Process Design and Resource Allocation at the ProRail ERTMS Integration Lab

M.I. Zwanenburg

April 2021

ProRail

UNIVERSITY OF TWENTE.

Process Design and Resource Allocation at the ProRail ERTMS Integration Lab

April 2021

Author
M.I. (Margriet) Zwanenburg
s1685279

Educational Institute
University of Twente
Faculty of Behavioural, Management and Social Sciences
Section of Industrial Engineering and Business Information Systems

Educational Program
MSc. Industrial Engineering and Management
Specialisation: Production and Logistics Management
Orientation: Supply Chain and Transportation Management

Graduation Company
ProRail B.V.
Moreelsepark 3
3511 EP Utrecht
The Netherlands

Examination Committee
Dr. P.C. Schuur
University of Twente

Dr. Ir. A.G. Leeftink
University of Twente

Ing. H.M. van Houten
ProRail B.V.

Preface

This thesis finalises my master Industrial Engineering and Management at the University of Twente, and my time as a student. It has become a whole 'life's work' where I had some support and help during the process.

At first, I would like to thank all involved people in this research from ProRail. I have spoken to various employees, and they were all very open and willing to help. Although I never saw the office in Utrecht from the inside due to the pandemic, I felt very welcome. I would like to thank my daily supervisors Monique Voorderhake-Borst and Henri van Houten in particular. The meetings we had were always very helpful, whether it was focused on the research or just a nice chat!

Secondly, I would like to thank my supervisors from the University of Twente, Peter Schuur and Gréanne Leeftink. Peter was involved in the research from the very beginning. We were able to go to the lab in Amersfoort once together, which was nice. I enjoyed our online meetings filled with your enthusiasm, which really helped me during the research. Gréanne joined us later on. I would like to thank you for making time and providing useful feedback.

Finally, I would like to thank my friends and family for their support and feedback during the research. They showed great interests in me, the research and the progress, which kept me motivated.

I hope you enjoy reading this thesis!

Margriet Zwanenburg
Enschede, April 2021

Management Summary

A new automatic train protection system is going to be implemented in the Netherlands: the European Rail Traffic Management System (ERTMS). ProRail B.V. established the ProRail ERTMS Integration Lab (PREI) to test the integration of this new system extensively, to prevent breakdowns on the already busy Dutch rail network.

Problem Description

PREI was established in 2018. This lab is already operational, but still in a development phase. The processes are not fully designed yet, which results in a lack of clarity. Furthermore, the activities of PREI that require resources of the lab, such as testing, demonstrations and trainings, are not optimally scheduled. The scheduling is done manually and there is no method yet to support this process. At this point in time, this is still going well, as the lab capacity is not fully utilised yet. However, the workload is expected to increase in the future, which makes it harder to schedule the activities with their requirements, especially when it is done manually.

Based on these problems, we define the following research question:

> ***How can the processes of the ERTMS Integration Lab be designed based on a suitable typology, and how can the resources be allocated to the activities under various growth scenarios?***

The research consists of two parts: *process design* and *scheduling*. Both parts ask for a different approach and have different results.

Process Design

The processes of PREI are categorised into three types: *management processes*, *core processes* and *supporting processes*. We gathered the required information during meetings with various stakeholders. We selected suitable methods to design the processes based on existing literature, the requirements of ProRail, and already by PREI designed processes.

Different business process modelling methods are elaborated, and we selected the flowchart method with the addition of swimlanes to design the three *core processes*, which resulted in eight flowcharts. A snapshot of the demonstration flowchart can be found in Figure 1. The involved roles in the processes are determined and added as swimlanes. The ETCS System Compatibility (ESC) testing process, as defined on European level, is complex and involves many different stakeholders. Therefore, a table with additional remarks is added to the flowcharts of this process for better understanding.

A list of fifteen *management* and six *supporting processes* is proposed. These processes are captured in less detail, as the core processes are of greater importance at this point in time. A form is proposed based on the forms-based approach. This form can help with the design of the processes.

The (designs of the) processes are validated with the stakeholders.

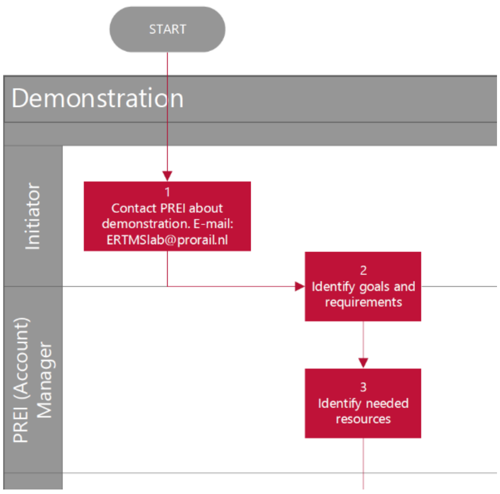

Figure 1: Snapshot demonstration flowchart

Scheduling

The *Resource-Constrained Project Scheduling Problem (RCPSP)* addresses how limited resources should be assigned to activities, optimising a defined objective. Many well-known optimisation problems are special cases of the RCPSP and have overlap with the scheduling problem encountered at PREI. Therefore, we model the PREI scheduling problem using an RCPSP based model.

The proposed model is an integer linear programming (ILP) model. We defined two objectives based on literature and meetings with ProRail: *minimising the maximum lateness (ML)* and *minimising the total tardiness (TT)*. Various constraints are taken into account that define the use of the workplaces, lines and resources, and the release dates of the activities.

We did seven experiments with different data sets. The first experiment is with a toy-sized data set, and a time horizon of 8 working weeks. The second experiment is done with a data set composed by ProRail. We expanded the provided data set in five ways in the following experiments, based on growth scenarios of the lab. The time horizon of experiments 2-7 is set to half a year (1 January 2021 – 30 June 2021). The time periods are defined as half working days. We used the Gurobi solver in Python 3, and a computer with an Intel Core i7-8550 CPU and 8GB RAM. We were able to validate the model and we found optimal solutions.

Two graphs are generated for every experiment. Figure 2 includes the operational schedule and Figure 3 the resource allocation graph of experiment 2 with the objective total tardiness. We see that the operational schedule is still quite empty, in line with the current experiences of relatively low resource utilisation. The Post21 resource has the highest utilisation.

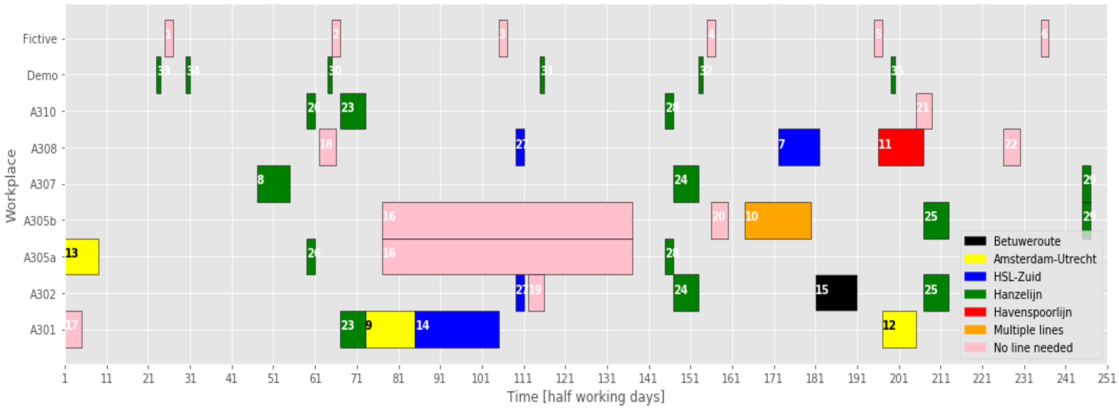

Figure 2: Operational schedule with data set ProRail and the objective to minimise the total tardiness

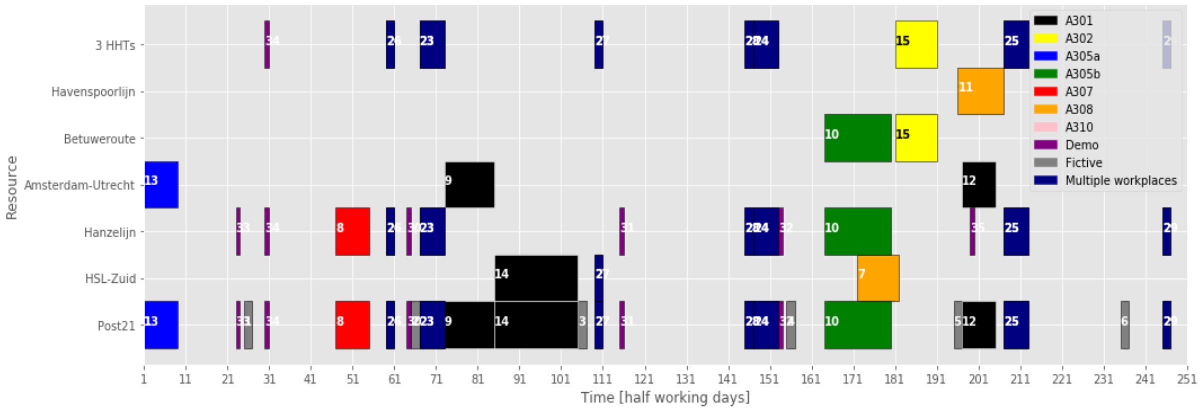

Figure 3: Resource allocation with data set ProRail and the objective to minimise the total tardiness

Conclusion

We designed three core processes with eight flowcharts including additional remarks, which will be implemented at ProRail and included in the quality management system. Also, we proposed fifteen management and six supporting processes, which we recommend to further specify in the implementation phase.

The proposed model is able to generate optimal schedules for the PREI scheduling problem, under various growth scenarios. In every experiment, an optimal solution was found within 1½ hours. The maximum lateness and total tardiness of all experiments with both objectives can be found in Table 1. We do not know whether the manually composed schedules are optimal, so there is no validated benchmark. We can say that the use of the model eases the scheduling process and generates at least as good schedules, if not better schedules. The planner of PREI supports the method and the results.

Table 1: Maximum lateness (ML) and total tardiness (TT) of all experiments, expressed in half working days, with both objectives. The objective value is bold, as both penalties were zero in every case

Experiment	1	2	3	4	5	6	7
Data set / **Objective**	Toy-sized	Provided	Tests 2x	Tests and demos 2x	Tests and trainings 2x	Tests, demos and trainings 2x	ESC tests 4x, other tests 2x
Maximum lateness	**ML: 0** TT: 0	**ML: 3** TT: 3	**ML: 3** TT: 3	**ML: 3** TT: 14	**ML: 7** TT: 76	**ML: 7** TT: 125	**ML: 3** TT: 3
Total tardiness	ML: 0 **TT: 0**	ML: 3 **TT: 3**	ML: 3 **TT: 3**	ML: 3 **TT: 6**	ML: 8 **TT: 30**	ML: 9 **TT: 39**	ML: 3 **TT: 3**

The most suitable objective for the PREI scheduling problem is to minimise the total tardiness. We observed that the maximum lateness is in all experiments and with both objectives acceptable, but the total tardiness becomes too large when the utilisation of the resources increases and the objective is to minimise the maximum lateness.

Recommendations

We recommend to implement the defined processes. Some designed processes are not yet fully performed in real. When it becomes clear that a process is not completely sound, the design should be changed. Also, when a process changes, the design should be changed accordingly. The roadmap of the implementation of the processes can be found in Figure 4.

Figure 4: Roadmap implementation of the processes

The model is suitable for the PREI scheduling problem and implemented in the Python language, but ProRail is not able to use this language (yet). Therefore, we recommend to decide on suitable software to implement the scheduling model, and requirements for running the model. Suitable software can either be existing software or it can be appropriate to develop new software (inhouse).

Although the model is suitable, the model can still be improved in order to fit better to reality. Therefore, we recommend to expand the model and add slack to the activities to deal with the uncertainty in the processing times, and add priorities to differentiate the importance of the activities. Slack can only be added when more historical data is available to determine the mean and standard deviation of the activity types, and priorities when a priority policy is established.

Once a schedule is composed, it is not possible to add extra activities to the schedule without rerunning the model and thus start over and compose a new schedule. This is not desired, and can be solved by creating a (more) dynamic model. To do so, we recommend to do further research in the directions of shortening the time horizon, a rolling horizon approach, and adding weights to already scheduled activities. Then, it will be possible to schedule additional activities on short notice without having to reschedule the already scheduled activities.

Our model is suitable for a small list of activities. However, when the number of activities increases, it can occur that the proposed model is not able to generate optimal or even feasible solutions anymore, in reasonable time. In that case, we recommend to look into the possibilities of constructive and improvement heuristics.

Figure 5 includes a roadmap for the implementation of the scheduling model.

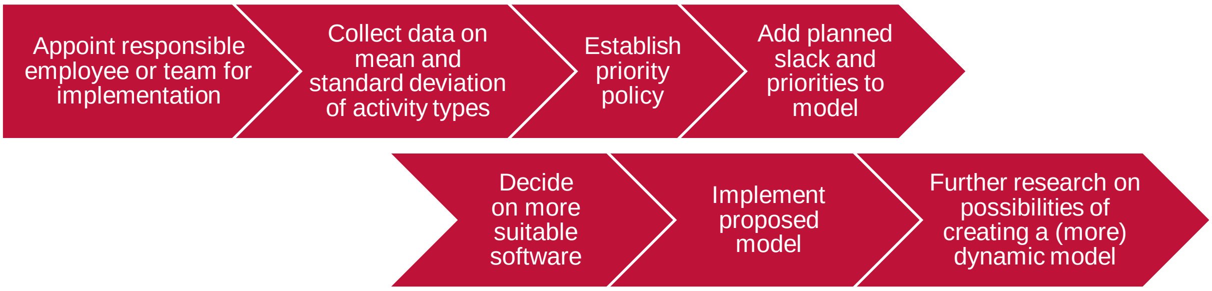

Figure 5: Roadmap implementation of the scheduling model

List of Abbreviations

Abbreviation	*Explanation*	*Introduced on page*
ATB	“Automatische Trein Beïnvloeding”	1
ATP	Automatic Train Protection	1
CSS	Centralised Safety System	15
DMI	Driver Machine Interface	11
EC	European Commission	9
ERA	European Union Agency for Railways	4
ERTMS	European Rail Traffic Management System	1
ESC	ETCS System Compatibility	14
ETCS	European Train Control System	14
GSM-R	Global System for Mobile Communications – Railway	10
HHT	Hand Held Terminal	15
HSL-Zuid	“Hogesnelheidslijn-Zuid”	13
ILP	Integer linear programming	66
IXL	Interlocking	11
KMS	“KwaliteitsManagementSysteem”	50
KPI	Key Performance Indicator	51
LWB	“LeiderWerkplekBeveiliging”	20
MRCPSP	Multi-Mode Resource-Constrained Project Scheduling Problem	66
NS	“Nederlandse Spoorwegen”	1
OBU	On-Board Unit	15
PREI	“ProRail ERTMS IntegratieLab”	1
RBC	Radio Block Centre	11
RCPSP	Resource-Constrained Project Scheduling Problem	65
TCL	Test Control & Logging	18
TEN-T	Trans-European Transport Network	9
TSI	Technical Specifications for Interoperability	9

List of Keywords

Keyword	*Explanation*	*Introduced on page*
Activity	Performing a test, demonstration or training, which requires a workplace in the lab.	2
Processes	A series of needed activities to achieve a goal. Input is used to create valuable output.	2
Scheduling	The allocation of certain resources to certain activities, during a given time period.	4
Test	Tests are performed in the lab to demonstrate the integration of ERTMS trackside equipment and ERTMS onboard equipment (in the train).	1
Test Campaign	Series of tests that need to be performed for one application.	15
Workplace	A workplace at the lab is a place with twelve screens, where the activities can be performed.	4

Glossary

English	*Dutch*	*Introduced on page*
Level crossing	Overweg	1
Line	Baanvak	13
Rail network	Spoorwegnet	1
Rail traffic controller	Treindienstleider	2
Vehicle authorisation	Materieeltoelating	7
Signal	Sein	1
Point	Wissel	1
Timetable	Dienstregeling	2
Track	Spoor (railinfra)	1
Track workers	Baanwerkers	13
Train driver	Machinist	1
Railway undertaking	Vervoerder	1

Contents

1 Introduction

The first train track in the Netherlands was opened in 1839. Only a few trains were in use in the beginning, but more and more trains and tracks were added to the railway network over the years. After a big accident (Harmelen, 1962), an Automatic Train Protection (ATP) system was implemented in the Netherlands: “Automatische Trein Beïnvloeding” (ATB) (Pieters, 2019), which brakes automatically when the train driver does not react (on time) on signals. However, this system is getting outdated, and the tracks are getting busier with (international) trains. This calls for a new system: the European Rail Traffic Management System (ERTMS). The subsystems for ERTMS are integrated into both the infrastructure and trains (Programma ERTMS, 2020). The integration and interaction of the subsystems can be tested within a lab in a safe and controlled manner. But how can such a lab and the tests be organised the best? That is what this research will be about.

This research is executed as graduation project for the master Industrial Engineering and Management. In this chapter, the research is introduced. In Section 1.1, ProRail B.V. is introduced. The ProRail ERTMS Integration Lab (PREI) and the problems encountered in the lab are discussed in Section 1.2. The scope of the research is defined in Section 1.3. The research questions in Section 1.4 follow from the problem description and the scope of the research. At last, the deliverables are discussed in Section 1.5 and the structure of the report in Section 1.6.

1.1 Company Description

Due to changes in the European regulations, the “Nederlandse Spoorwegen (NS)” was reorganised in 1995. The infrastructure and exploitation of the rail network had to be split. NS took up the exploitation, and Railinfratrust B.V. was founded to construct, maintain and manage the tracks. Since 2013, the trade name ProRail is used by Railinfratrust B.V, and when the subsidiaries of Railinfratrust B.V. merged in 2015, ProRail B.V. was officially founded (ProRail, 2020b).

ProRail is responsible for the construction, maintenance, management and safety of the Dutch railway infrastructure. As an independent party, ProRail divides the space on the tracks, arranges all train traffic, builds and manages stations and builds new tracks. Also, existing tracks, points, signals and level crossings are maintained (ProRail, 2020e).

The mission of ProRail is to connect people, cities and companies by rail. Now and in the future. They want to make pleasant travelling and sustainable transport possible. Three goals follow from this mission (ProRail, 2020e):

1. Connect — Developing the capacity for the mobility of the future.
2. Improve — Make rail mobility as reliable as possible, now and in the future.
3. Sustainability — Make rail mobility as sustainable as possible.

The Dutch rail network is one of the busiest rail networks in Europe, with 7,097 km of tracks, 6,560 points, 11,586 signals, 2,477 level crossings and 400 stations in 2019 (ProRail, 2020d). ProRail is an independent organisation with over 4,000 employees (2019) and works closely with the government, railway undertakings and contractors. Also, several cooperations with international sector partners in various areas are formed over the years (ProRail, 2020e).

1.2 Problem Description

As already stated, the currently used system for automatic train protection in most of the Netherlands is ATB. This system is getting outdated and is not suitable for the increasing bustle on the tracks. Due to European regulations, further development of ATB is not allowed. It is decided to implement the new (international) system ERTMS in the Netherlands. The implementation already started and should be finished by 2050. ERTMS will be the new international standard for automatic train protection. The system is integrated into trains and infrastructure (Programma ERTMS, 2020).

In the remainder of this section, the problem description is further elaborated. First, ProRail's ERTMS Integration Lab is introduced. Next, the encountered problems are discussed, and the problem cluster is shown and described. The research goal follows from the problem cluster.

