

2.4 Product Environment

The product environment model specifies the composition and information content of an ecosystem: the environment in which the product operates and entities that operate within that environment. The information content, of the environment and of a contained entity, is specified as a referenceable information space. Optionally, this model may also specify the behavior of the environment or of any entity as support for product evaluation or for various other purposes discussed below¹.

The essential purpose of the product environment model is to provide developers with an accurate portrayal of the environment as a dynamic information space within which the product will operate. The environment exhibits properties and phenomena that particular entities have the means to detect or modify. The product accesses entity-provided services to obtain data that represents ecosystem information content and to initiate actions intended to modify the information content of the ecosystem.

The described ecosystem can be some blend of physical and virtual, as appropriate. Physical means existing in the real world, whereas virtual means simulated in software (and which may but need not approximate physical reality). Similarly, each entity may be some combination of physical, emulated, and virtual. Besides specifying a product's actual ecosystem, the product environment model can be viewed as specifying an artificial ecosystem in which the product can operate with, some degree of verisimilitude, as if it were in its actual ecosystem.

Each entity is described as a "logical device", whose information space is related to but apart from that of the environment. Each entity, providing services for sharing information or coordinating action as appropriate, is characterized as an edge device, a user interface (operating via associated interface devices), or a system interface (optionally including an associated set of other logical devices). An entity's information space has two primary partitions: its view of the environment's information content (whether obtained directly or indirectly via other entities) and its view of its own

¹ Multiple products built to operate in the same or overlapping ecosystems may share a common product environment model, referencing only relevant portions of that broader model.

information content (e.g., operational health); in some cases (e.g., a system interface), other partitions may provide indirect access to information content of other entities.

Strictly for descriptive purposes, the environment is specified here as an abstract entity. However, its interfaces are encapsulated within other entities. The environment “entity” is accessible to the software only via other entities that are able to access its information content or initiate phenomena that influence that content.

This model provides a basis for:

- Developing software that will interact with the environment and entities as specified, obtaining and sharing information and initiating actions via related devices to effect the product’s intended behavior
- Building an ecosystem simulation in which anomalous conditions can be injected to evaluate product behavior under such conditions
- Developing empirically-based scenarios (capture history of inputs and resulting effects as a basis for regression testing as a product evolves, for comparative possibly instrumented evaluation of alternative product versions, or to discover implications on product behavior of changes in behavior of the environment or an entity)
- Building a virtual world (a simulated environment and entities that represent an artificial/ non-physical or augmented/ hybrid reality) in which a product can operate (e.g., a game or other fictional endeavor; or as a contained training platform for users; or to evaluate effects on product of changed/ different configuration of environment/ entities)
- Building an alternative reality model (a reality-based model of the environment and entities in which selected properties and phenomena can be modified to analyze resulting effects on behavior of the environment or entity)
- Building a temporal projection (a reality-based model that guides understanding by capturing and analyzing the chronology of changing information (and resulting product reaction) over a span of time or informs decision-making by

projecting and comparing outcomes of alternative options; an aid to understanding (rationale for) product behavior or effects of potential actions on that behavior)

Specifying an Operational Environment

A specification of an operational environment must properly cover all aspects of an ecosystem within the scope of a product but may encompass aspects of the actual operational environment beyond what a particular product needs. In this way, an operational environment model may support the development of multiple products that can operate in the same or similar environments. An operational environment model may be created as a collaborative effort among multiple projects that are intended to operate in the same or sufficiently similar operational environments. The environment model will accommodate all entities that are accessible by any product whereas each product is built to interact only with those entities that are referenced in its requirements model.

Types of Modeled Product Environments

A Basic Product Environment Model

A basic product environment model specifies an ecosystem information space in which a product may operate. This model defines a built environment and entities that operate within that environment. The built environment is the natural environment (i.e., the naturally occurring real world) augmented with human-made objects and structures. A product is typically developed to operate in a relevant portion of the built environment further augmented with access to co-resident entities (i.e., people, devices, and systems). The built environment is characterized by observable phenomena and properties, expressed as measurable quantities, that can be detected by associated entities and be changed as a result of naturally occurring processes or due to the effects of entity actions.

<—————

(what information does each device monitor / affect)

A Physical Environment Model

aspects which correlate with software capabilities (e.g., defining devices needed to monitor or act on the environment, considering effects of terrain and buildings on mobility, how communications is affected by environmental conditions). Processes integral to the environment induce phenomena that influence the expected behavior of the product. Entities operating in the environment may enhance or inhibit the ability of software to obtain needed information or to operate as intended (e.g., devices that monitor environmental phenomena, entities that threaten safety or security of product-critical devices or information). Requirements model constraints may identify assumptions about physical environment characteristics that limit choices about the product's realization, such as communications protocols, hardware selection, or other aspects of infrastructure.

