Program and Evaluation Planning Lite: Planning in the Real World

Justus J. Randolph, Pasi J. Eronen
Department of Computer Science
University of Joensuu
P.O. Box 111
FIN-80101 Joensuu
Finland
justusrandolph@yahoo.com, pasi.eronen@cs.joensuu.fi

Abstract

Background: Although there are many high-quality models for program and evaluation planning, these models are often too intensive to be used in situations when time and resources are scarce. Additionally, there is little added value in using an elaborate and expensive program and evaluation planning procedure when programs are small or are planned to be short-lived.

Purpose: To meet the need for simplified models for program and evaluation planning, we describe a model that includes only the most essential outcomes-based program and evaluation planning steps.

Evaluation Design: The four steps described in this article include

- how to create a logic model that shows how the program is causally expected to lead to outcomes, (the role of mechanisms, moderating mechanisms, and links between mechanisms are discussed);
- how to use the logic model to identify the goals and objectives that the program is responsible for;
- how to formulate measures, baselines, and targets from the goals and objectives; and
- how to construct program activities that align with program targets.

Examples from a computer-science-oriented HIV/AIDS prevention project are given for each step.

Conclusions: The model described in this article is less time-consuming and resource intensive than other full-scale models but is still within the realm of good practice in program and evaluation planning.

1. Introduction

In the ideal professional world, program planners and evaluators in computer science education (CSE), and most other educational endeavors, would have unlimited time and resources to design high quality programs, to make evaluation plans, and to get projects generously funded. However,

program designers and evaluators do not live in an ideal professional world. For most, the reality of the professional situation is a continual struggle against ever-impending deadlines and limited resources.

In terms of CSE program design and evaluation planning, there are many high quality models (Billings, 1985, 1986; Randolph, Virnes, & Eronen; 2004, and Torvinen; 2004), yet it has been found that in many cases they are too heavy, too complicated, too time-consuming, or require too many resources (Almstrum et al., 1996; Randolph & Hartikainen, 2004). This is especially true for small, short programs that are intended to be carried out by only a few practitioners.

The model described in this article addresses the need for practical and realistic program and evaluation planning models in computer science education. In fact, the motivation for writing this article was a result of criticism received about the feasibility of a more complicated CSE/technology education evaluation and planning model, which the authors of this paper had reported in Randolph et al., (2004).

This article is not intended for planning and evaluation professionals since the information presented here is far from original. Rather, this article is intended to bridge the research-to-practice gap for practitioners in the computing sciences, or other fields, who have been given the task of planning or evaluating programs. This model is appropriate when time is of the essence or when the program is so small that full-scale program and evaluation planning is inappropriate.

Examples from this article are based on a project proposal for using ICT and computer science education as a tool for HIV/AIDS prevention (See Duveskog, Sutinen, Tedre, & Vesisenaho, 2003 and Duveskog, Sutinen, Vesisenaho, & Gasso, 2003 for background on the prevention project). Although the authors of this article put this model in the context of computer science education for HIV/AIDS prevention, there is no reason that it would not work in other contexts as well. The model is simple; if it is followed correctly, it should stay within the realm of appropriate practice in program and evaluation planning. The evaluation portion of the model is especially appropriate for summative, goal-oriented evaluation (e.g., Tyler, 1949) rather than evaluation for program improvement or organizational learning (e.g., Preskill & Torres, 1999).

2. Program and Evaluation Planning Lite

The following sections summarize the Program and Evaluation Planning Lite model (hereafter [*the Planning Lite model*].) It consists of four systematic steps:

1. Creating a logic model
2. Identifying goals and objectives
3. Formulating measures, baselines, and targets
4. Aligning program activities with targets

Figure 1 illustrates the logic of the Planning Lite model. Creating a logic model clarifies the nature of the program and enables goals and objectives to be identified. After goals and objectives have been identified, measures, baselines, and targets can be formulated. Finally, program activities can be planned that so that they cause the targets to be met. If everything works out as planned, (e.g., (a) if the logic model is correct, (b) measures are valid and reliable and (c) align with objectives and activities, and (d) activities cause targets to be met), the Planning Lite model should enable successful programs and evaluations to be designed when time and resources are scarce.