1.2.1 ProRail ERTMS Integration Lab (PREI)

ERTMS consists of several subsystems. The separate components of the system are tested during the development of those components, but the integration is not. It is important and even required by European regulations to test the integration of the subsystems before the system can be fully used. Those tests can be performed on the rail network, but the Dutch rail network is very busy. Complete testing programs on the regular tracks are therefore not desired. When a test fails, a train can strand for example. This has major consequences on the complete timetable. Such an occurrence is too time-consuming and expensive. Therefore, the ProRail ERTMS Integration Lab (PREI) is established. This lab gives the possibility to extensively test the whole chain, from rail traffic controller to train driver (ProRail, 2020f).

PREI is opened at the end of 2018 in the Railcenter in Amersfoort. At PREI, integration tests are guided, demonstrations provided, and trainings facilitated on all ERTMS infrastructure. These activities are the most important activities of the lab. Besides, the lab increases, shares and secures knowledge of system integration within the rail systems and on the interfaces of the rail systems. The lab contributes therefore to the further development and implementation of ERTMS in the Netherlands (ProRail, 2020f). The lab is already operational, but ProRail is still working on defining and expanding the roles and tasks of PREI.

1.2.2 Encountered Problems

The lab is still in a development phase. We can distinguish different problems at PREI. The first encountered problem is that the process design is not fully elaborated and captured. The processes contain various aspects, such as everything that is needed to ensure the lab is available for possible tests, demonstrations and trainings, and everything that is needed to organise, run and complete tests properly. Because the processes are not fully designed yet, there is a lack of clarity about the lab: who is allowed to use the test facilities, what can be tested, and what should be delivered before a test can start? Also, the test plan as currently used is not sound. When the process design is elaborated, a test plan can be made based on the design. This will result in more clarity about which parts (such as documents, hardware and software) should be present before a test can start, and who is responsible for what.

At this point, the testing, demonstrating and training activities are not fully scheduled. This, together with the lack of a test plan, results in a not optimised execution of the activities. 'Just doing' is still going well because only three lines can be tested, and the lab is not busy yet. However, more lines will be equipped with ERTMS and added to the lab, and there are new updates of ERTMS (onboard equipment) coming, which will increase the workload. The way of working should be well organised, so everyone knows what is expected of them. An organised way of resource allocation can also help to create more clarity.

Because the processes are not fully elaborated and captured yet, it is unknown how much capacity is needed for a single test. Another problem is that the demand of the lab is uncertain. This, together with the upcoming updates and the implementation on more lines, results in unclarity in how to determine sufficient capacity of the lab, especially in the future.

1.2.2.1 Problem Cluster

The encountered problems as described before and the relations between the problems are visualised in the problem cluster in Figure 1.1. In this figure, the problems are depicted with rectangles and the cause-consequence relationships are indicated with arrows. Five possible core problems are determined (rectangles with bold borders). A possible core problem is a problem that does not have any causes (Heerkens & Van Winden, 2012). Also, five action problems are determined (rectangles with grey backgrounds). An action problem is a problem that is perceived by the problem-owner and indicates a difference between the norm and the reality (Heerkens & Van Winden, 2012).

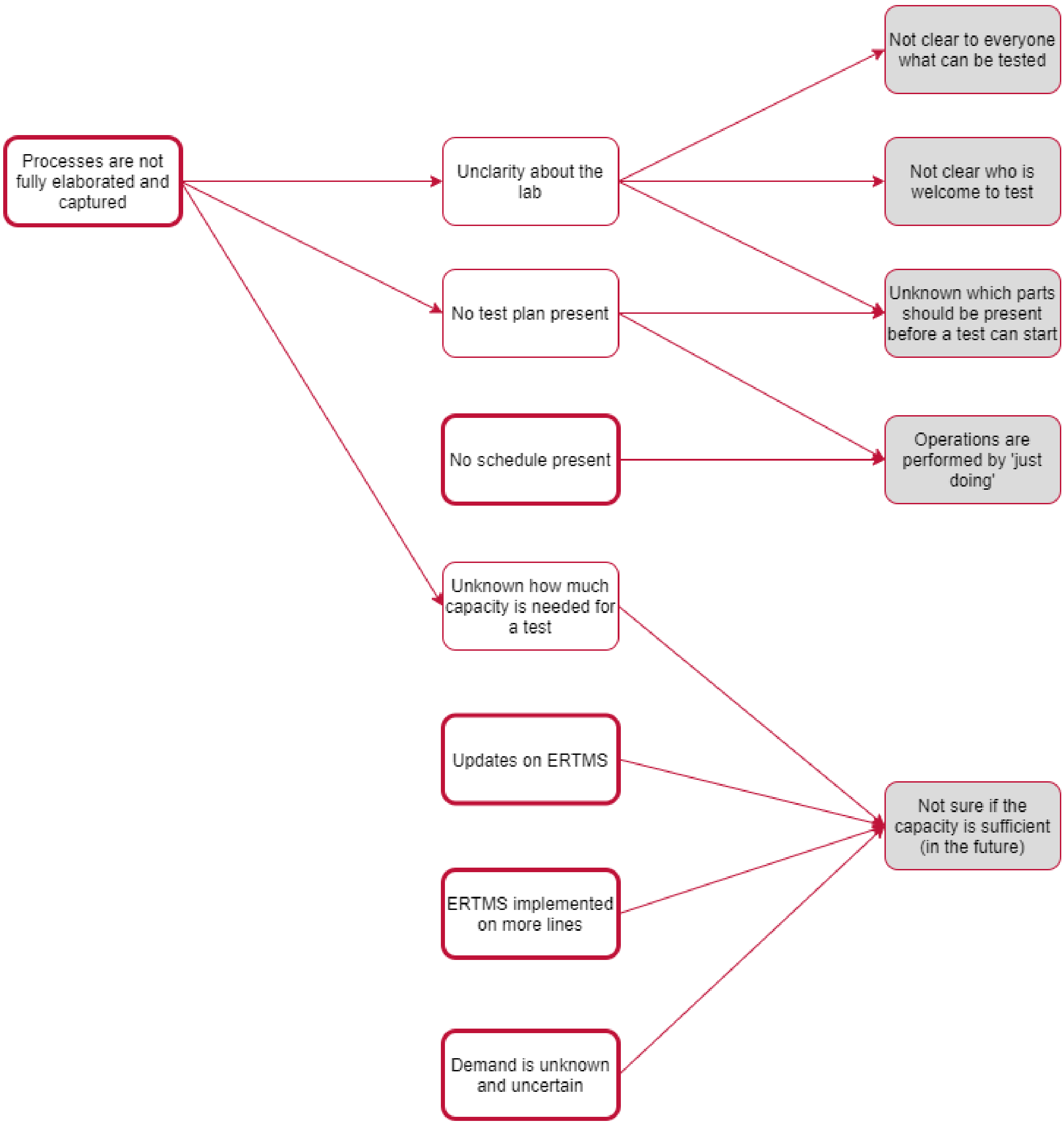

Figure 1.1: Problem cluster

The biggest need for ProRail is that the core-problem *Processes are not fully elaborated and captured* is solved. The solution to this problem has the highest value for ProRail. When the problem is solved, there will be more clarity around PREI.

At the moment, there is no schedule of the available resources in the lab. Due to the quietness, the lack of a schedule is not a big problem now. However, the allocation of the resources might become harder when the lab becomes more crowded. Therefore, the core problem *No schedule present* is also important for the integration lab.

The system specifications of ERTMS are out of the scope of this research. Updates on ERTMS are initiated by the European Union Agency for Railways (ERA), so ProRail has largely no influence on the updates of the system. Therefore, the core problem *Updates on ERTMS* is out of scope. As already said, the implementation of ERTMS on more lines in the Netherlands is already in progress. This is out of the scope of this research, as we do not have an influence on this. So, the core-problem *ERTMS implemented on more lines* is out of the scope of this research as well. The demand for the coming years has already been roughly estimated but is very uncertain due to regulations of the European Union and the developments in the sector. At this point in time, it is not possible to solve this problem. Therefore, this core problem (*Demand is unknown and uncertain*) is out of the scope of this research.

1.2.3 Research Goal

The main goal of this research follows from the problem description and problem cluster, and is:

> ***To design and align the processes of the ERTMS Integration Lab, and to develop a model that is able to allocate the resources of the ERTMS Integration Lab for the testing, demonstrating, and training activities.***

1.3 Scope of the Research

An important remark is that this research is focused on PREI and the processes within this lab. The operations about ERTMS itself, such as the development and implementation, are out of the scope of this research.

Designing all the processes of the lab is a quite big and broad task. In this research, a list of processes that apply to the lab is established. It is decided to focus on the most important processes and design these processes in detail, where the most important processes are the core processes. The other processes (management processes and supporting processes) are not elaborated in detail.

It is important to have the processes clear before it is possible to start analysing, changing and optimising them. The scope of the research may be large in the beginning but will narrow down once the processes are designed.

An important note: when talking about scheduling of activities, reference is made to the scheduling of tests, demonstrations or trainings, which require a workplace in the lab. A workplace is a place at the lab with twelve screens, where the activities can be performed. A workplace forms the operating interface of the testing environment.

1.4 Research Questions and Design

Based on the problem description and the scope of the research, the following main research question is extracted:

> ***How can the processes of the ERTMS Integration Lab be designed based on a suitable typology, and how can the resources be allocated to the activities under various growth scenarios?***

The research question is answered with the help of several sub-questions. The sub-questions are divided into different groups to keep the research clear. After the introduction of the sub-questions, the research approach is described.

Current Situation

The first set of sub-questions is about the current situation. It is important to know what the current situation is, as it is the starting point of the research. This results in the following sub-questions:

RQ 1: What does the current situation in the ERTMS Integration Lab look like?

1.1 What information about ERTMS and the ERTMS Integration Lab is already available?
- 1.1.1 What is ERTMS?
- 1.1.2 How does ERTMS work?
- 1.1.3 What activities can be distinguished in the ERTMS Integration Lab?

1.2 Which processes within the ERTMS Integration Lab can be distinguished?
- 1.2.1 How are the processes in the ERTMS Integration Lab performed now?

1.3 How are tests performed?
- 1.3.1 Who and what is necessary to perform a test?
- 1.3.2 Which tests can be performed?
- 1.3.3 To what extent is the set of tests complete, for now and the future?

1.4 How are the activities scheduled that require a workplace?

1.5 To what extent are there other (integration) labs that are useful for this research?

Literature Review: Process Design

Literature is needed to form a good basis. The research consists of two parts: process design and scheduling. The first literature review is about process design. In this literature review, the existing literature regarding process design is analysed to select the most suitable method for process design at PREI.

RQ 2: What methods are available in the literature regarding process design?

2.1 Which methods can be used to design and align processes?
- 2.1.1 What are the strengths and weaknesses of the methods?

2.2 Which method proposed in the literature is suitable for this research?
- 2.2.1 How can the chosen method be used to design the processes of the ProRail ERTMS Integration Lab?

Process Design

In this part of the research, the processes of the ProRail ERTMS Integration Lab are designed with the use of a suitable method, based on literature and the desires and requirements of ProRail.

RQ 3: How can the processes of the ProRail ERTMS Integration Lab be designed, such that the processes are understandable, workable and accessible to the stakeholders?

3.1 What do the designed processes look like?

3.2 Where should the designed processes be stored, such that everyone who should be able to access them, can do this?

3.3 To what extent are the designed processes understandable, workable and accessible to the stakeholders?

Literature Review: Scheduling

The next part of the research is regarding the allocation of resources. This part starts with a literature review as well. The literature review is done to find and analyse existing literature regarding scheduling. In the remainder of the research, this literature review is used.

RQ 4: What methods are available in the literature regarding the scheduling of activities?

4.1 Which methods for scheduling of activities on multiple workplaces are proposed in the literature?
   4.1.1 How can be dealt with uncertainty?
   4.1.2 How can prioritisation be included?

4.2 Which method proposed in the literature is suitable for this research?

Proposed Scheduling Model

When the processes are elaborated and a literature review is done on how the scheduling problem can be approached, a model can be developed to help ProRail with the scheduling of the activities in the ERTMS Integration Lab.

RQ 5: How can a model be developed to schedule the activities of the ERTMS Integration Lab that require a workplace?

5.1 How can a suitable scheduling method be put into a model?
   5.1.1 Which parameters are important?
   5.1.2 Which variables are important?

5.2 How to collect and process the inputs of the model?
   5.2.1 Which variables include uncertainty?
   5.2.2 How can a prioritisation be made, based on the operations?

Model Performance

Various experiments are performed to determine the performance of the proposed model. Also, the model should be validated. The following research questions are answered during the experiments.

RQ 6: How does the proposed scheduling model perform under various growth scenarios?

6.1 How does the proposed model perform when multiple activities have to be scheduled?
  6.1.1 How are the activities scheduled?
  6.1.2 How are the resources allocated?
  6.1.3 What does the utilisation of the capacity look like?

6.2 How can the model be validated?

Implementation

When the model is validated, it is important to determine how the model can be implemented at ProRail. ProRail will be able to implement and use the proposed model when an implementation plan is given.

RQ 7: How should the model be implemented at the ERTMS Integration Lab?

7.1 Who should be able to use the model?

7.2 How can the model be implemented?

7.3 What is needed to implement the model?
  7.3.1 What is needed on the harder side?
  7.3.2 What is needed on the softer side?

1.4.1 Research Approach

The main goal of this research is to solve the core problems as already defined in Section 1.2.2.1. The two core problems that are going to be solved ask for a different approach. In this section, we elaborate on the approaches.

The first core problem that is considered is the problem regarding the process design. The first step in solving this problem is to explore and analyse the processes. Due to the current pandemic, it is not possible to be present at the location. Therefore, the information and data will mainly be gathered with online meetings. The meetings are held with various stakeholders, such as managers, employees of vehicle authorisation and employees of the lab itself. This ensures that the processes are well understood. The next step is to find a suitable method to design the processes. This is done with existing literature and by using the information of the meetings about the requirements of ProRail. Also, other already designed processes are looked into, to be sure the chosen methods are in line with the existing organisation. When all processes are elaborated and captured, the results will be presented to all stakeholders for validation.

The second problem is regarding the lack of a schedule for the activities. To solve this problem, it is important to dive deeper into the process and to determine the parameters and variables of the activities. When the processes are understood and the parameters and variables are known, a literature review is done to investigate which scheduling method fits the problem and how this method can be modelled. The next step is to propose a suitable model and to validate the proposed model. The last step is to look into the implementation of the proposed model, such that ProRail can use the model in the best possible way. To determine the implementation plan, literature can be used, but it is of great importance that ProRail agrees with the plan. Therefore, meetings are planned to understand the requirements well and to get a better insight into the possibilities within ProRail. The results are presented again, to validate the model and implementation plan.

1.5 Deliverables

This research will deliver different products. The first deliverables are the designed processes of PREI. The second deliverable is a model that helps with the scheduling of the activities within the lab.

The substantiations of the deliverables will be captured in this thesis. The thesis can contain confidential information. Therefore, the thesis is assessed by ProRail before it is publicly distributed, so any confidential information can be covered.

1.6 Structure of the Report

The research questions are answered in this report. We start with a description of the current situation in Chapter 2. Here, we also explain ERTMS in more detail.

The research can be divided into two parts. The first part is about the process design. We start with a literature review on process design in Chapter 3, after which the processes of PREI are elaborated in Chapter 4.

The second part of the research is about resource allocation or scheduling. This part starts with a literature review as well in Chapter 5. A model to solve the resource allocation problem is proposed in Chapter 6. The proposed model is tested and validated in Chapter 7, and Chapter 8 includes important remarks on the implementation of the model.

The research is discussed and concluded in Chapter 9.

2 Current Situation

In this chapter, the current situation is discussed. We start with an introduction of the European Rail Traffic Management System (ERTMS) in Section 2.1. It is important to broadly know what ERTMS is and what it does, to better understand PREI and the processes within the lab. The lab is introduced in Section 2.2, and the current situation of the processes in the lab are analysed in Section 2.3. The availability of tests is discussed in Section 2.4. The current scheduling method for the activities is elaborated in Section 2.5, and in Section 2.6 the lab is compared to other integration labs. The chapter is concluded in Section 2.7.

2.1 European Rail Traffic Management System

The currently used system for railway protection in most of the Netherlands is ATB. This system is getting outdated and is not suitable for the increasing bustle on the tracks. Due to European regulations, further development of ATB is not allowed. ERTMS will be the new international standard for automatic train protection. The possibilities for further development of ERTMS are more extensive than the possibilities of ATB. This makes ERTMS more future proof.

Furthermore, the European Union obligates ERTMS for freight and passenger transport on the Trans-European Transport Network (TEN-T). By 2030, the most important (international) corridors should contain ERTMS. All countries had their own ATP system (ATB in the Netherlands) in the past, but most of these systems were not compatible with each other. Trains crossing a border should have different ATP devices implemented, or the train has to exchange the locomotive at the border crossing station. This changes when all parties implement ERTMS. This is what the European Union strives for, and therefore ERTMS is obligated in the (near) future (European Commission, 2020a). This will result in an interoperable railway system in Europe, for both passenger and freight transportation (European Commission, 2020a; ERTMS | The European Rail Traffic Management System, 2020).

Another regulation of the European Union states that specific newly built tracks should contain ERTMS as the only ATP system. So, when a new track is built in the Netherlands, ATB is not allowed anymore and ERTMS should be integrated into the tracks (ProRail, 2020c).

Because of the above-described facts and regulations, the new (international) system ERTMS is going to be implemented in the Netherlands. ERTMS offers advantages in terms of safety, reliability, speed, an increase in capacity and easier moving train traffic (Programma ERTMS, 2020). The Netherlands focuses on several goals when looking at ERTMS. The first goal is to increase capacity. Research has shown that the demand for rail capacity in the Netherlands will grow between 27 and 45 per cent until 2040. Major steps have to be taken to ensure that there is enough capacity. The implementation of ERTMS helps to increase capacity. Another important goal is to keep up with technology (ProRail, 2019). This can be ensured with ERTMS, as already discussed.

The implementation of ERTMS in Europe is a major project. The ERA is responsible for the ERTMS specifications, and the changes and additions of the specifications. When the ERA draws up specifications, the specifications are submitted to the European Commission (EC) as a proposal. The EC presents the proposal to the joint meeting of the Member States. When a proposal is adopted by the Member States, the proposal is included in the technical specifications for interoperability (TSI) and published by the EC. This is how new European laws are created (ProRail, 2020c).

In the remainder of this section, we elaborate on important components and versions of ERTMS, and ERTMS in the Netherlands.

2.1.1 Components

The two basic components of ERTMS are ETCS and GSM-R (European Commission, 2020c; ERTMS | The European Rail Traffic Management System, 2020). However, the system also consists of other parts that are not only ERTMS oriented. Figure 2.1 gives an overview of the components used within ERTMS.

