

THE TEMPORAL LOGIC MODEL CONCEPT

Molly den Heyer
Ottawa, Ontario

Abstract: This article proposes an alternative program logic model based on the concepts of learning organizations and systems theory. The piece redefines time from a space or duration in which the preset program activities take place, to time as an evolutionary process in which the implementation of the program stimulates learning and adaptation. The latter concept of time was embedded into an alternative logic model, titled Temporal Logic Model. It provides a space for stakeholders to record changes in program context, interim assessments, and program modifications in a format that requires periodic updates. The documentation of how and why the program changes over time should enhance the program's accountability as well as learning.

Résumé: L'article qui suit propose un nouveau modèle de logique de programme axé sur les concepts des organisations d'apprentissage et la théorie des systèmes. L'article redéfinit le temps non comme un espace ou une durée où les activités pré-établies du programme se déroulent, mais comme un processus évolutionnaire dans lequel la mise en oeuvre du programme stimule l'apprentissage et l'adaptation. Ce concept du temps a été intégré à un nouveau modèle de logique intitulé le modèle de logique temporelle. Il offre aux intervenants un espace où ils peuvent consigner les changements au contexte du programme, les évaluations intérimaires et les modifications au programme et ce, dans un format qui exige des mises à jour périodiques. La documentation expliquant comment et pourquoi le programme change au fil du temps devrait rehausser l'imputabilité du programme, ainsi que de l'apprentissage.

■ The Logic Framework Analysis (LFA) was invented in the 1960s by a team of consultants led by Leon Rosenberg (Gasper, 1997). Despite widespread criticisms, its use has proliferated throughout the international development field. Over the last 40 years it has become part of standard procedures for program and project management and evaluation at the Canadian International Development Agency, the Danish International Development Agency, the German Agency for Technical Corporation, the Norwe-

gian Agency for Development, the British Oversea Development Agency, the Swedish International Development Agency, and the US Agency for International Development

The LFA is part of a family of logframe analyses that includes results chains (primarily used in the Canadian domestic development field) and program logic models (primarily used in the healthcare field). Given slight variations of terminology and structure, the purpose of a logic model is to illustrate the program's components and how they logically link together. A logic model is also used to measure the success of the program against its original plans. Although current logic models can be continuously revised to reflect new program realities, they do not capture the contributing factors and fluid motion of how the program adapts to a chaotic environment and the stakeholders who are constantly learning.

In order to address this issue, the International Development Research Centre's (IDRC) Evaluation Unit has researched and developed an alternative logic model that could include program responsiveness to environmental changes and the organizational learning process. The alternative model, Temporal Logic Model (TLM), achieves this through periodically recording contextual changes, making an interim assessment, and integrating any subsequent modifications. This article will provide a brief theoretical framework for the subsequent presentation of the TLM's attributes, outcome orientation, and internal mechanisms.

THEORETICAL FRAMEWORK

The debate surrounding logic models is rooted in our basic perception of programs and how we envision the process of implementation. General Systems Theory has three distinct branches that describe how we can perceive programs as systems. They are: hard systems, which are closed representations of structures; open systems, which are dynamic evolutionary processes of change; and soft systems, which are methods of modelling, interpreting, and planning interventions.

The hard systems approach formed the basis of social engineering, policy analysis, and management science that emerged in the post-WWII era. This type of approach describes a cause and effect process contained in a closed environment, and introduced terminology such as inputs, outputs, and feedback loops (Friedmann, 1987). In

the design phase, potential issues and components are identified and assembled (Checkland, 1981). Implementation is merely a matter of executing the prescribed activities according to the plan. The success of the system or program can then be measured by comparing the end results against the original blueprint.

The introduction of feedback loops expanded the concept of closed systems by illustrating some aspects of the process. While feedback loops enrich the models, they still rely on the designer's powers of foresight to establish possible connections and influences over the system before implementation. This is connected to two key assumptions: the original designers/planners have comprehensive knowledge of the situation, and there are no external factors that could interfere with implementation.

As mentioned above, logic models provide a sketch of the program components and how they are linked together. Although useful for providing a quick administrative overview, they have been highly criticized for presenting a closed system image that rarely occurs in the field. Figure 1 depicts the Logical Framework Analysis (LFA) as a model of a "fixed" or closed system.

