LOCKHEED MARTIN

Using MBSE to Perform Functional Analysis to Analyze Customer Requirements

Dr. Yvonne Bijan

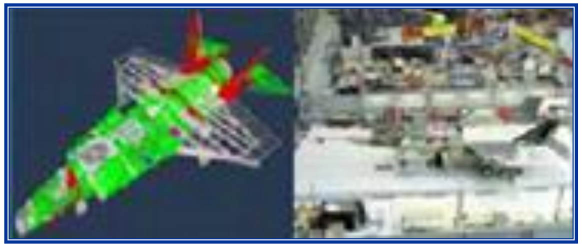
Lockheed Martin
October 2020

Yvonne Bijan

- **Abstract**: The theory of Systems Engineering says we start at the top with customer needs, define requirements for the top level system in our product hierarchy, then decompose those requirements into the subsystem requirements at the next level down in the hierarchy- or does it? Some believe architecture work can begin once the requirements work is completed at any given level, but is that the only time that the tools and methods for architecture are useful? Have you ever been on a program where the practice of Systems Engineering aligns with the theory of Systems Engineering 100 percent? How do you cope with the deviations from the theory? What happens when the original requirement at the top is not an appropriate one that people should start with? This presentation will have a brief overview of systems engineering and then walk through an example of a top level requirement handed down from a customer to a program. We will use functional analysis to improve the requirement and answer the preceding questions.

- **Bio**: Yvonne Bijan is an LM Fellow and has worked for Lockheed Martin for twenty-one years. She has a Doctor of Philosophy in System Engineering from Southern Methodist University. Yvonne has worked on numerous aeronautic and space programs including F-35 and Space Based Infrared System. She is a Certified Enterprise Architect, Certified Systems Engineering Professional, Certified SysML Model Builder Advanced, SAFe Agilist, and holds a QFD Greenbelt. She has a B.S. in Physics with a minor in Math and an M.S. in Computer Science with a concentration in Software Engineering. Dr. Bijan has also been the President of the North Texas Chapter of INCOSE.

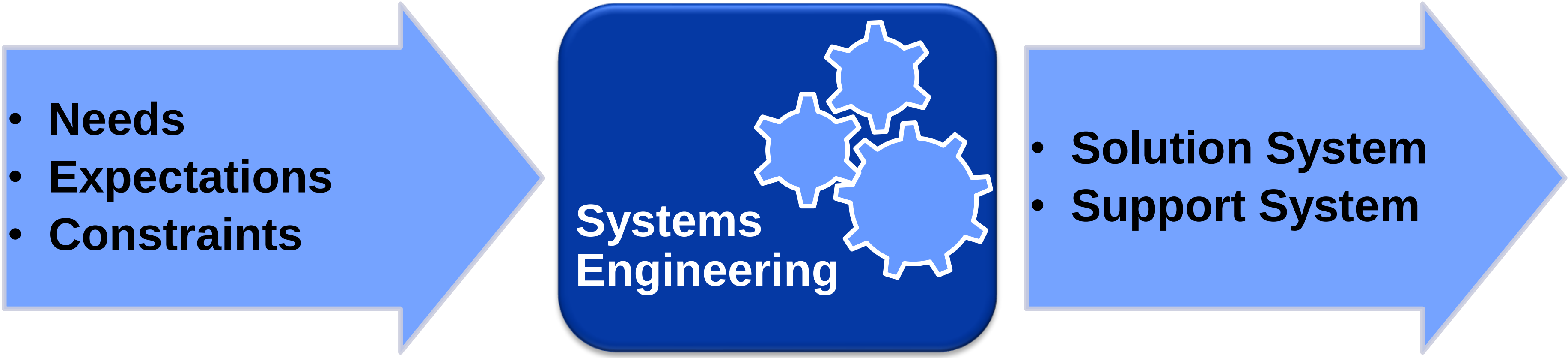
Vision



Meet our customers' needs by applying Systems Engineering to deliver high quality solutions

What is Systems Engineering?

“Interdisciplinary approach governing the total technical and managerial effort required to transform a set of stakeholder needs, expectations, and constraints into a solution and to support that solution throughout its life” (ISO/IEC/IEEE 15288:2015)



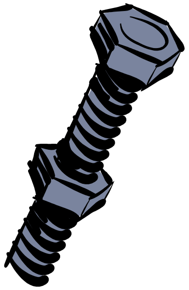
- Needs
- Expectations
- Constraints

Systems Engineering

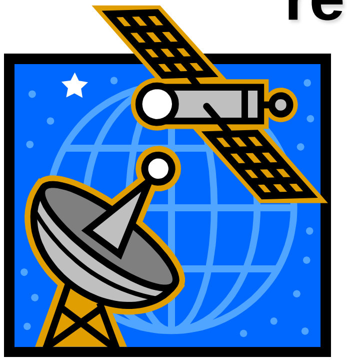
- Solution System
- Support System

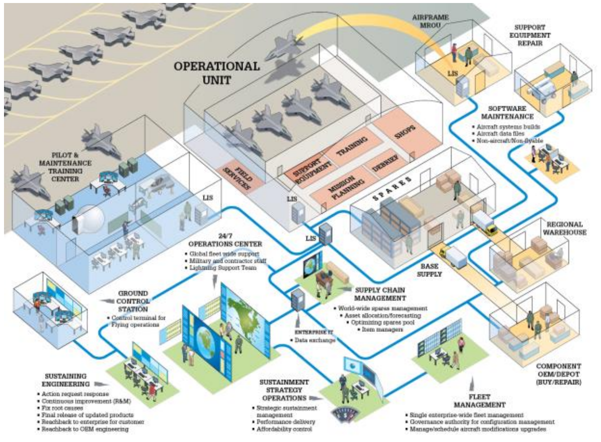
- SE is not an organization or a tool
- Applied across programs rigorously
- Analogy: Technical "glue" that makes separate design disciplines and subsystems function together to provide an integrated system that performs a specific mission

What are Systems?

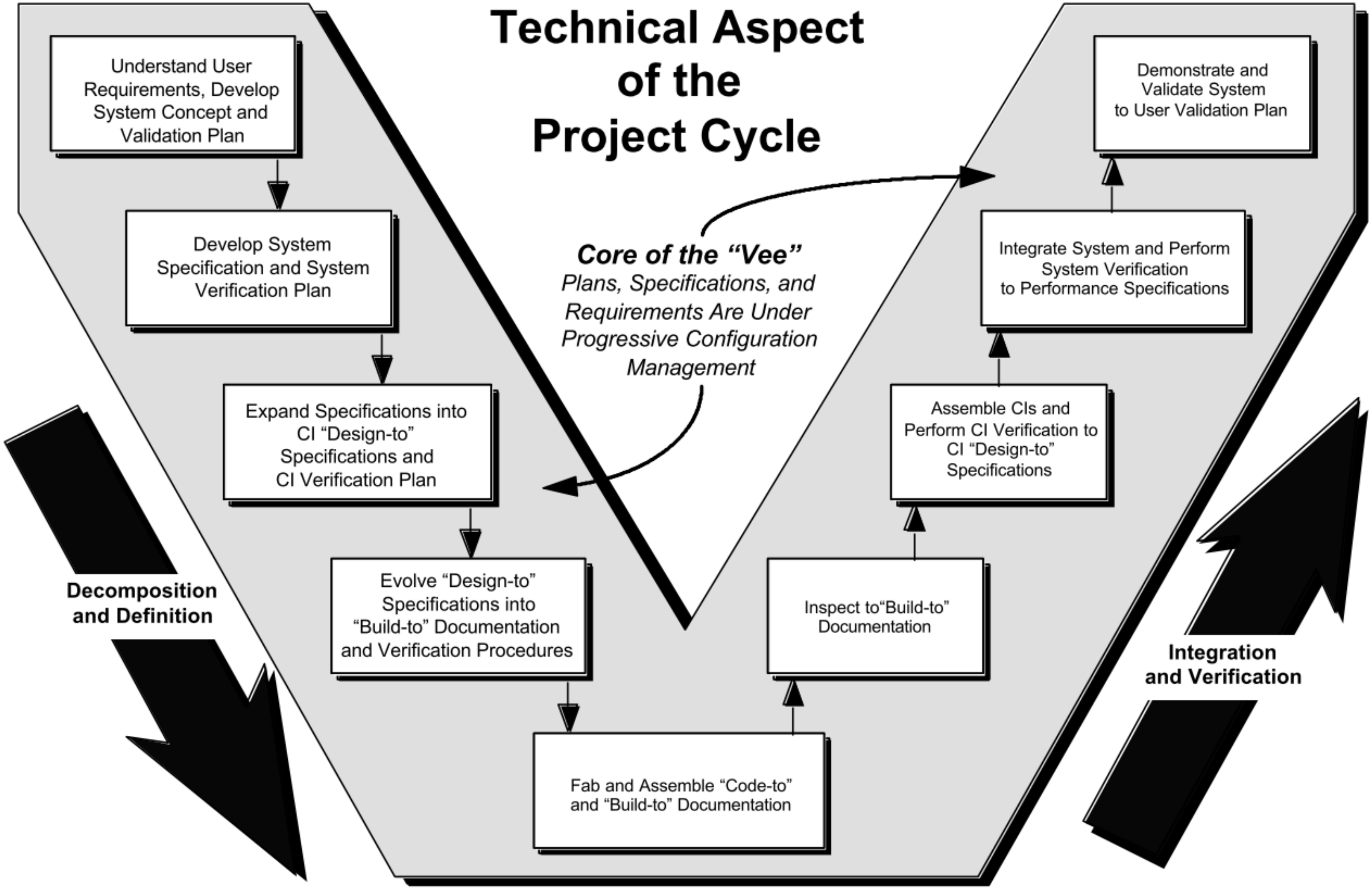


- A set of interacting or interdependent components forming an integrated whole
- Characteristics of Systems
  - Structure: Parts (or components) that are directly or indirectly related to each other
  - Behavior: Processes that fulfill its function or purpose
  - Relationships: Parts and processes are connected by structural and/or behavioral relationships




OPERATIONAL UNIT
AIRFRAME MROU
SUPPORT EQUIPMENT REPAIR
PILOT & MAINTENANCE TRAINING CENTER
FIELD SERVICES
SUPPORT EQUIPMENT
TRAINING
SHOPS
MISSION PLANNING
DEBRIEF
SPARES
LIS
LIS
LIS
LIS
SOFTWARE MAINTENANCE
• Aircraft systems builds
• Aircraft data files
• Non-aircraft/Non-flyable
24/7 OPERATIONS CENTER
• Global fleet wide support
• Military and contractor staff
• Lightning Support Team
REGIONAL WAREHOUSE
BASE SUPPLY
GROUND CONTROL STATION
• Control terminal for Flying operations
SUPPLY CHAIN MANAGEMENT
• World-wide spares management
• Asset allocation/forecasting
• Optimizing spares pool
• Item managers
ENTERPRISE IT
• Data exchange
COMPONENT OEM/DEPOT (BUY/REPAIR)
SUSTAINING ENGINEERING
• Action request response
• Continuous improvement (R&M)
• Fix root causes
• Final release of updated products
• Reachback to enterprise for customer
• Reachback to OEM engineering
SUSTAINMENT STRATEGY OPERATIONS
• Strategic sustainment management
• Performance delivery
• Affordability control
FLEET MANAGEMENT
• Single enterprise-wide fleet management
• Governance authority for configuration management
• Manage/schedule aircraft modifications upgrades

Systems Engineering Vee



Technical Aspect of the Project Cycle

Understand User Requirements, Develop System Concept and Validation Plan

Develop System Specification and System Verification Plan

Expand Specifications into CI "Design-to" Specifications and CI Verification Plan

Evolve "Design-to" Specifications into "Build-to" Documentation and Verification Procedures

Fab and Assemble "Code-to" and "Build-to" Documentation

Inspect to"Build-to" Documentation

Assemble CIs and Perform CI Verification to CI "Design-to" Specifications

Integrate System and Perform System Verification to Performance Specifications

Demonstrate and Validate System to User Validation Plan

Core of the "Vee"
Plans, Specifications, and Requirements Are Under Progressive Configuration Management

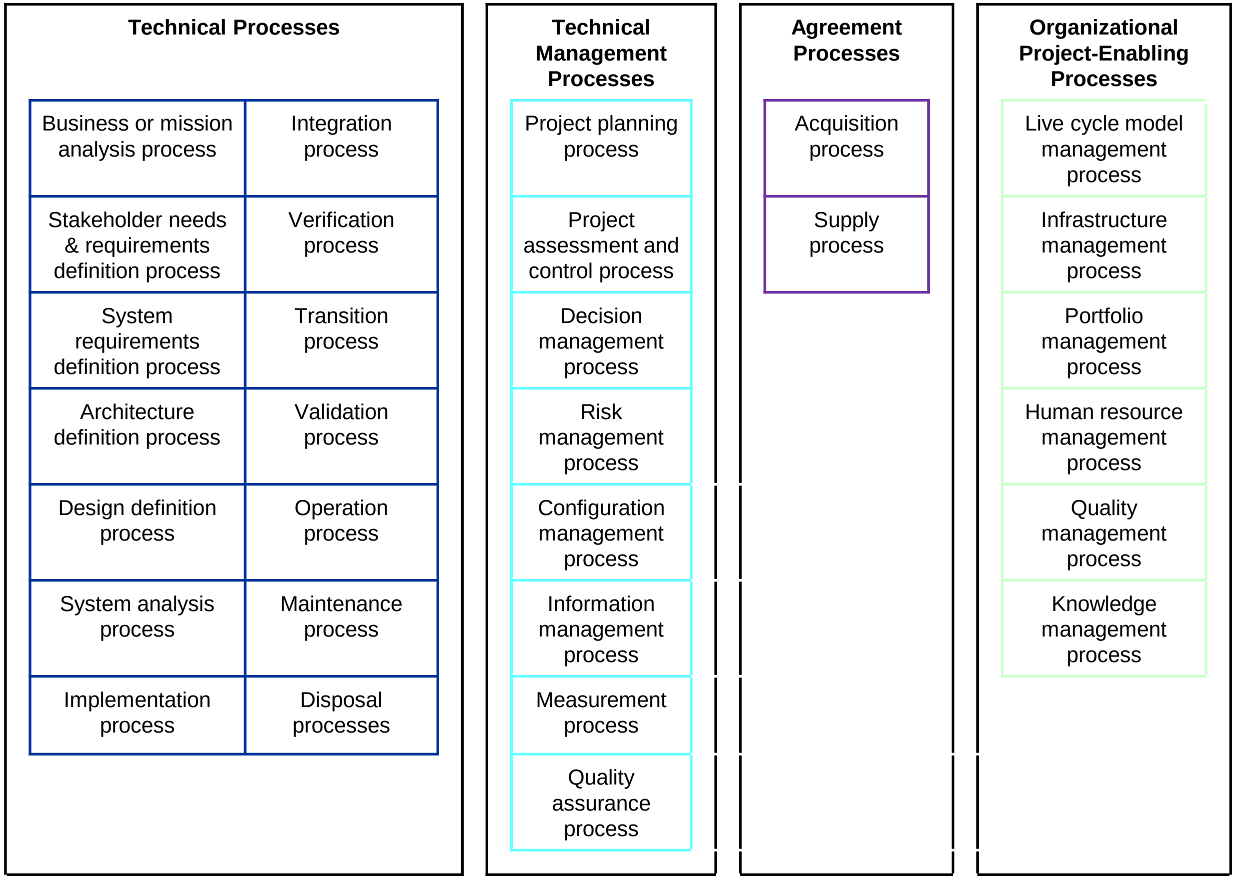
Decomposition and Definition

Integration and Verification

Forsberg and Mooz, 1998

Linear?

- **Is Systems Engineering a set of linear processes that are implemented once?**



Technical Processes

Business or mission analysis process	Integration process
Stakeholder needs & requirements definition process	Verification process
System requirements definition process	Transition process
Architecture definition process	Validation process
Design definition process	Operation process
System analysis process	Maintenance process
Implementation process	Disposal processes

Technical Management Processes

- Project planning process
- Project assessment and control process
- Decision management process
- Risk management process
- Configuration management process
- Information management process
- Measurement process
- Quality assurance process

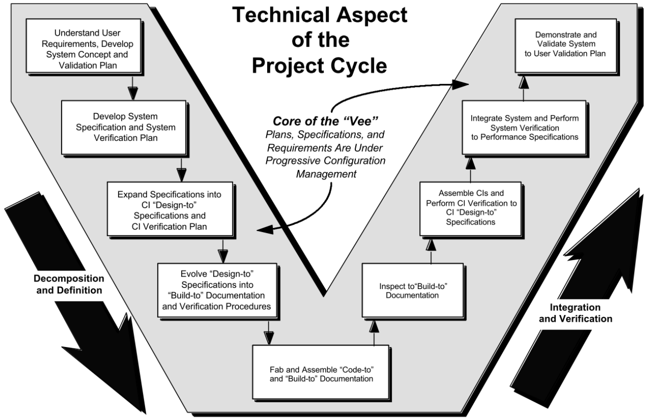
Agreement Processes

- Acquisition process
- Supply process

Organizational Project-Enabling Processes

- Live cycle model management process
- Infrastructure management process
- Portfolio management process
- Human resource management process
- Quality management process
- Knowledge management process

Figure 1.1 System life cycle processes per ISO/IEC/IEEE 15288. This figure is excerpted from ISO/IEC/IEEE 15288:2015, Figure 4 on page 17.


Technical Aspect of the Project Cycle
Understand User Requirements, Develop System Concept and Validation Plan
Develop System Specification and System Verification Plan
Expand Specifications into CI "Design-to" Specifications and CI Verification Plan
Evolve "Design-to" Specifications into "Build-to" Documentation and Verification Procedures
Fab and Assemble "Code-to" and "Build-to" Documentation
Inspect to"Build-to" Documentation
Assemble CIs and Perform CI Verification to CI "Design-to" Specifications
Integrate System and Perform System Verification to Performance Specifications
Demonstrate and Validate System to User Validation Plan
Core of the "Vee"
Plans, Specifications, and Requirements Are Under Progressive Configuration Management
Decomposition and Definition
Integration and Verification

Forsberg and Mooz, 1998

DoDI 5000.02	**Materiel Solution Analysis**	**Technology Development**	**Engineering and Manufacturing Development**	**Production and Deployment**	**Operations and Support**

Messy?

