Analysis of Form

Lecture 04, v04

John G. Artus

BSEE

MSSE

INCOSE ESEP

About This Courseware

- The majority of the material presented in this course is sourced from the textbook “System Architecture” by Edward Crawley, Bruce Cameron, and Daniel Selva
- I, John Artus, make no claim of ownership of the material sourced from this textbook
- I, John Artus, am using the material sourced from this textbook, and other indicated sources, as content for this courseware for educational purposes only
- This courseware lecture material has been sourced, interpreted, assembled, formatted, and copyrighted by John G. Artus for use in this educational context
- Anyone may freely access, and reuse this material in an educational context provided the copyright owner, John G. Artus, is recognized as the interpreter, assembler, and formatter of the source material used in the generation of this courseware, and provided that Edward Crawley, Bruce Cameron, and Daniel Selva are recognized as authors of the textbook “System Architecture” from which the majority of the content of this courseware has been sourced

Definition of Form

Form is the physical or informational embodiment of a system that exists or has the potential for stable, unconditional existence, for some period of time, and is instrumental in the execution of function

- Form includes the entities of form and the formal relationships among the entities
- Form exists prior to the execution of function
- Form is a product/system attribute

Form In Architecture

- It is surprisingly difficult to separate form from function
- Form is what has been or is eventually implemented (built, written, composed, manufactured, or assembled)
- Form is about existence
- The first test for form is that it exists
  - Form is what the system is
  - It is the concrete and often visible manifestation of the system
  - The clause “for some period of time” is included in the definition to enable us to discuss the form of things that
    - Have existed in the past (analysis of historical systems)
    - Will exist in the future (analysis of system designs)
- The second test for form is that it is instrumental in the execution of function
- For function to occur, there usually needs to be some instrument to facilitate the function, or to “carry” the function
- Form is what is eventually operated to deliver function and value

Form In Architecture (continued)

- Form includes the sum of the system entities, which are the elements, or chunks of form
- These are the elements of the whole
  - This is the whole-part relationship for form
- Form also includes the relationships among the entities of form (their arrangement)
  - This is often called the structure
- In total, form is the entities of form plus the structure
- Form is a system attribute
  - It is an independent way in which to view or characterize the system
- Form is also part of the solution that the architect proposes

Objects Analytically Represent Form

- Systems can be represented with only two classes of abstractions
  - Objects
  - Processes
- Objects and processes faithfully describe the system’s structure, function, and behavior in a single, coherent model in virtually any domain
- An object is that which has the potential for stable, unconditional existence for some period of time
- Recall that form is the physical or informational embodiment of a system that exists or has the potential for stable, unconditional existence, for some period of time, and is instrumental in the execution of function
- Thus, objects could be used to represent form
- The definition of form contains the additional criteria that the form must be instrumental of function and must exist before function

Objects Analytically Represent Form (continued)

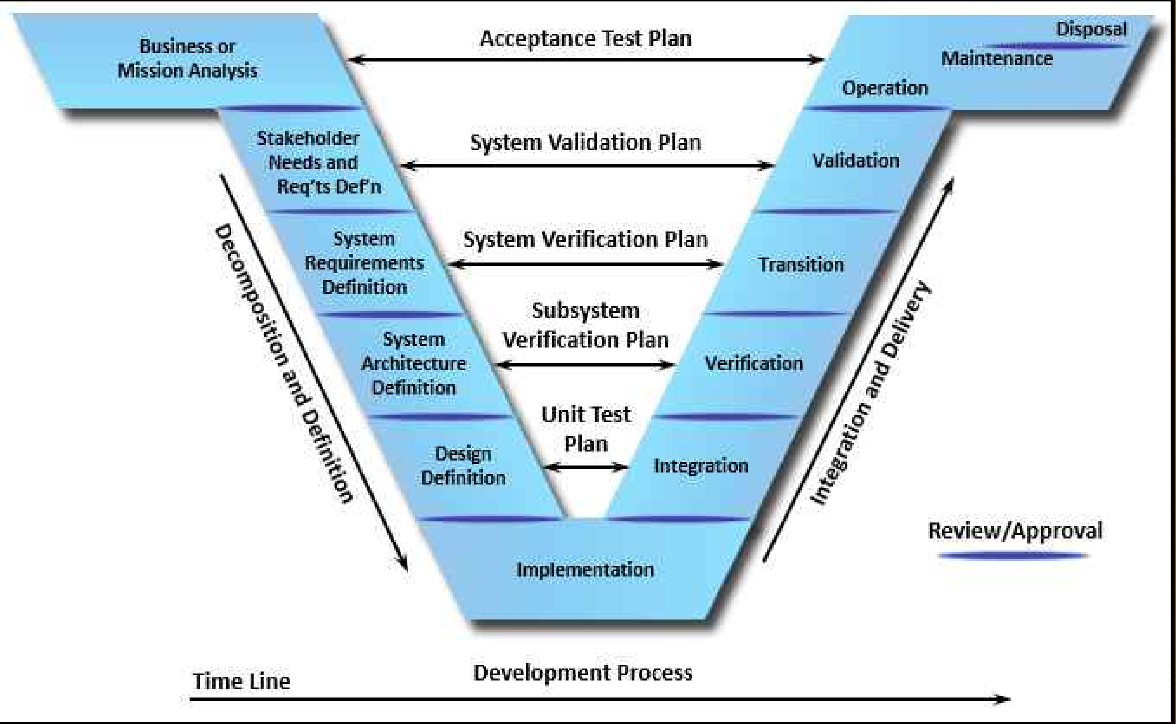
- Objects can also be informational
  - As in anything that can be comprehended intellectually
  - Which implies that it exists
- Informational objects include ideas, thoughts, arguments, instructions, conditions, and data
- Domains such as control systems, software, mathematics, and policy make extensive use of informational objects
- One of the defining characteristics of objects is that they have characteristics or attributes that describe the object
  - An object can have physical, electrical, or logical attributes
  - Some of these attributes can be considered as states, in which an object can exist for some duration of time
- A process can change an object's state
  - The combination of all states describes the possible configuration of state of the object

