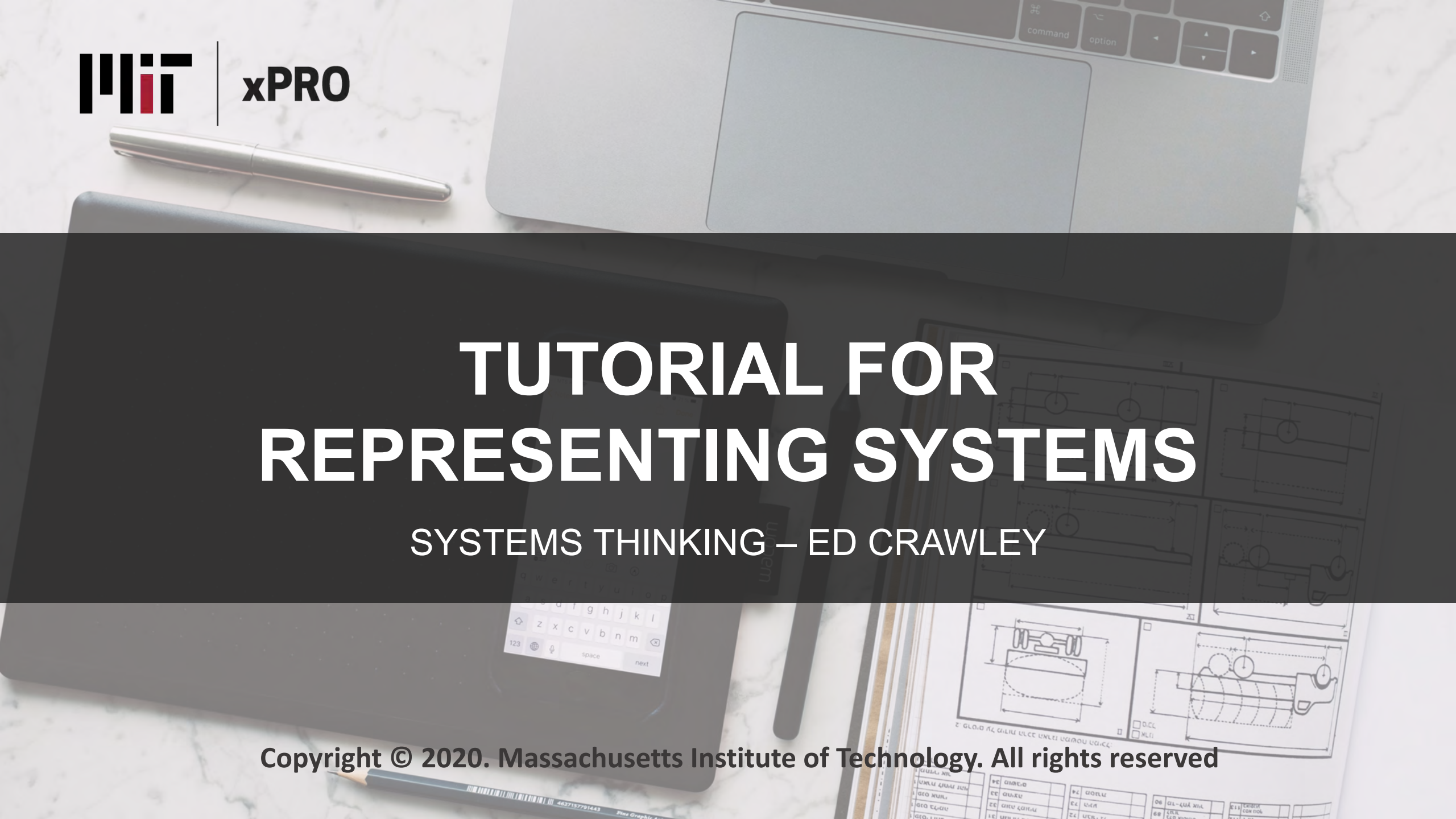


TUTORIAL FOR REPRESENTING SYSTEMS

SYSTEMS THINKING – ED CRAWLEY

About This Tutorial

01 This is not a course in modeling, model-based design, or model-based engineering

02 This is a course on system thinking and system understanding

03 But even so, it is useful to have some informal sketches or icons that allow us to visualize and discuss the important issues

