

OPM – OBJECT PROCESS METHODOLOGY

2025 – Prof. Dr. Christopher Shneider Cerqueira





OPM

Learning Objectives

- 1 Knowledge level:
 - L0-01 Match Systems Engineering description to OPM
 - L0-02 Remember OPM Basic symbols
- 2 Apply level:
 - LO-03 Use the OPM symbols
- 6 Create level:
 - LO-04 Create a concept model using OPM









- OPCloud
- Logged account:
 - <u>https://opcloud.systems/</u>
 - Connect w/ Google SSO using your @ga account
- Not logged
 - It **does not save** it, but works similarly
 - https://www.opcloud.tech





Some Examples

Starting Simple



What is the fundamental need?



be free of ghosts

Simple Example – states of the water





Figure A2. SD1 - CT scanner Project-Product Lifecycle Management in-zoomed (2)





Model-Based Protocol Engineering: Specifying Kerberos with Object-Process Methodology

Yaniv Mordecai Faculty of Industrial Engineering and Management Technion – Israel Institute of Technology Haifa, Israel yanivmov@technion.ac.il Dov Dori Engineering Systems Division Massachusetts Institute of Technology Cambridge MA, USA; Faculty of Industrial Engineering and Management Technion – Israel Institute of Technology Haifa, Israel dori@mit.edu

A client computer attempting to connect to a server is required to provide proof of its authenticity, verifying its identity and access to the server. In order to be authenticated, the client must contact the Kerberos 3^{rd} party service and receive a ticket indicating that it is who it says it is. A simple liagram of Kerberos is shown in Fig. 3 [9].



Fig. 3. Simplified View of Kerberos Authentication Flow [9]

Fig. 6. Top-Level OPM Diagram of the Kerberos Authentication Protocol



Article

Minding the Cyber-Physical Gap: Model-Based Analysis and Mitigation of Systemic Perception-Induced Failure

Yaniv Mordecai 1,2,* and Dov Dori 3,4







COMPLEXITY

Even Vader does not know what are those buttons for.

VERY DEMOTIVATIONAL .com



- OPM helps manage the system's complexity with three refinement mechanisms:
 - In-Zooming/Out-Zooming: primary used to model **process** flow.
 - Un-Folding/Folding: primary used to model **object** structure.
 - State Expression/Suppression: show only the **states** of an **object** that are relevant in the current context of the model.



- Creates a new diagram with the unfolded object at the top.
- The unfolded object can now be refined by adding its parts and attributes.
- The attributes and parts can be further refined in the same diagram or with further unfolding





- Creates a new diagram with the inzoomed process centered and enlarged.
- The in-zoomed can now be refined by displaying the internal processes that compose it.
- The objects that are linked to the process can also be refined and connected to the processes that compose the in-zoomed process.
- The inner processes can be refined further by inzooming or unfolding.











- An in-zoomed process can contain many subprocesses.
- When the flow of information in the model does not state otherwise, the processes inside an in-zoomed process are executed from top to bottom.
 - Upon subprocess completion within the context of an in-zoomed process, the subprocess immediately invokes the one(s) below it.
 - Top: Subprocesses A and B initiate in parallel as soon as Processing starts.
 - Bottom: Subprocesses B and C initiate in parallel as soon as subprocess A ends.



Product Terminating zooms into Product Finishing and Product Shipping, in that sequence.



Processing zooms into parallel A and B.



Processing zooms into A and parallel B and C, in that sequence.



The modelling of asynchronous process refinement shall use the aggregation-participation fundamental structural link either through in-diagram aggregation unfolding or as a new-diagram aggregation unfolding of the process.



Home Safety Maintaining consists of Burglary Handling, Fire Protecting, and Earthquake Alarming. Detection Module exhibits Detection Treat.

Detection Treat can be burglary, fire, or earthquake.

Burglary Detected Threat initiates Burglary Handling, which requires burglary Detected Threat.

Fire Detected Threat initiates Fire Protecting, which requires fire Detected Threat.

Earthquake Detected Threat initiates Earthquake Alarming, which requires earthquake Detected Threat





A can be s1, s2, s3, s4, or s5. P changes A from s1 to s3.



A can be s1, s3, or other states. P changes A from s1 to s3.

