more specifically, what tariff would be required to achieve payback after a period of five years, or four years, or even just three years.

Note: the following exercises assume you have access to the STEM Model Editor. Please refer to the One-hour tutorial demo if you are unsure about how to load the software.

The space between sections below is left blank intentionally for your own notes.

2. Service *Broadband connectivity*



The service element in STEM represents a group of potential subscribers with similar usage characteristics for a given business:

- the three separate measures of connections, traffic and busy-hour traffic can all be used to infer requirements for capacity in whichever resources are required to deliver the service (with their corresponding costs)
- revenue is accrued from connection, rental and usage tariffs which are billed to current subscribers of the service.

Exercise 1: Market potential and penetration

The first thing we need to do is create a service element:

1. Click the *Service* button on the toolbar, and then click again to place the ghosted icon at a suitable position in the view. The service element is created.
2. Select *Rename…* from the icon menu or press <F2>. The *Rename Element* dialog is displayed. Enter the new name, *Broadband connectivity*, and press <Enter>. The icon is now displayed with the new name.

Now we will enter the stated assumptions about the market:

3. Select *Demand* from the icon menu. The *Demand* dialog is displayed.
4. Enter the *Connections Unit* and *Customer Base* inputs as shown below.

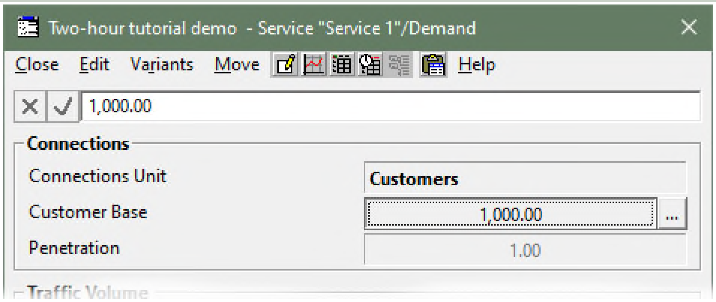


Figure 3: The *Connections* section of the *Demand* dialog for the service

Note: it is not necessary to click in the formula bar to enter a value, or to click the tick box to complete it. Just select the input you wish to edit and start typing, and then press <Enter> when you are finished, or press <Tab> to move on to the next input.

The *Customer Base* is straightforward to enter as a constant value, whereas we want to enter the *Penetration* input as a time series so that it will evolve over time as indicated:

5. Right click the *Penetration* input. A context menu is displayed.
6. Select *Change Type*. A cascading sub-menu appears, listing time-series alternatives such as *Constant*, *Straight Line*, and so on.
7. Select *S-Curve*. An *S-Curve* dialog is displayed.
8. Enter the essential six inputs as shown below (disregarding the *Options*).

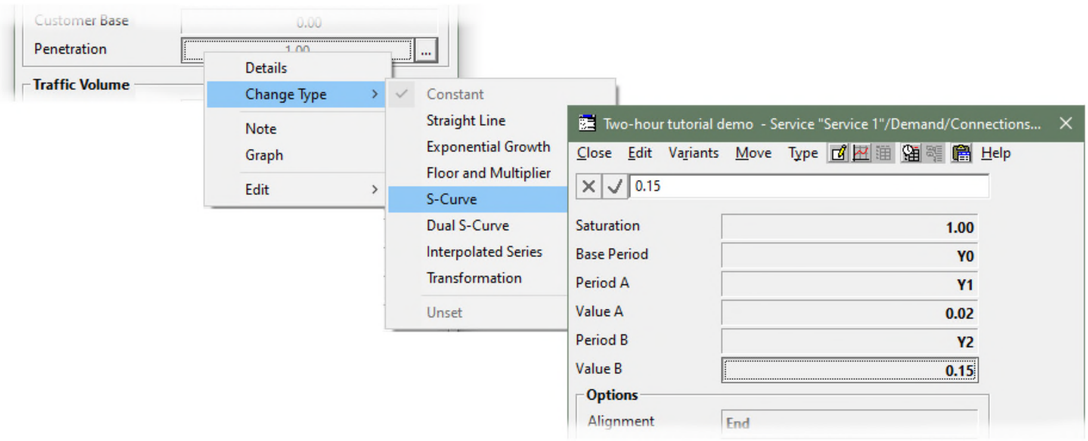
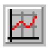


Figure 4: Entering *S-Curve* parameters for the *Penetration* input

The parameters are interpreted as follows:

- *Saturation* = `1.0` indicates that we aim to connect every potential subscriber eventually (i.e., 100%)
- *Base Period* = `Y0` directs that the curve should start from zero in Y0
- *Value A* = `0.02` specifies the level in *Period A* = `Y1`, and similarly
- *Value B* = `0.15` the level in *Period B* = `Y2`
- (each of these values to pertain at the end of the period by default).

We can easily preview the inferred curve and corresponding numbers:

9. Click the *Graph* button  on the dialog menu (or, if you prefer, select *Graph* from the *Move* menu, or press <Alt+G>). A graph of the *Penetration* input is displayed.
10. Right-click the background of the graph and select *Show Separate Table* (or select the same command from the graph *Format* menu). A corresponding table is displayed.

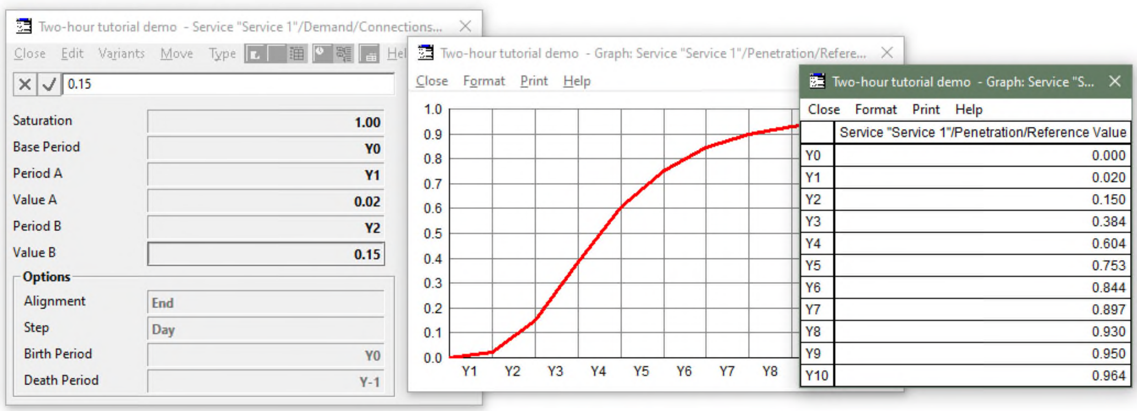


Figure 5: Drawing a graph and the associated table of the *Penetration* input

The graph provides the best overview of the time-series evolution, while the table allows you to check the inferred values in individual years. Are they as you expected?

Things that you should have seen and understood

STEM Model Editor (‘the Editor’), toolbar, service element, Demand
Connections Unit, Customer Base, Penetration
Change Type, S-Curve
Graph, Table

Exercise 2: Business-plan years and optional detail in quarters or months

Note: this next exercise is not strictly necessary to complete the model, but it covers crucial parameters which should always be considered when framing a business plan.

The graph we drew in the first exercise shows the evolution of the *Penetration* input over a default period of ten years (strictly speaking, from the end of Y0 to the end of Y10).

You may wish to relate your model assumptions to actual calendar years (such as 2021, or whatever year it is when you read this). Additionally, ten years might be too long for many exercises, and you may wish to set a shorter horizon for the model results:

1. Select *Run Period* from the main *Data* menu (or from the context menu available by right-clicking anywhere on the view background). The *Run Period* dialog is displayed.
2. Enter the *Model Start Date* and *Model Run Length* inputs as shown below. The graph and table are updated to show the evolution of the *Penetration* input over the five years, 2021–2025; i.e., from the end of 2020 to the end of 2025.

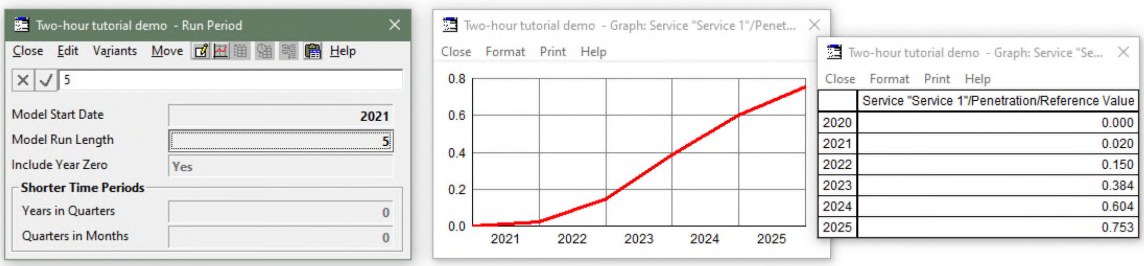


Figure 6: Limiting the *Run Period* to five years from 2021

Note: if your reporting year doesn’t start on 01 Jan, then you can enter a month, such as Apr 2021, or even a specific date for the Model Start Date. Try it!

Reducing the run length removes superfluous detail if you are focused on the nearer term, but you may seek additional granularity in the results instead, especially in the crucial launch phase:

3. Ensure the *Run Period* dialog is still active.
4. Enter the *Years in Quarters* and *Quarters in Months* inputs as shown below. The graph and table are updated to show additional, intermediate values, by month for the first two quarters, and then by quarter for the (remainder of) the first two years.

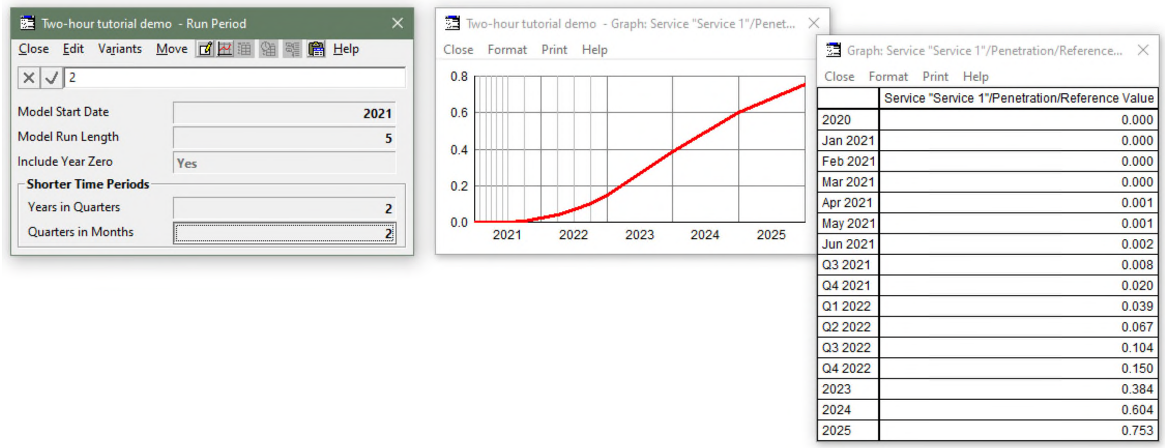


Figure 7: Generating optional quarterly and monthly results from the *Model Start Date*

These extra values are available because all time-series inputs are parameterised with a formula which provides for values to be inferred at any time, rather than just at yearly intervals. However, with all this extra detail, you may wish to date your assumptions more precisely too, or you may already have a mid-year data point to accommodate. Let's try bringing forward the 15% penetration assumption (currently end of Y2):

5. Switch back to the *S-Curve* dialog for the *Penetration* input.
6. Try replacing the *Period B* value of `Y2` with `Q3 Y2`, or even `Aug Y2`. Now what?
7. When does the *Penetration* input reach 15%?
8. How does an `Aug Y2` assumption work if the model only has quarterly results in Y2?

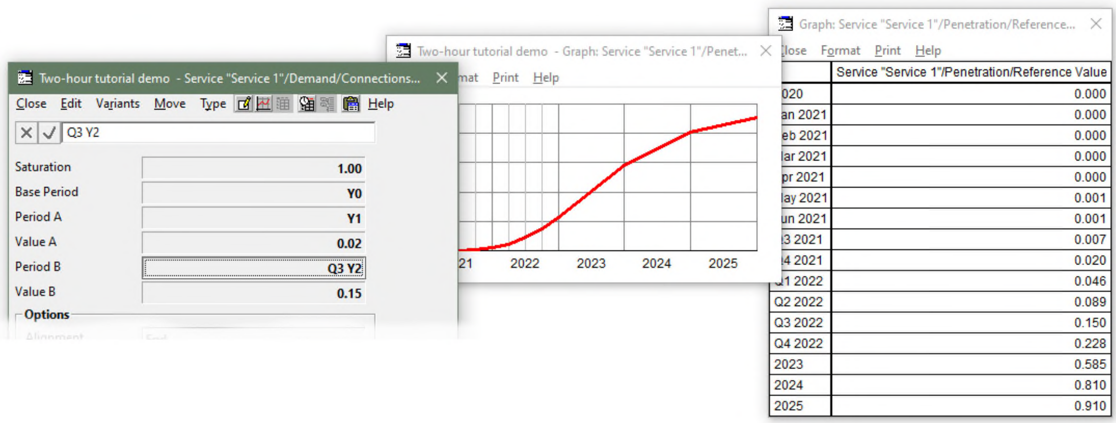


Figure 8: Entering a mid-year data point

This detail for *Shorter Time Periods* is optional and automatic, and can be turned on and off at will, as above, without affecting the integrity of your model. All of the calculations in STEM are 'period aware' so you don't have to waste time on it or worry about it.

The Results program includes a *Consolidation* option (to aggregate quarterly or monthly results) so that financial results can be presented on an annual basis even when the results are generated in quarters or months. *(Beyond the scope of this tutorial.)*

For sake of simplicity (and to avoid the exhibits aging from the continued use of calendar years), we will turn these options off for the remainder of this tutorial:

9. In the *Run Period* dialog, press <Ctrl+A>. All the fields are selected.
10. Press <Del>, or select *Unset* from the dialog *Edit* menu. All the fields are returned to their default values, such as *Model Run Length* = `10`, and the graph and table revert to showing annual values.
11. In the *S-Curve* dialog for the *Penetration* input, re-enter *Period B* = `Y2`. The graph and table update to show 15% in Y2 once again.

The beauty is that you can enable these options at any time, and your model will just keep working without having to start from scratch. Try that in a spreadsheet!

Note: the Run Period dialog is (optionally) displayed automatically when you create a new model, but the design allows for you to revise these fundamental settings at any time.

Things that you should have seen and understood

Data menu, view-background menu

Run Period, Model Start Date, Model Run Length

Years in Quarters, Quarters in Months

Using quarters, months or dates for period inputs such as Period B

Exercise 3: Average service usage characteristics

With most services, there is a crucial distinction between how much work is done (value or volume delivered) and the rate at which this work is done (how busy the service is). Revenue may be proportional to value delivered, or flat-rate, but the fixed assets in a system must be dimensioned to deliver the required work at the busiest rate.

Note: the following preamble will help you understand which inputs to enter in the service Demand dialog in order to complete the Broadband connectivity service description.

A service in STEM has characteristics of *Traffic Volume* and *Busy Hour Traffic*, and the separate *Traffic Unit* and *Busy Hour Traffic Unit* inputs can be tailored to the nature of the service. For example, a voice service might deliver `Call Minutes` and be dimensioned in `Erlangs` (simultaneous circuit density), while a data service might deliver `GBytes` at a rate of `Mbit/s` (transmission speed). Both unit-label inputs are relevant, regardless of the *Traffic Calculation* (see below), and should always be entered for clarity.

The separate *Traffic Volume* and *Busy Hour Traffic* inputs may be entered independently, but more commonly the corresponding results are related (one way or the other) by the *Distribution* of service delivery over a typical day, and over the year as a whole.

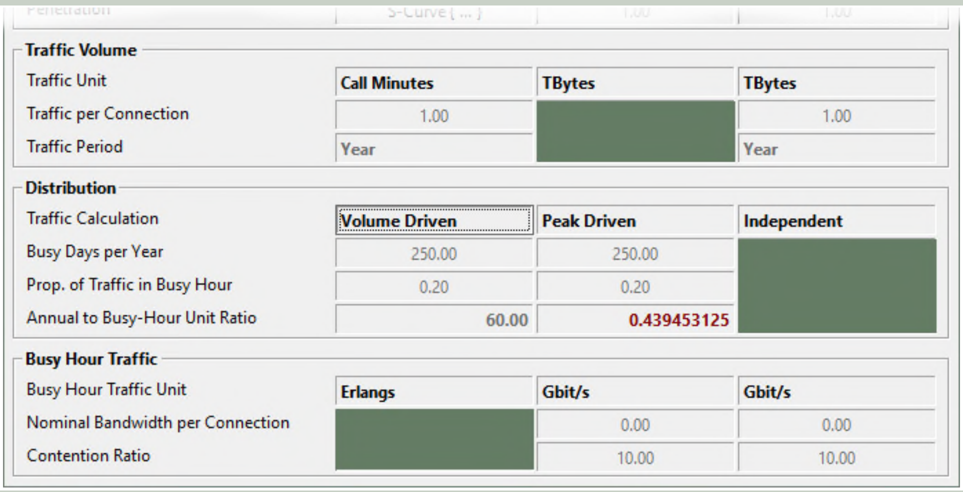


Figure 9: Separate inputs for *Traffic Volume, Distribution* and *Busy Hour Traffic* in the *Demand* dialog

The *Traffic Calculation* input is a crucial choice which determines how the associated inputs are reconciled (one set of which is always ignored, as per the shading above):

- `Volume Driven`: *Traffic* is defined by the *Traffic per Connection* and *Traffic Period* inputs, *Busy Hour Traffic* is inferred from *Traffic* (according to the *Distribution*), and the *Nominal Bandwidth per Connection* and *Contention Ratio* inputs are ignored
- `Peak Driven`: *Busy Hour Traffic* is set by the *Nominal Bandwidth per Connection* and *Contention Ratio* inputs, *Traffic* is inferred from *Busy Hour Traffic* (with the inverse *Distribution*), and the *Traffic per Connection* and *Traffic Period* inputs are ignored
- `Independent`: *Traffic* and *Busy Hour Traffic* are determined separately, according to their respective inputs, and the other *Distribution* inputs are ignored.