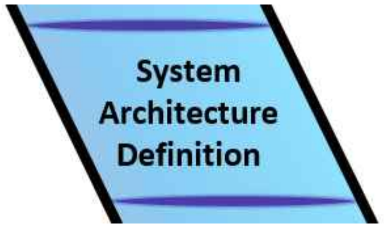
Decomposition of Form

- Decomposition of form comes naturally to architects
  - Form is concrete
  - Form decomposes easily
  - The aggregation (building up) of form traces (follows) the decomposition in a simple way

In the larger Systems Engineering VEE, we continue decomposing the system definition until we reach the Implementation Phase

However, in Architecture Definition, we decompose the system to identify the components, and then aggregate them together

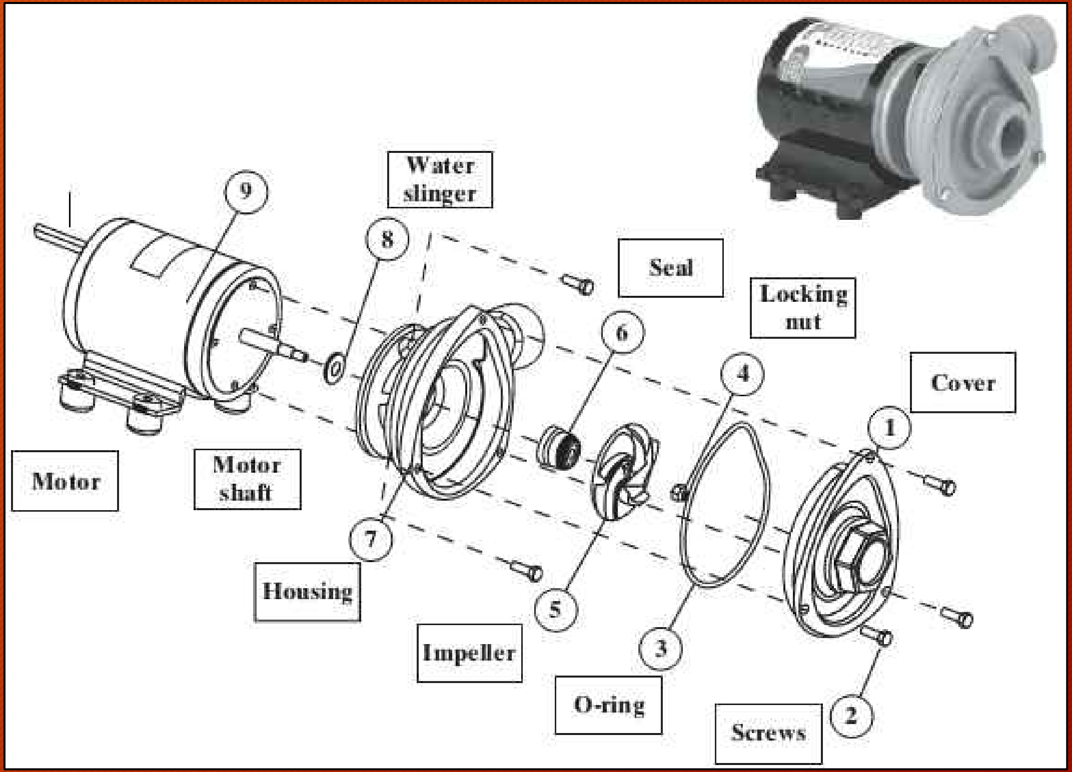




- Decompose the system definition down to lowest meaningful component
- Then aggregate the components into the physical architecture to define their form and formal relationships
- The Design Definition phase takes the decomposition further to the lowest component in order to produce manufacturing specifications

