Syscience

System Engineering Science

# Concept of Operation

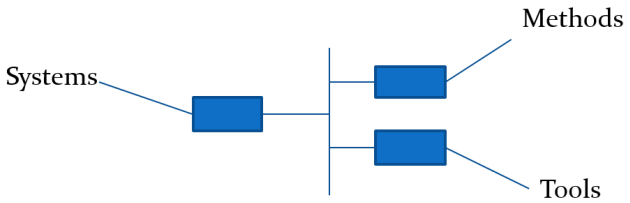
Needs capture and operational analysis

**System: G05**

Pascal Krapf, SÃ©bastien Berthier

**Syscience**

**18/09/2025**




**Reference:** Syscience R001, V2

Syscience

System Engineering Science

# Table of contents

# 1 Introduction

## 1.1 Object of the document

The purpose of this document is to establish the operational view of the system. It covers use-case identification, operational scenarios, needs capture and requirements definition. The different processes can be repeated cyclically. Several methods to capture needs and expectations are proposed.

Syscience has developed the Syscience Workshop software. It is a system engineering workshop consisting in a SaaS software associated to a database. It has been specially designed to support engineers for the deployment of systems engineering.

Examples illustrating the needs capture methods and the corresponding outcomes of the syscience workshop are presented in the following paragraphs.

## 1.2 References


- IEEE1220 (ISO1220): Standard for Application and Management of the Systems Engineering Process
- IEEE15288 (ISO15288): Systems Engineering - System Life Cycle Processes
- IEEE1471 (ISO1471): Recommended Practice for Architectural Description of Software-Intensive Systems
- EIA 632: Processes for engineering a system
- NASA SEH: NASA Systems Engineering Handbook
- Sys2016: P. Krapf, D. Loise, 2016, MÃ©thode dâ?identification des risques basÃ©e sur les modÃ¨les, 20e congrÃ¨s LambdaMu de maitrise des risques et de sÃ»retÃ© de fonctionnement, Saint-Malo, Octobre 2016.
- Sys2018: P. Krapf, S. Rakotosolofo, S. Berthier, 2018, Use of a system engineering workshop to identify the risks of a connected vehicle, 21e congrÃ¨s LambdaMu de maitrise des risques et de sÃ»retÃ© de fonctionnement, Reims, Octobre 2018.
- Sys2020: S. Berthier, P. Krapf, 2020, Understanding the risks caused by global warming using the System Engineering tool "Lâ?Atelier Syscience", 22e congrÃ¨s LambdaMu de maitrise des risques et de sÃ»retÃ© de fonctionnement, France, Octobre 2020.

## 1.3 Terminology

### 1.3.1 Terms

- Diagram: Graphical representation of a view of a system.
- Durability: capacity of the system to keed its desired properties during time.
- The functional needs: what is awaited from the SOI, for which users, and how it should be used.
- Non-functional needs: technical constraints that the SOI must respect (security criteria, number of users, computing power, etc.).
- Lifecycle: Succession of phases characterizing the system evolution, from the elaboration of its concept until its end of life.
- Lifecycle phase: A phase of the lifecycle of a system.
- Regulation: laws, rules or standards, defined by authorities, whose application is mandatory.
- Requirement: Formalized description of some characteristics of a system.
- Scenario: description of what happens to a system in a defined timespan.
- Sequence diagram: diagram representing actors and the succession of actions, events, messages and state changes. Sequence diagrams are used to represent scenarios.
- Stakeholder: Tangible or intangible entity, including persons, organizations, and company departments, likely to express needs, expectations or constraints about the system of interest [IEEE1220] 6.1.1, 6.1.2, 6.1.3.

### 1.3.2 Acronyms definitions

- COTS: Commercial Off The Shelf
- HMI: Human Machine Interface
- MBSE: Model Based System Engineering
- ppm: part per million
- RBSE: Requirement Based System Engineering
- SaaS: Software as a Service
- SOI: System Of Interest
- SOP: Start of Production
- TGA: Tooling Go Ahead

## 1.4 Document overview

This document describes an external view of the system of interest as a whole, without details about its internal design, using graphical model views. It defines the requirements that the system of interest shall satisfy.

# 2 System overview

## 2.1 Missions

The missions of the system of interest consists in providing a service or a product to main users. The missions are characterized in detail by diagrams and requirements which are described in the following paragraphs.