A voice service may be easily characterised in terms of call minutes and distribution (`Volume Driven`), while a data service might be all about speed (`Peak Driven`), but if you have good data for both volume and rate, then use `Independent`.

Important: if *Traffic Calculation* is either `Volume Driven` or `Peak Driven`, then the *Distribution* inputs must be calibrated in order for the respective *Busy Hour Traffic* and *Traffic* results (which are always available) to be meaningful. It is very likely that some resources will be dimensioned according to *Busy Hour Traffic*, and revenue will be a function of *Traffic* unless the service tariff is flat-rate only.

Note: now we are ready to complete the service description!

According to the introduction, we wish to offer a 100 Mbit/s service, with an optimistic 10:1 contention ratio in the access equipment:

1. Select *Demand* from the service icon menu. The *Demand* dialog is displayed, as per the figure below (which also illustrates the required inputs which follow).
2. Enter the *Traffic Unit* and *Busy Hour Traffic Unit* inputs first for clarity. Remember that the results are aggregated across many customers, so it is usually better to work with units ‘up one order of magnitude’ from the per-customer perspective.
3. Enter the *Traffic Calculation* next, as this governs which other inputs are required.
4. Enter the *Nominal Bandwidth per Connection* and *Contention Ratio* inputs.

5. The *Busy Hour Traffic* result will be calculated directly as *Connections* × *Nominal Bandwidth per Connection* / *Contention Ratio*. (We will verify the numbers shortly.)

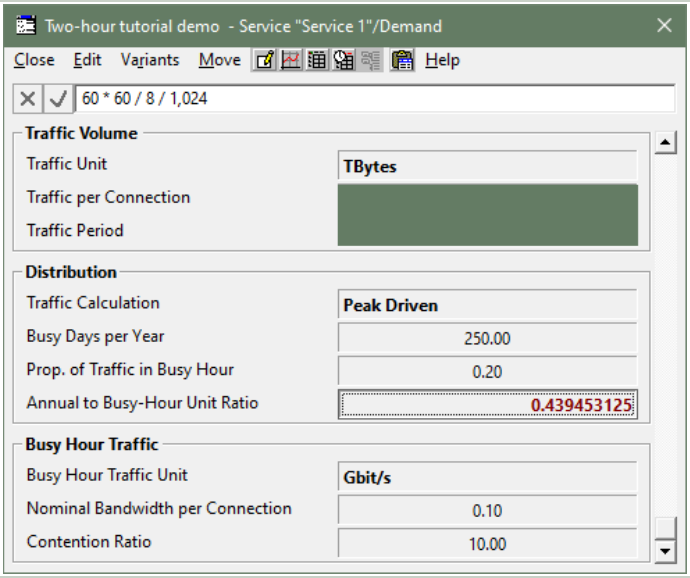


Figure 10: Entering the *Busy Hour Traffic* and *Distribution* inputs for the *Broadband connectivity* service

Note: it is not essential at this stage, but it is nevertheless highly recommended to set the Distribution inputs at this stage so that the Traffic (volume) result will also be meaningful.

6. Enter the *Busy Days per Year* input at the default value of `250.0` (corresponding to a business service with no significant usage at weekends or on public holidays).
7. Enter the *Prop. Of Traffic in Busy Hour* input at the default value of `0.2`, and consider what this means.
8. The *Annual to BH Unit Ratio* input is crucial, and requires careful thought. Helpfully, the default value of `60.0` (for `Call Minutes` and `Erlangs`) provides a helpful clue as to the sense of the ratio. In this case, we are converting from seconds to hours, from bits to bytes, and from GB to TB (for `Gbit/s` and `Tbytes`).
9. Enter the value as a formula so that the separate factors are visible (see above).
10. The *Traffic* result will be calculated as (average over the period) *Busy Hour Traffic* × *Annual to BH Unit Ratio* / *Prop. Of Traffic in Busy Hour* × *Busy Days per Year*.

This provides only a very crude estimate of the traffic volume, as it assumes the service is fully-utilised throughout the busy hour. Why might this overstate the likely traffic volume? You may wish to refine the *Contention Ratio* and *Prop. Of Traffic in Busy Hour* inputs according to a careful analysis of the expected daily usage pattern. *(Beyond the scope of this tutorial.)*

Things that you should have seen and understood

Traffic Calculation

Traffic Volume: Traffic Unit, Traffic per Connection, Traffic Period

Distribution: Busy Days per Year, Prop. Of Traffic in BH, Annual to BH Unit Ratio

Busy Hour Traffic: BH Traffic Unit, Nominal Bandwidth per Connection, Contention Ratio

Exercise 4: Concise inputs and detailed outputs

Now we are going to run the model and draw the first results graphs.

Note: if you haven't saved the model yet, you will be prompted to do so now.

1. Select *[Save and] Run* from the *File* menu (or press <F5>). The model is first checked for consistency and viability. STEM detects a potential problem, and prompts with a *Yes/No* choice, *Model has input data warnings. Continue?*
2. Click *No* to see what happens. A message is displayed in the Editor, warning that our *Service has no Requirements.* Don't worry; we will soon address this!

For now, we just want to review the impact of the service assumptions entered so far.

3. Run the model again, and this time, at the warning prompt, click *Yes* to continue. The model is run and the Results program loads, initially with a blank canvas, within which we will graph some results.
4. Select *Draw…* from the *Graphs* menu. The *Draw* dialog is displayed, with tabs for *Elements*, *Graphs* and *Format*. The first tab lists the *Available* elements as *(Network)* and our service *Broadband connectivity.*
5. Select *Broadband connectivity* in the list, and then click the *Graphs* tab. The service is automatically added to the *Selected* list, and then the *Graphs* tab is displayed, listing the available *Graphs to draw.*
6. Select *Connections* in the list and then press *OK*. A graph is drawn, showing the evolution of the result over the ten-year run period. This should have the same shape as the graph of the *Penetration* input that we drew earlier, but scaled up by the 1000 of the *Customer Base* input. Are the actual numbers what you expect?
7. Right-click the background of the graph and select *Show Separate Table* (or select the same command from the main *Format* menu). A corresponding table is displayed. You should see that there are 20 customers in Y2 (2%) and 150 customers in Y3 (15%). (These may differ if you still have shorter time-period inputs from earlier on.)

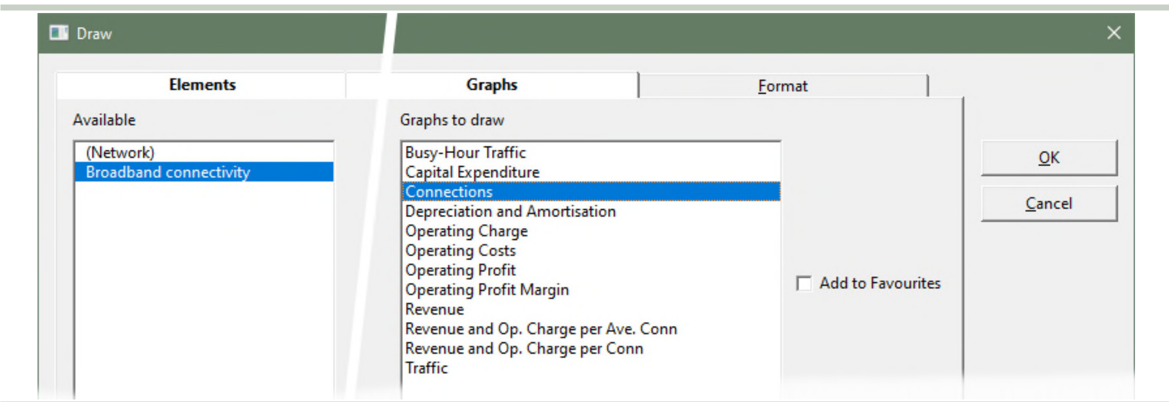


Figure 11: Selecting from the *Elements* and *Graphs tabs in the Draw dialog*

This trivial graph illustrates the difference between the input domain of the Editor and the calculated output domain of the Results program.

We will check the traffic outputs next, and learn two essential shortcuts at the same time:

8. Select *Draw Another…* from the main *Graphs* menu. The *Draw* dialog is displayed, with the same element and graph selections as the last graph drawn.
9. Go straight to the *Graphs* tab, this time select *Busy Hour Traffic*, and press *OK*. The graph is drawn. With a contention ratio of 10, the nominal 100 Mbit/s service will average to 10 Mbit/s per customer in aggregate across the 1000 customers, so the peak traffic should approach 10 Gbits/s. Does your graph reflect this?
10. Select *Draw Similar…* from the main *Graphs* menu (or right-click the background of either graph to access the same command). The *Draw* dialog is displayed, with the same element and graph selections as the current graph.
11. Go straight to the *Graphs* tab again, and now draw *Traffic* (i.e., volume). The graph is drawn. That's a lot of data! The limit should be something like 10 × 60 × 60 / 8 / 1024 / 0.2 × 250 ~ 10 × 0.44 / 0.2 × 250 = 4.4 × 1250 = 5500 Tbytes (i.e., 5.5 Pbytes).
12. Select *Tile* from the *Graphs* menu to tidy the presentation. The three graphs are tiled in the order of most recently active. Try experimenting with this.

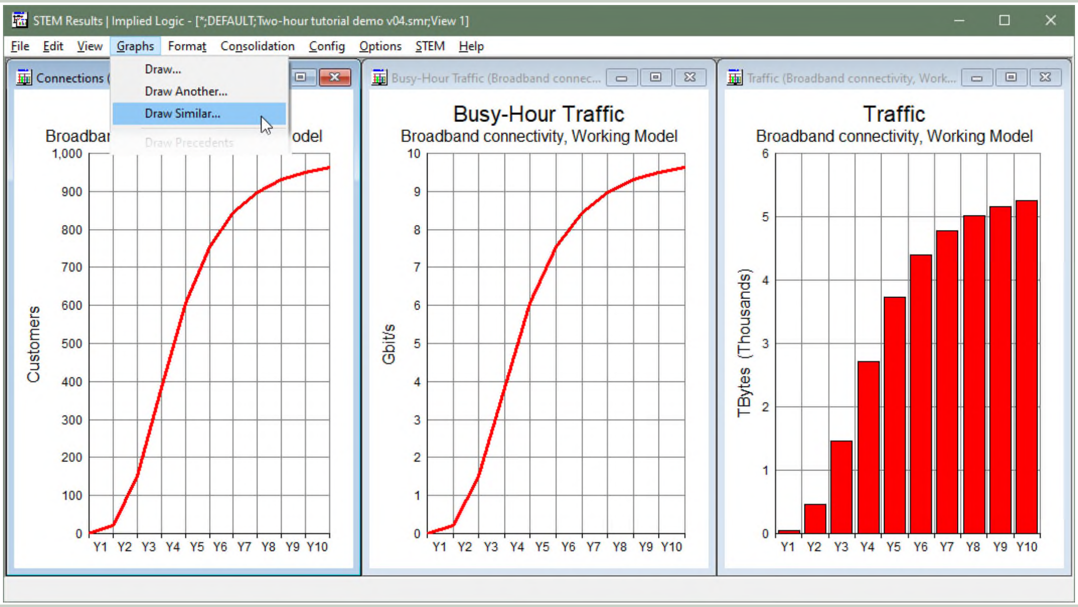


Figure 12: Connections and traffic results for the *Broadband connectivity* service

Things that you should have seen and understood

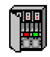
Run, warnings, Results program

Draw, Elements, Graphs

Draw Another, Draw Similar

Connections, Busy Hour Traffic, Traffic

3. Resource *Optical interface*

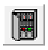

Optical interface

We have projected the demand from customers. Now we must dimension the assets required to deliver the service. The **resource** element in STEM represents one source of cost, and captures the physical parameters and costs associated with one unit of provision.

We will start (as a trivial warm-up) with the *Optical interface*, one of which is required per customer (to connect the subscriber line to the *Access card*).

Exercise 5: Capacity and lifetime; capital cost and maintenance

Always start with the physical characteristics:

1. Click the *Resource* button  on the toolbar, to create the resource element, and then rename it as *Optical interface* (in the same way as we did for the service earlier).
2. Access the *Capacity and Lifetime* dialog from the icon menu.
3. Enter the *Capacity Unit*, *Capacity* and *Physical Lifetime* inputs, as shown below.

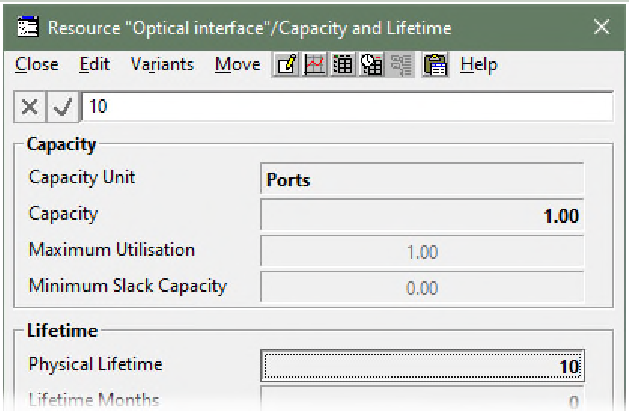


Figure 13: The relevant inputs of the *Capacity and Lifetime* dialog for the *Optical interface* resource

Then add the relevant cost attributes:

4. Access the *Costs* dialog from the icon menu (or directly from the dialog *Move* menu).
5. Enter the *Capital Cost* input as shown below, and press <Enter> to commit the value.
6. Select the *Maintenance Cost* input (or press <Tab> twice).
7. Click in the formula bar (or press <F2>, or even just start typing). The 'tick and cross' buttons are enabled, showing that the formula bar is active.
8. Move the mouse around the dialog. The cursor changes to a 'pointing hand' over other fields which could be referenced in a formula.
9. Click the *Capital Cost* input (i.e., the box where the value `500.00` is displayed). The text `Capital` is inserted into the formula bar as a reference to the *Capital Cost* input.
10. Enter the rest of the formula as shown below, and press <Enter> to commit it. The calculated value is displayed in red because the input is entered as a formula.

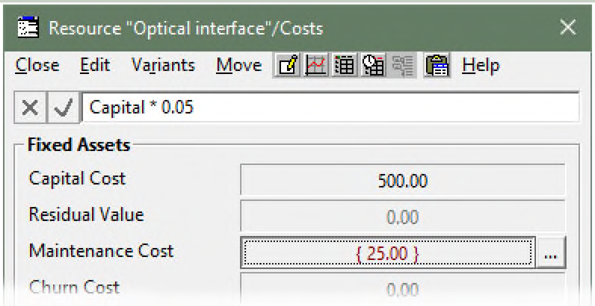


Figure 14: The *Maintenance Cost* entered as a formula in the *Costs* dialog for the *Optical interface* resource

We also need to define the currency for these assumptions:

11. Select the *Global Currency Unit* input in the *Units* section in the same dialog.
12. Enter `USD` as shown below, and press <Enter> to commit the value.

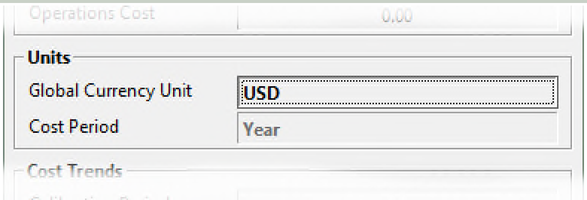


Figure 15: The *Global Currency Unit* input is common to all tariff and cost inputs

This unit is used to label all financial results (which must be aggregated across services and resources in a single currency). You may wish to specify some equipment costs in the currency of an overseas supplier, and this is commonly accomplished with reference to per-element and global *User Data*. *(Beyond the scope of this tutorial.)*

The *Cost Period* input indicates that the *Maintenance Cost* input entered above will be interpreted as an annual cost. We may enter monthly costs for other resources later.

We have created the resource element, and entered the self-explanatory characteristics which pertain to one installed unit. Now, how is this related to the service?

Things that you should have seen and understood

Resource element, Capacity and Lifetime, Costs

Capacity Unit, Capacity, Physical Lifetime

Capital Cost, Maintenance Cost, Global Currency Unit, Cost Period

Exercise 6: Connecting the service to the resource

Unlike this tutorial model, a STEM model typically comprises many separate service and resource elements, the majority of which are not directly related. We will use a separate tool to connect those elements which are, and to specify the basis for that connection. The actionable links thus created are just as important as the elements themselves.

1. Click the *Connect* button on the toolbar (or press <Ctrl+Q>, or select *Connect* from the main *Element* menu). The Editor enters 'connect mode', and the mouse cursor changes to match the image on the *Connect* button.
2. Click once on the icon for the *Broadband connectivity* service. Now a thick grey line is ghosted from the icon to the mouse cursor.
3. Hover over the icon for the *Optical interface* resource, but don't click just yet! A 'green square' appears over the bottom-left of the icon. This quick-link target enables you to avoid unnecessary prompts in the most common link contexts.
4. Click again on the green square. A green arrow is drawn between the elements. This is a *requirement* link.
5. Hover over the green arrow. A popup 'tooltip' describes the link between the elements as a *Requirement* for the resource with *Basis* = `Connections`.

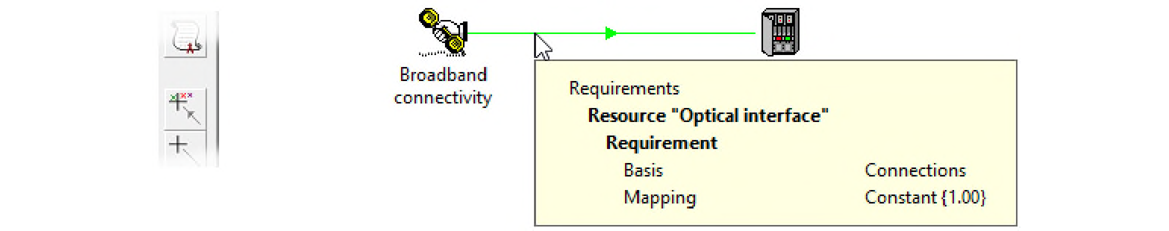


Figure 16: Reviewing the basis of the requirement link from the service to the resource

The green arrow indicates a requirement link and denotes that **each service connection requires one unit of capacity of the resource**. The roll-out for this resource is easy to predict (why?), but the results will be more interesting as we add the other resources.

