Theoretical Foundations and Philosophical Orientation of Relational Systems Evaluation

William M. Trochim and *Jennifer Brown Urban*

Abstract

Evolutionary Evaluation, relational developmental systems theory, and systems thinking are three broad foundational theoretical perspectives that characterize the shifting paradigm in contemporary evaluation. They are central to the development of Relational Systems Evaluation (RSE). From Evolutionary Evaluation, we focus on the variation and selection role of evaluation, as well as the notions of ontogeny, phylogeny, symbiosis, and co-evolution. From relational developmental systems theory, we highlight how mutually beneficial bidirectional relations between a program and its environmental context can promote adaptive regulations. From systems thinking, we emphasize part-whole relationships, static and dynamic processes, the related concepts of scale and boundaries, multiple perspectives, and causal pathways. We propose that these theoretical elements—all of which are operationalized through RSE—can be transformative in our thinking about the next evolutionary stage for evaluation.

Trochim, W. M., & Urban, J. B. (2021). Theoretical foundations and philosophical orientation of Relational Systems Evaluation. *New Directions for Evaluation*, 2021, 19– 30. https://doi.org/10.1002/ev.20449

In this chapter, we discuss three broad foundational theoretical perspectives—Evolutionary Evaluation, relational developmental systems theory, and systems thinking—that characterize the shifting paradigm in contemporary evaluation and that are central to the development of Relational Systems Evaluation (RSE). While each represents different traditions and histories, they are most effectively considered in concert as part of the new ways to think about what evaluation is, how it can best be accomplished, and how it might help to advance both knowledge and practice.

Evolutionary Evaluation

We begin by extending our earlier efforts (Urban, Hargraves, & Trochim, 2014) to encourage evaluation based on a foundation of evolutionary theory and the idea of natural selection (Darwin, 1859; Mayr, 2001) and especially on the idea of evolutionary epistemology (Bradie & Harms, 2006; Campbell, 1974, 1988; Cziko & Campbell, 1990; Popper, 1973, 1985). The central thrust of this line of research is that our knowledge, including knowledge of interventions and programs, evolves according to the principles of natural selection, the trial-and-error cycle of (blind) variation and selective retention. Here we discuss several important evolutionary principles and their implications for evaluation generally and RSE in particular.

The Variation and Selection Role of Evaluation

Over time, programs and their theories evolve. Programs do not last forever. They typically have a beginning and an ending. Some get selected to continue, while others get eliminated. Evaluation plays a key role in that "artificial" selection, both in encouraging and enhancing variability and in providing feedback and influencing selection. Program variations and adaptations can survive that subsequently make little apparent sense. Program features may

Trochim, W. M., & Urban, J. B. (2021). Theoretical foundations and philosophical orientation of Relational Systems Evaluation. *New Directions for Evaluation*, 2021, 19– 30. https://doi.org/10.1002/ev.20449

exist today that were adaptive in the past but are essentially residual, lasting long beyond their original adaptive genesis.

Natural selection involves several concepts, most notably the ideas of variation and selection. Popper (1985) maintains that variation in nature and in science is "at least partly random." Campbell describes natural selection using the term "blind" in his phrase "blind variation and selective retention" (BVSR) in order to convey the notion that there is no certain a priori knowledge that a variation will be adaptive. According to Campbell (1988), BVSR encompasses three essentials: "(a) mechanisms for introducing variation; (b) consistent selection processes; and (c) mechanisms for preserving and/or propagating the selected variations" (p. 402). Campbell (1988) states: "In going beyond what is already known, one cannot but go blindly. If one can go wisely, this indicates already achieved wisdom of some general sort" (p. 403). However, in the world of programs and program theory, the process of creating new programs or program variations often does not feel blind or random. We tend to think of new programs as rational, sometimes obvious, responses to problems or challenges in our environment, not as blind or random endeavors.

The notion of "blind variation" does not mean that our attempts at program construction are irrational; it simply means that when we construct programs, we do not yet know whether they will work or survive—we are "blind" to their eventual success or failure. Evaluation certainly plays a role in the program variation generation process. We conduct needs assessments (Witkin, 1995) to identify important problems or issues. We construct program theories using methods such as concept mapping (Trochim, 1989) and logic modeling (Cooksy, Gill, & Kelly, 2001; Frechtling, 2007; Kellogg Foundation, 2001). We assist planners and program developers in thinking through program implementation challenges (Bammer, 2005; King, Morris, & Fitz-

Gibbon, 1987) and in constructing approaches for formative and process evaluation. All of these can be considered activities that help to generate or describe program variation.

