# Checklist of Key Considerations for Development of Program Logic Models

Goldie MacDonald

A logic model is a graphic representation of a program that depicts stakeholder understandings of program components and relationships between and among components.[1] Specifically, logic models convey the relationships among the resources to operate a program, activities planned, and the desired changes or results.[2] As used here, *program* refers to the intervention, collection of activities or services, or project depicted in a logic model. Logic models are a popular format to illustrate a program, or aspects of a program, to aid in program planning, implementation, or evaluation. For example, when an existing program description is outdated or incomplete, preparing a logic model can reveal how stakeholders expect a program to bring about change in a specific setting. Practitioners use different types of evidence, practice-based knowledge, stakeholder understandings of a program, and other input to develop logic models.

This checklist distills selected literature and practice wisdom into a tool to support development of logic models and provides a common vocabulary for those who participate in this work. It is not an inventory of all perspectives concerning how to create or use logic models. The checklist addresses five crucial questions:

- How do stakeholders determine the scope of a logic model?
- What basic components should a logic model include?
- How do logic models illustrate the underlying logic of a program or distill program theory?
- What are key items to include when formatting a logic model?
- What should a narrative explanation of a logic model include?

The tool includes five sections that correspond to the questions above. Each section contains fundamental information about how to depict a program in a logic model and how to avoid common missteps that can limit a logic model's quality and utility.

Read the checklist in its entirety before applying the information for the first time. Share it with stakeholders who will be involved in creating a new logic model or reviewing an existing one. Discuss each checkpoint as a group to develop accurate and meaningful content for each logic model component and determine how to present the content visually. While there is no agreed-upon, single process or template to develop a logic model, the content in this checklist is applicable in a range of settings and practice areas.

## 1. Determine the Scope of the Logic Model

Practitioners use different types of logic models to emphasize different aspects of programs (e.g., logic models that emphasize program activities, pathways to outcomes, or program theory).[2] The scope of any logic model depends on the intended audience and how they will use the graphic.

- ☐ **Decide on the focus or scope of the logic model in collaboration with stakeholders.** Determine whether the logic model should present the program as a whole or pieces of the program in more detail. Consider whether to create a single logic model or set of related logic models. For example, at the outset of a program, developing one logic model that conveys the program as a whole can be a practical starting point. Discussing how stakeholders intend to use the logic model and the information needed to support that use narrows the focus or scope of the logic model. In many cases, the ideal focus or scope becomes clearer as development of the logic model progresses.
- ☐ **Determine the level of detail needed to adequately represent the program and reveal its underlying logic or distill program theory.** Logic models that are too general or vague will not be helpful to intended users and can be misleading regarding important aspects of the program. For example, logic models that oversimplify associations between activities and outcomes can adversely influence program planning, implementation, or evaluation. Similarly, a logic model is not an inventory of actions or tasks (e.g., a work plan) and overemphasizing one component of a logic model can skew representations of the whole, including relationships among components.

## 2. Include the Basic Components of a Logic Model

At a minimum, a logic model includes inputs, activities, outputs, and outcomes presented in a series of boxes arranged purposefully. Although the content and format of logic models vary, the graphic should include these components in a purposeful progression from inputs to outcomes. The presence of all four components aids in depicting the program and its underlying logic more clearly. The necessary level of detail for each component depends on users' information needs and intended uses of the logic model.

- ☐ **Define program inputs, activities, outputs, and outcomes for presentation in a logic model**.
  - **Inputs are the actual or anticipated resources to plan and implement the program, both material and intellectual**.[3] Resources can include program personnel or participation by other stakeholders; funding or in-kind contributions; necessary capabilities, infrastructure, or supplies; or relevant guidelines, local knowledge, or expertise.
  - **Activities are what the program does to create, or contribute to, change.**[4] Program activities can include a range of interventions, programming, or processes to address conditions or desired changes.
  - **Outputs are the direct products of program implementation**.[5] Outputs include tangible artifacts of activities that are in process or complete (e.g., educational programming or training delivered). Activities are not changes in the conditions the program aims to address (e.g., increased knowledge or skills after educational programming or training). Avoid presenting outputs as specific data points, indicators, or metrics (e.g., number of participants)—this practice

can impede discussion of how outputs relate to other components and hinder full examination of options for program evaluation. For example, it is problematic to present outputs as evaluation indicators or metrics when stakeholders have not yet defined overarching evaluation questions or other pieces of an evaluation plan.