Figure 1:

**CallDiag: Figure(s) not found**

## 2.2 Key measures of effectiveness

Key measures of effectiveness reflect the overall satisfaction level of stakeholder expectations [IEEE1220] Â§6.1.5. They are identified in the following list.
The project identifies the technical performance measures (TPMs), which are key indicators of system performance. Selection of TPMs are usually limited to critical characteristics that, if not met, put the project at cost, schedule, or performance risk. Specific

TPM activities are integrated into the project report to periodically determine achievement to date and to measure progress against a planned value profile [IEEE1220] Â§6.1.13.

- Percentage of lifecycle phases without identified stakeholder expectation
- Number of expectations without link to system requirements
- Number of system requirements without link to stakeholder expectation

## 2.3 Lifecycle

User's utilization phase is an important phase of the project, but other phases should not be underestimated. A system that is too difficult to produce or too expensive is a waste of time and money. A system that cannot be maintained will not satisfy users for a long period of time. It is thus worth to define the whole lifecycle and to go through all phases to identify stakeholders. The system lifecycle is adapted from the standard [IEEE15288]. Whether they are individuals or organisations (enterprise departments, associations, â), stakeholders express needs and expectations for one or even several lifecycle phases. The following diagram gives a synthetic view of the lifecycle phase and corresponding stakeholders. Hyperlink give connections to diagrams listing stakeholder needs.


Figure 2:

**CallDiag: Figure(s) not found**

# 3 Stakeholder needs

## 3.1 Methods

### 3.1.1 Document analysis

Some needs and expectations concerning the SOI are written in specifications or in standards. These documents have to be analyzed in order to identify these needs and to define how the SOI will answer them. This activity can be broken down into the following steps:

- Identify the stakeholders that are likely to write such document
- Collect the documents
- Identify the expectations concerning the SOI
- Review system requirements with stakeholders in order to validate them.

For each stakeholder, block diagram are used to represent needs and expectations. The SOI is represented in the center. External elements are represented around the SOI. Needs and expectations appear in the corresponding compartment. The way the system answers needs and expectations is written on the links between the SOI and the stakeholder.

### 3.1.2 Stakeholder interviews

Sometimes needs and expectations are not formalized, and thus, it is necessary to capture them by interviewing stakeholders. This activity can be broken down into the following steps:

- Identify the stakeholders that have unformal expectations
- Organize and carry out interviews with each stakeholder
- Identify the expectations concerning the SOI
- Define what the SOI shall do as a black box to satisfy the expectations (write system requirements)
- Review system requirements with stakeholders in order to validate them.

For each stakeholder, block diagram can be used to represent needs and expectations. The SOI is represented in the center. External elements are represented around the SOI. Needs and expectations appear in the corresponding compartment. The way the system answers needs and expectations is written on the links between the SOI and the stakeholder.