Simple System Example – Centrifugal Pump

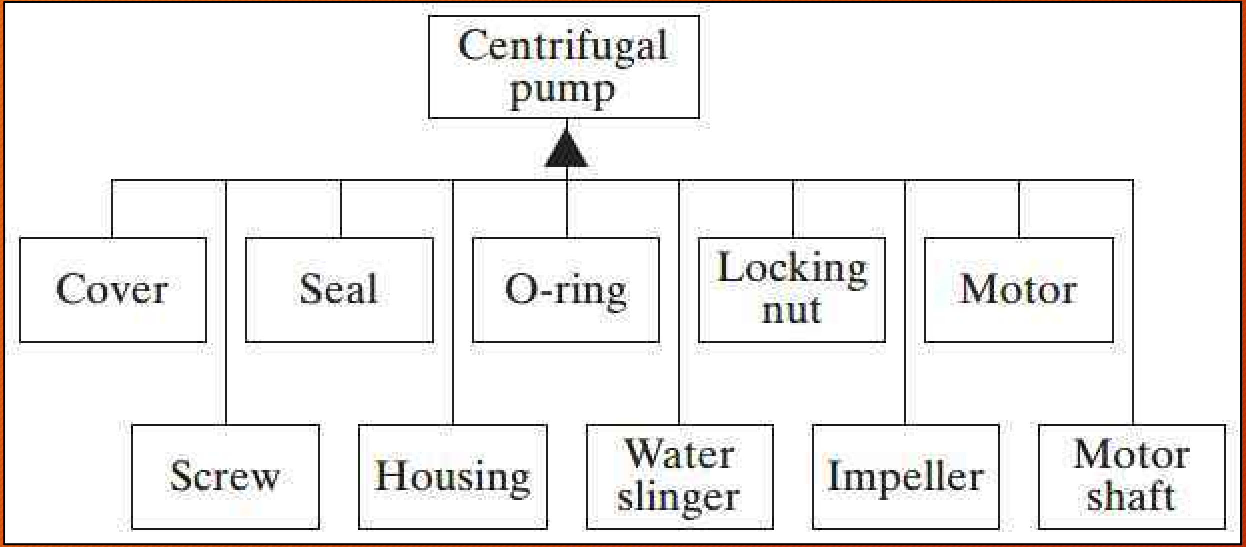
- A centrifugal pump works by increasing the energy of the fluid that passes through it
- The turning impeller does the work on the flow
- Depending on the design of the pump, the increase in energy can be delivered either as an increase in velocity of the outlet flow or as an increase in static pressure
- The flow enters the pump on the axis of rotation of the impeller through the hole in the center of the cover
- The impeller turns the flow radially and does work on the flow, increasing its velocity internal to the pump
- The housing contains a passage that slows the velocity of the water, or diffuses the flow
- The newly gained energy is converted from kinetic energy into higher pressure
- Mass is conserved, so the same amount of fluid exits the pump as enters it
- Energy is conserved, so the electrical power going into the electrical motor is reflected in the increased pressure of the outlet mass flow



The choice of this trivial example system is for the purpose of explaining and understanding the methods of analyzing form presented in this course

Defining the System

- Examine the system
- Create an abstraction of form that
  - Represents the abstraction for the entire system
  - Conveys the important information
  - Implies a boundary of the system
  - Separates the system from the context



The Object-Process Methodology (OPM) can be used to conceptualize system form and function. OPM was created by Dr. Dov Dori at MIT. OPM includes a clear and concise set of symbols that form a language enabling the expression of the system's building blocks and how they relate to each other.

This is an OPM representation of a hierarchical decomposition that extends one level down. There are other graphical languages that could be used to represent system design (such as SysML). OPM is appropriate for this setting of conceptualizing a system definition.

No formal structure is shown here, other than decomposition.

No functional interaction is implied.

▲ Represents aggregation of Tier 1 elements into the Tier 0 element

Identifying the Entities of Form

- Create a parts list as the initial set of abstractions
- Decompose some elements further if necessary
  - It is not uncommon to find entities that can be combined simply for convenience
  - Because they are integrally part of delivering a single function
  - Combine or eliminate elements where possible
- Use hierarchy to identify the most important elements of form
  - Identify the five key parts of the pump
    - The cover and housing contain the fluid and provide interfaces
    - The impeller does the work on the fluid
    - The motor drives the shaft and supports the housing
    - The motor shaft drives the impeller
  - Identify parts in the middle rank of lesser importance
    - The o-ring prevents water escaping the system along the motor shaft
    - The seal prevents water escaping the system at housing attachment points
  - Identify the least important elements
    - Screws
- Based on this information, first try to reason about the five most important objects, to understand them thoroughly before including the lesser important elements

Hierarchy is a system in which layers are ranked one above the other because they have more scope, importance, performance, responsibility, or function

Centrifugal Pump Parts	Level Of Importance
Cover, Impeller, Housing, Motor Shaft, Motor	Most important parts
O-ring, Seal, Water Slinger	Moderately important parts
Screws (class), Locking Nut	Least important parts