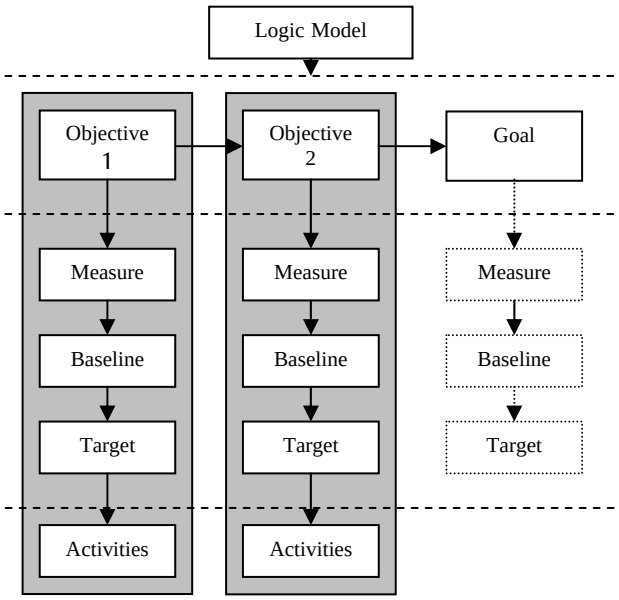

Figure 1. The Planning Lite Model

2.1 Logic Model

Weiss (1998) defines a logic model (she calls it a *program theory*) as "the *mechanisms* that mediate between the delivery (and receipt) of the program and the emergence of the outcomes of interest." Simply put, a logic model is an illustration of the causal links between the program and the outcomes of the program. It consists of mediating mechanisms, moderating mechanisms, and links between mechanisms. A mediating mechanism is simply a step in a causal chain. A moderating mechanism is one that affects the links between one or more mechanisms. A moderating mechanism can stop or affect the progression of the causal chain.

To make an analogy, in a chain of falling dominoes the program would be the first domino, the overall program goal would be the last. The dominoes in between the first and last would be the mediating mechanisms. The block or barricades that interrupt the chain of falling dominoes would be the moderating mechanisms.

Creating a logic model is useful for a number of reasons. First, it helps program designers, evaluators, and grant writers understand the program and codify their perceptions about the nature of the program. Secondly, it serves as a framework for the rest of the program and evaluation planning process.

Figure 2 illustrates a logic model for an ICT-related HIV/AIDS prevention program. This figure illustrates how the prevention program is intended to bring about knowledge of prevention strategies, HIV/AIDS-related behavioral change, reduction of new HIV/AIDS cases, and, subsequently, to global health. Knowledge of prevention strategies, HIV/AIDS-related behavioral change, and reduction of new HIV/AIDS cases are the mediating mechanisms between the program and its long-term goal – global health. The moderating mechanisms in this logic model are the accessibility and utilization of the program since they can interrupt the flow of the causal events in the logic model if they are not addressed.

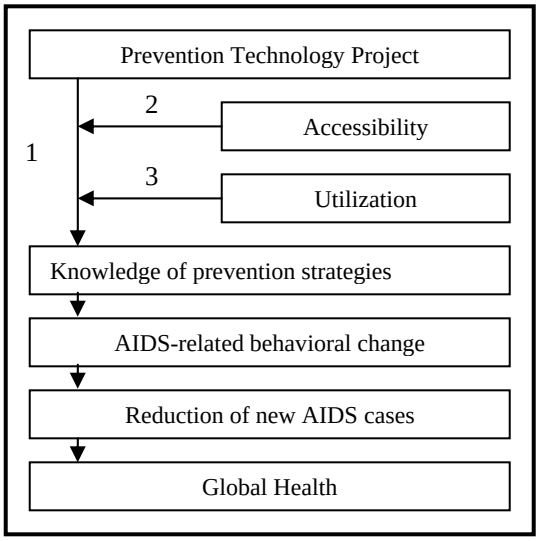

Figure 2. The Logic model example

It is helpful when making a logic model to brainstorm all of the stakeholders and systematically answer each of the following questions for each stakeholder group:

- What are the needs of this stakeholder group?
- What are the benefits and risks that this program or evaluation will have for this group?
- How should this stakeholder group be included in the planning of this program of evaluation?

Considering the stakeholder questions will add depth and focus not only to the logical model, but to the rest of the planning process as well. Additional resources for constructing a logic model can be found in CDC Evaluation Working Group (n.d.); Julian, Jones, and Deyo (1995); McLaughlin and Jordan (2004); Program Development and Evaluation, University of Wisconsin – Extension, (n.d.); and W.K. Kellog Foundation (2001).