Case Study: Epaper Project

- Aggregating news content from several providers
- Content management
- Building subscribers profiles
 - Initial profile (content-based)
 - Collaborative filtering
- Sending news items that matching the user's profile



• SD: High level view





user can be Registered or Not registered. Not registered is initial. user exhibits Profile, Personalized items, and Device. Device exhibits Connection Details and Type. News Provider exhibits Content. ePaper System consumes Content, Profile, and Connection Details. ePaper System yields Personalized items. Registering changes user from Not registered to Registered. Registering yields Profile.

OPM Model cont.

user can be Registered or Not registered. OPL Not registered is initial. user exhibits Profile, Personalized items, Device, and Reading Habits (Clicks). Device exhibits Connection Details and Type. • SD1.1 (Zooming-In) News Provider exhibits Sources Detail. New Registering changes user from Not registered to Registered. Registering yields Profile. ePaper System exhibits Items, Aggregator, Content Management, Items, Personalization Engine, Personalized Items, and Content Delivery. Sources ePaper System consists of Items Aggregating, Items Managing, Items Personalizing, Items Delivering, and Profile Updating. ePaper System zooms into Items Aggregating, Items Managing, Items Personalizing, Profile Updating, and Items Delivering, as well as Content Delivery, Personalized Items, Personalization Engine, Items, Content Management, Aggregator, and Items. Items is Classified. Classified is final. Items can be Active or Archived. Active is initial. Items Aggregating requires Aggregator. Items Aggregating consumes Sources Detail. Items Aggregating yields Items. Items Aggregating invokes Items Managing. Items Managing requires Content Management. Items Managing consumes Active Items. Items Managing yields Classified Items. Items Managing invokes Items Personalizing. Items Personalizing requires Personalization Engine. Items Personalizing consumes Classified Items and Profile. Items Personalizing yields Personalized Items. Items Personalizing invokes Items Delivering. Profile Updating requires Personalization Engine. Profile Updating affects Profile. Profile Updating consumes Reading Habits (Clicks). Items Delivering requires Content Delivery. Items Delivering consumes Connection Details and Personalized Items. Items Delivering yields Personalized items. connection Type Details

SD1.1

Case Study cont.



Aggregator

Provider Detail



Active is initial

NAS is physical.

MySQL is Master.

MySQL.







• Item Delivering (SD1.1)









Conditions & Events



YOU DON'T SAY

Advanced Procedural Links

- Event links: triggers process activation if the event is satisfied. They are used when agents are not controlling the process. Two kinds of event links:
 - Instrument event link: the process is triggered if the instrument object exists (or is in a specific state).
 - Consumption event link: the process is triggered if the consumed object exists (or is in a specific state). The object is then consumed.
- Condition link: conditions the execution of a process to the existence of an object or to the object being in a specific state. If the condition is not matched, the process is skipped.











Logical operations





- A group of two or more procedural links of the same kind that originate from, or arrive at, the same process shall have the semantics of logical AND.
- A group of two or more procedural links of the same kind that originate from a common point, or arrive at a common point, on the same object or process shall be a link fan. A link fan shall follow the semantics of either a XOR or an OR operator.
 - The **XOR operator** shall mean that **exactly one** of the things at the divergent link end of the link fan exists.
 - The **OR operator** shall mean that **at least one** of the two or more things at the divergent end of the link fan exists.







Chef handles Meal Preparing. Meal Preparing yields Starter, Entree, and Dessert. Meal Eating affects Diner. Meal Eating consumes Dessert, Entree, and Starter.







Exactly one of Safe Owner A and Safe Owner B handles Safe Opening. At least one of Safe Owner A and Safe Owner B handles Safe Opening.

Add probabilistic chances





P yields s1 B with probability 0.32, s2 B with probability 0.24, or s3 B with probability 0.44.

The analogous deterministic case:

P yields exactly one of s1 B, s2 B, or s3 B.

P yields A with probability 0.3, B with probability q, or sc1 C with probability 0.7-q.

The analogous deterministic case:

P yields exactly one of A, B, or sc1 C.







Following path carnivore, Food Preparing consumes Meat. Following path herbivore, Food Preparing consumes Cucumber and Tomato. Following path carnivore, Food Preparing yields Stew and Steak. Following path herbivore, Food Preparing yields Salad.

Simple Example – Logics



p1 changes a from s1 to s2.

p2 changes a from s1 to s2.