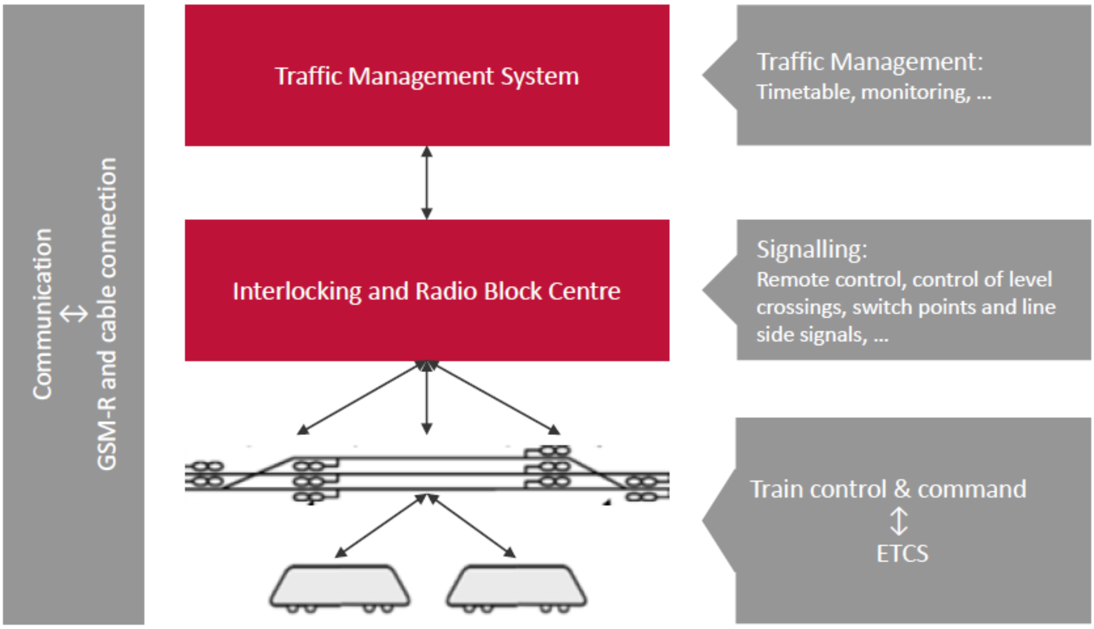

Figure 2.1: Overview components ERTMS

- **European Train Control System (ETCS)**

The European Train Control System (ETCS) is the control command part of ERTMS (ERA * UNISIG * EEIG ERTMS USERS GROUP, 2016). ETCS consists of trackside equipment and onboard equipment (European Commission, 2020c). Trackside equipment aims to exchange information with the train, so the train circulation is supervised safely. Information can be exchanged either continuous or intermittent, depending on the ERTMS level and the nature of the information (ERA * UNISIG * EEIG ERTMS USERS GROUP, 2016). An example of trackside equipment is a eurobalise, see Figure 2.2. A eurobalise is installed between the rails and provides information to ERTMS trains. Eurobalises are mostly placed in pairs. The distance between pairs depends on the characteristics of the block section. Using the eurobalises in the tracks, the position of the train and the direction of travel can be determined (system specialist, personal communication, September 15, 2020).

Figure 2.2: Eurobalise

- **Global System for Mobile Communications – Railway (GSM-R)**

The Global System for Mobile Communications – Railway (GSM-R) is the international standard for wireless railway communication and railway applications. GSM-R is used for voice and data communication within ERTMS and is the radio bearer for ETCS (ERA * UNISIG * EEIG ERTMS USERS GROUP, 2016).

- **Interlocking (IXL) and Radio Block Centre (RBC)**

Interlocking (IXL) and Radio Block Centre (RBC) are important components of the signalling system and are used as the central safety unit. They ensure safe routes and train movements. GSM-R is used to receive train position information and to send movement authorities and track data to trains. Information regarding signalling and route status is obtained by an interaction between the RBC and IXL (European Commission, 2020b).

- **Driver Machine Interface (DMI)**

The Driver Machine Interface (DMI) is installed in the cabin. The DMI allows the drive to enter the required input data and visualises the output data to the driver (European Commission, 2020b). An example of a DMI is shown in Figure 2.3.

Figure 2.3: Example of a Driver Machine Interface (ERSA by CLEARSY, 2021)

2.1.2 Versions of ERTMS

There are different versions of ERTMS, specified in three levels. In all levels, both the tracks and trains are equipped with ERTMS (ERA * UNISIG * EEIG ERTMS USERS GROUP, 2016).

- **Level 1**

Level 1 is designed as add on to a line with signals, and trackside train detection equipment that locates the train. Eurobalises are installed on the track and are connected to the control centre. The eurobalises contain pre-programmed track data and pre-programmed movement authorities. The train detection equipment sends the position of the train to the control centre. The control centre receives all positions of all trains on the line. Based on all positions, a pre-programmed movement authority in the eurobalise is selected. When a train passes a eurobalise, the train receives the movement authority and track data. The onboard computer calculates the speed profile for the movement authority and the next braking point. The information is displayed to the driver, on the DMI in the cabin.

- **Level 2**

Level 2 is more digital than Level 1. Signals are not needed anymore in Level 2, but the train detection equipment in the tracks is still used. The trains are equipped with an onboard radio system, that allows the onboard computer to communicate with the RBC using GSM-R. The eurobalises on the track are used as position markers. Track data is sent by the RBC to the onboard computer in the train. The train detection equipment sends the position of the train to the RBC. The RBC receives all positions of all trains on the line. Based on all positions, the RBC determines the movement authorities and sends them directly to the trains using GSM-R. The onboard computer calculates the speed profile for the movement authority and the next braking point. The information is displayed on the DMI in the cabin. The onboard computer determines the position of the train continuously and checks if the current speed is correct for the location.

- **Level 3**

Level 3 does not require train detection equipment on the tracks, as the train is equipped with an onboard train integrity system that monitors if the train is complete. The trains are equipped with an onboard radio system, that allows the onboard computer to communicate with the RBC using GSM-R. The eurobalises on the track are used as position markers. Track data is sent by the RBC to the onboard computer in the train. The onboard computer determines the position of the train continuously and checks if the current speed is correct for the location. The onboard computer sends its position via the train radios to the RBC. The RBC receives all positions of all trains on the line. Based on all positions, the RBC determines the movement authorities frequently and sends them directly to the trains using GSM-R. The onboard computer calculates the speed profile for the movement authority and the next braking point. The information is displayed on the DMI in the cabin.

2.1.3 ERTMS in the Netherlands

The Netherlands decided to implement ERTMS Level 2. Therefore, this level is used to explain how ERTMS works in more detail. In Figure 2.4, a schematic overview of ERTMS Level 2 is depicted.

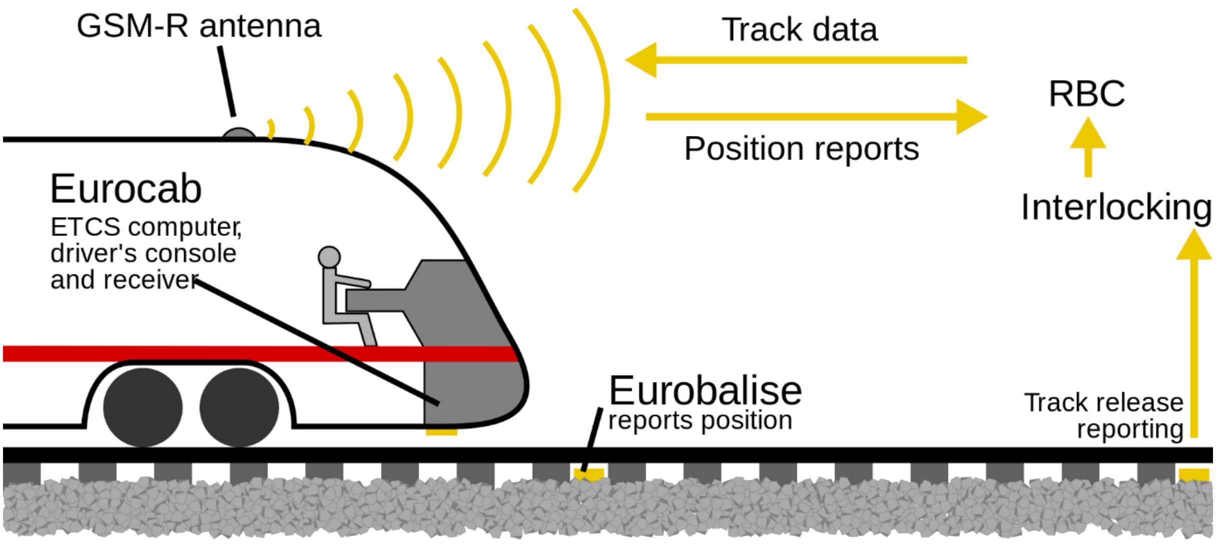

Figure 2.4: Schematic overview of ERTMS Level 2 (European Union Agency for Railways, 2017)

As already stated, ERTMS is a digital system that works with wireless communication. With ERTMS Level 2, there is continuous contact between the train, the track and the rail traffic control. GSM-R is used to pass information about the route and the maximum speed. The eurobalises on the tracks are used to determine the position of the train. The position of the train is reported to the RBC. It is also reported to the RBC when a block section is empty. Because the RBC knows the position of the train and whether the route of the train is clear, the RBC can send the necessary movement authorities to the train if possible.

The information about the route and the maximum speed is shown on the DMI in the train. Because the signalling is done digitally, the physical signals as currently used become redundant. When a train driver does not follow the instructions of the system on time, the system will intervene at any location, speed and time. The speed can be adjusted, or the train can be brought to a stop. Because the system intervenes automatically when the train driver does not react (in time), ERTMS reduces the chance that a train will enter a not (yet) allocated route. Safety is also insured at higher speeds compared with ATB because ERTMS can determine and monitor the maximum speed of the train. Furthermore, ERTMS offers possibilities to better and easier reserve a part of the track, for example for maintenance. This results in a safer workspace for track workers.

Sufficient distance between trains is required to guarantee safety on the tracks. With ERTMS, the distance between trains can be shortened. Trains can therefore follow each other quicker. The driving times on some routes can also be shortened due to higher speeds. This results in shorter travel time for passengers. Faster train follow-ups can improve the stability of the timetable and thus the reliability of the rail network (Programma ERTMS, 2020).

2.1.3.1 Implementation of ERTMS

The implementation of ERTMS in the Netherlands already started and should be finished by 2050. The implementation is done in multiple steps: not all lines are adapted to ERTMS at the same time (Programma ERTMS, 2020). The first four lines are already equipped with ERTMS (the blue lines in Figure 2.5):

- Betuweroute,
- Line Amsterdam-Utrecht,
- Hogesnelheidslijn-Zuid (HSL-Zuid),
- Hanzelijn.

Figure 2.5: Lines with ERTMS in the Netherlands (Ministerie van Infrastructuur en Waterstaat, 2020)

ERTMS is going to be implemented in all trains as well. The implementation of ERTMS will positively influence the safety and the speed of the (European) connections (Programma ERTMS, 2020).

2.2 PREI

ERTMS is a new system. A new system needs to be tested before it can be fully operational. The tests of the separate subsystems are done by the suppliers themselves. When a component is developed, the component is extensively tested and certified. But as explained in Section 2.1, ERTMS consists of different components. The interaction of these components should be tested as well. It is even obligated to demonstrate that the whole system works before a train may be operational on the tracks. It is for example very important that the train gets the right information from the RBC and the other way around. To perform those tests, PREI (ProRail ERTMS Integration Lab) is established.

2.2.1 Activities

Various activities are performed at PREI. The most important activity is testing. The need for a testing environment was the biggest reason why PREI is established. One group of tests done at PREI are ETCS System Compatibility (ESC) tests. The European Train Control System (ETCS) is the control command part of ERTMS. During an ESC test, the integration of the train equipment and the track equipment is tested, based on the ESC guidelines (RLN00445). The ESC guidelines are composed by ProRail and contain tests for all lines that are equipped with ERTMS. Before a test starts, it is determined what the initiating party (applicant) of the test should demonstrate, and thus which tests should be performed exactly. However, not all tests in the ESC guidelines can yet be tested in the lab. An example is a border crossing, as only subsystems for the Dutch infrastructure are available. Also, links with foreign labs are not yet established. When all equipment that is needed to perform a test is available and it is clear which tests should be done in the integration lab, the testing can start.

Other tests are performed as well, such as functionality tests of other systems or cybersecurity tests. Next to the tests, demonstrations and trainings are performed with the ERTMS equipment.

Figure 2.6: Demonstration workplace

2.2.2 Resources

PREI has seven regular workplaces and one demonstration workplace. The demonstration workplace (Figure 2.6) could also be used as a regular workplace, as the equipment is almost the same. The demonstration workplace has one extra: a look-alike train operator cabin (on the right of the figure). A workplace contains twelve screens (on the left of the figure) to show all relevant operations, such as the train simulation and information on the infrastructure. This is enough for most activities. When a test (campaign) needs more than twelve screens, two workplaces can be combined. The needed data for an activity can be loaded on all workplaces. However, it is not desired to perform tests of different test campaigns on the same line simultaneously, because the tests can influence each other. A test campaign is a series of tests that needs to be performed.

Next to the workplaces, the lab contains a data centre. In this centre, the Centralised Safety Systems (CSS) systems of the infrastructure are stored. CSS is the most important part of the ERTMS functionality. The systems in the data centre are identical to the systems used on the real tracks. Therefore, it was possible to create appropriate simulations and emulations of the tracks. Also, hardware for the ERTMS functionality is present in the centre. Non-direct ERTMS functions, such as the movement of the train over the infra, are simulated. The ERTMS functions are not simulated but are exact copies of the real products. The required data and hardware can, together with the equipment in the cabinet, be connected to the screens.

At this moment, four lines are already equipped with ERTMS in the Netherlands. The Betuweroute, HSL-Zuid and Hanzelijn are also represented in the lab, so activities can be executed on those lines. It is not possible yet to do tests on the line Amsterdam-Utrecht, as not all test connections are accomplished yet. When more lines are equipped with ERTMS, the CSS of those lines are added to the lab.

The responsible party of the train that is involved in the test or other activity delivers a cabinet with the ERTMS equipment (On-Board Unit, OBU) and a simulation of the train (On-Board Unit Adapter, OBU Adapter). The train equipment is delivered in a cabinet (behind the glass door in the middle of the figure).

Other resources that are necessary for some of the activities are the Post21 Functionality and Hand Held Terminals (HHTs). The Post21 Functionality is the traffic management system and can be used only once simultaneously. There are three HHTs available.

PREI is not only used for testing new or modified equipment. Other activities performed in the lab are demonstrations and trainings. In general, the same equipment is needed for these activities: the systems are necessary to demonstrate or train how the systems work (system specialist, personal communication, September 15, 2020).

2.3 Processes at PREI

Various processes can be distinguished at PREI. To be able to elaborate better on the processes, they are divided into two categories for now:

- Management of the lab,
- Testing and other activities.

The management processes contain everything that is needed to ensure the lab is available for possible tests, demonstrations and trainings. The testing processes contain everything that is needed to organise, run and complete the tests (and other activities) properly. Both categories are elaborated in the remainder of this section. Also, the documentation of the processes in the current situation is discussed.

2.3.1 Management

Management is needed to keep the lab up and running and to be able to create valuable output. The management of the lab consists of several processes, roles and responsibilities. In this section, the current situation of the processes related to the building and its setup, and the currently involved employees are elaborated.

2.3.1.1 Building and Setup

PREI is situated in the Railcenter in Amersfoort. The workspace of the lab is rented from the Railcenter. The rental contract is established by the facility management of ProRail and the Railcenter. If adjustments have to be made to the building, this is done and paid for by ProRail. All maintenance related to the premises is done by the Railcenter, also for the adjustments done by ProRail. Other services, such as filling the coffee machines and cleaning, are in hands of the Railcenter as well. These services are established with service-level agreements (SLAs). Within ProRail, the manager ERTMS Central Systems is responsible for the SLAs. The Railcenter has a planning system, which ProRail can use to plan activities on the workplaces.

The setup of the lab is the property of ProRail. Everything in the lab, including the software and hardware, is purchased by ProRail. The maintenance is also in hands of ProRail. However, at this moment, it is considered to draw up maintenance contracts with the suppliers of the systems.

2.3.1.2 Involved Employees

There are several employees involved in the lab. At first, there is a senior ERTMS system specialist. This person also holds the role of caretaker of the lab in the current situation. It is his responsibility that the lab is up and running. He coordinates, or carries out where possible, various activities in the lab. When a test should be planned, for example, he is consulted and ensures that the supplies are available. Also, he supports the activities (test campaigns for example) of the lab. He knows how the lab and its equipment work.

The system specialist is supported in his operational tasks where necessary and possible. This is currently done by one person with experience with and in the Railcenter: a junior ERTMS system specialist. However, it is being examined whether this is enough for now and the future, as an increase in workload is expected.

Larger projects, and projects and activities outside the operational activities in the lab itself, are carried out by a project leader. Things around the lab are arranged by this person. Examples are public relations and the website.

The management of the lab is done by the policy advisor ERTMS. He overviews the lab. When adjustments or developments have to take place, he is responsible for the result. The adjustments or developments are often proposed and initiated by the system specialists, but the policy advisor ERTMS ensures that it can happen and that the funds are concentrated.

In the current situation, the system specialist and the project leader both report to the policy advisor. However, the lab and its roles are quite new. A strong hierarchy is therefore not present. The policy advisor in turn reports to the manager ERTMS Central Systems. This manager is part of the management team of the department "Techniek" (technology). This structure is visualised in the organigram in Figure 2.7. Also, the responsibilities of the different roles are included.

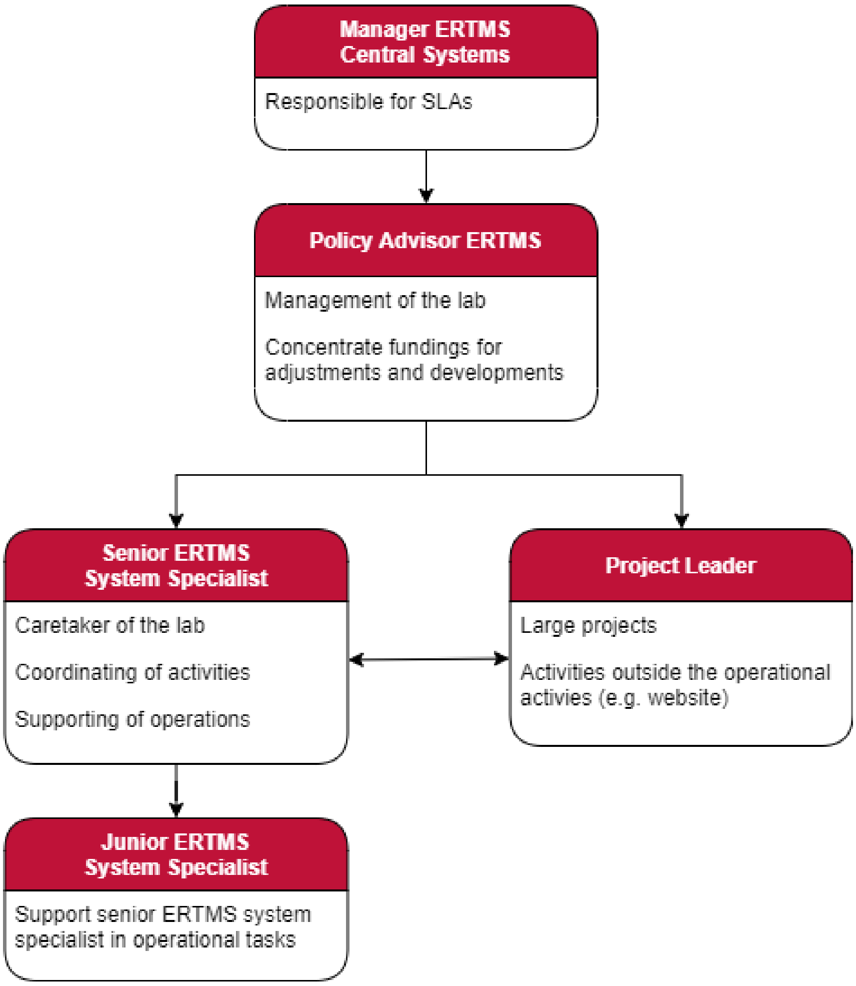

Figure 2.7: Organigram current situation. Every block corresponds to one employee and their responsibilities

2.3.2 Testing

The main purpose of the lab is to perform tests. Different types of tests can be performed at PREI. The biggest group of tests are the tests on the integration of ERTMS onboard equipment and infrastructure: ESC tests. In addition, tests can be performed to test the Post21 Functionality, but also other tests can be distinguished, such as tests for cybersecurity.

In the remainder of this section, we will elaborate on several aspects of the tests. At first, the stakeholders are discussed. Also, the performance of a test is discussed. A test campaign is a process with different steps. The steps are more or less the same for each type of tests. Some data and specific information are available of the processes of the different types of tests and, of course, there are costs involved.