## 3.2 Captured needs

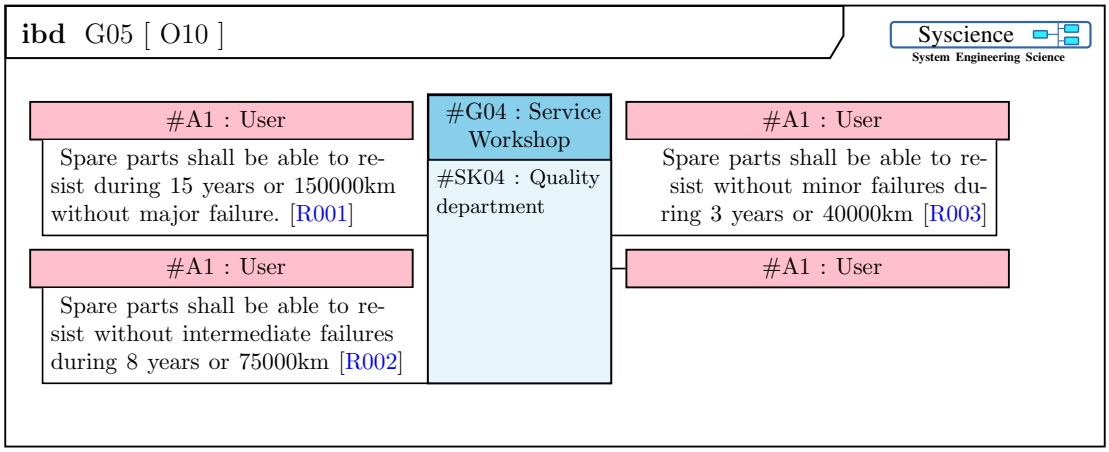



Figure 3: **MultiBlock_O10**

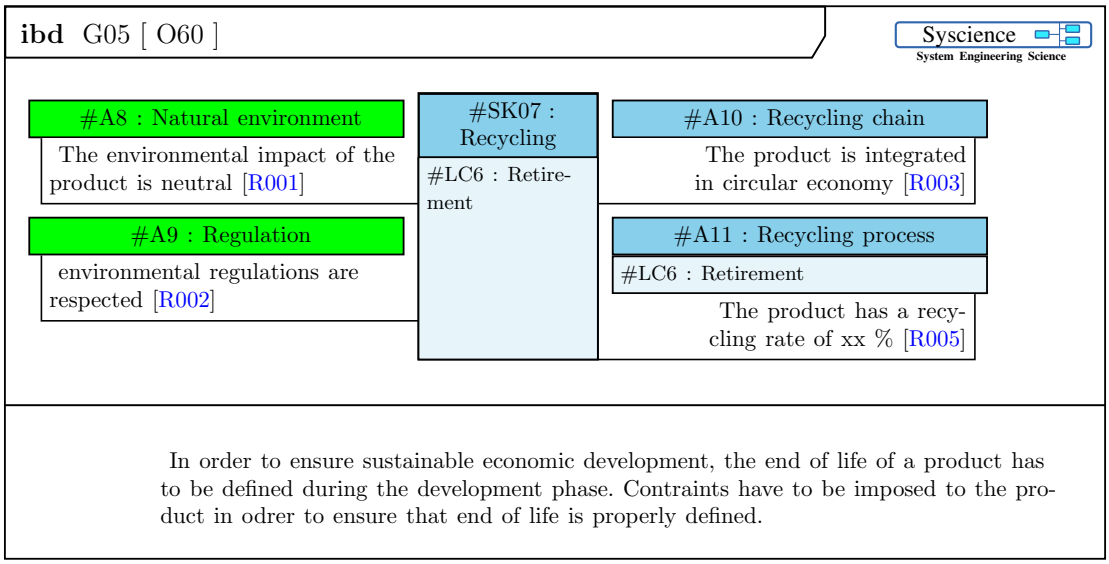



Figure 4: **MultiBlock_O60**

# 4 Operational analysis

## 4.1 Use case analysis

Use cases summarize the way people and external elements interact with the SOI [IEEE1220] Â§6.1.6. Expectations about each use case have to be captured. A good way to do this is to consider each lifecycle phase, and to detail use cases in each phase. A standard lifecycle is defined in ISO standard [IEEE15288]. The approach includes the following steps:

- Formalise the lifecycle of the SOI.
- For each lifecycle phase, identify the list of external elements which interact with the SOI.
- Scan the elements interacting with the system and for each of them name the use case in which this interaction occurs.
- For each use case, describe one ore more corresponding scenarios.

Use cases and their dependancies can be represented on use case diagrams.

### 4.1.1 Use case identification

In the beginning of an innovative project, it is quite frequent that detailed requirements from product marketting are not yet formalized. Thus needs capture can be carried out either by the analysis of scenarios detailing the use case. Use cases are inferred from system mission and scenarios can be used to detail use cases. Reciprocally scenarios can be attached to use cases.
A use case diagram represents utilization targets for the system. The following diagram is generated automatically from the system operational scenarios and contains hypertext links to the corresponding sequence diagrams.