2.2 Goals and objectives

After the logic model has been constructed, defining the program's objectives and goals becomes rather easy. Although objectives and goals are synonyms, in this paper we will refer to *objectives* as the mechanisms that are within the scope of the program's activities and *goals* as the mechanisms outside the scope of the program's activities. The mediating and moderating mechanisms that fall under the scope of the program in the logic model are the program objectives. The last step in the logic model is the overall goal or goals that should describe the benefits to the program beneficiaries. The mediating mechanisms outside of the scope of the program might be referred to as intermediate goals.

It is important to note that the program will usually not be accountable for all of the mediating and moderating mechanisms in the logic model. The decision about which mechanisms to include as program objectives, is usually based on several factors. The first factor depends on where in the logic model the causal chain will continue without program intervention, either because the links are firmly established or they logically follow. The second factor depends on the amount of

resources available to the program. Referring back to Figure 2, the program designers decided that the program needs only to be accountable for increasing the knowledge of intervention strategies and ensuring the accessibility and utilization of the program. It was decided that if these objectives (knowledge, accessibility and utilization) were accomplished, the rest of the mechanisms (HIV/AIDS-related behavior change, reduction of new HIV/AIDS-cases) would occur without further program intervention and contribute to the overall goal of global health. Examples, then, of the overall goal and objectives derived from the program theory of the HIV/AIDS prevention program presented in Figure 1 are listed below:

- Overall Goal – to increase global health
- Objectives
    - Create an ICT-education-based prevention program that increases knowledge of prevention strategies
    - Ensure that the program is accessible
    - Ensure that the program is utilized

If the logic model has been correctly thought out, the overall goal should be accomplished if the objectives are accomplished.

2.3 Defining measures, baselines, and targets

After the goals and objectives have been made clear, measures, baselines, and targets should be derived for the objectives (and if appropriate for the overall goal, and/or intermediate goals.) See Table 1 for an example of the measurements, baselines, and targets for each objective in the HIV/AIDS prevention example.

Table 1. AIDS Prevention Example

Objective	Measure	Baseline	Target
Prevention Knowledge	Student performance on measures of HIV/AIDS prevention knowledge	Performance on prevention measures from students before the intervention	Postintervention students will have educationally and statistically significantly higher scores on prevention measures. 95% mastery on prevention measures, overall
Accessibility	Number of students who are able to participate in the intervention	33% of students have access to prevention programs (UNAIDS)	90% of Tanzanian schools will gain access to the prevention program.
Utilization	Percent of schools able to participate and that choose to participate	0% utilization at inception of program	80% of schools will choose to participate in program.

2.3.1 Measures

For each objective, one or more valid and reliable instruments need to be adopted or constructed that can measure the causal impact of each mechanism on the next mechanism or on a link between mechanisms. For example, to measure the link between the program and increased

AIDS knowledge, the program designers decided to construct and validate a test of a student's knowledge of HIV/AIDS. To measure accessibility, the designers decided to measure the number of schools able to participate in the program. To measure accessibility, the designers decided to measure the number of schools that were able to participate that decided to participate. The overall goal and intermediate goals were not intended to be measured in the HIV/AIDS prevention example since the scope of the program's accountability ended at increasing knowledge. However, in other cases, it may be appropriate to measure overall and intermediate goals if resources are adequate.

2.3.2 Baselines and targets

In order to answer the question, "What was the impact of the program?", one has to answer two questions – "What was, or would have been, the state of the objective before, or without the program?" and "What is the state of the objective after, or with, the program?". A baseline measurement indicates the state of the objective before the program existed or predicts the state of the program had the program not existed. A target measurement indicates the intended state of the objective after, or with, the program. If the program has not yet been carried out, the target measurement indicates where the state of the objective is intended to be after, or with, the program. Pre-tests and control groups measurements are types of baseline measurements. Post-tests and experimental groups are types of target measurements. See Table 1 for an example of baselines and targets for the HIV/AIDS prevention program.

2.4 Activities

Finally, after measures, baselines, and targets have been formulated, it is now appropriate to design program activities that are aligned with the measures. Without examining how program activities lead to meeting targets, it is plausible that some influence other than the program led to the target being met. The activities should directly lead to the objectives meeting their target, as indicated by the measures.