This most common basis was selected automatically by clicking the quick-link target. We will see later on how to choose and then review other bases, such as Busy Hour Traffic.

Things that you should have seen and understood

Connect button, quick-link target

Requirement link, Basis

Exercise 7: Installed and incremental units; capex and opex results

Now we are going to examine the essential results for the *Optical interface* resource, applying techniques we learned earlier for drawing graphs:

1. Run the model again. (You can just press <F5> in the Editor.) There should no longer be any warnings at this stage, and the Results program is activated directly.
2. Draw the graph *Installed Units* for the resource *Optical interface* the same way that we drew the first graph for the service earlier. The numbers should be the same as the *Connections* result for the service.
3. Use the *Draw Similar* technique to draw the additional graphs *Capital Expenditure* and *Operating Costs*. The latter should match the shape of the *Installed Units* graph, as an annual maintenance cost of USD 25 will arise from each installed unit, and this is the only opex item we have defined so far. As the installed base approaches 1000, the total annual opex at this stage should approach USD 25 000.
4. Tile the graphs again before proceeding.

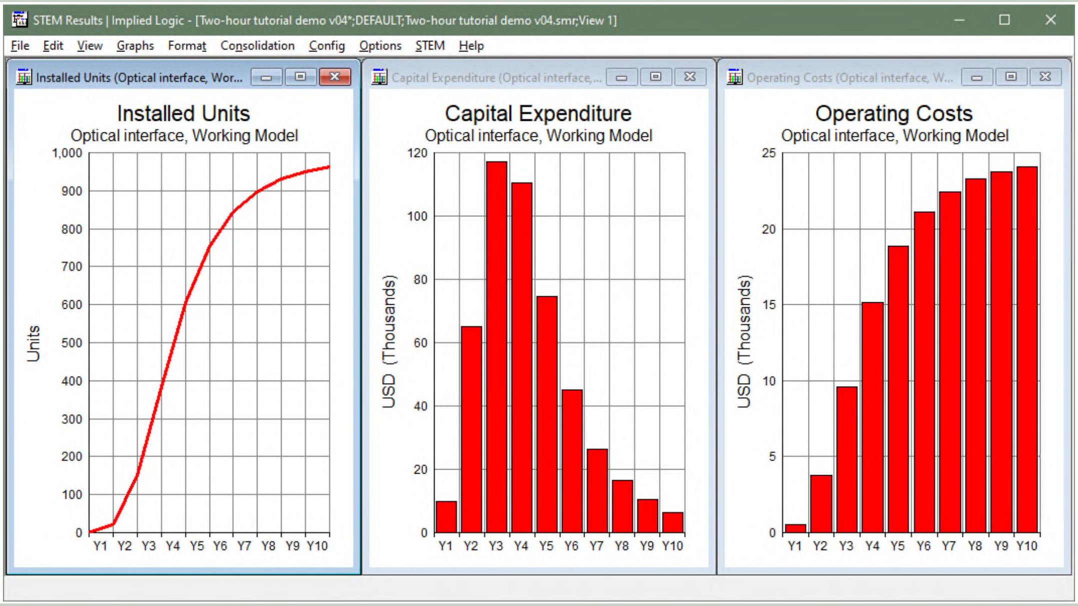


Figure 17: Installation and cost results for the *Optical interface* resource

But what about the *Capital Expenditure* graph? What drives the shape of this graph? We will take this opportunity to learn another useful technique:

5. Right-click the background of the *Installed Units* graph and select *Change Selections…* from the context menu. The *Draw* dialog is displayed again, with the current element and graph selections for this graph.
6. Go straight to the *Graphs* tab, select *Installed and Incremental Units*, and press *OK*. The graph is redrawn showing both results. Evidently, the red line reflects the total, while the blue one indicates how many units are added in a given year; i.e., the 'delta'. You should see that the *Capital Expenditure* graph matches the shape of the blue line, and that the magnitude is scaled by the specified USD 500 per unit. It is also easy to verify that the total capex over the ten years amounts to almost USD 500 000.

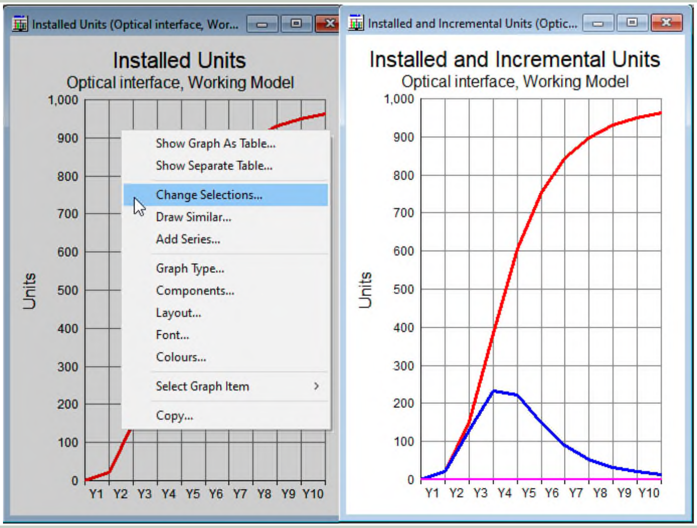


Figure 18: Using the *Change Selections* command to alter the current graph

Note: the Change Selections command is very similar to Draw Similar, except that it alters the current graph, rather than cloning it.

Things that you should have seen and understood

Change Selections

Installed Units, Capital Expenditure, Operating Costs, Installed and Incremental Units

4. Resources *Access card, Access chassis* and *Uplink*

We used the simple example of the dedicated *Optical interface* to learn or re-familiarise ourselves with essential techniques and results in the Results program. Now we are going to add the other physical elements of the infrastructure, and contemplate the provisioning of capacity which is shared, but only between customers at the same site.

Exercise 8: Using a tabular dialog to load, verify or compare assumptions

We are going to load the assumptions for these other resources, all at once, in a table. This is often quicker than doing it piecemeal, and is also a useful technique for checking consistency later:

1. Create three more resources, and name them *Access card*, *Access chassis* and *Uplink*.
2. Select all three icons just created, and then select *Capacity and Lifetime* from the icon menu (for the whole selection). The *Capacity and Lifetime* dialog is displayed with three columns, one for each resource.
3. Enter the *Capacity Unit*, *Capacity* and *Physical Lifetime* inputs, as shown below.

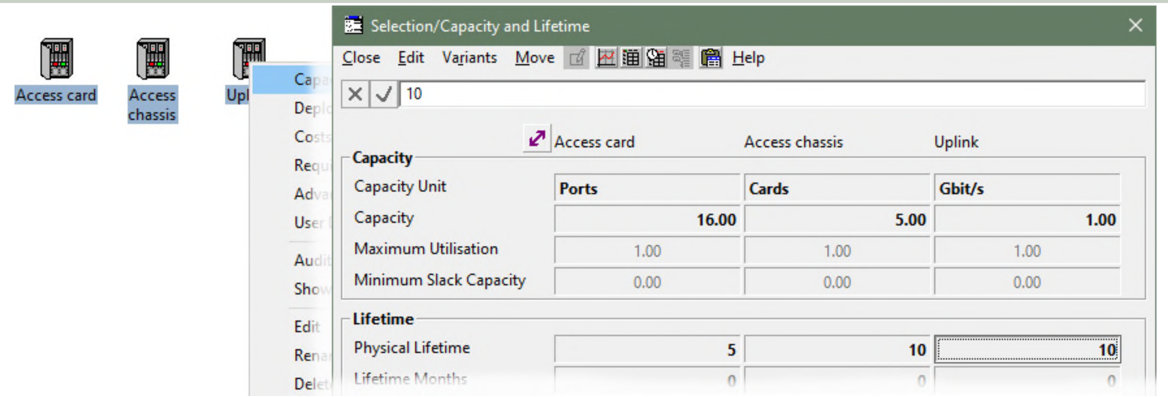


Figure 19: Accessing a multi-column dialog from a multiple selection of icons

Note: the dialog we have used here is a 'selection dialog'. From any such dialog (even for a single element), you can select Table from the Move menu (or just press <Alt+T>) to access a 'tabular dialog' showing inputs for every resource in the model (or service, or whatever).

We will use the same approach to enter the corresponding cost assumptions, but this time it will be helpful to include the *Optical interface* resource too:

4. Select all four resource icons, and then select *Costs* from the icon menu (for the whole selection). The *Costs* dialog is displayed, with four columns this time.
5. First enter the *Capital Cost* inputs as shown below. Don't forget to press <Enter> to commit each value.
6. All the resources have the same '5% of capex' assumption, so it would be great if we could just copy the formula we used before.
7. Select the *Maintenance Cost* input for the *Optical interface* (first column below).
8. Select *Copy* from the dialog *Edit* menu (or press <Ctrl+C>). The formula is copied.
9. Select the *Maintenance Cost* input for the *Access card* (second column below).

10. Select *Paste Formulae* from the dialog *Edit* menu (or press <Ctrl+V>). The formula is copied, but you will see immediately that the value is different. The reference to the original *Capital Cost* input is treated as a ‘relative reference’ because it is ‘within the same element’, and so the copied reference is ‘relativised’ to the *Capital Cost* input of the target resource, which is exactly what we want now.
11. Repeat the last two steps for the *Access card* and *Uplink*.

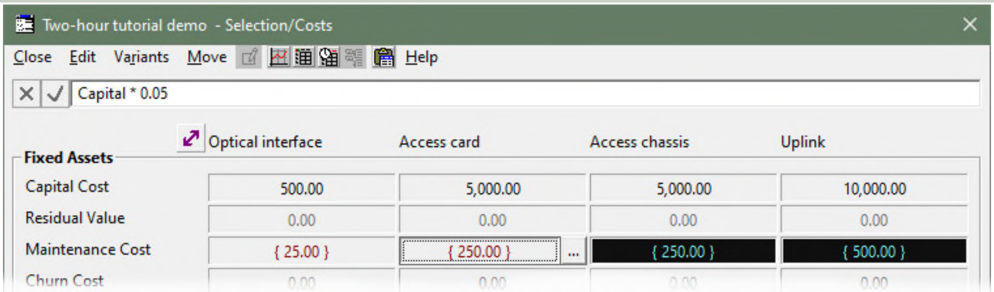


Figure 20: Copying a formula with a relative reference to the *Capital Cost* input for the same element

Note: it is quicker to select the three inputs first (shift-click columns 2–4, as illustrated), and then just paste once, but it is easier to explain the mechanics of a single paste.

Things that you should have seen and understood

Selection dialog, tabular dialog, copy and paste, relative reference

Exercise 9: Connecting the *Access card* and adding it to the results

The next step is to relate the first of these resources to the service. Before we proceed, let’s rearrange the icons in anticipation of the likely connections.

1. Click and drag each resource in turn to arrange them as shown below.
2. Press <Ctrl+Q> to connect the service to the *Access card* resource, being careful to use the ‘quick-link target’ again to pre-select *Basis* = `Connections` (which means that **each customer requires one port**). The service is linked to the resource as illustrated.

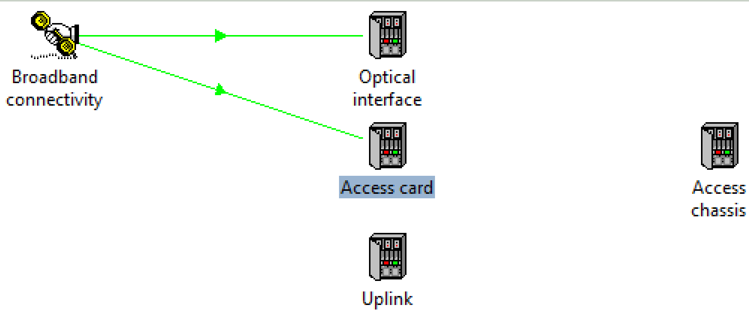


Figure 21: Arranging the icons, and connecting the *Access card* resource

Note: you may consider that each customer requires an interface, that the interface in turn connects to a port on the card, and therefore that we should instead connect from Optical interface to Access card. However, each interface is dedicated to a single customer, so the same outcome is achieved more straightforwardly by connecting from the service directly.

Now let's examine the results for the *Access card* (and learn a useful shortcut too):

3. Run the model, and skip the likely warnings about the lack of requirements for the other resources. The Results program is activated.
4. Locate the existing graph of *Installed and Incremental Units*, and double-click the minor title, *Optical interface*. This is an intuitive shortcut for the change selections command, and the *Change Selections* dialog is displayed.
5. Double-click *Access card* on the left to add it to the *Selected* list, double-click *Optical interface* to remove it, and press *OK*. Results for the *Access card* are now shown. The total approaches '60 something'; i.e., approximately 1000/16.
6. Using a similar technique, but this time retaining the *Optical interface,* add the *Access card* to the *Capital Expenditure* and *Operating Costs* graphs. The combined results are stacked to indicate the total capex and opex.

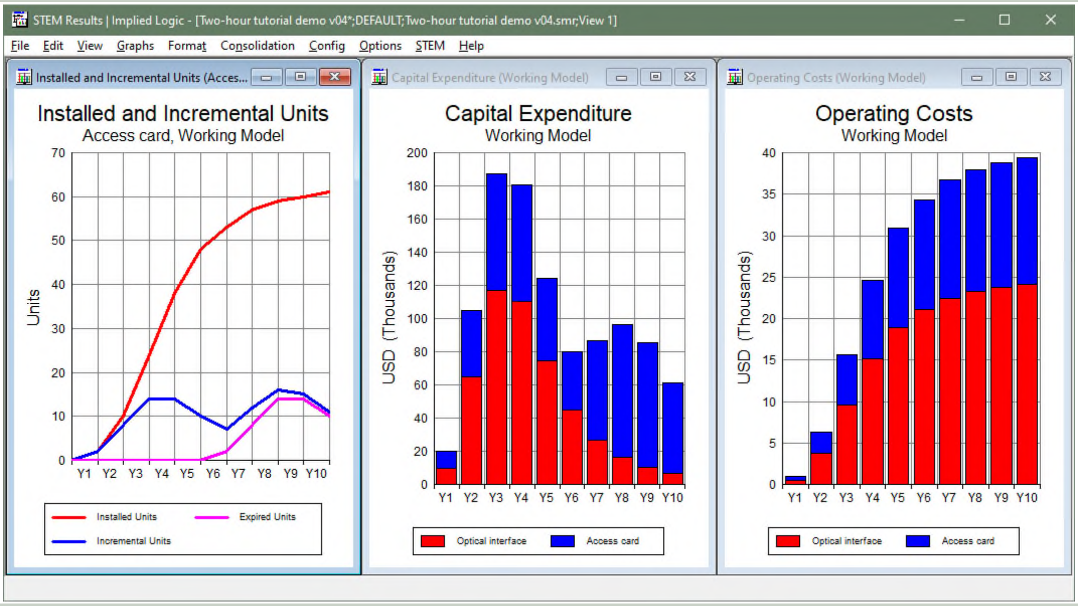


Figure 22: Adding the *Access card* to the results view

You can check the cost numbers in your own time, but let's first take a more critical look at the deployment. Do we really have enough cards? Where are the customers?

Things that you should have seen and understood

Double-click minor title to modify element selection

Exercise 10: Thinking about the geographical deployment of capacity

We are going to draw another graph, one which is possibly the most helpful in assessing whether the installed base is actually realistic:

1. Use the *Draw Similar* command to draw the *Capacities* graph for the *Access card*. The graph shows three results: *Installed Capacity*, *Used Capacity*, and *Slack Capacity*.
2. Draw a separate table to examine the numbers. The *Used Capacity* simply reflects the number of customers, while the *Installed Capacity* appears to be just this number rounded up to the nearest multiple of 16. The *Slack Capacity* confirms that there are always fewer than 16 slack ports. Is that realistic? Where are these ports located?
3. Let's also draw the graph *Utilisation Ratio* for the same resource. You should see that the apparent deployment is highly (and suspiciously) efficient!

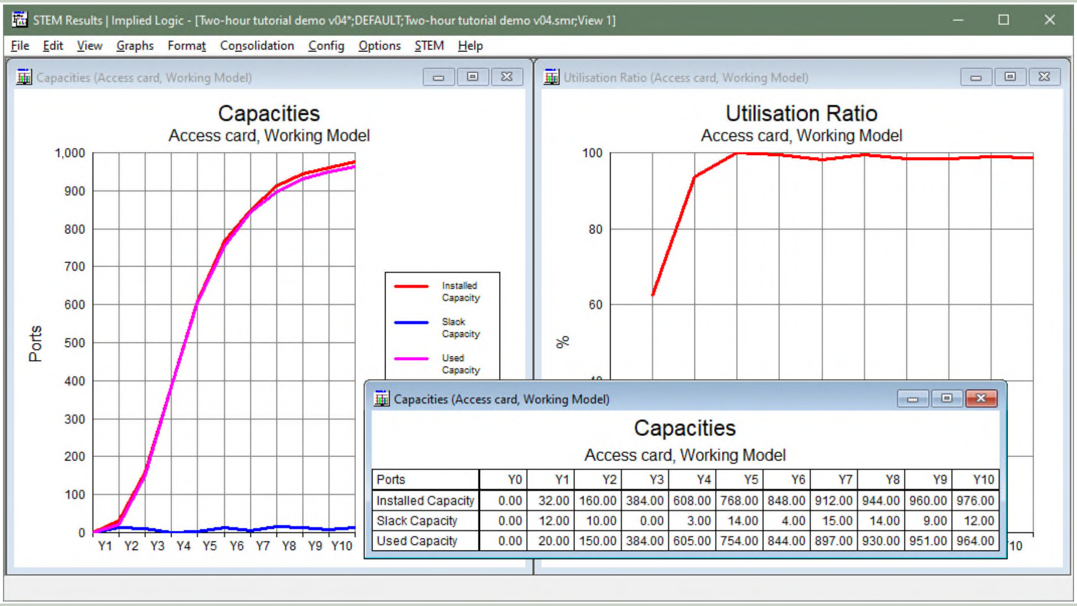


Figure 23: *Capacities* and *Utilisation Ratio* graphs for the *Access card*

A common response to these results is to say that an engineer would always provision 'a few extra ports' as contingency for imminent additions. Indeed, there are options which support this approach in the *Capacity and Lifetime* dialog (see below) in the context of an individual site. However, even without such a provisioning strategy, there is a more fundamental reality about where capacity is required. The plan is to offer service in the vicinity of 25 separate sites, and so capacity must be provided at each site.