Ludwig von Bertalanffy's distinction between closed and open systems sparked the development of a more organic view of systems as a continual process accessible to outside influences (Checkland,

Figure 1
Standard Logical Framework Analysis (LFA)

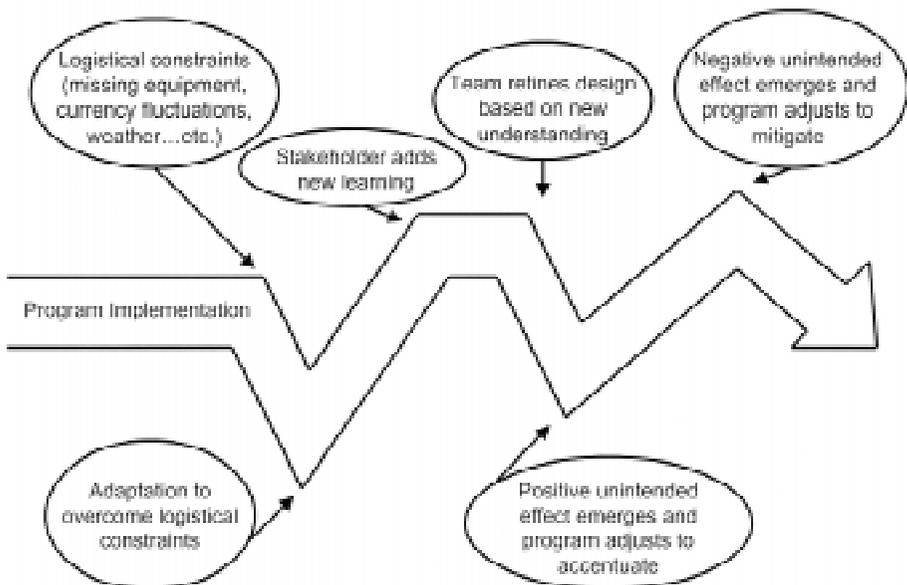
	Narrative Summary	Objectively Verifiable Indicators	Means of Verification	Assumptions
Goals				
Purposes				
Output				
Input				

1981). Open systems create a more dynamic and complex understanding of systems that better reflects the program and project implementation process. They contain several key elements summarized by Carden:

The *process* in a system is copoetic; order is created out of chaos through a collective process of interaction. The *structure* is dissipative; that is, order is achieved through fluctuation; stability is long-term not short-term and in the short term structures may appear highly unstable and uncertain. *Evolution* is self-transcendent. That is, the evolution of a system is through its changing its own consciousness and breaking the symmetry in which it exists. (1990, p. 51)

Figure 2 illustrates that the program is in an ongoing dialogue with complex external factors. Program design should be modified as the program adapts to situations where program assumptions were undermined, unintended effects occurred, or the program environment changed. From this perspective, program management becomes more of an art form than a science.

Figure 2
Program Implementation in an Open System



The open systems perspective can be found in a variety of conceptual frameworks including the social learning paradigm. This approach states that individuals/groups who are involved in program activities should enter into a learning process. The process consists of continuous learning loops of action, reflection, and adaptation in which the stakeholders gain tacit and explicit knowledge (Korten & Klaus, 1984). As a result of the learning process, the program design and activities can change several times as actors engage in a dialogue with the self, each other, and their former decisions. Thus, the stakeholder's learning becomes both subjected to and the cause of program fluctuations. This type of responsiveness and incorporation of lessons learned is an integral part of a learning organization and a primary characteristic of open systems.

Soft systems are unique in General Systems Theory because they identify a method of interpreting systems rather than describing characteristics and rules surrounding the system itself. According to Checkland and Scholes:

The basic shape of the approach is to formulate some models which it is hoped will be relevant to the real-world situation, and use them by setting them against perceptions of the real world in a process of comparison. That comparison could then initiate debate leading to a decision to take purposeful action to improve the part of the real life which is under scrutiny. (1990, p.6)

Checkland and Scholes (1990) further describe how soft system models are in a continuous process of feedback, assessment, and adjustment. Through this cycle the stakeholders are continually trying to incorporate more and more complexity to gain better representation of our immense reality. The soft system's cycle can easily be corresponded with the learning cycle (described above as action, reflection, and adaptation). Thus, it can be said that the stakeholder is utilizing a soft systems approach of interpreting an open system of which s/he is a part.