Identifying the Entities of Form for Pump Example

- The “parts list” for a system is a useful place to start creating abstractions
  - Use the parts list as the set of abstractions defining elements of form
- A few exceptions for the pump
  - Motor
    - Has a non-rotating element and a rotating motor shaft
    - Some elements connect to the motor and some connect to the shaft
    - This difference is important
    - Calling all of this “the motor” would hide too much information
    - One can guess that the shaft also is involved with different functions than the motor
    - Therefore, we create two elements to describe the motor and the motor shaft
  - Screws
    - In the decomposition we identified only one abstraction called “screw”
    - But, there are five screws
    - So, create a class called “Screw”

There could be several ways in which a system can be decomposed, depending on the objective of the process

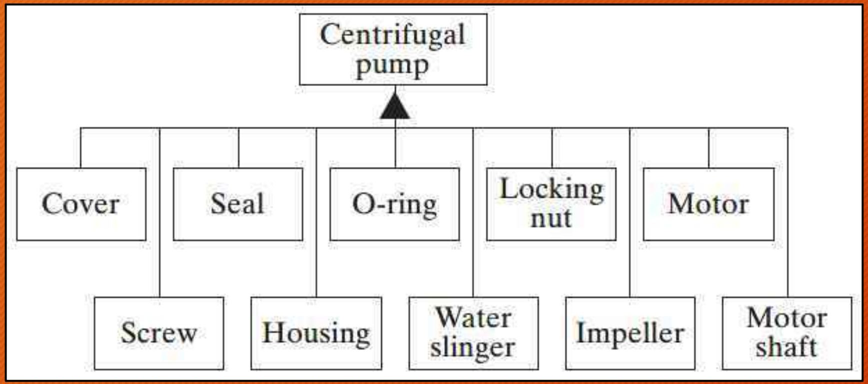
Example:

- The only function of the locking nut is to secure the impeller to the shaft
- So, the locking nut could be combined with the impeller

Creating Multiple Levels of Decomposition

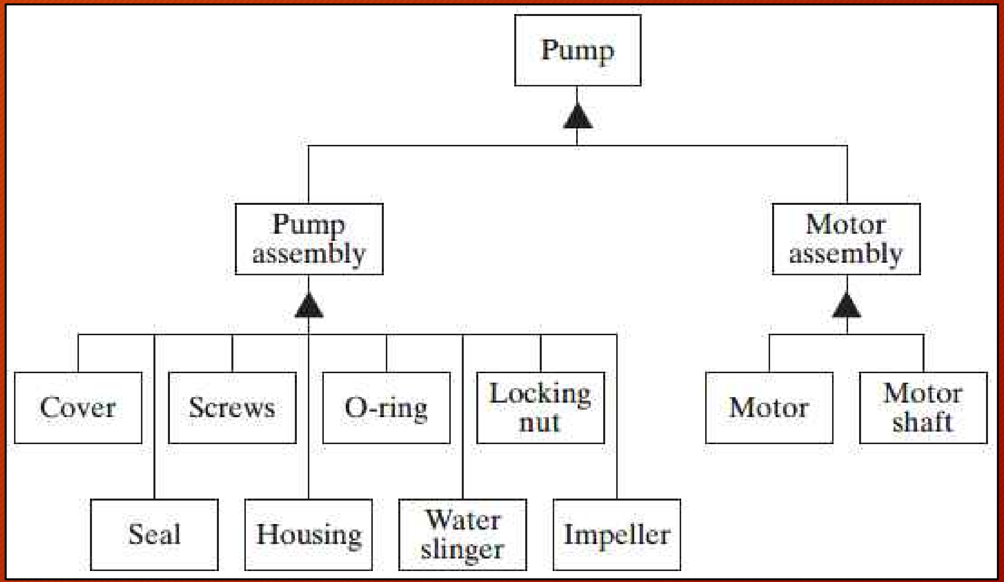
- Even though this pump has only ten entities of form, it is possible to convert its representation from a one-level decomposition to a two-level decomposition
  - “Assemblies” are created to represent groupings of related elements
  - Other terms can be used to represent groupings

One Level of Decomposition → Two Levels of Decomposition



Level 0

Level 1



Level 0

Level 1

Level 2

Analysis of Formal Relationships

- Relationships of form are sometimes called formal relationships or (more commonly) structure
  - Structure is the set of formal relationships among the elements of form of a system
  - Structure is not conveyed by hierarchical decomposition alone
  - Structure is additional information
- If you are merely given a random set of mechanical parts, this is not sufficient information to know how to assemble the parts
  - What is missing is information about the structure
  - Structure shows where the elements of form are located and how they are connected
- After elements of form, formal relationships are the next most tangible aspects of a system
- Formal relationships (structure) are the relationships between objects of form
  - They have the potential for stable, unconditional existence for some duration of time
  - They may be instrumental in the execution of functional interactions