As soon as a first customer presents at a given site, a first unit is required at that site (or it might be deployed there as part of an initial deployment prior to launch). Once there are many customers, it is unrealistic to expect these to be coordinated in multiples of 16 at each site. At any point in time, the required installation will leave something in the range of 0–15 slack ports **at each site** (i.e., up to 375 slack ports in total). The current results reflect what you would expect for a single-site business. We need to make STEM aware of the number of sites, and what this means for the *Access card* resource.

Note: press <F1> in the Capacity and Lifetime dialog to access the help for the Maximum Utilisation and Minimum Slack Capacity options. (Beyond the scope of this tutorial.)

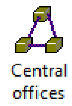
Things that you should have seen and understood

Capacities: Installed Capacity, Used Capacity, Slack Capacity

Utilisation ratio

5. Location *Central offices*

The third, crucial consideration for business planning, after service demand and resource capacity, is the number of discrete locations where those resources must be deployed in order to provide capacity to customers (wherever they actually are).



The location element in STEM is just a number (or time series) which captures the scope of this characteristic, geographical diversity just as a number of sites. It is not a visual construct and does not 'pin resources to a map'. The cost of these individual resources is assumed to be the same, wherever they are located; from a financial perspective, what matters is their number, not their specific location.

Exercise 11: Geographical scope without detail

The first step is to create the location element:

1. Click the *Location* button  on the toolbar, in order to create the location element, and then rename it as *Central offices*.
2. Access the *Details* dialog from the icon menu.
3. Enter the *Unit* and *Sites* inputs as shown below.

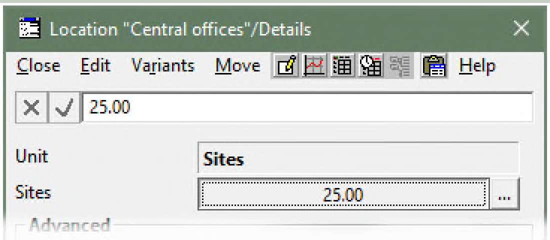


Figure 24: The location element is basically just a number

Although it is true that the customers are clustered around these 25 sites, it is the dimensioning of the resources actually located at these different sites which is impacted, not the service numbers. The round-up to a whole number of units, according to the resource capacity, must be simulated as if there were 25 separate 'piles of cards'. Therefore, we must connect the location element to the resource, not the service:

4. Press <Ctrl+Q> to connect the location to the *Access card* resource.
5. This time there is no 'quick-link target' as this connection can mean only one thing. A light-blue arrow is drawn between the elements, as illustrated below. This is a *location* link.

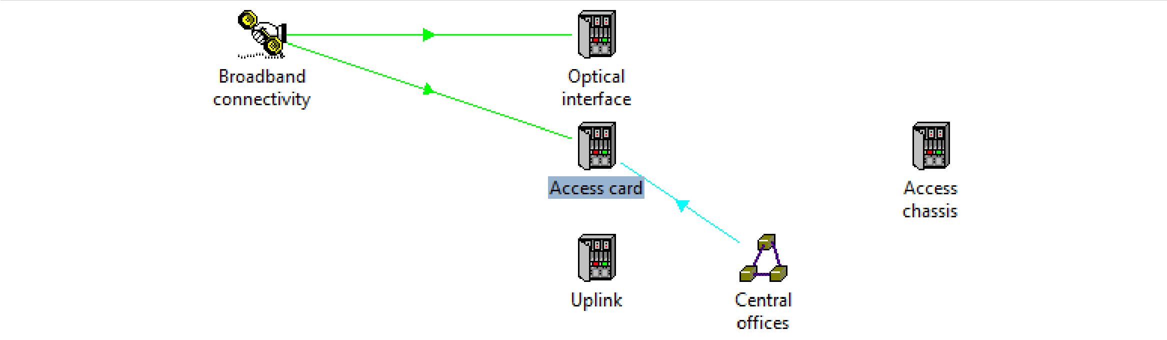
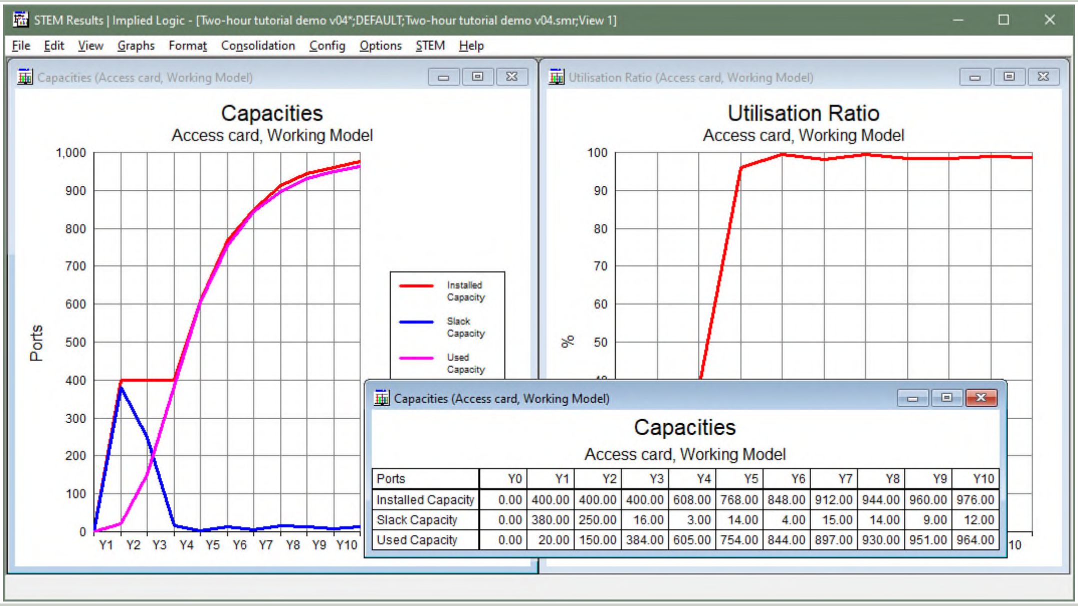


Figure 25: Connecting the location to the *Access card* resource

Let's see how this impacts the results for the *Access card*:

6. Run the model, and again skip the likely warnings about the lack of requirements for the other resources. The Results program is activated.
7. You will see immediately that more equipment is installed at the start, which in turn triggers a significant cashflow hit in Y1.



Ports	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9	Y10
Installed Capacity	0.00	400.00	400.00	400.00	608.00	768.00	848.00	912.00	944.00	960.00	976.00
Slack Capacity	0.00	380.00	250.00	16.00	3.00	14.00	4.00	15.00	14.00	9.00	12.00
Used Capacity	0.00	20.00	150.00	384.00	605.00	754.00	844.00	897.00	930.00	951.00	964.00

Figure 26: *Capacities* and *Utilisation Ratio* graphs for the *Access card* after connecting the location

Evidently, 400 ports = 25 × 16 ports = 25 units = one unit per site, which is what you need to offer service at each location. There is much more slack capacity at the start because the deployment leads the demand.

The initial results are now more prudent, but what happens when that slack is taken up as customer numbers grow? Beyond the first 25 units, it appears that there is still only slack at one site, somewhere in the range 0–15 ports inclusive. In reality, we would expect ongoing customer additions at each site, and for there to be slack 'in every pile'. Across all 25 sites, this would be far in excess of that shown above.

The location connection by itself only addresses the principal cashflow correction at the start by default. Further options exist with greater impact on the subsequent behaviour, and which will enable us to 'fix' the ongoing results; i.e., make them more realistic.

Things that you should have seen and understood

Location element, Details, Unit, Sites

Location link

Exercise 12: Simulating variability between sites

Now we are going to think more carefully about how the numbers of customers are spread out between the various locations where equipment is installed:

1. Double-click the light-blue, location link (or select *Deployment* from the icon menu for the *Access card* resource). The *Deployment* dialog is displayed, as shown below, with the input *Distribution* = `One for One` by default. This is what determines the minimal results we have seen so far. The only impact is that, literally, at least one unit will be installed for each site (as soon as there is a non-zero demand). We will explore two of the available alternatives as follows.
2. Click the *Distribution* field, and select `Homogeneous` from the drop-down in the formula bar, as illustrated below. This setting indicates that the customer demand should be assumed to be split evenly between all of the sites. When the first card is full at every site, then another must be installed at every site, and so on.

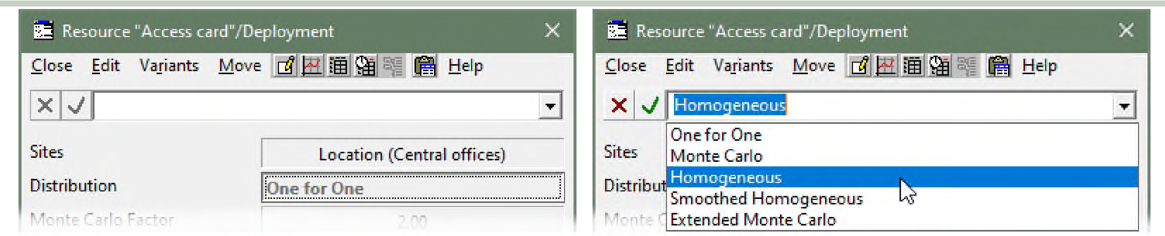


Figure 27: Changing the *Distribution* input in the *Deployment* dialog for the *Access card* resource

3. Re-run the model (still skipping warnings) and review the updated results.

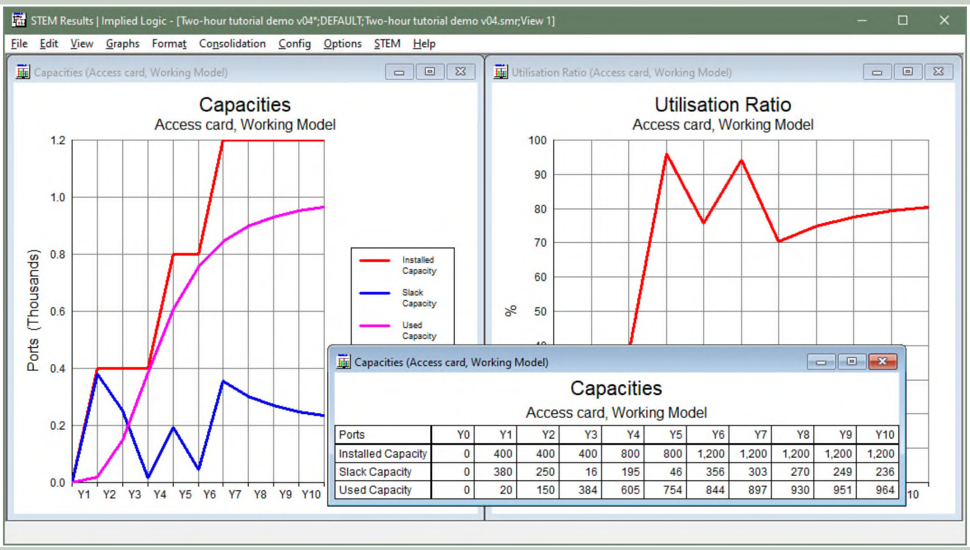
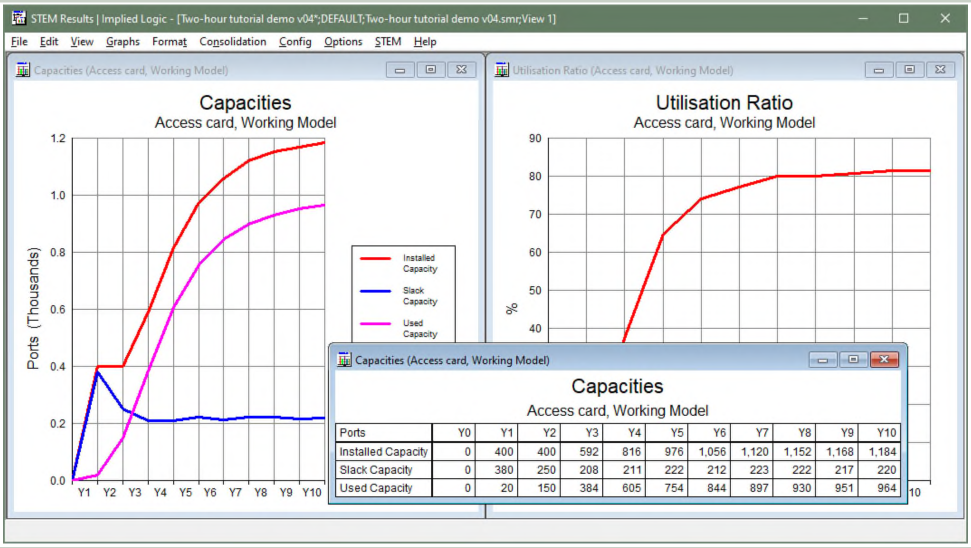


Figure 28 *Capacities* and *Utilisation Ratio* graphs with *Distribution* = `Homogeneous`

As soon as the demand exceeds the initial 400 ports, another 25 units are installed. Such a distribution is unlikely in real life, but is useful as an extreme example.

In practice, it is likely that there will be ‘hot’ sites where the initial capacity is exceeded sooner than Y3, and other ‘cooler’ sites where this happens later. At any given site, as demand increases, the number of slack ports will vary in the range 0–15 inclusive. Given the likely variation in timing across sites, a fair estimate of the slack capacity required at any point in time will amount to ‘half a unit’ per site; i.e., 200 ports. This would have the required installation tracking some way above the pink line for *Used Capacity*, cutting a more gradual path compared to the simplistic red steps shown above.

4. Switch back to the Editor, and set the input *Distribution* = `Extended Monte Carlo`.
5. Re-run the model (still skipping warnings) and review the impact on the results.



Ports	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9	Y10
Installed Capacity	0	400	400	592	816	976	1,056	1,120	1,152	1,168	1,184
Slack Capacity	0	380	250	208	211	222	212	223	222	217	220
Used Capacity	0	20	150	384	605	754	844	897	930	951	964

Figure 29: *Capacities* and *Utilisation Ratio* graphs with *Distribution* = `Extended Monte Carlo`

As you can, this setting maintains a roughly constant overhead of just over 200 slack ports beyond the number of ports actually in use at any given point in time. Beyond the initial constraint of one unit per site, the used capacity is proportional to the demand, whereas the slack capacity is more or less proportional to the number of sites. This is the most prudent approach; so, if in doubt, use `Extended Monte Carlo`.

(Beyond the scope of this tutorial.)

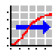
By all means experiment with the other two options for the *Distribution* input:

- `Monte Carlo`: as above, but without the initial constraint of one unit per site; suitable if a ‘just-in-time’ deployment model is feasible for the first customer at each site
- `Smoothed Homogeneous`: like `Homogeneous`, but also without the initial constraint of one unit per site, but only really included as an academic example for completeness.

Things that you should have seen and understood

Deployment, Distribution, One for One, Homegeneous, Extended Monte Carlo

6. Transformation *No of access cards*



No of access cards

Now that we have worked out how many access cards are needed, the next step is to determine how many access chassis are required to house those cards. The transformation element in STEM provides the necessary glue to establish chains of dependent drivers in a model.

Exercise 13: Customers need ports; cards need slots

There are various reasons why, in this case, we will not seek to drive the access chassis calculation directly from the service:

- as we have seen above, there is a non-trivial, deployment calculation for the number of cards, and the chassis must be provisioned to match this actual number
- the number might also be boosted for contingency, as mentioned in *Exercise 10* above
- a chassis might need to accommodate other types of cards.

The latter points are beyond the scope of this tutorial, but the first is good enough.

Instead, we must drive the chassis calculation by the number of cards installed.

Intuitively, we wish to connect the *Access card* and *Access chassis* elements. However, a new transformation element is required to reference the *Access card* as a demand driver, and, crucially, to define more specifically what result is intended. This structure also allows for richer, secondary drivers, as we shall see in *Exercise 18* below.

1. Click the *Transformation* button on the toolbar, to create the element, and then rename it as *No of access cards*.
2. Access the *Input and Transformation* dialog from the icon menu. You will see that this default transformation comprises an *Output Unit*, a *Multiplier*, and an *Input*. These fundamentals are common to all transformation types.
3. Click *Type* on the dialog menu. The drop-down menu, as illustrated below, indicates that this is a *Multiplier* transformation, and lists many further possibilities.
4. Press <Esc> (or click `Multiplier`) to leave the *Type* as-is.

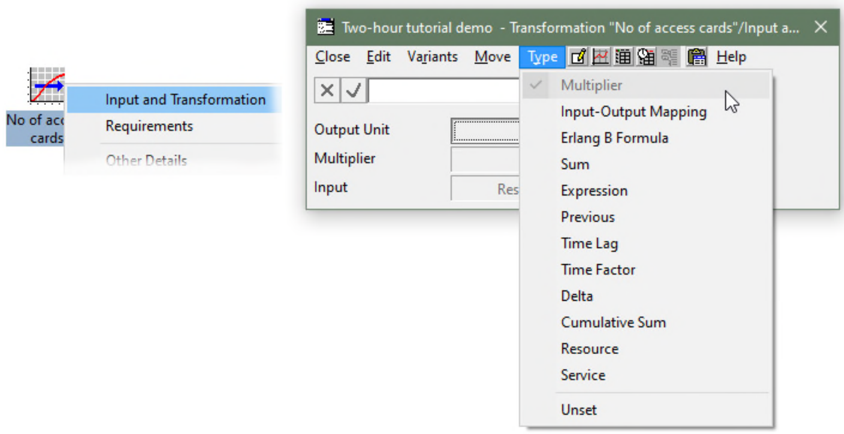


Figure 30: *Output Unit*, *Multiplier*, and *Input* are common to all transformation types

Note: the other transformation types are beyond the scope of this tutorial.