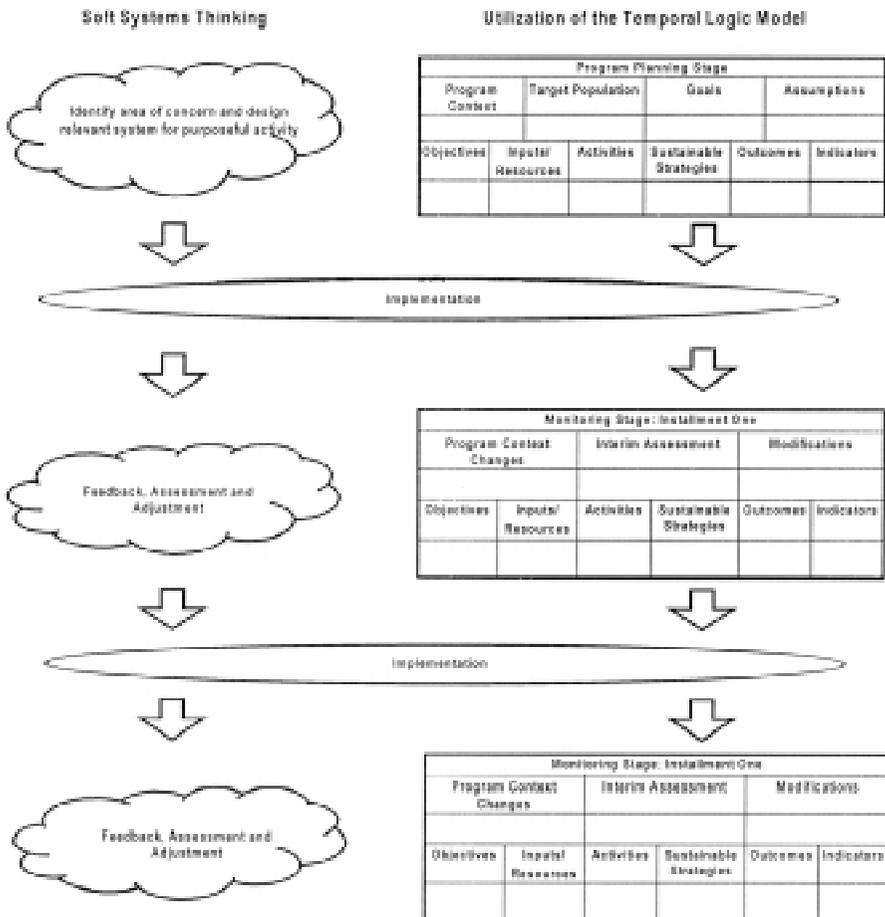
Given this analysis, traditional logic models illustrate a hard systems perspective of social development programs. In contrast, the TLM was designed as a soft system that could assist stakeholders to understand the program as an open system, essentially expanding the model to reflect an increasing understanding of our reality. This allows the stakeholder's understanding and documentation of

the program to evolve with the program's implementation. These concepts are illustrated by Figure 2, which depicts program implementation in an open system, and Figure 3, which juxtaposes soft systems thinking with the implementation of the TLM.

TLM ATTRIBUTES

The development of the TLM was approached much like a puzzle. The dimension of time was combined with the positive attributes from existing logic models and fitted into an easily understood for-

Figure 3
Soft Systems Thinking and the Utilization of the Temporal Logic Model



mat (shown in Figure 4). Although the general purpose is to provide program stakeholders with an effective tool to monitor process, the TLM's potential has evolved to include several additional theoretical and practical attributes that standard logic models lack. The possible range of attributes for the TLM are:

1. The TLM is *complementary to the social learning process*. By recording program context changes and interim assessment, the TLM encourages reflection on past and present activities. The emphasis on revising the program design row promotes critical thinking about its structure. Furthermore, it tracks the program's adaptation as stakeholders learn more about the issues. For these reasons, the TLM supports the learning process approach to social development.
2. The TLM *does not have a specified time-frame*. The term "Project-ism" is used by Gasper (1997) to illustrate the development industry's obsession with short-term programs and projects. He states that the LFA supports this tradition by illustrating only one set of interventions within a specific time-frame. The standard logic models cannot move beyond the first phase of a program to incorporate a long-term vision. In contrast, the TLM incorporates a series of

Figure 4
The Temporal Logic Model (TLM)

Program Planning Stage					
Program Context	Target Population	Goals	Assumptions		
	Modification Row: Installment One				
Objectives	Program Context Changes	Interim Assessment	Modifications		
	Modification Row: Installment Two				
Objective	Program Context Changes	Interim Assessment	Modifications		
Objectives	Inputs Resources	Activities	Sustainable Strategies	Outcomes	Indicators

stages that grow with the program. The stages should record and monitor the different phases without a specified end.