The other major component of natural selection is the selective retention of better-adapted variants. Evaluation certainly plays a key role in this selection process. Evaluation results are presumably used as a key determinant of whether a program continues, is reformulated, or is eliminated. But it is important to note that program selection or survival takes place within a broader social, political, and economic context. Evaluation is not the only factor that determines selection. Nor should it be. Evaluation is a major contributor, but its results need to be weighed and balanced by the many other factors that also determine what survives.

Ontogeny: The Program Evolutionary Lifecourse

The idea of ontogeny in evolutionary theory is concerned with the origin and the development of an organism from inception through maturity. Much like in thinking about the evolution of an organism, it makes sense from an evolutionary perspective to think of our programs as dynamic entities, as continuously evolving through different phases in a lifecycle. While this lifecycle will manifest itself differently for each program, much as different people develop at different rates at various times in their lives, we can sketch out a hypothetical sequence that would likely fit many programs.

The first stage in this hypothetical program lifecycle might be termed the initiation phase and spans from the initial conceptualization through design and initial piloting. The first time you implement a program—even one that was well established elsewhere—you will likely be dealing with the usual issues of initiation: identifying and training program staff, localizing the program to the immediate context, reacting to the unanticipated problems that arise, and so on. The second lifecycle stage might be labeled the growth or development phase and would

encompass the process of successive revisions of the program as it gets implemented repeatedly over time. With programs in this stage, you are getting used to the program and how it plays out in practice. You have some idea of what to anticipate. You might still be surprised by what occurs, and are still adapting the program as you learn, but you are also increasingly able to anticipate problems before they arise, and you are developing a storehouse of experience in how to deal with them. In the third stage, which we might term the maturity or stability phase, the program has been carried out at least several times with some degree of implementation success. By this point, you pretty much know what to expect. You have stable processes for implementation, and the program is routinized and often has a written protocol or process guide. The program is no longer dependent on particular individuals for implementation. If something happens to the initial implementers of the program, it can still be carried out with high fidelity. The fourth stage might be termed the translation or dissemination phase. The primary focus of this phase is on extending the program to other settings or populations of interest, pushing the ecological boundaries of the program as originally conceived into new niches or applications.

These stages are not meant to be a straight-jacket or an inflexible taxonomy. For any given program, the lifecycle progression may not be sequential. A program may be precocious. It may, for instance, quickly evolve through the development phase and become stabilized or routinized. Or, a program can revert to an earlier stage, much like the young adult that temporarily reverts to juvenile behavior before resuming more mature behaviors. At any phase, we may decide whether to continue the program or not. Sometimes it is apparent even early in a program's development that it is not able to be implemented well or that it has a fundamental flaw in its conception or structure. In the best of all worlds, probably only a small minority of programs will or should survive to the translation or dissemination phase.

Trochim, W. M., & Urban, J. B. (2021). Theoretical foundations and philosophical orientation of Relational Systems Evaluation. *New Directions for Evaluation*, 2021, 19– 30. https://doi.org/10.1002/ev.20449

This notion of a program lifecycle has practical implications for evaluation. How should a program be evaluated at each stage of its lifecycle? What role can evaluators play in helping program administrators and organizations assess where their programs are in their development and in encouraging them to think about when and how they will evolve their programs to their next phase?

In many of our program contexts, we become committed to the program as it currently exists. The program evolves up to a point, and then we get a type of "lock-in" where we seemingly get stuck in a phase and are unable to move any further. Program decisions turn into a struggle between program preservationists who fear change and the potential loss of their familiar context or even their jobs, and program critics who push for ever-extending demonstrable results and emphasize ever-shrinking funding and resources. An evolutionary perspective on programs and the idea of ontogeny emphasize program change as something to be expected and embraced. Instead of the commitment to preserving the program as it is, it encourages the idea that programs have a limited life-span and they should not be assumed to live forever. Rather, it is normal to see them as part of an ongoing trial-and-error learning process, and that the abandonment of an older program and the development of new ones is part of the normal cycle-of-life. From the beginning of the program, and throughout its evolution, the focus is on where the program is in its development and how to move it to the next phase. In effect, the idea of a lifecycle creates system pressure to move programs along and not allow them to become static.