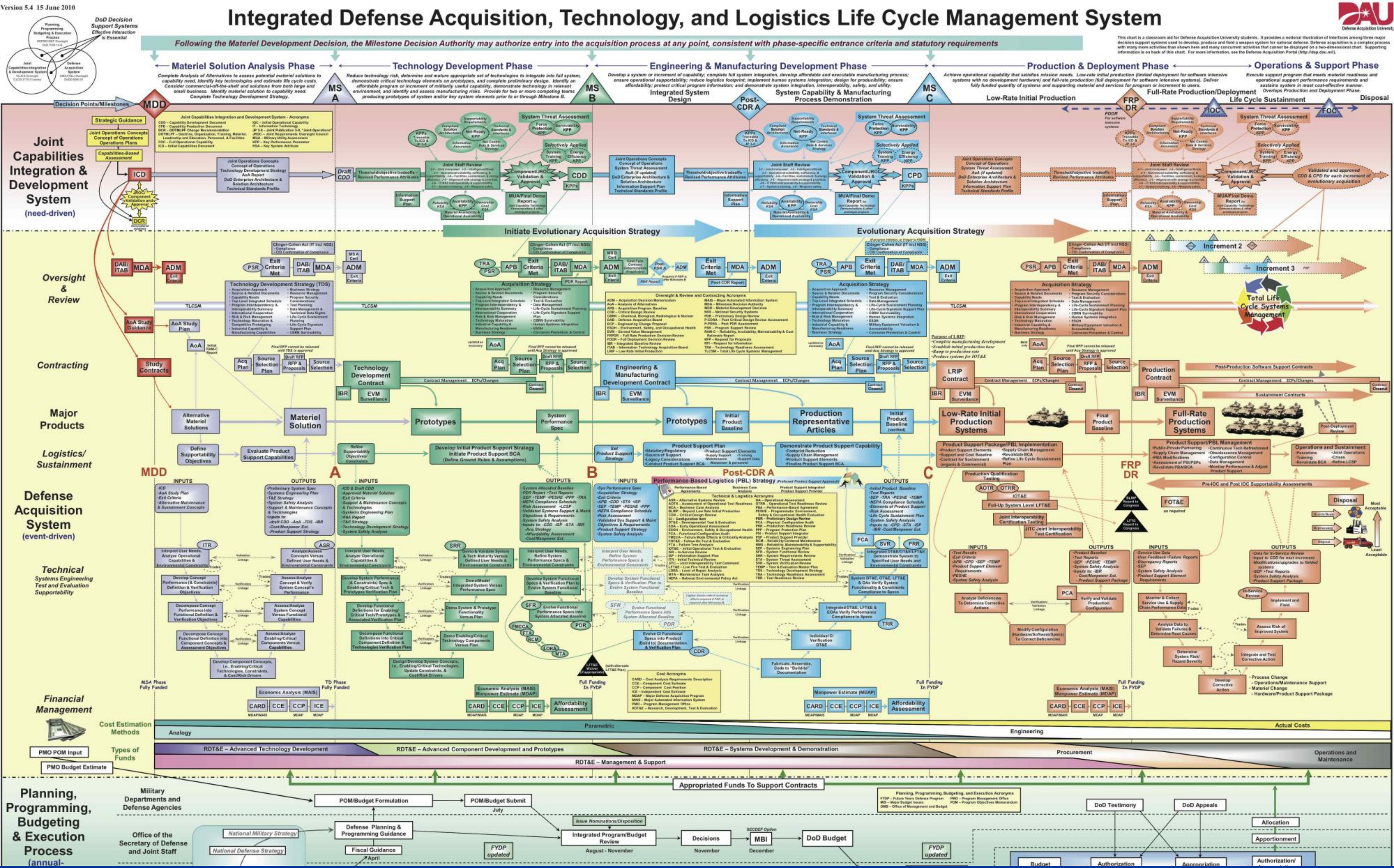

Version 5.4 15 June 2010
DoD Decision Support Systems Effective Interaction is Essential
Integrated Defense Acquisition, Technology, and Logistics Life Cycle Management System
DAU Defense Acquisition University
Following the Materiel Development Decision, the Milestone Decision Authority may authorize entry into the acquisition process at any point, consistent with phase-specific entrance criteria and statutory requirements
Materiel Solution Analysis Phase
Technology Development Phase
Engineering & Manufacturing Development Phase
Production & Deployment Phase
Operations & Support Phase
MS A
MS B
MS C
MDD
Integrated System Design
Post-CDR A
System Capability & Manufacturing Process Demonstration
Low-Rate Initial Production
FRP DR
Full-Rate Production/Deployment
Life Cycle Sustainment
Disposal
IOC
FOC
Decision Points/Milestones
Joint Capabilities Integration & Development System
(need-driven)
ICD
CDD
CPD
DCR
Initiate Evolutionary Acquisition Strategy
Evolutionary Acquisition Strategy
Increment 2
Increment 3
Oversight & Review
Total Life Cycle Systems Management
Contracting
Study Contracts
Technology Development Contract
Engineering & Manufacturing Development Contract
LRIP Contract
Production Contract
Major Products
Alternative Materiel Solutions
Materiel Solution
Prototypes
System Performance Spec
Initial Product Baseline
Production Representative Articles
Initial Product Baseline (verified)
Low-Rate Initial Production Systems
Final Product Baseline
Full-Rate Production Systems
Logistics/ Sustainment
Defense Acquisition System
(event-driven)
Technical
Systems Engineering Test and Evaluation Supportability
Performance-Based Logistics (PBL) Strategy
Financial Management
Cost Estimation Methods
Analogy
Parametric
Engineering
Actual Costs
Types of Funds
RDT&E – Advanced Technology Development
RDT&E – Advanced Component Development and Prototypes
RDT&E – Systems Development & Demonstration
Procurement
Operations and Maintenance
RDT&E – Management & Support
PMO POM Input
PMO Budget Estimate
Appropriated Funds To Support Contracts
Planning, Programming, Budgeting & Execution Process
(annual-
Military Departments and Defense Agencies
Office of the Secretary of Defense and Joint Staff
POM/Budget Formulation
POM/Budget Submit
July
National Military Strategy
National Defense Strategy
Defense Planning & Programming Guidance
Fiscal Guidance
April
Issue Nominations/Disposition
Integrated Program/Budget Review
August - November
Decisions
November
MBI
December
DoD Budget
FYDP updated
DoD Testimony
DoD Appeals
Allocation
Apportionment
Authorization/
Budget
Authorization
Appropriation

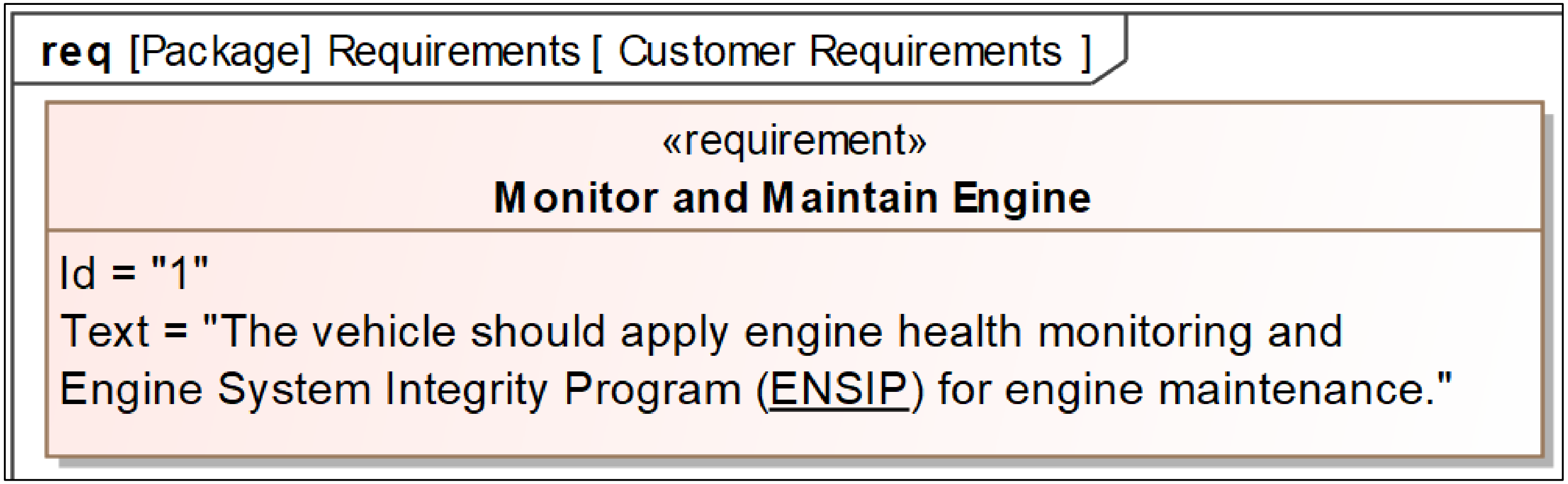
Image source: “Integrated Defense Acquisition, Technology, Logistics, Life Cycle Management,” Defense Acquisition University, https://ilc.dau.mil/html/ILC_Main.htm (accessed October, 2017).

In theory, there is no difference between theory and practice. In practice, there is.

- Yogi Berra

Theory vs. Practice

- Top down SE process says requirements should start at the top level system and be decomposed to lower level systems.
- Requirements are supposed to be clear and singular and have shall for US contracts
- But…
- You get this requirement in your specification:



req [Package] Requirements [Customer Requirements]

«requirement»

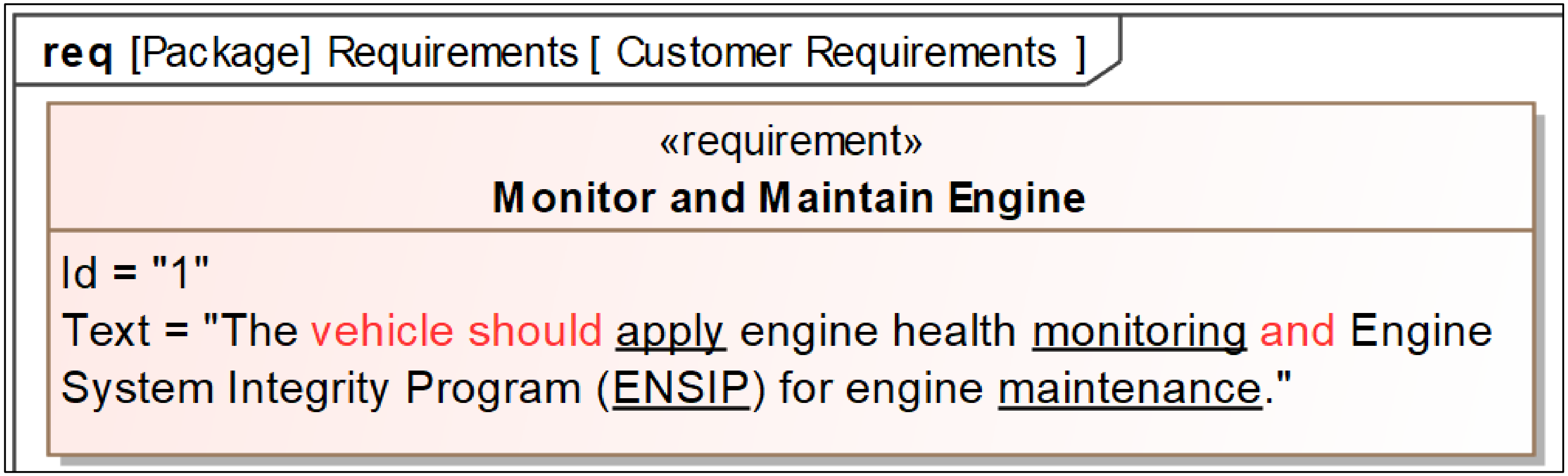
Monitor and Maintain Engine

Id = "1"

Text = "The vehicle should apply engine health monitoring and Engine System Integrity Program (ENSIP) for engine maintenance."

Example

- **What is this requirement about?**
- **What is wrong with the requirement?**



req [Package] Requirements [Customer Requirements]

«requirement»

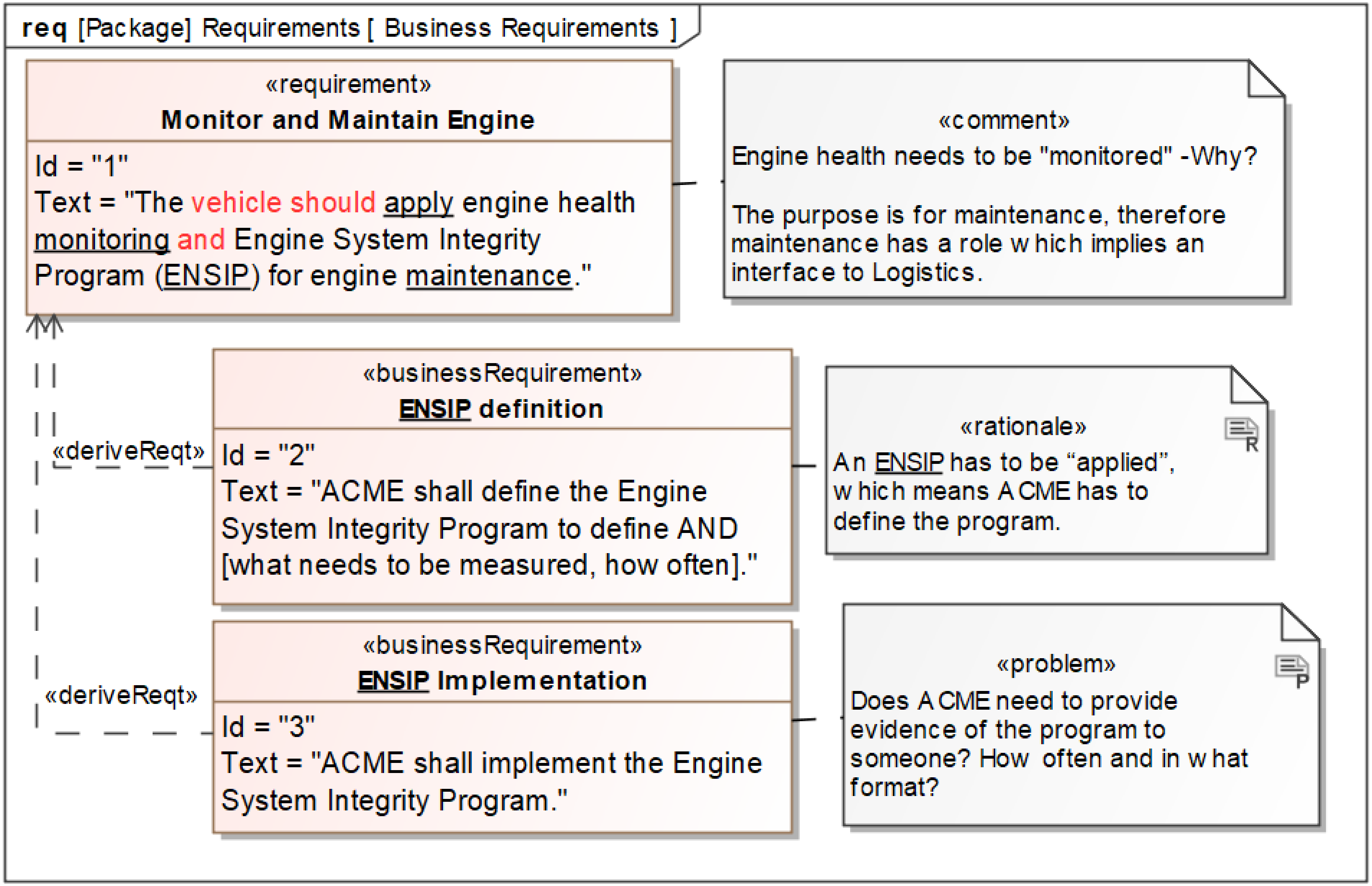
Monitor and Maintain Engine

Id = "1"

Text = "The vehicle should apply engine health monitoring and Engine System Integrity Program (ENSIP) for engine maintenance."

- **Let's do some analysis of this…**

Example



req [Package] Requirements [Business Requirements]

«requirement»
Monitor and Maintain Engine

Id = "1"
Text = "The vehicle should apply engine health monitoring and Engine System Integrity Program (ENSIP) for engine maintenance."

«comment»
Engine health needs to be "monitored" -Why?

The purpose is for maintenance, therefore maintenance has a role which implies an interface to Logistics.

«deriveReqt»

«businessRequirement»
ENSIP definition

Id = "2"
Text = "ACME shall define the Engine System Integrity Program to define AND [what needs to be measured, how often]."

«rationale»
An ENSIP has to be “applied”, which means ACME has to define the program.

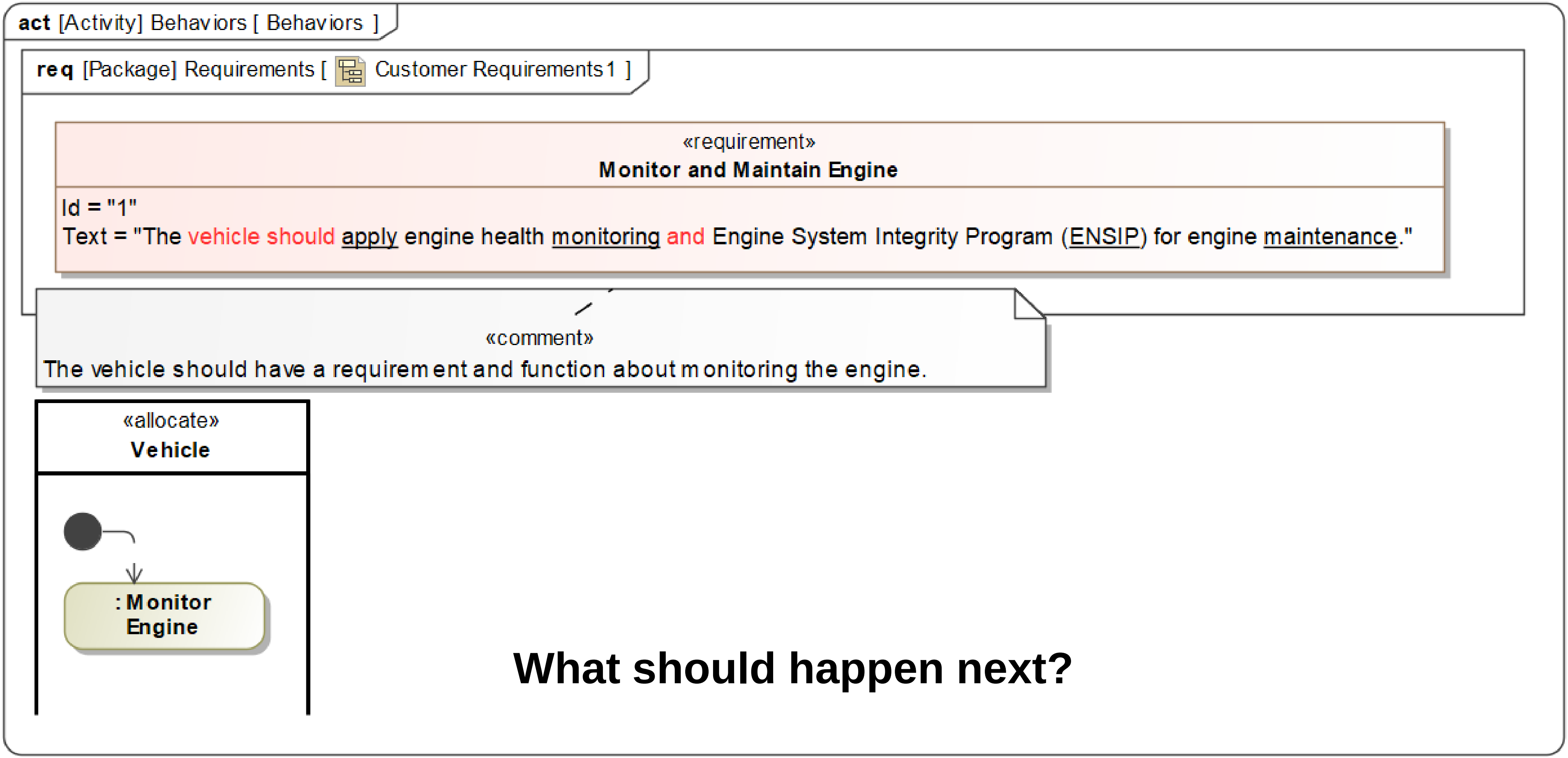
«deriveReqt»

«businessRequirement»
ENSIP Implementation

Id = "3"
Text = "ACME shall implement the Engine System Integrity Program."

