

Running Head: PROGRAMME DEVELOPMENT MODEL

Programme Development Model

Brian Leahy, Christopher MacAulay, Diane Murphy, Robert Power, and Jing Yang

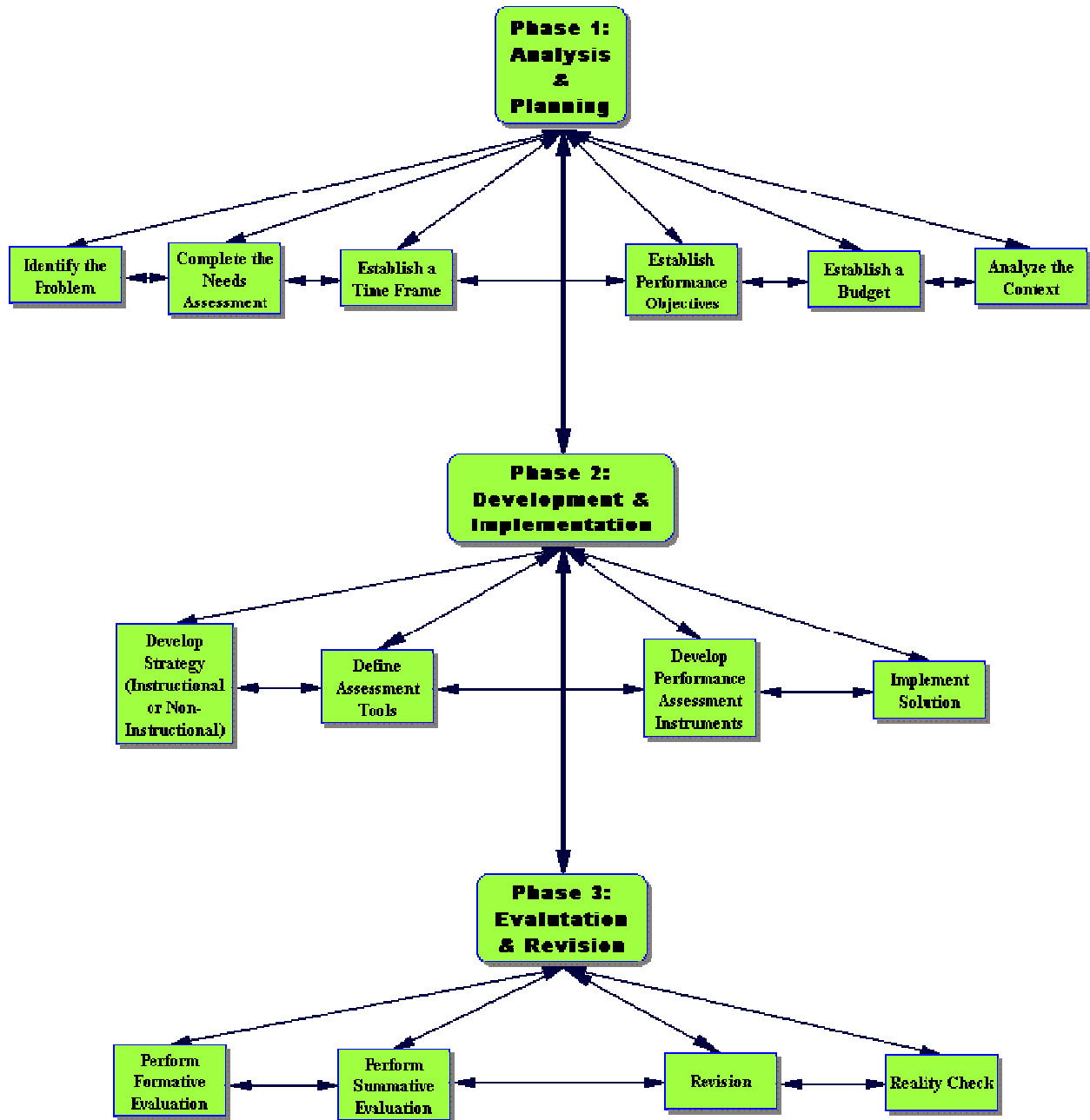
Memorial University of Newfoundland

In partial fulfillment of the requirements for ED6823

Dennis Sharpe

June 27, 2003

Program Development Model:



Programme Development Model

Our programme development model is based on a simple premise: Separate the tasks of programme design into common groups. What we found were three common groups visible in the programme design process, and we called these groups “Phases”. Each phase is a separate combination of tasks, but is not meant to be autonomous outside of the model itself. Each phase is part of the other two and, as we hope to have shown in our final phase, is part of an ongoing revision process.

