A Foundation for Empirical Software Engineering

Dewayne E Perry
Empirical Software Engineering Lab
ECE, The University of Texas at Austin
Austin TX 78712
1.512.471.2050
perry@ece.utexas.edu

ABSTRACT
The goal of this paper is to lay a foundation for rigorous empirical software engineering. I do this by introducing several theories and their models. I first present an abstract theory *TM* about theories and models and related processes. I then apply *TM* to itself, yielding *MTM*, a theory and model about *TM*. These two theories provide a unified approach to design disciplines. I then provide 4 models of them relevant to empirical software engineering: products, developments, instruments and experiments. I introduce theory *E* and apply *E* to *TM* and *MTM* yielding *ETM* and *EMTM*. These latter two theories provide a taxonomy of empirical studies for design disciplines. I discuss briefly models of such empirical studies. I then apply *E* to itself yielding *EE* that provides a taxonomy of evaluations of empirical studies. Finally, I present a list of challenges for empirical software engineering research.

Categories and Subject Descriptors
D.2 [**Software Engineering**]: Empirical Software Engineering

General Terms
Software Engineering Theories and Models; Empirical Theories and Models, Empirical Taxonomies.

Keywords
Empirical Software Engineering, Theories and Models, Empirical Theories and Models, Software Engineering Theory and Models, Classes of Empirical Studies, Empirical Taxonomy.

1. INTRODUCTION
The motivation for this research is to establish the same rigorous empirical foundations for software engineering that we find in natural and behavioral sciences. In natural sciences, their rigorous basis rests on 1) theories that have to be testable, 2) testing done in the physical world that 3) provides hard constraints on the theories. In behavioral sciences, their rigorous basis rest on 1) theories that have to be testable, 2) testing done in the behavioral world that 3) provides probabilistic constraints.

Currently, we do not have this same rigor in the sciences of the artificial [27]. Indeed, we are woefully inadequate with respect to empirical studies. Granted, as a field we are improving, but we are a long way from achieving the rigor we find in both natural and behavioral sciences. It is certainly easy to see why: in natural sciences education, students are subjected to a stream of experimental work in the laboratory components of their basic courses; in behavioral sciences, students are subjected to experimental design and experimental statistics courses as both undergraduates and graduates.

1.1 Experimental Science
Let us first take a basic look at science, even though one might argue that it is not necessary since everyone understands it thoroughly. My reason for doing this is to set the stage for the theories and models relevant for empirical software engineering.

Science is basically an iterative process consisting of the following steps (see Figure 1):

- Observations and abstractions are use to create a theory T.
- We test theory T against reality W with an experiment E using one or more instruments I.
- We then reconcile theory with reality.
- When predictions don't agree with reality, we change the theory.

Gooding et al. [11] argue for the critical importance of the instruments we use in experimental work. They are the lens through which we observe the world. To paraphrase Wittgenstein [31], *the limits of my instruments are the limits of my world*. They enhance, limit, and color our view of the world. In natural sciences, instruments are often physical creations; in behavioral sciences they are often intellectual creations. Humans are common instruments in both. Instruments may be active or passive. They may be theory-laden or transparent and neutral. They may be reliable and standardized or not. In any case, they are a critical part of the empirical apparatus and as such will play a critical part in any scientific endeavor.

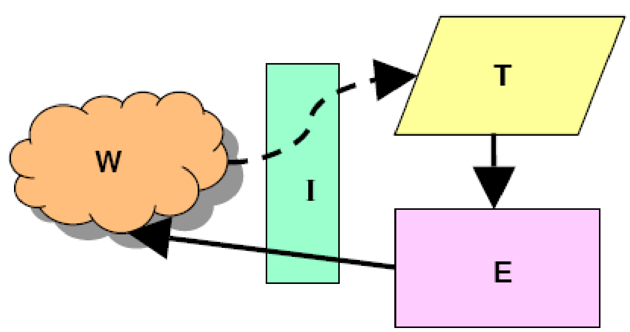

Figure 1: Basic Science

ESEC/FSE '07, September 3-7, 2007, Dubrovnik, Croatia.

1.2 Natural, Behavioral & Artificial Sciences

In remedying our lack of rigor, it is critical to understand how the sciences of the artificial differ from behavioral and natural sciences. Obviously, we must have theories that are testable just as they do. The differences come in the context of testability and the constraints faced. The sciences of the artificial have some aspects in common with natural and behavioral sciences: testing is done in both physical and behavioral contexts. However, testing is also done in intellectual and technological worlds as well. For the physical and behavioral contexts we have the same hard and probabilistic constraints. For the technological context, we have selectable constraints. For the intellectual context, we have malleable constraints – i.e., we have constraints that we can change, perhaps even arbitrarily.

There are interesting differences between natural and behavioral sciences that are relevant to design disciplines. The general goals of natural sciences are to understand natural phenomena and create a theoretical basis for prediction. Further, natural sciences provide a basis for invention and engineering. The general goals of behavioral sciences are to understand human and societal phenomena and provide a theoretical basis for prediction *and interventions*. The later are important because of the need to change the world, not merely to observe it. Software engineering has this latter property as well.

1.3 Theories and Models

The terms "theory" and "model" are used and misused in a variety of ways, often informally and interchangeably. I want to use them in a very specific way: a theory (a more or less abstract entity) is reified, represented, satisfied, etc by a model (a concrete entity).

Scientific theories are based on *observations of* the world. They change on the basis of new observations, or new interpretations of observations. Legal theory is quite different: it is based on *decisions about* the world and is changed on the basis of new decisions or new interpretations of decisions.

Theories in design disciplines are a combination of both of the above. They are based on both observations of and decisions about the world. They change on the basis of new observations or decisions, or on the basis of new interpretations of those observations or decisions.

This view of theories is derived in part from Turski and Maibaum [28] where they state *"A specification is rather like a natural science theory of the application domain, but seen as a theory of the corresponding program it enjoys an unmatched status: it is truly a postulative theory, the program is nothing more than an exact embodiment of the specification"*. I note, however, that I want a theory in TM to be broader than a specification and, more than likely, less formal.

