3.4.2 Domain Definition

The domain definition model specifies the nature and extent of an envisioned set of similar products to be represented by the domain. This model has five elements: a synopsis that characterizes the essential purpose that every encompassed product will satisfy, a catalog of essential terminology, a compendium of relevant legacy knowledge, assumptions that characterize similarity among encompassed products, and a set of deferred decisions that in being resolved reduce the product family to a single product.

The domain definition serves three purposes:

- As an informal characterization by which developers maintain a shared understanding of what sort of products are needed by a market;
- As a basis for determining whether capabilities needed by a customer are likely to correspond to an instance of the product family;
- As the conceptual foundation for defining a product family and an associated means for deriving customized products.

Synopsis

The domain synopsis element is a concise description of an *abstraction* that characterizes the essential nature of any product that is within the scope of the domain. The synopsis is meant to provide, using domain terminology, only a general sense of what products properly belong in the product family. Any product that fails to adhere to this description in every respect is excluded from the product family but the synopsis can be modified if the membership of the family is found to be overly broad or too restrictive. The synopsis is elaborated as assumptions of commonality, providing more particular criteria that all instances of the product family must satisfy.

Terminology

The domain terminology element specifies a common topically-organized vocabulary for describing the subject matter and problem-solution space addressed by the domain. This element establishes a uniform market-encompassing terminology for the domain,

supporting an understanding by developers of how customers perceive their endeavors. References may be provided to materials that elaborate in more depth on more complex concepts.

This further identifies and correlates any differing nomenclatures that different customers are found to use in discussing their endeavors such that there is a definitive translation between this common and alternative terminologies.

Legacy

The domain legacy element identifies relevant sources of information that serve as the foundation on which the domain is built. The initial motivating basis for creating a domain, beyond the existence of a perceived market for its products, is accessible developer competence for the development of envisioned products. Legacy content includes any supporting scientific, technical, or market references, any previously developed products or components that fall within the scope of the product family, and any material that may be useful in understanding capabilities that future products may need to support.

Assumptions (of Commonality and Variability)

The domain assumptions element elaborates the domain synopsis to provide a marketspecific description of how products that belong to the envisioned product family are similar. Assumptions of commonality informally expand on the abstraction by which all instances of the product family are seen as being alike. Assumptions of variability informally identify the various ways that included instances can differ from each other.

As an elaboration of the domain synopsis, domain assumptions provide a still informal but more detailed basis for determining the feasibility of deriving a product that will fit a given customer's specific needs. A potentially derivable product is one that satisfies the commonality assumptions of the domain and can be distinguished from other similar products entirely by reference to variability assumptions associated with the domain. A brief informal analysis of assumptions relative to a potential customer's

perceived needs should suffice to indicate whether initiating a product manufacturing effort is warranted.

Commonality assumptions provide an intuitive sense of the sort of products that are properly within the buildable scope of the domain. Based on these assumptions, any envisioned instance of the set is an equally good representative for the general purpose of all products in the set. The instance that is best for a particular customer's needs is characterized in terms of associated variability assumptions (formalized in the decision model element as the basis for the selective derivation of customized products).

A variability assumption can be associated with one or more commonality assumptions to informally identify differences that are a basis for distinguishing among products that are otherwise alike, identifying similarity-based subsets. Each such subset would have additional commonality assumptions not shared by other products in the full set of products and additional variability assumptions that further distinguish among products included in that subset. Continuing this subsetting results in a similarity hierarchy as an informal profile of the nature of products that comprise the domain (i.e., its subdomains, each characterized by a more restrictive abstraction).

Decision Model

The domain decision model element is a formalization of the differences, as informally expressed in domain assumptions, among the set of products represented by the abstraction associated with the domain. These differences are coalesced into the form of deferred decisions that are meaningful to customers and that in being fully resolved determine a single buildable instance of the product family. Developers having the competence to build products encompassed by the domain lack only the resolution of these decisions to be able to build a product that fits a given customer's needs. A domain leverages this competence by defining a product family in terms of these decisions and providing the means to resolve these decisions and derive the corresponding instance of that family.

Each deferred decision is expressed as a choice related to aspects of product behavior that are meaningful to a customer. The basis for resolving each decision lies in its

3

relevance to needs as understood by the customer. A decision may concern the capabilities a product provides or tradeoffs that affect behavioral quality factors. A decision model resolution is "complete" if it is sufficient for deriving a product model from the product family model.

As a primary organizing medium for domain engineering and an essential framework for product manufacturing, a decision model expresses the diversity in the product family defined by a domain. Alternative resolutions of any deferred decision partition the product family into mutually exclusive subfamilies. Each possible resolution of a decision corresponds to a subfamily whose members all share that resolution, meaning those instances are more alike in that respect versus other instances of the containing family. A complete resolution of deferred decisions reduces the candidate set of derivable products to a non-partitionable subfamily¹, which can then be arbitrarily reduced to a single representative instance of the subfamily.

The decision model specifies a normative form for each decision associated with a domain:

- A decision can be discrete or composite with an initially unknown value.
- A decision can be designated as required or optional and single- or multi-valued.
- A discrete decision is an irreducible unit value of prescribed form (such as numeric, textual, graphical, photo, video, or hyperlink, or a computation over decisions that produces such a value).
- A composite decision is an aggregation of one or more constituent decisions (decisions are generally aggregated into a composite due to being logically related to elaborate the aspects of some unifying concept).
- A decision can have an associated discrete form that abbreviates its content (i.e., a discrete or composite decision whose potential values can be categorized into alternative symbolic values).

¹ A non-partitionable subfamily is one in which all instances have, from a customer perspective, equivalent (not necessarily identical) behavior and therefore cannot be further usefully partitioned. The product family model determines which instance is selected.

- A decision can have an associated resolution or consistency constraint based on its relationship with the values of other related decisions or among the elements of a composite or multi-valued decision.
- A resolution can be conditionally derived based on a computation referencing current or prior resolutions of itself or related decisions. A decision's derived resolution may change if the resolutions of those decisions change.

A decision model as a whole is a composite decision (i.e., a set of related decisions that establishes the diversity expressed in the associated product family).