We will make the relevant connections directly between the icons, as before.

5. Arrange the transformation, relative to the access resources, as shown below.
6. Press <Ctrl+Q> to connect the *Access card* resource to the transformation, again using the 'quick-link target' which appears over the transformation icon to pre-select *Basis* = `Installed Units`. A pink, *transformation* link is drawn between the elements.
7. Hover over the pink arrow. A popup tooltip describes the link between the elements. This detail is also reflected in the *Input* field in the *Input and Transformation* dialog.
8. Double-click the pink arrow (or double-click the *Input* field in the *Input and Transformation* dialog) to access the *Input* dialog. Click the drop-down in the formula bar to review the other options for the *Basis* input (which are offered directly if you skip the 'quick-link target' in *step 6* above).
9. Press <Esc> (or click `Installed Units`) to leave the *Basis* as-is. *The other alternatives are beyond the scope of this tutorial.*
10. Set the *Output Unit* = `Cards` (for clarity in the results, as always).
11. Leave the *Multiplier* = `1.0` as per default.

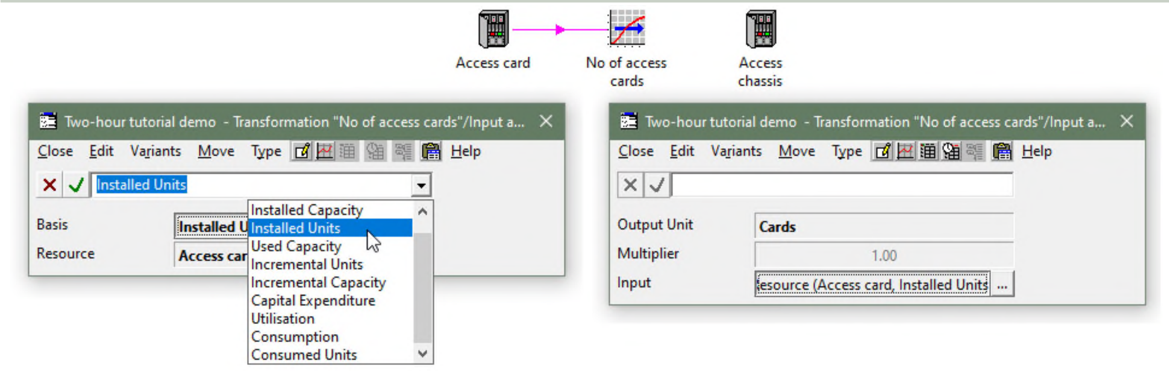


Figure 31: The connection from the *Access card* resource identifies *Installed Units* as the demand driver

12. Press <Ctrl+Q> to connect the transformation to the *Access chassis* resource. There is no 'quick-link target' this time, and a prompt appears for which resource input to set, with options *Requirements* (the default), *Sites* and *Planned Units.*
13. Click *OK* (or press <Enter>) to select *Requirements.* The transformation is linked to the resource as illustrated. *The other options are beyond the scope of this tutorial.*

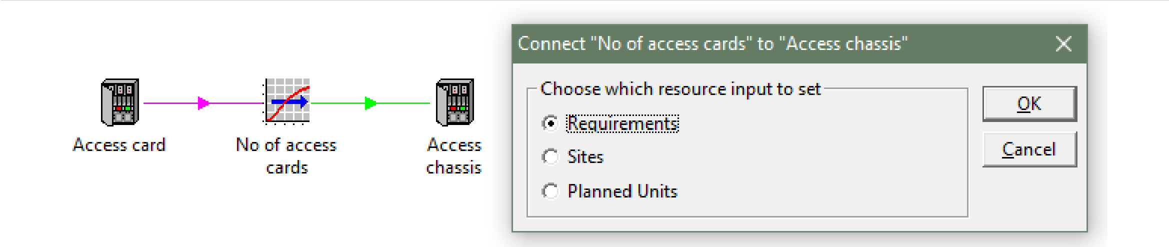


Figure 32: The connection to the *Access chassis* resource is a familiar requirement link

This structure means that **each installed unit of the card requires one slot of a chassis**.

Things that you should have seen and understood

Transformation element, Input and Transformation
Output Unit, Multiplier, Input; Type; Basis, Resource
Choose which resource input to set

Exercise 14: Location is not implied by requirement

Now let's look at the results:

1. Run the model. (Still skip the warnings about the last resource which we haven't connected yet.) The Results program is activated.
2. Change selections for the existing *Capacities* and *Utilisation Ratio* graphs to show the *Access chassis* instead of the *Access card*.
3. Draw *Installed Units* as a table for *Optical interface*, *Access card* and *Access chassis*, using the option in the corner of the *Draw* dialog to *Show Graph as Table*.
4. Add *Access chassis* to the stacked *Capital Expenditure* and *Operating Costs* graphs.

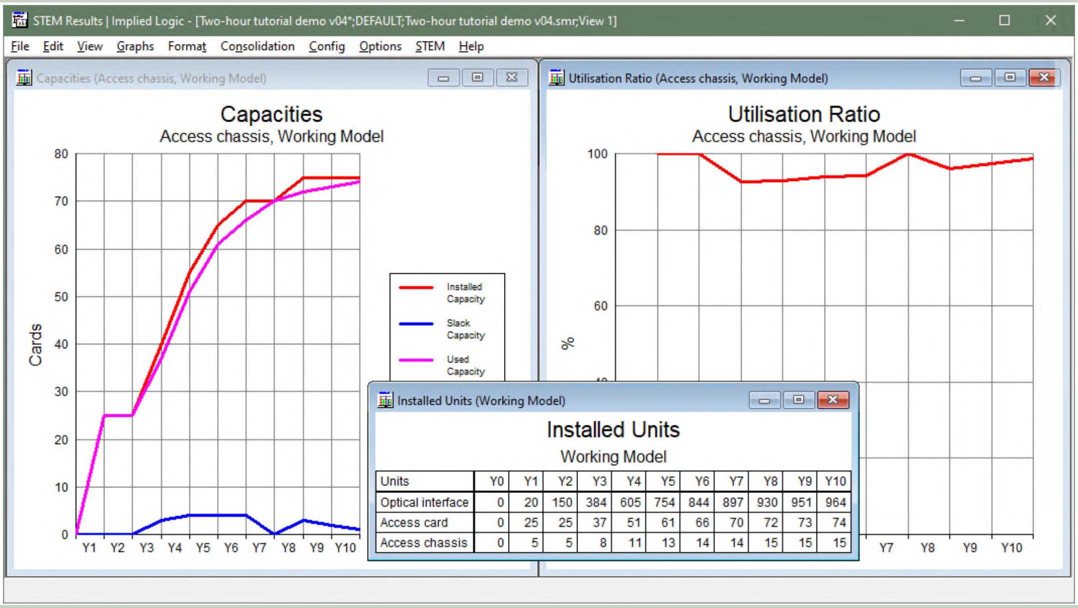


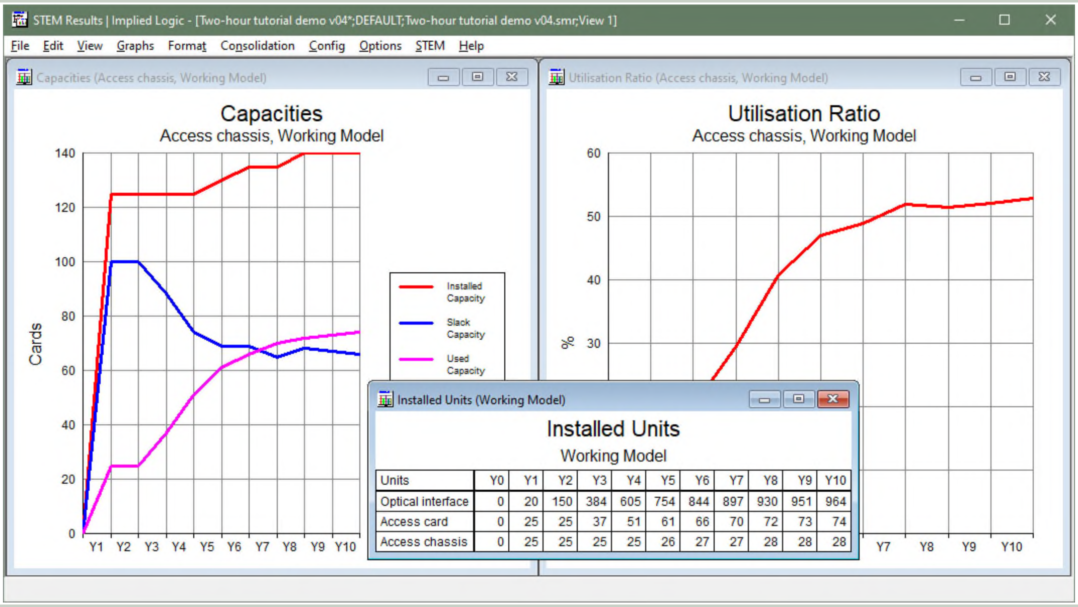
Figure 33: *Capacities* and *Utilisation Ratio* for the *Access chassis*, plus inventory of resources

The *Capacities* and *Utilisation Ratio* graphs may be suspicious, but it is immediately clear from the *Installed Units* table (inventory) that there are only have five chassis for 25 cards in Y1, and still only 15 chassis in Y10, whereas we should have chassis at all 25 sites.

STEM is aware that the cards are in different places, but not that the chassis are too. Collocation cannot be inferred from the dependency between the elements; e.g., some management software could be virtualised and shared in a separate data centre location.

Therefore, we must tell STEM about the deployment of the chassis too:

5. Connect the location to the *Access chassis* resource, as we did for the *Access card*.
6. Double-click the location link and set *Distribution* = `Extended Monte Carlo`.
7. Re-run the model (still skipping warnings) and review the impact on the results.



Units	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9	Y10
Optical interface	0	20	150	384	605	754	844	897	930	951	964
Access card	0	25	25	37	51	61	66	70	72	73	74
Access chassis	0	25	25	25	25	26	27	27	28	28	28

Figure 34: One *Access chassis* per site at the start, and then a few more later at the busier sites

Now there is a chassis at every site from Y1, with very low utilisation at the start, and then just a few more are added from Y5 onwards. There is an average of c. three cards per site in Y10 (74/25), but the simulation suggests a few of the busier sites actually need a sixth card, and thus an extra chassis. In contrast, some of the quieter sites might only need one or two cards, but still need one, under-utilised chassis.

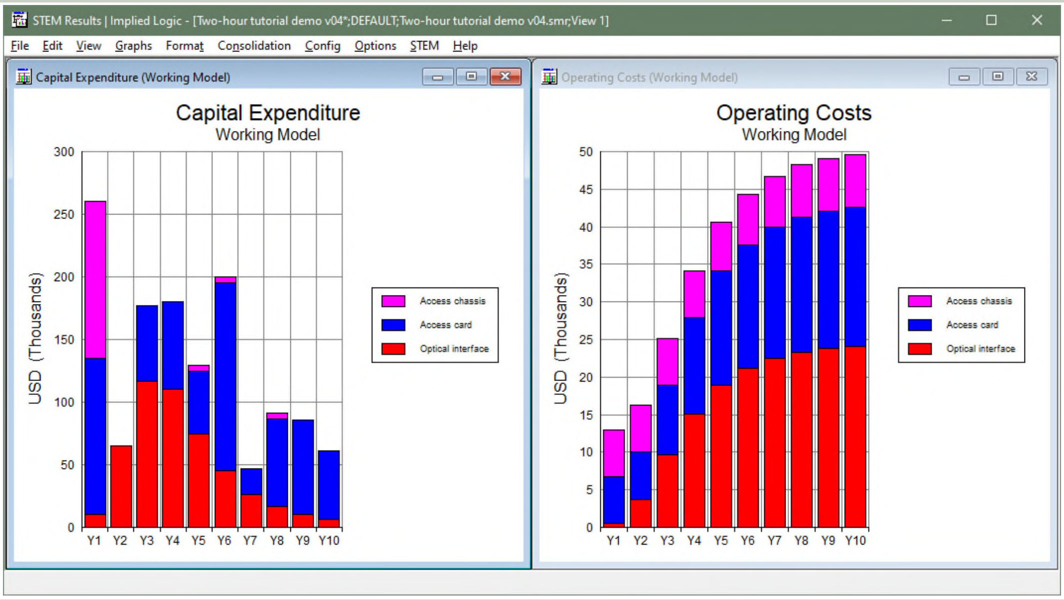


Figure 35: Stacked *Capital Expenditure* and *Operating Costs* graphs for all the resources so far

You might wonder why the location for the cards is not based on the number of chassis. In this example, the number of *Central offices* is a given, and more than one chassis might be needed eventually at some sites, as shown above. An extra chassis should not demand further cards; it is the cards that need a chassis, not the other way round!

The remaining commentary explores refinements beyond the scope of this tutorial.

The number of locations is really part of the marketing plan. With more time, we could compare the merits of a phased roll-out; perhaps launching at only five sites (to limit the initial capex), and then adding the others gradually. This would spread the capex, but also delay revenue, so it is not obvious which would yield the greater return.

With more data about the distribution of potential customers, we could also model the dimensioning of cards and chassis at individual sites, rather than estimating the impact with the deployment feature. This can be done very efficiently and consistently using a feature called *template replication*.

With such an approach, the choice of specific locations could be tailored such that none would ever require more than one chassis. However, the number and location of viable sites may not bear any correlation to the proximity of potential customers, so this may not be possible in practice.

As it is, the aim for this short exercise is to keep things simple, and to estimate the impact of geography, as you might for an initial business plan, rather than being specific about where the chassis are. We might re-visit the ideas above in a future sequel!

Things that you should have seen and understood

Show Graph as Table

7. Resource *Uplink*

All of the resources we have considered so far have been dimensioned on a discrete basis where one customer requires one interface and one port, and one card requires one slot. Each step reflects the physical path into the network, but the requirement is logical, in the sense that a customer is either connected, or not; there are no partial connections.

The *Uplink* is different, because the requirement is a function of how much a customer uses the network, which is also likely to vary over time.

Exercise 15: Equipment capacity based on collective use of the service

Customers connect to use a network (more generally, a business function), and their use can be measured in two different ways, namely the *Traffic Volume* and *Busy Hour Traffic*. These results correspond to the cumulative volume of traffic carried (or work done) in a period, and the fastest, instantaneous rate at which the network must carry that traffic (or do that work). Please review the relevant service inputs, as described in *Exercise 3*.

In this simple exercise, we are concerned only with the *Busy Hour Traffic*, and 'how fast the *Uplink* needs to go'. We will revert to the service, because the other resources have no awareness of the traffic. We will learn how to create a requirement for the *Uplink* on a basis other than `Connections`:

1. Press <Ctrl+Q>, and then click the service to start the 'connection'.
2. This time, avoid the quick-link target which appears when you hover over the *Uplink* resource, and instead click somewhere else on the background of the icon. A prompt appears to choose the service *Basis* for the requirement, with numerous options as illustrated below.
3. Select `Busy Hour Traffic` and click *OK* (or press <Enter>). *The other options are beyond the scope of this tutorial.* The service is linked to the resource. This is still a requirement link, but the thicker green arrow highlights the non-default basis.

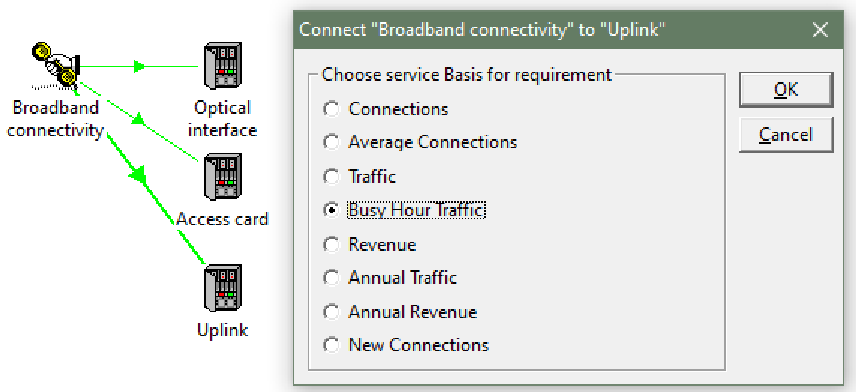


Figure 36: The thick green arrow indicates a requirement on the basis of `Busy Hour Traffic` (non-default)

This option signifies that **each unit of busy-hour traffic requires one unit of capacity of the resource**.

One might expect the distribution of traffic to be somehow linked to the simulated distribution of customers between sites. However, there will be an independent variation in traffic between customers too. At this level of detail, there is no justification for any more subtle approach than just simulating the traffic distribution independently.

With more time, and sufficient data, the option remains to model the sites individually as alluded to at the end of Exercise 14 above.

4. Connect the location to the *Uplink* resource.
5. Double-click the location link and set *Distribution* = `Extended Monte Carlo`.

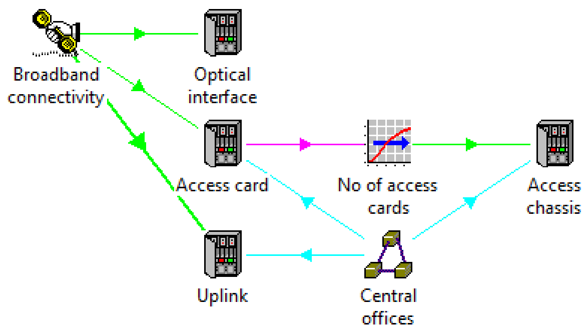


Figure 37: The four principal, connectivity resources are now all linked directly or indirectly from the service

Things that you should have seen and understood

Choose service basis for requirement

Exercise 16: Complete inventory of the fixed assets

Now we are ready to review the results for the *Uplink* resource:

1. Run the model again. (There should not be any warnings at this stage!) The Results program is activated.
2. As we have done before, change selections for the existing *Capacities* and *Utilisation Ratio* graphs to show the *Uplink* instead of the *Access chassis*.
3. Add *Uplink* to the stacked *Capital Expenditure* and *Operating Costs* graphs, and also to the table of *Installed Units*.
4. If you need to redraw any of these, you can simplify the element selection by filtering the available elements to *Type* = `Resources` in the *Draw* dialog.

