



Basics of Program Design

“Doing the Right Things”

“Management is doing things right; leadership is doing the right things.”—Peter Drucker

What is program design?

The Drucker quote at the top of this page is a good rubric for program design. When it comes to program design, “doing the right things” means asking—and rigorously answering—the right questions. Answering the questions in this article and using the attached tool will ensure that you follow the *critical path* of program design and should be assured that your program will do the right things. Doing things right is important (it’s the focus of the next article in this series), but doing the wrong things the right way is, at best, a waste of time.

A design process can build your program’s capacity through greater efficiency, effectiveness, and client satisfaction.

Why should one design a program?

This article assumes that the program you are designing does not yet exist. In a sense, even if you already have a program up and running, this could be true: if you haven’t *designed* the program, then it may just be a set of activities. Going through a formal design process can increase your efficiency, effectiveness, and client satisfaction.

Big Picture Questions

- What is program design?
- Why design a program?
- What are the steps to program design?
- What common mistakes need to be avoided?
- Where can we get more help?

A well designed program—one that is aligned with the organization’s vision, mission, values, and capacities—will make the most difference, achieve the most return on its investment, and fit organizational priorities. It’s possible that your current activities are not the best available. If so, a design process may allow you to treat these as a sunk cost: a loss of resources that you can’t recover, and can’t allow to affect your

thinking going forward. A design process, in this case, will let you make a fresh start.

What are the steps to program design?

This article uses some of these “right questions” to provide a program design framework that helps ensure that you address key program design issues while avoiding common design pitfalls. Each of the subheadings below is a step in the program design process, and each corresponds to one of the “right questions mentioned above.

Assessing need: What does our target population need?

When assessing need, familiarize yourself with both data collection tools and sources. Two tools for determining the needs of a population are the needs assessment, often called a community needs assessment, and an environmental scan. While similar, community needs assessments are typically more narrowly tailored to the conditions of a specific population (children in a school district, the elderly in a given census tract) while an environmental scan will also include political, social, cultural, economic, and other

Programs that are aligned with organizational priorities are data-driven programs. Do not assume you know what is needed, you must be able to prove—with data—that that a need exists.

factors that influence the ability of an individual or an organization to function in a given context. Each of these tools has advantages and disadvantages, and, if you have to choose between them, you will want to determine which of these is more appropriate for determining need.

Whichever tool you use, you should conduct your research with the best data sources available. The United States Census, the Bureau of Labor Statistics, The Annie E. Casey Foundation's KidsCount, local crime, housing, employment, and other statistics are available to the public. Most of these are now online. Once you have determined what need(s) to address, you can begin to design a program to address it.

Outcome development: How do we determine what we want to achieve?

Identifying needs leads to identifying desired changes in the target population: if you identify increased literacy as a need, you'll probably want to see a change in reading ability. This desired change is an outcome that your program will try to effect. You'll know when you have achieved your outcomes because you will have observed the proper indicators: the specific, measurable, observable data collected to determine whether or not an outcome has occurred. In this case, reading test scores or grades in school could each be indicators.

Evaluation language can be filled with jargon. Keep it simple:

- the ultimate aim of your program is its **goal**,
- the changes you want to see are **outcomes**, and
- proof of an outcome is an **indicator**.

Careful planning, including the development of a logic model, can help ensure you are achieving your outcomes. A logic model is a sequence of steps that your program follows. It starts with the resources you can leverage and ends with the intended outcomes (planned changes) of your programming.

You can state that a difference has been made when you can accurately, effectively, and efficiently track the changes in your target population.

You can state that a difference has been made when you can accurately, effectively, and efficiently track the changes in your target population.

Identifying resources: What do we have the capacity to do?

Once you decide on a need and a desired outcome, you must determine what resources your organization can leverage for the program (program design literature often calls these inputs). This will help you determine what your organization can actually do about the need. Because organizational capacity depends in part on the resources you can leverage, knowing your capacity is crucial to determining your outcomes

Let's assume your program wants to address the digital divide. You may want to give every child in your county a laptop and a personal technology tutor. You may, however, only have the capacity to have a portable laptop lab of twenty computers that you bring to three different after school programs once a week. As you can see, these two programs are drastically different, and the difference is driven by resource capacity.

Benchmarking: How should we address the needs we identify?

The bulk of a program's time and consists of implementing strategies. Strategies are combinations of organized, coherent activities designed to produce an outcome (as measured by indicators). If you do not have a well thought out theory about why your activities work together, then you do not have a strategy: you only have activities.

Benchmarking is identifying the most successful practices available. Once you've benchmarked them, you should determine which ones fit your vision, mission, values and needs best. Plug these into your program design and logic model.