Trochim, W. M., & Urban, J. B. (2021). Theoretical foundations and philosophical orientation of Relational Systems Evaluation. *New Directions for Evaluation*, 2021, 19– 30. https://doi.org/10.1002/ev.20449

Phylogeny: The Program Family Tree

Phylogeny refers to the macro-level evolution of a species. When we look at the phylogeny of a species, we are in effect examining its history, the story of the organism variations that survived and those that did not, and the species' family tree.

Every program can be viewed as an organism in a population of similar programs that constitutes its species. Program theories, whether stated explicitly or not, make up the essential instructions of the program. Programs have variations within each species of program. Programs have unique characteristics: the people who implement them, the activities that constitute them, the setting and assumptions that guide them, the participants who take part in them. This program variation is essential for their evolution.

Program variations are implemented, have consequences, and are selected for in subsequent program generations. Some programs and their characteristics and theories survive over time; most become extinct. Programs and program theories get selected and survive because of the fitness of their characteristics to a specific environmental or ecological niche. While most of us probably hope or believe that programs are selected for using rational criteria to yield specific desirable characteristics or outcomes, in many situations, they probably survive because people like them, get used to them, or because there are institutional, political, and economic forces that favor their survival.

There is an integral connection between ontogeny and phylogeny in evolutionary theory and in its evaluation analog. Just as with other organisms in nature, in addition to their participation in a broader species, each program has its own individual life (ontogeny), a unique life course that moves through various phases. Individual programs are born or initiated. They grow and change as they are implemented and revised. They mature and may reach a relatively

stable state, sometimes becoming routinized and standardized. And, they regenerate, die, are translated and disseminated, and so on, starting new cycles of program instances. And it is this churning of many program variations over time that leads to the selective retention of those programs with fitness or adaptability to their environment that makes up the family tree or history (phylogeny) of that program species.

Symbiosis and Co-Evolution

The ideas of symbiosis and co-evolution are critically important in evolutionary biology and should be considered equally important in program development and evaluation. One of the most familiar examples of this phenomenon is the relationship between the flower and the bee. Each provides something to the other. The flower provides nectar that is converted into honey, and the bee acts as the vehicle for sexual reproduction by moving pollen from one flower to another. Both benefit from the exchange. Neither participates in this exchange consciously. Flowers did not strategize one day that they needed bees as a vehicle for reproduction. And bees did not decide that flowers would be good vessels for honey production. They co-evolved over millennia in a manner that makes them co-dependent.

There are several ways that symbiosis and co-evolution are important for evaluation. First, if all programs evolve through different stages over time, then we must recognize that the evaluation approaches we use at each stage need to differ throughout the life of the program. That is, the way we would evaluate a program during its initiation stage would not likely be appropriate for evaluating it during its growth stage, and so on. In effect, the evaluation of a program has a lifecycle, separate from the lifecycle of the program itself, and one of the major tasks of Evolutionary Evaluation is to encourage the symbiotic or co-evolutionary relationship

between program and evaluation lifecycles. In the initiation phase, an evaluation needs to be dynamic and flexible, providing rapid feedback about implementation and process.

In many program evaluations, this is accomplished with simple monitoring or post-only feedback forms, unstructured observation, qualitative methods, informal debriefing, and feedback and communications systems. In the development phase of an evaluation, the focus tends to shift to the observation and assessment of change. We focus on things like designing observational procedures and measures of key outcomes, assessing the consistency and construct validity of measures, looking at pre-post differences and examining the relationships among different observations using qualitative and/or quantitative methods. The mature phase of an evaluation tends to emphasize the idea of control. At this point, the program is routinized and stable enough to compare the performance of participants with some standard expectation of performance or with outcomes of people who participate in alternative programs or none at all. This is the realm of experimental and quasi-experimental designs and more structured and comparative qualitative approaches. The translation or dissemination phase in evaluation is typically concerned with adaptation, implementation, sustainability, and generalizability or external validity. It examines the consistency of outcomes across different settings, populations, or program variations. This is the realm of secondary and meta-analysis and of program review approaches that seek general inferences about the transferability and adaptability of the program. Encouraging a symbiotic relationship between the evaluation approach and the program lifecycle is a critically important Evolutionary Evaluation process.