We often use models as a representation of a theory. In natural sciences, the model is often a set of mathematical formulas. In logic, a model is an interpretation of a theory and has certain logical properties. Here again, I want to broaden the notion of a model to be a representation (indeed, a reification) of the theory. The model is of paramount importance in design disciplines as it is the visible manifestation of the theory. And, of course, a theory can have an arbitrary number of models.

1.4 Theory/Model Roadmap

In the sprit of the underlying idea of this paper, I will apply the described approach to the presentation here. In section 2, I present my theory of theories and models, TM. I then apply TM to itself to create the meta-theory MTM in section 3. I then present four models of TM and MTM and discuss one of the models in depth. In section 5, I introduce a theory of an empirical theory and model E and in sections 6 and 7 apply E to TM and MTM yielding empirical theories ETM and EMTM. I then apply E to itself, yielding EE representing a theory of empirical evaluations of empirical evaluations themselves. Analogous to theories TM and MTM, I present four models of ETM and EMTM and, again, discuss one of them, but only briefly. I conclude in section 10 with challenges and conclusions.

2. THEORY & MODEL *TM*

I first present the full theory of TM and its model, and then simplify it to pare it down to its essential elements.

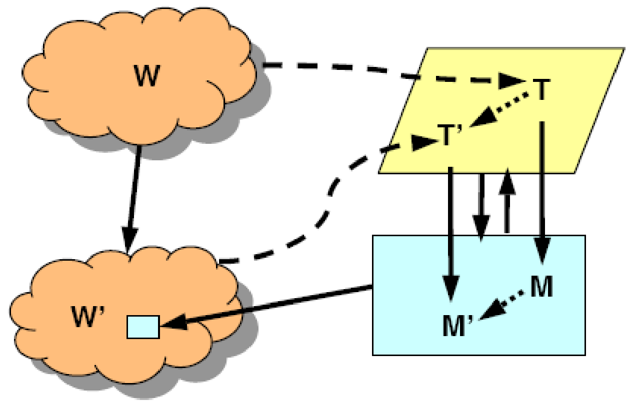

Figure 2: Basic Theory & Model *TM*

2.1 Theory of *TM*

Theory TM is meant to capture the typical cycle of creating a theory that is then reified into a model where the model is then injected into the world and changes the world (see Figure 2). I summarize it as follows:

- We observe and abstract some specific part of the world and create a theory
- From that theory we create a usable model to reify or represent that theory
- We iteratively adjust both the theory and the model as our understanding of the theory and its model evolves, both iteratively and interactively
- When satisfied that the model adequately represents the theory, we inject the model into the world
- Injecting the model into the world changes the world
- The changes brought about by this injection as well as other changes often lead to adjustments and extensions to the original theory
- Changes to the theory in turn lead to further changes in the model and the world

This abstract theory is then reified into a concrete model as described below.

2.2 Model of *TM*

The model of TM consists three elements and eight transformations (or, if you will, processes). The elements are as follows:

- **W** – The world, but more specifically, the part of the world relevant to the theory
- **T** – The theory initiated by observations and abstractions
- **M** – A model that reifies, represents or satisfies the theory T

The transformations involving these elements of the model are as follows:

1. **W → T –** Generate a theory: observe and abstract from the world (W) to create a theory (T)
2. **T → M –** From the theory (T) create a model (M)
3. **T → T** – Evolve theory T until satisfied
4. **M → M** – Evolve the model until satisfied
5. **T ←→ M** – We frequently adjust the theory T and model M to each other, i.e., we change the model to better represent the theory and sometimes change the theory to better conform to the model
6. **M → W** – Inject model M into the world W thereby changing it to W'
7. **W → T** – the changed world W' induces changes in theory T yielding T'

It should be clear that this model represents the theory T above.

On the basis that we are basically iterative and evolutionary in our endeavor, we can simplify the model (see figure 3) as follows: 1) fold 1 and 7 above together; and 2) simplify T←→ M into M→T as we already have T→M in 2. We will use this simplified model in the sequel and in MTM discuss only its simplified model.

3. APPLYING TM TO ITSELF: *MTM*

TM provides a basic theory and model of design discipline artifacts. We then apply TM to itself to yield a meta-theory and meta-model *MTM*. The intuition for this comes from Osterweil's ICSE9 paper [18] where his seminal insight is that a software process system goes through the same life-cycle as a software product system. MTM is the design theory and model of how to produce a design theory and model.

3.1 Theory of *MTM*

Theory MTM is meant to capture the typical cycle of creating a theory of TM (i.e., a theory of producing a design product) that is then reified into a model of TM and the model is injected into the world and changes the world (see Figure 3). I summarize it as follows:

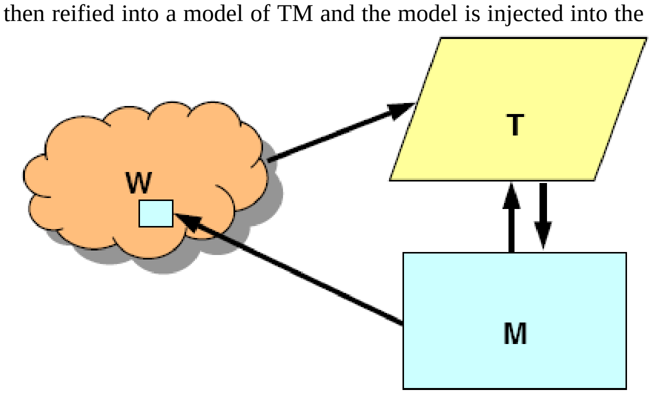

Figure 3: Theory & Model *TM* Simplified

- We observe and abstract some specific part of the world and create a theory of
  - What the world of TM is like
  - What form a theory in TM should take
  - What form a model of TM should take
  - What form the processes of creating the theory and its model of TM should take
  - How the resulting model of TM should be injected into the world
- From that theory we create a usable model to reify or represent that theory of
  - What the world of TM is like
  - What form the theory of TM should take
  - What form the model of TM should take
  - What form the processes of creating the theory and model TM should take
  - How the resulting model of TM should be injected into the world.
- We iteratively adjust both the theory and the model as our understanding of the theory and its model evolves, both iteratively and interactively
- When satisfied that the model adequately represents the theory we inject the model into the world
- Injecting the model into the world changes the world
- The changes brought about by this injection as well as other changes often lead to adjustments and extensions to the original theory
- Changes to the theory in turn lead to further changes in the model and the world

3.2 Model of *MTM*

From this theory, we then create a usable model of MTM. The model is quite similar to that of TM's model, except as noted in the sub-bullets listed in the preceding section.