04 These are not likely to be preserved, and do not have the rigid semantics and syntax that OPM or SysML have. They are intended to support your reasoning

Representation of a System

Benefit: get a boater to work in a harbor

Transports Boater

Motorboat system

Transports Boater

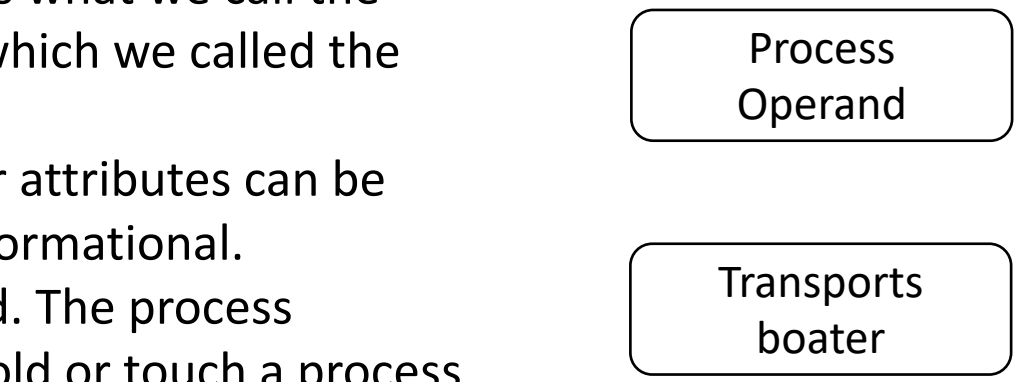
Motorboat system

Cost: purchase and operations

Representing Systems: Function

- Functions is what the system DOES. It's the activities, operations and transformations that cause or contribute to performance. Function is what actually happens when a system operates.
- A function contains two parts: The change itself, which is what we call the process; and the thing that is changed or operated on, which we called the operand.
- The OPERAND is a thing that exists, and whose states or attributes can be changed by a process. An operand can be physical or informational.
- A PROCESS is a pattern of transformation of the operand. The process creates, changes or destroys the operand. You cannot hold or touch a process – it is fleeting.

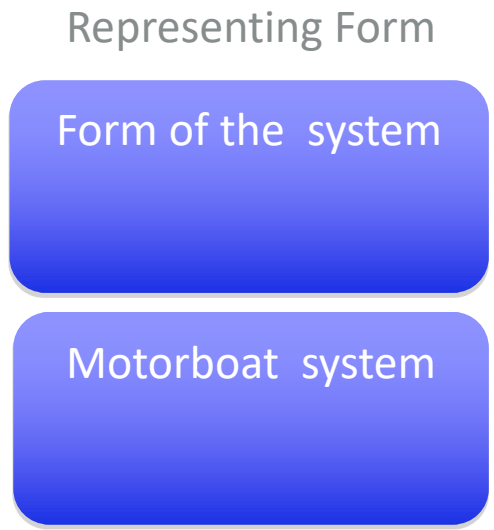
Representing Function



- We represent a function by a white rectangle with a black line. The rectangle contains a word or short phrase for the process, and a second line for the operand.
- The generic example is "process operand" and the worked example is "transport boater."

Representing Systems: Form

- Form is different than function. Form is what the system IS. It's the physical or informational embodiment of the system. It involves the material, configuration, text, and instructions of the system.
- If it was implemented, it is probably form.
- Form has many names. Mechanisms, molecules, facilities, circuits, building blocks, and infrastructure are all examples of physical form. Informational form includes poetry, literature, code, and instructions. People are part of the form of an organization.
- Form is the instrument of function, and must be present for the function to be delivered.