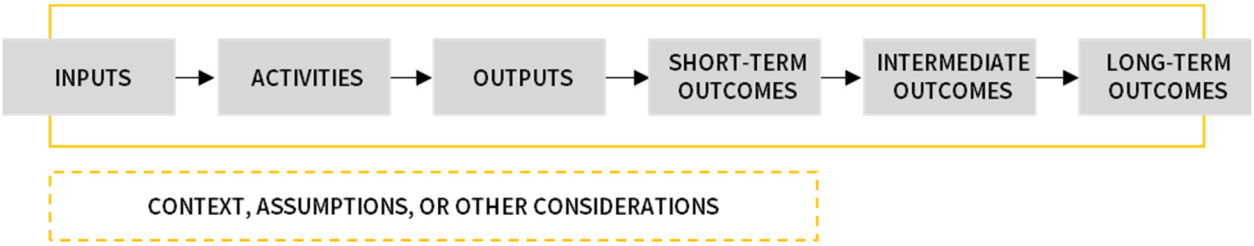
- **Outcomes are the expected benefits, changes, or results of the program**.[3] Typically, logic models include outcomes in progression from proximal to distal, or short-, intermediate-, and long-term. Some logic models also specify impacts as the most long-term aim or desired change in conditions. Logic models tend to present only *intended* outcomes. A narrative explanation of a logic model (oral or written) can include potential unintended outcomes, positive or negative.

☐ **Present program inputs, activities, outputs, and outcomes as a series of boxes arranged in a purposeful order.** Use lines and arrows to connect these boxes and represent the associations between and among components. Despite the simplicity of Figure 1, most programs are more complex than can be conveyed in any logic model. Addressing all of the checkpoints that follow will aid in creating a more robust representation of a program in a logic model.

☐ **Incorporate contextual information relevant to the program.** Contextual information includes factors external to the program but directly relevant to program planning, implementation, or outcomes. "Context describes the important features of the environment in which the project or intervention takes place."[6] For example, cultural or geographic factors, demographic characteristics of communities or persons, economic or political conditions, and historical events or legacies can shape the program or its components. Exploring contextual factors when developing a logic model can deepen understanding of the problem the program aims to address, the suitability of programming in a specific setting, and potential weaknesses in program design.[7] Present crucial contextual information explicitly in a box below the logic model (Figure 1), or discuss contextual information in a narrative that accompanies the logic model.

☐ **Incorporate assumptions or other considerations relevant to program planning or implementation.** Relevant assumptions or other considerations can include predisposing, enabling, or reinforcing factors relevant to the program (e.g., types of incentives for participation), or barriers or constraints relevant to the program (e.g., limited or unequal access to programming, uncertain availability of resources over time). Whether implicit or explicit, these considerations are integral to how stakeholders understand a program. The importance of exploring these items when developing a logic model cannot be overstated. Explain crucial assumptions or other considerations in a short statement below the logic model (Figure 1) or at the bottom of the page. You can discuss this content in more detail in a narrative that accompanies the logic model.

**Figure 1. Simple presentation of the basic components of a logic model**




## 3. Illustrate the Underlying Logic of a Program or Distill Program Theory

A logic model should not reduce a program to solely its component parts; a logic model distills program theory to represent underlying relationships between and among components. Program theory is distinct from a logic model but not unrelated.[8] *Program theory* refers to a "plausible, sensible model of how a program is supposed to work,"[9] including how a program "is understood to contribute to its intended or observed outcomes."[10] The checkpoints in this section address working with stakeholders to understand and depict how a program is expected to work.

- ☐ **Construct the underlying logic of a program or distill program theory in collaboration with stakeholders.** Stakeholders rarely understand program components, or associations among these components, in exactly the same ways. Logic models should represent how stakeholders expect a program to create or contribute to desired changes. For example, engaging stakeholders to co-create a logic model aids in identifying divergent understandings of a program or its mechanisms of change.
- ☐ **Draw on existing evidence and other relevant information to complement stakeholder understandings of the underlying logic of a program or program theory.** In addition to stakeholder input, practitioners can use published and unpublished research, evaluations of the same or similar programs, and other types of information (e.g., program records, historical documents) to understand and illustrate the underlying logic of a program and relationships among components.
- ☐ **Confirm that boxes and arrows depict the underlying logic of the program or distill program theory. Program theory includes *what* we expect to occur, as well as *how* and *why*.**[6] Logic models distill and represent program theory using content in boxes connected by lines or arrows in progression from use of resources (i.e., inputs, activities, and outputs) to results (i.e., outcomes). Examine whether the activities presented in a logic model can bring about, or contribute to, desired changes in a specific setting. Be able to explain the progression from use of resources to achievement of results presented in the logic model and why it represents sound reasoning—this progression is a key characteristic of a logic model.