—————>

A Dynamic Environment Model

A dynamic product environment model is a basic product environment model extended with specifications of the behavior of the environment and associated entities.

dynamics (processes that produce behavior): simple state model or simulation

A Scenarios-Enhanced Environment Model

{define related / addressable elements of ecosystem & transactional interface specs: systems & devices?; info content / communication specs / perf specs; how elements chg envr vars} {maintain as elements evolve} {how ecosystem operates on the envr or as a distinct source or target of info} {scenarios that portray how other systems are behaving + simulation of other systems sending / receiving messages} {simulate people / business processes?} {scenarios that portray how the envr is behaving, in terms of envr vars; simplification: vars are static except due to sys effects}

An Instrumented Environment Model

Ecosystem Simulation

{incl simul of ecosystem envir and entities behaviors within which product can operate in lieu of actual envir/ecosystem}

A product can be built to operate in a simulated realization of an envisioned environment (e.g., as a medium for game or scientific experimentation products) rather than in an actual physical environment. In this case, the operational environment model is the basis for constructing an environment simulation within which a product is injected for its intended use.

An Artificial Environment Model

An artificial operational environment can be needed either as a simulated reality in which a product will operate or as a platform for evaluating a product within an approximation of its actual operational environment. The provision of an artificial operational environment is the realization of a product in its own right, having its own underlying environment, associated devices, and capabilities that define its expected behavior. A product can be built to operate in such an environment without any knowledge that the environment is not natural or physically realized beyond its computational mechanisms.

In extended form, this model defines a framework for creating a software platform that approximates the envisioned operational environment. Such a platform can be built to accommodate actual or artificial (emulated or simulated) realizations of entities as may exist in the actual environment. Interactions with entities can be instrumented to provide experimental monitoring and control of their behavior, supporting analyses of product behavior. If a product is being built to operate in a virtual environment (e.g., games or scientific simulations), this platform would become the actual operational environment for the product. Building such a platform would entail a recursive application of the activities of software product engineering. In any case, the operational environment model must be verified as a sufficient approximation of the

actual environment to the degree that it is an adequate basis for ensuring that the product will operate correctly in the actual environment.

A Product-Embedded Virtual Environment Model

Product Engineering for an Artificial Environment

(either a virtual op env for a product or a simulated environment product)

Injecting, Exercising, and Monitoring a Product in an Artificial Environment

A product is injected into a system to transform its behavior within a containing ecosystem. A product operates by obtaining information from the ecosystem and initiates actions via devices that may change that information. The operational environment model is a representation of the ecosystem within which that system operates. An environment and systems that operate within it consist of information and agents by which such information changes. This representation specifies the means by which a product is able to interact with the environment and other elements of the ecosystem. It can also be elaborated as a specification for creating a facsimile of the actual ecosystem into which the product can be experimentally deployed and evaluated.

A product relies on a suitable facsimile of/verisimilitude to/accurate tracking of relevant aspects of the ecosystem information space to guide its behavior. Its ability to foresee changes, either organic or as the effect of entity actions, depends on the

product's model of ecosystem behaviors. Environment and related entity processes express as dynamics of change.

- formulate an infrastructure that represents the composition and working of the ecosystem as a whole, identifying hardware devices and other (black-box) systems/elements, that serve as a product's operational environment
- provide a model that defines the information content of the operational environment and simulates associated natural processes that determine values
- formulate a means to represent data/control channels that provide connections between infrastructure elements
- provide the means to inject a software object as the behavior of any designated element of the infrastructure; provide software objects that emulate the behavior (including timing and errors; normal/expected and abnormal/unexpected behaviors) of black-box systems operating in the environment
- formulate the means to connect hardware devices or to inject software objects that emulate the behavior of devices (that sense environmental data and effect changes in that data)
- formulate the means to emulate business process and the associated expected behavior of human users in scenarios that correspond to their performance of responsibilities in designated working roles within the system
- provide the means to instrument virtual (simulated or emulated) elements for observability (to monitor actions and collect data)

This includes specifications of (1a) the physical environment (an abstract data model) (1b) hardware devices that map data (a concrete data model), (2) user roles and scenarios for performing customer business processes, (3) other systems with which the product will need to interact, and (4) capabilities that users need in a product to monitor or modify the ecosystem to better meet their needs. This includes as-is models of the

environment and potential to-be models that might better support the envisioned product.

an infrastructure that simulates the ecosystem, emulating hardware devices and other systems/elements, that serves as a product's operational environment {what are the relevant elements and behavior of the operational environment (product context) within which the business process/procedures/practices/capabilities are applied to pursue business objectives?}

- *representation of environment (ecosystem?) into which the product will be injected to operate*
- *basis for defining product interactions (monitoring or controlling) with relevant real-world data values (an information model of externally occurring values)*
- *basis for creating software encapsulations that emulate supporting hdw devices (to what degree of fidelity: how accurate and precise is the device to reality, match that in virtual space and time)*
- *basis for creating product elements that model the environment to predict potential changes or action effects*
- *basis for creating a simulated environment for pre-deployment testing*
- *basis for simulating business/user/operator processes (operator interacts with hdw devices, user performs data processing functions) based on scenarios representing product usage and utility*

In describing the operational environment, the envisioned product is a black-box from the perspective of other elements. Interactions with it are "conceptual": information and acts that are directed into or out of the product. (how much is the OpEnv specification just descriptive of relationships? Interactions should be expressed in processes/scenarios only, which are inherently limited/anecdotal; does the OpEnv need to be a mirror of the Rqmts or more statically descriptive? Scenario describes a process that user or system performs with effect on the product.)