Second, the ideas of symbiosis and co-evolution also have important practical implications for the level of support people have for evaluation. In many evaluation contexts, one hears a series of laments about how unmotivated people are to evaluate or their resistance to

doing evaluation. For instance, the evaluator asks, “Why don’t these program people just cooperate when I ask them for data?” Program practitioners ask, “Why don’t these evaluations address something that would be useful for us?” Program participants want to know, “Why do they keep bugging us for data? We don’t get anything from this.” Ideally, we would want the situation to be co-evolutionary where program participants are providing information naturally as part of their participation, where program administrators are getting what they want from the provided data, and where evaluation happens almost transparently as an integrated aspect of program implementation. The ideal is the flower and the bee. This is a difficult ideal to achieve in practice. It requires that evaluation systems be engineered in such a way that each stakeholder group’s incentive to participate in the evaluation is well understood and integrated into the approach.

Relational Developmental Systems Theory

Relational Developmental Systems (RDS) meta-theory, is a leading theory of developmental science and can rightfully be considered part of a broader systems perspective that holds that development is the result of bidirectional individual context coactions across multiple levels (i.e., from cells to society; e.g., Gottlieb, Wahlsten, & Lickliter, 2006; Lerner, 2006; Overton, 2006, 2010). When these coactions are mutually beneficial, they are referred to as adaptive developmental regulations (Brändtstadter, 1999). It is important to note that development is not viewed as unidirectional, but rather as a bidirectional process whereby an organism shapes and is shaped by the environment. RDS rejects a split-reductionist view of development that sees genes as the primary engine of development. Organisms are viewed as autopoietic, active agents that are self-creating, self-organizing, and self-regulating, which, in turn, produces greater complexity and the potential for plasticity (Lerner, 2006; Overton, 2013).

Relative plasticity highlights the potential for adaptive development across the life span within certain constraints.

Although RDS meta-theory describes human development, many of the same concepts can be applied to program development. Program theory is akin to DNA; but, just as human development is not the result of genes alone, program theory (program DNA) alone does not account for what actually happens when a program is implemented. The interaction of program theory (what was intended) with the context in which the program is situated produces growth and change in the program. When those changes are positive, this process is akin to adaptive regulations.

Therefore, it should come as no surprise that practitioners often make changes to evidence-based programs in order to increase the chances of a mutually beneficial relationship between the program and the local environment. It should also come as no surprise that a program will likely evolve and change due to the influences of the context in which it is situated. The question for evaluation is whether these adaptations are indeed adaptive. Therefore, the question we really should be asking is what fundamental attributes of which programs (e.g., what programmatic features); addressing the needs of individuals of what status attributes (e.g., people at what portion of the life span, and of what sex, race, ethnic, religious, or social location); in relation to what characteristics of the context; are likely to be associated with what facets of adaptive functioning (adapted from the Bornstein Specificity Principle; Bornstein, 2006)?

Evaluation is inherently a part of Popper's adaptive behavioral learning. It can be considered one of a class of processes whereby we attempt to learn about the world around us. It is a mechanism for creating the conditions for observing the world and providing feedback about it and about the observation itself. Evaluation is a form of feedback, and as such, is part of the

selection process. In the view proposed here, programs and their theories are selected over time because they have characteristics that promote adaptive regulations (there is a mutually beneficial relationship between the program and its environmental context).

System Thinking and Evaluation

One of the most striking characteristics of the evolution of evaluation in the past several decades is the increasing degree to which evaluation encompasses the idea of "systems" (Gates, 2016; Hargreaves & Podems, 2012; Williams & Hummelbrunner, 2010). In this section, we characterize how systems thinking relates to evaluation by presenting a number of metaphors that each describe a particular aspect of systems thinking. For each, we provide a short name for the metaphor and indicate parenthetically the systems thinking principle it is meant to represent.