Explanation of Model

Phase One – Analysis and Planning

Identify the problem.

Programme design begins when problems are seen in what is desired of a situation. Like a needs assessment, *identify the problem* points out discrepancies in “what is” and “what should be”. The difference between the two is in their complexity. In this beginning stage the task is to identify, not explain as in a proper needs assessment, and is therefore much less complex.

In most situations the problem that is first seen is only a symptom caused by the discrepancy or gap in that actual situation. It is not the true problem first seen in most cases, rather it is a perceived problem, and the true problem is hidden behind these symptoms. When followed to the cause the true problem emerges from behind these symptoms and is exactly what programme designers want to find to accurately solve the discrepancy.

The causes of problems stem from a deficiency in knowledge or in the environment, or a combination of the two. A deficiency in environment exists when there is a barrier to performance. For instance: Poorly explained job descriptions, or improper working environment. A deficiency of knowledge exists when people do not know how to perform because of not having the knowledge

to perform the skill, or lack of information because of being poorly informed on how to accomplish the task.

The goal of the task *identify the problem* is to start the programme design process with the right information, giving designers the knowledge of what exactly is the problem and some basics on what is causing the problem to be solved. It can be anything from a deficiency in knowledge to environment, or any combination of the two. A proper *identify the problem* sets the stage for the rest of the process, starting with needs assessment, by giving designers a starting point to expand into the “need”.

Complete the needs assessment.

In this stage of programme development, discrepancies between what is, and what should be, are identified. Conducting a needs assessment is critical to programme development, as it answers questions that are central to the planning and implementation of every other stage in the programme development process. Kaufman (1997, in Brown, 2003) summarizes the importance of needs assessment by stating:

Needs assessments provide the direction for useful problem resolution through identifying, documenting and selecting appropriate problems. By selecting important problems and deriving useful objectives before rushing off to resolve them, performance system professionals may improve the effectiveness and efficiency of any organization and its individual operations. Specifically, it: provides the rationale needed to move forward in planning effective interventions to solve human performance problems; ensures the plans are evidence based; enables the programme planner to sell the program plan to key stakeholders with decision making power.

Needs assessment must be carried out in a systematic manner that addresses each of the following concerns:

- 1- Setting objectives for the needs assessment.
- 2- Identifying the target group (including both the target audience of the programme, and the target audience of stakeholders and decision makers who will read the needs assessment, lend support to the programme development project, and make key decisions about programme development and implementation).
- 3- Choose an appropriate sampling procedure for gathering the data used in the needs assessment process.
- 4- Determine the data collection strategies that will be used, based on the desire to gather a wide variety of data and the need to carry out the needs assessment within budgetary and contextual constraints.
- 5- Specifying which tools and instruments will be used to gather the data, and the protocol for data collection.
- 6- Determining methods of data analysis.
- 7- Descriptions of how decisions will be made based on the data collected.
- 8- Assessing the feasibility of the developed plan.

Establish a time frame.

This stage of the programme development process is critical, as it sets out a blueprint, or roadmap, for all other activities. Setting a time line can be done in various ways, including scheduling and control charts, critical path method, and program evaluation and review technique. Time lines serve as a reference point for the completion of tasks and subtasks towards meeting the ultimate project goal or goals. They present graphical information about the expected time frame

for completion of tasks and subtasks, and help to indicate what needs to be completed, by whom, and when. Rothwell and Kazanas (1998, pp. 307-309) identify five primary benefits to establishing a time line in programme development:

- 1- Focus attention on identifying procedures, tasks, and subtasks.
- 2- Help to allocate responsibilities by identifying who is to do what, and when.
- 3- Provide a basis for controlling project time, budgeting, and determining staffing requirements.
- 4- Minimize the work needed to complete the work effectively.
- 5- Provide a basis for estimating project duration.

Establish performance objectives.