Testing is not the only value-adding activity of the lab. Other activities that add value to the lab are demonstrations and trainings. These activities do look very familiar with the testing process, but some differences can be distinguished as well. The differences are elaborated in this section.

2.3.2.1 Stakeholders

The testing process is a process with multiple stakeholders, both internal and external. For every test campaign of ESC tests, a test manager is appointed. A test manager is someone who is competent for the job and is not influenced by the parties involved in the decision making. This role does not have to be fulfilled by someone from ProRail. The initiating party is the organisation that developed new equipment or updated their equipment, and has to perform ESC tests.

PREI is the property of ProRail. Therefore, ProRail is responsible for the workplace. They are responsible that the workplace is up to date and ready to use. It is important that the lab is up to date to be able to represent real infrastructure. Another responsibility of ProRail is to facilitate, witness, or execute the tests. This is dependent on the type of test and the agreements with the involved parties.

ProRail is responsible for the infrastructure in the Netherlands. Therefore, it is their responsibility that the different lines are represented in the lab. The central part of the CSS (such as the RBC) and the Infra Simulator are purchased from the supplier of the infrastructure. The Test Control & Logging (TCL) controls and logs the test and was also purchased from the supplier in the past. However, there are different suppliers for the infrastructure, and they have their own TCLs. So, not every line has the same TCL, making the test environment look different. Also, the method of connecting the train and the infrastructure is slightly different with the different TCLs (often supplier dependent), and not all TCLs meet the expectations of the testers. Therefore, ProRail is thinking about creating its own TCL. Especially because more lines are going to be equipped with ERTMS (system specialist, personal communication, September 15, 2020). If there is a standard TCL, there will be a more standardised way of working.

Every train type that has to operate on an ERTMS track, has to be tested. It is the responsibility of the railway undertaking that the train is authorised. Different railway undertakings, for domestic and international transport of passengers and freight, have an access agreement with ProRail and use the Dutch railway infrastructures. The list of railway undertakings is very long, but the most known ones for passenger transport in the Netherlands are (ProRail, 2020a; NVBS, 2020):

- Abellio Rail NRW
- Arriva B.V.
- Connexxion Openbaar Vervoer N.V.
- DB Regio AG
- Keolis Deutschland GmbH en Co. KG
- Keolis Nederland B.V.
- NS Reizigers B.V.
- Qbuzz B.V.
- NS International B.V.

A railway undertaking can decide to put the responsibility for the tests on the ERTMS supplier of train supplier. In that case, the train supplier should test the integration of the ERTMS onboard equipment in the lab. To connect the ERTMS onboard equipment with the lab environment, it is the responsibility of the railway undertaking or the train supplier that the OBU and OBU Adapter are present at the lab.

Different stakeholders, their roles and responsibilities are summarised in Table 2.1. *Test facility manager*, *(Account) manager* and *Tester* are roles within ProRail that are involved in the testing process. The test facility manager is the person who is responsible for an up to date lab. The (account) manager of the test is responsible for everything around the tests. This person is for example the contact person for all PREI related affairs for all involved parties. The tester is responsible for (witnessing or facilitating) the execution of the test plan.

Table 2.1: Stakeholders testing process with their roles and responsibilities

Stakeholder	Role	Responsibilities
ProRail	Owner of PREI	Up to date, representative and ready to use workplace (*Test facility manager*)
		Arranging tests (*Account manager)*
		Perform/witness/facilitate tests (*Tester*)
		Presence of all systems (CSS, Infra Simulator, TCL, etc.)
	Owner of the infrastructure in the Netherlands	All lines with ERTMS are represented in the lab
	Test new developments (equipment, software, etc.)	The needed resources are present
Railway undertaking or train supplier	Test new or modified onboard equipment	OBU and OBU adapter are present

2.3.2.2 Performing an ESC Test

The lab is called in when equipment needs to be tested. It is the responsibility of the test manager that a sound test plan is composed. However, this is not always the case in the current situation. It is also not always clear what should be delivered to the lab before a test can start and what this should look like. Because it is still quiet at the lab, there is more time to perform tests and it is less important that the test plan is sound. It is the responsibility of the test initiator. However, when the lab will be more crowded in the future, this is not desirable.

When a test plan is composed and the tests can be started, the OBU and OBU Adapter are connected to the TCL. This is done by ProRail at PREI. But as stated before, this connection is not always the first time right. The specifications that the OBU Adapter and the TCL must meet have been composed in such a way that there is still some room for different interpretations. Therefore, this process step contains some uncertainty. Connecting can take some minutes, but it has also lasted several days in the past. When that is the case, the supplier of the OBU Adapter and TCL are needed to adapt the systems. Because the suppliers are different and the problem is not always easy to determine, this can take some time.

The systems can be started when the connection is established. Once the train is placed on the right track, the tests can start. The tests are based on the ETCS System Compatibility (ESC) checks. The checks describe exactly what must be demonstrated in a particular test. A test plan is composed based on the needed checks. The test plan is followed in the lab and the different tests are performed. This process step also contains some uncertainty. As stated earlier, the test plan is not always solid. It is possible that the tests are not performed efficiently, because the order in the test plan was not efficient. For example at the HSL-Zuid line. The HSL-Zuid consists of two sections. It can take two hours to switch the test environment between the sections.

A test can also fail. If that is the case, ProRail helps to find out why the test failed to some extent. For example, if the settings were not correct, the test will be repeated with the right settings. If the test failed because the equipment contains errors, the test initiator will have to further develop the equipment and the test should be performed again at another time.

The outcomes of all tests are captured in a preliminary test report. The preliminary report is analysed by the test manager, OBU supplier, the infrastructure manager and if required, the trackside supplier. When the preliminary test report uncovers issues, the issues have to be handled by the right party. Once the issues are solved, it can be necessary to re-execute the tests or re-evaluate the test results. When all parties agree with the preliminary test report, the test manager composes the final test report. The final test report is sent to the notified body, which checks the outcomes. When the confirmation of the notified body is received, the test initiator informs the OBU supplier, and the OBU supplier draws up the ESC statement.

2.3.2.3 Performing a Post21 Functionality Test or Other Test

The process of Post21 Functionality tests and 'other' tests look different than the process of ESC tests. The process of ESC tests contains many steps, whereas the Post21 and other tests contain fewer steps. One of the reasons why this is the case is that the stakeholders in Post21 and other tests are mostly internal (ProRail), and the steps do not have to be elaborated in detail. The Post21 tests and other tests are combined into one group in the remainder of this research, as the processes look very similar.

The process always starts with an intake meeting. During this meeting, the test initiator and the account manager discuss the characteristics of the intended test. When they agree about the characteristics, the test initiator is supposed to compose a test plan. This test plan includes the characteristics of the test as agreed, such as the intentions, goals and needed resources.

The account manager makes sure that the needed resources are available when the test is performed. The setup of the workplace is also done by the account manager. However, for some tests, the setup of the workplace is part of the test. In that case, the account manager does not have to set up the workplace. The test initiator is responsible for the start and execution of the test, and for recording the findings. A PREI tester should always be present in the lab to provide (technical) support during the execution of a test.

The environment can be changed during a test. When a Post21 Functionality test is performed, for example, the Post21 Functionality can be (slightly) changed. As stated earlier, ProRail is responsible for an up to date, representative workplace and environment where tests can be performed. When, as in the example, the Post21 Functionality is changed, it is decided whether the environment is still representative after the test is completed or not. When the environment is not representative anymore, the test initiator should make sure that the environment is changed back to make it representative again.

2.3.2.4 Costs

There are costs associated with the testing process. The rule of thumb is that the initiating party pays for the test. So, when for example the infrastructure of the tracks changes and new tests have to be performed, the responsible party for the infrastructure has to bear the costs. And when a new train wants to enter the ERTMS tracks, the responsible party for the new train has to pay. When an internal project needs to test at PREI, the incurred costs are covered by the project.

2.3.2.5 Other Activities

Other activities that are performed in the lab and add value to the lab are demonstrations and trainings. During a demonstration, it is shown what the lab does in an instructive form. Demonstrations are given to several types of interested parties. The trainings that are currently given in the lab are the initial education and the re-instruction "LeiderWerkplekBeveiliging" (LWB). Broadly the same equipment is needed for these activities and the testing activities. However, there are some differences in the processes.

The stakeholder roles are the same for all test campaigns, although the specific party or person can differ. The stakeholders and their roles are very activity specific for demonstrations and trainings.

The preparation phase has differences as well. With a demonstration, it often does not matter which line and train equipment is used. Also, a test plan is not necessary, as there is nothing specific that should be demonstrated. This is slightly different for a training. With a training, specific actions should be taken. It is useful to make a plan, although this plan looks different than a test plan for a test campaign.

The next differences occur in the executive phase. Examples of the differences are the workplaces or employee support needed. The characteristics of demonstrations and trainings are more case-specific than the characteristics of the testing process. However, the processes are broadly the same for all activities: workplaces are used to test, demonstrate or train on ERTMS equipment or the integration of ERTMS equipment.

The outputs of the different kinds of activities are also different. As already said, the most important output of the testing process are the results of the tests. A possible output of a training can be a certificate that proves participation. The most important output for a demonstration is that the audience understands what is done and that any questions are answered.

There are also differences in duration and resources. These differences are included in the next section.

2.3.2.6 Available Data

In the past, no data is recorded of the test campaigns except the lessons learned. There are two main groups and a residual group in which the tests can be divided: ESC tests, Post21 Functionality tests and other tests. In the process description above, the Post21 Functionality tests and other tests were combined into one group, because the processes are similar. However, some characteristics, e.g. duration, differ. Therefore, it is decided to discuss the groups separately here.

The setup of the workplaces is designed in such a way (e.g. twelve screens) that one workplace is sufficient for most tests. When more than twelve screens are required, two workplaces are used. The other resources (Post21 Functionality, line, etc.) a test requires are dependent on the test characteristics and its goals. This should be agreed on at the beginning of the process.

- ESC Tests: Interoperability tests with ERTMS infrastructure and ERTMS onboard equipment
  - PREI tester is present during test execution to facilitate and witness tests.
  - RLN00445 guidelines are used to determine the specific tests.
  - In the past: an average of one week of setup time and one week for the execution per line.
- Post21 Functionality Tests: Integration tests with Post21 systems (in ProRail Test Centre) and ERTMS test environment (in PREI)
  - The test initiator is responsible for the execution of the tests. PREI tester is present to provide (technical) support.
  - These tests are characterised by several test phases. During a term of, for example, a year, there are several time blocks of one week (on average) in which tests are performed. Setup time is not applicable, as this is part of the test.

- Other tests: E.g. Proof of Concept for Cybersecurity and Tests with GSM-R
    - The test initiator is responsible for the execution of the tests. PREI tester is present to provide (technical) support.
    - Durations are dependent on the test.

In 2019, demonstrations were given about two to three times a week in the lab. The average duration of a demonstration, including setup and completion, was approximately 2 hours. The necessary resources are quite flexible. A demonstration is done in the demonstration room and with the Post21 Functionality. The ERTMS test environment (line, train) does not matter that much.

The trainings in the lab are performed by the Railcenter. An employee of the lab should still be present for support. If the technology fails, for example, they are there for assistance to prevent inconveniences. The resources needed for the trainings are three HHTs, the test environment of the Hanzelijn or the Betuweroute and the Post21 Functionality. The durations and maximum group size differ per training. The maximum group sizes are smaller now due to COVID-19.

- Initial education LWB
    - Three working days needed in the lab.
    - Maximum group size of eight participants.
- Re-instruction LWB
    - One working day needed in the lab.
    - Maximum group size of four participants.

2.3.3 Documentation

Most described processes are not captured yet. However, there is information available that can help understand the processes.

2.3.3.1 Management

The management processes are not fully documented, although some processes are running already. There are some documents available at ProRail that include information about the ideas of the lab and rough expectations and vision. However, an important note about this information is that it cannot be simply copied, as the content needs to be completed and formalised. When this information will be used, it is necessary to carefully consider what information is useful and how it can be used.

2.3.3.2 Testing

On European level, the principles for the demonstration of ESC have been recently described. The resulting PREI test execution process is not documented yet in the current situation. There is no document, scheme or other data available that describe the test execution process. In the principles, an organisational framework is defined that supports the conduction of ETCS Test Campaigns in an efficient, flexible and reliable way. The principles include a description of the overall test process, the participants and the respective contributions. All information in the principles is intended to be used to perform ESC Tests (Schuster, 2019; European Union Agency for Railways, 2020).

The description of the overall test process in the principles includes all steps necessary for a testing campaign, from the beginning to the end. The overall process is documented in a flowchart and an explanatory table. In the flowchart, the flow through the process is visualised. A table is added to the flowchart that includes written remarks on the steps.

2.4 Test Availability

There are three complete lines available in the lab on which activities can be done: Betuweroute, HSL-Zuid and Hanzelijn. The lab has the ERTMS equipment of the line Amsterdam-Utrecht, but it is not yet implemented in the test environment of the lab. These four lines are the first lines where ERTMS has been implemented in the infrastructure in the Netherlands. To demonstrate whether the integration of ERTMS trackside and ERTMS onboard equipment is going well, different tests are determined and composed for these lines.

The possible ESC tests are included in the RLN00445 guidelines. These guidelines describe the ProRail ESC tests, which are based on European regulations and are approved by the ERA. The guidelines contain mainly tests for possible errors and are composed of lessons learned in the past. It must be demonstrated that this is going well before a train is allowed to enter the tracks. When a test fails and after investigation appears that the ERTMS specifications are incorrect or unclear, a change request can be submitted to the ERA. The ERA will investigate the request and probably propose it to the EC. When the change request is accepted, the specifications for ERTMS are adapted.

The guidelines need to be adjusted when more lines are equipped with ERTMS and tests should be performed on those lines as well. Tests for those lines have to be determined and composed. The most important test is the confidence run. This test demonstrates if the train can run under normal conditions on the whole line without any problems. For each line separately, it will be necessary to determine which additional tests are needed.

Most of the tests in the guidelines can be performed in the integration lab. However, there are also tests included that cannot be performed in the lab yet. The border crossings, for example, cannot be tested in the lab yet, as the infrastructure of other countries is not available and there is no connection with other labs. For those tests, on-site tests have to be performed on the tracks.

Which tests should be performed depends on the situation. And as stated earlier, not all tests can be performed in the lab yet. Therefore, it is important that someone checks which tests are needed to demonstrate the integration and where these tests can and should be performed. The initiating party proposes a list of needed tests, after which the infrastructure manager checks this list. The infrastructure manager can indicate what the initiating party of a test must demonstrate before they are allowed to use their equipment on the real tracks.

2.5 Scheduling of the Activities

One of the problems that is going to be solved in this research is the lack of a scheduling method for the activities that require a workplace. The lab is not fully utilised yet: there are seven workplaces, with only three lines present. The lack of a scheduling method is not noticed as a big problem right now, but the problem may arise when the workload of the lab increases.

There is one fulltime employee in the lab who is able to manage the activities, and there is one employee in training to be able to manage at least the line HSL-Zuid (September 2020). When a test, demonstration or training has to be done, it is mentioned to the lab. In the current situation, the agendas of the employees and the workplaces are consulted to find a suitable time and place to perform the activities, and the activity is scheduled. This works fine at the moment, but this could cause problems in the future. Also, when it becomes more crowded and activities have to be performed simultaneously, resource constraints can affect the schedule. At this moment, there are for example three HHTs. So, no more than three HHTs can be used simultaneously.

2.5.1 Toy-Sized Problem

The mentioned scheduling problem can be explained with a toy-sized problem. In a toy-sized problem, less data is used than in the real sized problem. In this case, we chose to compose a list of ten different activities. For every activity, it is stated what kind of activity it is, which line is needed, the release and due date, and the processing time. An important note is that the activities in this list are just examples and not completely truthful. Table 2.2 displays the activities and their parameters. Five lines are used. The Havenspoorlijn was not introduced earlier, but this line is added here as it is already equipped with ERTMS Level 1.

- Betuweroute,
- Line Amsterdam-Utrecht,
- Hogesnelheidslijn-Zuid (HSL-Zuid),
- Hanzelijn,
- Havenspoorlijn.

Activity number ten does not require a specific line. The lines Amsterdam-Utrecht and Havenspoorlijn are not yet available at PREI, but these lines are the first lines that are going to be added. In the future, even more lines are added, so the list of lines used in this problem is still small in comparison to the future situation. Therefore, it is chosen to limit the available workplaces to four: three regular workplaces and the demonstration workplace. For this problem, it is assumed that every activity requires one workplace.

Table 2.2: Activities toy-sized problem and their characteristics

Number	Activity	Line	Release date (day)	Due date (day)	Processing time (days)
1	Test ESC	HSL-Zuid	0	20	10
2	Test ESC	Amsterdam-Utrecht	0	15	10
3	Test ESC	Amsterdam-Utrecht	5	25	10
4	Test ESC	Hanzelijn	10	30	10
5	Test ESC	Betuweroute	5	35	10
6	Test ESC	Havenspoorlijn	15	25	10
7	Test Other	Hanzelijn	25	40	5
8	Training	Betuweroute	0	5	3
9	Demonstration	Hanzelijn	0	5	0.5
10	Demonstration	Flexible	10	15	0.5

The release date is the moment in time when the activity can start and the due date is the moment the activity should be finished. For some activities, the time window is larger than for other activities. The time window for activity number one is for example twenty time units, whereas the time window for activity number six is only ten time units. This does not mean that these activities require the resources for twenty and ten time units, respectively, but that they can be performed in a time span of twenty and ten time units, respectively. The time units in this problem are defined as working days. It is assumed that a week contains five working days. Activity number seven has the latest due date: working day 40. This list can therefore be seen as a fictional list of activities, spread over a time window of eight working weeks.

The processing times are based on the estimates from the past, as described in Section 2.3.2.6, and expressed in working days. They include the eventual setup and completion times. In Section 2.3.2.6 is stated that the needed time for a demonstration is two hours. However, in this problem, it is assumed that a demonstration will take half a working day because of for example group movements.

The set of defined activities is just a small sample of real-life activities. Therefore, it is still relatively simple to schedule the activities. Figure 2.8 shows a feasible solution for this scheduling problem. The activities are scheduled on one of the available workplaces, based on their release and due date, and processing time. Every activity is depicted by a coloured bar with a number in Figure 2.8. The number corresponds with the number in the first column of Table 2.2. The colour indicates the line on which the activity should be executed.

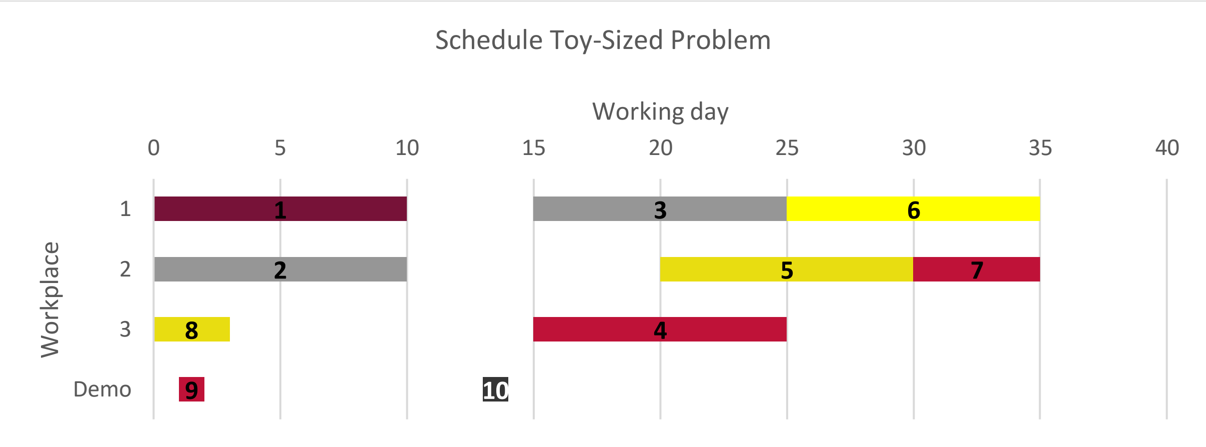

Figure 2.8: Schedule toy-sized problem

In this schedule, we see that some constraints are included. It is for example not possible to execute different activities on the same line simultaneously. When looking at the colours, it becomes clear that this is never the case: there is no overlap in colours, at any time. Also, it is preferred to execute demonstrations at the demonstration workplace. Activities nine and ten are demonstrations and are both scheduled at the demonstration workplace. It was also not needed to schedule other activities at the demonstration workplace, although it is possible to use the demonstration workplace for other activities (tests or trainings). Other constraints regard the release and due dates. The activities can only start after the release date. This constraint is met for every activity. The activities should also be completed by the due date. This constraint is met for every activity as well.