Greater Than the Sum (Part-Whole Relationships)

Systems are by their very nature collections of multiple things, so it is not surprising that one of the most fundamental distinctions in systems theory is that of "part" and "whole." Part-whole relationships are everywhere in evaluation. For instance, an organization (whole) will often operate in multiple program areas (e.g., an educational or outreach organization might have programs for children, teens, adults and the elderly; or programs in health, education, environment, science, etc.); each program area (whole) might have multiple programs (parts); each program (whole) will usually consist of multiple activities (parts); each activity (whole) can typically be broken down into different tasks (parts); and so on. Or we can view different levels of part-whole hierarchies in terms of stakeholder groups. In an educational or outreach program we might think of stakeholders at the program level (participants, their families, and program deliverers), the organizational level (program managers and organizational administrators), the local context (local officials or the local public), the funders level (e.g., private, state or national)

and even the societal level (Congress or society as a whole). The idea of part-whole relationships is essential in the development and implementation of programs. It is central to our description of the program, the development of program models, and the analysis of stakeholders and their interests.

The Rock and the Bird (Static and Dynamic Processes)

There is a parable in systems thinking that illustrates well the difference between static and dynamic processes. If you throw a rock into the air, you can predict with some accuracy where it will go. The harder you throw, the farther it will generally go. The higher you aim, the higher its trajectory. And, if we eliminate the variability of the human thrower and use mechanical devices like a catapult, we can predict even more accurately where the stone will go. A rock is a static object, one that cannot direct itself. On the other hand, if you throw a bird (gently, please!), there is virtually no way to predict which way the bird will go and where it will land. The bird can sense its surroundings and may head off in any direction. A bird is a living, dynamic system that gathers and processes input and interacts with its environment.

This distinction between static and dynamic processes is important in systems theory and in evaluations that are done from a systems perspective. Since programs involve people and organizations they are inherently dynamic, it is difficult to predict where they will go and what will happen. As programs unfold, the directions they take are influenced by the surroundings and by the interactions of the participants. In this sense, programs are more like birds than like rocks. On the other hand, the idea of a "program" suggests that we are trying to do something systematically, that we are attempting to follow a pre-determined set of steps in order to achieve some predictable result. In this sense, programs are more static; they are more like the stone in the parable.

So, which is it? Are programs static or dynamic? Should our evaluations be constructed for one or the other? The short answer is: both are important. Both the rock and the bird can be understood from a systems perspective. Both are parts of a larger whole. Both have relationships with the other parts. Over time, programs are likely to evolve through different phases, some more static and others more dynamic. For instance, when a program is first being developed and piloted, it is likely to be more dynamic and unpredictable. That dynamism is essential for learning and adaptation, for enhancing the focus and quality of the endeavor. Over time, many programs tend to become more static. They become routinized and develop standard operating procedures that can be implemented consistently. They can be transferred to new contexts and settings with some degree of predictability. This standardization is also essential. Over even a longer period of time, the program may become too static or rigid, or it may lead to insights that suggest even better variations that might be tried. In either case, we might be motivated to begin other cycles of dynamic-static program development and evolution. We need to recognize that both static and dynamic systems have their place in evaluation and be able to identify how evaluation approaches need to evolve both to encourage program evolution and to provide feedback and learning about it.

The 'Local' and the 'Global' (Scale)

We can look at any system from many different viewpoints. For instance, if we are looking at an organization with multiple programs, each program can be viewed as a "part" in the system that constitutes the organization. When we talk about the relationship between a program and its organization, we can think of the program as "local" and the organization as "global" in relation to each other because they are at different but related levels in the hierarchy. On the other hand, when we compare or contrast two programs within an organization, we can

think of that as a "local-local" relationship because both are at the same level of scale in the hierarchy. If we shift our perspective to a higher level of scale, we are also shifting what we consider "local" and "global." For instance, if we think about an organization as one part in a larger system of similar organizations (e.g., a county office in a state-wide system of such offices), then the organization becomes "local" to the system's "global." When we compare two county-level offices, we are looking at a "local-local" relationship. When we look at the county level office in relation to the state office, we have a "local-global" relationship because we are looking across different levels of scale.