The elements are as follows:

- **MW** – The world, but more specifically, the part of the world relevant to TM
- **MT** – The theory (initiated by observations and abstractions) of TM and its elements and processes
  - *T(W), T(T), and T(M)* – Theories about the world, theory and model of TM
  - *T(W→ T), T(T → M), T(M → T), T(T → T), T(M → M), T(M → W)* – Theories about the processes of creating and evolving TM and injecting it into the world.
- **MM** – A model that reifies, represents or satisfies the theory MT
  - *M(W), M(T), and M(M)* – Models of the world, theory and model of TM
  - *M(W→ T), M(T → M), M(M → T), M(T → T), M(M → M), M(M → W)* – Models of the processes of creating and evolving TM and injecting it into the world.

The transformations involving these elements of the model are as follows:

- **MW → MT –** Generate a theory: observe and abstract from the world (MW) to create theory (MT); the changed world MW induces changes in theory MT yielding a new MT
- **MT → MM –** From the theory (MT) create a model (MM)
- **MT → MT** – Evolve theory MT until satisfied
- **MM → MM** – Evolve the model MM until satisfied
- **MM → MT** – we frequently adjust the theory MT and model MM to each other, i.e., we change the theory to conform to the model
- **MM → MW** – Inject model MM into the world W thereby changing it to MW

4. MODELS OF *TM* & *MTM*

Having presented theories TM and MTM, I now talk about some models (interpretations, if you will) of those two theories: systems (TMS and MTMS), developments (TMD and MTMD), empirical instruments (TMI and MTMI) and empirical studies (TME and MTME). I will explore the systems model in depth and sketch the remaining models

4.1 Systems – TMS and MTMS

I claim that a software system at a suitable level of abstraction is a model of TM and that a significant part (some may argue even the whole) of software engineering itself is a model of MTM (hence we call these models TMS and MTMS).

4.1.1 Systems – TMS

TMS.W (i.e., W in theory TM as interpreted in model TMS) contains what Jackson [12] calls the *problem space*. It is that part of the world that represents the problem that we want to address with our software system. We observe and abstract from this problem space to create a theory TMS.T (i.e., theory T in TM) of the problem we want to solve. We refer to TMS.T as requirements. TMS.W also contains what Jackson calls the *solution space*. It is in this space that we find the elements that we put together to create the model TMS.M (the software system itself) that reifies and represents those requirements TMS.T.

TMS.W→T is the process of deriving the requirements from the chosen problem space by observing and abstracting what is considered to be critical and central to the problem to be solved. It is also the process of understanding the effects of a changing world on the requirements that exist as the basis for an existing system TMS.M. TMS.T→M is the process of creating and evolving the model/system from the theory/requirements, while TMS.M→T is concerned about adjusting the theory/requirements to better conform to an existing model/system. This latter happens regularly as we find that some requirements may be too costly, or too complex, etc. And as the entire enterprise of design is an iterative venture, TMS.T→T and TMS.M→M are those processes of evolving both the theory and the model from its initial incomplete state eventually to its sufficiently detailed state. And, finally, TMS.M→W releases the model/system into the world to be used in solving the intended problem, and, in doing so, often radically changes the world.

4.1.2 Systems – MTMS

It is in the model MTMS that things get really interesting. It is here that we find a variety of worldviews on software systems, software system artifacts, and software system development itself.

- MTMS.MW is the world of software solvable problems and the world of software development combined.
- MTMS.MT theorizes about what forms TMS.T requirements and TMS.M systems should take, and how the various related TMS processes should be structured and used.
- MTMS.MM reifies MTMS.MT into realized models of requirements (TMS.T), systems (TMS.M) and development processes (TMS.W→T, etc).

4.1.2.1 MTMS.MW – World of Software Systems

The world of software systems is a varied and multi-faceted world. It is a world of problems and solutions [12]. It is a world where some problems are not solvable at all by automation as well as a world where some problems are just too hard to solve at all [10]. For the problems that are solvable, there are those that are solvable by what Vincenti [29] calls normal design and those that are solvable only by radical design. We may or may not be successful in solving problems that require radical design, but when we are successful we almost always need several iterations before we achieve that success [7].

It is a world of rapid technological change where software-intensive systems are increasingly invading our lives, where computation is constantly getting faster and cheaper, and where electronic storage is getting larger, faster and cheaper as well. It is a world where the bases for design decisions are constantly changing, where the tradeoffs we previously made must be re-examined in the light of the current state of the world.

4.1.2.2 MTMS.MT – Theories of Software Systems

Frustratingly, there is little theory that is explicit in MTMS.TM; it is by-and-large implicit. Or, more specifically it is often stated normatively rather than descriptively (as one would find in natural sciences, for example). In one way, this is not surprising as our theories in TMS are largely normative: the system ought to do …; it ought to respond within …; it must provide …. Indeed, this normative approach is a feature of the sciences of the artificial [27]. And, of course, it is seen all too easily in every new *salvation du jour*.

However, as my goal in this paper is to lay a foundation for empirical software engineering, I claim that to make progress towards the kind of rigor we find in natural and behavioral sciences, that for this level of discourse we need to be more descriptive – that is, we need to be more explicit about our theories in such a way as to be easily testable.

Ignoring those issues for the time being, let's consider some of the relevant theories found in MTMS.[1]

Finkelstein and Nuseibeh's multiple viewpoints [17] approach implicitly embodies theoretical implications about TMS.W, TMS.T and TMS.W→T: there are different stakeholders with respect to the problem to be solved; these stakeholders have different views on what is important in the software solution;

[1] Please note that I am not trying to be in any way complete, or even representative. The intent here is merely to be illustrative.

these different views need to be captured in the requirements; and eventually any and all apparent and real conflicts need to be resolved to provide a consistent set of requirements (i.e., a consistent theory).