## 4. Include Key Items When Formatting a Logic Model

There is no single template to format a logic model, but the vast majority of logic models are a single page of content read from left to right. It is crucial to include several key items when formatting a logic model:

- ☐ **Include a title for the logic model at the top of the page.** The focus or scope of the model is not always apparent to readers. In some cases, it is helpful to place a short description below the title to introduce the program presented in the logic model.
- ☐ **Include explicit headers for each category of content (e.g., inputs, activities, outputs, outcomes, and contextual or other considerations).** Although logic models might be commonplace in some organizations, stakeholders' and other users' familiarity with them varies. Definitions for each header rarely appear in the logic model itself, but it is important to define each category of content in advance of developing a logic model. Share these definitions with stakeholders who participate in developing a logic model, and include the definitions in a narrative accompanying the logic model.
- ☐ **Use plain language.** *Plain language* is "straightforward expression, using only as many words as are necessary."[11] For example, avoid inflated vocabulary and jargon, spell out acronyms at first use, and define technical terms. The Plain Language Action and Information Network provides a collection of open-access resources to support clear communication at https://www.plainlanguage.gov/resources/
- ☐ **Examine logic models relevant to your practice area or intended uses in preparation for formatting a logic model**. Carefully reviewing other logic models reveals the range of options and variations relevant to formatting, including commonalities that exist across disciplines and organizations. The U.S. Centers for Disease Control and Prevention (CDC) and University of Wisconsin-Extension (UWEX) maintain well-curated collections of open-access logic models from diverse programs and settings.

  Library Guide on Logic Models, Stephen B. Thacker Library, CDC
  https://www.cdc.gov/library/researchguides/LogicModels.html

  Logic Model Examples, Program Development and Evaluation, UWEX
  https://fyi.uwex.edu/programdevelopment/logic-models/bibliography/
- ☐ **Balance attention to content and format.** An appealing format will not compensate for flawed content, and poor formatting can hinder users' understanding of sound content. Multiple resources at the sites mentioned above discuss how authors developed and used logic models in a specific setting. Several authors examine the challenges of formatting content in a logic model such that the information is useful in program planning or evaluation.

## 5. Prepare a Narrative to Accompany the Logic Model

Pictures do not always speak for themselves. Do not assume that users will understand a logic model as intended, or even in the same ways as other users. To promote a shared understanding of the logic model, explain it in narrative form. A narrative explanation of a logic model aids users unfamiliar with the program to understand it as intended, including how and why stakeholders expect outcomes to occur. At a minimum, a narrative explanation of a logic model should address certain items:

- ☐ **Explain what is included in each component of the logic model.** The text in a logic model is abbreviated due to formatting constraints. Use the narrative to elaborate on what is presented in each box—do not simply repeat the text in the boxes. For example, an activity presented in a logic model can require multiple pieces to be considered complete (e.g., a training program that includes recruitment of qualified participants, classroom-based instruction on pre-defined competencies, and supervised fieldwork to apply new knowledge). Discuss what is represented in each box in the narrative to avoid oversimplification of program components.
- ☐ **Explain why each component represents an important aspect of the program.** Substantiate the inclusion of each component of the logic model in the narrative. For example, the narrative provides an opportunity to explain the importance of certain inputs to program implementation, the necessity of an activity to make progress toward outcomes, or the placement of a particular outcome in the progression from short- to long-term results. The importance of each component in the progression from use of resources to achievement of results can be hard to understand by just looking at the graphic.
- ☐ **Present a rationale for why stakeholders expect components to lead to, or contribute to, other components in the logic model.** This explanation can include research or practice-based knowledge, stakeholder input, or other sources. Regardless of the type of evidence presented, the narrative should include a plausible explanation of why stakeholders believe a component advances other components of the model toward the program's long-term outcomes. Specify the information sources used to develop this portion of the narrative, and include notes or references, as appropriate (e.g., in situations where funders or partner organizations expect and value references). This part of the narrative can be difficult to prepare but it helps users of the logic model understand *how* stakeholders expect the program to bring about desired changes.
- ☐ **Explain whether the logic model indicates attribution of or contribution to outcomes.** *Attribution* implies that the program *causes* the outcomes presented in the logic model. Whereas, *contribution* implies that a program *plays a part* in bringing about desired outcomes. Be careful when using *if-then* statements to construct or discuss the content in a logic model—not all associations presented in logic models are actually *causal* associations (as is implied by *if-then* statements). For example, a program can contribute to outcomes in a community that other organizations address too. It is important to explain *how* activities create or contribute to the benefits, changes, or results presented in the logic model.

