Developing a program logic model

What is a program logic model?

A program logic model (or logic model) is a visual tool to show, in simple terms, how a program operates to produce change. It usually includes a program's:

- key components (described below)
- theory of change, and the assumptions that inform how inputs and activities convert to outputs and outcomes
- operational framework, to which planned evaluation activities can be aligned.

Why use program logic?

It can improve your understanding of:

- the resources and activities required to achieve desired outcomes
- the types of influences (inside or outside of the program) that may positively or negatively affect outcomes and which require monitoring
- what to measure, for what purpose and in which context.

When to use program logic

Planning: A logic model can be used for both designing a program and planning an evaluation. Early development of a logic model will help to guide these processes and create a shared understanding of the program.

Implementation: A logic model should be treated as an iterative exercise. It is beneficial to update your model as more is learnt about the program, through implementation and regular feedback. This will help to ensure any new learnings, assumptions and linkages are reflected in the evaluation.

Review: An updated logic model should be included as part of reporting. This will allow you to show how the evaluation has shaped your understanding of the program, including the identification of the key activities, linkages and pathways that led to desired program outcomes.

What are the key components of a program logic model?

	Description	Examples for a program that provides training for unemployed persons to help them obtain a job
Objectives	What the program aims to achieve (desired program outcomes)	Assist unemployed persons to find employment through training
Inputs	The resources needed to deliver the program	Human, financial, organisation or physical
Activities	Actions (including events, processes and systems) required to produce program outputs	Promotion, administrative processes, service delivery or support
Outputs	Quantifiable products or services delivered as part of activities performed	Number of training sessions provided; number of emails answered; percentage of cases resolved
Outcomes	The intended or unintended, positive or negative results that may be directly or indirectly influenced by the program. Changes may occur in the short, medium or long term	Increased confidence among unemployed persons to apply for jobs; improved attitudes among employers to hire trained unemployed persons (intended); training is unsuitable for particular industries, occupations, age-groups or localities (unintended)
Assumptions	The required conditions, expected interactions or intended pathways to outcomes, according to a program's theory or framework of change	Eligible unemployed persons are aware of the program; participants are able to be trained (through access, willingness and ability etc); suitable jobs are available for participating unemployed persons
Threats to validity	Non-program factors (external to and beyond the control of a program) that interact with the program or influence how it operates.	Competing employment training programs; social and economic conditions; availability of other services; legislative factors

How to develop a logic model

There is no best way to develop a logic model. Your approach should depend on the model's purpose, how it will be used (for program design, implementation or evaluation), who will use it and the complexity of the program and its context.

A logic model can be created using reverse logic (working backwards from intended outcomes to required inputs), forward logic (working from proposed inputs to likely outcomes) or a combination of both (updating the logic's various components as more is learnt about a program).

Whatever your approach, developing a logic model is best undertaken as an analytical, consultative and iterative process, informed by evidence and stakeholder input.

A few key steps that can assist in developing a logic model for evaluation include the following.

1. Identify the intent of the program – why was the program designed and what is it trying to achieve?

   Use this information to write program objectives and as a frame of reference for developing your logic model.

2. Review available program information – what resources have been allocated to the program, how will it be implemented, how is it designed to operate and how will it be managed?

   Use this information to understand the program's inputs, activities, outputs and associated assumptions.

3. Conduct an environmental scan – what factors could act as a barrier, enabler or competitor to the program?

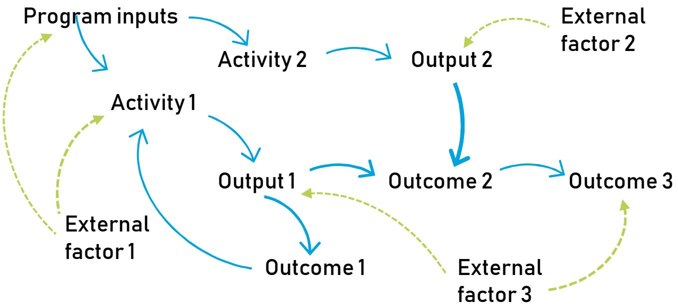
   Use this information to identify threats to validity and contextualise the program.

4. Consult with program stakeholders – identify the program's stakeholders and understand their characteristics, needs and expectations.

   Use this information (along with knowledge gained from steps one to three) to determine a program's likely intended and unintended outcomes, and its associated assumptions.

Key considerations

- When developing your logic model, thoroughly mapping a program's pathways is just as important as mapping its likely outcomes. It is also a good idea to accompany your model with a short explanation about how and why the program is understood to work.
- Programs rarely have simple, linear, causal pathways from design and implementation through to outcomes. There are usually multiple possible pathways to achieve an outcome and program activities are often implemented with some variation, such as over time or geography.
- Inputs, activities, outputs, assumptions and outcomes can all be influenced by factors outside of the program (threats to validity).
- When reporting findings, you can emphasise the pathways that your evaluation's evidence suggests have stronger linkages to outcomes, such as with line thickness to depict strength of evidence.



Queensland Government Statistician's Office, 2019

- Due to its iterative nature, a program logic model should be updated when
  - a program's design is modified
  - changes are made to its inputs and activities
  - more is learnt about how and why a program operates
  - additional intended or unintended outcomes are identified or measured.