However, there are constraints in the real-life problem that are not included in this example. Section 2.3.2 describes which resources can be necessary to perform an activity: HHTs, Post21 Functionality or PREI support (employees). These resources are not included. Therefore, it is not possible to look at whether the constraints for these resources are met within this schedule. One can imagine that scheduling activities with more characteristics and therefore more constraints that should be met (e.g. the Post21 Functionality can be used only once at the same time) is a more complex task.

The schedule of this simplified, toy-sized sample is a feasible solution to the scheduling problem. However, the depicted schedule in Figure 2.8 is not the only feasible solution. Another feasible solution can for example be created when activities one and two are swapped (Figure

2.9). They are executed in the same time window, but they are now executed at another workplace. Also, different gaps can be distinguished in the schedule. A new schedule can arise when gaps are filled. When activity five is started earlier, activity seven can be started earlier as well (Figure 2.10). The question is which schedule is the best. This is highly dependent on the situation. Later on in the research, the best scheduling method is determined for the scheduling problem at PREI. This will result in good and suitable schedules.

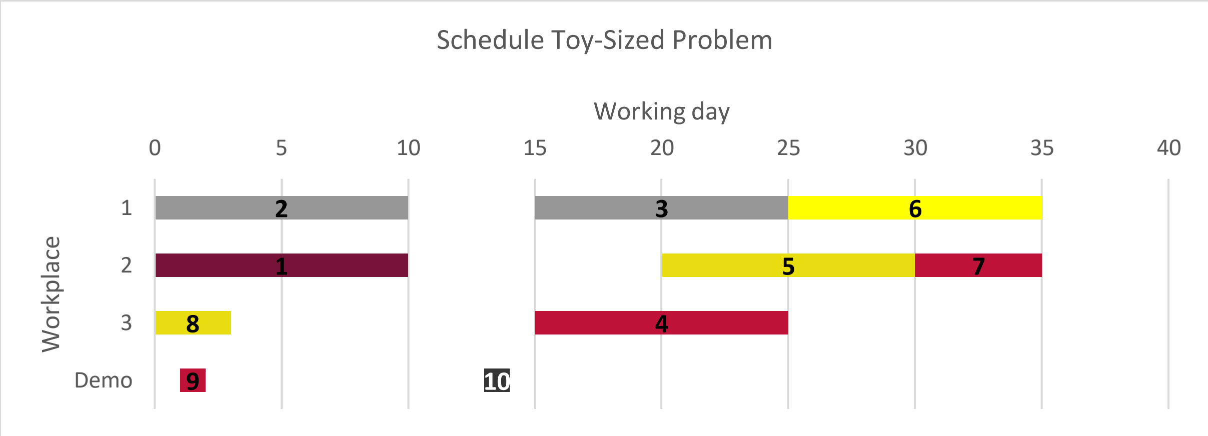

Figure 2.9: Schedule toy-sized problem, activities one and two swapped

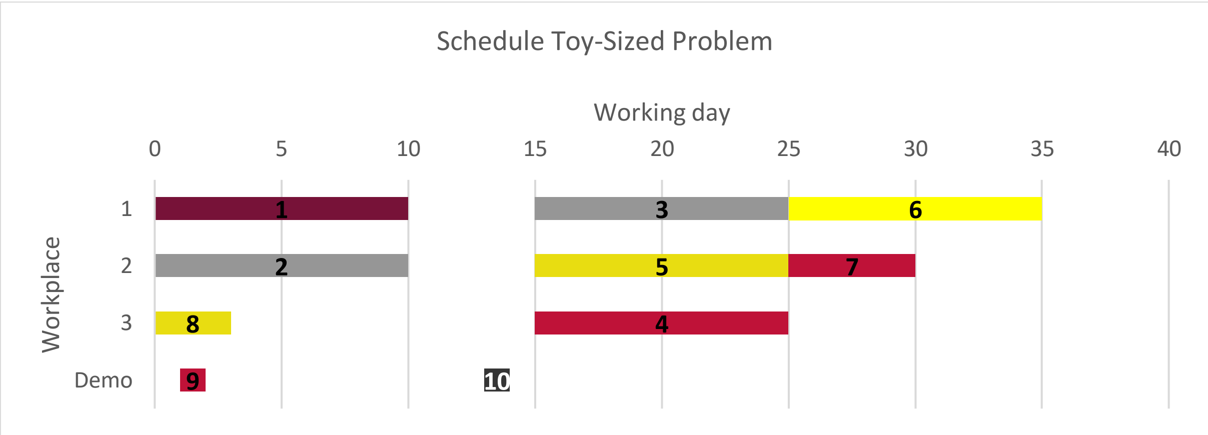

Figure 2.10: Schedule toy-sized problem, activities five and seven started earlier

2.6 Other Integration Labs

PREI is not the only lab of its kind. In the development of advanced aerospace and defence vehicles, the System Integration Lab (SIL) has become a key component. An SIL is a test facility where a complete vehicle (hardware and software) can be integrated, tested and evaluated. The test facility is a combination of simulation, emulation and real components of the final system. Therefore, the risks of testing are reduced. Also, it is proven that the use of an SIL results in cost savings (Applied Dynamics International, Inc., 2007).

One of the first SILs was Boeing's 777 Systems Integration Lab. This lab reached full functionality in June 1993 (Lansdaal & Lewis, 2000). But SILs are not only used in the aerospace and defence industry. Another example can be found in the automotive industry. Adenmark, Deter, & Schulte (2006) describe the use of an integration lab for Scania trucks and busses. Integration labs are also used in the rail industry. There are multiple labs to test the integration of ERTMS. In Denmark, the Joint Test Laboratory is initiated. A similar lab can be found at Multitel in France, and of course at ProRail: PREI. The available data and the possibilities of using the data for this research are considered in the remainder of this section.

2.6.1 Available data

Available data about the processes of the labs in the aerospace, defence and automotive industry is limited, as those labs are mostly privately owned, and competitors can abuse the data. Also, the available information about the testing processes in those labs is less useful in this research, as the testing processes are different from the testing processes at PREI. Therefore, those labs will be disregarded in this research.

The integration labs in the rail industry, and especially the ERTMS labs, are useful in this research. Both the management and testing processes of those labs resemble or are related to the processes of PREI. Some information about those labs is published. An example is the set of possible tests. Every European Infrastructure Manager composes a set of tests. The set needs the approval of the ERA due to European regulations. Once approved, the set is published on their website and available to everyone. The processes of the labs are not published. The labs are independent and have their way of operating. However, there is a shared interest in an optimal rail network. Therefore, a large network is established with different parties that work with ERTMS. This can help with sharing information. The Joint Test Laboratory in Denmark made their testing process for example available to help designing the testing processes PREI.

2.7 Conclusion

ERTMS is the new international standard for train and railway protection. The EU aims for an interoperable railway system in Europe, whereas the main goals in the Netherlands are to increase the capacity and keep up with the technology. PREI is invented to perform tests on the interaction of the different components of the ERTMS systems. Other activities that are executed at the lab are other tests, demonstrations and trainings. To execute these activities, several resources are present.

Different processes can be distinguished in the lab. The processes are divided into *management* and *testing* processes. Both categories are elaborated.

Three complete lines are currently available in the lab on which activities can be performed. The ESC tests are included in the RLN00445 guidelines. When more lines are equipped with ERTMS, these lines will be added to the test environment and the tests to the RLN00445.

The lab has not yet a scheduling method to schedule the activities which require a workplace. At this moment, the agendas of the employees and the workplaces are consulted to find a suitable time and place to perform the activities. When the lab will be more crowded, this can cause problems. A toy-sized problem is introduced to show the problem.

PREI is not the only lab of its kind. Similar labs can be found in the aerospace, defence and automotive industry. Also, there are multiple labs to test the integration of ERTMS. The information about the ERTMS labs is useful in this research, as the processes resemble or are related to the processes of PREI.

Part 1

Process Design

3 Literature Review: Process Design

The first part of the research is about process design. A process can be seen as a series of activities that are needed to achieve a goal. During the activities in the process, input is used to create value-adding output (Theisens, Harborne, & Verreijt, 2018). Processes can be captured in different ways. It depends on the process which method is desirable. This chapter starts with a classification of business processes in Section 3.1. Different methods that can be used for the design and documentation of processes are analysed and described in Section 3.2 and 3.3. In Section 3.4, the most suitable method for every process type at PREI is selected and elaborated. Section 3.5 concludes the chapter.

3.1 Classification of Business Processes

A business process is a series of activities within a company to achieve a goal (Weske, 2012). Business processes can be classified in different ways. Processes can for example be tagged in categories based on their role within an enterprise (von Rosing, Kemp, Hove, & Ross, 2015). Porter (1985) introduces a classification with three types of processes: management processes, core processes and support processes. The three types can be constructed like a house (Figure 3.1). The support processes form the fundament, the core processes the body, and the management processes the roof. According to Weske (2012), business processes can be classified along three so-called 'dimensions': organisational business processes, operational business processes and implemented business processes. Von Rosing, Kemp, Hove and Ross (2015) classify business processes according to the following three categories: management processes, main processes and supporting processes. Although the names of the categories are different, the categories are more or less the same.

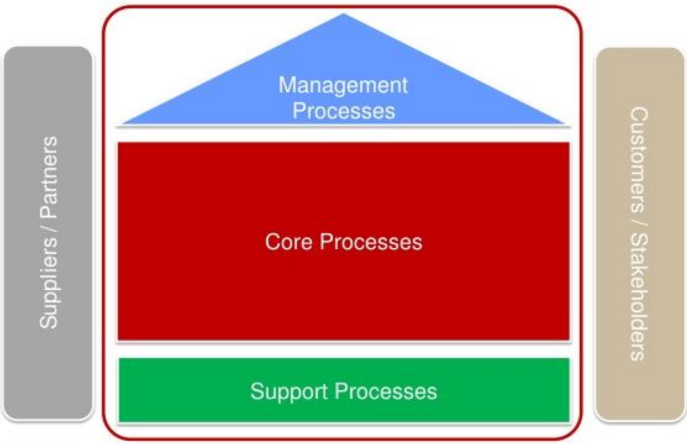

Figure 3.1: Process architecture according to Porter (1985)

In the remainder of this section, the three categories are discussed. This results in a classification of business processes based on their type. The classification can help to determine the best method to design and implement business processes, as there are many methods available.

3.1.1 Organisational Business Processes

The first category covers high-level processes: the organisational business processes. These processes can also be seen as management processes and control the organisation (von Rosing, Kemp, Hove, & Ross, 2015). Inputs, outputs and expected results of the process, and the dependencies on other processes are specified (Weske, 2012). Organisational business processes are commonly expressed informally, often even with a written explanation, because they involve many persons and activities in an organisation. Diagrams and figures can be used to clarify the written text (Weske, 2012). Examples are strategic management, innovation and budgeting. In the remainder of this research, these processes are referred to as management processes, because this is the most widely used term.

3.1.2 Operational Business Processes

Operational business processes can be found in the second category (Weske, 2012). Operational business processes are also referred to as the main or core processes of an organisation (von Rosing, Kemp, Hove, & Ross, 2015; Porter, 1985). They establish the core of the business and deliver the output (Weske, 2012; von Rosing, Kemp, Hove, & Ross, 2015). Value is often created for customers in this category (Aguilar-Savén, 2004). For operational business processes, activities are specified and the relations between the different processes are determined. Operational business processes are typically captured with the help of business process models (Weske, 2012). Manufacturing is a typical operational business process. In the remainder of this research, these processes are referred to as core processes.

3.1.3 Implemented Business Processes

The last category contains implemented business processes. The information and resources needed to execute (management and core) process activities and information on the environment in which the activities are executed are captured with implemented business processes (Weske, 2012; von Rosing, Kemp, Hove, & Ross, 2015). These processes can also be seen as supporting processes, as they support the core processes of an organisation (Aguilar-Savén, 2004; von Rosing, Kemp, Hove, & Ross, 2015), and contain both organisational and technical aspects. Organisational aspects are for example the people in the organisation and their roles and responsibilities. An example of a technical aspect is the use of a system (Weske, 2012). Because the processes vary widely, there are multiple possible methods to capture them (Weske, 2012). It depends on the process and the organisation which method is most suitable. Examples of supporting processes are technical support and human resource management. In the remainder of this research, these processes are referred to as supporting processes.

3.2 Written Explanation

The first method that can be used to design processes is a written explanation, which is an informal way of capturing processes. A process is explained in plain text with this method, as the name already suggests. Although there are no strict rules for written explanations, it is important that they are clear to everyone. Therefore, clear language should be used, such that everyone understands the process in the same way. Figures and diagrams can be used to support the text. Written explanations can be used for different kinds of business processes.

A written explanation is often used to define working guidelines. A working guideline can be used to communicate the process and often contains a list of important tasks that must be done within the process or a stepwise instruction of activities to be performed. Sometimes, a working guideline contains only a goal that should be reached with the process. The level of detail of a working guideline is dependent on the situation. The guidelines should be explained in clear language and can be supported with figures if necessary (Weske, 2012).

Written explanations are also often used to define stakeholders and their roles and responsibilities (Weske, 2012). Lists can be made to indicate the roles and responsibilities of all stakeholders, such that it is clear immediately. In Figure 3.2, a written explanation of the roles and responsibilities of the ESC Test Facility Manager can be found. This is an example from the process elaboration of ESC in European Union Agency for Railways (2020).

4.3 Roles and responsibilities of the ESC Test Facility Manager

The ESC Test Facility Manager shall

- be responsible for managing the ESC Test Facility;
- provide the ESC Test Facility for ESC Tests;
- ensure in case of lab tests the integration of the OBU Test Bench with the ESC Test Facility and coordinate the provision corresponding to the technical concept described in the Subsets (e.g. Subset 111, solution for communication link as determined by the OBU);
- set up a generic database of ESC Tests for execution in the ESC Test Facility with relation to the activities according to Section 5;
- operate the ESC Test Facility during the activities according to Section 5 and ensure a smooth running of the Test Campaign;
- ensure the maintenance/update of the ESC Test Facility.

Figure 3.2: Example roles and responsibilities ESC Test Facility Manager (European Union Agency for Railways, 2020)

The forms-based approach is described in the remainder of this section. The forms-based approach is a (written explanation) method that can be used to design management processes. Also, the process landscape diagram is described, which can be used to define the relations between processes.

3.2.1 Forms-Based Approach

The forms-based approach is commonly used as a written explanation method for management processes. An example of the use of this approach can be seen in Figure 3.3. Individual activities and the arrangement of the activities are not addressed. The process can be seen as a black box at this level, as the details of the process itself are not important. This results in a process description without strict execution constraints (Weske, 2012).

Process Name: Product Development Process	**Responsible Process Manager:** Dr. Myers
From: Requirements **To:** Rollout	**Type:** Development Project
Process Inputs: Requirements Document, Product specification, Budget Plan, Prototypes	**Supplier Processes:** Product Planning Process, Innovation Process
Process Results: Integrated and completely tested innovative product with complete documentation	**Customer Processes:** Order Management Process, After-Sales Service Process

Figure 3.3: Example of the Forms-Based Approach (Weske, 2012, p. 46)

Relations between different processes can be visualised with a process landscape diagram. Figure 3.4 shows an example of a process landscape diagram for a manufacturing organisation. Processes are depicted with blocks and relations with arrows. Processes can have relations based on the transfer of information or products. It is important that the processes are designed carefully, as unclear processes can be a source of inefficiency (Weske, 2012).

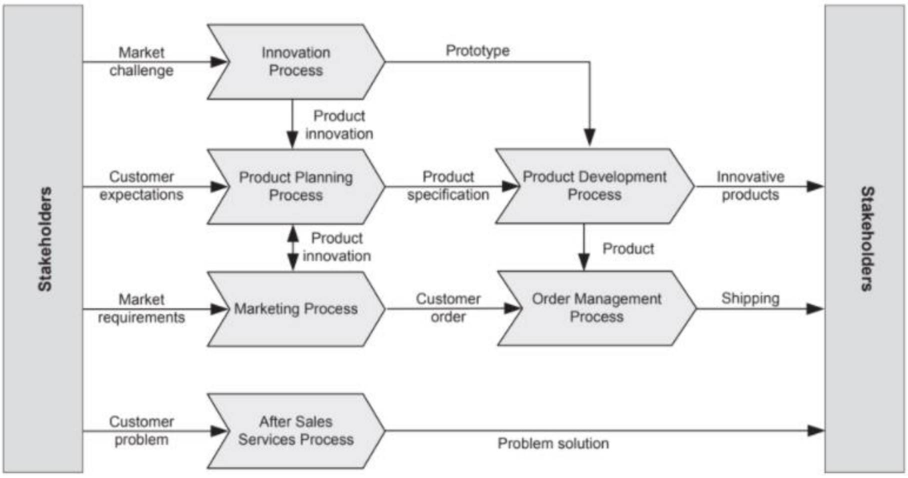

Figure 3.4: Example of a Process Landscape Diagram (Weske, 2012, p. 47)

3.3 Business Process Modelling

Business process modelling is another method that can be used to design processes. This method can help understand and analyse a business process (Aguilar-Savén, 2004). A business process model can be seen as a scheme for a set of business processes. Also, the relations between the processes are indicated and visualised (Weske, 2012). One or more inputs are used by the activities in the process to create one or more valuable outputs (Theisens, Harborne, & Verreijt, 2018). A business process model captures the process that is represented by the block in Figure 3.5.

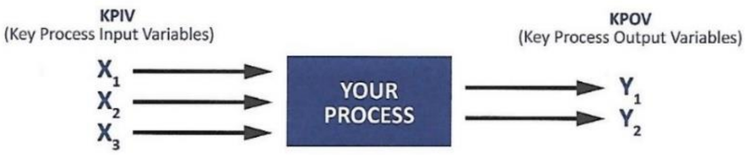

Figure 3.5: Process model definition (Theisens, Harborne, & Verreijt, 2018, p. 131)

There are different methods available to model processes (Theisens, Harborne, & Verreijt, 2018). The most suitable method that should be used depends on the situation in which the model is needed (Aguilar-Savén, 2004). In the remainder of this section, commonly used process modelling methods are described.

3.3.1 Flowcharts

One of the most frequently used process modelling method is the flowchart or process map. With this method, the flow of a process is visualised instead of described with text. Flowcharts are also commonly used in Six Sigma and Lean projects (Theisens, Harborne, & Verreijt, 2018).

In a flowchart, all steps of the process are depicted in a chart and the relations are indicated (Theisens, Harborne, & Verreijt, 2018). Flowcharts are commonly used to describe the logic or path of execution of a process (Dufresne & Martin, 2003). Therefore, they can be used to communicate processes, describe and understand processes and define responsibilities and competencies within processes (Theisens, Harborne, & Verreijt, 2018).