Common theories in MTMS.TM about the form that a model TMS.M (or parts of the model) should take include structured programming [8], object oriented programming [4], aspect-oriented programming [13], etc. Looking at TMS.M in a different way, there are the theories about creating systems bottom up or top down, or about structuring them for future change [19], or about organizing them hierarchically [20], as networks of cooperating processes [9], or to reflect the shape of the problem [12]. There are those who theorize that the components in software systems should be orthogonal and each component do one thing well [14] while others such as Jackson indicate we should be mindful of the fact that the world where we find our problem space has been implemented with the full exploitation of the Shanley Principle [12] of efficient design where each element serves multiple purposes.

There are a variety of theories in MTMS.MT about how we do the transformation from requirements to the system (TMS.T→.M). The more or less standard ones include waterfall development [23], Boehm's spiral development [5], refinement [30], etc. A more radical departure from these standard approaches is that of Extreme Programming [2]. An interesting variation of refinement can be found in Batory's algebraic compositional approach [1].

4.1.2.3 MTMS.MM – Models of Software Systems

There are a wide variety of models we use for various aspects of both TMS.T and TMS.M. For example, we often use scenarios [22] to provide examples of behavior in T. We often provide checklists, templates, style guides, etc for both requirements documents (as well as system architecture, design and code) to represent the models for our theories of requirements and systems.

We have a variety of design methods that give us the models for TMS.T→M – for example Bergland [3] describes and compares functional decomposition, dataflow decomposition, data structure driven, and the programmer's calculus, design methods.

In a more formal approach, van Lamsweerde proposes formally specified goal-oriented descriptions of TMS theories in KAOS [16]. Both van Lamsweerde [15] and Brandozzi and Perry [6] provide models of TMS.T→M – that is, they provide methods for transforming requirements into architectures.

The technical transfer of systems from development into use (i.e., injecting the model into the world, TMS.M→W) can be accomplished in different ways. One way is to ensure that the system fits seamlessly into the intended processes (cf Perry et al [32]). Alternatively, Extreme Programming calls for the involvement of the customer as a critical component in this process (cf [2]).

4.2 Developments, Instruments, Experiments

While the above model is an obvious interpretation of TM and MTM, project management is not such an obvious one. I claim, however, that project management plans and project management processes are also designed artifacts and hence proper models of TM and MTM. The world of projects (i.e., developments, TMD.W) include resources (including people, time, cost, etc), the part of the world to which the system is intended to help, as well as TMD and MTMD discussed above. Developments are subject to specific requirements (TMD.T) and have project plans (TMD.M) that satisfy those requirements. And the processes of TMD for designing, evolving and managing development plans are analogous to those of producing systems in TMS. Analogously MTMD theorizes about, and reifies, the nature and form of development requirements, project plans, project management processes.

The creation of empirical instruments and empirical designs are also examples of models for TM and MTM: TMI and MTMI in the case of instruments and TME and MTME in the case of empirical designs. The models of TMI worry about creating empirical instruments and designs and the models of MTMI address the issues of the form and structures of instruments and designs and the processes of creating and evolving them. Similarly, the models of TME worry about creating empirical study designs and the models of MTME address the issues of the form and structures of empirical study designs and the processes of creating and evolving them.

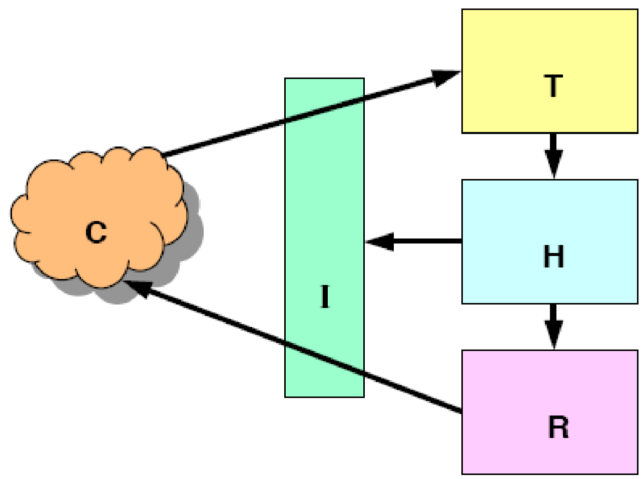

Figure 4: Empirical Theory & Model *E*

5. EMPIRICAL THEORY & MODEL *E*

As I did with TM and MTM, I here posit a theory about a theory and model E for empirical studies (see figure 4 – note that I use this term instead of experiments to provide general approach to empirical work to avoid confusing the general use of the term with the specific technical use of the term). As I did with TM I will also apply E to itself below. I will them apply E and EE to the models described above to generate empirical models for these models.

5.1 Theory of E

Not surprisingly, the theory E basically an elaboration of basic empirical science discussed above.

- Given a theory T, generate an hypothesis H to test some part of the theory
- From the hypothesis H, generate a treatment – i.e., a regimen R – that 1) manipulates the independent variables and 2) uses instruments I to observe the manipulations of the regimen on the dependent variables in the context C relevant to the empirical study

- On the basis of the instrumented observations of the context C, reconcile the observations with the theory T, revising the theory T if necessary

I note that this is a very basic theory, but it still is sufficiently rich to cover the entire range of studies from exploratory through to rigorously explanatory studies. Of course, theory T may be vague and ill-formed (as it would be for exploratory work) or well-formed and mature (as it should be when doing explanatory work). Similarly the hypothesis may be generic and open-ended or focused and specific. Instruments and regimens may be human and opportunistic (for exploratory work) or specifically and well-designed. Further, theory T supports both theory generation (in the case of exploratory work) and focused evaluation of existing theory.

5.2 Model of E

The basic elements in the model and their interrelationships are: theory **T**, hypothesis **H**, regimen (treatment) **R**, instrument **I** and context **C** (where R is to be applied and observed by I, and where the independent and dependent variables are).

The following transformations represent the processes of conducting an empirical study.