3. The TLM provides the flexibility to *be responsive to the program context*. It is an articulation of the dynamic relationship between what Shadish (1987) termed the microtheory and the macrotheory. The microtheory, an expression of the internal workings of a program, is envisioned as a program's "black box" in a clinical environment. Yet, the "box" is often distorted from the unpredictable environment in which it is situated. The program environment is chaotic, and changes can occur in a variety of areas including changes in the political economy, weather shifts, stakeholders learn new information, equipment does not arrive, and so on. Adaptability is essential for programs to maintain a steady course toward their original goals.
4. The TLM focuses on *monitoring and recording the process*. Monitoring systems support, if not provide, a catalyst for the learning process. They promote intermittent change in order to correct small problems before they become larger.
5. The TLM can be used as a *strategic decision-making tool*. The stakeholders go through the process of recording program context changes and any monitoring results. Based on this layout, they can then openly discuss the key issues and possible modifications to the program design, and record them in the subsequent program design rows. The process of reflection can promote a dialogue among the stakeholders in order to prompt strategic decision-making.
6. The TLM *captures unintended results*. The stakeholders can expand their causation focus to include unforeseen results by allowing outcomes to change throughout the lifespan of the program. Once the unintended results are incorporated into the framework, actions can then be taken to mitigate the negative effects and accentuate the positive effects.
7. The TLM provides a *chronology of program history*. It builds up an institutional memory and conveys key information for a summative evaluation by recording the program context changes, interim assessment, and modifications. This is essential to an organization that is dealing with a high

degree of change, is geographically diverse, or has a high turnover of staff.

8. As a monitoring tool, the TLM *promotes dialogue between the donor and recipient*. On the one hand, the TLM requires justification for any changes to the program design. On the other hand, it provides the field worker with a flexible framework, allowing him or her to be responsive to the program context. This balance can act as a catalyst for more informed dialogue between the levels of administration.

In addition to the theoretical characteristics, a number of practical issues are embedded in the design. They are as follows:

1. The TLM is *applicable to different stages of the program*. It can be used to test the logical linkages of the program, to monitor progress and lessons learned during the implementation phase, and, finally, to document the program history for a summative evaluation.
2. *The structure of the TLM should explicitly illustrate that the program is an ongoing process in which monitoring and periodical reflection of the program design is necessary*. Although some authors, such as Sawadogo and Dunlop (1997), recommend that logic models be revised as the program changes, it is not explicit in their structure. By leaving the open spaces to be filled in later, the structure explicitly identifies the need for updates.
3. The TLM was formatted to read like a newspaper, without cross-referencing within the framework. *Keeping the design simple* is important for three reasons: it can be easily read and understood by all stakeholders, it is a simple technology that can be applied anywhere in the world, and it can be used in a group setting.

Given the nature of implementation, problems with the TLM are bound to arise. Every tool is subject to the dangers of conceptual misunderstandings and misuse. There are also a number of program-specific issues (entrenched hierarchical structures, irreconcilable differences among stakeholders, etc.) that can impede the development of a program vision, a multi-stakeholder approach, or a program modification. These types of situations highlight the fact that

logic models are simple tools designed to assist humans through program planning and management. Without stakeholder “buy-in” to actualize the tool, the logic model remains just a chart.

OUTPUTS, OUTCOMES, IMPACTS, AND SUSTAINABLE STRATEGIES

The literature on outputs, outcomes, and impacts reveals a host of definitions with slight variations. Generally, the literature associates the terms with immediate, short-term, and long-term results of a program or a project. The time-frame of the results also corresponds with the depth of the information and the types of changes. Because profound results tend to take a longer time to emerge, outputs, outcomes, and impacts should have deeper and deeper roots in behaviour and societal structures as time progresses.

Outputs are closely related to the activities or program “deliverables.” They are usually the number of workshops held, the number of clients, the number of grants distributed, and so on. Often already captured in the activities category, this type of detailed accounting was deemed not necessary for the general overview of program design illustrated by the TLM.