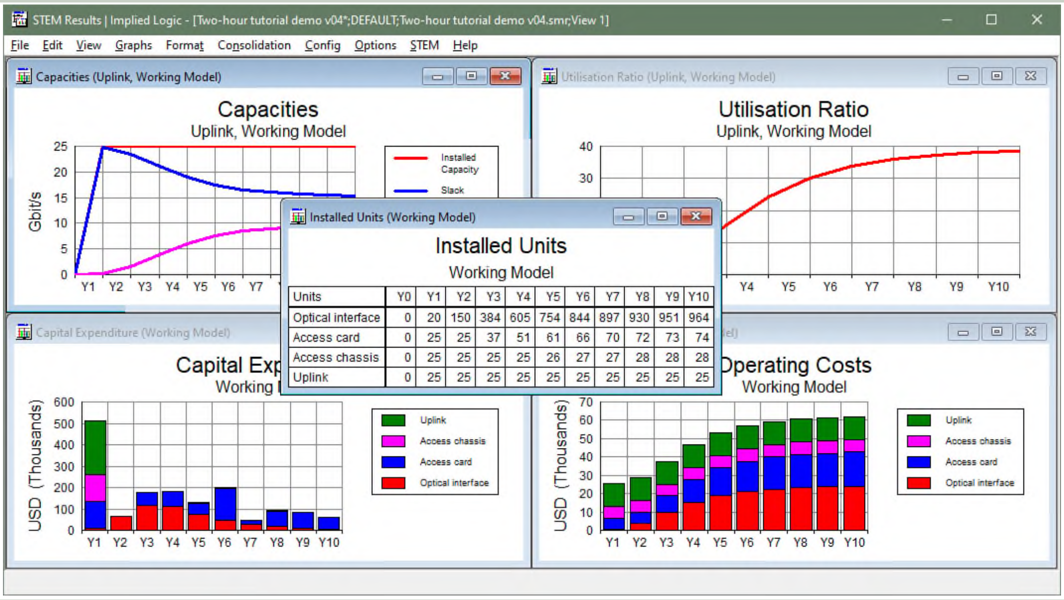


Figure 38: One *Uplink* resource per site is sufficient throughout on the current assumptions

From the *Capacities* graph, it is clear that the demand is not quite 10 Gbit/s, even at the end of the run period, and this is less than half the installed capacity, as reflected in the *Utilisation Ratio* remaining below 40%. This is not sufficient for the simulation to trigger any extra units, so the *Installed Units* remains at 25 throughout; i.e., one per site.

If your results don't match those shown above, then flick back through the other exercises and see if you can pick up where your results first differed from those shown here.

Now we have a complete inventory of the principal fixed assets. In the next two sections, we are going to add some overhead costs, before establishing a viable pricing point for the service.

Things that you should have seen and understood

Filter available elements by type in the Draw dialog

8. Resource *Space* and transformation *Space required*

We have modelled the direct costs of the solution, but the equipment we have considered so far must be housed somewhere. This itself will incur a cost, if the facility is a third-party location, or at least an internal cross-charge if it is owned by the same business.

As with the original service and resource elements, we make a clear distinction between the 'cost of a thing' and the 'requirement (or driver) for that thing'.

Exercise 17: How space is priced

Overheads are just another kind of resource, and the assumption we are given for space is that the cost per (one) sq m amounts to USD100 per month. There are two differences with how the previous resources were calibrated:

1. Create a new resource, and name it *Space*.
2. Open the *Capacity and Lifetime* dialog, and then enter the *Capacity Unit* and *Capacity* inputs as shown below.
3. The *Physical Lifetime* can be ignored as we are not considering any one-off costs.
4. Open the *Costs* dialog, and enter *Rental Cost* = `100.0` (under *Leased Facilities*).
5. This USD 100 is stated as a monthly cost, so we must set *Cost Period* = `Month` (under *Units*) instead of the default `Year`. (Maintenance costs for the previous resources were defined as 5% per annum.)

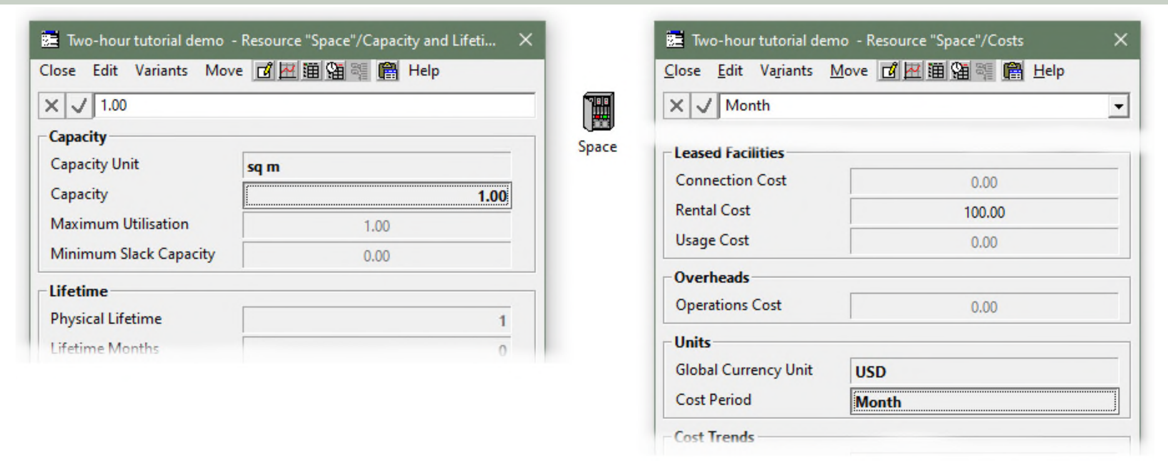


Figure 39: The unit capacity for pricing, and the corresponding unit cost per month, for resource *Space*

For simplicity, we will assume that we are charged only for the space we use at each site, and that the operator will round-up the total bill to a whole number of square metres.

The alternatives below are beyond the scope of this tutorial.

If the space should be charged in whole sq m at each site, then the resource of the sq m 'cannot be shared'. A location link should be added, as we have done before.

Alternatively, if there should be no rounding at all (e.g., an efficiently priced internal cross-charge), then use the *Usage Cost* attribute instead, as this is charged in proportion to used capacity, rather than the installed capacity which determines the *Rental Cost*.

Things that you should have seen and understood

Leased Facilities: Rental Cost, Usage Cost; Units: Cost Period

Costs charged in proportion to installed capacity or used capacity

Exercise 18: The requirement for space

The stated assumption is that each chassis requires 0.5 sq m of floor space. The other resources are either contained within or placed upon the chassis, so the space required is just a function of the number of chassis installed. The way this requirement is captured is a generalisation of the technique we used in *Exercise 13* to drive the chassis itself:

1. Create a new transformation, and name it *Space required*.
2. Connect from the *Access chassis* resource to the transformation, using the quick-link target again to pre-select *Basis* = `Installed Units`.
3. Open the *Input and Transformation* dialog. The *Input* is already wired up.
4. Set the *Output Unit* and *Multiplier* inputs as shown below. (We didn't need to set the *Multiplier* last time because each card required exactly one slot in the chassis.)
5. Connect on from the transformation to the *Space* resource, and choose *Requirements* as the resource input to set as before.

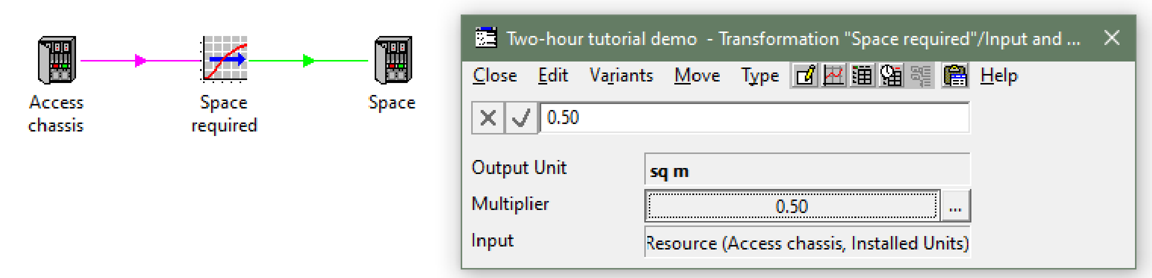


Figure 40: The *Multiplier* input governs the overhead in sq m per chassis, independent of the pricing unit

This means that **each installed unit of the chassis requires 0.5 sq m of capacity from the *Space* resource**.

We can quickly check the arithmetic:

6. Run the model. The Results program is activated.
7. Draw the graph *Instantaneous Output* for the transformation, *Space required*. You should see that this starts at 12.5 sq m (0.5 sq m × 25 units), and then gradually increases to 14 sq m as additional chassis are installed.
8. We should expect the monthly opex for *Space* to rise to USD 1400 (USD 100 × 14), corresponding to an annual opex of USD 16 800.
9. Add *Space* to the stacked *Operating Costs* graph. (Why not *Capital Expenditure*?)
10. Show as a separate table to check the numbers. How many sq m are being charged for in Y1?

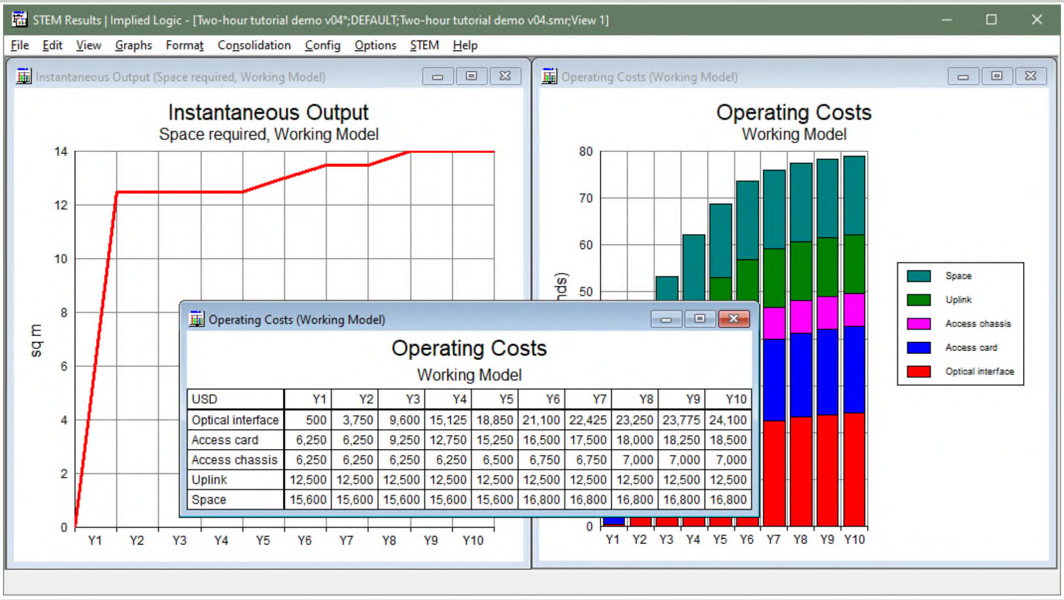


Figure 41: The opex for *Space* is not quite proportional to the *Space required* in sq m

Things that you should have seen and understood

Multiplier, Instantaneous Output

9. Resource *Power* and transformation *Power required*

The equipment we have just housed also requires power. We will add similar elements as we did for space, and the modelling approach will be very familiar in outline. However, what we are really paying for here is energy which, unlike space, gets **used-up in time**.

Exercise 19: How energy is priced

While the static quantity of floor space has a simple price per month, energy is typically priced by the kWh. Two additional concepts are required to model this authentically:

- A **consumable resource** element in STEM is very much like a regular resource, except that its capacity is understood to be used-up in time, as opposed to remaining static like the capacities of the persistent equipment – see *Capacity Mode* below.
- An optional **time factor** allows the cumulative energy consumption in a period to be inferred from the instantaneous power requirement of the equipment, regardless of whether the model runs in years, quarters or months.

We will proceed much as we did in the previous section:

1. Create a new resource, and name it *Power* (or *Energy* if you prefer).
2. Open the *Capacity and Lifetime* dialog, and enter the inputs *Capacity Unit* = `kWh` and *Capacity* = `1.0`. (The *Physical Lifetime* can be ignored in this context.)
3. Locate the *Capacity Options* section in the same dialog, and then enter the inputs *Capacity Mode* = `Consumable`, *Time Factor* = `Yes` and *Capacity Period* = `Hour`. This is the duration of the capacity (i.e., how long it lasts if the demand is 1 kW).
4. Open the *Costs* dialog, and enter *Operations Cost* = `0.15` (under *Overheads*). This is the cost of consuming a kWh, regardless of when, and so the *Cost Period* is ignored for a consumable resource. (In this example, the capacity is separately related to time with the *Capacity Period*, rather than the cost.)

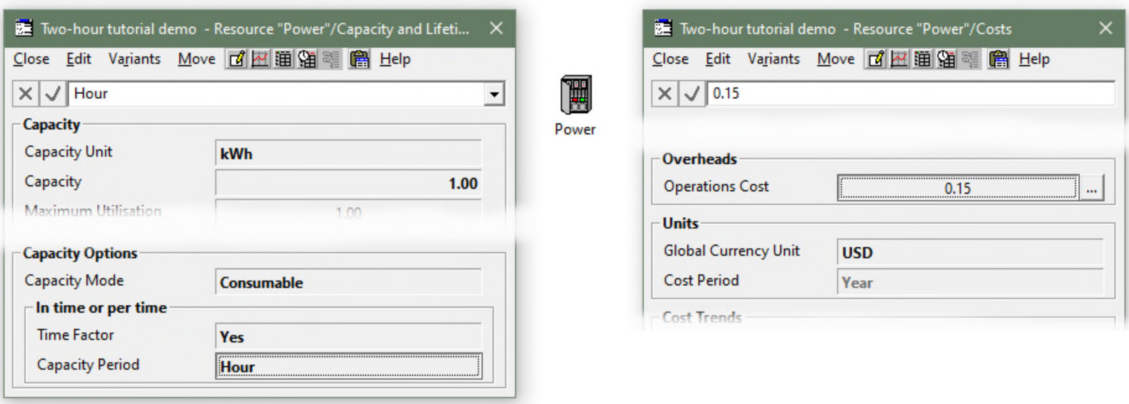


Figure 42: The unit capacity and duration for pricing, and the absolute unit cost, for resource *Power*

This means that **one nominal unit of capacity from the *Power* resource last an hour**.

The consumable resource and built-in time factor allow you to enter the assumptions at face value, and (in due course) to get a read-out of power required by the equipment (kW) vs energy consumed in the period (kWh, the actually billing quantity).

The same financial results could be obtained with a persistent resource for a kW with *Cost Period* = `Hour`, but the output would show *Installed Units* of kW instead of *Consumption* of kWh. *(Beyond the scope of this tutorial.)*

Without the time factor, this consumable resource would need to be driven by an **aggregate transformation** which calculated the energy required directly. This is readily achieved with the aptly named *Time Factor* transformation type, but does not provide a read-out of power required by the equipment. *(Beyond the scope of this tutorial.)*

Things that you should have seen and understood

Capacity Options: Capacity Mode, Time Factor, Capacity Period

Operations Cost

Exercise 20: The requirement for power

The transformation side of this is directly comparable to the structure for space. In this case, the stated assumption is that each chassis has a power consumption of 200 W:

1. Create a new transformation, and name it *Power required*.
2. Connect from *Access chassis* to *Power required*, using the quick-link target.
3. Open the *Input and Transformation* dialog. The *Input* is already wired up.
4. Set the *Output Unit* and *Multiplier* inputs as shown below. As we did with the service, we are matching our units to the resource by entering 0.2 kW rather than 200 W.
5. Connect on from *Power required* to *Power*, and choose *Requirements* when prompted.

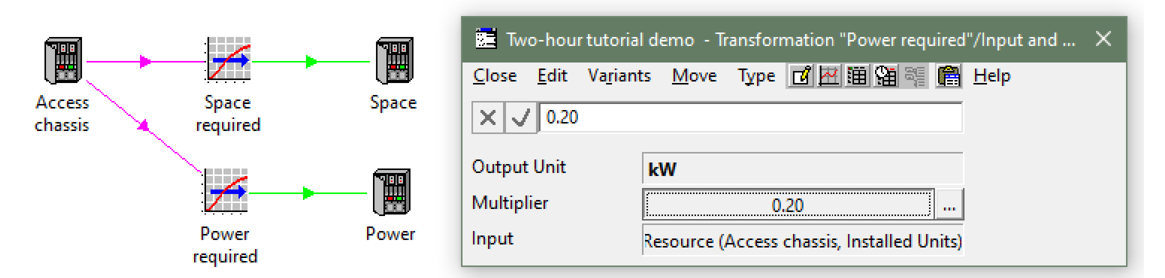


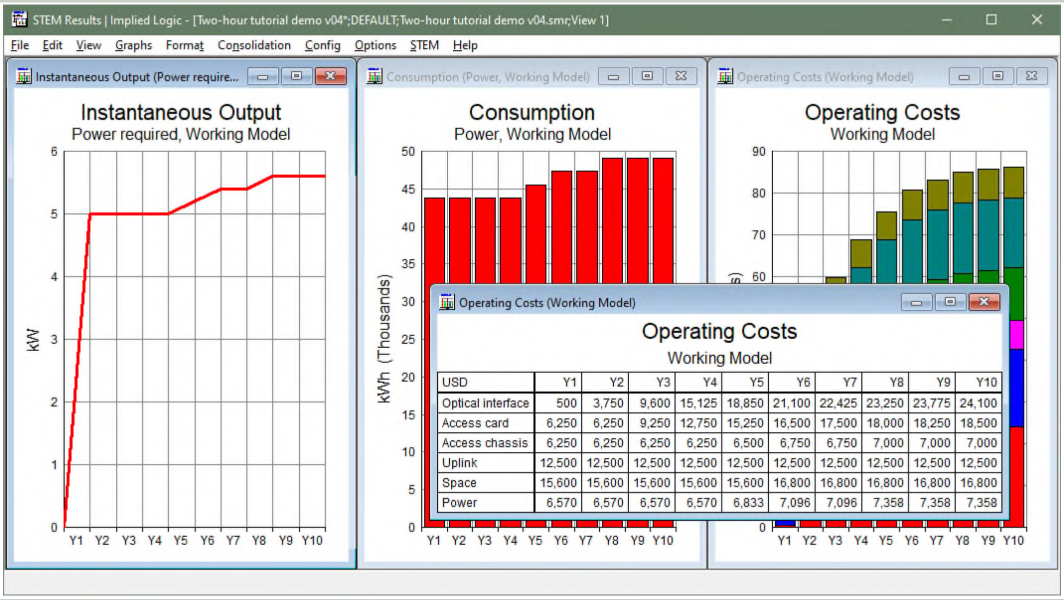
Figure 43: The *Multiplier* input governs the power consumption in kW per chassis

This means that **each installed unit of the chassis requires 0.2 kW of instantaneous capacity from the *Power* resource**. (At that rate, a kWh of capacity will last five hours.)