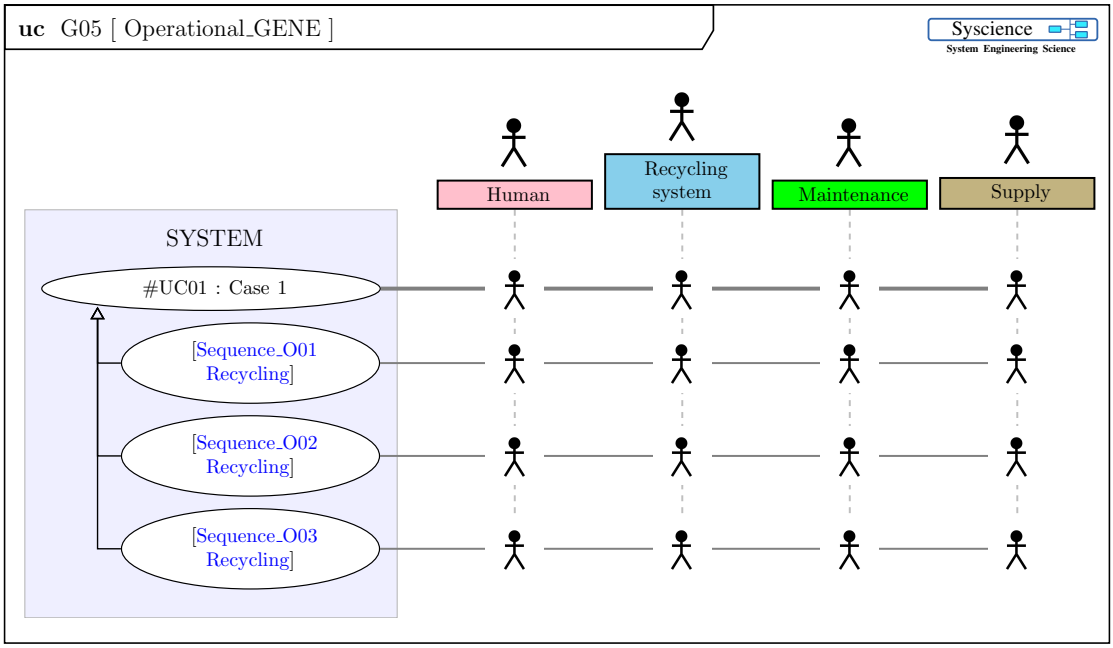



Figure 5: **UseCase_Operational_GENE**

## 4.2 Scenario analysis

At the beginning of a project, it may happen that the concept of the system is under construction. Needs and expectations concerning new features are neither identified nor formalized. Engineers have to begin with some vague concepts and ideas. However, they have to ensure traceability, and thus they need a structured approach. A good way to do this is to imagine the ways users will handle the system and capture these scenarios into sequence diagrams. Scenarios are then used to define requirements. This approach includes the following steps:

- Define the system mission.
- List the actors interacting with the SOI.
- Describe scenarios in which the system is used.
- Identify requirement concerning the SOI.

### 4.2.1 Product marketting needs

An operational scenario is a description of the succession of activities done by the system and external actors exchanging messages between each other. They are described by sequence diagrams. Activities are represented by rectangular boxes centered on the lifeline associated to each actor. Arrows represent messages exchanged between actors. In each

scenario, activities are described in text boxes on the left of the diagram. Hypertext links give access to automatically generated requirements.