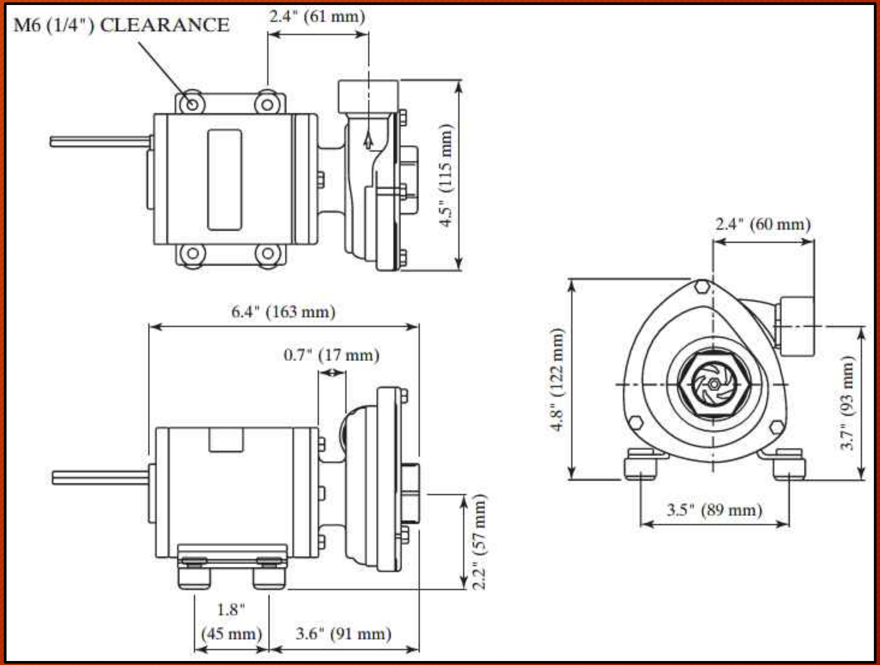
Analysis of Formal Relationships (continued)

- Structure is often instrumental in functional interactions
  - If there are functional relationships among the elements of form, in most cases there must also be a carrier of these functional interactions
  - Formal relationships often carry the functional interactions
  - Given: A supplies electrical power to B (a functional interaction)
    - Hence, there may be a connecting wire (the structure) that serves as the carrier of the functional interaction
- Relation of structure to emergence of function
  - Functional interactions are crucial to emergence
  - Functional interaction is often enabled by formal structure
  - Therefore, understanding structure is essential to understanding the system and the emergence of function
- Structure is the set of relationships that exist and do not require continuous change, exchange, or any similar process that happens along a timeline
  - Structure could change over discrete intervals of time, as in phases of operation, but not continuously
  - At any instant of time, the formal relationships that constitute structure just "exist"
  - Like all of form, formal relationships are about what exists, not what occurs
  - In our pump, the motor is connected to the housing — a connection that simply "exists"
  - This concept may be challenged by the introduction of technological advances that allow "dynamic structure" – structure that changes form over time

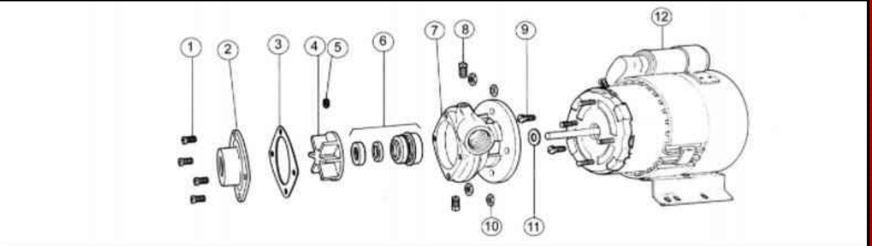
Analysis of Formal Relationships (continued)

- Formal relationships can be changed
  - Elements can be connected, unconnected, or moved
  - Formal relationships often implicitly represent things that have happened in the past
- The two main types of structural relationships — both are needed to fully understand formal relationships
  - Spatial/topological relationships
    - A layout drawing shows where components are located and how they are spatially related to each other
    - The connection of components cannot be fully understood from a layout diagram alone
  - Connections
    - A connectivity diagram shows how components are connected together
    - The spatial relationship of components cannot be fully understood from a connectivity diagram alone

Spatial/topological View



Connectivity View



Pump No.	1 Screw 4 Reqd	2 Cover 1 Reqd	3[1] Gasket 1 Reqd	4[1] Impeller 1 Reqd	5 Set Screw 1 Reqd	6[1] Seal Assy. 1 Reqd	7 Body 1 Reqd	8 Plug 2 Reqd	9 Screw 2 Reqd	10 Hex. Nut 4 Reqd	11 Slinger 1 Reqd	12 Motor 1 Reqd	Repair Kit[1]
172B	5385	7375	7020	7678	6079	32458	7456	5395	7424	7686	6651	—	11071

Spatial/Topological Formal Relationships

- Spatial/topological relationships explain where things are: the location or placement of the objects of form
  - These relationships imply only location and placement, not the ability to transmit anything between the objects
  - In physical systems, spatial and topological relationships are largely issues of geometry, but in information systems, one must also consider the more abstract issues of space and topology
- Spatial relationships capture absolute or relative location or orientation, such as
  - Above and below
  - Ahead and behind
  - Left and right
  - All of these relationships convey relative location
  - Absolute location may be given with some reference, such as an Earth geodesic reference or a local reference system
- Topology is about placement and suggests relationships, such as
  - Within
  - Contained in
  - Surrounded by
  - Overlapping
  - Adjacent to
  - Touching
  - Outside of
  - Encircling
  - Aligned and concentric
  - Near to and far from
  - etc.