«problem»
Does ACME need to provide evidence of the program to someone? How often and in what format?

Engine Example


act [Activity] Behaviors [Behaviors]
req [Package] Requirements [Customer Requirements1]
«requirement»
Monitor and Maintain Engine

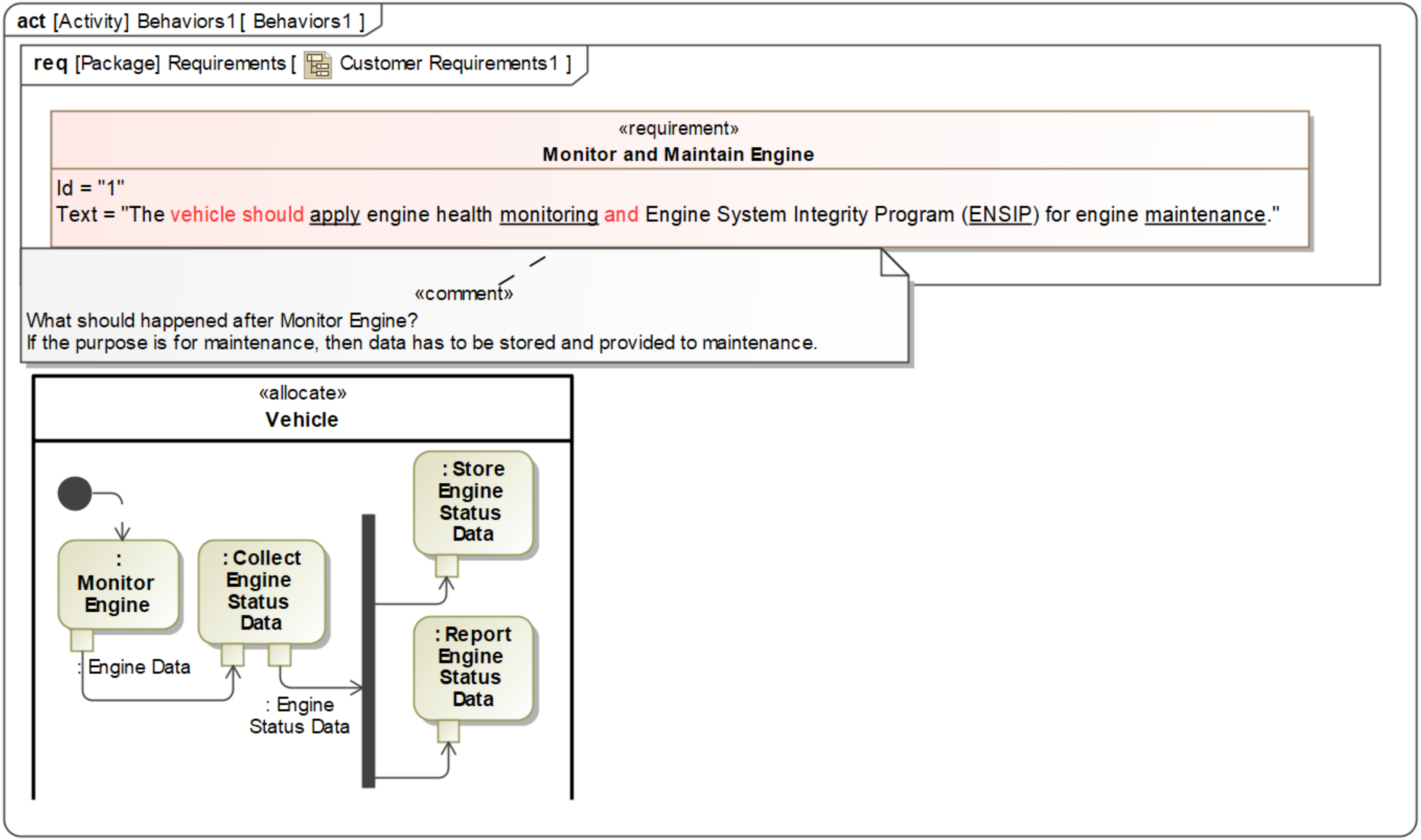
Id = "1"
Text = "The vehicle should apply engine health monitoring and Engine System Integrity Program (ENSIP) for engine maintenance."

«comment»
The vehicle should have a requirement and function about monitoring the engine.

«allocate»
Vehicle
: Monitor Engine

What should happen next?

Engine Example



act [Activity] Behaviors1 [Behaviors1]

req [Package] Requirements [Customer Requirements1]

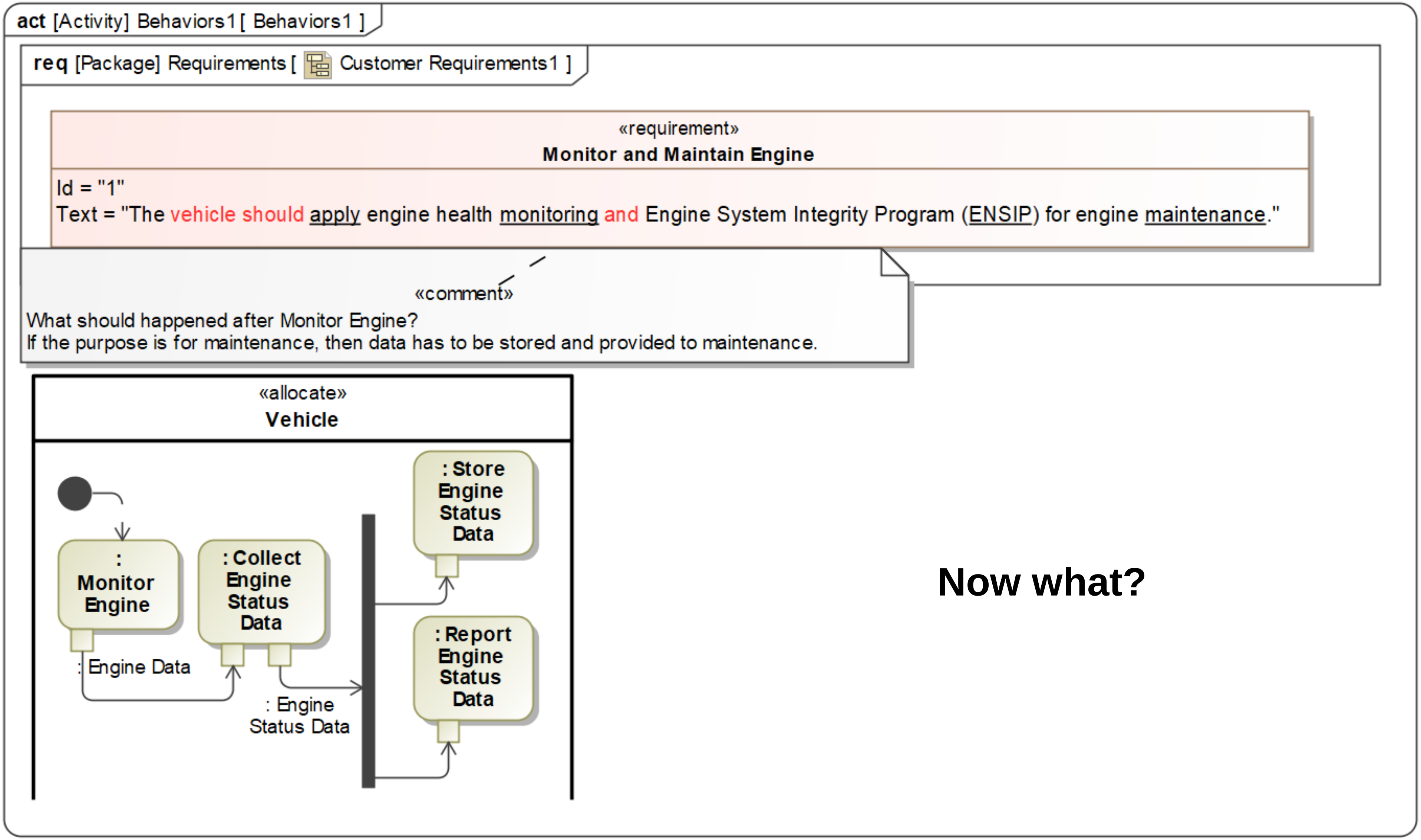
«requirement»
Monitor and Maintain Engine

Id = "1"
Text = "The vehicle should apply engine health monitoring and Engine System Integrity Program (ENSIP) for engine maintenance."

«comment»
What should happened after Monitor Engine?
If the purpose is for maintenance, then data has to be stored and provided to maintenance.

«allocate»
Vehicle
: Monitor Engine
: Collect Engine Status Data
: Store Engine Status Data
: Report Engine Status Data
: Engine Data
: Engine Status Data

Engine Example



act [Activity] Behaviors1 [Behaviors1]
req [Package] Requirements [Customer Requirements1]
«requirement»
Monitor and Maintain Engine

Id = "1"

Text = "The vehicle should apply engine health monitoring and Engine System Integrity Program (ENSIP) for engine maintenance."

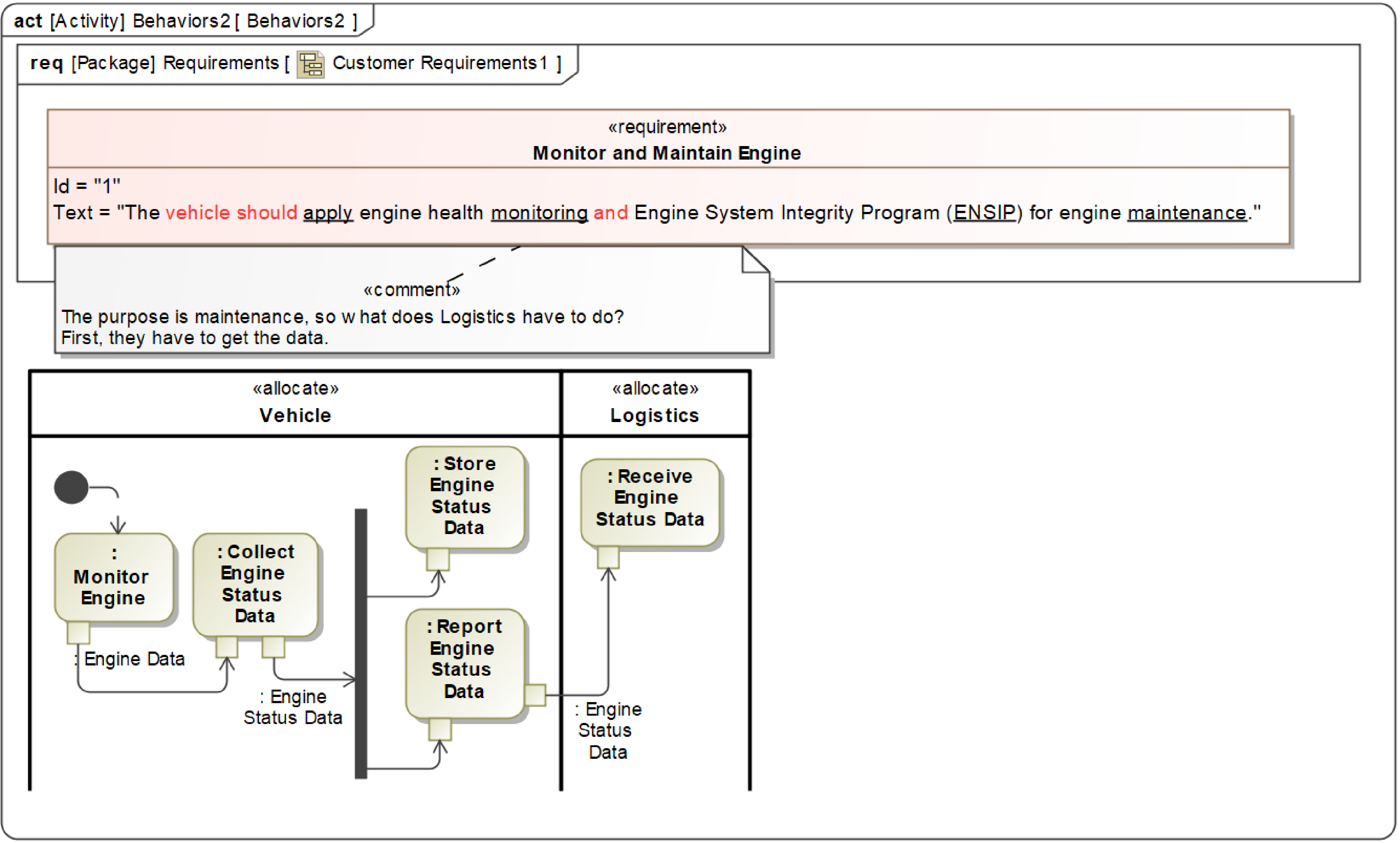
What should happened after Monitor Engine?

If the purpose is for maintenance, then data has to be stored and provided to maintenance.

«allocate»
Vehicle
: Monitor Engine
: Engine Data
: Collect Engine Status Data
: Engine Status Data
: Store Engine Status Data
: Report Engine Status Data

Now what?

Engine Example



act [Activity] Behaviors2 [Behaviors2]

req [Package] Requirements [Customer Requirements1]

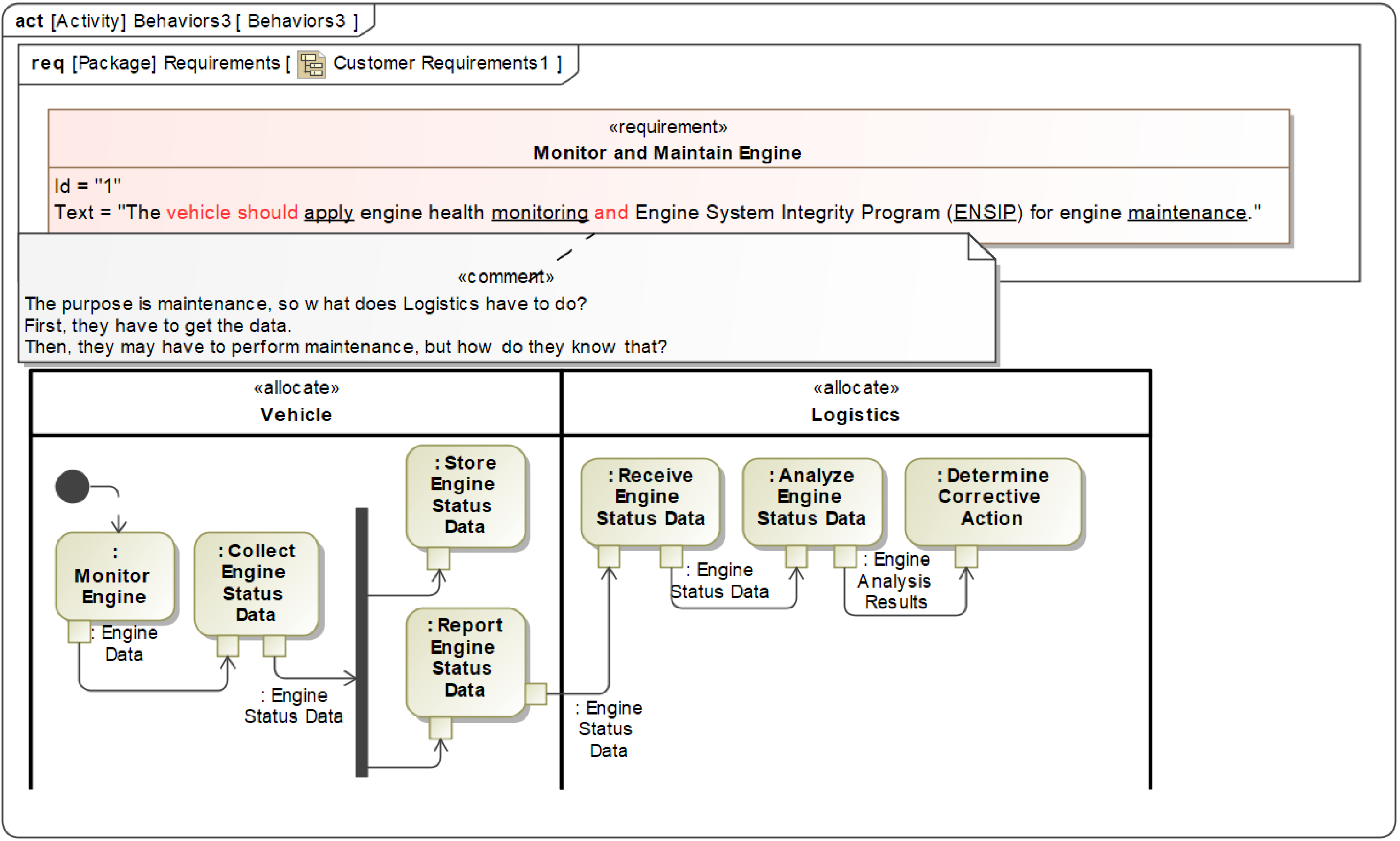
«requirement»
Monitor and Maintain Engine

Id = "1"
Text = "The vehicle should apply engine health monitoring and Engine System Integrity Program (ENSIP) for engine maintenance."