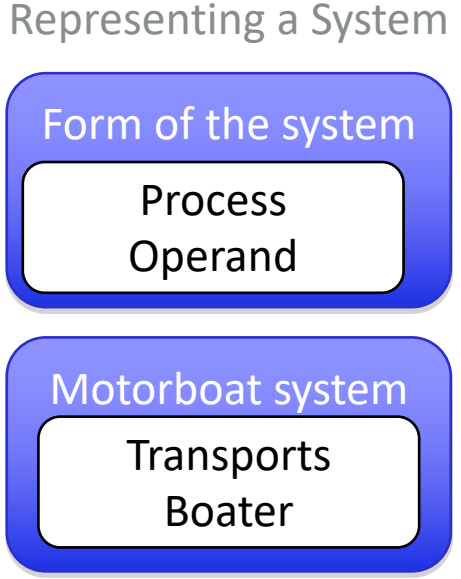


- We represent form by a shaded rectangle, with the name in the upper 1/3 (leaving room for the associated function).
- The generic example is "form of the system" and the worked example is "Motorboat system."

Representing Systems: System

- A system has form and function. The form enables the function.
- Form is what the system is, function is what the system does

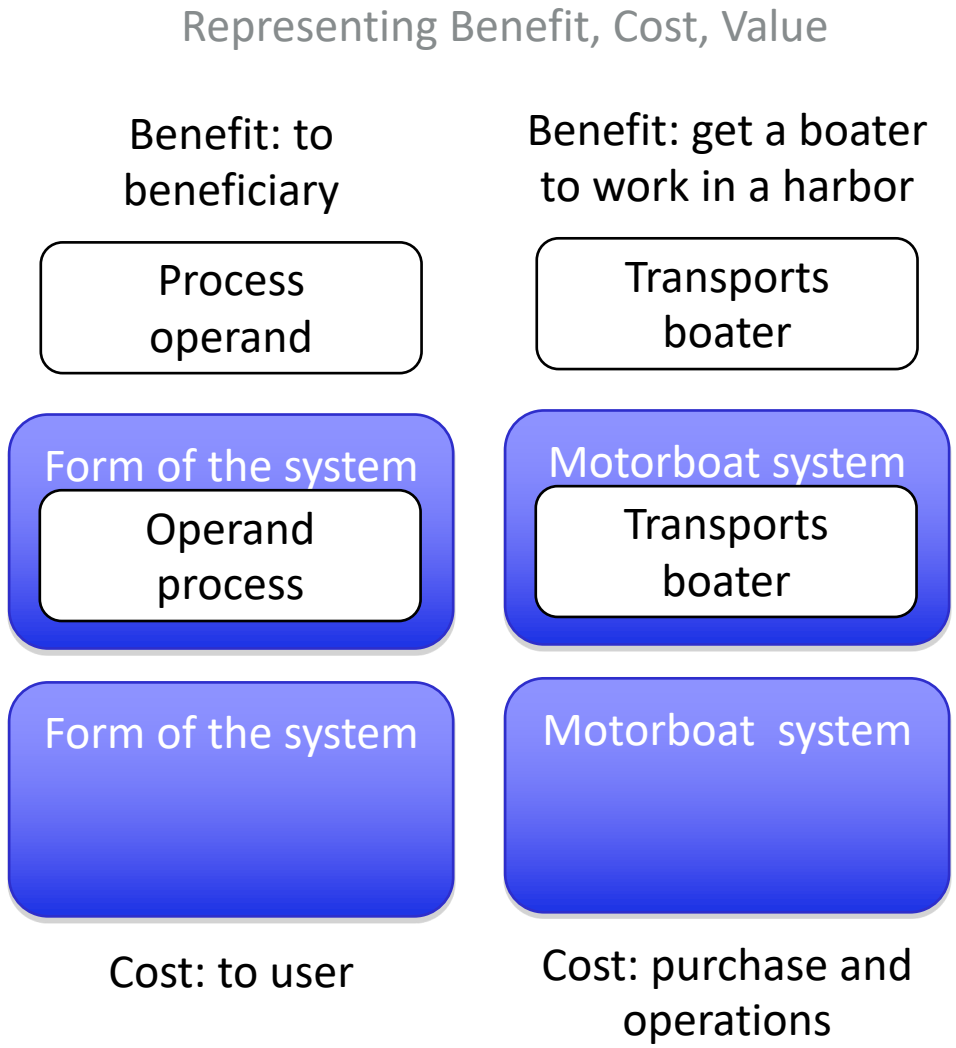
- Let's introduce a simple icon for a system – with a white rectangle within a shaded one. The inner white box shows the function – the operand and process. The outer shaded box is named for the form. This reminds us that form and function are separate and complementary aspects of a system.
- The generic example is "form" (executes) "process operand" and the worked example is "Motorboat system transports boater."