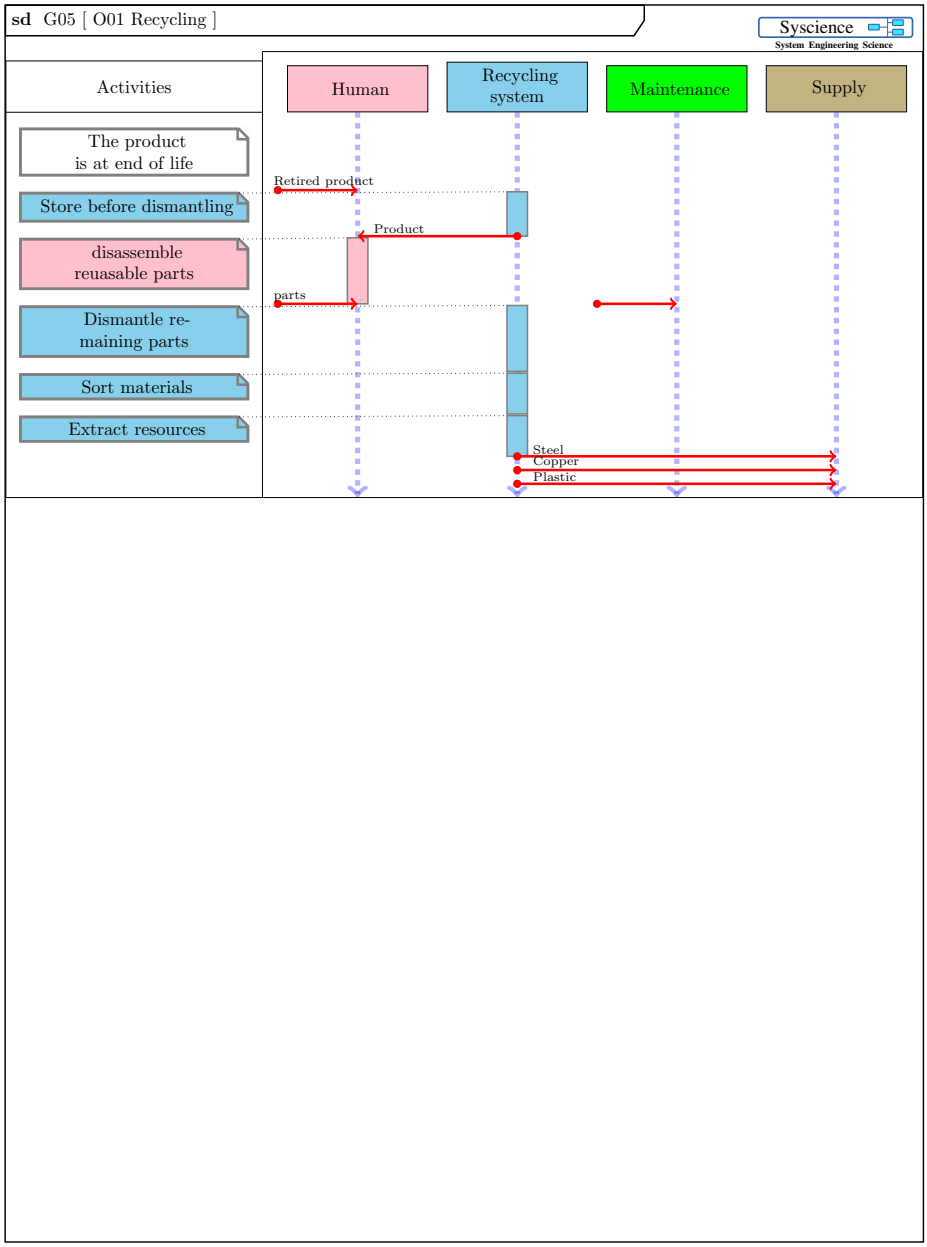



Figure 6: **Sequence_O01 Recycling**

**This figure was cited in UseCase_Operational_GENE.**

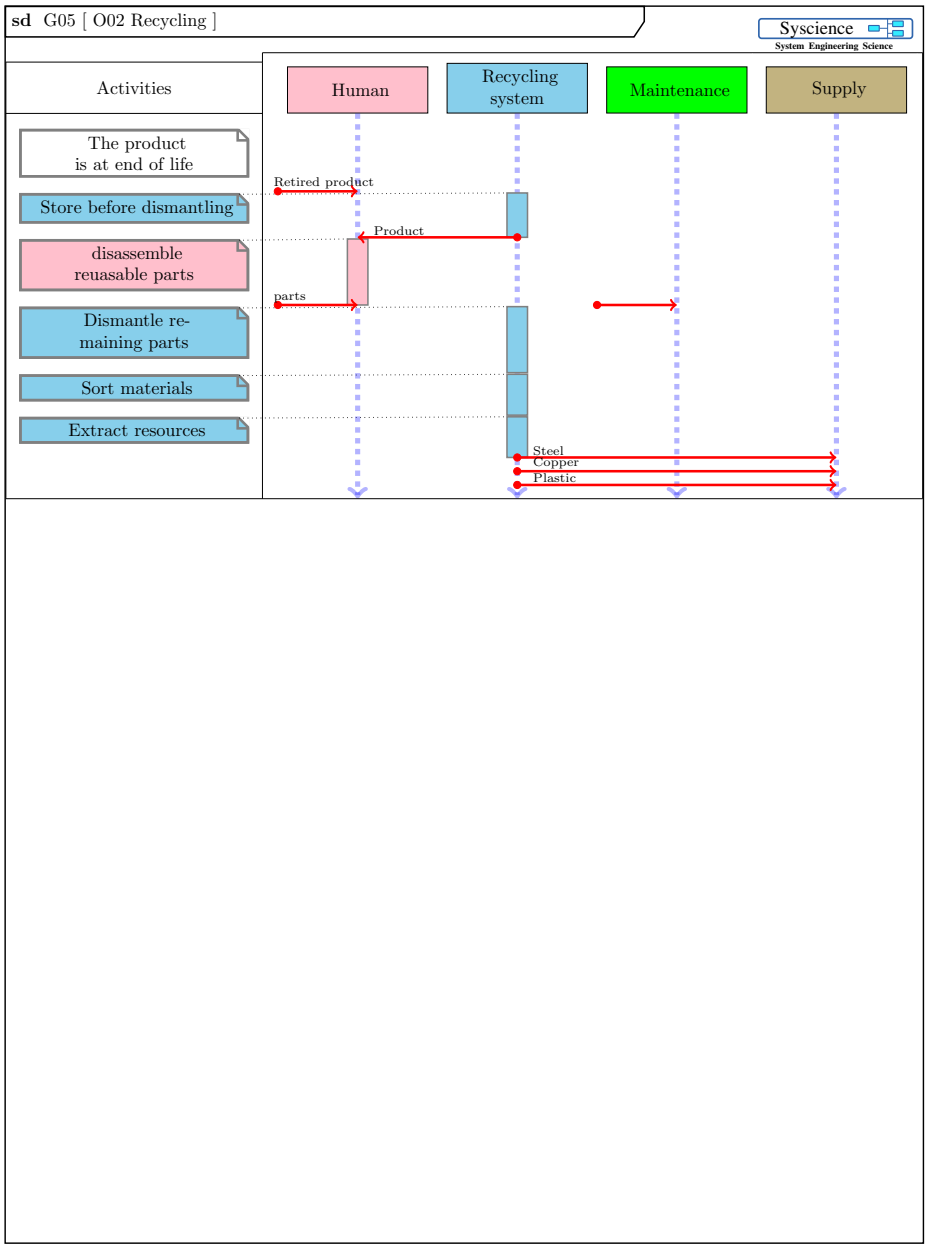



Figure 7: **Sequence_O02 Recycling**

**This figure was cited in UseCase_Operational_GENE.**

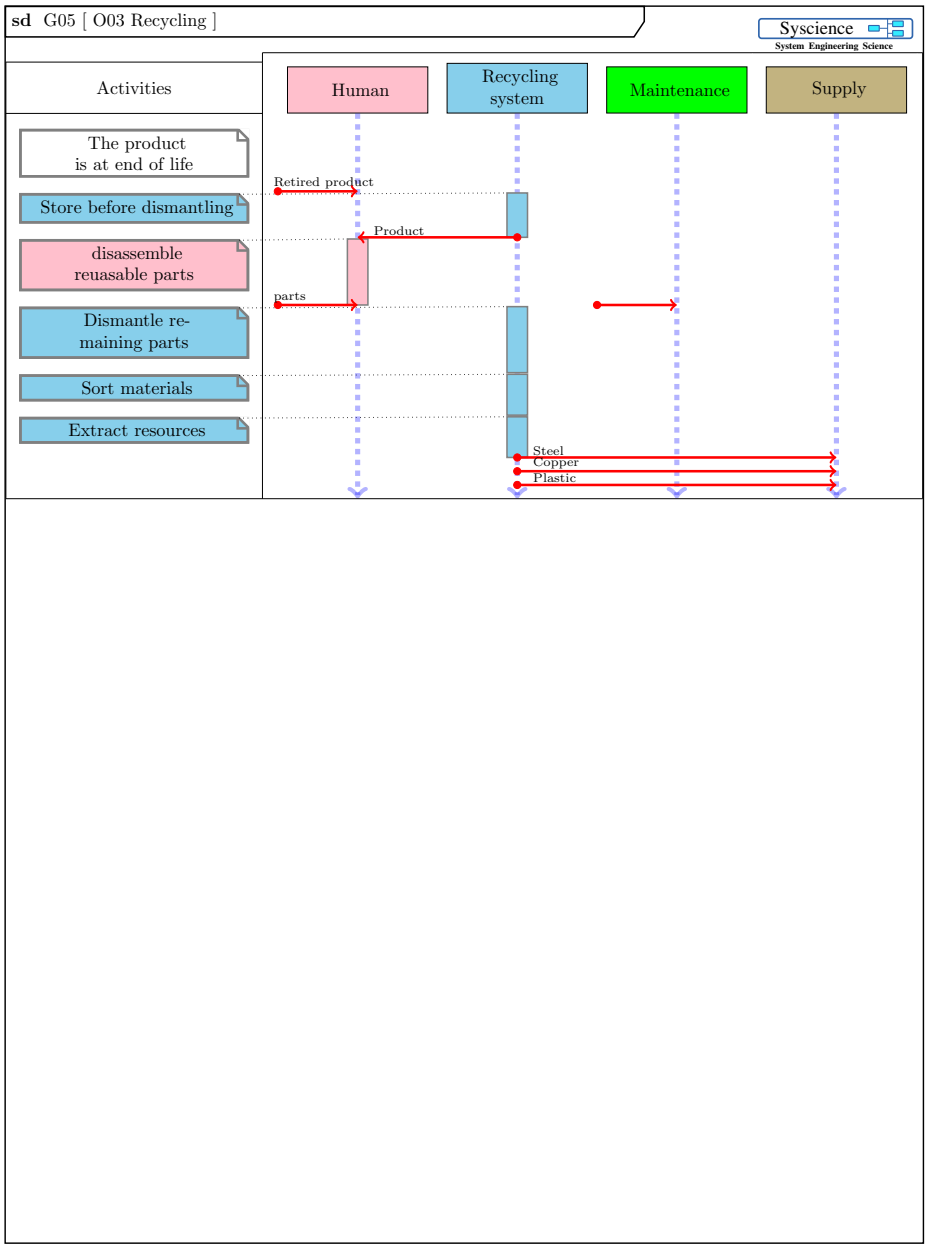



Figure 8: **Sequence_O03 Recycling**

**This figure was cited in UseCase_Operational_GENE.**

# 5 Model of the environment

External elements interact with the system of interest and exchange data, energy or matter with it. External elements and corresponding flows are modeled in the following diagram.


Figure 9:

**CallDiag: Figure(s) not found**

# 6 Requirements synthesis

## 6.1 Requirement formulation

High level system requirements result from the capture of needs. Many authors insist on the statement of system requirements in the form "The system shall do something". A few people say that such requirements shall be described from the user point of view and shall be stated "The user shall. . . " or "The user expects. . . ". However, a system engineer has to describe what his system shall do to answer userâ?s needs and expectations. He will not focus on userâ?s expectations that are about other systems. Thus, we think that the system under investigation shall be mentioned in the requirement. Stated "The user shall", requirements may be uderstood as limitations imposed to the user. Therefore, we decided to write these requirements as statements concerning the system of interest.