«comment»
The purpose is maintenance, so w hat does Logistics have to do?
First, they have to get the data.

«allocate»
Vehicle
«allocate»
Logistics
: Monitor Engine
: Engine Data
: Collect Engine Status Data
: Engine Status Data
: Store Engine Status Data
: Report Engine Status Data
: Receive Engine Status Data
: Engine Status Data

Engine Example



act [Activity] Behaviors3 [Behaviors3]

req [Package] Requirements [Customer Requirements1]

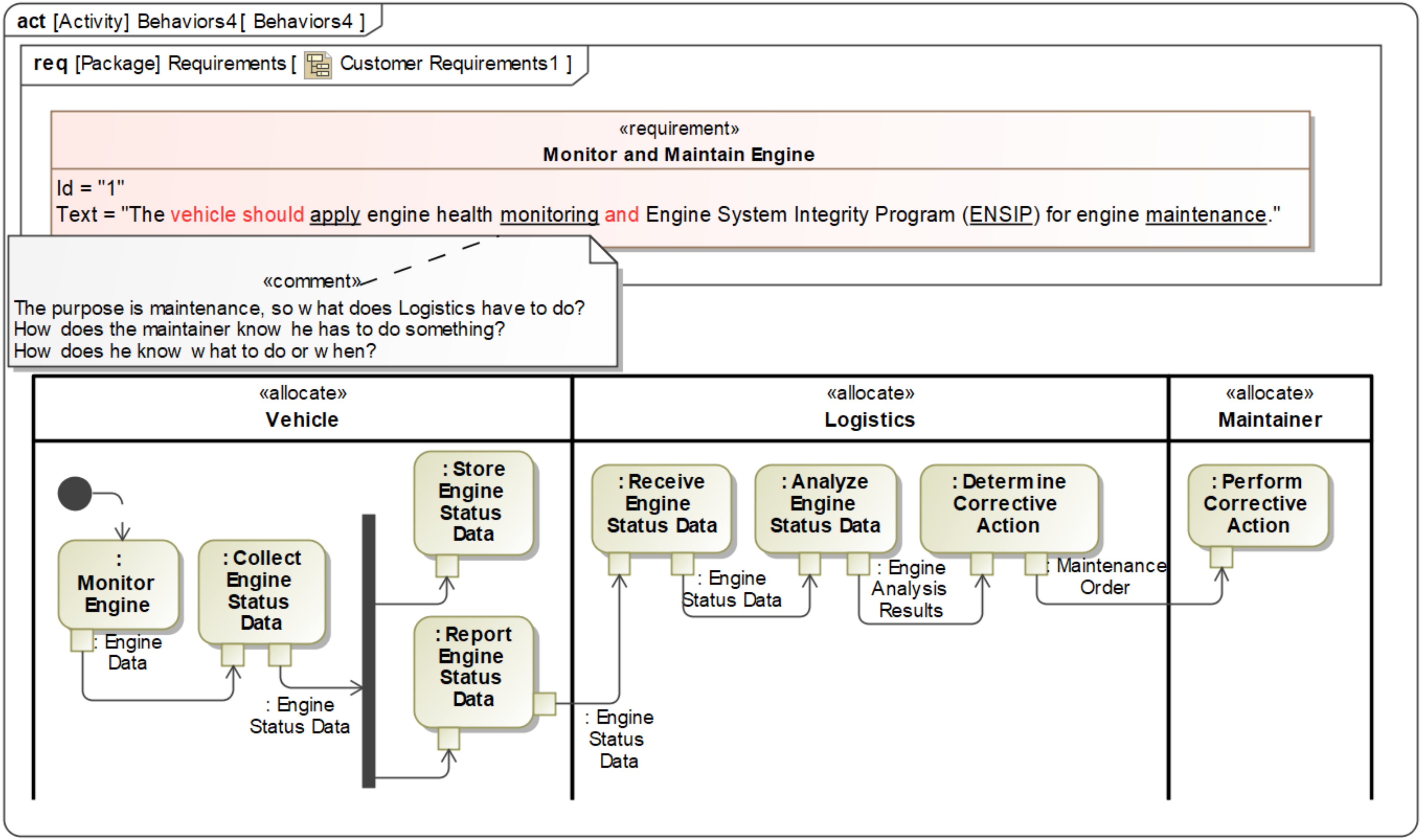
«requirement»
Monitor and Maintain Engine

Id = "1"
Text = "The vehicle should apply engine health monitoring and Engine System Integrity Program (ENSIP) for engine maintenance."

«comment»
The purpose is maintenance, so what does Logistics have to do?
First, they have to get the data.
Then, they may have to perform maintenance, but how do they know that?

«allocate»
Vehicle
«allocate»
Logistics
: Monitor Engine
: Engine Data
: Collect Engine Status Data
: Engine Status Data
: Store Engine Status Data
: Report Engine Status Data
: Engine Status Data
: Receive Engine Status Data
: Engine Status Data
: Analyze Engine Status Data
: Engine Analysis Results
: Determine Corrective Action

Engine Example



act [Activity] Behaviors4 [Behaviors4]

req [Package] Requirements [Customer Requirements1]

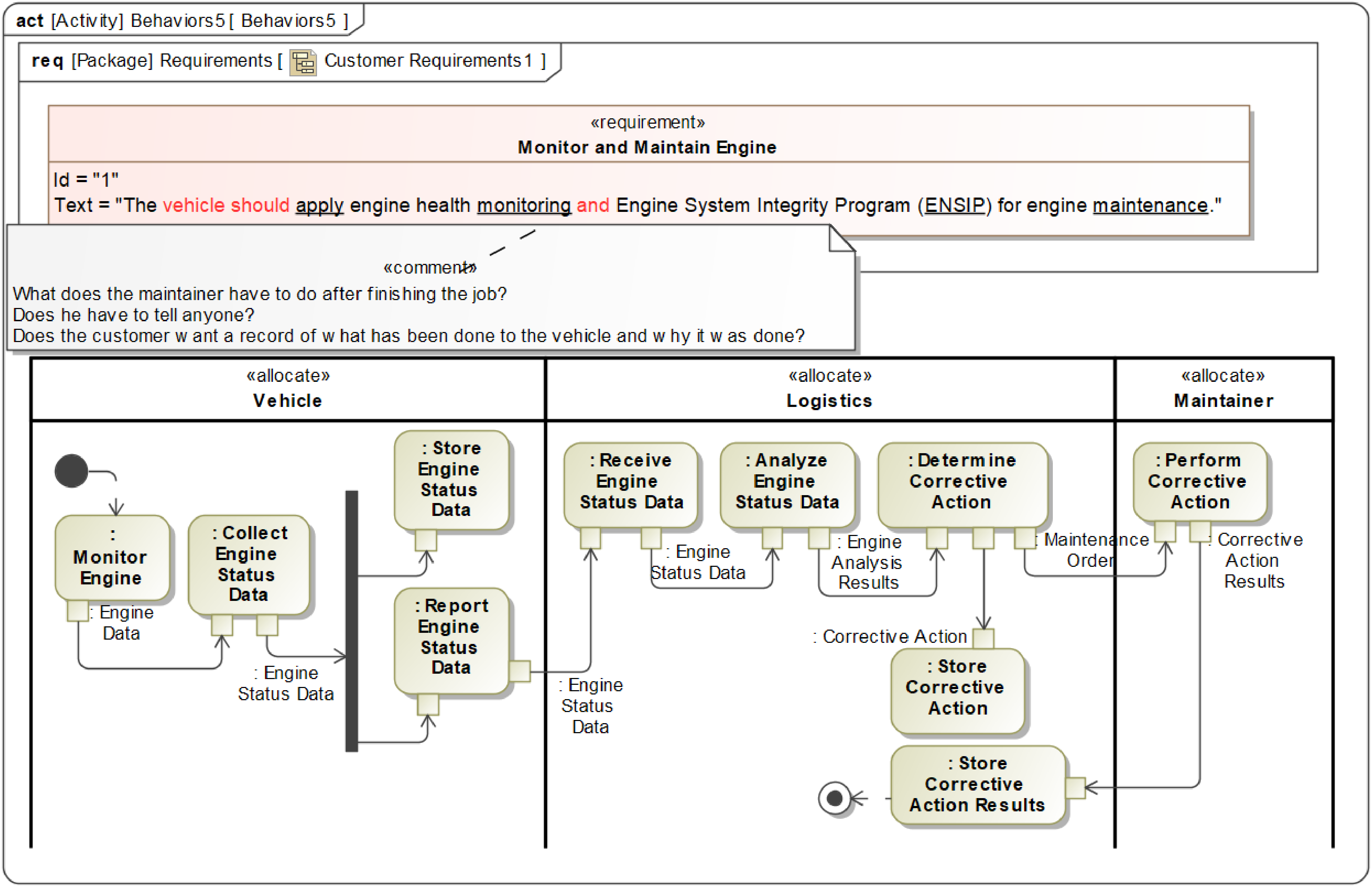
«requirement»
Monitor and Maintain Engine

Id = "1"
Text = "The vehicle should apply engine health monitoring and Engine System Integrity Program (ENSIP) for engine maintenance."

«comment»
The purpose is maintenance, so w hat does Logistics have to do?
How does the maintainer know he has to do something?
How does he know w hat to do or w hen?

«allocate» Vehicle
«allocate» Logistics
«allocate» Maintainer
: Monitor Engine
: Engine Data
: Collect Engine Status Data
: Engine Status Data
: Store Engine Status Data
: Report Engine Status Data
: Engine Status Data
: Receive Engine Status Data
: Engine Status Data
: Analyze Engine Status Data
: Engine Analysis Results
: Determine Corrective Action
: Maintenance Order
: Perform Corrective Action

Engine Example



act [Activity] Behaviors5 [Behaviors5]

req [Package] Requirements [Customer Requirements1]

«requirement»
Monitor and Maintain Engine

Id = "1"
Text = "The vehicle should apply engine health monitoring and Engine System Integrity Program (ENSIP) for engine maintenance."

«comment»
What does the maintainer have to do after finishing the job?
Does he have to tell anyone?
Does the customer w ant a record of w hat has been done to the vehicle and w hy it w as done?

«allocate»
Vehicle

«allocate»
Logistics

«allocate»
Maintainer

: Monitor Engine

: Engine Data

: Collect Engine Status Data

: Engine Status Data

: Store Engine Status Data

: Report Engine Status Data

: Engine Status Data

: Receive Engine Status Data

: Engine Status Data

: Analyze Engine Status Data

: Engine Analysis Results

: Determine Corrective Action

: Maintenance Order

: Corrective Action

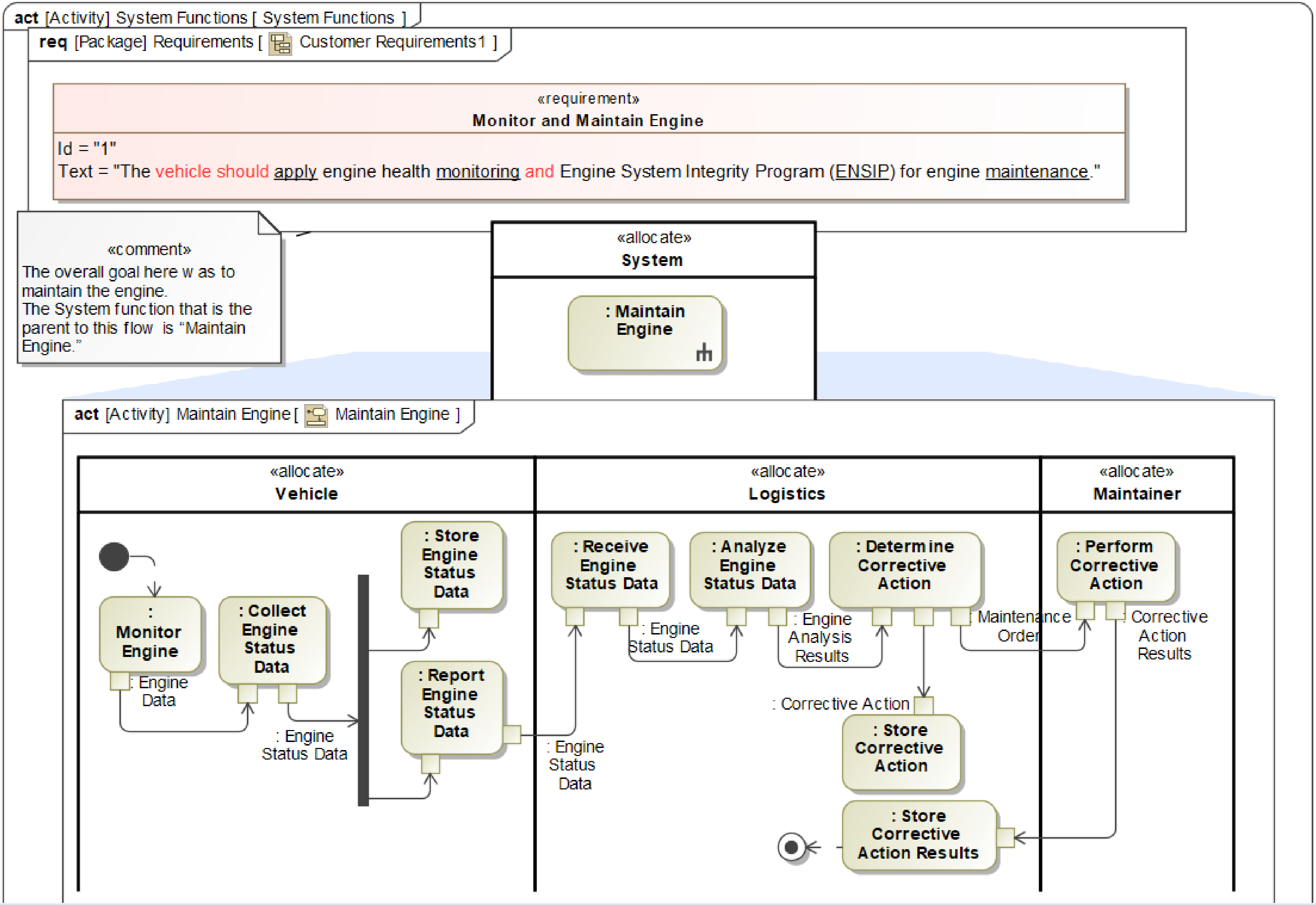
: Store Corrective Action

: Perform Corrective Action

: Corrective Action Results

: Store Corrective Action Results

Engine Example



act [Activity] System Functions [System Functions]

req [Package] Requirements [Customer Requirements1]

«requirement»
Monitor and Maintain Engine

Id = "1"
Text = "The vehicle should apply engine health monitoring and Engine System Integrity Program (ENSIP) for engine maintenance."

«comment»
The overall goal here was to maintain the engine.
The System function that is the parent to this flow is "Maintain Engine."

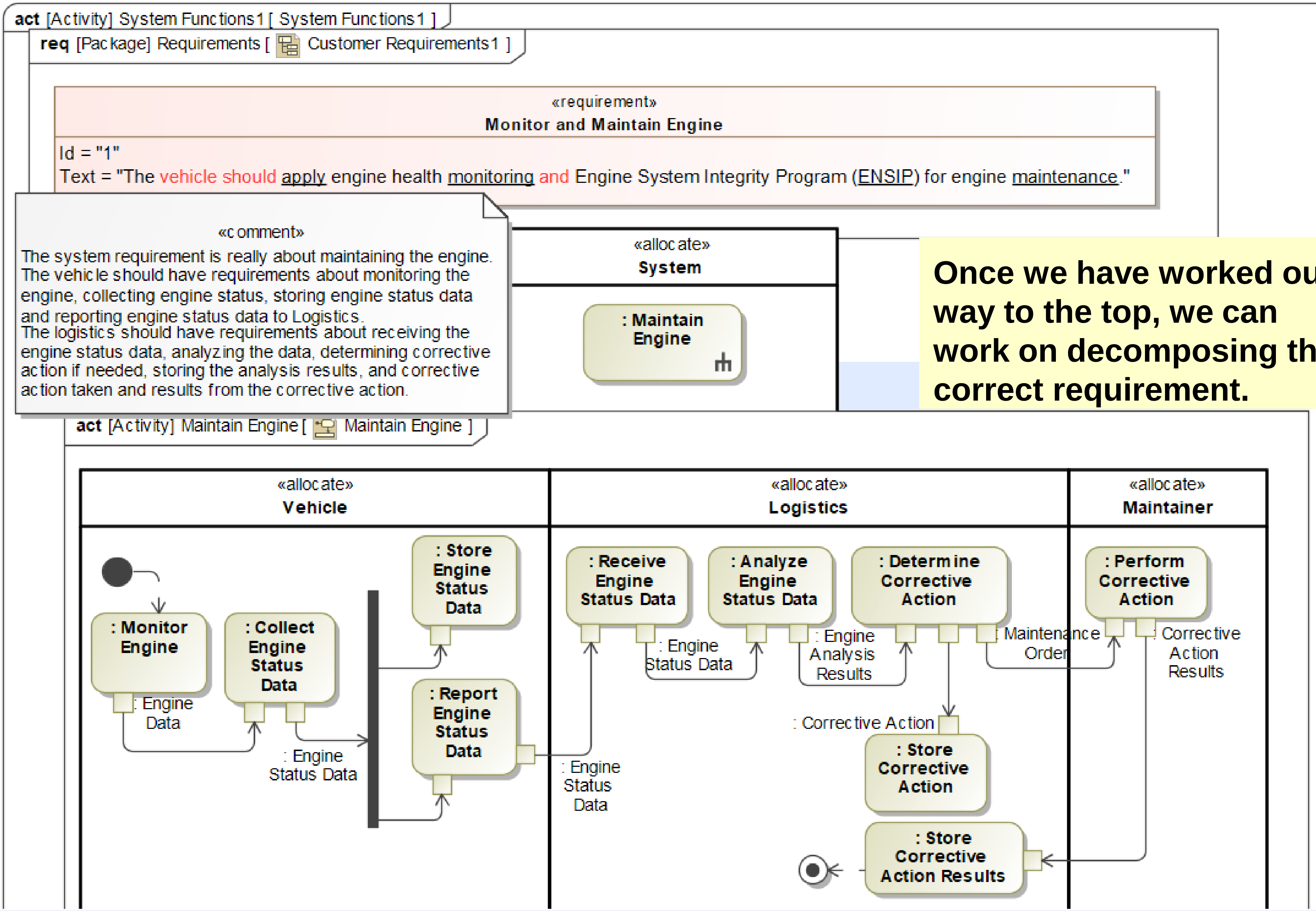
«allocate»
System

: Maintain Engine

act [Activity] Maintain Engine [Maintain Engine]

«allocate» Vehicle
«allocate» Logistics
«allocate» Maintainer
: Monitor Engine
: Engine Data
: Collect Engine Status Data
: Engine Status Data
: Store Engine Status Data
: Report Engine Status Data
: Engine Status Data
: Receive Engine Status Data
: Engine Status Data
: Analyze Engine Status Data
: Engine Analysis Results
: Determine Corrective Action
: Maintenance Order
: Corrective Action
: Store Corrective Action
: Store Corrective Action Results
: Perform Corrective Action
: Corrective Action Results

Engine Example



act [Activity] System Functions1 [System Functions1]

req [Package] Requirements [Customer Requirements1]

«requirement»
Monitor and Maintain Engine

Id = "1"
Text = "The vehicle should apply engine health monitoring and Engine System Integrity Program (ENSIP) for engine maintenance."