Representing Systems: Benefit, Cost, and Value

- The system function is closely related to the benefit delivered by the system. Benefit is synonymous with utility and good.
- To get the benefit is why we build a system.
- The system form is closely related to cost – including the cost to build and operate.
- Value is benefit at cost – so value is closely related to function and form

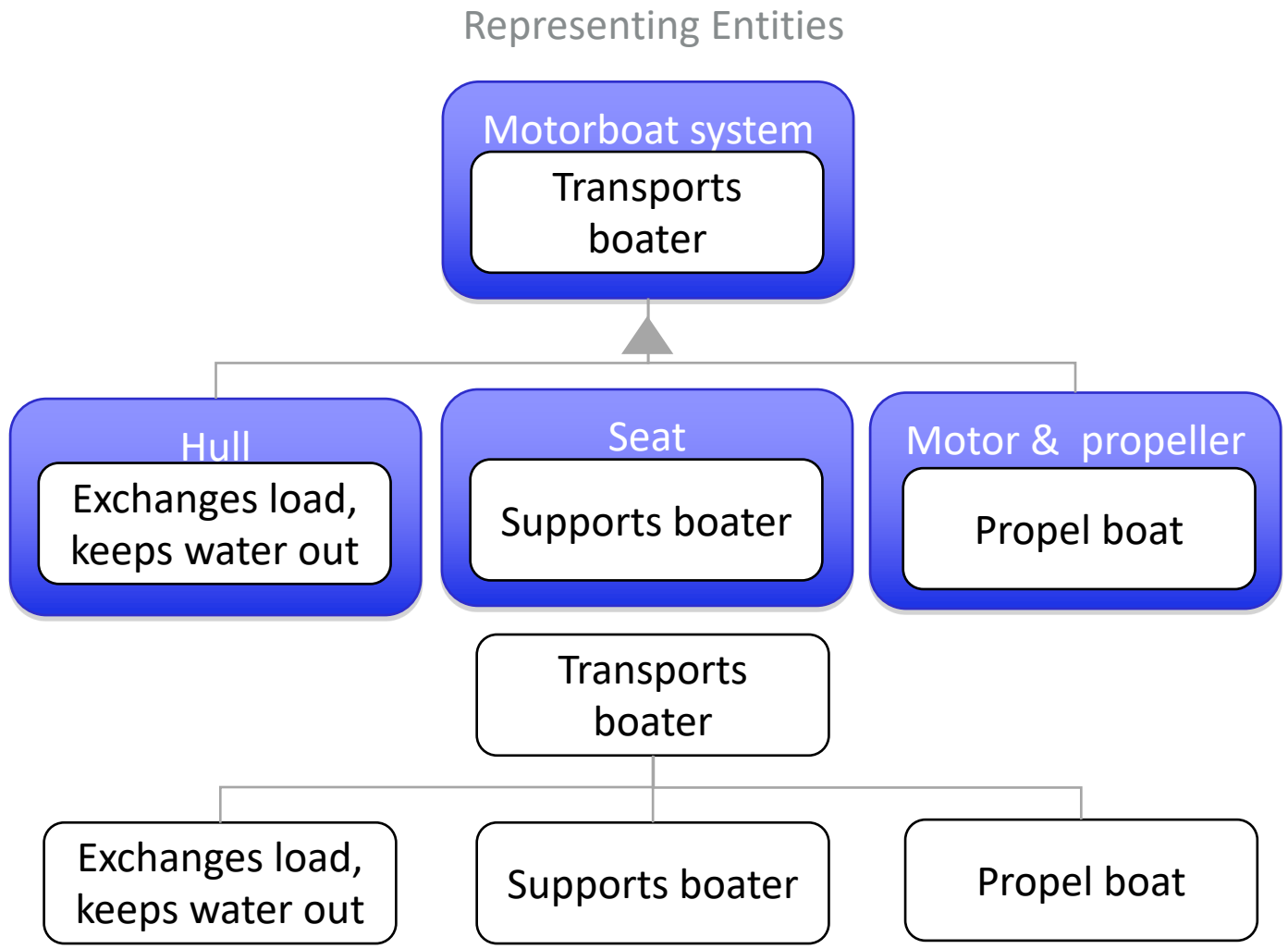
- We don't have a special icon for benefit and cost, but if needed the function and form can be annotated.
- The worked example gives benefit as "get a boater to work" while the cost is in purchase and operations, which is largely set by the design of the form of the boat