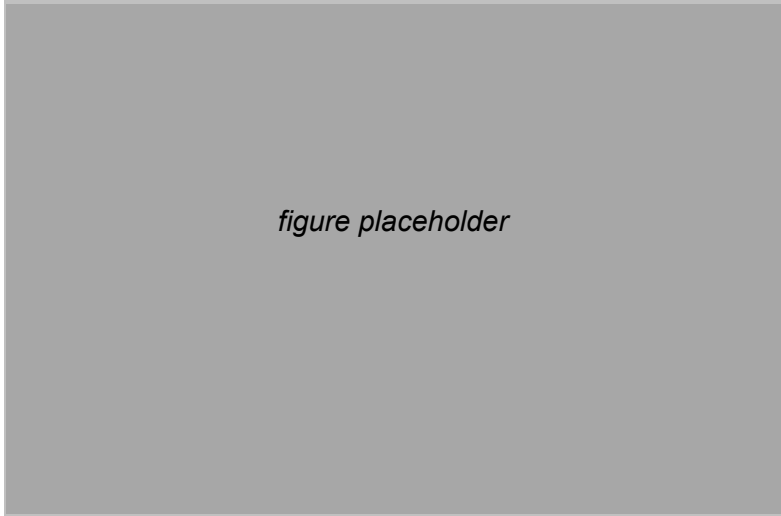


Figure 2.3-1. Conceptual Structure of the Operational Environment Model

Specifying an Entity

This model consists of a set of entity descriptions, each consisting of three elements:

- (information) what is the accessible (product-relevant) information content of the entity?
- (behavior) what is the behavior of the entity that can cause its information content to change (e.g., as a basis for simulating an approximation of the entity's behavior)
- (interfaces) what are the associated conventions for interacting with the entity in order to (1) obtain or share information and (2) monitor or effect changes in information content?

(what dev needs to interact in ecosystem, messages (actions/info) coming in & messages (actions/info) going out)

An Information Element

An information element specifies the product-relevant information that is associated with each relevant entity. Information is expressed in terms of the data form in which each entity defines it. Information is specified in terms of a set of discrete or aggregate values encoded in a digital or analog representation.

Environment content is defined, ideally in the form of an active semantic data model², as an abstract information space. This is the definition on which the environment information content partition of each entity is defined. All such content must be traceable either directly to environment content or as a transformation of such content. A transformation can convert between analog and digital representations, derive composite values, convert between units of measure, derive interpolated or smoothed values, time-sampled values, etc. Environment content itself may have associated transformations that define how values change over time or in response to other data changing.

² provide reference or describe separately if no ref is suitable

{define info model of environmental monitored/controlled variables; models for behavior (how values chg)} {relevant info content that changes naturally or due to ecosystem effects} {also info assoc (sent/received) with related systems}

The element specifies the data that each of the accessible entities in the product's ecosystem provides.

A Behavior Element

A behavior element specifies the observable behavior that an entity is expected to exhibit. Behavior is specified as discrete events that an entity initiates and is expressed in terms of enabling conditions, potential effects, and criteria for which effects result. Types of behavior include initiating action that modifies the environment's information content, transforming and/or storing information obtained from the environment or other entity, or transferring information to another entity.

Relevant Operation of the Customer Enterprise

(products, processes, user roles)

Environment Behavior

Edge Device Behavior

User Interface Behavior

System Interface Behavior

An Interfaces Element

An interfaces element specifies the information and actions supported by each ecosystem-associated entity. Each entity is characterized as being one of three types: edge, user, or system. An information space is associated with each entity that defines access to corresponding data that it shares. Information and action interactions with an entity is specified in a form appropriate to its type. An edge entity is a logical device that has direct access to the environment, monitoring or effecting associated properties. A user entity characterized according to specified “roles” that define an associated set of capabilities to obtain or provide information or initiate actions related to the environment, other entities, or itself. A system entity is a source of information or effector of actions.

Edge Device Entities

An edge device entity interface is specified in terms of data that it can convey between the physical world and computation, including translations between analog and digital representations. A device may operate independently or as an agent of the product. A device’s intrinsic interface may be encapsulated and enhanced within a software-defined abstract interface.

User Interface Entities

A user entity interface is specified in terms of a user role that has associated information and action capabilities. It is a composite of a physical device interface and abstract mechanisms required to enact supported user-system interactions.

System Interface Entities

A system entity interface is specified in the guise of one or more associated logical devices.