Why are the ideas of scale and of local and global relationships important in evaluation? Different parts of a system do not exist in isolation. If we do not take them into consideration throughout our evaluation efforts, we can run into significant problems that can jeopardize the whole endeavor. For instance, very often, something in one part of a system may be in conflict with something in another part of the system. A program activity may conflict or compete with the activity of another program (a local-local relationship in a system) or with an organizational policy or effort (a local-global relationship). Or the expectations that stakeholders at one level of scale have for an evaluation may be very different than those of stakeholders at a different level. Funders (generally more global) may expect that the evaluation will focus on accountability and impact while program implementers (typically operating at the local scale) may be more interested in how evaluation can contribute more immediate feedback that can be used to improve programs.

The Eye of the Beholder (Multiple Perspectives)

A system can be viewed from many different perspectives. Almost everyone is familiar with the famous drawing from the psychology of perception that shows either faces or a vase

depending on how you look at it. When you stare at this picture, you can experience the shift in perspective that psychologists have described as the "figure-ground" effect. The same system can seem very different when looked at from different viewpoints.

The issue of multiple perspectives is essential in evaluation for several reasons (depending on your perspective). For instance, all program evaluations involve a multiplicity of stakeholders, including the participants, program developers, administrators, support staff, families and community, funders, policymakers, politicians, and the general public. One of the most important things an evaluator can do is to help different stakeholders see the system of a program from the perspectives of other stakeholders. For instance, program deliverers may not perceive why they are being pressured to evaluate their programs "from an outside perspective" or why they need to demonstrate outcomes and impacts. If they understand the system pressures on different stakeholders, in this case, the funders, they may gain a greater appreciation of how their view fits into the larger system.

Conversely, funders may not understand why the organizations they fund are resisting their calls for evaluation. If they can begin to view the program through the eyes of those who deliver it or participate in it, they are likely to understand their pressures better. In this example, it is easy to see that the issue of perspective is intimately related to the motivations and incentives of different stakeholders. The field of evaluation has long emphasized the values of participatory evaluation approaches, in part because of the critical importance of multiple perspectives.

But multiple perspectives are also critical for understanding the content and meaning of programs. Throughout an evaluation, it is valuable to have key stakeholders look at different parts of the program, to share their views, and to consider how others might perceive them. For

instance, it is surprising how many times, even in simple programs, different people will have remarkably different views of what they are trying to do or what they think the program is affecting. We find that when people share their perspectives, they can uncover such differences and that this learning is critical for informing the evaluation.

Inside – Outside (Boundaries)

All systems have boundaries that distinguish the system from what is outside it. Developing an understanding of boundary issues is an important part of systems evaluation. For instance, Schwandt (2018) draws from critical systems heuristics to explore the collaborative social practice of boundary setting in evaluation. There are no simple answers when making boundary judgements and often reasonable people involved in the same program will disagree. In some sense, boundary discussions require that stakeholders negotiate a consensus about what they mean by their "program." For instance, in a teacher-training program, is the program just the set of activities used to train teachers, or does it also include the activities that the teachers subsequently do in teaching their students? Discussions about program boundaries often become important learning events for stakeholders because they lead to discussions about the meaning of what they are doing with their programs and the evaluations of them.

And So On, and So On… (Causal Pathways)

The idea of cause and effect is central to systems thinking. The field of System Dynamics, for instance, develops cause-effect models or "causal" chains and uses them to think about the way causes produce effects throughout the system and the resulting types of feedback loops that can often lead to unanticipated outcomes. In effect, you are describing the chain of thinking in a system: "X leads to Y, which leads to Z, and so on, and so on…"

The notion of causality is critically important in RSE. It is central to theory of change approaches to evaluation, path analysis, theory-driven approaches, and much more. In program logic models, there is a general idea of causality—activities are expected to lead to outputs which are, in turn, expected to produce short-, medium-, and long-term outcomes and ultimately impacts. However, one problem with traditional logic models is that they are "columnar" in nature. The entire set of program activities, or outputs, or each phase of outcomes are typically treated as a whole. That is, in traditional logic modeling, while we expect that program activities produce outputs, we do not specify *which* activities are expected to produce *which* outcomes. In other words, traditional logic models do not spell out the specific cause-effect relationships that are expected.