- **T → H** – deriving an hypothesis H from theory T
- **H → (I, R)** – generating the experimental design from H
  - H → I – create an appropriate mechanism (based on H) for observing independent and dependent variables
  - H → R – create appropriate mechanisms and manipulations (based on H) of independent variables
- **(R, I) → C** – performing the experiments
  - R → C – applying the regimen to the context (ie, the independent variables)
  - I → C – using the instrument I to observe the context (i.e., the dependent variables)
- **(I, C) → T** – reconciling theory and reality: using instrument I to provide observations about Context C to compare against the current theory T

In this model, I use **E(T, H, R, I C)** to represent an empirical study.

6. APPLYING E TO TM: *ETM*

While it is often the case that we experiment using models, I apply E here to TM rather than to its models of TM because I want to apply E to all of TM's models. The first three empirical studies (ETM1 – ETM3) evaluate the qualities of the theory T and model M with respect to

- The adequacy of theory TM.T representing some part of the world,
- The adequacy of the model TM.M representing theory TM.T, and
- The utility of the model TM.M in the world TM.W

The other seven empirical studies (ETM4 – ETM9) provide the means for "in process" as well as "post mortem" evaluations of the various basic processes or transformations in TM:

- In generating and evolving theory TM.T,
- In creating and evolving model TM.M, and
- In injecting model TM.M into the world TM.W

I describe and explain each empirical study in turn. To keep things as simple as possible, I have not prefaced any of the elements of TM since T is the only ambiguity between the elements of TM and E, and in what follows in this section, I always mean TM.T when I refer to T.

ETM1: E(T, H, R, I, W) – Theory Representativeness
This study addresses the question "How well does the theory T represent the part of the world W it is meant to capture?" The regimen R is a comparison based on selected criteria; the instrument I is a mechanism that captures the data relative to those criteria. In a sense this the fundamental empirical question, but in the sciences of the artificial (i.e., design disciplines) there are special considerations apart from those found in natural and behavioral sciences. Since we have selectable technological constraints and malleable intellectual constraints that are part of the equation, the question of how good the theory is, or how well it represents the problem we want to solve, assumes extra dimensions. The idea is to evaluate these different aspects of these dimensions in this empirical study.

ETM2: E(M, H, R, I, T) – Model Adequacy
This study addresses the question "How well does the model M represent the theory T?" or "How good a model is M relative to the theory T?" The hypothesis H focuses on some part of the of the theory T, regimen R is a set of tests to exercise model M for the part of theory T represented in hypothesis H, and instrument I is the mechanism that provides the needed observations of model M in response to those tests in R. Assuming that H is the alternative hypothesis (rather than the null hypothesis), then the observations provided by I should match the predictions embodied in H if M is an adequate representation of T for that part of T tested by H. Obviously, a full set of hypotheses is needed to cover all of T and thus EMT2 represents a set of studies to determine the adequacy of M relative to T.[2]

ETM3: E(M, H, R, I, W) – Model Utility
This study addresses the question "How useful is the model M of theory T in the world W?" Again, hypothesis H focuses on some part of the model M, regiment R is the normal use of M, and instrument I is a set of mechanisms that provides appropriate observations of the use of M to determines its utility – i.e., how well it addresses the problem it was meant to solve.

ETM4a: E(W→T, H, R, I, W)
This study evaluates the theory generation process.

ETM4b: E(W→T, H, R, I, W)
While the abstract form of this study is identical to the preceding, its interpretation is different. It evaluates the process of theory adaptation in a changing world. The changes in the world may result from injecting the model M into it, or the changes may be independent of the theory and model. In either case, changes may need to be made to the theory as a result of those changes in the world.

ETM5: E(T→M, H, R, I, W)
This study evaluates the process of creating or generating model M from theory T.

---

[2] In the process of reverse engineering and/or evolution, one might want to use the study **ETM2B: E(T, H, R, I, M)** to address the question of whether a theory T is a theory for model M – i.e., to understand how good a theory T is with respect to model M when trying recreate a theory for an existing model.

ETM6: E(M→T, H, R, I, W)
This study evaluates the process of adapting theory T to model M.

ETM7: E(T→T, H, R, I, W)
This study evaluates the process of evolving theory T.

ETM8: E(M→M, H, R, I, W)
This study evaluates the process of evolving model M

ETM9: E(M→W, H, R, I, W)
This study evaluates the process of injecting model M into the world W.

7. APPLYING E TO MTM: *EMTM*

As MTM was significantly more complex than TM, so EMTM is significantly more complex than ETM. I note, however, that this complexity only holds for the issues of theory representativeness, model adequacy and model utility. EMTM4 – EMTM9 are essentially identical to ETM4 – ETM9 given appropriate substitutions of MT for T, MM for M and MW for W. I will thus focus on the former and leave the latter as an exercise for the reader.

MW here takes on specific significance: it is the world involving the creation and production of a design and design artifact. MT is the theory about such creation.

7.1 Theory Representativeness in EMTM

The general question is how well does a theory represent or capture the part of the world it is meant to apply to. As I mentioned above, this goes beyond the basic focus of empirical studies in the natural and behavioral worlds because of the selectable technological constraints and the malleable intellectual constraints. In doing this evaluation, we focus both on MTM as a whole and then on MTM's individual aspects. Regimen R is a comparison using criteria selected to specifically evaluate a theory's representativeness. Instruments I are mechanisms that capture the data relative to those criteria.

EMTM1: E(MT, H, R, I, MW)
This study evaluates how well theory MT represents world MW.

EMTM1a: E(MT.W, H, R, I, MW)
This study evaluates the representativeness of MT's theory of TM's world W in world MW.

EMTM1b: E(MT.T, H, R, I, MW)
This study evaluates the representativeness of MT's theory of TM's theory T in world MW.

EMTM1c: E(MT.M, H, R, I, MW)
This study evaluates the representativeness of MT's theory of TM's model M in world MW.

EMTM1d1: E(MT.W→T, H, R, I, MW)
This study evaluates the representativeness of MT's theory of TM's process of theory generation in world MW.

EMTM1d2: E(MT.W→T, H, R, I, MW)
This study evaluates the representativeness of MT's theory of TM's process of theory adaptation in a changing world MW.

EMTM1e: E(MT.T→M, H, R, I, MW)
This study evaluates the representativeness of MT's theory of TM's process of model generation of M from T in world MW.

EMTM1f: E(MT.M→T, H, R, I, MW)
This study evaluates the representativeness of MT's theory of TM's process of adapting theory T to model M in world MW.