Oakley, Pratt, and Clayton define outcomes in the statement:

The crucial first stage of measurement will be to assess what has been the outcome of the project in terms of the *effect* it has had on the initial situation.... by effect we mean the more immediate tangible and observable change in relation to the initial situation and established objectives, which it is felt has been brought about as a direct result of the project. (1998, p. 35)

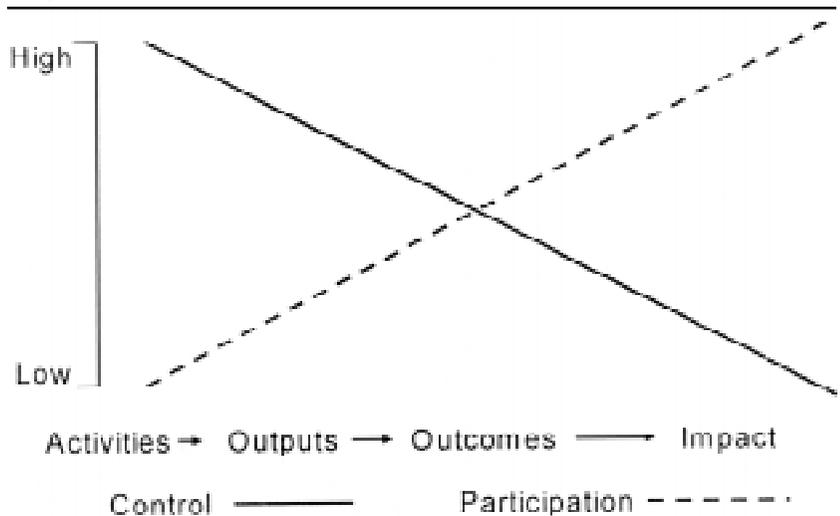
From this description, outcomes become the focus of the majority of evaluations due to their accessible time frame and depth of change.

Impacts are the long-term results that imply a broader societal change that may only be measured several years after the program is completed. Some examples of impacts are an increased standard of living, human resource development, gender equality, and increased political awareness of environmental issues. Although this information is highly desirable for the stakeholders, impacts are rarely measured because of the length of time it takes for real impacts to emerge as well as being too general for direct attribution (Oakley, Pratt, & Clayton, 1998).

In an effort to address the ambiguity surrounding the categorization of different results, IDRC's Outcome Mapping is directed solely toward outcomes. The approach redefines outcomes to accentuate contextual change without restricting it to a specific time frame. For Outcome Mapping, outcomes are "changes in the behaviour, relationships, activities, or actions of the people, groups and organizations with whom the program works directly" (Earl, Carden, & Smutylo, 2001, p. 1). This reflects a shift in focus from evaluating changes in state, to evaluating changes in behaviour. Outcome Mapping also substitutes attribution for contribution by later emphasizing that "outcomes can be logically linked to a program's activities, although they are not necessarily directly caused by them" (Earl et al., 2001, p.1).

As illustrated in Figure 5, the depth or extent of change in behaviour forms an interesting outcome dynamic with accountability. As one moves from outputs to impacts, the level of program control or influence decreases. Conversely, the level of stakeholder participation increases along the same path. The results are inadequate participation for behavioural changes in outputs and inadequate program control for accountability in impacts. The two variables in-

Figure 5
Program Control and Participation



Source: Smutylo, 2001; Woodhill, 2000.

intersect at outcomes, providing partial participation to encourage behavioural change and partial program control to allow for accountability. Thus, outcomes become the target for measuring behavioural changes in programs (Smutylo, 2001; Woodhill, 2000).

In an effort to maintain a long-term emphasis without entering an expensive and undetermined search for impacts, the TLM replaced impacts with sustainable strategies. This still keeps an emphasis on long-term planning, but focuses on the ex-ante and implementation program stages rather than ex-post. Sustainable strategies ask the question, "What are the program strategies for ensuring that the initiative continues beyond your program's involvement?" This asks the program stakeholders to plan several organizational strategies to ensure that the activities are fully integrated into the target community. The strategies are not intended to promote self-perpetuating organizations but to assist the long-term goals of the program to take root inside the target community. Sustainable strategies can include empowerment activities, mentoring community members in program management, revolving funds for development, and so on.

The Australian International Development Agency (AusAid) has developed a similar concept of the same name. They define sustainable strategies as "the continuation of benefits after major assistance from a donor has been completed." Furthermore, they state that sustainable strategies should be clearly identified in the program's design stage and individually created on a case-by-case basis (AusAid, 2000).