Spatial/Topological Formal Relationships (continued)

- Differences between spatial and topological relationships
  - In practice, the differences are very subtle
  - Both can be considered part of one set of spatial/topological relationships
- Spatial/topological relationships are often very important
  - Example: Optical elements have to be aligned to work properly
- The procedure for identifying the spatial/topological relationships
  - Examine each pairwise set of objects of form
  - Ask whether there is an important spatial/topological relationship between these two objects
  - Thinking holistically, we recognize that there is a spatial relationship between every pair of objects
- To understand the functional interaction and emergence related to structure, the key question is the following:
  - Ask whether this spatial or topological relationship is key to some important functional interaction or to the successful emergence of function and performance

Spatial/Topological Relationships for Pump

- The matrix below presents the spatial/topological relationships for the simplified pump
- Both the forward and backwards relationships are shown
- The cover
  - Touches the housing
  - Is aligned with the motor shaft
  - Is close to the impeller
- The housing
  - Surrounds the impeller
- The impeller
  - Is within the housing
- The entries in this matrix do not imply direction or causality
- In particular, for structure the forward and backwards relationships must have the same semantic meaning

Parts List	Cover	Impeller	Housing	Motor	Motor Shaft
Cover	X	Close to	Touch		Aligned with
Impeller	Close to	X	Surrounds		Touch/ is encircled by
Housing	Touch	Within	X	Touch	Is encircled by/aligned
Motor			Touch	X	Within/touches
Motor Shaft	Aligned with	Touch/ encircles	Encircles/ aligned	Surrounds/ touches	X

Connectivity Formal Relationships

- Connectivity and spatial/topological relationships are quite different in principle
  - Two elements can be adjacent but not connected (two houses in a neighborhood)
  - Two elements can have unspecified location but be connected (a computer and a server)
- The formal relationship connectivity addresses
  - What is connected, linked, or joined to what
- In network systems, connectivity is the most important formal relationships to understand
  - It explicitly creates the ability to transfer or exchange something between the connected objects of form
- Connectivity relationships are often instruments of functional interaction
  - They directly support the emergence of function and performance
- In practice, the difference between spatial/topological structure and connectivity structure is usually evident if the explicit formal connection
- Connectivity can also refer to electrical, chemical, thermal, or biological connections

Connectivity Formal Relationships (continued)

- In mechanical systems, because things must be adjacent to be connected, the spatial/topological structure and connectivity structure tend to be less distinct
- Connectivity relationships actually encode information about how the objects were implemented: assembled, manufactured, coded, or written
- If two things are connected now, there was a process of connecting them in the past
  - But the connection currently exists, and therefore it is a relationship of form
- There are a few classes of functional interactions that do not require an explicit formal connection
  - Electrical forces and the electromagnetic spectrum, which includes visible and radio frequencies
  - Heat transfer through black body radiation
  - Gravitational interactions do not require an explicit connection
- The procedure used to determine connectivity relationships is similar to that used to identify spatial/topological relationships
  - Examine each pairwise set of objects of form
  - Ask whether this connectivity relationship is key to some important functional interaction or to the successful emergence of function and performance

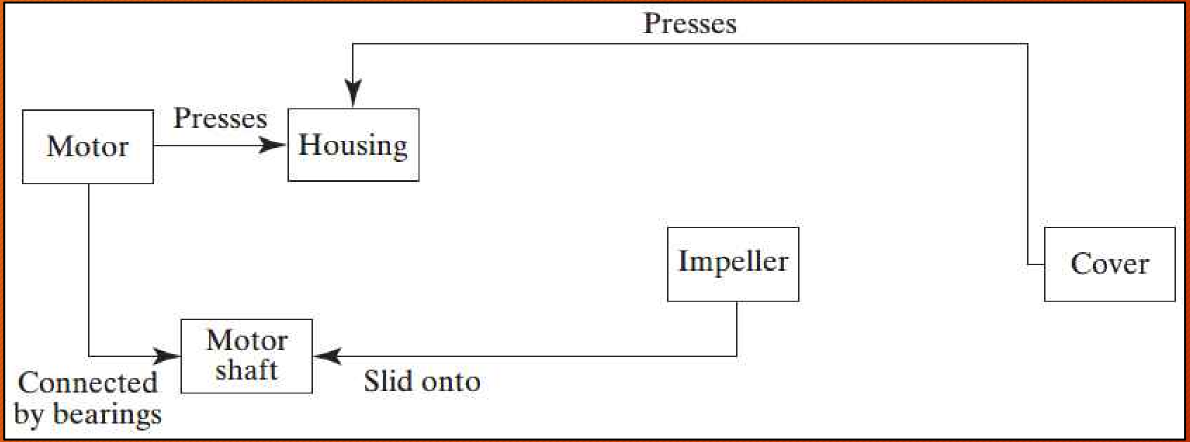
Connectivity Formal Relationships (continued)