Visual diagrams have been used to capture and represent needs about the system of interest. This approach is referred to as model based system engineering. Requirements based system engineering refers to an approach in which requirements about the system

of interest are managed as textual requirements. These both aprroaches complement each other: while MBSE is useful to check completeness of needs capture, RBSE allows to state clearly the engagement of the system owner. System requirements define unambiguously what has to be tested, while visual diagrams do not always distinguish the system engagement and informative description of the environment.
This chapter collects requirements build upon the needs capture. Each requirement shall be:

- Specific: the requirement is a usefull description of a system feature. Something would be missing if the requirement is not satisfied.
- Measurable: a measurement action (a test or a process check) can be defined to decide wheather a given system satisfies the requirement or not.
- Attainable: the defined target shall not be unreachable. The target is defined to be attained.
- Realistic: requirements are coherent with the state of the art.
- Traceable: it is possible to identify why this requirement has been defined, and which needs it satisfies.

## 6.2 Operational requirements

Operational requirements describe how the system is operated. They detail the mission of the system as well as specific use cases. Corresponding system requirements are listed in this paragraph.

### 6.2.1 Destination selection

### 6.2.2 Autonomous driving

## 6.3 Constraints requirements

System requirements corresponding to constraints are listed in the following paragraphs.

### 6.3.1 HMI requirements

HMI requirements define the characteristics of the system HMI. Corresponding system requirements are listed in this paragraph.

### 6.3.2 Durability

Durability requirement define the system mission profile and the ability of the system to maintain its characteristics during the lifecycle depending on mission profile. Corresponding system requirements are listed in this paragraph.

### 6.3.3 Regulation requirements

Regulation requirements describe system engagement concerning the compliance with regulations and standards. Corresponding system requirements are listed in this paragraph.

### 6.3.4 Assembling constraints

Assembling contraints requirements describe the system engagement concerning assembling needs and expectations. Corresponding system requirements are listed in this paragraph.

### 6.3.5 Maintenance constraints

Maintenance contraints requirements describe the system engagement concerning maintenance and repairing needs and expectations. Corresponding system requirements are listed in this paragraph.

### 6.3.6 Recycling constraints

Recycling contraints requirements describe the system engagement concerning the end of life of the system, and needs and expectations concerning recycling and reuse. Corresponding system requirements are listed in this paragraph.