We will quickly check the arithmetic as we did before:

6. Run the model. The Results program is activated.
7. Draw the graph *Instantaneous Output* for the transformation *Power required*. This starts at 5 kW (0.2 kW × 25 units), and then gradually increases to 5.6 kW.
8. Checking the opex is harder, as we need to count how many hours there are in a year. STEM does this for us, and has thus worked out how much energy is consumed.
9. Draw the graph *Consumption* for the resource *Power*. Working with the easier (round) number for *Power required*, this starts at 43 800 kWh (5 kW × 24 × 365).
10. Add *Power* to the stacked *Operating Costs* graph, and check the numbers in a table. The annual opex for *Power* starts at USD 6570 (USD 0.15 × 43 800 kWh).

If you were to run the model in quarters or months, STEM would count the number of hours in a quarter or a month in order to calculate the quarterly or monthly opex. How would these graphs look then? *(Beyond the scope of this tutorial.)*



USD	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9	Y10
Optical interface	500	3,750	9,600	15,125	18,850	21,100	22,425	23,250	23,775	24,100
Access card	6,250	6,250	9,250	12,750	15,250	16,500	17,500	18,000	18,250	18,500
Access chassis	6,250	6,250	6,250	6,250	6,500	6,750	6,750	7,000	7,000	7,000
Uplink	12,500	12,500	12,500	12,500	12,500	12,500	12,500	12,500	12,500	12,500
Space	15,600	15,600	15,600	15,600	15,600	16,800	16,800	16,800	16,800	16,800
Power	6,570	6,570	6,570	6,570	6,833	7,096	7,096	7,358	7,358	7,358

Figure 44: The *Consumption* graph reveals how many kWh are used-up in the period

For simplicity, we have only considered the power requirements of the chassis (as we did more justifiably for floor space).

Similar transformation drivers may be added for any resource that has an explicit power consumption. These can be connected to the same *Power* resource and its consumption will correspond to the aggregate demand. *(Beyond the scope of this tutorial.)*

Things that you should have seen and understood

Consumption

10. Pricing for profitability and cashflow

We have been through all the resources, working out how many are required to satisfy the direct or indirect requirements from the service, as its demand grows over time, and accounting for the costs of those resources. The service operator may have a market-led expectation for how to price the service, but some elementary profitability and cashflow analysis will allow us to identify a minimum viable tariff.

Exercise 21: Establishing the cost per customer

Let's review some of the existing results (or redraw them if necessary):

1. The stacked graph of *Operating Costs* for all six resources shows that the annual total approaches USD 90 000 as the number of customers approaches 1000, from which a per-customer share of c. USD 90 can be inferred.
2. There is also *Capital Expenditure* for the first four resources, but for an annualised perspective, we need to draw (similar) *Depreciation and Amortisation.* This tips past USD 160 000 in Y10, c. USD 160 per customer, making an annual total of c. USD 250.
3. It is instructive, but of course it is not necessary to do the arithmetic. Another graph, *Operating Charge*, shows the combined opex + depreciation. Try it!
4. Also, the same results can be drawn for the service directly (as an allocated total).

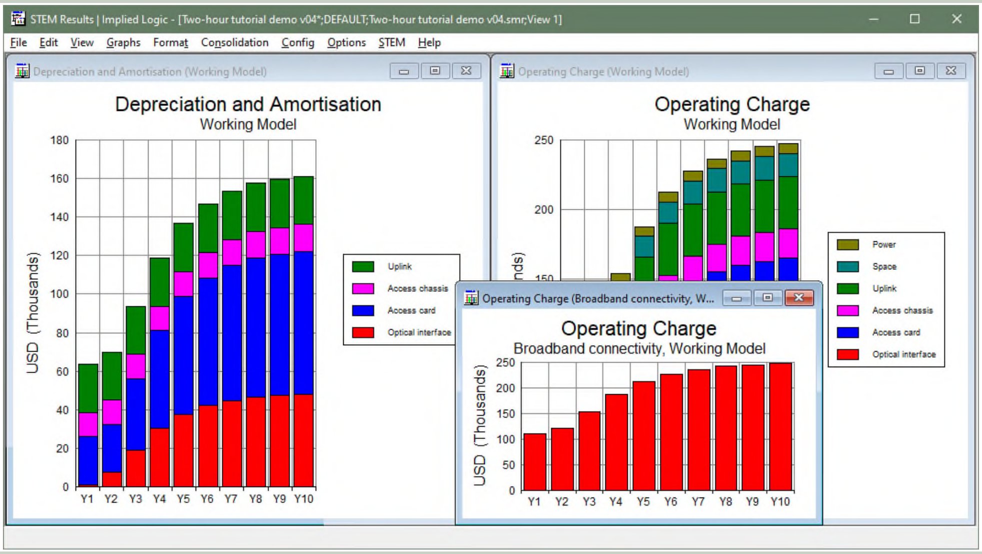
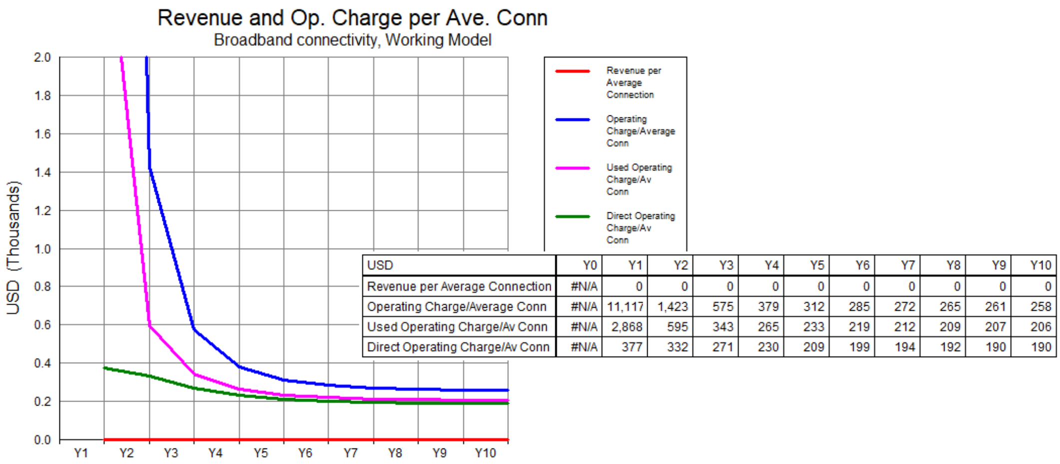


Figure 45: *Operating Charge* (opex + depreciation) amounts to an annual total of c. USD 250 per customer

In fact, we can read-out this result directly from another, important graph:

5. Draw the graph *Revenue and Op. Charge per Ave. Conn* for the service. The fully-allocated cost per customer is very large in Y1 when there are few customers, so it is helpful to change the scale in order to focus on the results from Y2 onwards.
6. Right-click the (value) y-axis and select *Format Axis* from the context menu. The *Format Value Axis* dialog is displayed.
7. Set *Maximum* = `2000` and click *OK.* The graph is redrawn at an amended scale.
8. It may helpful to show a separate table to check the numbers.



USD	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9	Y10
Revenue per Average Connection	#N/A	0	0	0	0	0	0	0	0	0	0
Operating Charge/Average Conn	#N/A	11,117	1,423	575	379	312	285	272	265	261	258
Used Operating Charge/Av Conn	#N/A	2,868	595	343	265	233	219	212	209	207	206
Direct Operating Charge/Av Conn	#N/A	377	332	271	230	209	199	194	192	190	190

Figure 46: The allocated cost per customer is an essential reference point for tariffing

It is clear from these results that the operator will need to earn, from each customer, at least USD 300 per annum, just to cover its annual costs once established.

Obviously, there is no revenue yet! Our next step will be to set a tariff.

The other used and direct-charge results on the graph include costs only for resource capacity which is actually used, and the latter also only in proportion to the used capacity of intermediate resource drivers. These results may be used to inform a notion of ‘efficient pricing’ in competitive markets. *(Beyond the scope of this tutorial.)*

Things that you should have seen and understood

Depreciation and Amortisation, Operating Charge (for resources and services)

Revenue and Op. Charge per Ave. Conn (services only)

Format Axis, Maximum

Exercise 22: Adding a tariff

We have just seen that USD 300 per customer is ‘sufficient’ annual revenue to cover costs, but only actually from Y6 onwards. Instead, we will aim for USD 360, as a ‘round multiple of 12’, corresponding to a monthly tariff of USD 30:

1. Access the *Tariffs* dialog from the icon menu for the service.
2. Enter the *Rental Tariff* and *Tariff Period* inputs as shown below (directly analogous to the *Rental Cost* and *Cost Period* we used for the *Space* resource).

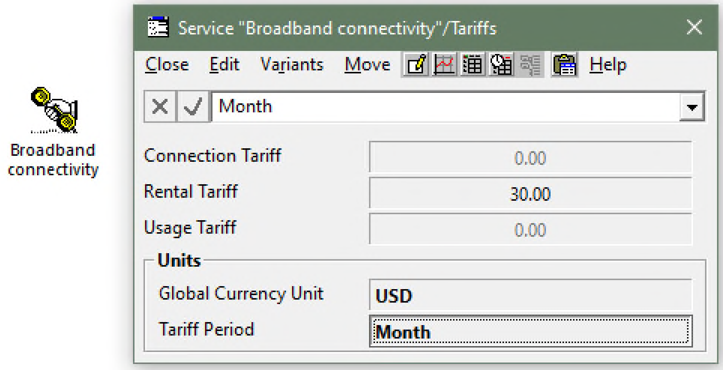


Figure 47: *Rental Tariff* and *Tariff Period* service inputs, comparable to *Rental Cost* and *Cost Period* for *Space*

3. Run the model. The Results program is activated.
4. Now the *Revenue per Average Connection* result comes to life, flat at USD 360 as expected, and exceeds the fully-allocated cost per customer from Y5 onwards.
5. Draw the graphs *Operating Profit* and *Operating Profit Margin* for the service. (You can make a multiple selection in the *Graphs* tab of the *Draw Similar* dialog.) The latter is very negative in Y1, so we will amend the scale again.
6. Set *Minimum* = `0` in the *Format Value Axis* dialog (for the value axis) and click *OK*. The graph is redrawn with the negative values suppressed.

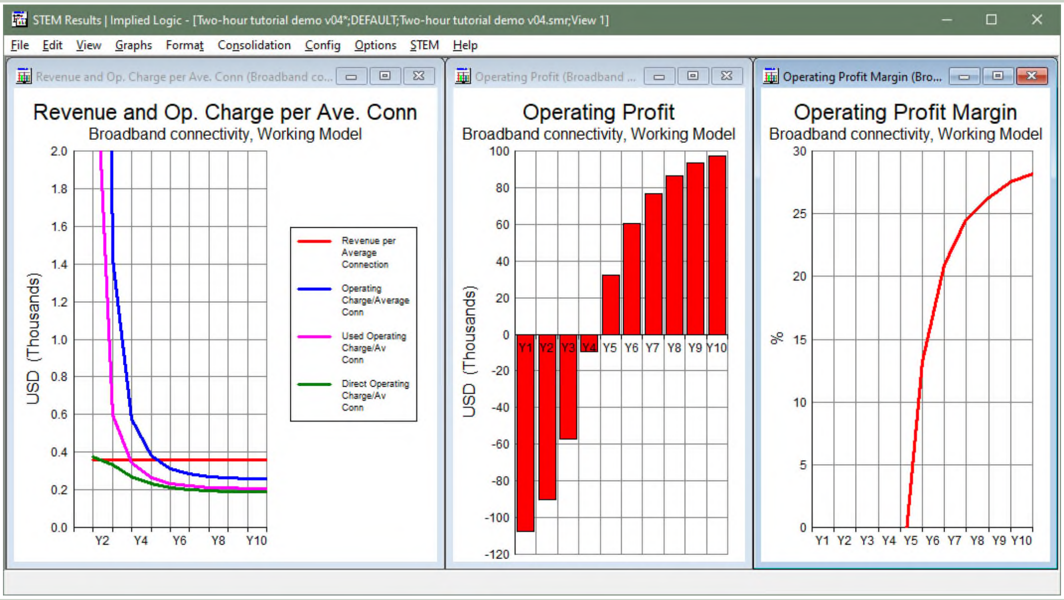


Figure 48: *Revenue per Average Connection* exceeds the fully-allocated cost per customer from Y5 onwards

The service is evidently profitable from the end of Y5, increasing from 14% to 28% over the next five years, but is that enough for a convincing business case?

Things that you should have seen and understood

Rental Tariff, Tariff Period

Operating Profit, Operating Profit Margin

Format Axis, Minimum

Exercise 23: Reviewing the cashflow position

Cumulative cashflow or, more precisely, the discounted cumulative cashflow (DCF, which factors-in reduced certainty into the future), is an essential metric for viability. The *Net Present Value* (NPV) result is a suitably qualified DCF, over the period modelled, plus some measure of terminal value (a similarly discounted projection into the future).

The *Discount Rate* is actually a model input, but we will just use the 10% default. There are also various ways of calculating a terminal value. *(Beyond the scope of this tutorial.)*

We will focus on the most cautious result which disregards any terminal value:

1. Select *Draw…* from the *Graphs* menu. The *Draw* dialog is displayed.
2. This time, select *(Network)* as the element. This unique element represents the aggregate of all service revenues and resource costs.
3. Draw the graph *NPV*, which includes the single result *NPV (Zero Terminal Value)*.

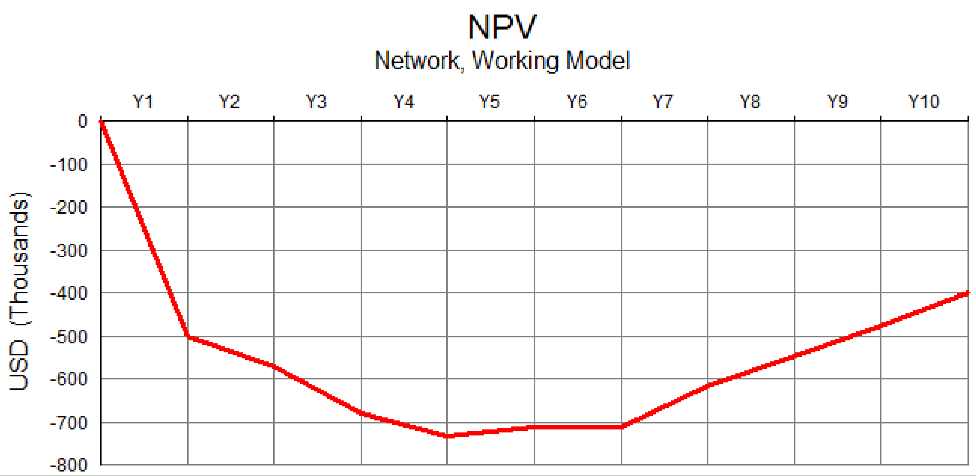


Figure 49: Initial investment is certain, whereas revenue is delayed and cumulatively discounted

It may be surprising to see that, even though the business is profitable for the last five years (and indeed cashflow-positive in the same time-frame), it is nowhere near payback at the end of the period. However, there is significant spending on initial infrastructure which is unavoidable, whereas the revenue only picks up as customers are added, and is cumulatively discounted over time, reflecting increasing risk in the assumptions.

The business is still repaying its investors with no reliable return in sight after ten years, and this will never be acceptable. 30% profitability is good, or at least OK, but this must be achieved earlier for the investment to be attractive. With the cost side as a given, the business must charge more for the service.

Considering the potential trade-off between charging either **more** at the start to cover costs vs **less** to stimulate initial customer take-up, we will stick to a constant tariff to keep things simple and avoid compound effects for now. On the strength of the current results, let's try doubling the tariff and see if that yields a more favourable outcome:

4. Set *Rental Tariff* = `60.0` in the Editor. (There is no need to close the *Tariffs* dialog, or any of the other data dialogs, between model runs; the inputs update immediately.)
5. Run the model. The Results program is activated, and the graphs are updated.