The ability of communication can be seen as the greatest strength of this method, as a flowchart model is easy to use. The needed time to design a process is relatively short. The main characteristic of flowcharts is their flexibility. Although a standard notation (see Section 3.3.1.1) is used, the way the activities are connected is flexible. This can be an advantage of the method, but is also seen as the biggest weakness: it can be hard to read a flowchart, as the design depends on the designer. Also, flowcharts can become very big, which can make it even harder to read (Aguilar-Savén, 2004).

Flowcharts are most suitable to deal with processes with a high level of detail. Flowcharts were not suitable to give an overview, as it was hard to describe responsibilities (Aguilar-Savén, 2004). However, the technique evolved. Responsibilities can namely be visualised with so-called swimlanes. Every department or person gets his or her horizontal lane. Different phases of a process can be represented by vertical lanes (Theisens, Harborne, & Verreijt, 2018). An example of a swimlane flowchart can be seen in Figure 3.6.

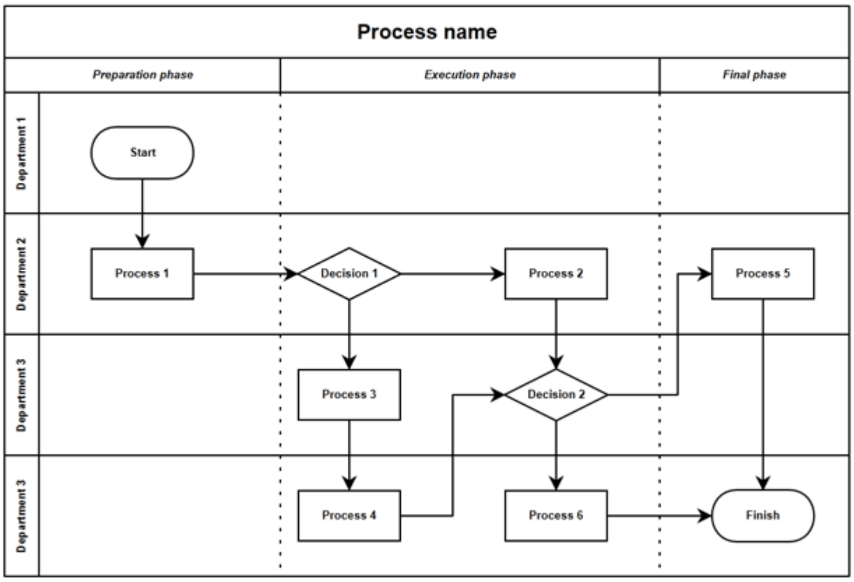

Figure 3.6: Flowchart example with swimlanes (Grapholite, 2020)

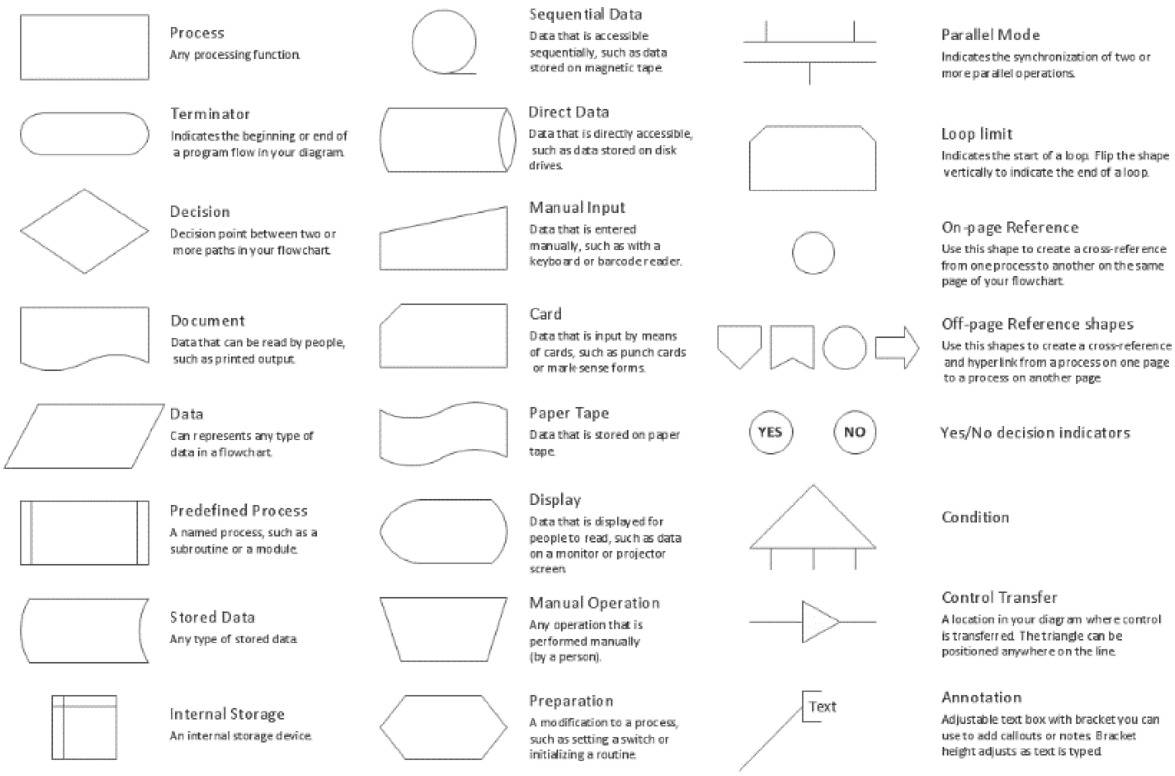

Figure 3.7: Flowchart symbols with explanation (ConceptDraw, 2020)

3.3.1.1 Notation

Different symbols are present in the flowchart in Figure 3.6. The International Organization for Standardization (ISO) incorporated the symbols used in flowcharts in their ISO 5807:1985 standard. This standard was established in 1985 but is last reviewed and confirmed in 2019. ISO 5807:1985 contains standards about information processing, which includes 'Documentation symbols and conventions for data, program and system flowcharts, program network charts and system resources charts' (International Organization for Standardization, 1985). The flowchart symbols described in the ISO standard are depicted and explained in Figure 3.7.

3.3.2 Data Flow Diagram

Another method of business process modelling is the data flow diagram. Data flow diagrams are used to show data or information flow in an information system (Dufresne & Martin, 2003). Material flow is not included. The linkage of the processes is described based on data storage and the relations to users and the world (Aguilar-Savén, 2004). An example of a data flow diagram can be seen in Figure 3.8.

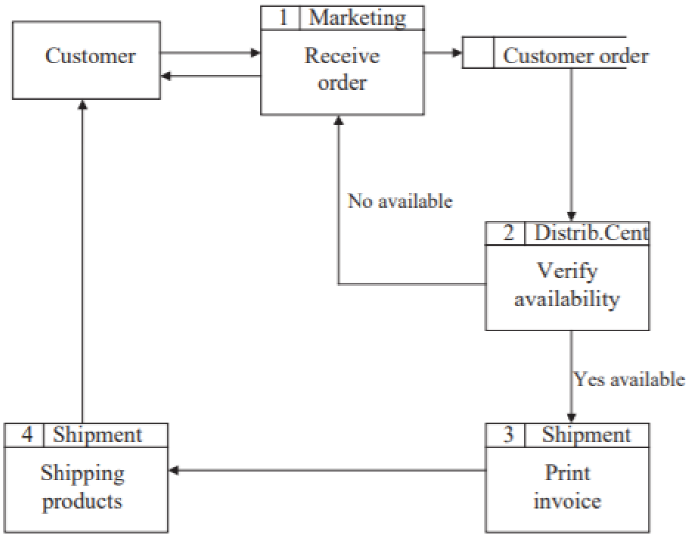

Figure 3.8: Data Flow Diagram example (Aguilar-Savén, 2004)

A data flow diagram can help to compose a process at a logical level: it is described what the process does, not how the process should be executed. This gives a good understanding of the process and helps with discussions between analysts and users. Also, it is clear how information enters the process, which activities change the information, and how the information leaves the process (Aguilar-Savén, 2004).

3.3.3 Role Activity Diagram

Another method is the role activity diagram. They are used to describe a process based on the perspective of individual roles. A role activity diagram concentrates on the responsibilities of and the interaction between the individuals. Software systems can also be included in the diagram (Aguilar-Savén, 2004). Figure 3.9 shows an example of a role activity diagram.

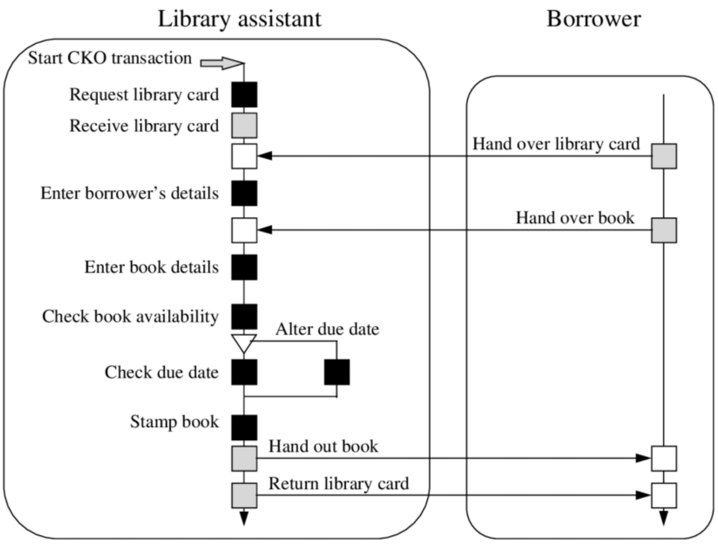

Figure 3.9: Role Activity Diagram example (Liu, Alderson, & Qureshi, 1999)

Role activity diagrams are object state transition diagrams, which means that they describe the way an object changes state. They are especially useful for communication purpose. A process can be shown in detail and is easy to read and understand. Also, parallel activities and the interaction between software systems can be described. A disadvantage of this method is that business objects cannot be included in the diagram. Examples of business objects are machines and products (Aguilar-Savén, 2004).

3.3.4 Role Interaction Diagram

Role interaction diagrams are a combination of role activity diagrams and object interaction diagrams. An example of a role interaction diagram can be seen in Figure 3.10. The roles can be found on the x-axis on top, the activities can be found on the y-axis on the left. The horizontal arrows between the roles show human interactions (Aguilar-Savén, 2004). Role interaction diagrams are relatively easy to read. However, they become messy when a lot of interactions are present in the process, as this results in a lot of arrows in the diagram. This makes it hard to compose or adjust this type of diagram. Another weakness is that inputs and outputs of activities cannot be added. This results in a loss of information (Aguilar-Savén, 2004).

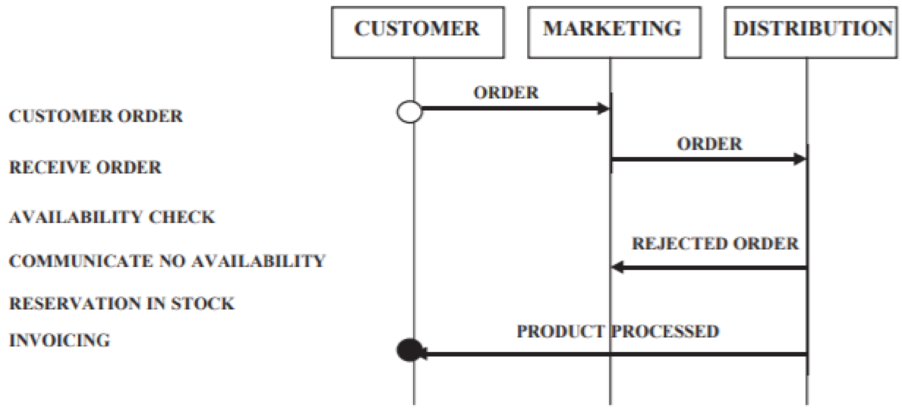

Figure 3.10: Role Interaction Diagram example (Aguilar-Savén, 2004)

3.3.5 Gantt Chart

Gantt charts are widely used in project management, but they can also be used for business process modelling (Dufresne & Martin, 2003). An example of a Gantt chart can be found in Figure 3.11. On the y-axis on the left in the figure, the activities are listed. The x-axis on top includes a time scale. The time scale can be in any time unit, such as hours, days or weeks, or periods. With a Gantt chart, the relation between activities and a given time period can be visualised. Analysis of the process itself is therefore hard. Also, relations between activities are hard to determine (Aguilar-Savén, 2004).

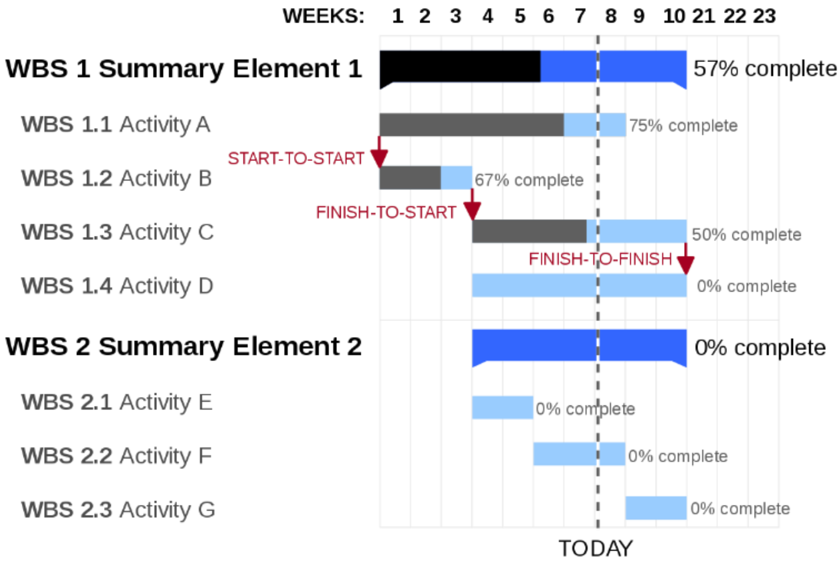

Figure 3.11: Gantt Chart example (Wikipedia, 2020)

3.3.6 Integrated Definition for Function Modelling

The Integrated Definition for Function Modelling (IDEF) methods come in different versions, with different applications. The most useful versions for business process modelling are IDEF0 and IDEF3, where IDEF0 is the most popular (Aguilar-Savén, 2004). Therefore, IDEF0 is elaborated in this review. An example of an IDEF0 diagram can be found in Figure 3.12.

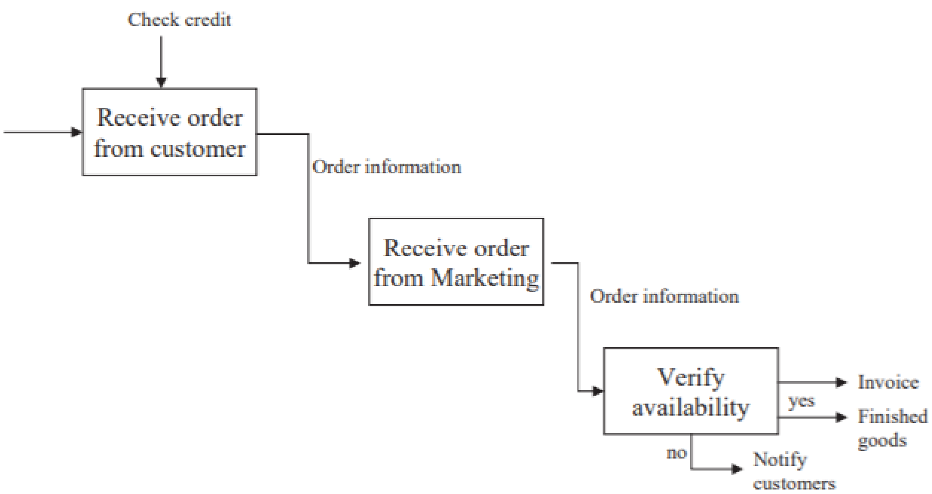

Figure 3.12: Integrated Definition for Function Modelling example (Aguilar-Savén, 2004)

IDEF0 is used to define function models (Aguilar-Savén, 2004). Function models are structural representations of the functions within a process or system. Functions are activities, actions, processes or operations (IGI Global, 2020). In the model, high-level activities are shown with their inputs, outputs and controls (Dufresne & Martin, 2003). Also, mechanisms that are associated with an activity can be shown (Aguilar-Savén, 2004).

IDEF0 is a popular method because strict rules make it possible to implement the models as computer software. Also, a lot of data and control can be defined in the model, which makes it possible to analyse and improve the process if necessary. However, the model does not represent a sequence of activities, which is the biggest weakness of this method. It is possible to, without intent, embed sequencing in the model by ordering the activities from left to right (Aguilar-Savén, 2004).

3.3.7 Coloured Petri Nets

Coloured Petri nets is a method for the design, specification, simulation and verification of systems, and is most suitable for systems that contain multiple processes. Especially when those different processes communicate with each other and should be synchronised (Aguilar-Savén, 2004). An example of a coloured Petri net can be found in Figure 3.13.

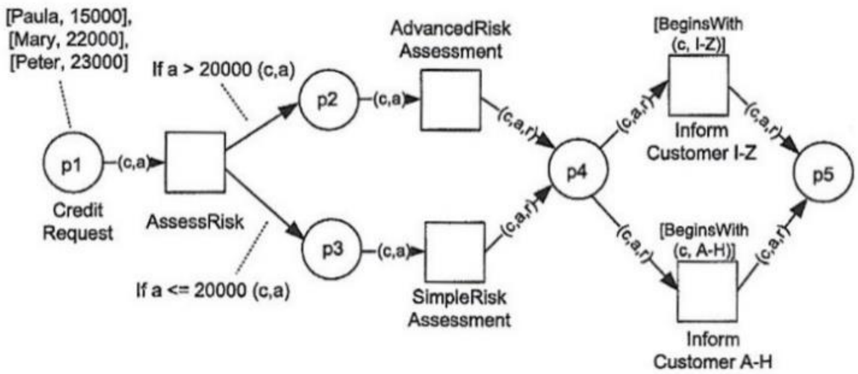

Figure 3.13: Coloured Petri Nets example (Weske, 2012, p. 158)

Dynamic systems with a static structure can be modelled with Petri nets (Weske, 2012). Petri nets consist of places, transitions and arcs to connect the places and transitions (Aguilar-Savén, 2004). In Figure 3.13, the circles represent places and the squares represent transitions. The dynamics of the system are modelled with tokens. Those tokens can be located on places and can change their positions based on rules. The distribution of the tokens determines the state of the system. Coloured Petri nets are an extension of Petri nets. The colours of a coloured Petri net allow tokens to have values. Therefore, tokens can be identified and the process can be modelled in more detail (Weske, 2012).

The first advantage of this method is that the model helps to understand how processes interact with each other. Also, the syntax is well defined and the model represents mathematical models. However, this is also one of the weaknesses, as the modelling is time-consuming and the model may not be understandable for everyone (Aguilar-Savén, 2004).

3.3.8 Unified Modelling Language

Unified Modelling Language (UML) use object-oriented methods for modelling. UML itself is a language that is used to specify, visualise, construct and document artefacts of software systems and non-software systems such as business models. Therefore, there is consistency across the design, analysis and programming of processes (Aguilar-Savén, 2004).