EMTM1g: E(MT.T→T, H, R, I, MW)
This study evaluates the representativeness of MT's theory of TM's process of evolving theory T in world MW.

EMTM1h: E(MT.M→M, H, R, I, MW)
This study evaluates the representativeness of MT's theory of TM's process of evolving model M in world MW.

EMTM1i: E(MT.M→W, H, R, I, MW)
This study evaluates the representativeness of MT's theory of TM's process of injecting model M in world W in world MW.

These 11 empirical studies cover the classes of studies that evaluate the theory representativeness of MT relative to its relevant world MW.

7.2 Model Adequacy in EMTM

The general question of concern here is how well, or adequately, does the MTM's model MM represent MTM's theory MT. Hypothesis H focuses on some part of theory MT with which to evaluate MM. Regimen R is set of tests to exercise MM selected to specifically evaluate how well the hypothesis is satisfied by the model MM. Instruments I are mechanisms to provide appropriate observations of MM to determine its adequacy relative to TM.

EMTM2: E(MM, H, R, I, MT)
This study evaluates how well model MM represents theory MT – that is, MM's adequacy as a model of MT.

EMTM2a: E(MM.W, H, R, I, MT.W)
This study evaluates the adequacy of MM's model of W relative to MT's theory of W.

EMTM2b: E(MM.T, H, R, I, MT.T)
This study evaluates the adequacy of MM's model of T relative to MT's theory of T.

EMTM2c: E(MM.M, H, R, I, MT.M)
This study evaluates the adequacy of MM's model of M relative to MT's theory of M.

EMTM2d1: E(MM.W→T, H, R, I, MT.W→T)
This study evaluates the adequacy of MM's model of the process of theory generation relative to MT's theory of the process of theory generation.

EMTM2d2: E(MM.W→T, H, R, I, MT.W→T)
This study evaluates the adequacy of MM's model of the process of theory adaptation in a changing world relative to MT's theory of the process of theory adaptation in a changing world.

EMTM2e: E(MM.T→M, H, R, I, MT.T→M)
This study evaluates the adequacy of MM's model of the process of model generation of M from T relative to MT's theory of the process of model generation of M from T.

EMTM2f: E(MM.M→T, H, R, I, MT.M→T)
This study evaluates the adequacy of MM's model of the process of adapting theory T to model M relative to MT's theory of the process of adapting theory T to model M.

EMTM2g: E(MM.T→T, H, R, I, MT.T→T)
This study evaluates the adequacy of MM's model of the process of evolving theory T relative to MT's theory of the process of evolving theory T.

EMTM2h: E(MM.M→M, H, R, I, MT.M→M)
This study evaluates the adequacy of MM's model of the process of evolving model M relative to MT's theory of the process of evolving model M.

EMTM2i: E(MM.M→W, H, R, I, MT.M→W)
This study evaluates the adequacy of MM's model of the process of injecting model M in world W relative to MT's theory of the process of injecting model M in world W.

These 11 empirical studies cover the classes of studies that evaluate the theory representativeness of MM relative to its theory MT in the relevant world MW.

7.3 Model Utility in EMTM

The general question is how useful the model of our theory is? How useful do users find the model? Where does it fail? Where does it excel? Hypothesis H focuses on some part of the model. Regimen R is a comparison using criteria selected to specifically evaluate a model's utility. Instruments I are mechanisms that capture the data relative to those criteria.

EMTM3: E(MM, H, R, I, MW)
This study evaluates how useful model MM is in world MW.

EMTM3a: E(MM.W, H, R, I, MW)
This study evaluates the utility of MM's model of TM's world W.

EMTM3b: E(MM.T, H, R, I, MW)
This study evaluates the utility of MM's model of the TM's theory T in world MW.

EMTM3c: E(MM.M, H, R, I, MW)
This study evaluates the utility of MM's model of TM's model M in world MW.

EMTM3d1: E(MM.W→T, H, R, I, MW)
This study evaluates the utility of MM's model of TM's process of theory generation in world MW.

EMTM3d2: E(MM.W→T, H, R, I, MW)
This study evaluates the utility of MM's Model of TM's process of theory adaptation in a changing world MW.

EMTM3e: E(MM.T→M, H, R, I, MW)
This study evaluates the utility of MM's model of TM's process of model generation of M from T in world MW.

EMTM3f: E(MMT.M→T, H, R, I, MW)
This study evaluates the utility of MM's model of TM's process of adapting theory T to model M in world MW.

EMTM3g: E(MM.T→T, H, R, I, MW)
This study evaluates the utility of MM's model of TM's process of evolving theory T in world MW.

EMTM3h: E(MM.M→M, H, R, I, MW)
This study evaluates the utility of MM's model of TM's process of evolving model M in world MW.

EMTM3i: E(MM.M→W, H, R, I, MW)
This study evaluates the utility of MM's model of TM's process of injecting model M into world W in world MW.

These 11 empirical studies cover the classes of studies that evaluate the utility of model MM relative to its relevant world MW.

7.4 MTM Transformations

The other seven empirical studies (EMTM4 – EMTM9) provide the means for "in process" as well as "post mortem" evaluations of the various basic processes or transformations in MTM:

- In generating and evolving theory TM.T,
- In creating and evolving model TM.M, and
- In injecting model TM.M into the world TM.W

I describe and explain each empirical study in turn. To keep things as simple as possible, I have, again, not prefaced any of the elements of TM since T is the only ambiguity between the elements of TM and E, and in what follows in this section, I always mean TM.T when I refer to T.

EMTM4a: E(MW→MT, H, R, I, MW)
This study evaluates the theory generation process.

EMTM4b: E(MW→MT, H, R, I, MW)
While the abstract form of this study is identical to the preceding, its interpretation is different. It evaluates the process of theory adaptation in a changing world. The changes in the world may result from injecting the model MM into it, or the changes may be independent of the theory and model. In either case, changes may need to be made to the theory as a results of those changes in the world.

EMTM5: E(MT→MM, H, R, I, MW)
This study evaluates the process of creating or generating model MM from theory MT.

EMTM6: E(MM→MT, H, R, I, MW)
This study evaluates the process of adapting theory MT to model MM.