☐ **Specify a time frame for each category of outcomes**. Estimate the time necessary to achieve each category of outcomes based on the nature of the programming, information from similar programs, stakeholder input, or information specific to the setting in which the program takes place. For example, time-limited funding for the program can dictate the window of opportunity for program implementation and achievement of desired results. Alternatively, research literature or programmatic experience can indicate that full achievement of a specific outcome occurs more than a year following program implementation. Be aware that a short-term outcome in one setting can be an intermediate or long-term outcome in another (e.g., due to features of the environment or resources dedicated to the program). Specify and explain the rationale for these time frames in the narrative that accompanies the logic model.

## References


[1] Taylor-Powell, E., Jones, L., & Henert, E. (n.d.). Enhancing program performance with logic models. Retrieved from https://fyi.uwex.edu/programdevelopment/files/2016/03/lmcourseall.pdf

[2] W. K. Kellogg Foundation. (2004). Logic model development guide. Retrieved from https://www.bttop.org/sites/default/files/public/W.K.%20Kellogg%20LogicModel.pdf

[3] Frye, A. W., & Hemmer, P. A. (2012). Program evaluation models and related theories: AMEE Guide No. 67. *Medical Teacher*, *34*(5), e288–e299.

[4] Centers for Disease Control and Prevention. (1999). Framework for program evaluation in public health. *Morbidity and Mortality Weekly Report*, *48*, 1–40.

[5] Savaya, R., & Waysman, M. (2005). The logic model: A tool for incorporating theory in development and evaluation of programs. *Administration in Social Work*, *29*(2), 85–103.

[6] Frechtling, J. (2007). *Logic modeling methods in program evaluation*. San Francisco: Jossey-Bass.

[7] Cullen, P., Clapham, K., Byrne, J., Hunter, K., Senserrick, T., Keay, L., & Ivers, R. (2016). The importance of context in logic model construction for a multi-site community-based Aboriginal driver licensing program. *Evaluation and Program Planning*, *57*, 8–15.

[8] Chen, H. T. (2015). *Practical program evaluation: Theory-driven evaluation and the integrated evaluation perspective* (2nd ed.). Thousand Oaks: SAGE Publications.

[9] Bickman, L. (1987). The functions of program theory. *New Directions for Program Evaluation*, 5–18.

[10] Funnell, S., & Rogers, P. (2011). *Purposeful program theory: Effective use of theories of change and logic models*. San Francisco: Jossey-Bass.

[11] Eagleson, R. (2016, December 27). Short definition of plain language. Retrieved from http://www.plainlanguage.gov/whatisPL/definitions/eagleson.cfm

## Acknowledgments

I developed this checklist with generous support and encouragement from Michael Scriven and The Faster Forward Fund (http://fasterforwardfund.org/). My most sincere thanks to Lori Wingate, Ellen Taylor-Powell, Valerie Caracelli, Kate Glynn, Rene Lavinghouze, Basia Tomczyk, Anita McLees, and two anonymous reviewers for detailed feedback on earlier versions of this document. I also appreciate Michelle Panneton and colleagues at the CDC Foundation who contributed to the preparation and dissemination of this work. Finally, I am indebted to those who took the time to field-test this checklist and share thoughtful recommendations for improvements: Scott Chaplowe, Cassandra Davis, Brittnee Hawkins, Melissa Jennings, Teresa Kinley, Michelle Leander-Griffith, and Christopher Voegeli.

The findings and conclusions in this document are those of the author and do not necessarily represent the official position of the Centers for Disease Control and Prevention.

## Suggested Citation

MacDonald, G. (2018). *Checklist of key considerations for development of program logic models.* Retrieved from https://wmich.edu/evaluation/checklists