The UML represents nine types of models, and each model has its purpose (Aguilar-Savén, 2004; Dufresne & Martin, 2003). Of all types of models, the UML sequence diagram (Figure 3.14) is most suitable to design processes. However, the biggest weakness is the lack of a converging construction (Dufresne & Martin, 2003). Also, modelling with the use of UML can be complex and very time-consuming (Aguilar-Savén, 2004)

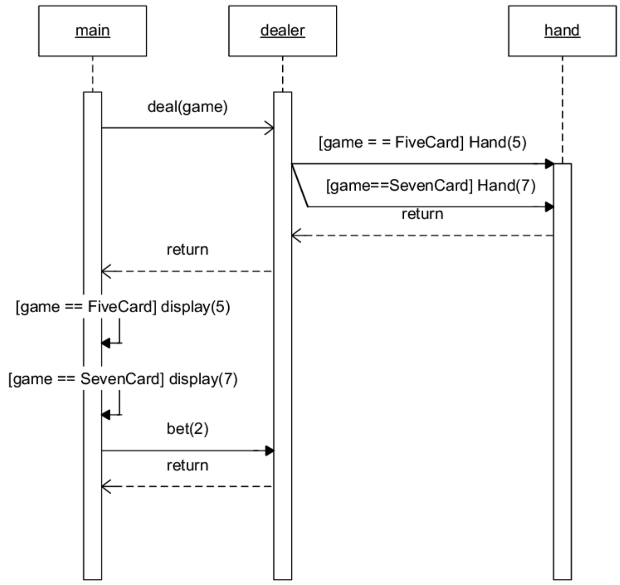

Figure 3.14: Example of a Unified Modelling Language Sequence Diagram (Pilskalns, Andrews, Ghosh, & France, 2003)

3.3.9 Business Process Model and Notation

A relatively new technique for business process modelling is the Business Process Model and Notation (BPMN). BPMN can be used to design an executable business process (Dufresne & Martin, 2003). The aim of BPMN is comparable with the aim of UML: to combine the best practices of existing methods. This results in different types of diagrams that can model multiple levels of abstraction, from organisational level to implementation level (Weske, 2012). A simple example of a BPMN model can be found in Figure 3.15.

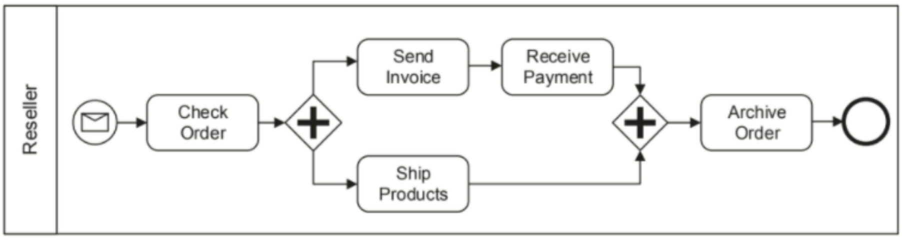

Figure 3.15: Business Process Model and Notation example (Weske, 2012, p. 7)

The strengths and weaknesses of BPMN are more or less the same as the strengths and weaknesses of UML. The focus of BPMN is, however, on the modelling of business processes, whereas the focus of UML is more on the modelling of software.

3.3.10 Summary

In the previous sections, nine business process modelling methods are described. All methods are summarised in Table 3.1. The table includes the attributes of the methods, and the characteristics, strengths and weaknesses. This summary can help to select the most suitable method when modelling a process.

A framework to classify the described and some other methods based on their characteristics, strengths and weaknesses is proposed by Aguilar-Savén (2004). The framework is depicted in Figure 3.16. In this framework, the methods are classified according to two dimensions: the purpose of the model and the model change permissiveness (to what extent are changes allowed). The framework aims to help to decide which method is most suitable to use in specific cases, as the best method depends on the situation.

Table 3.1: Overview of Business Process Modelling methods

Method	Attributes	Characteristics	Strengths	Weaknesses
Flowchart	Actions flow	Flexible No sub-layers High level of details	Ability of communication Easy to compose and use	Can become large Can be hard to read
Data Flow Diagram	Data or information flow	Description of what a process does (logical level)	Easy to understand Easy to compose	Only data or information flow, no material flow
Role Activity Diagram	(Individual) roles flow	High level of details Interactions can be displayed No overview	Ability of communication Easy to understand	Business objects (e.g. products) cannot be included
Role Interaction Diagram	Roles and activities flow	Roles are included Inputs and outputs are not included	Easy to read Can compose complex processes	Can become messy Hard to compose or adapt Important information (e.g. input) is missing
Gantt Chart	Activities and durations flow	Relations between activities and time can be displayed	Easy to overview Easy to compose	Hard to analyse a process Hard to determine relations between activities
Integrated Definition for Function Modelling (IDEF0)	Activities, in- and outputs, control and mechanisms flow	Sub-layers Strict rules	Can be implemented as software Data and control can be defined	No sequence of activities Roles are not included
Coloured Petri Net	Places and transitions network, connected with arcs	Extension of Petri nets Colours differentiate tokens	Interaction is displayed Well defined syntax Mathematical models	Hard to compose (time-consuming) Can be hard to understand
Unified Modelling Language	Structure and behaviour of objects	Object-oriented Different types of models with their purpose Focus on modelling of software	Consistency across design, analysis and programming Can be implemented as software	Lack of converging construction Hard to compose (time-consuming)
Business Process Model and Notation	Structure and behaviour of objects	Object-oriented Different types of models with their purpose Focus on modelling of business processes	Consistency across design, analysis and programming Can be implemented as software	Lack of converging construction Hard to compose (time-consuming)

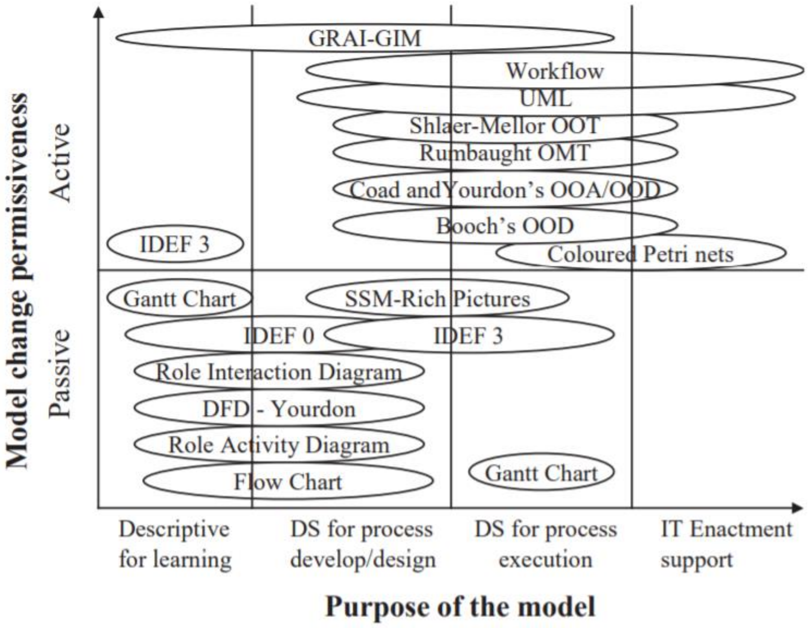

Figure 3.16: Classification framework (Aguilar-Savén, 2004)

The purpose of the model can be found on the x-axis in the framework and is divided into four categories. The first category covers 'descriptive models for learning' (Aguilar-Savén, 2004, p. 146). Those models can be used to describe a process and to learn about a process. The second category covers 'descriptive and analytical models for decision support to process development and design' (Aguilar-Savén, 2004, p. 146). These models can help to design or develop a process. The purpose of those models is to analyse processes. The third category covers 'enactable or analytical models for decision support during process execution, and control' (Aguilar-Savén, 2004, p. 146). These models help with decision making during the performance of a process. This can be done by controlling and monitoring a process or providing the decision-maker with the right information. The last category covers 'enactment support models to Information Technology' (Aguilar-Savén, 2004, p. 146). The purpose of these models is to support software development processes.

On the y-axis of the framework in Figure 3.16, the model change permissiveness can be found. This dimension is divided into two categories: passive and active. Passive models are models that cannot be changed without remodelling the process. Also, user interaction is not possible. These models are static. An example is a printed overview of a process. Active models allow user interaction or are even dynamic themselves. A simulation model is an example of an active model.

The described models are placed in the framework based on their characteristics, strengths and weaknesses. This classification can help to select the most suitable method for a certain situation. However, Aguilar-Savén (2004) states that further research on the framework is necessary to classify the methods on other criteria such as usability and experiences. Therefore, an important remark is that the framework can support the decision of which method is suitable, but should not be used on its own, as not all important characteristics, strengths and weaknesses are included. The framework can be used to narrow the selection of possible methods. The descriptions of the methods (or at least the summary in Table 3.1) should be consulted when the final decision is made.

Another remark about the framework is that the BPMN method is missing. The BPMN can be positioned near the UML and other object-oriented methods.

3.4 Suitable Methods

PREI can be seen as a small business, although it is a part of the company ProRail. To keep the 'business' running, it is important that all different types of processes are designed.

Many different business processes can be determined in the lab. These business processes can be classified with the use of the classification given in Section 3.1. The most suitable method depends mainly on the process type and characteristics. This section gives an overview of the different process types that are present in the lab and for all types. Suitable methods are determined based on the classification of the processes, and the descriptions and the proposed framework of the methods in Sections 3.2 and 3.3.

3.4.1 Management Processes

Most management processes can only be done by a person in real. The processes do not have to be designed with the purpose to be converted to software. Also, once the processes are captured, they do not need to be continuously adjusted. A passive and static method is, therefore, suitable for the design of the management processes.

It is important that everyone understands the processes, but the processes do not have to be elaborated in detail. Together with the other important aspects of the design, this results in the selection of written explanations as the most suitable method. With this method, the processes can be described in as much detail as necessary, and if clear language is used, everyone can understand the processes.

The forms-based approach can help to structure the process designs. A drawback of this method is that the process is seen as a black box: the details are not important. For some management processes, details are needed so everyone can understand them. The forms-based approach is used in this research, but the form will be set up in such a way that details can be added if necessary. A process landscape diagram can be used to indicate the relations between different management processes.

A written explanation can be composed in word-processing software, such as Microsoft Word. Microsoft Visio can be used to compose a process landscape diagram or other diagrams and figures if needed.

3.4.2 Core Processes

The main goal of the lab is to perform tests on the integration of ERTMS. The most important core process within the lab is therefore the testing process. This process does add the most value to the lab, for ProRail and the customers and should be captured in the right way. Other core processes that add value to the lab are giving demonstrations and trainings. These processes do not differ much from the testing process. The main activity of all processes is to execute or demonstrate activities with ERTMS equipment on one of the workplaces. Therefore, the same, suitable method is selected for all those processes. It is chosen to select a business process modelling approach because Weske (2012) and Aguilar-Savén (2004) both state that business process modelling is a useful technique to capture core processes.

The main purpose of the method should be that the process can be described and captured clearly because it is important that everyone understands the process. Also, the responsibilities should be clear and the model should ease communication between different parties. The designs of the processes can also help to analyse and optimise the processes in the future. The model does not have to be actively changed regularly. Once the process model is composed and the process is explained correctly, the model does not have to be changed until the process is changed. Therefore, a static model is sufficient in this case.

When looking at the dimensions of the framework as suggested by Aguilar-Savén (2004), the most important purpose of the model is 'descriptive for learning'. 'DS for process develop/design' is a nice to have for the future. Also, a passive model change permissiveness is sufficient. Therefore, the most suitable methods, based on these two dimensions, are positioned in the left lower quadrant of Figure 3.16. These methods are flowchart, data flow diagram, role activity diagram, role interaction diagram, Gantt chart and Integrated definition for function modelling (IDEF0).

The models of the core processes should meet several aspects:

- The main goal of the model is that everyone should *understand the process*.
- The responsibilities should be clear and the model should ease communication between different parties. These requirements can be met when the *sequence of activities* is visualised clearly, and the *responsibilities* are indicated.
- The model should focus on the *flow of activities and actions*, not on the flow and handling of data or information.
- The model should be able to *model inputs and outputs*, as different inputs and outputs, such as test results, are important aspects of the processes.

The methods selected above are analysed based on the important aspects. Table 3.2 contains the results of the analysis. We score the six remaining methods on the five most important aspects. For most aspects, a method either complies with the aspects or not. Small adjustments or additions can also ensure that a method is satisfactory. Therefore, three possible scores are used. A checkmark is used when the method does comply with the aspect, a dash is used when the method can comply with the aspect after an adjustment or addition, a cross means that the method does not comply with the aspect.

Table 3.2: Analysis of possible methods

Method	Under-standing	Sequence of activities	Respon-sibilities	Activities or actions flow	Inputs and outputs
Flowchart	✓	✓	—	✓	✓
Data Flow Diagram	✓	✓	✗	✗	✓
Role Activity Diagram	—	✓	✓	✓	—
Role Interaction Diagram	—	✗	✓	✓	✗
Gantt Chart	✗	—	✗	✓	✗
Integrated Definition for Function Modelling (IDEF0)	✓	—	✗	✓	✓

The comparison of the methods shows that the most suitable method to design the core processes is the flowchart method. Responsibilities can be indicated with the flowchart method when swimlanes are added. An important remark is that the models created with this method can become large and hard to read. During the design phase, this has to be taken into account. The models should not become too large, so the end-users can understand the models.

Flowcharts are also already used within ProRail, which makes it easier for the users to understand the notation and the working of the models.

There are a lot of tools available to make flowcharts. Within ProRail, Microsoft software is frequently used. Microsoft Visio is available to compose the flowcharts, so we choose to use this software. This will probably result in a better understanding and acceptance within ProRail. Also, no new software has to be purchased and it is easier for ProRail to adapt the flowcharts in the future when already available software is used.

3.4.3 Supporting Processes

For the supporting processes, broadly the same applies as for the management processes. Different supporting processes can be distinguished, and many of them should be done by a person in real. The most suitable method to capture the processes depends on the characteristics of the process and should be selected per process. However, it is possible to describe the processes with a written explanation, possibly with the support of figures and tables.

The written explanation method is chosen as the most suitable method for this research to capture the supporting processes. Figures can help to support the text. In that way, the process is likely to be understood by everyone. Also, a written explanation is an easy method to use. An important remark is that for some processes another method can be added to the written explanation to clarify the process. For example, when a flow of data or materials is present, a written explanation can be supported by other methods, such as a flowchart to visualise the flow of activities.

It is not necessary to compose a process landscape diagram at this level. The processes at this level do not need to have relations, as they support the core (and management) processes. However, it can be that some processes do have a relation. In that case, it is important to indicate what this relation is.

A written explanation can be composed in word-processing software, such as Microsoft Word. Microsoft Visio can be used for diagrams and figures if needed.

3.5 Conclusion

Business processes can be classified into three categories. Each category covers a set of processes on a certain level.

- Management processes: This category covers the highest-level processes. These processes characterise unpolished functionality.
- Core processes: These processes are the core of the organisation and create the output.
- Supporting processes: These processes describe all information needed to execute process activities.

Processes can be designed with many different methods. The first described method is the written explanation. A written explanation explains the process in plain text and can be supported by figures. The form-based approach can help to structure the design. A process landscape diagram can be used to indicate the relations between processes.

Another method is business process modelling. A business process model is a scheme for a set of processes and can indicate relations. There are different approaches within business process modelling, all with their purpose and characteristics. Nine often mentioned and widely used methods are discussed and summarised in Table 3.1. Aguilar-Savén (2004) proposed a framework to classify the methods along two dimensions: the purpose of the model and model change permissiveness. The most suitable method depends on the process and situation. For all three process types, the most suitable method is selected.

- **Management processes**

For the management processes, a passive and static method suffices. For most processes, no strict activities, sequences or flows have to be distinguished. At last, it is important that everyone can understand the processes. The written explanation is chosen as the most suitable method. The forms-based approach can help to structure the designs, a process landscape diagram to show relations between the processes.

- **Core processes**

Core processes should be *understood by everyone* and *responsibilities* should be clear. The model should ease communication, focus on *activities or actions flow*, indicate the *activity sequence* and it should be possible to model *inputs and outputs*. A static method suffices. Based on the classification proposed by Aguilar-Savén (2004), six methods remain. These methods are scored on the five most important aspects. From the analysis, it becomes clear that the flowchart method with swimlanes is the most suitable method for this research.

- **Supporting processes**

Many different supporting processes can be distinguished. The most suitable method depends strongly on the characteristics of the process, but all processes can at least be described with a written explanation. Therefore, this method is selected as the most suitable method for supporting processes. Figures and tables can support the text.

4 Process Design

In this chapter, the processes of PREI are elaborated and designed based on the literature described in Chapter 3. We found in the literature that the processes can be categorised based on three types: management processes, core processes and supporting processes. For these three types of processes, a list is composed of relevant processes within PREI. We determined the processes based on literature and in consultation with the stakeholders.

Section 4.1 starts with management processes, where the list of relevant processes is elaborated. Thereafter, in Section 4.2, the core processes are determined and elaborated. The last process type includes supporting processes. This process type is discussed in Section 4.3. Section 4.4 discusses the storage of the processes. The processes are validated by the stakeholders, which is described in Section 4.5. The chapter is concluded in Section 4.6.

4.1 Management Processes

The first process type that is going to be discussed is the type *management processes*. From the literature, we know that the management processes control the organisation. Three groups of management processes can be distinguished at PREI: *Lab Management*, *Development Management* and *Corporate Focus*. These groups include several processes that are important for PREI. In this section, we first indicate how the processes can be captured. However, this is not done for every individual process due to time reasons, and the problem owners stated that the management processes are not the most important processes at this point to design. Thereafter, all distinguished processes are given and we elaborate on why these processes are important (for PREI).

4.1.1 Proposed Form

We propose a form to capture the management processes. This form is based on the forms-based approach as described in Section 3.2.1. The form is composed in Dutch and English. The English version can be found in Table 4.1. Five parts can be distinguished in the form:

- The first part of the form includes the process group, process name, when the form is last modified and a description.
- Second, the process manager and their responsibilities can be captured. The RACI model can help with the identification of the roles and responsibilities. RACI stands for Responsible, Accountable, Consulted and Informed. For every process or activity, the role of all stakeholders can be indicated (Haughey, 2021):
    - Responsible: The stakeholder who does the work or gets the work done. The decisions are also made by this stakeholder.
    - Accountable: The stakeholder who is accountable for the correct completion of the process or activity.
    - Consulted: The stakeholder(s) who provide information.
    - Informed: The stakeholder(s) who should be kept informed. These stakeholders need to be up to date because the outcomes are important for their roles.
- The input, and output and results of the process follow.
- Then, relations with other processes can be indicated.
- Details (or figures) can be added in the last part if necessary.

Table 4.1: Form to capture processes, based on the forms-based approach

Process group	
Process name	
Last modified	
Description	

Process manager	
Responsibilities	

Input	
Output and results	

Relations with other processes	

Details	

4.1.2 Lab Management

The first group within the management processes is the process group *lab management*. This group includes most of the management processes and can be seen as the most important group for PREI since PREI is the lab. The processes in this group are related to the direct management. We distinguished nine processes that fit into this group. In the remainder of this section, the nine processes are discussed.

- **Annual Planning**

An annual planning is a planning that indicates the goals and objective within a specified year, and it is recommended to include a more detailed plan of activities as well. In this plan, it is documented which activities will be accomplished, who is responsible for the activities, when the activities are planned to be finished and which resources are necessary (ifex, 2021).

An annual planning for PREI is relatively hard to compose. Most activities within the lab are not known that far in advance and should be completed on shorter notice. However, it is still useful to think about and compose an annual planning, based on the strategy of the lab. The planning can for example include tasks related to the test capacity, or activities such as updating information, principles or processes to meet strategic goals.

The process annual planning is captured with the proposed form and can be found in Appendix A.

- **Weekly Meeting and Planning**

As already stated, most activities within the lab are not known far in advance and should be completed on relatively short notice. Therefore, it is useful to have weekly meetings. A weekly planning follows from this meeting.

Such a meeting does not add much value when nothing has changed or when no tasks or activities are released. This meeting can therefore be held on an event basis. This means that the meeting only has to take place when something has to be discussed. Because it is still quite at PREI, the meeting will often not be useful at this point. However, the workload is expected to increase, which means that a meeting will probably be useful every week.