The TLM provides a unique results approach by incorporating a more rounded definition of outcomes and sustainable strategies in process-orientated framework. It is intended to focus on meaningful change without over-reporting on details and without attempting to track attribution for long-term effects. The long-term vision is maintained by illustrating strategies that actively seek sustainability. By creating a space to illustrate both process and results, the TLM seeks to reach a balance between the two approaches.

THE TLM'S TECHNICAL ASPECTS

Simply defined, the Temporal Logic Model is an evaluation tool that provides an overall pictorial representation of the program components and tracks design modifications during implementation. By

including the temporal aspects through a series of stages listed in a table format, the TLM captures program context changes, modifications, and both intended and unintended results. These characteristics should enable the TLM to monitor the learning process.

The Temporal Logic Model reads like a newspaper. It comprises two stages, with the second stage repeated throughout the program. The first stage embodies the original program vision and is referred to as the program planning stage. The program planning stage is followed by a series of monitoring stages, which record changes over the span of the program.

The project planning stage should be completed with the initial program design documents. Its subsequent implementation can vary in frequency depending on the size, length, intensity, and complexity of the program. In deciding on the frequency, it is important to remember that the TLM should be used to monitor the program and assist in the decision-making process. Thus, in addition to a regular monitoring schedule, the TLM should be updated whenever the program design is modified.

Like other logic models, the TLM contains two forms of inquiry. First, the individual components ask for specific information. Second, the components should link together to form a logical statement. Coleman (1987) describes how the logic models are constructed to express program rationale in a series of “if-and-then” statements. These statements are the underpinnings of the model. If the linkages are not tested correctly, then the model is limited to what Gasper (2000) refers to as a “box-filling” exercise. This deflates the models to the questions asked by each component, essentially overlooking the opportunity to test how the components fit together to form the program rationale.

The TLM has three lines of horizontal logic and one vertical (see Figure 6). The horizontal logic is embedded in the program context row, program design row, and modification row. The vertical logic is articulated through the series of installments or stages. The combined logical statements should convey the program story.

Program Planning Stage

The program planning stage should be used to help design and test the logic of the program. It consists of two rows, the program con-

Figure 6
Conveying the Program Story

Program Planning Stage						
Program Context		Target Population		Goals		Assumption
Objectives	Inputs/Resources	Activities	Sustainable Strategies	Outcomes	Indicators	
Installment One: (Add date here)						
Program Context Change			Interim Assessment		Modifications	
Objectives	Inputs/Resources	Activities	Sustainable Strategies	Outcomes	Indicators	

text row and the program design row. The program context row describes the context and issue that the program will address, while the program design row demonstrates how the issue will be addressed through the program.

Program Context Row

The first row in the program planning stage provides the background for the program. It includes program context, target population, goals, and assumptions. It is intended to convey the information necessary to understand the contextual aspects of the program. The four boxes are designed to create the statement, “*If these are the issues for this population, then we hope to create this change based on these premises.*” This row is unique in the table as it remains unchanged, providing an overall trajectory and stability for the program.

Program Context: What is the program environment? This section should describe the defining characteristics of the program environment. It includes general societal and physical trends that affect the target population. This type of component is not found in most logic models.

Target Population: Whom do you want to reach? The target population box should contain a clear statement of who will benefit from the program. It is important to note the distinction between target population and stakeholders. Stakeholders are a larger group, who are involved in and/or affected by the program, whereas the target population is the program’s intended audience. The target population was incorporated into the TLM in order to shape the program and to focus attention on the beneficiaries. Also known as program reach, it is included in some versions of the results chain, but rarely incorporated into LFAs.

Goals: Where are you going? Goals reflect the overall direction of the program. Ideally they are clear, concise, and commonly agreed upon. It is important to balance the goal statement between attainable targets and motivating statements. The TLM does not hold the program directly accountable for obtaining the goals by placing the component in the program context row, thereby allowing the statements to be motivating. Although goals are not present in results chains, they are found in Logical Framework Analysis (LFA) and program logic models.

Assumptions: What are the influences beyond staff control? Assumptions are the external influences outside the control of the program. Although they are only found in LFAs, they represent a critical aspect of the program and project management. Assumptions prompt stakeholders to inquire about which aspects of the program are unstable and beyond the control of the program's staff. These elements exist in every program and require recognition and constant monitoring. Alison Mathie (personal communication, August 2000) distinguishes between two types of assumptions: circumstantial and theoretical. Circumstantial assumptions are embedded in the context in which the program is situated. They may include political unrest, natural disasters, drops in currency, or the cooperation of other organizations. Theoretical assumptions are relatively more controllable. Mathie argues that the implementors can establish a body of evidence for the program theory by correlating the assumption with the literature. This involves a closer examination of the dynamics and variables surrounding the casual linkages.