Because systems thinking suggests that distinguishing different cause-effect chains can be important, we prefer program logic models that describe the specific causal pathways involved in programs. For example, typical programs usually involve multiple activities, outputs, and outcomes. In a pathway approach, you would specify each connection that you think might be relevant. You might specify that Activity A effects short-term Outcomes A and C, which in turn affect mid-term outcomes E and F, and long-term Outcomes A and D. You might also expect that there will be feedback loops in your model. For instance, changing the results of an outcome could trigger a change in a program activity that would then influence other outcomes.

This kind of causal pathway model is useful in telling the story of the program and is essential in developing a high-quality evaluation of it. A program model is likely to have many pathways from activities to outcomes. Drawing pictures of the pathway model enables you to better understand how you think your program should operate. It is especially useful for tracing the "throughline" of your program, the major causal paths through the model of your program.

The throughline points out the key program activities that may lead to any outcomes and helps you to identify key outcomes that should be measured.

Concluding Thoughts

The ideas presented above constitute a rich and varied base for thinking creatively about what evaluation is and how it can best be pursued. Many features of evaluation existed before the thinking in this chapter was advanced, and many evaluation traditions will continue unchanged because of it. But at the least, the ideas offered here provide a fresh perspective on why traditional evaluation features may have evolved, and at best, these ideas can be transformative in our thinking about the next evolutionary stage for evaluation. How these ideas are operationalized through the practice of RSE is addressed in greater detail in subsequent chapters.

Trochim, W. M., & Urban, J. B. (2021). Theoretical foundations and philosophical orientation of Relational Systems Evaluation. *New Directions for Evaluation*, 2021, 19– 30. https://doi.org/10.1002/ev.20449

References

Bammer, G. (2005). Integration and implementation sciences: Building a new specialization. *Ecology and Society, 10*(2).

Bornstein, M. H. (2006). Parenting science and practice. In K. A. Renninger, I. E. Sigel, & W. Damon (Eds.), *Handbook of Child Psychology: Child psychology in practice.* (6th ed., Vol. 4, pp. 893–949). Hoboken, NJ: Wiley.

Bradie, M., & Harms, W. (2006). Evolutionary Epistemology *The Stanford Encyclopedia of Philosophy* (Fall 2006 Edition ed.).

Brändtstadter, J. (1999). The self in action and development: cultural, biosocial, and ontogenetic bases of intentional self-development. In J. Brändtstadter & R. M. Lerner (Eds.), *Action and self-development: Theory and research through the live span* (pp. 37-65). Thousand Oaks, CA: Sage.

Campbell, D. T. (1974). Evolutionary Epistemology. In P. A. Schilpp (Ed.), *The Philosophy of Karl Popper*. LaSalle, IL: Open Court Publishing Co,

Campbell, D. T. (1988). Evolutionary Epistemology. In E. S. Overman (Ed.), *Methodology and Epistemology for Social Science: Selected Papers of Donald T. Campbell*. Chicago: University of Chicago Press.

Cooksy, L. J., Gill, P., & Kelly, P. A. (2001). The program logic model as an integrative framework for a multimethod evaluation. Evaluation And Program Planning, 24(2), 119-128. doi:10.1016/s0149-7189(01)00003-9

Cziko, G. A., & Campbell, D. T. (1990). Comprehensive Evolutionary Epistemology Bibliography. *Journal Of Social And Biological Structures, 13*(1), 41-82.

Trochim, W. M., & Urban, J. B. (2021). Theoretical foundations and philosophical orientation of Relational Systems Evaluation. *New Directions for Evaluation*, 2021, 19– 30. https://doi.org/10.1002/ev.20449

Darwin, C. (1859). *On the Origin of Species by means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life*. London: John Murray.

Frechtling, J. A. (2007). *Logic Modeling Methods in Program Evaluation*. San Francisco, CA: Jossey-Bass.

Gates, E. F. (2016). Making sense of the emerging conversation in evaluation about systems thinking and complexity science. *Evaluation and Program Planning*, *59*, 62-73.

Gottlieb, G., Wahlsten, D., & Lickliter, R. (2006). The significance of biology for human development: A developmental psychobiological systems view. In W. Damon & R. M. Lerner (Eds.), *Handbook of child psychology* (Vol. 1, pp. 210-257). Hoboken, NJ: Wiley.

