Discerning the Intent of Maturity Models from Characterizations of Security Posture

Rich Caralli

January 2012

MATURITY MODELS

Maturity models in their simplest form are intended to provide a benchmark against which a characterization of achievement can be made. Maturity models typically represent a set of attributes, characteristics, patterns, or practices that are arranged in an evolutionary scale that represents measureable transitions from one level to another. In other words, maturity models depict the evolution or scaling of attributes, characteristics, patterns, or practices from some primitive state to a more advanced, or “mature” state.

The “measurable transitions” in maturity models should be based on empirical data that has been validated in practice; that is, each step in the model should be able to be validated as being more “mature” than the previous step. This is very difficult to do, and is often lacking in maturity model representations.

Progression Models

Models that represent a simple progression or scaling of an attribute, characteristic, pattern, or practice are typically referred to as **progression models** in that the movement up the maturity levels indicates some progression of maturity. The names of the levels should be indicative of the transition “state”; that is, the name of the level should represent the level of maturity represented collectively by the characteristics, patterns, attributes, or practices that exist at that state. For example, a simple level progression for a maturity model that characterizes states of human mobility might be as follows:

- Level 1: Crawl
- Level 2: Walk
- Level 3: Jog
- Level 4: Run
- Level 5: Sprint

Progression models are often criticized because the levels or “states” can be arbitrary and there is no validation for the transition between the states.

Software Engineering Institute
Carnegie Mellon University
4500 Fifth Avenue
Pittsburgh, PA 15213-2612

Phone: 412-268-5800
Toll-free: 1-888-201-4479

www.sei.cmu.edu

Capability Maturity Models

A **capability maturity model** is a unique application of the maturity model concept. In a capability maturity model, the dimension that is being measured is a representation of organizational capability around a set of attributes, characteristics, patterns, or practices. In other words, the transition between the "states" is an evolution of the capability of the organization relative to the subject matter of the maturity model. In the SEI's CMMI for Development for example, the organization's capability for managing software development/engineering processes is being measured. This extends to other models developed by the SEI which use the CMMI architecture:

- CMMI for Acquisition measures the organization's capability for managing the software acquisition process
- CMMI for Services measures the organization's capability for managing and delivering high-quality services
- CERT-RMM measures the organization's capability for managing the operational resilience of key assets, services, and missions

The common thread in these models is the measurement of capability against a benchmark of institutionalizing features which appear at each level of the model and for which the model levels are named. For example, at Level 2 – Managed, the organization is exhibiting the capabilities of planning their software engineering processes, assigning resources to these processes, periodically measuring these processes, etc. At Level 3 – Defined, the organization is exhibiting the capabilities of being able to follow consistent definitions of processes across business units and instilling the capability of sharing improvement information across the organization. These institutionalizing features are independent of the core subject matter of the model; in other words, the specific model content (security practices or characteristics, for example) is augmented with a consistent set of features that indicate capability.

In short, a capability maturity model seeks to measure organizational capability at each level based on a set of institutionalizing features (that have an empirical basis), regardless of the subject matter of the model. Thus, applying the capability maturity scaling in CMMI, capability can be measured for any virtually subject matter model.

Hybrid models

A hybrid maturity model can be created by overlaying characteristics of the progressive model with capability attributes from capability maturity models. This is the type of model being considered for the DOE. In a hybrid maturity model, the progression model architecture is used to characterize the attributes, characteristics, patterns, or practices, but the transition states reflect the hierarchy and characteristics of a capability maturity model. In other words, the institutionalizing features of capability are overlaid to the subject matter in the progressive model. The advantages of this type of model are

- It is easier to create, understand, and use
- It provides a representation of the evolution of attributes, characteristics, patterns, and practices (which may be helpful for organizations seeking to use the model as a roadmap)
- It provides "hooks" into the capability characterization (so that some representation of capability can be expressed and the model can eventually be evolved into a full-scale CMM)
- Assessment or appraisal against the model can be simplified (relative to the appraisal needs of a full-scale CMM)

CHARACERIZING SECURITY POSTURE

Models that attempt to characterize security posture operate at a different level than maturity models, for the most part, and attempt to measure something different than maturity models.

Posture is an expression of something relative to something else. **Security posture** is defined an organization's security state relative to some threat (or collection of threats) at a given point in time. Security posture is constantly changing as the organization's threat profile changes, but over time consistent patterns may be observable. Security posture is also highly unique to each organization because threat environments vary widely depending on many factors.

In characterizing security posture, the organization must *attempt* to determine how well it is currently protecting and sustaining key assets and services relative to an expression of its current and unique threat exposure. Thus, many different pieces of information may be necessary to attempt to accurately express security posture. For example, an organization might need to know

- What current risks it is exposed to (identified through risk assessment)
- What current vulnerabilities it is exposed to (identified through threat or vulnerability assessment)

- The incidents that it has been affected by and why they resulted in impact to the organization (identified through the organization's incident management process)
- Network flow data that indicates intrusion attempts, malware execution, etc. (identified through monitoring processes)
- Log data that identifies unauthorized access attempts or insider activity (identified through monitoring processes)
- Etc., etc., etc…

Interestingly, another piece of data that is useful for characterizing security posture is maturity level or capability maturity level. This tells the organization the degree to which security processes have matured and become institutionalized in the organization, and can be added to the list above.

Reductionism and federation

Security posture approximation is difficult at best. Typically, the concept of **reductionism** is applied and the process of **federation** is used. Reductionism means to attempt to characterize the "whole" by federating, or aggregating, the parts. In other words, when a doctor attempts to characterize you as healthy or sick, he is looking at many pieces of data and "federating" them to create a profile. This profile expresses your health posture relative to potential health threats for your age, weight, sex, etc. Relative to federating for security posture, many pieces of data must be identified, captured, aggregated, and analyzed. If the organization does not have a standard formula or framework for federation, security posture approximation will likely be impossible.

MATURITY MODELS VS. SECURITY POSTURE

This document attempts to discern how maturity models and characterizing security posture are different activities that have different intents, outcomes, and uses. Maturity models measure an expression of improvement and evolution (and in some cases, capability), the results of which may or may not directly correlate to a strong security posture. One can draw inferences between the two, particularly when using capability maturity models: high maturity organizations *should* have strong security postures because they exhibit more mature security processes which *should* result in higher levels of resistance of key assets and services to disruption or threat. But, evidence of correlation is often anecdotal. On the other hand, security posture characterization provides a more robust expression of what the organization is exposed to and how well it is poised (or *postured*) to address it, but much data must be collected, analyzed, and federated to be effective. And, without a consistent and validated process for federating this information, universal application of this concept is difficult for single organizations and perhaps impossible across a specific sector.