EMTM7: E(MT→MT, H, R, I, MW)
This study evaluates the process of evolving theory MT.

EMTM8: E(MM→MM, H, R, I, MW)
This study evaluates the process of evolving model MM

EMTM9: E(MM→MW, H, R, I, MW)
This study evaluates the process of injecting model MM into the world MW.

These 7 empirical studies cover the classes of studies that evaluate the theory representativeness of MT relative to its relevant world MW.

8. MODELS OF *ETM* & *EMTM*

The ultimate goal of empirical software engineering is to elaborate these two models with empirical studies. The previous sections have provided a taxonomy of classes of empirical studies. But even within each class there are a plethora of possible models, ranging from exploratory to explanatory, from descriptive to correlational to causal. What ETM and EMTM provide is the landscape of important empirical studies for design disciplines.

I use the same domains as models for ETM and EMTM as I did for TM and MTM (i.e., systems, developments, instruments and experiments) but will focus on ETMS and EMTMS (empirical studies for systems).

We currently do little more than a "hand-wave" argument about ETMS1 to establish theory representativeness. We do ETMS2 (model adequacy), very informally under the general topic of "testing' (and perhaps some analysis). We do better for ETMS3 (model utility) with our studies ranging from informal demonstrations to carefully constructed experiments [25]. For some system domains we use benchmarks [26]. For the software developments processes themselves, we do little except to propose them.

For EMTMS, we do little at all, except when portions of the processes of MTMS are incorporated into tools (which then reduces to ETMS and the state described above).

9. APPLYING E TO ITSELF: *EE*

The purpose of applying theory E to get theory EE is to provide a theory of empirical evaluation of an empirical study itself. I sketch out a set of evaluations that cover the elements and transformations of theory E and indicate what is important about each particular empirical study evaluation. Full discussions of the strengths and weaknesses to be found in empirical studies can be found in such texts as Rosenthal and Rosnow [24].

EE1: E(E.H, H, R, I, E.T)

Evaluate the hypothesis E.H in the context of E.T. That is, is E.H relevant to E.T? Does E.H have constructs that are appropriate to E.T? This goal of EE1 is to evaluate the intentional and representational aspects of construct validity.

EE2: E(E.R, H, R, I, (E.H,E.C))

Evaluate regimen E.R relative to the hypothesis E.H in the context E.C. The goal of EE2 is to evaluate the internal validity and robustness and reliability of the empirical study.

EE3: E(E.I, H, R, I, (E.H, E.C))

Evaluate instrument E.I relative to the hypothesis E.H and the empirical context E.C. Does E.I provide us with the ability to observe and measure what we need to observe and measure in the empirical context? Again, this addresses the issues of intentional and representational construct validity.

EE4: E(E.C, H, R, I, C)

Evaluate the study context E.C of the context of a broader, more inclusive context. The goal is to evaluate the external validity of the empirical study.

EE5: E(E.T→E.H, H, R, I, C)

Evaluate the process of deriving the hypothesis E.H from the theory E.T. This study addresses primarily the problem of internal validity, though there are some aspects of representation validity as well.

EE6: E(E.H→E.R, H, R, I, C)

Evaluate the process of deriving an appropriate regimen E.R from the hypothesis E.H. Again, this addresses primarily the problem of internal validity and some construct validity.

EE7: E(E.H→E.I, H, R, I, C)

Evaluate the process of creating appropriate instruments E.I from the study's hypothesis E.H. This study addresses the issue of observational (construct) validity.

EE8: E(E.R→E.C, H, R, I, C)

Evaluate the process of applying the treatments E.R to the subjects in context E.C (i.e., manipulating the independent variables and observing results in the dependent variables). This study evaluates internal validity.

EE9: E(E.I→E.C, H, R, I, C)

Evaluate the process of using the instrument to observe the treatment E.R being applied in the context E.C – i.e., the effect of the instrument on the context (process artifacts). This addresses primarily the problem of internal validity, but also observational validity.

EE10: E(E.C→E.T, H, R, I, C)

Analyze the effects of the manipulations in context E.C on the subjects and how it affects the theory E.T. This study addresses the issues of internal, statistical and external validity.

EE11: E(E.I→E.T, H, R, I, C)

Evaluate the analysis of the instrumented results – i.e., the effects of instrument E.I on the theory E.T. Again, this addresses the issues of internal and statistical validity.

These 11 studies establish the quality of an empirical design, specifically its reliability and robustness and its construct, internal, statistical and external validity. We as a field are even less mature in evaluating the characteristics of our studies. As an example to aspire to, consider the 11 volume set, *Test Critiques* [32] in which more than 800 tests are summarized as to what each provides, its strengths and weaknesses as well as its reliability and robustness.

10. CONCLUSIONS & CHALLENGES

I have presented a unifying foundation for software engineering and, more specifically, empirical software engineering. I have structured the paper in the fashion of the subject of the paper itself: as theories and models. I have introduced two basic theories: TM and E. I have used these theories to generate the remaining theories I believe we need for the foundations of empirical software engineering: MTM by applying TM to itself; ETM and EMTM by applying E to TM and MTM; and finally EE by applying E to itself. I have illustrated TM and MTM by introducing four interpretations (i.e., models: systems, developments, instruments and experiments) and discussed one (systems) in detail. I have then discusses the state of ETM and EMTM with a few examples from the models ETMS and EMTMS (i.e., empirical studies of systems).

EMT yields 10 classes of empirical studies, and EMTM yields another 40 classes of studies for a combined total of 50. EE yields another 11 classes of evaluations of empirical studies that should be applied to each of the 50 classes mentioned above. It is clear we have barely begun to explore this space of empirical studies in one model much less in all four.

The challenges in establishing a rigorous discipline of software engineering are significant:

- Explicate the theories behind the models and processes we propose. Further for MTM theories we should strive for more descriptive approaches to gain more specific advantages in testability.
- Create instruments and designs for the various kinds of empirical studies appropriate within each of the classes delineated above.
- Apply these designs rigorously to evaluate software engineering ideas, be they theories, models or the transformations embodied in TM and MTM.
- Treat the issues of theory representativeness, model adequacy (in TMS, testing), and model utility as the rigorous empirical enterprises that they, in reality, are (and should be).