Representing Systems: Entities

- The relationship between the **whole** (the system) **and the parts** (the entities) is shown. The **system** can be decomposed into entities. The **entities** can be composed into the whole system.
- The entities can be form (not shown), function, or both form and function (sub-systems), depending on the useful abstraction. Hierarchy is usually implied by a diagram with several explicit levels.

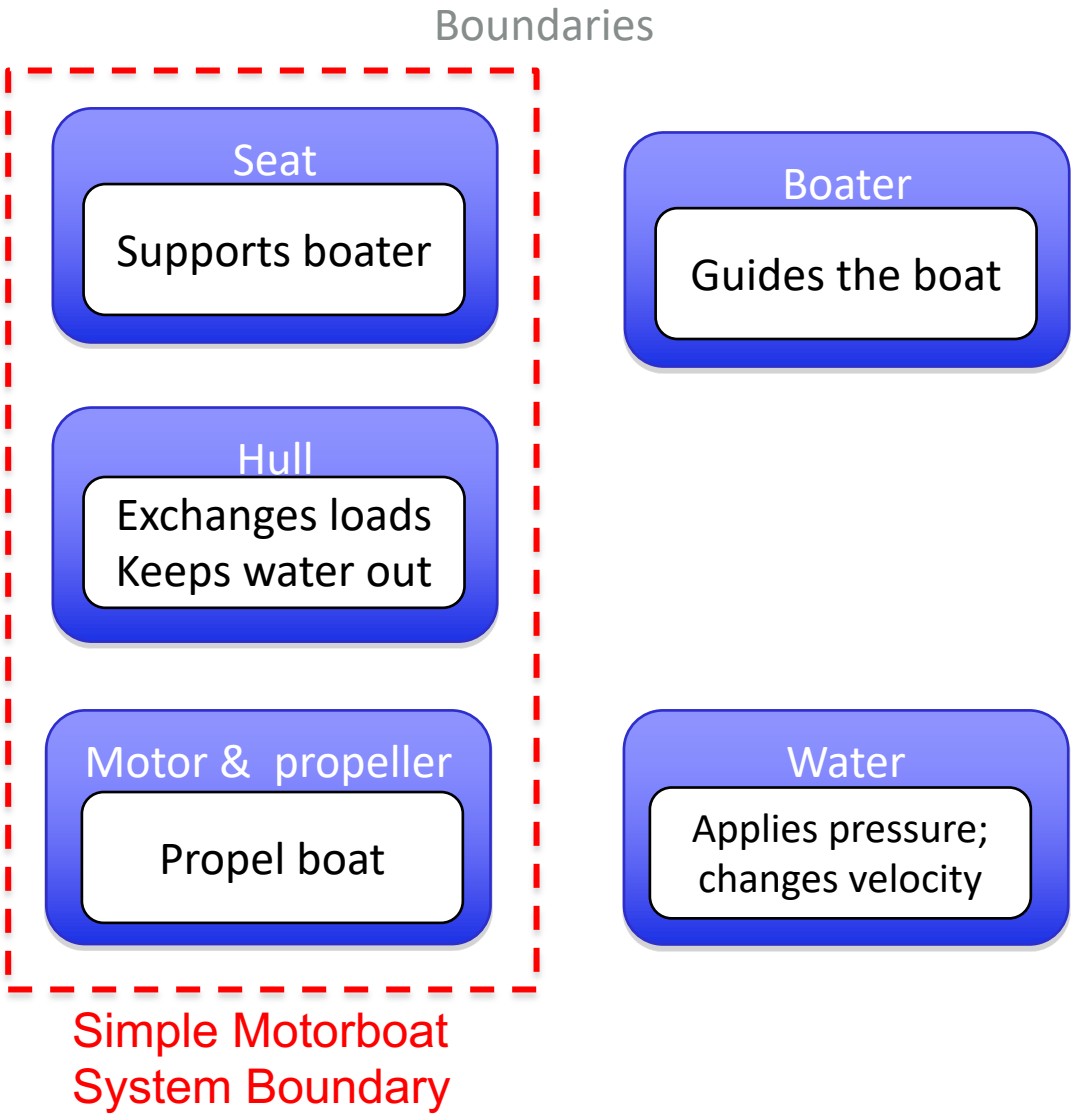
The filled triangle is sometimes used to emphasize composition and sometimes omitted.