Program Design Row A

The second row is based on versions of LFAs, logic models, and results chains. It includes objectives, inputs/resources, activities, sustainable strategies, outcomes, and indicators. The program design row is repeated in every stage of the TLM. The horizontal logic should illustrate how the internal workings of the program fit together, thus articulating the program's theory of intervention or Shadish's microtheory (1987). It states, "If this is what we want to realistically accomplish, and these are our resources to carry out these activities, and we will ensure long-term results with these strategies, then we will have this effect, which will be illustrated through these variables."

Objectives: What do you want to happen? Clear objectives are necessary for program design, administration, and evaluation. They clarify the direction of the program and break the goal into specific and obtainable targets. Patton (1997) describes a good objective as "clear, specific, and measurable." Although objectives are included in the standard LFA, they are frequently missing from results chains and logic models.

Inputs/Resources: What do you have? Inputs/resources refer to the overall cost of the program. This includes employees' salaries and benefits, materials, use of machines, office space, travel, in-kind con-

tributions, and so on. Determining if results justify the cost is necessary to evaluate efficiency.

Activities: How will you make it happen? Activities refer to planned events, workshops, consultation, research, proposals, writing, and other tasks carried out under the program name. Although dominant in logic models, activities are not found in results chains and LFAs.

Sustainable Strategies: What are the program strategies for ensuring that the initiative continues beyond your program's involvement? This section requests the program stakeholders to plan several organizational strategies to ensure that the activities are fully integrated into the target community. They are not strategies for self-perpetuating organization but are intended to make the long-term goals of the program take root inside the target community. Sustainable strategies could include empowerment activities, mentoring community members in program management, a revolving fund for development, and so on.

Outcomes: What are the expected short-term results? Outcomes are also found in LFAs, results chains, and logic models. As discussed earlier, the TLM utilizes Outcome Mapping's definition of outcomes as "changes in the behaviour, relationships, activities, and/or actions of boundary partners that can be logically linked to, although are not necessarily directly caused by, a project or program." This avoids the trap of attribution by focusing the evaluation on behavioural changes to which the program's activities have contributed.

Monitoring Stage

The subsequent monitoring stages require the program staff to analyze and respond to the program environment and evaluation recommendations. These can occur at regularly appointed intervals or as the need arises. In this manner, the TLM increases the program flexibility, but also requires staff to justify and record program changes.

Modifications Row

The monitoring stages consist of a modification row and a program design row. The modification row includes sections on program context changes, interim assessment, and modifications. The embed-

ded horizontal logic statement is simply, “If the program context has changed in this manner and the interim assessment have shown this, *then* the program should (or should not) adapt in this manner.”

Program Context Changes: What are the external factors influencing the program? Program staff must be responsive to the emergence of new information and environmental changes. Unknown factors and known assumptions can undermine a program’s stability and render its activities ineffective, if not harmful. The TLM recognizes this reality by allowing the program to not only identify the assumptions, but also change the program in response to the chaotic environment. Under the heading “program context changes,” program staff may record the factors that influence the program implementation, including external societal trends and internal cultural trends.

Interim Assessment: What worked? What didn’t work? The interim assessment section refers to the results of interim evaluation, monitoring, or professional observation. It allows the program administrator to record activities or procedures that are or are not working.

Modifications: What changes have been/should be made, if any? Modifications refer to changes in program design, and should correspond with program context changes and interim assessments. In some cases changes will not be necessary, but this option should be left to program stakeholders.

Program Design Row B

The modification row is followed by the newly amended program design row. It will incorporate the changes and provide an updated program vision. It is important to note that a change in one component can have a domino effect throughout the model. For example, changing an objective could affect the budget allocation, activities, sustainable strategies, indicators, and outcomes. Therefore, all the components in the row should be checked and adjusted every time a modification is made. To increase clarity, changes in the program design row can be italicized or bolded.

Vertical Logic Statement

The vertical statement moves beyond one “*if and then*” statement to include an additional sentence at each stage of the program’s development. The result is an animated story that develops from the fol-

lowing script: “*If* these are the contextual issues the program needs to respond to *then* this is the intervention designed to address it. But *if* the program needs to be adapted *then* this would be the modified designed. The script builds a chronology of how and why the program is changing.