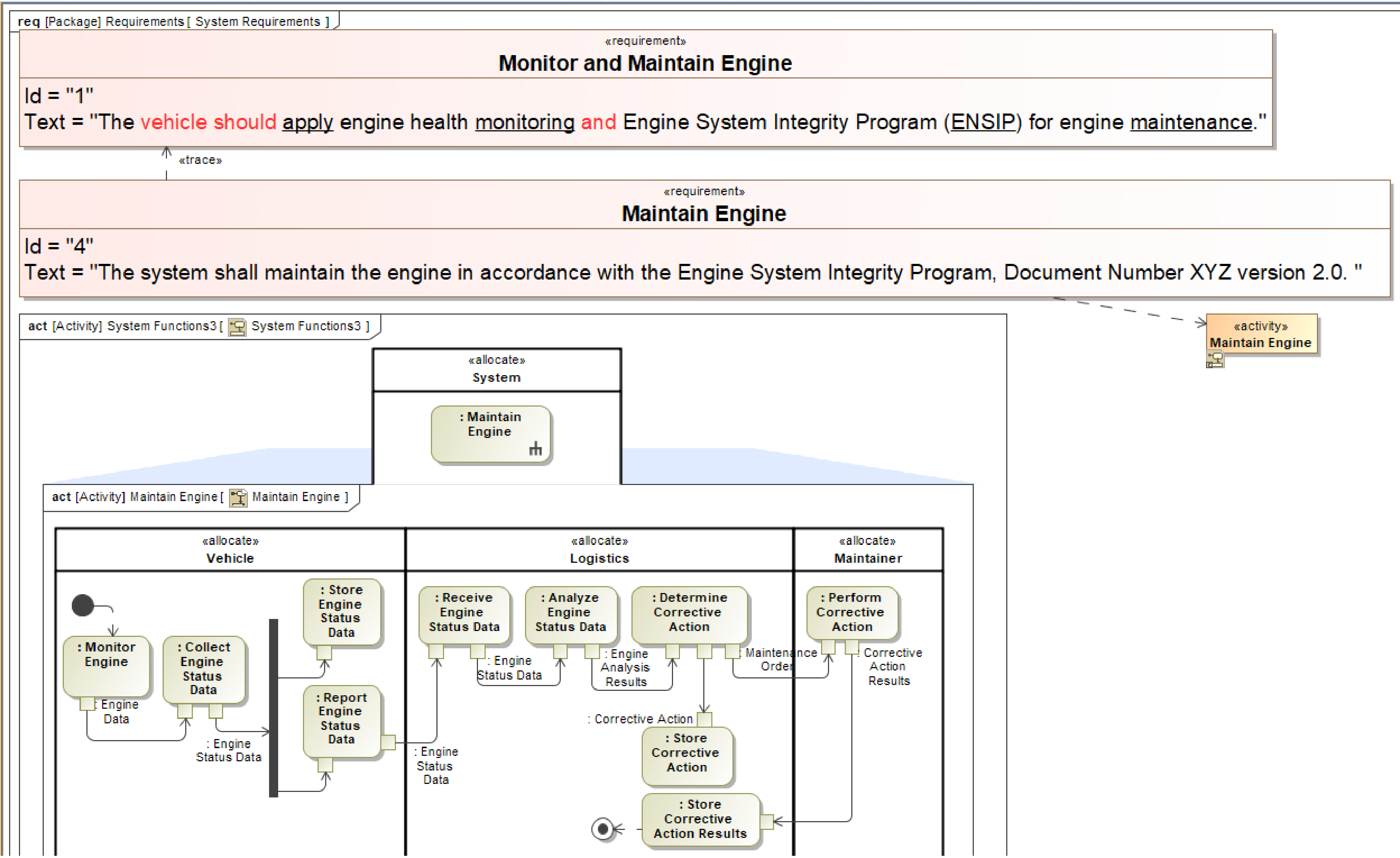
«comment»
The system requirement is really about maintaining the engine. The vehicle should have requirements about monitoring the engine, collecting engine status, storing engine status data and reporting engine status data to Logistics.
The logistics should have requirements about receiving the engine status data, analyzing the data, determining corrective action if needed, storing the analysis results, and corrective action taken and results from the corrective action.

act [Activity] Maintain Engine [Maintain Engine]

«allocate»
System
: Maintain Engine
«allocate»
Vehicle
«allocate»
Logistics
«allocate»
Maintainer
: Monitor Engine
: Engine Data
: Collect Engine Status Data
: Engine Status Data
: Store Engine Status Data
: Report Engine Status Data
: Engine Status Data
: Receive Engine Status Data
: Engine Status Data
: Analyze Engine Status Data
: Engine Analysis Results
: Determine Corrective Action
: Maintenance Order
: Corrective Action
: Store Corrective Action
: Store Corrective Action Results
: Perform Corrective Action
: Corrective Action Results

Once we have worked our way to the top, we can work on decomposing the correct requirement.

Engine Example



req [Package] Requirements [System Requirements]

«requirement»
Monitor and Maintain Engine

Id = "1"
Text = "The vehicle should apply engine health monitoring and Engine System Integrity Program (ENSIP) for engine maintenance."

«trace»

«requirement»
Maintain Engine

Id = "4"
Text = "The system shall maintain the engine in accordance with the Engine System Integrity Program, Document Number XYZ version 2.0. "

«activity»
Maintain Engine

act [Activity] System Functions3 [System Functions3]

«allocate»
System

: Maintain Engine

act [Activity] Maintain Engine [Maintain Engine]

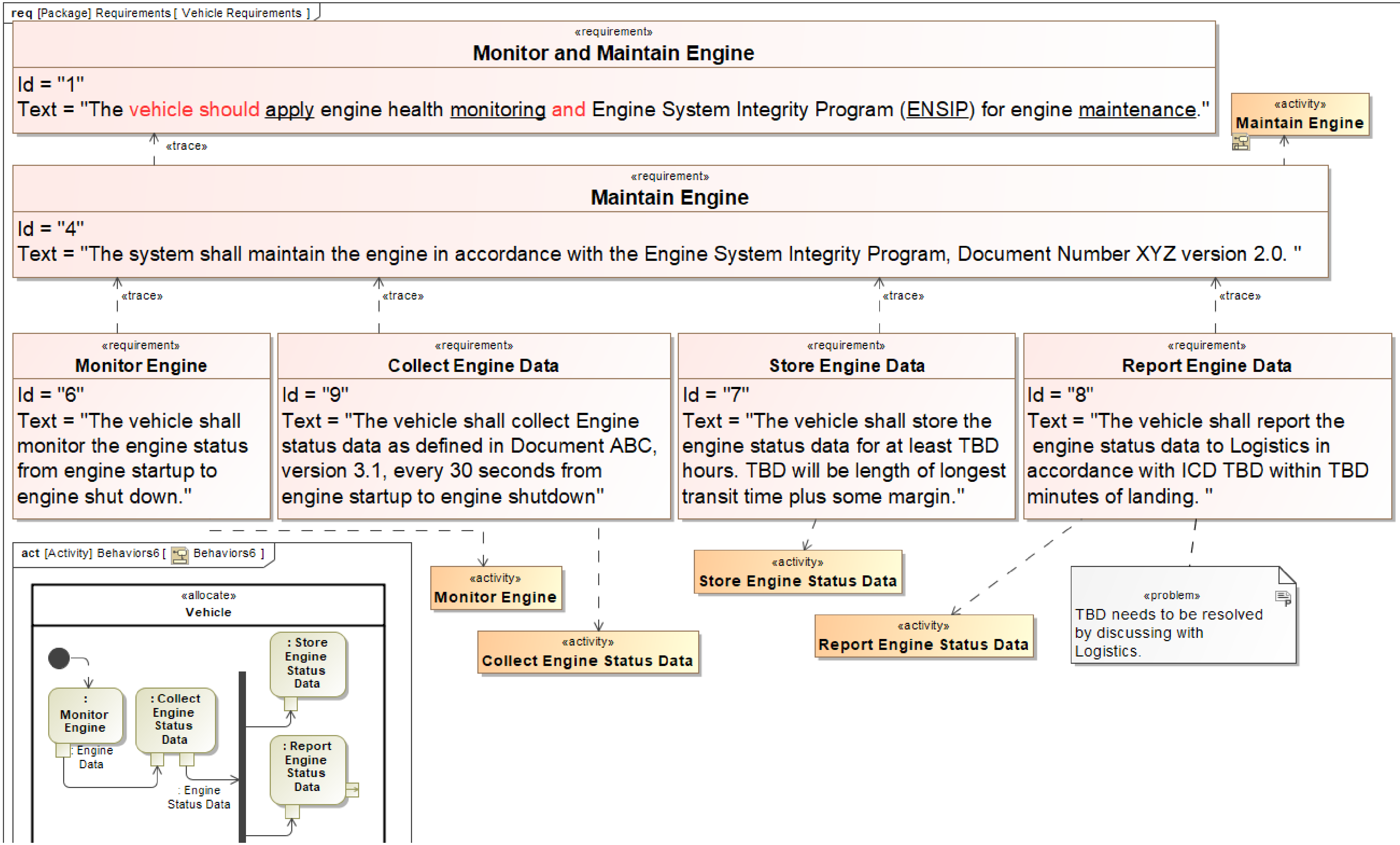
«allocate»
Vehicle

«allocate»
Logistics

«allocate»
Maintainer

: Monitor Engine
: Engine Data
: Collect Engine Status Data
: Engine Status Data
: Store Engine Status Data
: Report Engine Status Data
: Engine Status Data
: Receive Engine Status Data
: Engine Status Data
: Analyze Engine Status Data
: Engine Analysis Results
: Determine Corrective Action
: Maintenance Order
: Corrective Action
: Store Corrective Action
: Store Corrective Action Results
: Perform Corrective Action
: Corrective Action Results

Engine Example



req [Package] Requirements [Vehicle Requirements]

«requirement»
Monitor and Maintain Engine
Id = "1"
Text = "The vehicle should apply engine health monitoring and Engine System Integrity Program (ENSIP) for engine maintenance."

«requirement»
Maintain Engine
Id = "4"
Text = "The system shall maintain the engine in accordance with the Engine System Integrity Program, Document Number XYZ version 2.0. "

«requirement»
Monitor Engine
Id = "6"
Text = "The vehicle shall monitor the engine status from engine startup to engine shut down."

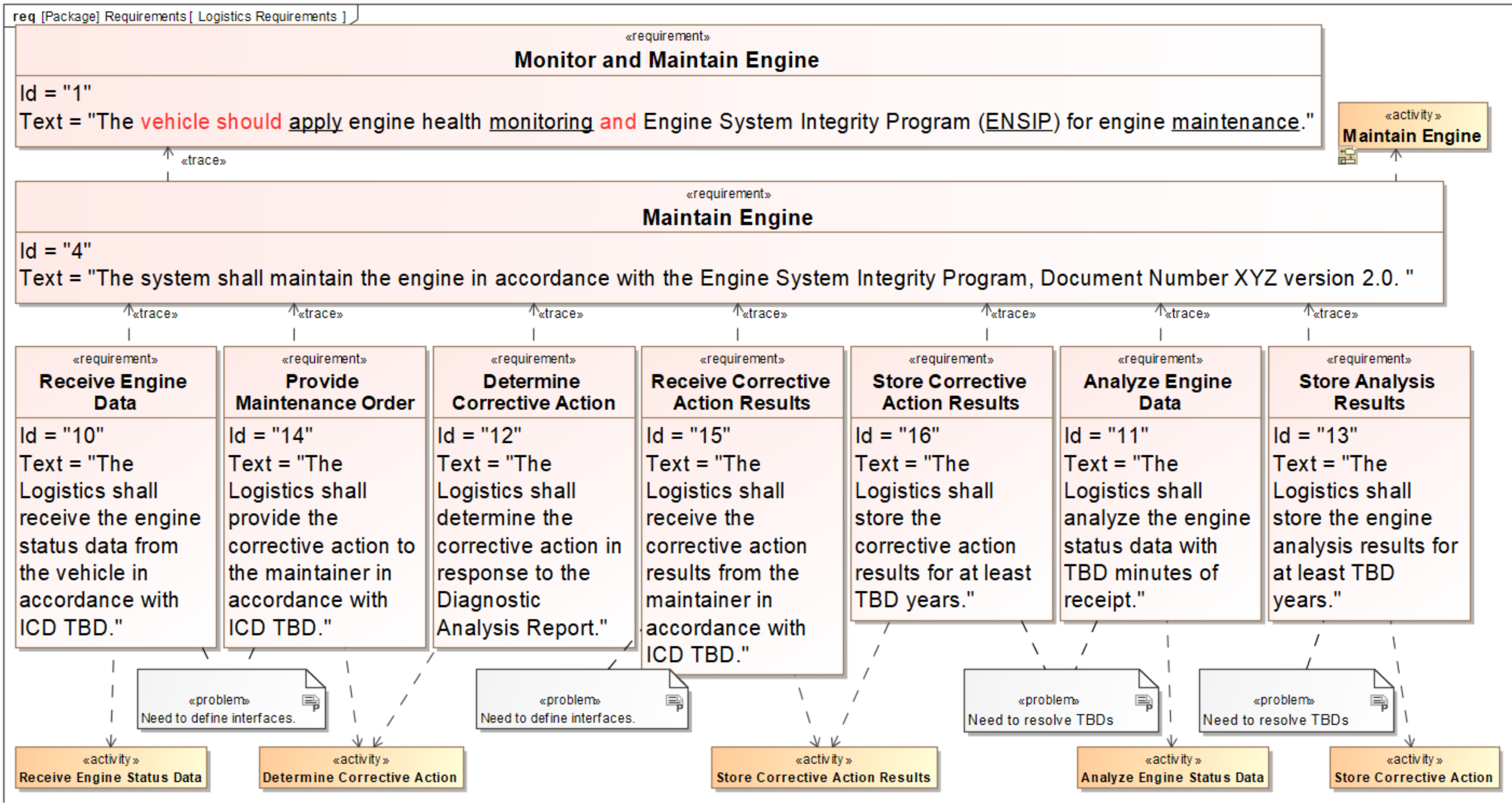
«requirement»
Collect Engine Data
Id = "9"
Text = "The vehicle shall collect Engine status data as defined in Document ABC, version 3.1, every 30 seconds from engine startup to engine shutdown"

«requirement»
Store Engine Data
Id = "7"
Text = "The vehicle shall store the engine status data for at least TBD hours. TBD will be length of longest transit time plus some margin."

«requirement»
Report Engine Data
Id = "8"
Text = "The vehicle shall report the engine status data to Logistics in accordance with ICD TBD within TBD minutes of landing. "

«problem»
TBD needs to be resolved by discussing with Logistics.

Engine Example



req [Package] Requirements [Logistics Requirements]

«requirement»
Monitor and Maintain Engine
Id = "1"
Text = "The vehicle should apply engine health monitoring and Engine System Integrity Program (ENSIP) for engine maintenance."

«activity»
Maintain Engine

«trace»

«requirement»
Maintain Engine
Id = "4"
Text = "The system shall maintain the engine in accordance with the Engine System Integrity Program, Document Number XYZ version 2.0. "

«trace»

«requirement»
Receive Engine Data
Id = "10"
Text = "The Logistics shall receive the engine status data from the vehicle in accordance with ICD TBD."

«requirement»
Provide Maintenance Order
Id = "14"
Text = "The Logistics shall provide the corrective action to the maintainer in accordance with ICD TBD."

«requirement»
Determine Corrective Action
Id = "12"
Text = "The Logistics shall determine the corrective action in response to the Diagnostic Analysis Report."

«requirement»
Receive Corrective Action Results
Id = "15"
Text = "The Logistics shall receive the corrective action results from the maintainer in accordance with ICD TBD."

«requirement»
Store Corrective Action Results
Id = "16"
Text = "The Logistics shall store the corrective action results for at least TBD years."

«requirement»
Analyze Engine Data
Id = "11"
Text = "The Logistics shall analyze the engine status data with TBD minutes of receipt."

«requirement»
Store Analysis Results
Id = "13"
Text = "The Logistics shall store the engine analysis results for at least TBD years."

«problem»
Need to define interfaces.

«problem»
Need to define interfaces.