Following path aa, p3 consumes s1 a. Following path bb, p3 consumes s2 a. p3 yields s3 a.

p4 changes a from s1 to s3.






- A process may have a **Duration** attribute with a value that expresses units of time. **Duration** may specialize into **Minimal Duration**, **Expected Duration**, and **Maximal Duration**.
 - The overtime exception link shall connect the source process with an overtime handling destination process to specify that if at runtime, performance of the source process instance exceeds its Maximal Duration value, then an event initiates the destination process.
 - The undertime exception link shall connect the source process with an undertime handling destination process to specify that if at runtime, performance of the source process instance takes less than its Minimal Duration value, then an event initiates the destination process.





Residue Length Computing requires the value I=0.22 m of Length of Shaft and the value of Size s=9 of Shaft Batch. Residue Length Computing changes the value of Length of Steel Rod from iI=3.00 m to residue=1.02.



Common logical structures



















12 OPM Principles

45



 1. The Function-as-a-Seed – Modelling a system starts by defining, naming, and depicting the function of the system, which is also its top-level process.



•2. The Model Fact Representation – An OPM model fact needs to appear in at least one OPD in order for it to be represented in the model.



3. The Timeline – The timeline wihin an in-zoomed process is directed by default from the top of the inzoomed process elipse to its bottom



 4. The Minimal Conceptual Modelling Language – A symbol system – a language – that can conceptually model a given system using ontology with fewer diagram kinds and fewer symbosl and relations among them is prefeable over a larger ontology with mode diagram kinds and more symbols and relations among them.



 5. The Thing Importance – The importance of a thing T in an OPM Model is directly related to the highest OPD in the OPD hierarchy where T appears.



 6. The Object Transformation by Process – In a complete OPM Model, each process must be connected to at least one object that the process transforms or one state of the object that the process transforms.



•7. The Procedural Link Uniqueness – At any level of detail, na object and a process can be connected with at most one procedural link, which uniquely determines the role of the object with respect to the process.



 8. The Singular Name – A name of an OPM thing must be singular. Plural has to be converted to singular by adding the world "Set" for inanimate things or "Group" for humans.



•9. The Graphics-Text Equivalence – Any model fact expressed graphically in an OPD is also expressed textually in the corresponding OPL paragraph.



 10. The Thing Name Uniqueness – Different things in na OPM Model which are not features must have different names. Features are distinguishable by appending to them the reserved word "of" and the name of their exhibitor.



•11. The Detail Hierarchy – Whenever na OPD becames hard to comprehend due to an excessive amount of details, a new, descendant OPD shall be created.



• 12. The Skip Semantics Precedence – Skip semantics takes precedence over wait semantics.



Final Remarks

58



- OPM is simple and powerful to talk with stakeholders and create the first architectures
- OPM uses one diagram type to handle structure and behavior
 - the language vocabulary has only a couple of symbols and semantics to mimic common sketching.
- OPM allows simple-formal modelling and enables to control the complexity.
- OPM is the only MBSE that **simulates** <u>CONCEPTS</u>.
- As it is an ISO, it is worth a try.
- OPM lacks transformational tools to other domains and a open metamodel (EMF).



- OPM main tool is the OPCloud ("web" based)
- It has been highly improved from the OPCat used through the course.
 - Usability is better
 - Allows dynamic behavior
 - IoT connectable through MQTT
 - Socket connection
 - Stereotypes
 - Styling
 - Timing





CONOPS FUNDAMENTALS

Concept Review





"Early in the system development activity, a system is conceptual in nature".



• The conceptual design provides a description of the proposed system that fulfills the stakeholders needs.





• During the concept stage, alternate concepts are created to determine the best approach to meet stakeholder needs.





 Concept design explores the feasibility of a system and guarantees that there is at least one solution that fulfils the operational need at "minimum cost".



Scenarios Methodology



- Scenarios serves as a methodology for designing, planning and decision making.
 - The exercise makes people think in a creative way
 - Reduce the chances of overlooking important factors
 - Enhances communications within and between organizations.

Operational Scenarios are stories which describe the expected utilization of the future system



Operational Scenarios

- A useful scenario is one which describes how a system is to be operated and maintained during a specific time, mission phase, operational mode, or critical sequence of activities.
- It enables one to establish the Who, What, Where, When and Why for the system.