- Connectivity (touching) is usually more evident and less subject to interpretation than spatial/topological (nearness) structure
- In identifying connections
  - Consider whether the connector itself must be modeled as an element of form
  - Or, conversely, can be abstracted away
  - If practical, abstract away the connectors in an attempt to minimize the number of objects that are being considered
    - An exception to this rule is the connector at a system interface
    - Such interfaces are sufficiently important to the system, and are a common source of problems
    - Modeling a system interface as a separate object is often appropriate

Connectivity Structure for Centrifugal Pump

- The connectivity structure for the simplified pump is shown here
  - The cover “presses” against the housing, which is a kind of mechanical connection
    - The devices that cause the pressing are the screws
    - The screws have been abstracted out of this simplified model
  - The impeller has a press fit, due to its being force-fit (pressed) onto the motor shaft
  - The relationship between the motor shaft and the motor is quite complex
    - The shaft is contained within the motor and is supported on bearings that allow rotation
    - It makes most sense to label the connectivity relationship with the type of the connector: bearings

Simplified Pump Connectivity Structure



Simplified Pump Connectivity Matrix

Parts List	Cover	Impeller	Housing	Motor	Motor Shaft
Cover	X		Presses		
Impeller		X			Had slid onto
Housing	Presses		X	Presses	
Motor			Presses	X	Connected by bearings
Motor Shaft		Slid onto		Connected by bearings	X

Difference Between Spatial/Topological and Connectivity

- Comparison of the connectivity structure with the spatial/topologic structure shows fewer connections than spatial/topologic relationships
- The spatial/topological relationships that are associated with touching reappear as a connectivity relationship
- However, those associated with pure topology (such as alignment, within, and encircles) are not reflected in the connectivity matrix
- In mechanical systems, the distinction between the spatial relationship "touching" and the connectivity "pressing" is subtle
- One implies only location, and the other also conveys the ability to transmit load
- If we had chosen a mechanical example in which the two objects were glued together, then the difference between touching (spatial) and glued (connectivity) would be more evident

Combined Spatial/Topological and Connectivity Structure

- As a practical matter, the spatial/topological and connectivity relationships are often combined into one representation of structure
- The matrix below shows a summary of the structure of the system, where the details have been abstracted to just the existence of a spatial (S) or a connectivity relationship (C) among the elements

Simplified Combined Spatial/Topological and Connectivity Structure

Object List	Cover	Impeller	Housing	Motor	Motor Shaft
Cover	X	S	SC		S
Impeller	S	X	S		SC
Housing	SC	S	X	SC	S
Motor			SC	X	SC
Motor Shaft	S	SC	S	SC	X

Other Formal Relationships

- There are other types of formal relationships that simply exist
- The procedure for identifying these is the same
  - Examine pairwise sets of objects
  - Determine whether there is a relationship of these types that has an important role in enabling functional interaction and emergence
- Address
  - The address of something is an indication of where the object can be found
  - It is an encoded form of the spatial location
    - It is common enough to be considered a separate type of formal relationship
    - Because addresses can also be virtual
  - A shared address is a connection through which data can be exchanged
- Sequence
  - The static sequence of entities is a type of formal relationship
  - If we know that A is always after B, then this establishes a formal relationship

Formal Context

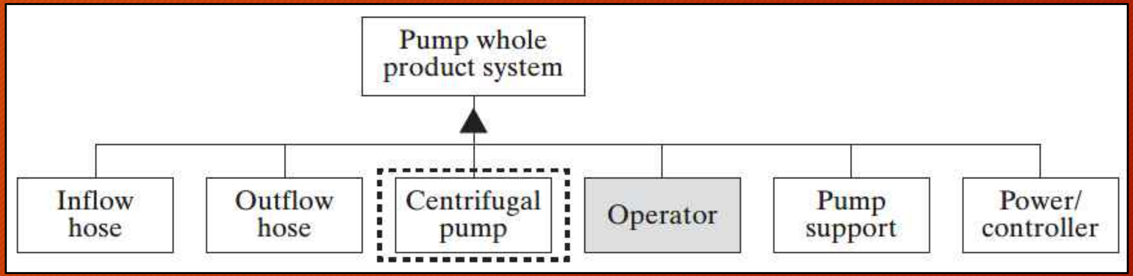
- Another way to analyze and understand systems is by taking increasingly broader or more holistic views of the system and its context
  - Whole Product System view
  - Use Context view
- Whole Product System view
  - Expand outward from the product/system
  - The first level of context encountered includes the other objects that are not part of the product/system but are essential for the system to deliver value
  - These are called the accompanying systems
  - The sum of the product/system and the accompanying systems is called the whole product system
  - The product/system is separated from the accompanying systems by the system boundary
- Accompanying Systems
  - It is vital for the architect to understand and model the accompanying systems, because they may play an important role in emergence
  - They must be present, connected, and working for the product/system to deliver value
  - There must be well-defined interfaces to accompanying systems at the system boundary