«problem»
Need to resolve TBDs

«problem»
Need to resolve TBDs

«activity»
Receive Engine Status Data

«activity»
Determine Corrective Action

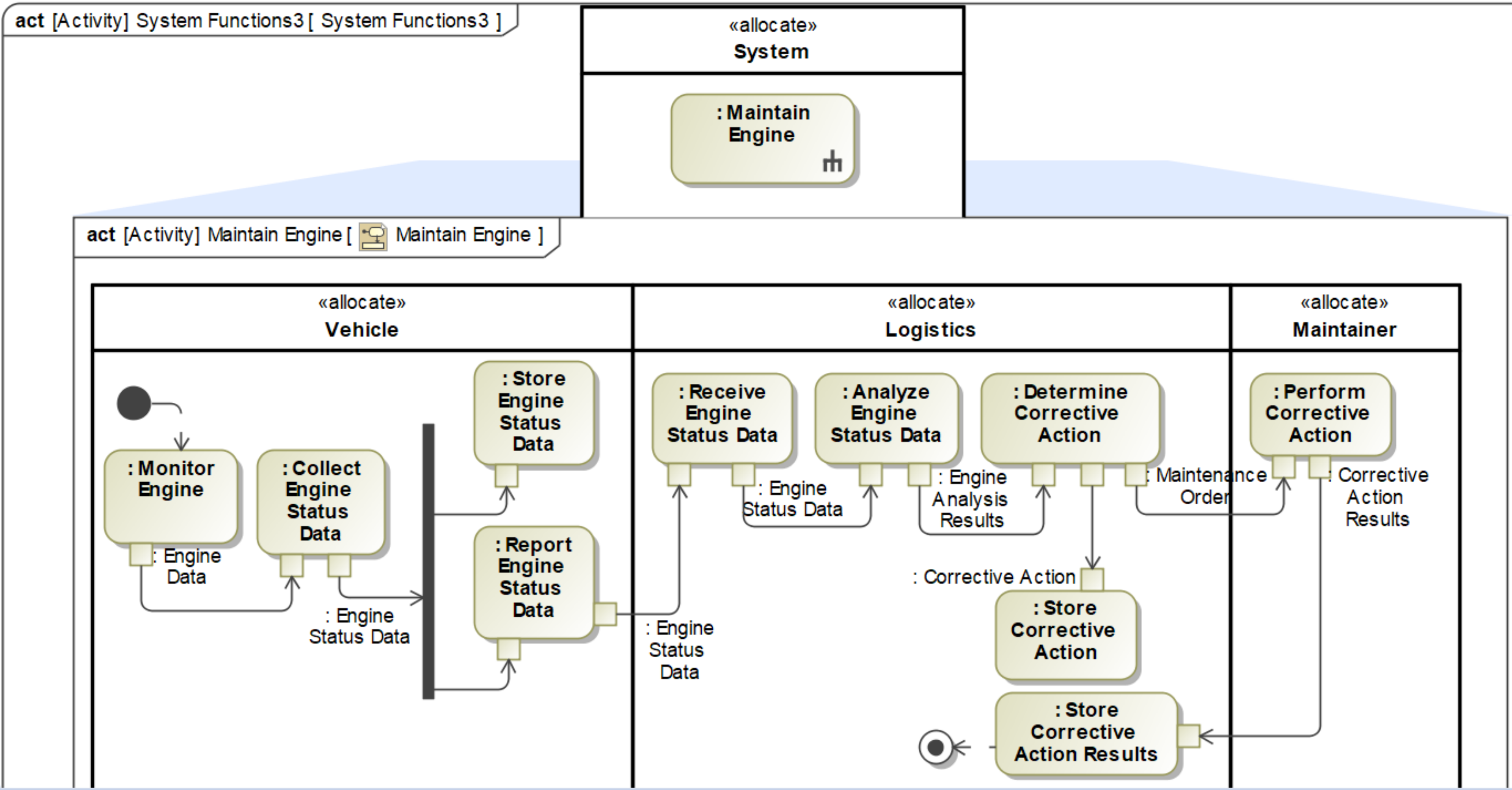
«activity»
Store Corrective Action Results

«activity»
Analyze Engine Status Data

«activity»
Store Corrective Action

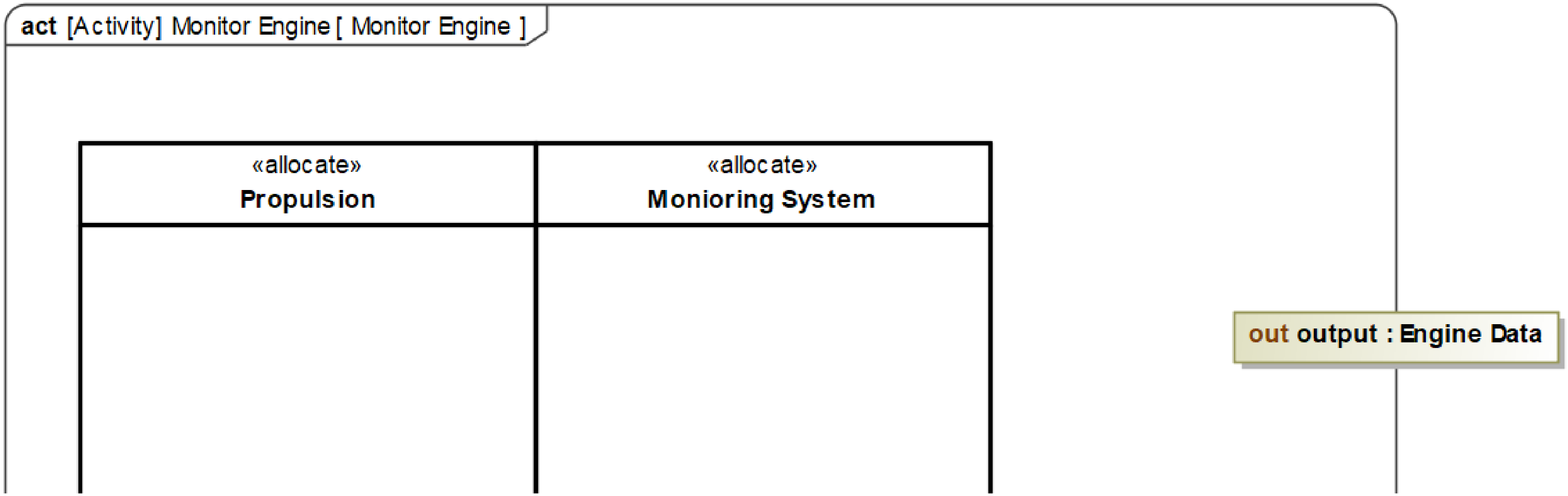
Engine Example

- The vehicle and logistics requirements need to be decomposed to vehicle and logistics subsystem requirements and subsystems.
- Pick a function and work on decomposing it


act [Activity] System Functions3 [System Functions3]
«allocate»
System
: Maintain Engine
act [Activity] Maintain Engine [Maintain Engine]
«allocate»
Vehicle
«allocate»
Logistics
«allocate»
Maintainer
: Monitor Engine
: Engine Data
: Collect Engine Status Data
: Engine Status Data
: Store Engine Status Data
: Report Engine Status Data
: Engine Status Data
: Receive Engine Status Data
: Engine Status Data
: Analyze Engine Status Data
: Engine Analysis Results
: Determine Corrective Action
: Maintenance Order
: Corrective Action
: Store Corrective Action
: Store Corrective Action Results
: Perform Corrective Action
: Corrective Action Results

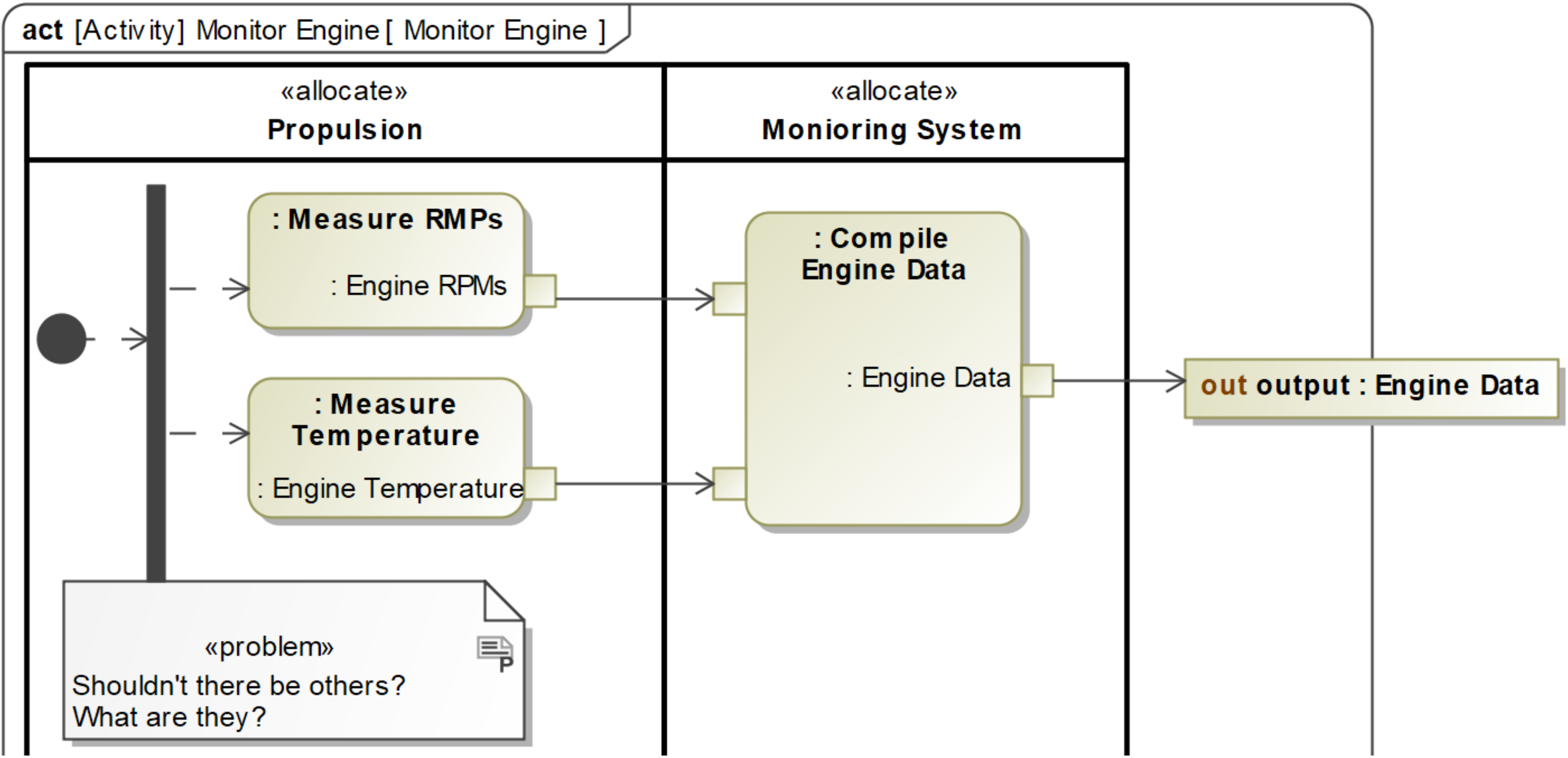
Monitor Engine

- What does it take to monitor and Engine?
- Create functions that consume the inputs and produce the outputs
- E.g. What creates the Engine data for the Monitor Engine Function?
- Where does that function get its data from?



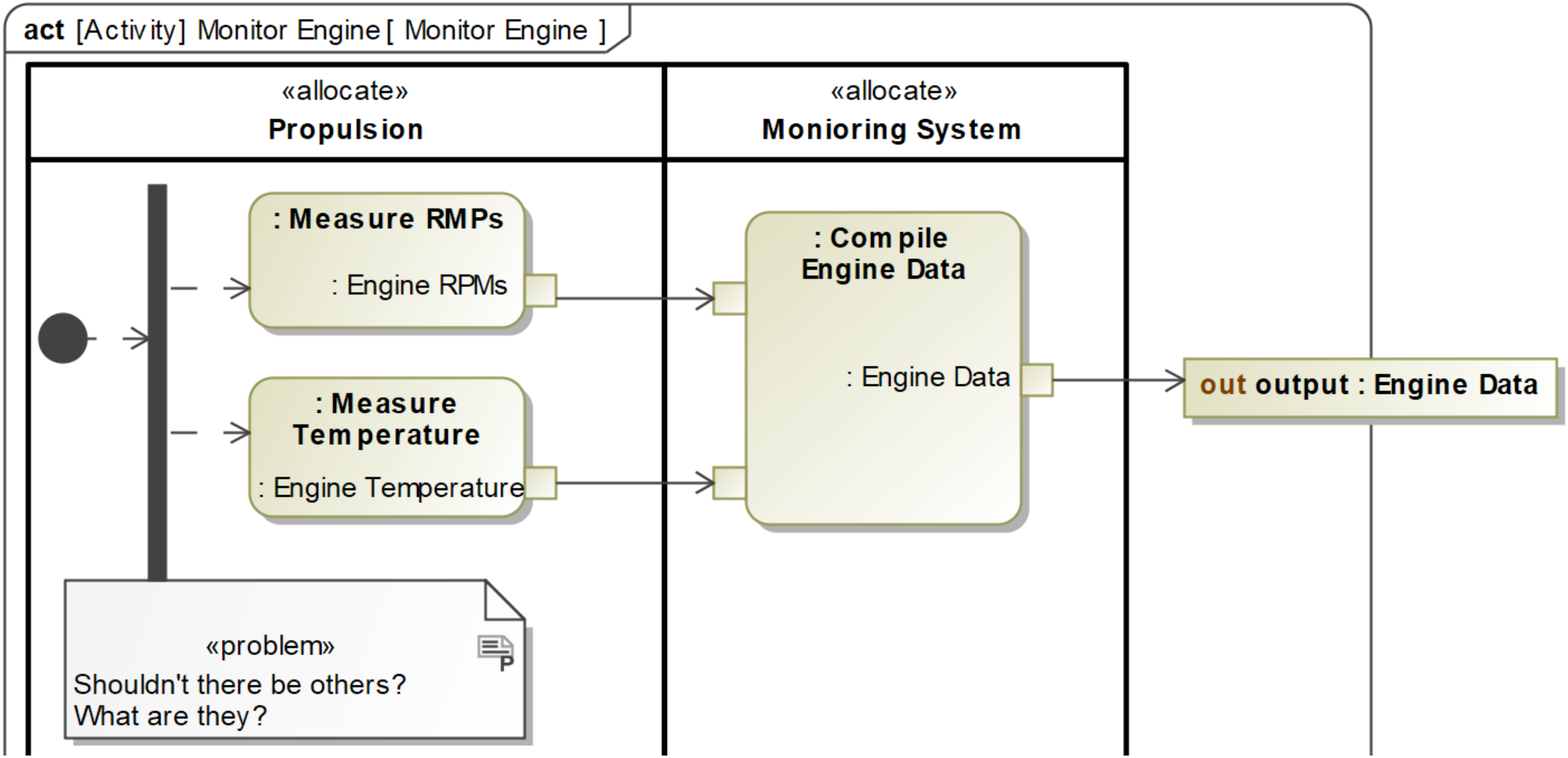
Monitor Engine

- **This is a good start**
- **What else is needed?**



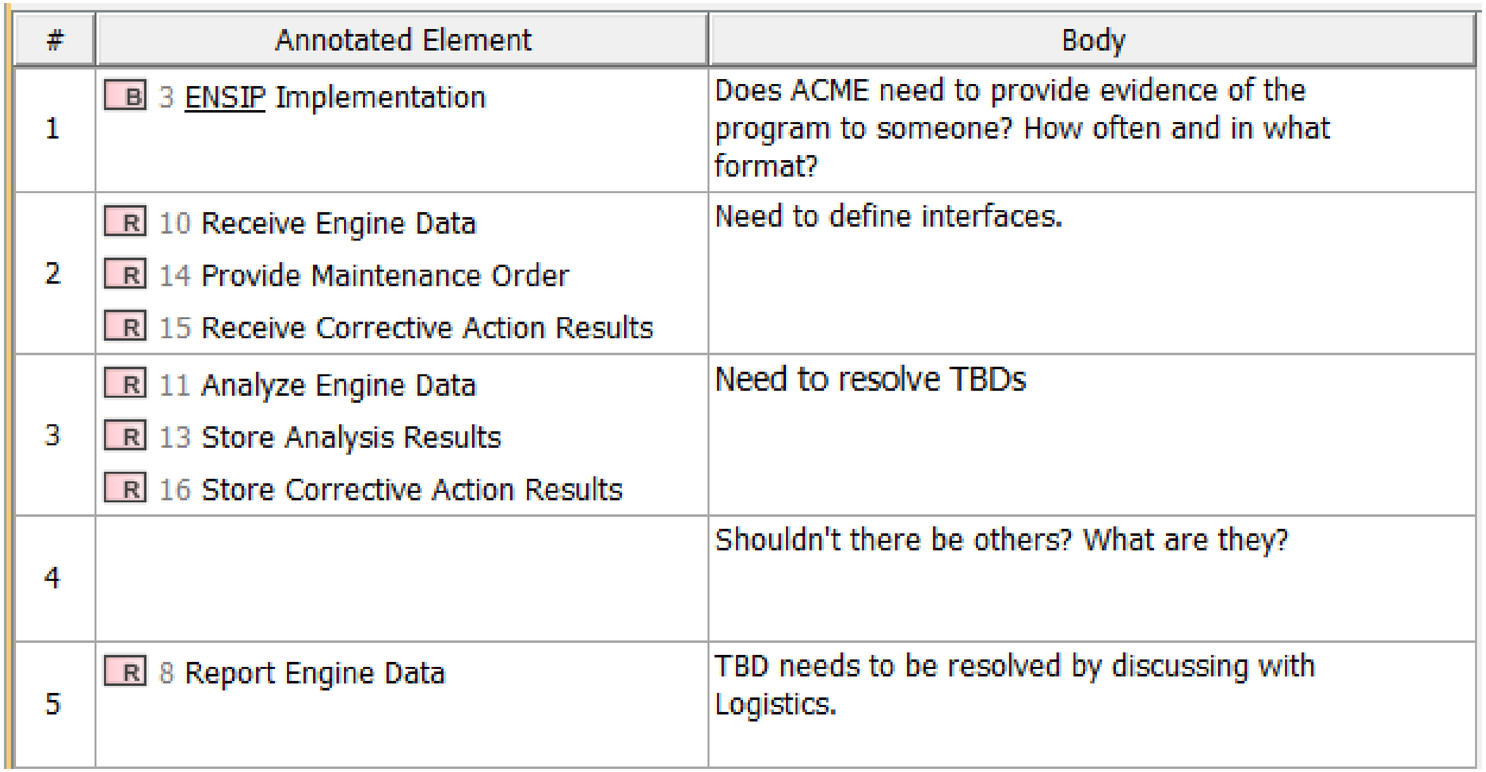
Decomposition for Monitor Engine

- **Keep doing this until you have enough information to provide a specification to a supplier to buy a part or software application or until you have enough information to build it internally.**