CONCLUSIONS

Traditional logic models are noted for being inflexible due to their roots in hard system or blueprint-style management. By adapting the structure to include a series of stages, the TLM monitors changes in the program design over time, increasing responsiveness and flexibility. Furthermore, the TLM's points of inquiry and logical statements prompt the stakeholders to reflect on results, sustainability, program design, and the process of implementation in order to review the program's effectiveness, efficiency, and rationale. This creates a soft system model for interpreting the program as an open system.

Given this analysis, the TLM is still in the conceptual stage. Repeated field applications are required before final conclusions can be drawn. Moreover, it also has to be recognized that the logic model is a management tool. Whether the TLM illustrates an open or closed system will depend on how it is implemented. This discussion is not intended to minimize the importance of a reflective practitioner, but merely propose a better tool to carry out their tasks.

Gordon Framst (personal communication, October 2000) stated that three logic models always exist in the program at the same time: the model of what the program is supposed to be; the model that represents how the project is actually being implemented; and the model that illustrates how the program should be implemented. By these definitions, standard logic models represent what the program is “supposed to be.” The TLM expands on the “supposed to be” model by including “how it is currently being implemented” in the modification row and “how it should be implemented based on new learning” in the subsequent design rows. Understanding the tension between these concurrent models should reveal valuable program lessons.

ACKNOWLEDGEMENTS

I would like to thank Dr. Fred Carden of the International Development Research Centre (IDRC) and Dr. Harry Cummings and David Douglas at the University of Guelph for their guidance and support.

Earlier versions of this manuscript received honourable mention from the Canadian Evaluation Society Essay Competition in 2000, appeared in my Master's thesis, and were revised as an IDRC report.

REFERENCES

- AusAid. (2000, September). *Promoting practical sustainability*. Australia: PIA/OPRE.
- Carden, F. (1990). *Boundary management: International collaboration between OECD countries and Sub-Saharan African countries*. Unpublished doctoral dissertation, Université de Montreal, Montréal, Quebec.
- Checkland, P. (1981). *Systems thinking, systems practice*. Toronto: John Wiley & Sons.
- Checkland, P., & Scholes, J. (1990). *Soft systems methodology in action*. Toronto: John Wiley & Sons.
- Coleman, G. (1987). Logical framework approach to the monitoring and evaluation of agricultural and rural development projects. *Project Appraisal*, 2(4), 251–259.
- Earl, S., Carden, F., & Smutylo, T. (2001). *Outcome mapping: Building learning and reflection into development programs*. Ottawa: International Development Research Centre.
- Friedmann, J. (1987). *Planning in the public domain: From knowledge to action*. Princeton, NJ: Princeton University Press.
- Gasper, D. (1997, December). *'Logical Frameworks': A critical assessment managerial theory, pluralistic practice* (Working Paper Series No. 264). The Hague, The Netherlands: Institute of Social Studies.
- Gasper, D. (2000). Evaluating the 'Logical Framework Approach' — Towards learning-oriented development evaluation. *Public Administration and Development*, 20(1), 17–18.
- Korten, D.C., & Klauss, R. (1984). *People centred development*. West Hartford, CT: Kumarian Press.

- Oakley, P., Pratt, B., & Clayton, A. (1998). *Outcomes and impact: Evaluating change in social development*. Oxford, UK: INTRAC.
- Patton, M.Q. (1997). *Utilization-focused evaluation: The new century text* (3rd ed.). Newbury Park, CA: Sage.
- Sawadogo, J.-B., & Dunlop, K. (1997). Managing for results with a dynamic logical framework approach: From project design to impact measurement. *Canadian Journal of Development Studies*, 18, 597–612.
- Shadish, W.R. (1987). *Program micro- and macrotheory: A guide for social change* (New Directions for Program Evaluation). San Francisco: Jossey-Bass.
- Smutylo, T. (2001, May 23). *Crouching impact, hidden attribution: Overcoming threats to learning in development programs*. International Development Research Centre (IDRC), Evaluation Unit. Paper presented to Block Island Workshop on Cross-Portfolio Learning. Retrieved from www.idrc.ca/evaluation/
- Woodhill, J. (2000, March). *Introduction to key concepts, approaches and terms*. Geneva: The World Conservation Union (IUCN).