Formal Context (continued)

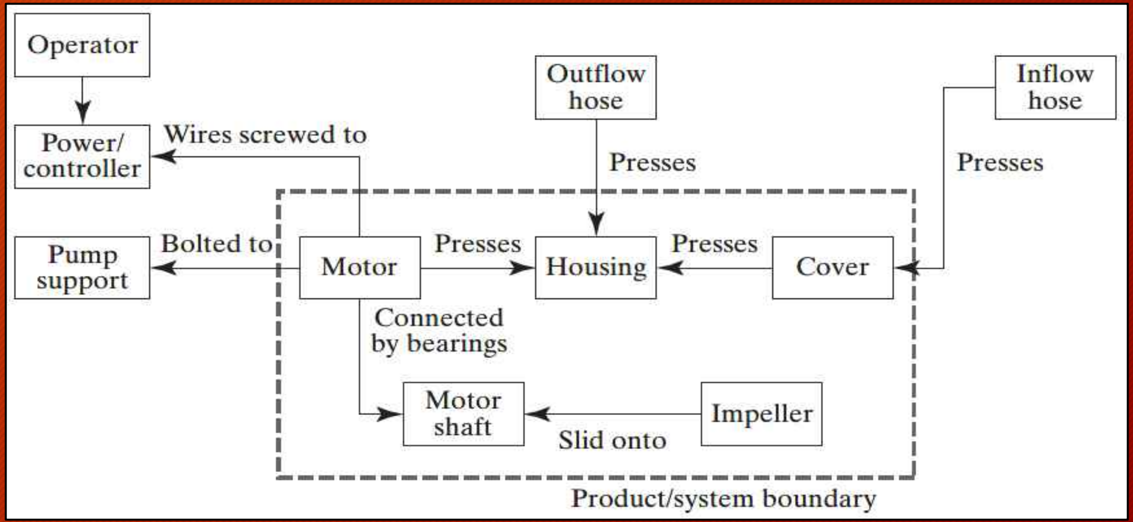
- Use Context view
  - Most often there is an operator involved in delivering value
  - Include the operator in diagrams to identify important human interactions with the system
- System Boundary
  - Look holistically at the objects of form that are near the system or are connected to it
  - Ask whether these are essential to the delivery of product/system value
    - The abstractions of form are created for the elements of the accompanying systems
    - The formal structure between the system and the accompanying systems is identified
  - The product/system boundary helps reduce ambiguity and explicitly identifies interfaces

- For the pump, there must be elements that transport fluid to and away from the pump: the inflow and outflow hoses
- The entire assembly must be structurally supported by a pump support, and the motor must be powered and controlled
- This diagram shows the pump whole product system, including these four accompanying systems

- No structural information is shown in the decomposition view shown above
- This is best presented in diagrams like the connectivity structure shown here
- Now one can clearly see the connection between the principal objects of form of the pump and the accompanying systems, and the interfaces at the system boundary



System Boundary Diagram



Use Context

- To find the next level of context, the use context, expand one more step outward
  - The whole product system fits within this use context
  - This includes the other objects that are normally present when the whole product system operates but are not necessary for it to deliver value
  - The use context is important because it informs the function of the product/system
  - It gives place to the whole product system
  - It gives us information on the environment in which the system operates and informs design
- The architect may have to speculate on the use context
  - The system might have different requirements based on these use contexts
  - The use context is rarely documented in system descriptions
  - Ask what elements of context that are not already in the whole product system help form an understanding of use and design

Use Context (continued)

- Responsibilities of the System Architect
  - The need for a clear understanding of the whole product system, informed by the use context, is driven by the accountability and responsibility of the architect
  - Even though the architect is responsible only for the product/system, it is inevitable that the architect will be held accountable for the function of the whole product system
  - If it does not work, the architect will likely be accountable regardless
  - Just as it is important for an architect to understand about two levels down in decomposition, it is important to understand about two levels out in context—the whole product system and the use context

Applying this idea to the centrifugal pump

- The available drawings do not contain any hint about use context
- It is necessary to speculate about the use context
- The pump might be an industrial sump pump for salt water removal or part of a home dishwasher
- The inflow hose for the pump might be supplied by a third party
- If the hose does not connect well to the pump, it is likely that the System Architect involved in designing the Whole System and Usage will be accountable for the poorly functioning inflow interface

References

1. Crawley, Edward; Cameron, Bruce; Selva, Daniel (2016). *System Architecture: Strategy and Product Development for Complex Systems*, Pearson Higher Education Inc, Hoboken, NJ
2. Dori, Dov (2002). *Object-Process Methodology*. Springer-Verlag, Berlin, Heidelberg, Germany