There is enormous potential to be realized in a rigorous empirical discipline of software engineering. While we have accomplished much in using software systems, we are still far from achieving a rigorous discipline for software engineering. We will achieve that rigor only when we have incorporated a rigorous empirical discipline into software engineering.

11. ACKNOWLEDGMENTS

Wlad Turski, University of Warsaw, Wen-Qian (Wendy) Liu, University of Toronto, and Mike Evangelist, The University of Texas at Austin, have contributed significantly to this paper through their discussions over the past several years. Of course, they are not responsible for any of its problems.

12. REFERENCES

[1] D. Batory, J.N. Sarvela, and A. Rauschmayer. "Scaling Step-Wise Refinement", IEEE Transactions on Software Engineering, June 2004.

[2] Kent Beck. Extreme Programming Explained. Reading: Addison-Wesley, 1999.

[3] G. D. Bergland. A guided Tour of Program Design Methodologies, IEEE Computer, Oct 1981, 13-37.

[4] Graham M Birtwistle, Ole-Johan Dahl, Bjorn Myhrhaug, Kristen Nygaard. Simula Begin. New York: Petrocelli/Charter, 1973.

[5] Barry W. Boehm. A Spiral Model of Software Development and Enhancement, IEEE Computer, May 1988, 61-72

[6] Manuel Brandozzi and Dewayne E Perry. "Transforming Goal Oriented Requirements Specifications into Architectural Prescriptions." Workshop From Software Requirements to Architectures (STRAW1), International Conference on Software Engineering 2001, Toronto, May 2001, 54-61.

[7] Frederick P. Brooks, Jr. Mythical Man Month, Anniversary Edition with Four New Chapters. Reading: Addison-Wesley, 1995.

[8] P-J. Dahl, EW Dijskstra and CAR Hoare. Structured Programming. London: Academic Press, 1972.

[9] E W DIJKSTRA. Cooperating Sequential Processes. Math. Dep., Technological U., Eindhoven, Sept. 1965

[10] L. Fortnow, Steve Homer. A Short History of Computational Complexity. In D. van Dalen, J. Dawson, and A. Kanamori, editors, *The History of Mathematical Logic*. North-Holland, Amsterdam, 2002

[11] Gooding, David, Pinch, Trevor, and Schaffer, Simon, Editors. *The Uses of Experiment: Studies in the Natural Sciences.* Cambridge: Cambridge University Press, 1989.

[12] Michael Jackson. The World and the Machine, 17th International Conference on Software Engineering, 1995, Seattle WA, 283-292.

[13] A19 Kiczales, G., Lamping, J., Mendhekar, A., Maeda, C., Videira Lopes, C., Loingtier, J.-M., and Irwin, J. Aspect-Oriented Programming. Proc. of ECOOP 1997.

[14] Butler W. Lampson. Hints for Computer System Design. IEEE Software, January 1984, 11-30.

[15] Axel van Lamsweerde. From system goals to software architecture. In Bernardo M., Inverardi, P., eds.: Formal Methods for Software Architectures. Volume 2804 of Lecture Notes in Computer Science. Springer-Verlag (2003) 25–43

[16] Philippe Massonet, Axel van Lamsweerde. Formal refinement patterns for goal-driven requirements elaboration. In: FSE-4 - 4th ACM Symposium on the Foundations of Sofware Engineering, San Fransisco, ACM Press (1996) 179–190

[17] Bashar Nuseibeh, Jeff Kramer, Anthony Finkelstein. Expressing the Relationships Between Multiple Views in Requirements Specification, International Conference on Software Engineering, 1993.

[18] Lee J. Osterweil. Software Processes are Software Too, *Ninth International Conference on Software Engineering,* Monterey, CA, pp. 2-13, 1987.

[19] David L.Parnas. Designing Software for Ease of Extension and Contraction. IEEE Transactions on Software Engineering, SE-5, 2 (March 1979), 128-138.

[20] David L Parnas. On the criteria to be used in decomposing systems into modules, Communications of the ACM, 15 :12, 1053-1058, Dec. 1972

[21] D.E. Perry, A.A. Porter, L.G. Votta and M.W.Wade. "Reducing inspection interval in large-scale software development", IEEE Transactions on Software Engineering, 28:7 (July 2002), 695-705

[22] Colin Potts, Kenji Takahashi and Annie I Anton. Inquiry-Based Requirements Analysis, IEEE Software, March 1994, 21-32.

[23] Winston W. Royce, Managing the Development of Large Software Systems: Concepts and Techniques. Technical Papers of Western Electronic Show and Convention (WesCon). August 1970, Los Angeles CA

[24] Robert Rosenthal and Ralph L. Rosnow. Essentials of Behavioral Research: Methods and Data Anlaysis. Boston: McGraw-Hill, Second Edition, 1991.

[25] Danhua Shao, Sarfraz Khurshid and Dewayne E Perry, "Predicting Faults from Direct Semantic Interference: An Evaluative Experiment", January 2007, submitted for publication.

[26] Susan Elliott Sim, Steve Easterbrook, and Richard C. Holt. Using Benchmarking to Advance Research: A Challenge to Software Engineering, Proceedings of the Twenty-fifth International Conference on Software Engineering, Portland, Oregon, 74-83, May, 2003.

[27] Simon, Herbert A. *The Sciences of the Artificial.* Cambridge: MIT Press, 1969.

[28] Turski, Wladyslaw M. and Maibaum, Thomas S. E. *The Specification of Computer Programs*. Reading, Mass: Addison-Wesley, 1987.

[29] A16 Walter G. Vincenti. What Engineers Know and How They Know It. Baltimore: The Johns Hopkins University Press, 1990.

[30] Niklaus Wirth, Program Development by Stepwise Refinement. Communications of the ACM, 14:4, April 1971, 221-227

[31] Wittgenstein, Ludwig. *Tractatus Logico-Philosophicus*, ed. A. J. Ayer, London: Routledge & Kegan Paul, 1961. Section 5.6.

[32] D J Keyser & R C Sweetland, Editors. Test Critiques. Austin TX: Pro-Ed. Updated Annually