Hargreaves, M. B., & Podems, D. (2012). Advancing systems thinking in evaluation: A review of four publications. *American Journal of Evaluation*, *33*(3), 462-470.

Kellogg Foundation. (2001). *Logic Model Development Guide: Using Logic Models to bring together Planning, Evaluation and Action*. Battle Creek, Michigan: W.K. Kellogg Foundation.

King, J. A., Morris, L. L., & Fitz-Gibbon, C. T. (1987). *How to Assess Program Implementation*.

Lerner, R. M. (2006). Developmental science, developmental systems, and contemporary theories of human development. In R. M. Lerner & W. Damon (Eds.), *Handbook of Child Psychology: Theoretical models of human development* (6th ed., Vol. 1, pp. 1-17). Hoboken, NJ: John Wiley & Sons.

Mayr, E. (2001). *What Evolution Is*. New York, NY: Basic Books.

Overton, W. F. (2006). Developmental psychology: Philosophy, concepts, methodology. In R. M. Lerner (Ed.), *Theoretical models of human development* (6th ed., Vol. 1, pp. 18-88). Hoboken, NJ: Wiley.

Trochim, W. M., & Urban, J. B. (2021). Theoretical foundations and philosophical orientation of Relational Systems Evaluation. *New Directions for Evaluation*, 2021, 19– 30. https://doi.org/10.1002/ev.20449

Overton, W. F. (2010). Life-span development: Concepts and issues. In W. F. Overton & R. M. Lerner (Eds.), *Handbook of life-span development: Cognition, biology, and methods* (Vol. 1 pp. 1-29). Hoboken, NJ: Wiley.

Overton, W. F. (2013). Relationism and Relational Developmental Systems: A Paradigm for Developmental Science in the Post-Cartesian Era. In R. M. Lerner & J. B. Benson (Eds.), *Advances in Child Development and Behavior* (Vol. 44, pp. 21-64).

Popper, K. (1973). *Evolutionary Epistemology*. Paper presented at the Sections I-VI of "The Rationality of Scientific Revolutions" given at the Herbert Spencer Lecture at the University of Oxford, Oxford.

Popper, K. (1985). Evolutionary Epistemology. In D. M. Miller (Ed.), *Popper selections* (pp. 78-86). Princeton, NJ: Princeton University Press.

Schwandt, T. A. (2018). Evaluative thinking as a collaborative social practice: The case of boundary judgment making. In A. T. Vo & T. Archibald (Eds.), *Evaluative Thinking. New Directions for Evaluation. 158*, 125–137.

Trochim, W. (1989). An Introduction to Concept Mapping for Planning and Evaluation. *Evaluation and Program Planning, 12*(1), 1-16.

Urban, J. B., Hargraves, M., & Trochim, W. (2014). Evolutionary Evaluation: Implications for evaluators, researchers, practitioners, funders and the evidence-based program mandate. *Evaluation and Program Planning, 45*, 127-139. doi:10.1016/j.evalprogplan.2014.03.011

Williams, B., & Hummelbrunner, R. (2010). *Systems concepts in action: a practitioner's toolkit*. Stanford University Press.

Witkin, B. R. A., J.W. (1995). Planning and Conducting Needs Assessments: A Practical Guide. Thousand Oaks, CA: Sage.

Trochim, W. M., & Urban, J. B. (2021). Theoretical foundations and philosophical orientation of Relational Systems Evaluation. *New Directions for Evaluation*, 2021, 19– 30. https://doi.org/10.1002/ev.20449

WILLIAM TROCHIM is Professor of Policy Analysis and Management at Cornell University who specializes in research methodology and program evaluation.

JENNIFER BROWN URBAN is professor of Family Science & Human Development, and co-Director of the Institute for Research on Youth Thriving and Evaluation, both at Montclair State University. Her research focuses on positive youth development and evaluation capacity building in youth-serving organizations.

Trochim, W. M., & Urban, J. B. (2021). Theoretical foundations and philosophical orientation of Relational Systems Evaluation. *New Directions for Evaluation*, 2021, 19– 30. https://doi.org/10.1002/ev.20449