There is a large body of literature on best practices and promising practices in many social service areas. You should conduct a benchmarking study in which you identify the best possible practices for your program. (Again, consider alignment!) The term "best practice," when used correctly, means that a practice has been proven to work in an experimental setting: this is the "gold standard." Promising practices look good but need further investigation before they are validated as "best"

practices. Under the right circumstances, your program may be a good fit for a promising practice and could help prove it is worthy of "best practice" status.

The final consideration for your activity and strategy selection activities is cost. You want to maximize your outcomes while minimizing costs because your program must be cost-effective if it is going to enjoy stable funding. You want to be very careful about what activities you consider essential because the more expensive your program gets, the less people you'll be able to impact on your organizational budget.

Logic modeling and assumptions: Why will our plan work?

Once you have identified inputs, strategies, and outcomes, you can create a logic model that explains why you believe your program will work. A logic model is a chain of actions that, together, lead to your outcomes. In the next article in this series, we develop a logic model, so we won't spend more time describing it here. The reasoning behind your logic model is often included in your assumptions.

Assumptions explain why the activities you've chosen will produce results, and are often based on results that others have obtained with those same activities. For example: if one-on-one literacy tutoring has been shown to be the most effective form of reading instruction in hundreds of different after school programs, that one-on-one tutoring, implemented correctly, should work for you. In some ways, assumptions may seem like a misnomer here, but it is a generally agreed upon term in the program design field.

Evaluation: Is our program working?

Once you have developed your program through the logic modeling phase, you can embed evaluation into it. It is absolutely essential that you build your program with evaluation in it. There are two kinds of evaluation: formative and summative. You can use your logic model to ask both formative and summative evaluation questions. Please see other tool kits in this series or the recommended resources to find guidance on this critical—yet most often overlooked—aspect of program design.

Formative evaluation helps answer the question "how are we doing?"

Summative evaluation answers the question "what did we achieve?"

Implementation: How we will do the work?

If you use your logic model effectively, you can create a program management plan that details the “who, what, when, where, how, and for how long” of your program. Once you have a solid logic model, program management plan, and evaluation plan, you should be able to visualize all the phases of implementation. All that is left is for you to actually implement it. Again, there are numerous volumes of best practices for program management to help you avoid common program implementation pitfalls.

Four common design pitfalls

A well designed program will make the most difference, achieve the most return on its investment, and fit the organizational priorities as best as possible.

Assess—don’t assume—need. It’s not what you believe, it’s what you can prove, that gets programs funded. Also, if you are wrong in your assumptions, you will provide the wrong program, wasting time, money, and opportunities to provide appropriate programming.

Include all the stakeholder groups in design. If you fail to include clients, staff, board members, or other stakeholders, you could fail to achieve program alignment, miss key implementation realities altogether, or misread data about need. Any of these mistakes could prove costly.

Benchmark your practices. One of the worst mistakes you could make is to prioritize practices over outcomes, and settle for inferior results because you like doing something a particular way—especially if those ways are less effective.

Improve continuously. Evaluate all the time, and make programmatic decisions based on data. Evaluation is a part of doing business—that’s the appropriate mindset.

Use the *Program Design Checklist* on the following page as you move forward with your project to ensure that you follow the critical path of program design. If done properly, you should be assured that your program will do the right things.

PROGRAM DESIGN CHECKLIST

Needs Assessment: How do we know what we need?

Choose methodology and tools	
Identify data sources	
Conduct interviews and other qualitative field research	
Survey other nonprofit providers in the area	
Analyze data	
Compile your findings	

Organizational Alignment: Does this program suit our priorities?

Conduct stakeholder interviews	
Build key stakeholders into the program design team	
Constantly feed decisions through the organizational vision, mission, and values	

Outcomes: How do we establish our goals?

Make choices using your findings and organizational vision, mission, and values	
Determine the capacity you have to work with; identify fiscal and human capital resources	
Explore partnerships, expertise, and other external resources you might leverage	

Benchmarking: What works?

Research best practices	
Evaluate your own practices against benchmarked practices	
Determine the appropriate practices that will produce the desired outcomes	
Build towards those appropriate practices	

Logic Modeling: What should we do?

Determine your inputs/resources, your strategies, and your outcomes	
Use measures of capacity to determine the level of service you can provide	
Be sure your assumptions clearly explain your program model	

Evaluation: Is it working?

Build both formative and summative evaluations around key questions	
Choose the most critical measures and keep the number small	
Use evaluation data constantly	

How do we conduct the work?: Implementation

Use your logic model to create a management plan	
Assess management constantly as part of your formative evaluation.	