Representing Systems: Boundaries

- An important step in system thinking is defining the **system boundary**. This uses holism and focus to establish the entities **inside** the system, and the ones **outside** the system, in what we call the **context**.

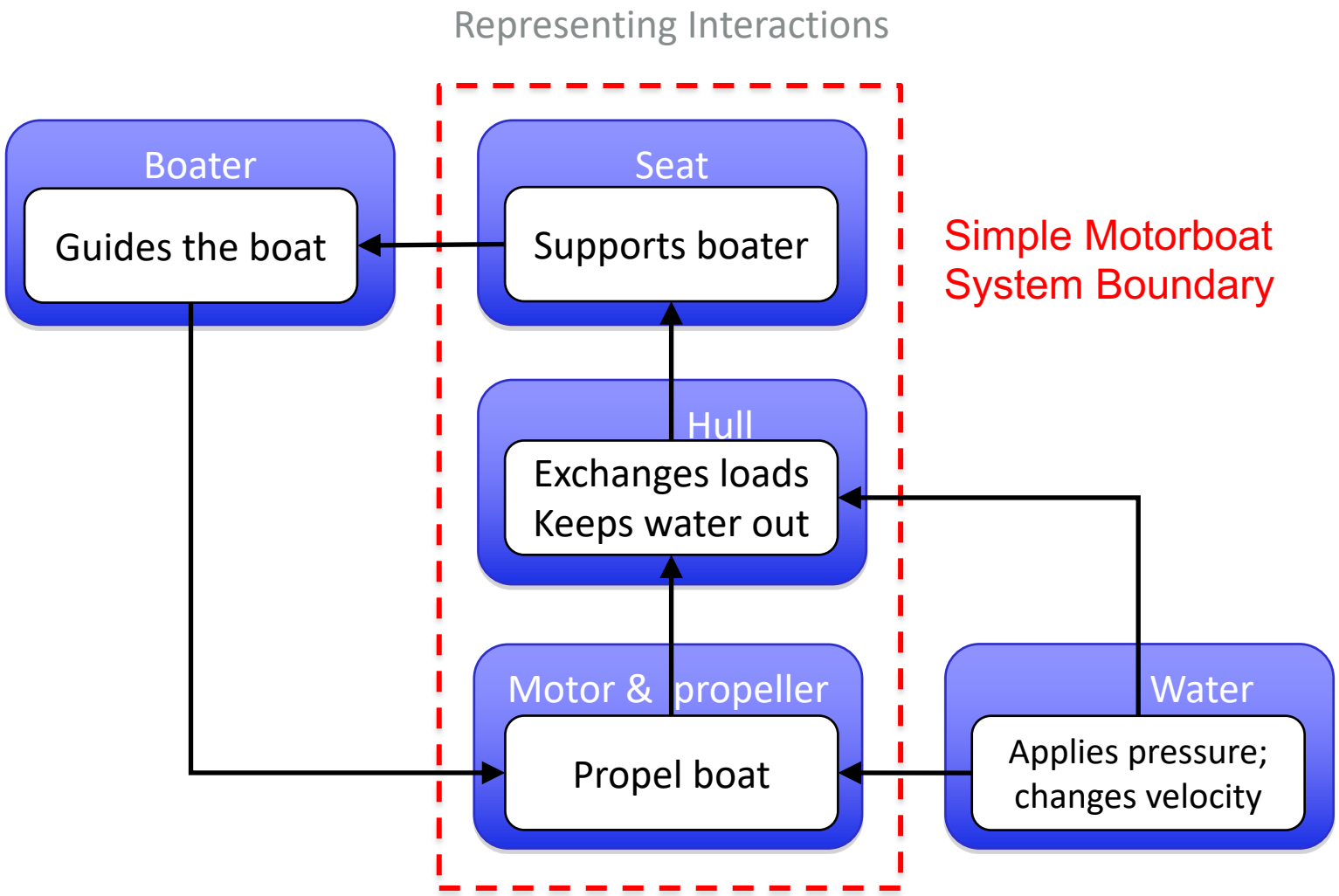
  - The system boundary is shown by the red dashed box (or sometimes by a dashed rectangle on entities not inside the boundary).
  - The entities inside the system are shown inside the dashed box, and those in the context are shown outside the dashed box. Those temporarily suppressed are omitted or shown in a different shading.



Representing Systems: Functional Relationships – Interactions

As there is form and function of the **system**, and of the **entities**, there is also form and function of **relationships**. The relationships of function is how things **interact:** one process acting on another, passing an operand to another or sharing an operand. Interactions can cross system boundaries.