Problems

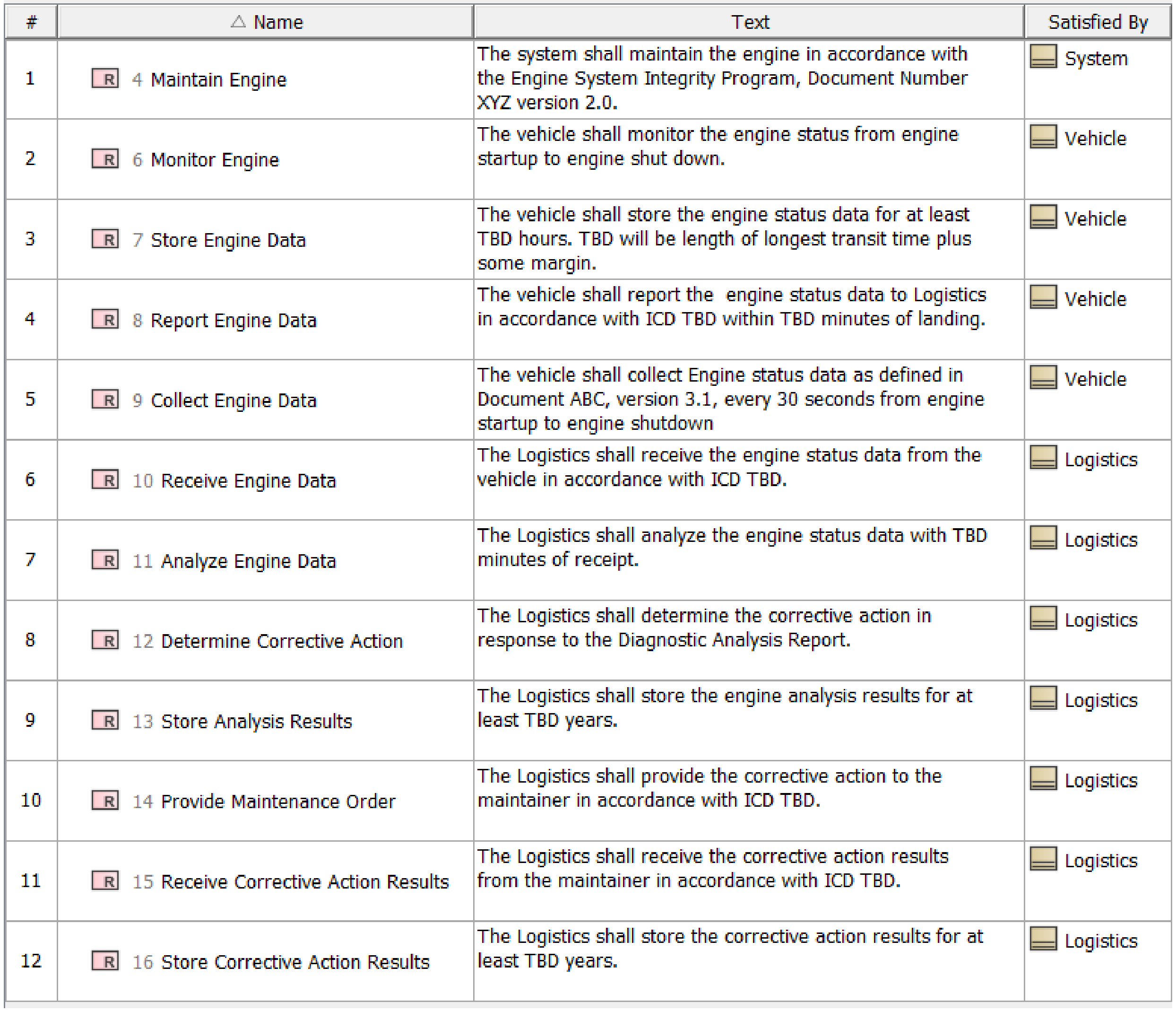
- As we worked, we kept track of the problems
- Its ok to that we don’t know the answers right away
  - We may need to get some help from another team to resolve these
  - We may need to have some conversations with the customer



#	Annotated Element	Body
1	B 3 ENSIP Implementation	Does ACME need to provide evidence of the program to someone? How often and in what format?
2	R 10 Receive Engine Data R 14 Provide Maintenance Order R 15 Receive Corrective Action Results	Need to define interfaces.
3	R 11 Analyze Engine Data R 13 Store Analysis Results R 16 Store Corrective Action Results	Need to resolve TBDs
4		Shouldn't there be others? What are they?
5	R 8 Report Engine Data	TBD needs to be resolved by discussing with Logistics.

Traceability

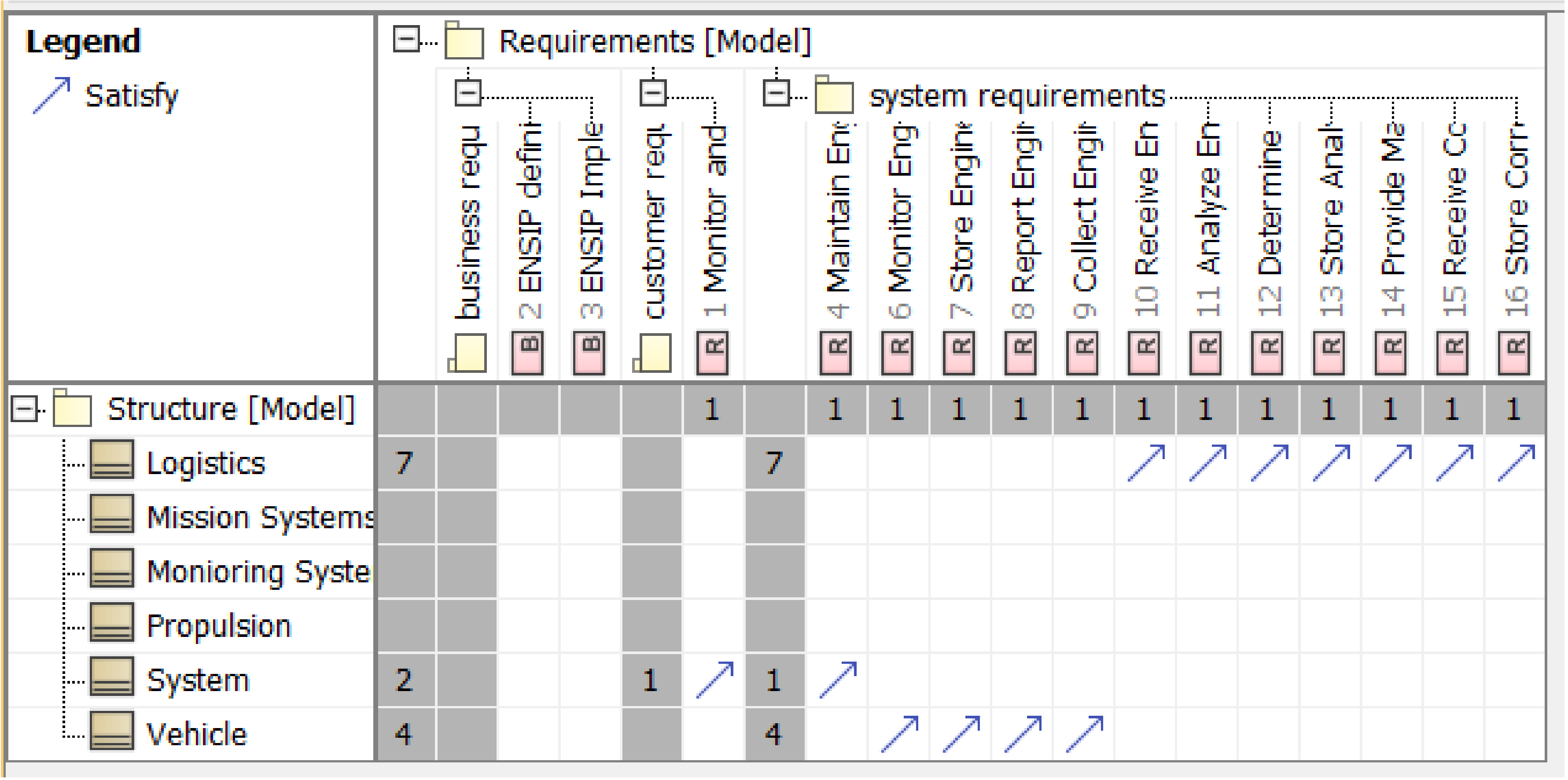
- **We also kept track of which system satisfies which requirement**
- **All requirements need to be satisfied**



#	Name	Text	Satisfied By
1	R 4 Maintain Engine	The system shall maintain the engine in accordance with the Engine System Integrity Program, Document Number XYZ version 2.0.	System
2	R 6 Monitor Engine	The vehicle shall monitor the engine status from engine startup to engine shut down.	Vehicle
3	R 7 Store Engine Data	The vehicle shall store the engine status data for at least TBD hours. TBD will be length of longest transit time plus some margin.	Vehicle
4	R 8 Report Engine Data	The vehicle shall report the engine status data to Logistics in accordance with ICD TBD within TBD minutes of landing.	Vehicle
5	R 9 Collect Engine Data	The vehicle shall collect Engine status data as defined in Document ABC, version 3.1, every 30 seconds from engine startup to engine shutdown	Vehicle
6	R 10 Receive Engine Data	The Logistics shall receive the engine status data from the vehicle in accordance with ICD TBD.	Logistics
7	R 11 Analyze Engine Data	The Logistics shall analyze the engine status data with TBD minutes of receipt.	Logistics
8	R 12 Determine Corrective Action	The Logistics shall determine the corrective action in response to the Diagnostic Analysis Report.	Logistics
9	R 13 Store Analysis Results	The Logistics shall store the engine analysis results for at least TBD years.	Logistics
10	R 14 Provide Maintenance Order	The Logistics shall provide the corrective action to the maintainer in accordance with ICD TBD.	Logistics
11	R 15 Receive Corrective Action Results	The Logistics shall receive the corrective action results from the maintainer in accordance with ICD TBD.	Logistics
12	R 16 Store Corrective Action Results	The Logistics shall store the corrective action results for at least TBD years.	Logistics

Progress

- We can check that all requirements are being satisfied easily
- We can also see if all the systems are satisfying a requirement
- Gaps here indicate unfinished work or unnecessary systems



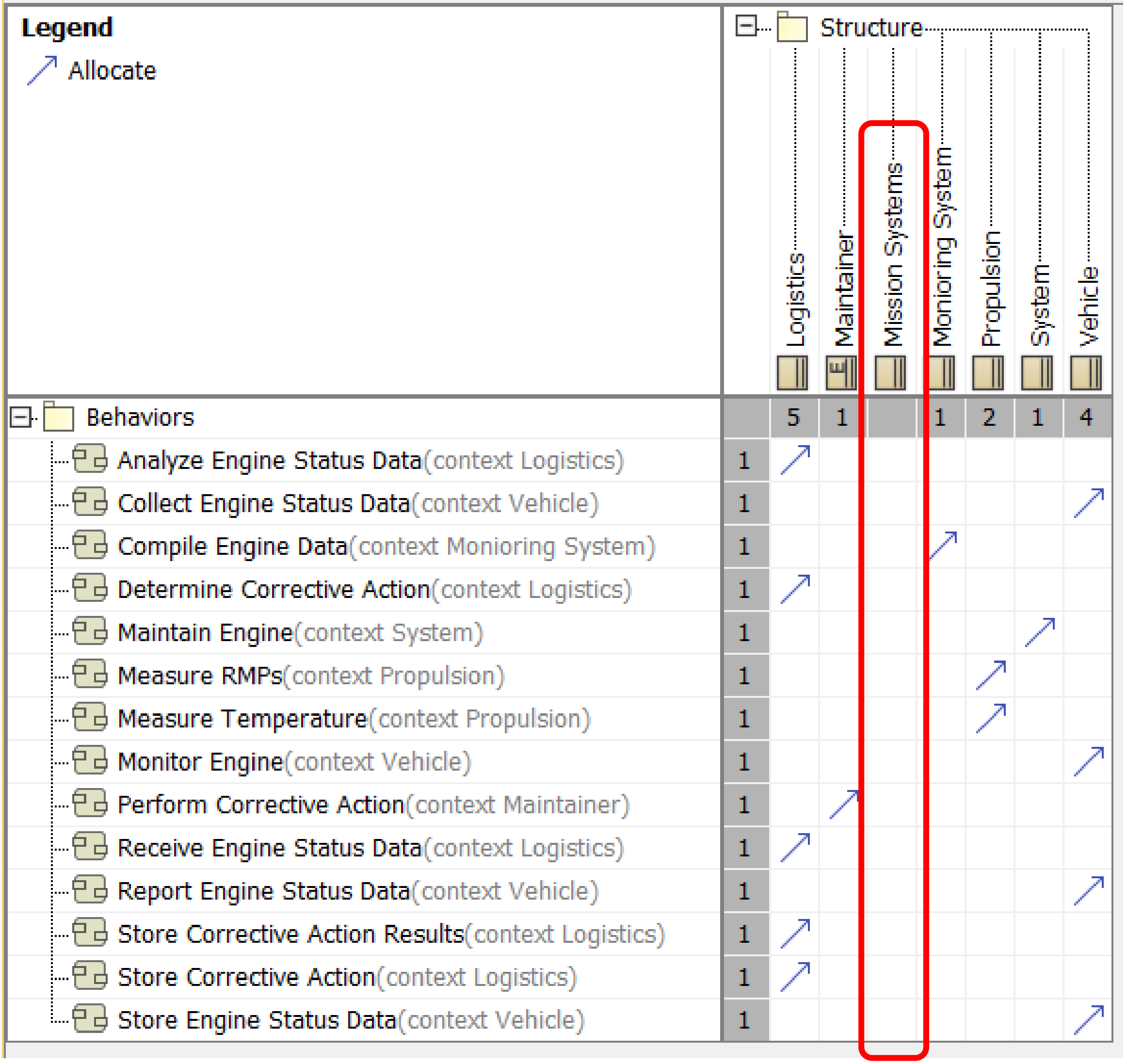
Legend

↗ Satisfy

	Requirements [Model]	business requ	2 ENSIP defini	3 ENSIP Imple	customer requ	1 Monitor and	system requirements	4 Maintain En	6 Monitor Eng	7 Store Engine	8 Report Engir	9 Collect Engir	10 Receive En	11 Analyze En	12 Determine	13 Store Anal	14 Provide Ma	15 Receive Cc	16 Store Corr
Structure [Model]						1		1	1	1	1	1	1	1	1	1	1	1	1
Logistics	7						7						↗	↗	↗	↗	↗	↗	↗
Mission Systems																			
Monioring Syste																			
Propulsion																			
System	2				1	↗	1	↗											
Vehicle	4						4		↗	↗	↗	↗							

Allocations

- The system behaviors need to be allocated to a system that will implement them
- If a system doesn't have a function, reconsider why you need that system
- A team will be assigned to create the products and if a function isn't allocated to their system, they won't address it

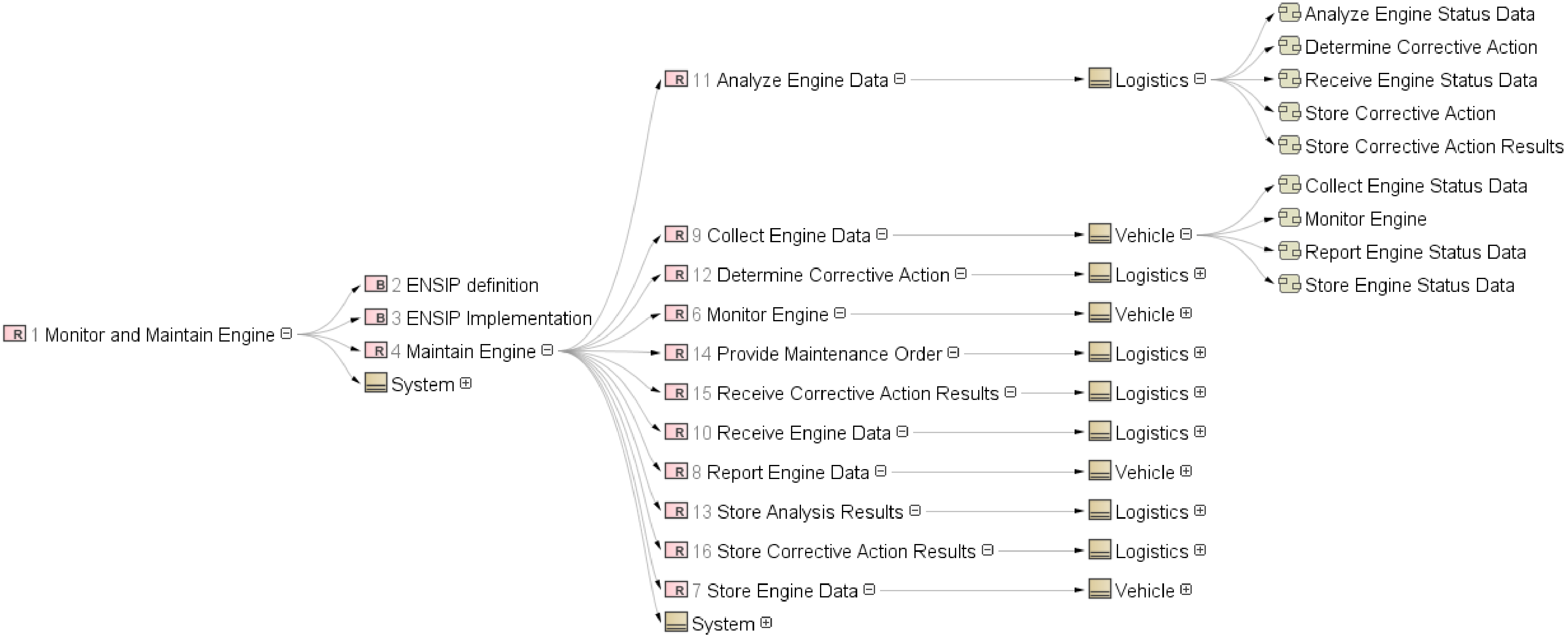


Legend
↗ Allocate

Behaviors		Logistics	Maintainer	Mission Systems	Monioring System	Propulsion	System	Vehicle
		5	1		1	2	1	4
Analyze Engine Status Data(context Logistics)	1	↗						
Collect Engine Status Data(context Vehicle)	1							↗
Compile Engine Data(context Monioring System)	1				↗			
Determine Corrective Action(context Logistics)	1	↗						
Maintain Engine(context System)	1						↗	
Measure RMPs(context Propulsion)	1					↗		
Measure Temperature(context Propulsion)	1					↗		
Monitor Engine(context Vehicle)	1							↗
Perform Corrective Action(context Maintainer)	1		↗					
Receive Engine Status Data(context Logistics)	1	↗						
Report Engine Status Data(context Vehicle)	1							↗
Store Corrective Action Results(context Logistics)	1	↗						
Store Corrective Action(context Logistics)	1	↗						
Store Engine Status Data(context Vehicle)	1							↗

