Architecture of Complex Systems

Glossary

An **architect** is someone who manages the early stages of a design and who makes a set of decisions that have an outsized impact on the metrics for the system.

Architecture is a representation of entities organized in a way that supports reasoning about the entities and describes behaviors and relationships amongst the entities.

A **concept** is a product or system vision, idea, notion, or mental image that maps function to form in brief. It is a simplification of the system architecture that allows for high-level reasoning.

Connectivity is the degree to which making a given decision influences other decisions.

Decomposition is the breaking up of a larger entity into smaller entities.

Design Structure Matrix (DSM) is a simple, compact representation of a system or project in the form of a square matrix. It is the equivalent of an adjacency matrix in graph theory and is used in systems engineering and project management to model the structure of complex systems or processes.

Emergence refers to what appears, materializes, or surfaces when a system operates. Obtaining the desired emergence is why systems are built. Understanding emergence is the goal – and the art – of systems thinking.

Emergency is an undesirable type of emergence. For example, cars can lose traction and spin. A natural example of emergence was Hurricane Katrina in 2005.

Form is what a system is. It is the physical or informational embodiment that exists or has the potential to exist.

Formal relationships are relationships that exist; they are static. These relationships exist stably for some period of time and often include a connection or a geometric relationship.

Functional analysis is the process of evaluating which functions should be part of the system.

Functional architecture is the mapping between the functions and the operands of the system.

Function is what a system does: its actions, outcomes, or outputs. Function emerges when a system comes together.

Functional relationships do something; they are dynamic in nature. For example, the heart exchanges blood with the lungs.

"Ilities" are other attributes of operation that emerge from a system, such as reliability, maintainability, operability, safety, and robustness. How robustly and reliably does a car run? How reliably does a clock keep time?

Interface is a point where two systems meet and interact.

Object-Process Methodology (OPM) is used to document systems with a very confined and minimal set of conceptual building blocks (objects and processes) to represent systems in an effort to reduce complexity. In contrast to SysML, OPM was initially designed to support modeling of general purpose systems, thus it has no inherent "software-oriented" language semantics.

An **operand** is something that is changed by a process. It is the thing whose change in state is associated with value.

Performance is how well a system operates or executes its function(s). For example, how quickly a car transports people; how accurately an hourglass keeps time.

Primary value function is the purpose for which the system was originally conceived and built, which continues to deliver the most value.

Process architecture is the mapping of the interrelationships between the processes in the system.

Process is a pattern of transformation over time applied to one or more objects. Generally, processes are creating, destroying, or changing something.

Secondary value functions are functions that are either necessary for the system's operation (but not the primary value function) or functions which are optional.

Sensitivity is a measure of the impact on metrics caused by a given decision.

A **solution-neutral function** explains a function without specifying the solution that allows one to achieve that function. Solution neutral is not an absolute property, but rather a spectrum of more solution neutral to less solution.

Structure is how objects of form are connected.

A **system** is a set of entities and their relationships, whose functionality is greater than the sum of individual entities.

System architecture is the embodiment of concept, the allocation of physical/informational function to the elements of form, and the definition of relationships among the elements and with the surrounding context.

System boundary divides what the architect and the firm controls (the system) from the accompanying systems and surrounding context. The boundary makes it clear what is in "the system" and what is outside it.

System thinking is the understanding of a system by examining the linkages and interactions between the components that comprise the entirety of that defined system.

Systems Modeling Language (SysML) is a branch of the Unified Modeling Language (UML) for systems engineering applications. It is a graphical modeling language with defined diagrams and syntax.

Two down, one up is the principle that the best information about how to cluster or group the system one level down from the reference is present in the structure and in the interactions two levels down. Therefore, we need to drill down to Level Two to better identify the Level One decomposition.

Value-related operand is the most important operand, the reason the system was created.

Whole product system is the product system plus accompanying systems outside the direct control of the firm that interact together with the architecture to deliver value to the customer.