3. Conclusion

This article presented a practical, outcomes-based model for program and evaluation planning that could be useful for practitioners in computer science education, or other evaluation and planning contexts. Its strength is its feasibility; it cuts the evaluation and planning process down to the most essential steps. The steps included (a) creating a logic model; (b) identifying goals and objectives; (c) formulating measures, baselines, and targets; and (d) aligning program activities with targets. The model's weakness is that the emphasis on evaluation feasibility comes at the price of a de-emphasis on evaluation utility, accuracy, and propriety. Furthermore, the model works best in rather static, foreseeable environments, which makes it weak in programs with unknown or undecided outcomes.

Overall, given that every precaution is taken to reduce harm to stakeholders, if there is only time or resources for a few evaluation and program planning activities, we suggest that these are the activities that be done.

Acknowledgements

This research was partly funded by a special projects grant from the Association for Computing Machinery's Special Interest Group on Computer Science Education (ACM SIGCSE). Authors want also to thank the European Social Fund (under grant ISLH-2002-04159/Ha-7) for the financial support.

References

Almstrum, V.L., Dale, N., Berglund, A., Granger, M., Little, J.C., Miller, D.M., Petre, M., Schragger, P., Springsteel, F. (1996). Evaluation: Turning technology to tool – report of the Working Group on Evaluation. *Proceedings of the 1st conference on integrating technology into computer science education* (pp. 201-217). New York: ACM Press.

Billings, K.J. (1986). The development of an evaluation handbook for a computer education program. *Dissertation Abstracts International*, 47 (06), 2131A. (UMI no. AAT 8620322)

Billings, K.J. (1985). *An evaluation handbook for a computer education program.* Eugene, OR: International Council for Computer in Education. (ERIC Document Reproduction Service No. ED291338)

CDC Evaluation Working Group. (n.d.). *Logic model resources.* Internet WWW-page, Retrieved November 2, 2004, from http://www.cdc.gov/eval/resources.htm#logic%20model

Duveskog, M., Sutinen, E., Tedre, M. & Vesisenaho, M. (2003). *In search of contextual teaching of programming in a Tanzanian secondary school*. Paper presented at the IEEE Frontiers in Education Conference, Boulder, Colorado, November 5th-8th, 2003.

Duveskog, M., Sutinen, E., Vesisenaho, M., & Gasso, C. (2003). HIV/AIDS education in Tanzania blended with a programming course. In *Proceedings of the 2003 International Conference on Information Technology: Research and Education (ITRE 2003)*. (pp. 179-183). Washington, DC: IEEE.

Julian, D. A., Jones, A., & Deyo, D. (1995). Open systems evaluation and the logic model: Program planning and evaluation tools. *Evaluation and Program Planning* 18, 333-341.

McLaughlin, J. A. & Jordan, G. B. (2004). Using logic models. In Wholey, J., Hatry, H. P., & Newcomer, K. E. (Eds.). *Handbook of practical program evaluation* (2nd ed., pp. 7-32). San Francisco: Jossey-Bass.

Preskill, H. & Torres, R. T. (1999). *Evaluative inquiry for learning in organizations.* Thousand Oaks, CA: Sage.

Program Development and Evaluation, University of Wisconsin- Extension. (n.d.). *Logic model.* Internet WWW-page, Retrieved November 2, 2004, from Http://www.uwex.edu/ces/pdande/evaluation/evallogicmodel.html

Randolph, J.J., & Hartikainen, E. (2004). *A review of resources for K-12 computer science education program evaluation*. Paper presented at Kasvatustieteen päivät 2004 (Educational Research Days Conference 2004), Joensuu, Finland, November 25th and 26th, 2004.

Randolph, J.J., Virnes, M., Eronen P.J. (2005) A model for designing and evaluating teacher training programs in technology education. In Courtiat, J-P., Davarakis, C., & Villemur, T. (Eds.), *Technology enhanced learning* (pp. 69-79). New York: Kluwer.

Torvinen, S. (2004). *Aspects of the Evaluation and Improvement Process in an Online Programming Course. Case: The ViSCoS program*. Licentiate Thesis, University of Joensuu, Finland.

Tyler, R. (1949): *Basic Principles of Curriculum and Instruction.* Chicago, University of Chicago Press.

Weiss, C. H. (1998). *Evaluation: Methods for Studying Programs and Policies* (2nd ed.). Upper Saddle River, NJ: Prentice Hall.

W. K. Kellogg Foundation. (2001). *Using logic models to bring together planning, evaluation, & action: Logic model development guide.* Internet WWW-page, Retrieved November 2, 2004, from http://www.wkkf.org/Pubs/Tools/Evaluation/Pub3669.pdf