Establishing a delineated list of performance objectives is a complex but critical task that sets out an expression of what results are desired from the implementation of a program. These results are measurable, and focus on the outcomes of the program itself, not on the activities undertaken to get there. Programme developers should be able to classify and then write performance objectives based on understandings gained through task, or concept, and learner, or target audience, analyses.

According to *The Standards* (Rothwell & Kazanas, 1998), these performance objectives should:

...clarify, in measurable terms, what [members of an organization] should be able to do at the end of [the program], how well they should be able to do it, and what conditions have to exist or equipment has to be available for them to exhibit the performance (p. 159).

Rothwell and Kazanas (1998, pp. 165-166) also identify four common mistakes that should be avoided when writing delineated performance objective lists:

- 1- Avoid writing long-winded objectives.
- 2- Avoid using vague language.
- 3- Avoid descriptions of criteria linked to evaluator satisfaction.

4-Avoid lengthy lists of equipment, resources, or conditions necessary for performance (include only those not immediately obvious to a reasonable person).

Establish a budget.

Establishing and maintaining a budget is critical to the success or failure of any program development project (Rothwell & Kazanas, 1998, pp. 309-311). Budgets help program developers to set out a plan in financial terms that can be used to guide the development and implementation processes. They can also be used to help keep track of the project in financial terms, allowing program developers and key decision makers to make sure that the project stays on track.

Establishing a budget is an essential step from the perspective of key decision makers within an organization, as a detailed and accurate budget will allow them to determine whether or not the project is financially feasible given available resource levels (Rothwell & Kazanas, 1998, pp. 309-311). It will also allow them to determine whether or not the benefits of the project will outweigh the costs associated with it.

Program developers should prepare detailed budgets that include such cost estimates as salaries, equipment, and overhead. It is also a good idea to prepare budgets for each stage of smaller projects, and each task in larger projects, and to combine all of these budgets into a master budget for the entire project. Rothwell and Kazanas (1998, pp. 309-311) also identify several key considerations for program developers:

- Know the budgeting system of the organization, and the relationships between departmental, project, and organizational master budgets.
- Know the budgeting cycle of the organization.
- Know organizational forms and procedures for submitting budget estimates and requests.
- Know key deadlines for submitting budget estimates and requests.

- Know organizational requirements and procedures for monitoring and reporting project expenditures, and comparing them to original budget estimates.
- Know any governmental regulations related to the monitoring and reporting of project budgets and expenses.

Preparing a detailed project budget is not always an easy task, especially considering the often unique and temporary natures of individual projects. It is important, however, to give this stage of program development due consideration, as organizational decision makers want, and need this information. It is also important, because poor budgeting will either lead to over-projections that could send projects into the hands of a developer's competitors, or under-projecting, which could prevent developers from ever making a profit from projects (Rothwell & Kazanas, 1998, pp. 309-311).

Analyze the context.

This stage of the programme development process includes conducting an analysis of instructional objectives, including such things as job, task, and content analysis. It also includes an analysis of the target audience of the instructional design solution, and their specific contexts.

Analyzing instructional objectives is a logical next step from the completion of the needs assessment stage of programme development. Once the needs assessment has been completed, and the specific needs to be addressed have been identified, decisions can be made about what needs to address with instructional design solutions, which needs are feasible to address in such a way, and what kind of instruction would be needed to address those needs. In order to effectively make such decisions, programme developers must analyze the jobs performed by target learners, and how instructional design solutions will fit into the context of those jobs, and the tasks performed by the target learners.

Programme designers must also analyze the tasks that they will expect target learners to complete while participating in the instructional design solution, and they must make key decisions about the content to be included in such solutions. A number of different approaches have been identified as useful for determining content in programme development projects. These approaches can be used by themselves, or in combinations, and include a Philosophical Basis, Personal Introspection, Function Approach, Critical Incident, Delphi Technique, DACUM Process, and Occupational Research.