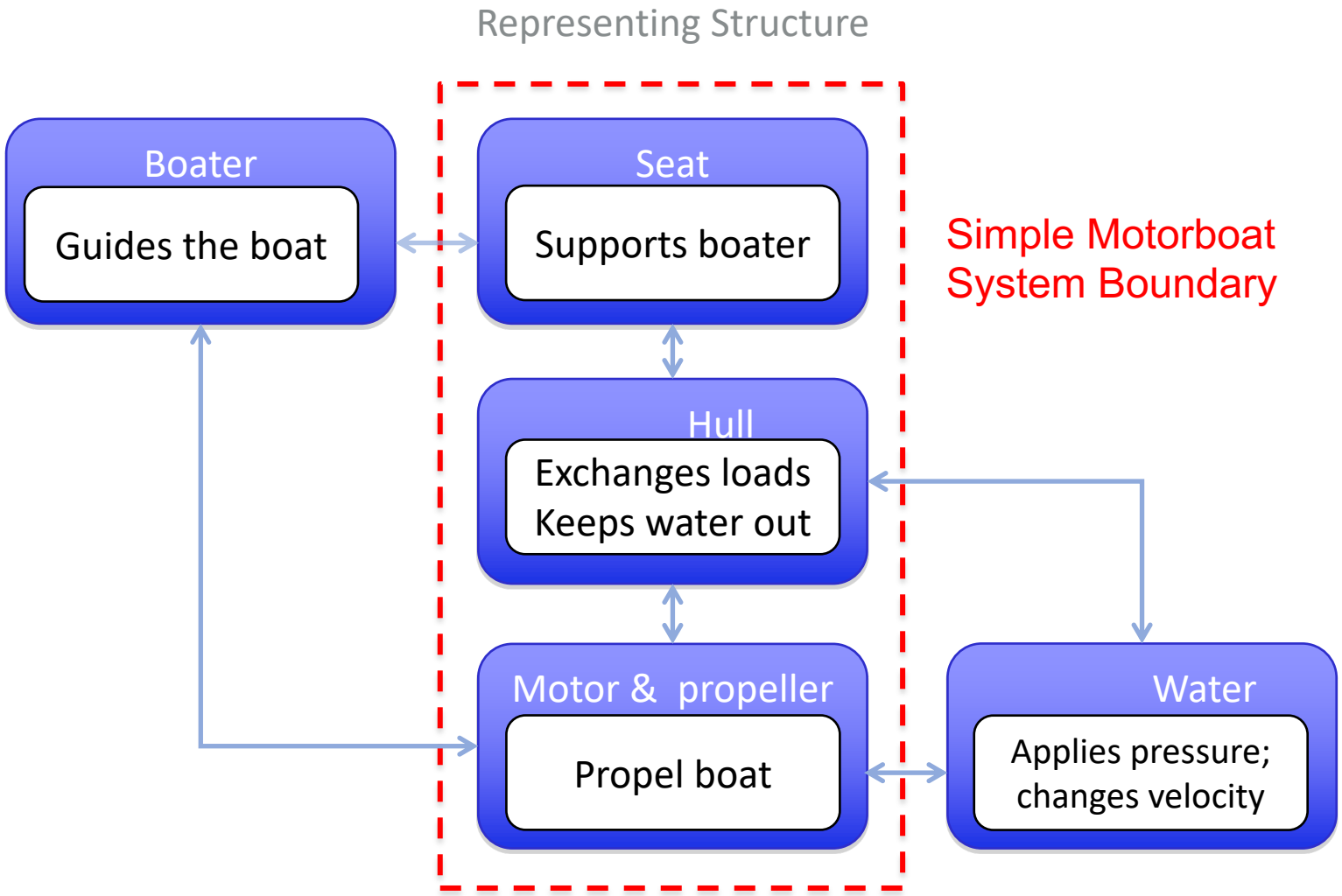
Interactions are shown by a black solid-head arrow touching the black rectangle around the function



Representing Systems: Formal Relationships – Structure

The form of relationships is how things are structured: how they are connected, where they are located, and the sequence they are in. Structure enables interaction, just as form enables function.

Structure is shown by a shaded open-headed arrow touching the outside of the shaded rectangle that indicates form. Connections can cross the system boundary.



Representing Systems: Labelling Interaction and Structure

Pairs of components	Structure (connection, location, or sequence)	Interaction (processes interacting directly, passing, or sharing an operand)
Boater and seat	Boater seated on seat	Exchanges load
Boater and Motor	Boater holds motor handle	Sets motor angle
Seat and hull	Seat screwed to hull	Exchanges load
Hull and motor	Motor attached to back of hull	Exchanges load
Hull and water	Hull partially surrounded by water	Water pressure acts on hull
Motor and water	Some water passes through propeller	Force acts on propeller due to change in water velocity

- Both the structure and interaction of entities should be explicitly shown to support system thinking, particularly how the structure enables the interaction.
- One option is to label the arrows on the system diagram, but this gets busy, especially if you label structure and interaction.
- An alternative is to build a table like the one shown that explicitly shows structure and interaction, and implicitly how structure enables interaction.