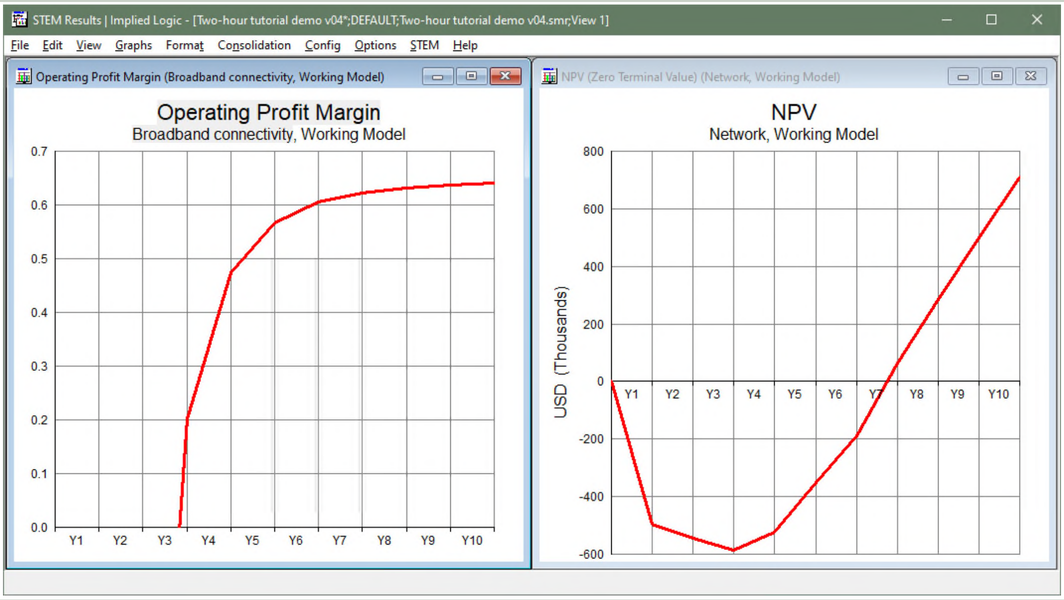


Figure 50: The business now achieves payback, but at what risk?

As you should see reflected in your own results, the business is now profitable in Y3, and achieves payback by the end of Y7. This is much better, but it's 'still too long' by modern standards, where customer expectations change every decade because of innovation.

Things that you should have seen and understood

(Network) element, NPV (Zero Terminal Value), Discount Rate

11. Linking margin to payback with scenarios

Now that we have established some prototype results, you might well ask what tariff is required to achieve payback in five, four, or even three years? As well as answering this question at face-value, seeing the corresponding results together will yield a broader understanding of the correlation between input and output.

The **dimension** and **variant** elements in STEM allow a model to be run multiple times in parallel with different, saved inputs, and for the results to be graphed together, in order to support this kind of scenario analysis.

Exercise 24: Maintaining differing sets of assumptions

We could just iterate the tariff input through a series of increasing values as a single exercise until we found values that would yield the desired payback results, but what if we want to re-visit one of these values? Rather than write down and re-enter by hand, we can structure our model to retain a set of alternative values for the tariff input:

1. Select the *Rental Tariff* field in the body of the *Tariffs* dialog for the service.
2. Click *Variants* on the dialog menu, and select *Add as Scenario Parameter…* from the drop-down. A *Variant Data* dialog is displayed with three copies of the original value. The three columns are headed *Variant 1*, *Variant 2* and *Variant 3*, and the first of ten rows is labelled *Broadbandconnecti...l.CostIndepTariff*. You will see behind that the original input cell is shaded pink to indicate that the value is volatile and will change.
3. We know all the values need to be higher, so let's try 80/100/120 as illustrated below (i.e., doubling again for the highest).

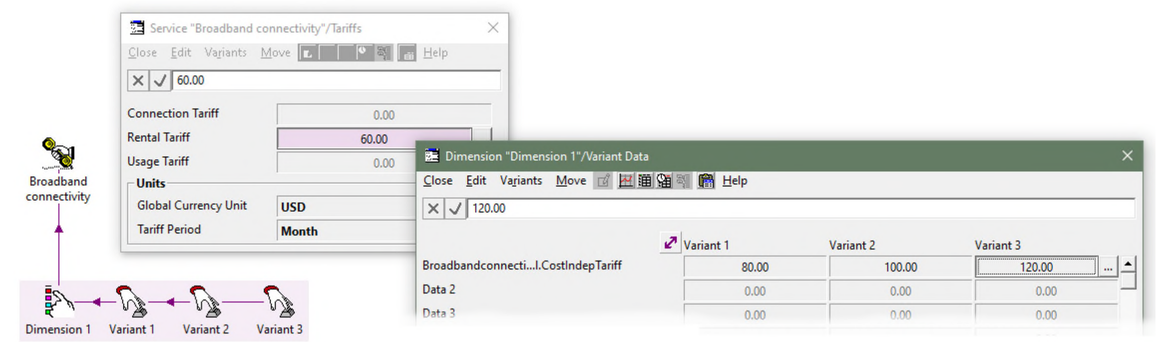


Figure 51: Entering three alternative values for the *Rental Tariff* input as a scenario parameter

The *Variants* menu includes commands to *View Original* or *View Variant Data* which you can use to jump between the *Variant Data* dialog and the original *Tariffs* dialog.

Four new icons are visible in the background: a dimension, and three variant elements. The dimension element identifies the tariff input as a relevant parameter, and the associated variant elements provide alternative values which may be used for that input.

We will rename these elements to make our intentions clearer:

4. Click and drag around the four new icons to (rubber-band) select them.
5. Right-click the selection, and then select *Rename…* from the icon menu (or press <Ctrl+R> or <F2>). The Editor prompts for a new name for each element in turn.
6. Name the elements as shown below. The column headings in the *Variant Data* dialog are updated accordingly.

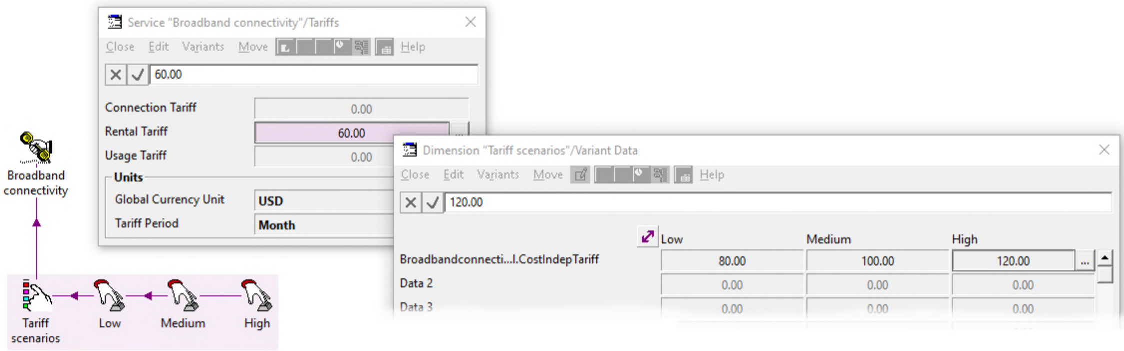


Figure 52: The model now maintains *Low, Medium* and *High* scenarios for the *Rental Tariff* input

As you would guess from the user interface, this *Variant Data* can be readily extended to cover multiple, correlated inputs (e.g., higher tariff, lower demand), and you can have as many variants as you like. Other assumptions, such as technical evolution or supplier pricing, may be varied independently with the addition of further dimensions. *(Beyond the scope of this tutorial.)*

Things that you should have seen and understood

Dimension element, variant element

Add as Scenario Parameter, Variant Data, View Original, View Variant Data

Exercise 25: Running and graphing scenarios to compare outcomes

In all the earlier exercises, we have just ‘run the model’ (*File/Run* or <F5>). Now we have added these variants for the tariff input, we need to be more specific:

1. Select *Scenarios and Sensitivities* on the *File* menu (after *Run*). The *Scenarios and Sensitivities* dialog is displayed, with a prominent list of the scenarios defined for the model. Below *Low*, *Medium* and *High* in the list, there is a checkbox labelled *Include Working Model* which allows for the model to be run with the current value displayed in the *Tariffs* dialog. This is currently still `60.0`, but may be varied independently of the *Variant Data*. (The *Working Model* is like a separate, default scenario.)
2. Check *Include Working Model* (so our existing results will remain up to date), and then click *All* under *Save and Run* (or you can first select all the scenarios in the list, and the click *Selection* under the same heading).

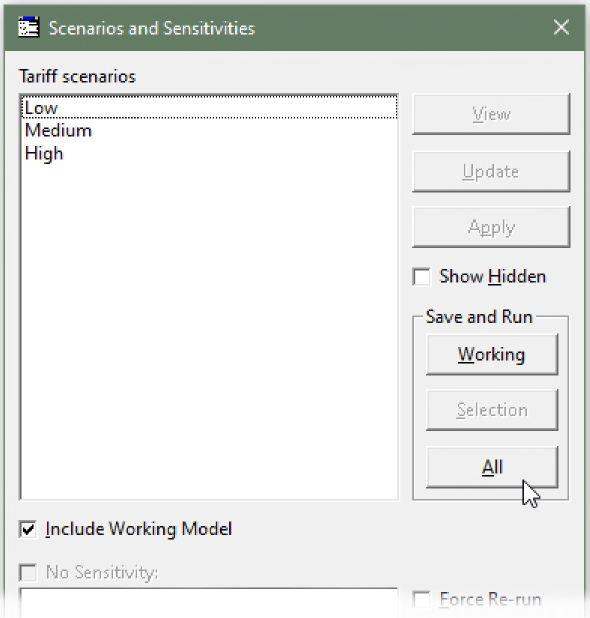


Figure 53: Running all scenarios, including the working model

The model is run four times in parallel (once for each scenario, with the new values `80.0/100.0/120.0` plugged in, plus the working model at `60.0`). If you are quick, you will see some fleeting messages in the status bar at the bottom left of the Editor.

The Results program is activated, and the graphs are updated, but nothing appears to have changed! All of the existing graphs relate to the working model (which was all we could graph before), and we have not changed the working value in the *Tariffs* dialog.

Now we will modify the *NPV* graph to show the different scenario results instead:

3. Double-click where it says *Working Model* on the *NPV* graph. The *Change Selections* dialog is displayed with an extra *Scenarios* tab listing the available scenarios.
4. Double click each of *Low*, *Medium* and *High* to add them to the *Selected* list (or click and drag to select them all and then click the >> button).
5. Double click *Working Model* in the *Selected* list to remove it (or click to select it and then click the << button).
6. Click *OK*. The graph is redrawn with three lines, one for each scenario.

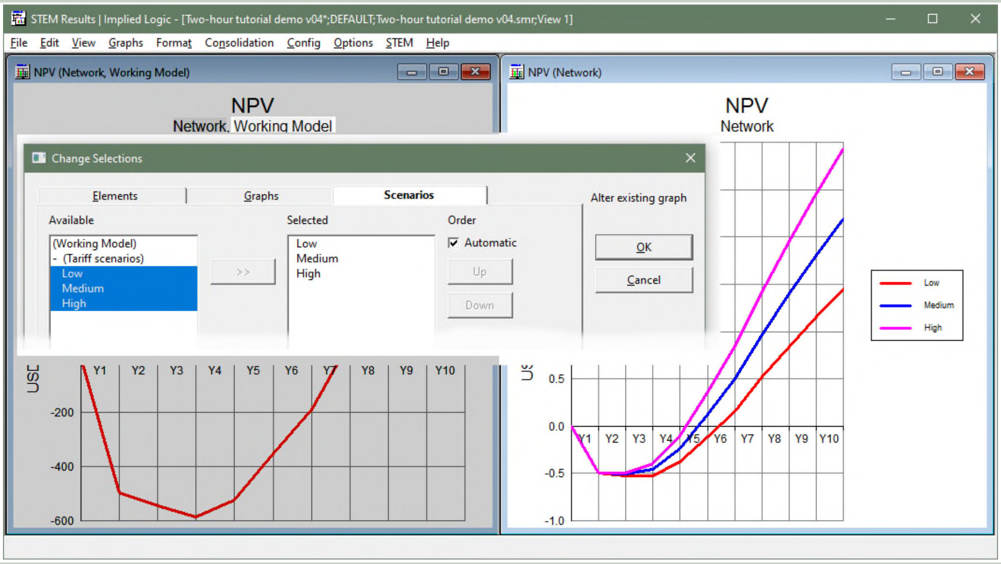


Figure 54: Changing the *NPV* graph to show *Low*, *Medium* and *High* instead of the *Working Model*

Our choices were not aggressive enough to achieve the desired results, but the spread on the graph makes it much easier to assess what values we might need. For payback at the end of Y5, it looks like we need a value between the current *Medium* and *High*, say `90.0`. For Y4, it needs to be slightly more than the current *High*, perhaps `125.0`, whereas for Y3 we will still have to guess at something like `150.0`. Try it!

7. Enter the new values `90.0/125.0/150.0` for *Low/Medium/High* in the *Variant Data* dialog in the Editor. (You can access this directly from the *Tariff scenarios* icon.)
8. You can still just press <F5> to re-run the model. All the scenarios that were run last time will be run again. The Results program is activated, and the graphs are updated.
9. Repeat this process until the lines on the *NPV* graph cross the axis as desired.

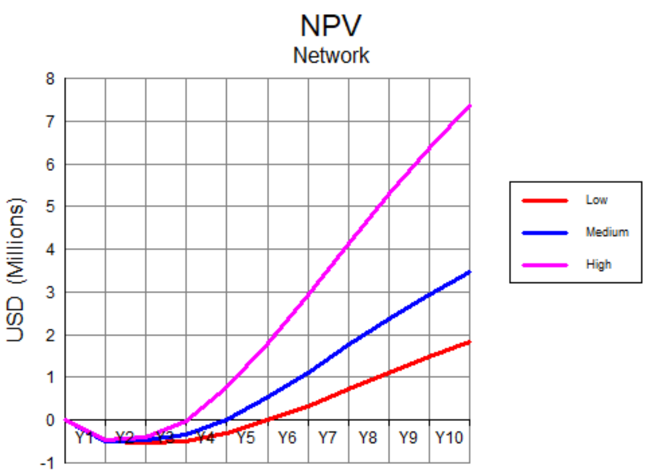


Figure 55: The business achieves payback in Y5/Y4/Y3 in the *Low/Medium/High* scenarios

Things that you should have seen and understood

Scenarios and Sensitivities, Include Working Model, Run Selection, Run All

Scenarios tab, Working Model

12. A business model is for flexing until it breaks

Most of this modelling exercise has been focused on the essential foundation of thinking through the various cost headings, and determining how they scale with demand. In the last two sections (ten minutes of the associated video), we have used the model to calculate the cost per customer, and hence the profit margin for a given tariff, and then 'turned the handle' repeatedly to explore the business-case dynamics.

This is what a business model is for. We create a shiny set of icons and calculations that best capture the uncertain parameters facing a business, but then it is the flexing of that model through varying input assumptions and more structured scenario and sensitivity analysis that delivers the real insight.

Look for an explanation if the results are strange. Is it the model or the business which is broken? It will be much cheaper to fix either before the dollars modelled become real!

The topics in this tutorial reflect the kind of conversations you should be having with the project team evaluating a business opportunity, focusing on the principles and dynamics, rather than the workings and the math. A STEM model is built on a rich fabric of pre-validated elements which can be wired up much more visibly and consistently than hand-crafted logic in a spreadsheet. Your colleagues and/or customers will value a model that connects **technical credibility with reliable financial impact**.

Please contact sales@impliedlogic.com if you would like advice on working with STEM in your business.

13. Homework suggestions for further experimentation

If you have enjoyed this tutorial, and would like more practice with the software, then please review the side-notes mentioned in the text as *beyond the scope of this tutorial*. These are recapped below for your convenience, in two separate categories:

Extensions to the narrative of this case study

1. more thorough analysis of traffic in the busy period
2. using optional *Maximum Utilisation* and *Minimum Slack Capacity* inputs to provision extra resource capacity as contingency for imminent additions
3. assuming *Distribution* = `Monte Carlo` (i.e., the pure, non-extended version) when a 'just-in-time' deployment model is feasible for the first customer at each site
4. a chassis might need to accommodate other types of cards (even the power supply)
5. trying a phased roll-out which would limit the initial capex, but also delay revenue
6. using *template replication* to model cards and chassis at individual sites
7. charging for space in whole sq m at each site, or conversely working with the *Usage Cost* attribute which is charged only in proportion to used capacity
8. trying a persistent resource for a kW or an aggregate, energy-required transformation as alternatives to the recommended, consumable resource with built-in time factor
9. comparing *Consumption* results in years, versus quarters or months
10. connecting multiple power-required drivers to the same *Power* resource.

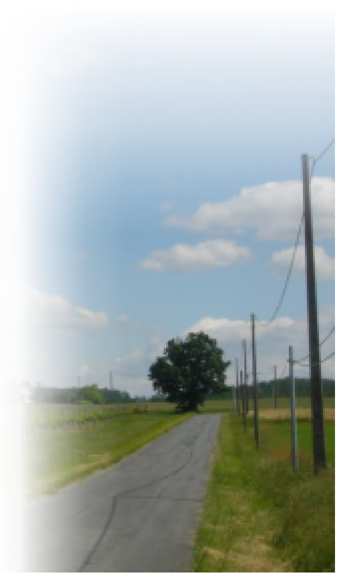
Other software features mentioned in the text

1. using *Consolidation* to graph aggregate quarterly/monthly results on an annual basis
2. working with per-element and global *User Data*; e.g., to work with multiple currencies
3. other transformation types, such as *Input-Output Mapping* or *Expression*
4. other resource input bases, such as `Installed Capacity` or `Incremental Units`
5. using a transformation to govern *Sites* or *Planned Units* for a resource
6. other service input bases, such as `Traffic` or `New Connections`
7. efficient pricing based on used or direct cost results
8. choosing a suitable *Discount Rate*
9. considering the terminal value of a business
10. adding extra scenario dimensions, such as technical evolution or supplier pricing.

Press <F1> anywhere in the STEM software to access context-sensitive help, or learn more from our comprehensive, online-help resource at http://help.stem.impliedlogic.com/.

Business-modelling software, training and expert advisory services for strategy-planning and cost-allocation managers

- Telecommunications
- Information technology
- Cloud computing
- Serviced properties
- Transport logistics
- Energy supply



Implied Logic Limited, Milton Hall, Ely Road, Milton, Cambridge, CB24 6WZ, UK
Tel: +44 1954 252120 ■ Email: info@impliedlogic.com
Registered in the United Kingdom, number 07233952 ■ www.impliedlogic.com