Target audience analysis involves the identification of target learners of the instructional design solution, and analysis of their characteristics, and specific contexts. Learner characteristics can include such demographic data as their range of ages, gender, geographic locations, positions within the organization, years of work experience, amount and type of educational experience, and their social and cultural backgrounds. All of these characteristics are important, as they can play a role in determining the preferred and most effective learning styles of the target audience, which will in turn be valuable information when determining instructional delivery strategies. Knowledge of these characteristics can also be useful to programme developers in helping to understand the social, cultural, and motivational contexts of the target learners—factors which will, again, influence decisions about instructional strategies. They can also be used to help appreciate the level of support that target learners may show for the programme development initiative.

Phase Two – Development and Implementation

Develop strategy (instructional or non-instructional).

Instructional strategies are the techniques, methods, sequences, media, and other means that we have of delivering the program content to the target group. To develop and specify instructional strategy is to determine the preferred media to be used, which include sections of

pre-instructional activities, presentation of information, practice and feedback, testing and follow-through activities. Non-instructional strategies address human performance problems through means other than training, education or development. These strategies require programme designers to broaden their focus to provide performance consulting and avoid restricting to instructional strategies alone. Developing and specifying instructional or non-instructional strategies can systematically move the target audience toward clearly defined programme goals and standards.

Define assessment tools.

Assessment is a critical component of the programme design process. Designers need to create assessment tools that are valid, reliable, and efficient in evaluating what the target audience can do or can know. These performance measures will enable the monitoring of the audience's progress in learning what the instruction is designed to deliver. To define the assessment tools is to define the assessment instruments that can reliably document changes in programme performance. A combination of standardized and alternative and program-developed tools can be used to assess learner progress, which includes portfolios, checklists, interviews, observations, and performance-based tests. The changes in programme design and staff development need to be ensured that current and new assessment tools are reliably used. Research and development should define the kinds of assessment that are needed to best match program requirements and measure learners' progress toward their goals and to develop improved tools to assess progress and impact.

Develop performance assessment instruments.

Programme designers should usually develop performance assessment instruments and measurements during or sequentially following the preparation of performance objectives (Rothwell

& Kazanas, 1998). In a programme design model, the assessment step is very important to the whole design process. Assessment reflects authentic, real-world applications of knowledge and understanding. It confirms gains and leads to restructuring, reorganizing, program modifications and new objectives for a given learner/customer or group of people. The development of goals, performance standards, scope, and sequences, benchmarks and assessment instruments is based on identified needs and objectives.

Performance measurements provide guidance in the preparation of instructional programs. They ensure economical choice of instructional content; they provide a basis for learner accountability; they can help link up learner achievement to organizational strategic plans (Brown, 1995). Programme designers should be capable of developing tests, questionnaires, interviews and other methods of measuring performance. They should be able to determine the purpose of the measurement and focus on appropriate methods of measuring instruction.

Implement solution.

Confirm that the participants have been diagnosed for their readiness for the solution. Make sure the solution is convenient in location and delivery to the participants. The Programme designer should also have informed all involved in delivery that readiness is important for the solution, and the convince of the participants should be a factor. There should be more than one type of delivery format in order to satisfy the needs of the participants. State the terminal performance objectives first then assess the motivation of the participants taking part in the design solution.

*Phase Three – Evaluation and Revision**Perform formative evaluation.*

Bhola (1990) notes that formative evaluation “is a method of judging the worth of a program while the program activities are forming or happening” (retrieved from internet). The feedback or results obtained through formal formative evaluation should result in modifications to the programme to ensure programme objectives are being addressed. Formative evaluation is typically an internal process conducted by the programme development team in consultation with “content experts”. Properly conducted formative evaluation will enable programme developers identify problems prior to widespread implementation to a wider audience. Rothwell and Kazanas (1998) state, “formative evaluation is conducted before instructional materials are delivered to targeted learners” (p. 264). Such corrective measures, or revisions, will help to ensure the final product addresses the objectives established at the onset of the project, which will be verified through the process of summative evaluation.

Perform summative evaluation.

The summative evaluation process is conducted at the completion of the implementation phase of the programme development model. A thorough analysis of all relevant data/results generated as a result of programme implementation is used to determine if the programme did or did not meet the programme objectives initially established. As opposed to formative evaluation, which enables programme developers to identify and correct measures during development, summative evaluation analyses the final results of the implementation. The Northwest Regional Educational Laboratory, a third-party educational evaluation and research organization, website notes that “summative evaluation is generally produced for policy makers, funding sources or potential adopters of [educational] program[s], and is aimed at delineating clearly the benefits

generated by [the] program as well as the costs and essential conditions necessary to gain those benefits” (retrieved June 10, 2003, from <http://www.nwrel.org/evaluation/overview.shtml>).