• Operational scenarios will be input to operational concept



Operational Concept

- An abstract model created by an organization to
 - Describe the likely operation of a future or existing system in the terminology of its users
 - Describe the operational needs, desires, visions, and expectations of the user without being overly technical or formal.
- An Operational Concept generally evolves from a (abstract) concept and is a description of how a set of capabilities may be employed to achieve desired objectives or end state.

Goals of an Operational Concept

- To provide a clear vision of the intended use and the resulting benefits of the system;
- To provide a document which can be understood and utilized to facilitate a clear understanding of the system context and the users; and
- To provide the basis for system development and validation.



 Operational concept implements the required capabilities addressed by the operational scenarios




Using the OPM Model to do a MBSE CONOPs

2h30 – Ghost Catching CONOPS



 STATEMENT OF NEED: (Ghost Catching) A ghost appeared at a party; we need to have a clear way of stopping and capturing in future occurrences. This must be handled by an external trained staff, that should promptly arrive in next events, so that they can work effectively and provide good service.





Stakeholder Analysis relates to Operational Scenarios.

- Stakeholder Analysis: Identify who are involved and their intents / needs / requirements.
- **Operational Scenario**: Map some required capabilities options that accomplish the needs
- Concept of Operation: Candidate Conceptual Arquitectures that optimally answer the stakeholders' needs

[02a] Elements of Stakeholder Analysis Metamodel



Stakeholder is physical.

Stakeholder exhibits Need.

Need results into Requirement.

Stakeholder Analysis relates to Requirement.

Stakeholder Analysis relates to Stakeholder.

Stakeholder Analysis relates to Need.

[02b] Elements of Stakeholder Analysis Model

- One approach is: elicit the possible stakeholders first
- Zoom-in of the Set of Stakeholder



[02c] Elements of Stakeholder Analysis Model

• Get each stakeholder desired emergence: needs



[02d] Elements of Stakeholder Analysis Model

• Start writing some mission/stk requirements

Set Of Requirement	
The Club Shall Not Have Ghosts	
In Case Of Ghosts The Club Must Be Cleaned In [tbd] Minutes.	
The Catching Ghost System Must Be Able To Catch Ghosts Anywhere In The City	

Set Of Requirement from SD2 zooms in SD2.3 into The Club Shall Not Have Ghosts, In Case Of Ghosts The Club Must Be Cleaned In Minutes., and The Catching Ghost System Must Be Able To Catch Ghosts Anywhere In The City, in that vertical sequence.

[02e] Elements of Stakeholder Analysis Model

• Tie the requirements with the needs



Set Of Requirement from SD2 zooms in SD2.3 into The Club Shall Not Have Ghosts, In Case Of Ghosts The Club Must Be Cleaned In Minutes., and The Catching Ghost System Must Be Able To Catch Ghosts Anywhere In The City, in that vertical sequence. Need results into Set Of Requirement. Stakeholder Analysis relates to Need. Be Called To Catch Ghosts>, Cause Trouble, and Club Without Ghost are Need. Club Without Ghost relates to The Club Shall Not Have Ghosts. Be Called To Catch Ghosts relates to The Catching Ghost System Must Be Able To Catch Ghosts Anywhere In The City. Club Without Ghost relates to In Case Of Ghosts The Club Must Be Cleaned In Minutes..





It is all about the offerings that satisfy needs





Concept Of Operation relates to Club Without Ghost. Club Owner is physical. Ghost is physical. Specialists is physical. Club Owner and Specialists handle Ghost Freeing. Ghost Freeing affects Ghost. Ghost Freeing consumes Club Without Ghost.





Ghostbuster is physical and systemic.

Ghost is physical and systemic.
Ghost can be free or locked.
Tools is physical and systemic.
Ghostbuster consists of Tools.
Ghost Catching is physical and systemic.

Ghostbuster handles Ghost Catching.

Ghost Catching requires Tools. Ghost Catching affects Ghost.





Ghost Catching

[3] Organizating possible scenarios



[3] Scenario 05: Ghost Appearing



Ghost Appearing from SD1 zooms in SD1.1 into Temperature Dropping, Ghost Materializing, and Phantasmagic Ectoplasma Stabilizing, which occur in that time sequence. Ghost is physical. Ghost Materializing affects Ghost. Phantasmagic Ectoplasma Stabilizing affects Ghost.