Impact Analyses

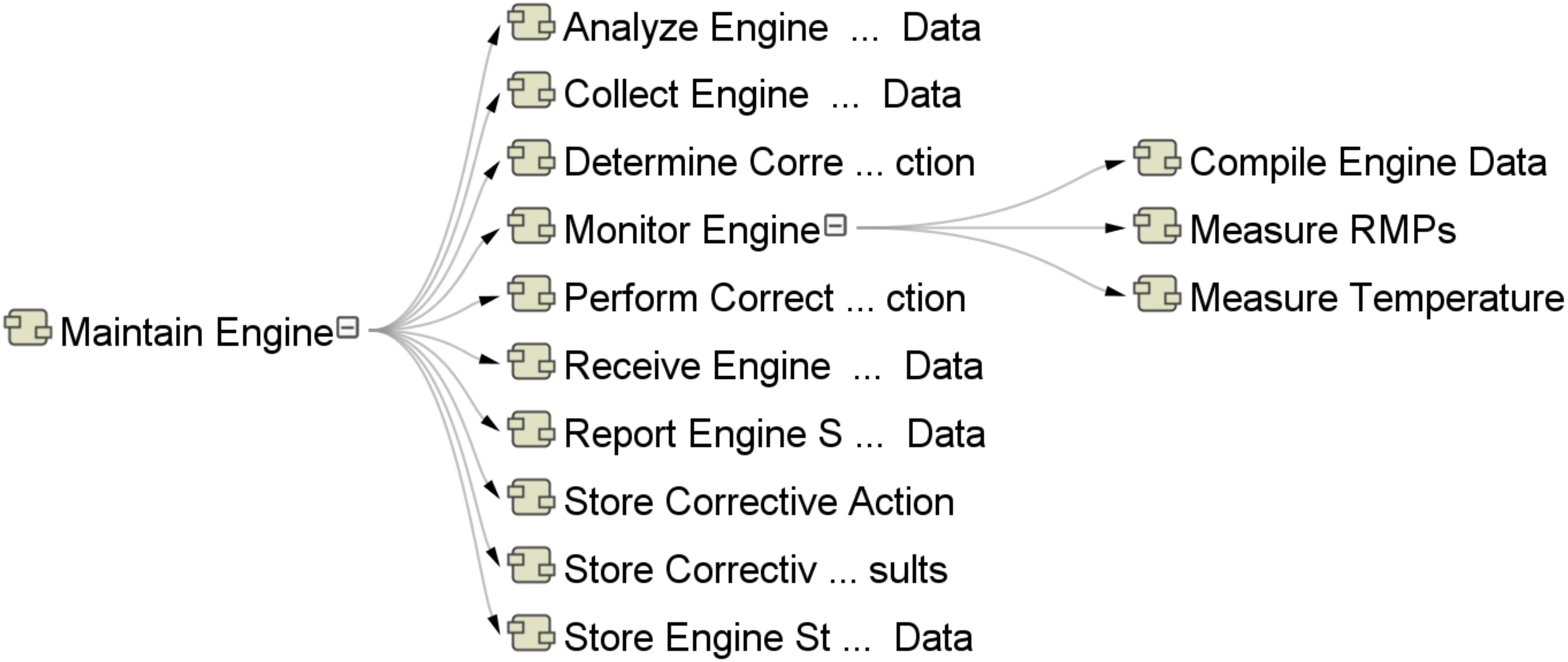
- **The relationships can be used to determine impacts if a requirement were to change**
- **This can happen because of customers changing their mind, evolving environment, and defects**



R 1 Monitor and Maintain Engine ⊟
B 2 ENSIP definition
B 3 ENSIP Implementation
R 4 Maintain Engine ⊟
System ⊞
R 11 Analyze Engine Data ⊟
R 9 Collect Engine Data ⊟
R 12 Determine Corrective Action ⊟
R 6 Monitor Engine ⊟
R 14 Provide Maintenance Order ⊟
R 15 Receive Corrective Action Results ⊟
R 10 Receive Engine Data ⊟
R 8 Report Engine Data ⊟
R 13 Store Analysis Results ⊟
R 16 Store Corrective Action Results ⊟
R 7 Store Engine Data ⊟
System ⊞
Logistics ⊟
Vehicle ⊟
Logistics ⊞
Vehicle ⊞
Logistics ⊞
Logistics ⊞
Logistics ⊞
Vehicle ⊞
Logistics ⊞
Logistics ⊞
Vehicle ⊞
Analyze Engine Status Data
Determine Corrective Action
Receive Engine Status Data
Store Corrective Action
Store Corrective Action Results
Collect Engine Status Data
Monitor Engine
Report Engine Status Data
Store Engine Status Data

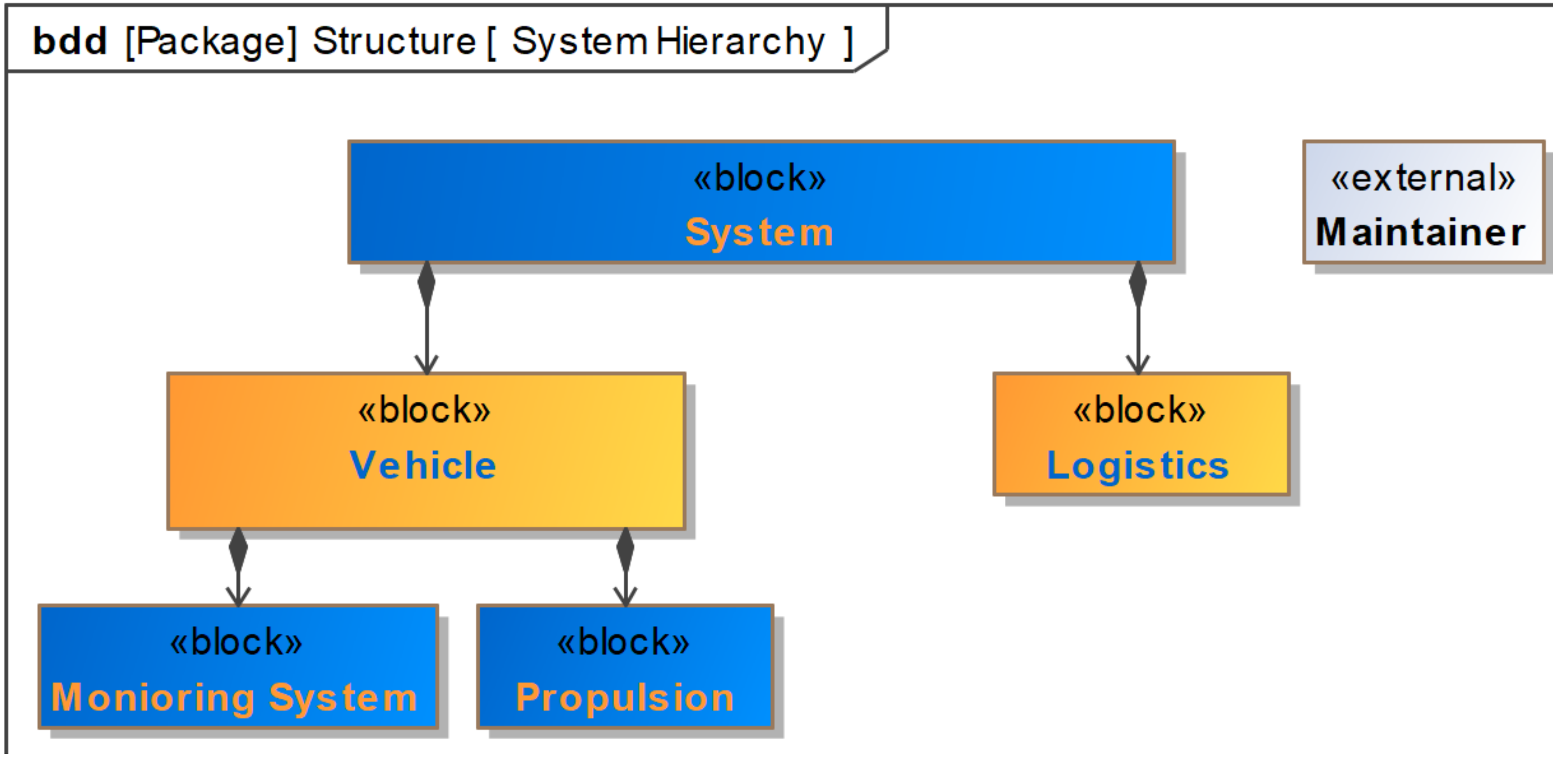
Functional Decomposition

- A Decomposition diagram can help you redundancies and gaps
- You can see from this one, that there is still a lot of work to do



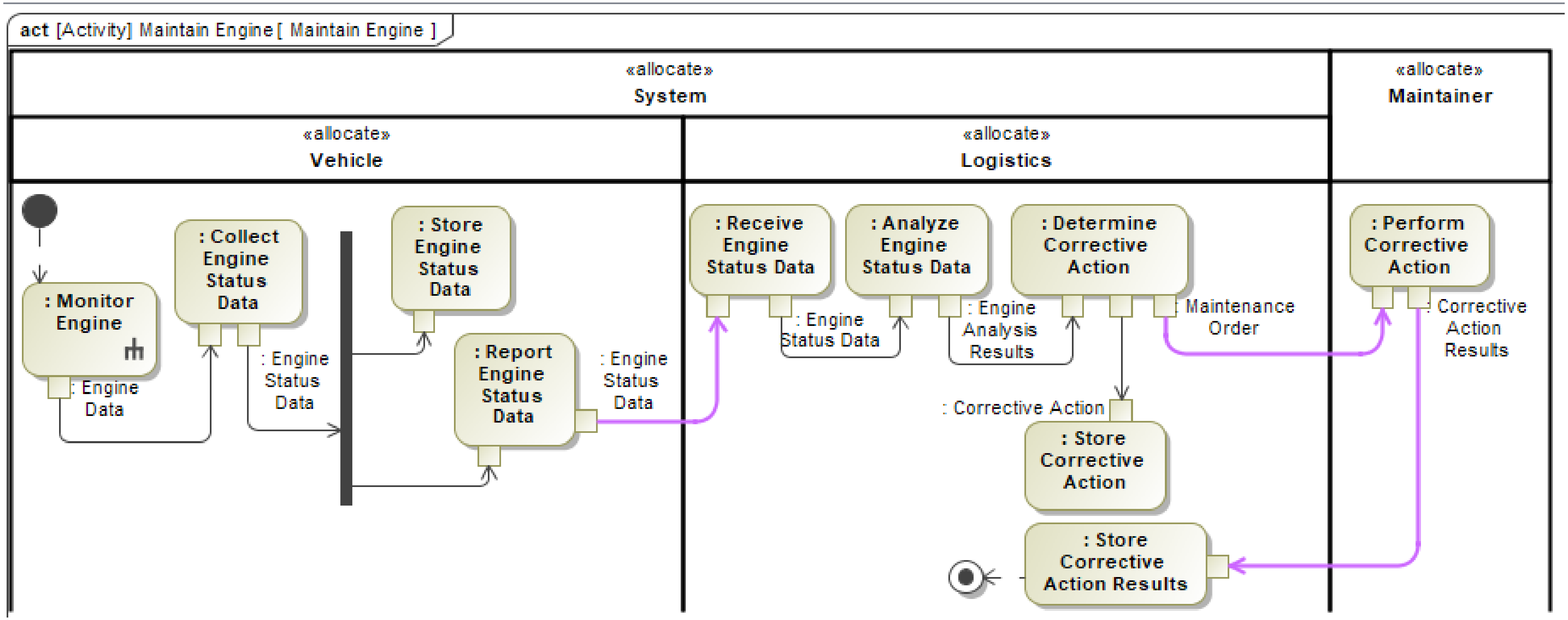
Engine Example

- **Does this help with the product structure?**
- **We know that propulsion and monitoring systems are under vehicle**
- **We have identified interfaces between Vehicle and Logistics, Logistics and maintainer**
- **What else did we identify?**



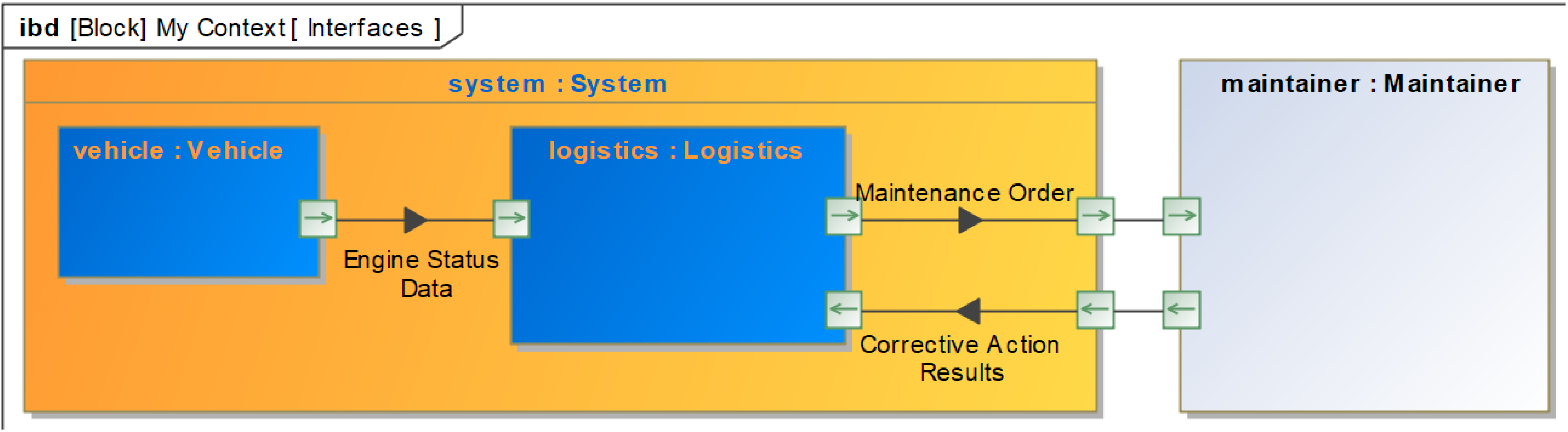
Interfaces

- **Wait, how did we identify the interfaces?**


act [Activity] Maintain Engine [Maintain Engine]
«allocate»
System
«allocate»
Maintainer
«allocate»
Vehicle
«allocate»
Logistics
: Monitor Engine
: Engine Data
: Collect Engine Status Data
: Engine Status Data
: Store Engine Status Data
: Report Engine Status Data
: Engine Status Data
: Receive Engine Status Data
: Engine Status Data
: Analyze Engine Status Data
: Engine Analysis Results
: Determine Corrective Action
: Maintenance Order
: Corrective Action
: Store Corrective Action
: Store Corrective Action Results
: Perform Corrective Action
: Corrective Action Results

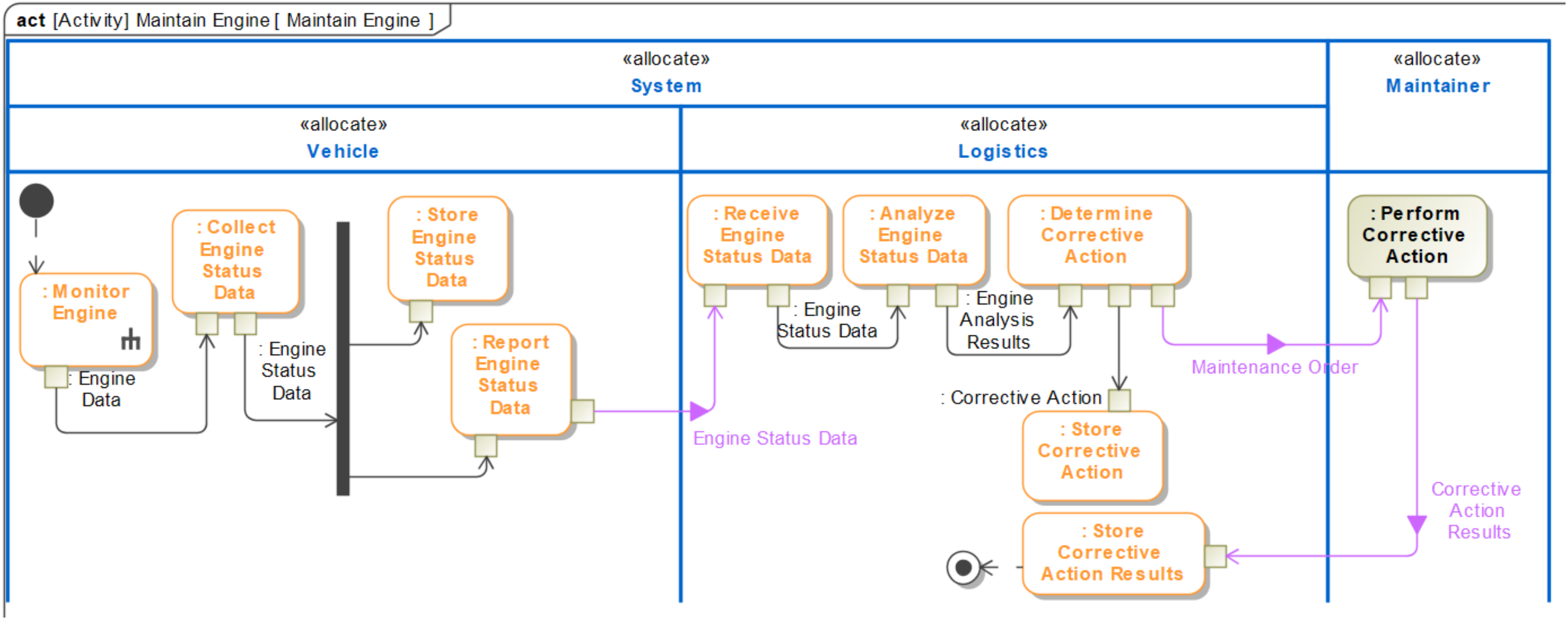
Different Interface views

- **More interfaces and data will be added as we progress through system design, but this is what we have so far**



How are we doing so far?

- At the beginning, I stated Systems have:
  - Structure: Parts (or components) that are directly or indirectly related to each other
  - Behavior: Processes that fulfill its function or purpose
  - Relationships: Parts and processes are connected by structural and/or behavioral relationships


act [Activity] Maintain Engine [Maintain Engine]
«allocate»
System
«allocate»
Maintainer
«allocate»
Vehicle
«allocate»
Logistics
: Monitor Engine
: Engine Data
: Collect Engine Status Data
: Engine Status Data
: Store Engine Status Data
: Report Engine Status Data
Engine Status Data
: Receive Engine Status Data
: Engine Status Data
: Analyze Engine Status Data
: Engine Analysis Results
: Determine Corrective Action
Maintenance Order
: Corrective Action
: Store Corrective Action
: Perform Corrective Action
Corrective Action Results
: Store Corrective Action Results

Summary

- One customer requirement written at the wrong level led to
  - One system requirement
  - Several statement of work items
  - Many subsystem requirements and functions
  - Hints about the system structure
  - Identification of interfaces
- A top down, iterative systems engineering approach is still needed even when detailed requirements are provided.