External evaluators should be allowed the opportunity to provide recommendations for programme improvement as opposed to simply completing an impact study to pass judgment on the success or failure of the programme (Gredler, 1996, p.251).

Revision.

As the evaluation portion of the model is completed one of the critical parts is the benefit of revision. From the evaluation one would gather information about what needs to be done more efficiently and what changes may need to be made in order to make the program more beneficial to the learner. Instructional materials and methods should be evaluated and revised prior to widespread use to increase their instructional effectiveness. Revision can be carried out during the program as well as at the end once the program is completed. Then determine what steps from the program need to be revisited in order to develop and implement changes as the result of the revision.

Reality check.

The key features of project planning are function, plan, schedule and control. There are many tools that one could use to help plan/monitor (reality check) a project and they are:

- preparing a time line
- budgeting projects
- monitoring the time of Instructional Designers
- tracking project accomplishments
- establishing and using methods to reallocate funds
- planning and monitoring equipment and facility requirements

Finally after the instruction has been delivered and successful the instructional designer must assess the whole process from start to finish in order to determine if the learners are informed and satisfied. As well evaluation after the completion will identify problem areas or very successful areas of the program to be considered if the program is to be implemented again.

References

- Bhola, H. S. (1990). Evaluating "Literacy for development" projects, programs and campaigns: Evaluation planning, design and implementation, and utilization of evaluation results. Hamburg, Germany: UNESCO Institute for Education DSE [German Foundation for International Development]. Xii, 306 pages. Retrieved June 10, 2003, from <http://www.sil.org/lingualinks/literacy/ReferenceMaterials/GlossaryOfLiteracyTerms/WhatIsFormativeEvaluation.htm>
- Blank, W. E. & James, W. B. (1993). Learning styles of post-secondary industrial education students. *Journal of Industrial Teacher Education*, 32(1), 84-101.
- Brown, Heather (2003). Needs Assessment in the Program Development Process. *Education 6823: Principals of Programme Design and Development, Discussion Forum*. Retrieved June 24, 2003, from http://webct.mun.ca:8900/SCRIPT/81628200203/scripts/serve_home
- Brown, K. (1995). Strategic Performance Measurements. *CPA Journal*, 1995, 65 (10), 65.
- Derry, S.J. (1988). Putting Learning Strategies to Work. *Educational Leadership*, December 1988/January 1989, pp. 118-124. Retrieved May 1, 2003, from http://elearning.uccb.ca:8080/Edu535_77/Unit01/Derry.pdf
- Gray, K. C. & Herr, E. L. (1998). *Workforce education, the basics*. Toronto: Allyn and Bacon.
- Gredler, M. E. (1996). *Program evaluation*. Englewood Cliffs, NJ: Prentice-Hall, Inc.
- MacAulay, Chris (2003). Needs Assessment Presentation. *Education 6823: Principals of Programme Design and Development, Discussion Forum*. Retrieved June 24, 2003, from http://webct.mun.ca:8900/SCRIPT/81628200203/scripts/serve_home
- Northwest Regional Educational Laboratory (2001). Program Evaluation: Who, What, When, Where, How, and Why. Retrieved June 10, 2003, from <http://www.nwrel.org/evaluation/overview.shtml>

Pratt, D.D. (1997). *Indicators of Commitment* (pp. 22-25). Retrieved May 1, 2003, from

http://elearning.uccb.ca:8080/Edu535_77/Unit01/Pratt.pdf

Rothwell, William J. & Kazanas, H.C. (1998). *Mastering the Instructional Design Process: A Systematic Approach*, Second Edition. San Francisco: Jossey-Bass / Pfeiffer.

Spoon, J. C. (1998). Aligning student learning styles with instructor teaching styles. *Journal of Industrial Teacher Education*, 35(2), 41-56.

Worthen, B. R., Sanders, J. R., & Fitzpatrick, J. L. (1997). *Program evaluation: alternative approaches and practical guidelines* (2nd ed.). White Plains, NY: Longman.