[3] Refining Scenario Options



Magic Spelling>, Plasma Cannons Shooting, and Proton Pack Striking are Ghost Catching.

[3] Proton Pack Striking Study



Proton Pack Striking from SD1.2 zooms in SD1.2.1 into Starting Pack, Shooting, and Stopping, which occur in that time sequence. Specialists is physical.

Specialists handles Proton Pack Striking.

Proton Pack Striking requires Proton Pack.













SYSTEMS ARCHITECTURE USING OPM



- Systems Engineering engineer systems that will deliver a value to the stakeholders.
- The value comes with a form (architecture elements and properties) and with functions (actions and events)
- Complex Systems have multiple functions.
- The Systems Engineers must **identify** and **organize** those functions, managing the complexity.





- As a matter of definition, systems are composed of a set of entities and their relationships. These relationships can have two characters: they can be functional relationships or formal relationships.
- Functional relationships are relationships between entities that do something; they involve operations, transfers, or exchanges of something between the entities. We sometimes call functional relationships *interactions* to emphasize their dynamic nature.
- Formal relationships are relationships among the entities that exist or could exist stably for some period of time. We sometimes call formal relationships structure to emphasize their static nature.



FIGURE 2.11 Formal structure and functional interaction for the amplifier circuit.





In general, a functional relationship usually requires a **formal relationship**. The formal relationship is the instrument of the **functional relationship**.

The heart cannot exchange blood with the lung without a connection.

Distinguish form from function

- It is surprisingly difficult to separate form from function. In common speech, we refer to form with function words.
- Try describing a paper coffee cup, a pencil, or a spiral notebook without any reference to function. If you used the words "handle," "eraser," and "binding," you were using words rooted in a function.
- To stay entirely in the form domain, we might use "flat cardboard halfcircle," "rubber cylinder," and "metal spiral."



- Form is what has been or is eventually *implemented*.
- Form is about existence.
- Form is what the system is. It is the concrete and often visible manifestation of the system.
- Form is the physical or informational embodiment of a system that exists or has the **potential for stable**, **unconditional existence**, **for some period**, **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.



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





FIGURE 5.2 OPM diagram of process + operand yielding function. From top to bottom, these represent the process affecting the operand, consuming the operand, and producing the operand.

FORM AND FUNCTION



- Form, a system attribute consisting of elements and structure.
- Function, a system attribute consisting of entities of function and interactions through operands.
- System Architecture, the combination of form and function.

OPM FUNCTION + FORM



FIGURE 5.3 OPM representation of the canonical system architecture: The function as a process and an operand that the process affects, and the form as an instrument object.



Functional architecture

System architecture

Formal structure

FIGURE 6.1 System architecture as the combination of functional architecture and elements and structure of form.



FIGURE 6.3 Simple system architecture of sliced bread making.



FIGURE 6.4 A not-so-simple system architecture of sliced bread making.





FIGURE 6.16 Projection onto the objects of making sliced bread.





FIGURE 6.19 Projection onto the form of making sliced bread. (Compare with Figure 6.16.)



B Processing

Attribute Of

Transforming

Intent

Form Specific System Form Attribute Of Form

Removing a wine bottle cork



FIGURE 7.1 Concepts for removing a wine bottle cork.

Selectable options template



FIGURE 7.5 Tree showing the options among concepts, with different specific operating and specific instruments of form, but all with the same specific operand 1.

Transportation service options



FIGURE 7.8 Solution-neutral function and solution-specific concept options for "transportation service" system.


TABLE 8.4 | A DSM array showing the coupling through operands of the processes of the air transportation service with suggested Level 1 clustering. Matrix entries show the number of operands linking the row, column processes.



nourishing

Blue Origin ss Virgin Galacts



IAC-18.D1.4A.8

Model-Based Concept Framework for Suborbital Human Spaceflight Missions

Yaroslav Menshenin^a, Edward Crawley^b

^a Skoltech Space Center, Skolkovo Institute of Science and Technology, 3 Nobelya str., Moscow, Russia 121205, ymenshenin@iskoltech.uu ^b Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, 77 Massachusetts Avenue



Fig. 16. OPM representation of the integrated concept (fourth assertion) at the second level decomposition for Blue Origin's "Propulsion module" and "Capsule"