Different affairs can be discussed, such as new activities that are to be executed, the planning of the resources and short-term maintenance. It is recommended to schedule the meeting on a fixed time in the week, e.g. every Wednesday at 10:00, and to plan a GO/NOGO deadline, e.g. every Tuesday at 10:00. Appoint one employee, e.g. planner, to decide whether a meeting is useful (an event occurred) or not. It is important that the other employees send information and documents to the, in this example, planner before the GO/NOGO deadline, so the planner can base the GO/NOGO decision on the received information.

- **Resource Planning**

Allocating resources and the allocation of tasks to these resources is part of the resource planning process. The resources can be both human and non-human. Important aspects of this process are resource utilisation and resource capacity (Meier, 2020). It is important for PREI to keep track of all resources.

It is important to ensure that enough human resources are available to take full advantage of the other resources. Next to that, it is important to allocate or schedule the human resources in an efficient way, based on their competencies, so the activities at PREI can be executed according to the schedule.

The non-human resources are always available for PREI, provided that they are functional and not in maintenance for example. However, some resources may get outdated. We discussed already that it is the responsibility of PREI that the test environment is always up to date. Therefore, some resources may never get outdated. This is a very important aspect of this process. It is also important that the capacity and the use of these resources are monitored and managed. When for example a type of resources causes a bottleneck (e.g. activities that need Post21 Functionality cannot be executed on time, because it can only be used for one activity at the same time), this should be noted, so actions can be taken.

- **Financial Management**

Financial management is the process of planning, organising, controlling and monitoring financial resources and activities. Financial activities include procurement and the utilisation of funds. This is done with the organisational goals and objectives in mind (Juneja, 2021).

The main activity at PREI within this process is invoicing the activities to the customers. To be able to invoice the costs, it is important to know what costs are made and how these costs are built up. Then, it can be determined how much the activities have cost, and can the costs be invoiced.

Another important aspect of this process is funding. Funding is needed for different kinds of expenses, such as maintenance. Also, funding is needed when new resources are procured, e.g. when a new line is added to the lab. With this process, it should be ensured that the needed funding is available or made available.

- **Process Management**

Process management is an important process for organisations and contains multiple aspects. The first aspect is that the processes of the organisation are aligned with the strategic goals. Secondly, the processes should be designed and implemented. Also, process management helps to manage the processes effectively (Appian, 2021). Another aspect is that processes can be improved to become more efficient (Aguilar-Savén, 2004).

Within PREI, it is especially important that the core processes are managed well, as external stakeholders are involved in these processes. However, this does not mean that other processes do not have to be managed.

Process management also includes continuous monitoring of the processes and checking whether the processes are still up to date. When this is not the case (anymore), the processes should be updated. ProRail's quality management system (KMS) supports this aspect. In the future, process management can help to improve the processes at PREI to become more efficient.

- **Risk Management**

With risk management, (potential) risks are identified, analysed, evaluated and treated. Things that can go wrong are identified, stopped from going wrong, the consequences when they go wrong are reduced and things are recovered when they went wrong (Slack, Brandon-Jones, & Johnston, 2013). In every organisation, things can go wrong. It is important to think about the risks to be able to prevent big problems. Therefore, risk management is also included in the list of management processes. A potential risk at PREI can be that the lab is understaffed.

The ISO 31000 standard provides guidelines for risk management (International Organization for Standardization, 2018a). The process of risk management is shown in Figure 4.1. Different steps can be distinguished.

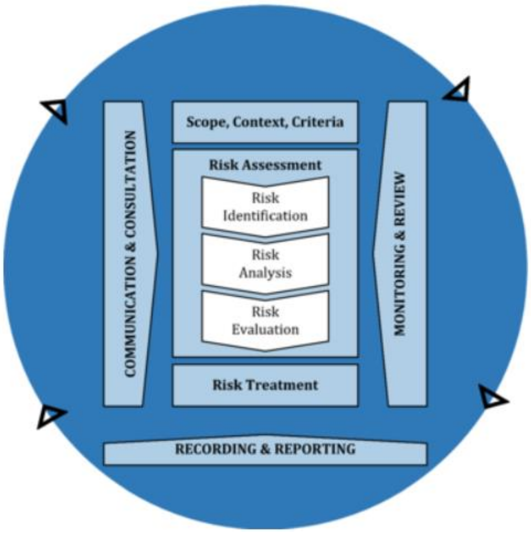

Figure 4.1: Risk Management process (International Organization for Standardization, 2018a)

A risk matrix (Figure 4.2) is a useful method that can be used during risk assessment. The matrix can help to determine the level of risk, based on the likelihood and the impact of the risk. All potential risks can be scored with this matrix. Based on the scores, the risks can be prioritised (Kinney & Wiruth, 1976).

Likelihood \ Impact	Negligible	Minor	Moderate	Significant	Severe
Very Likely	Low Med	Medium	Med Hi	High	High
Likely	Low	Low Med	Medium	Med Hi	High
Possible	Low	Low Med	Medium	Med Hi	Med Hi
Unlikely	Low	Low Med	Low Med	Medium	Med Hi
Very Unlikely	Low	Low	Low Med	Medium	Medium

Figure 4.2: Risk Matrix (RISK-ACADEMY, 2019)

- **Information Management**

The definition of information is, according to Cambridge Dictionary (2021), ‘facts about a situation, person, event, etc.’. Information means something to the receiver (Boddy, 2014). Information management is the process of managing the information and making sure that the right information is at the right people, at the right time. It is a process that manages people, processes and technology that deliver, process and use information for management and business intelligence (O'Neal, 2017). This process helps to structure all necessary information and keep all involved stakeholders informed, which can support decisions. This will result in more clarity to the stakeholders.

An information management system can help in this process. Different kinds of information management systems are known, such as business intelligence systems and customer relationship management systems (Smartsheet, 2021).

Information management is an important process, as it makes sure all involved stakeholders are informed with the right information. Therefore, it is recommended to investigate the possibilities of information management. A question that arises is for example: who needs which information at what time? An information management system can be useful at PREI but has no priority at this point in time.

- **Quality Management**

Quality management describes the process of managing and controlling all activities and tasks that are necessary to achieve and maintain the desired level of excellence (Barone, 2020). It is a very important process in both the manufacturing and the service sector (Boddy, 2014). At PREI, the quality of the tests must be good: the outcomes must be reliable. ProRail has a quality management system, named KMS. This system can support quality management at the lab.

- **Performance Monitoring**

As the name of this process already suggests, performance monitoring is about the monitoring of the performance. Performances can be monitored with key performance indicators (KPIs). KPIs are a set of the most important measures. These measures indicate how well the organisation performs on specific measures (Boddy, 2014). An example of a KPI is capacity utilisation.

The set of KPIs should be composed based on the goals of the organisation, as with KPIs, it can be monitored if the organisation is meeting its goals. KPIs are expressed with a number and are related to a norm or goal. Visualising KPIs can help with monitoring the performance. A popular way of visualising (the scores on) the KPIs is a dashboard (Zwanenburg, 2018).

4.1.3 Development Management

The second process group within the management processes is the group *development management*. As the name already suggests, the developments are managed in this group. At PREI, most developments are based on the lifecycle of the products. When something breaks down, new products should be added to the lab. Also, when products or techniques become outdated, they should be renewed. Therefore, new developments should be monitored.

Developments can be done both for the long-term and the mid-term. This depends on the product or technology. For example, a computer monitor does not become outdated quickly. They have to be replaced when they break down, but the technology does not have to be monitored all the time. On the other hand, other technologies used in the lab may get updated. To ensure a representative test environment, these technologies should be updated. It is recommended to make a long-term and mid-term planning for developments and determine for all products and technologies how important developments are. This can help as input for the long-term and mid-term planning. An additional tool can be a scrap list with tasks that should be done.

- **Project Management**

Developments can be accommodated in projects. This can help to structure the process of development. Processes, methods, skills, knowledge and experience are applied to the project, such that the project is completed within a specified timespan, with the desired outcomes and deliverables (Association for Project Management, 2021).

4.1.4 Corporate Focus

The last group of management processes is the group *corporate focus.* PREI is part of ProRail. The activities at the lab are performed in name of ProRail. Also, the lab depends largely on ProRail. The corporate focus of ProRail and PREI must be secured. Four processes are distinguished that are important to do so. These four processes are discussed next.

- **Capacity Requirements**

Capacity requirements are important to determine. This can be done within the lab itself, or more organisation wide. The capacity requirements must be determined because the number of activities that are executed at the lab depends on the capacity. When there is not enough capacity, not all activities can be executed. But when there is too much capacity and the capacity is not fully used, the capacity utilisation will drop. This is also not desired, as costly facilities are not used.

- **Budget Negotiations**

A sufficient budget is necessary to cover the costs and make any investments. Part of the budget comes from the activities that are executed at the lab, for which the applicant pays. The remaining part comes from ProRail. For this, it is important that the required budget has been elaborated and substantiated. Based on the elaboration and substantiation, a decision can be made for what and how much money is being released.

- **Organisational Adjustments**

Organisational adjustments will both have an influence on ProRail as an organisation and PREI as part of ProRail. Any kind of adjustments should therefore be communicated to the involved stakeholders.

- **Investors, Suppliers, and Partners Management**

Investors, suppliers and partners are important parties that should be kept satisfied. Also, it is important to keep track of these parties and their occurrences. When a party does not act as they should, actions are required.

Public Relations is also included in this process. Another activity that is included in this process is establishing and maintaining the SLAs. SLAs are for example established for some of the facilities.

4.2 Core Processes

The core processes are the next type of processes we will discuss. It is known from the literature that the core processes establish the core of the organisation and deliver (valuable) outputs. The core processes that are determined at PREI are *testing*, *demonstrations* and *trainings*. Three different groups of tests can be distinguished: ESC tests, Post21 Functionality tests and other tests. However, the processes of the Post21 Functionality tests and the other tests look very familiar and are combined into one group: other tests.

The ESC testing process is the most important process to design because different internal and external stakeholders are involved. Therefore, all steps in this process must be well elaborated and designed. This will result in less finger-pointing, as the roles and responsibilities should be clear to everyone.

From the literature review in Chapter 3 followed that the most suitable method to capture the core processes is the flowchart method with the addition of swimlanes. This is done for the processes of ESC tests, other tests, demonstrations and trainings. Different symbols are used within the flowcharts. A legend of the symbols can be found in Figure 4.3.

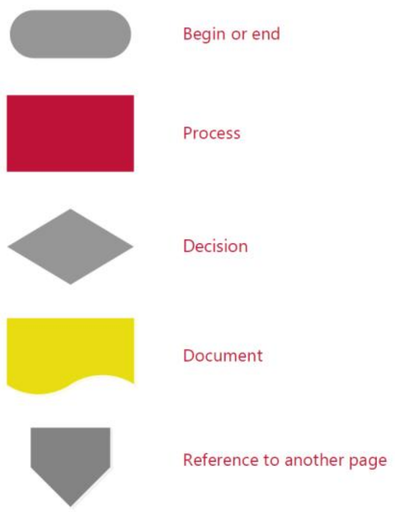

Figure 4.3: Legend of used symbols

In the remainder of this section, the defined core processes are discussed. First, the roles are elaborated. In some cases, more roles are defined in the flowcharts than there are in the current situation. For example, the role *Planner* is added to the ESC testing process, whereas this role is now occupied by the Senior ERTMS System Specialist. This role is defined as a separate role in the flowcharts, as it can be useful to separate the tasks in the future. An important note is therefore that it can be that one person or party occupies different roles. For the ESC tests, the responsibilities are also written down. It is decided to do this for this process, as many different stakeholders are involved and it is a good way to give a clear overview of the responsibilities of every stakeholder. The flowcharts are also given in this section.

4.2.1 Testing

We start with the testing process. As already discussed, two groups of test processes are distinguished: ESC tests and other tests. First, the ESC testing process is elaborated. Thereafter, the process of the other tests is discussed.

4.2.1.1 ESC Tests

The ESC testing process is complex and contains many different steps. And as already mentioned, there are different stakeholders involved in the process. This results in many roles and responsibilities.

4.2.1.1.1 Roles and Responsibilities

Because the roles are both internal and external, it is important to indicate the roles and their responsibilities. Fourteen roles are distinguished. The responsibilities of all roles can be found in Appendix B.

ERA – European Union Agency for Railways.

ESC Test Applicant – Or Entity applying for ESC Demonstrations. The party that initiates the ESC Test Campaign. This will typically but not necessarily be the OBU Supplier (e.g. vehicle manufacturer, railway undertaking, Infrastructure Manager, vehicle owner).

ESC Test Facility Manager – The employee of PREI who is responsible for (the management of) the test facility.

ESC Test Manager – The party who is responsible for managing the Test Campaign. Can be someone from ProRail or an external person.

ESC Tester – The party that is involved in the execution of the test campaign. The ESC Tester is appointed and informed by the ESC Test Manager. Can be someone from ProRail or an external person.

Notified Body (NoBo) – A body that has been notified by a Member State of the European Union to be responsible for assessing the conformity or suitability for use of the interoperability constituents or for appraising the 'EC' procedure for verification of the subsystems.

OBU Supplier – A party responsible for the design and implementation of the OBU.

Planner – The employee of PREI who is responsible for the schedule of the lab.

PREI (Account) Manager – The employee of PREI who the contact person for all parties for all PREI related affairs is. This person takes care of all the coordination that is required on the ProRail side.

PREI Test & Support – The employee of PREI that can execute tests in the lab.

ProRail ESC Manager – The employee of ProRail who the contact person for all parties for the Test Campaign is. This person takes care of all the coordination that is required on the ProRail side.

ProRail EG ERTMS – The ERTMS expert group within ProRail.

ProRail Vehicle Authorisation – The department of ProRail responsible for the vehicle authorisation.

Trackside Supplier – A party responsible for the design and implementation of ETCS trackside products (e.g. the RBC).

4.2.1.1.2 Flowcharts

The literature in Chapter 3 indicates that flowcharts can be hard to read when the flowchart becomes too large. Because this process is a complex process with many steps and roles, it is decided to divide the process of ESC tests into five phases: the preparation, PREI execution, field execution, completion and issue handling. All distinguished phases are captured within one flowchart. The whole process is thus captured and visualised with five flowcharts, which are linked to each other. For every flowchart, a table is composed with all steps. Additional remarks and clarifications are added to these tables.

The flowchart for the preparation phase of the ESC testing process can be found in Figure 4.4. The flowchart can also be found on a larger scale in Appendix C, together with the flowcharts of the other phases.

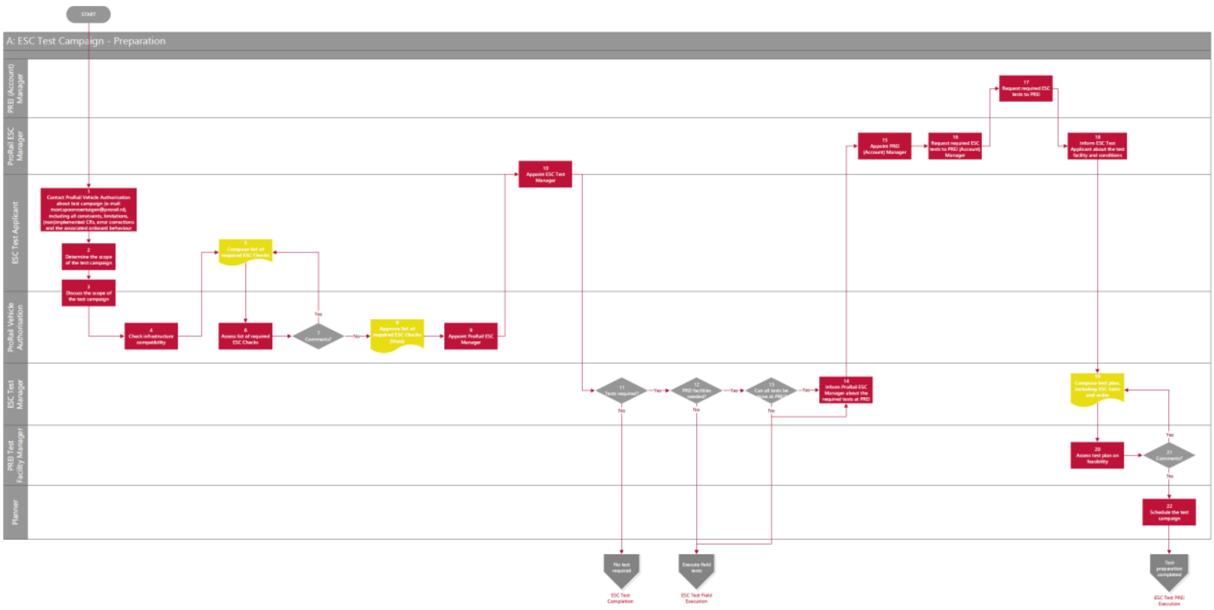

Figure 4.4: Flowchart of the preparation phase of the ESC test process

The phases are indicated with a letter. The steps in the phases are numbered, where:

- A Preparation,
- B PREI Execution,
- C Field Execution,
- D Completion,
- E Issue Handling.

For example, the first step in the completion phase is indicated by D1. This makes it easier to refer to the steps in the different phases.

Also, all steps of the testing process are appointed and remarks are added if necessary in tables. Every phase (and thus flowchart) has its table. This is done to give additional information on the steps that can be useful for a better understanding of the process steps and give more specific information, especially because different stakeholders are involved. Also, the remarks help to delineate the responsibilities of the stakeholders. The steps are also compared to the steps of the process as described by the ERA in the document called 'Principles for the demonstration of ETCS System Compatibility' (European Union Agency for Railways, 2020). This document is publicly accessible and is used as an indication for the ESC testing process by different parties. Similarities between the steps are indicated in the last column of the table to make it easier for the involved parties to understand the steps.

Table 4.2 includes a part (step A1-A4 out of the 22 steps) of the table composed for the preparation phase of the ESC testing process. The whole table and the tables for the other phases are included in Appendix C.

Table 4.2: Part of the table of the preparation phase of ESC testing process, with remarks and indication of the steps of the ERA process

A: ESC Test Campaign – Preparation			
Step	**Activity**	**Remark**	**Step ERA process**
A1	Contact ProRail Vehicle Authorisation about test campaign (e-mail: inzet.spoorvoertuigen@prorail.nl), including all constraints, limitations, (non)implemented CRs, error corrections and the associated onboard behaviour	This e-mail should contain: • The contact details for the test; • The involved train; • The ERTMS onboard equipment; • The involved software; • A time schedule; • All constraints, limitations, (non)implemented CRs, error corrections and the associated onboard behaviour	1
A2	Determine the scope of the test campaign	This step is about the extent of ESC tests and the conditions for their execution. The OBU Supplier and the Trackside Supplier may be consulted if necessary.	1
A3	Discuss the scope of the test campaign	This step is about the extent of ESC tests and the conditions for their execution. The OBU Supplier and the Trackside Supplier may be consulted if necessary.	1
A4	Check infrastructure compatibility	The constraints, limitations, (non)implemented CRs, error corrections and the associated onboard behaviour that are included in the e-mail in step A1 are checked and assessed.	

4.2.1.2 Other Tests

All other tests are included in this group. The stakeholders in this process are mostly only internal parties and collaborations between different parties or departments are common. Because this group includes different kinds of tests, the process is designed in less detail in comparison with the process for ESC tests. However, this is acceptable, as the process is mostly internal. An important remark is that it is still useful to have a description of the process, as this will help to understand the process.

4.2.1.2.1 Roles

The following roles are distinguished for the group other tests:

Planner – The employee of PREI who is responsible for the schedule of the lab.

PREI (Account) Manager – The employee of PREI who is the point of contact for the external parties.

PREI Test & Support – The employee of PREI that can execute and support tests in the lab.

Test Initiator – The party who initiates the test.

4.2.1.2.2 Flowchart

The flowchart of the other tests can be found in Figure 4.5. This flowchart is also included on a larger scale in Appendix C.

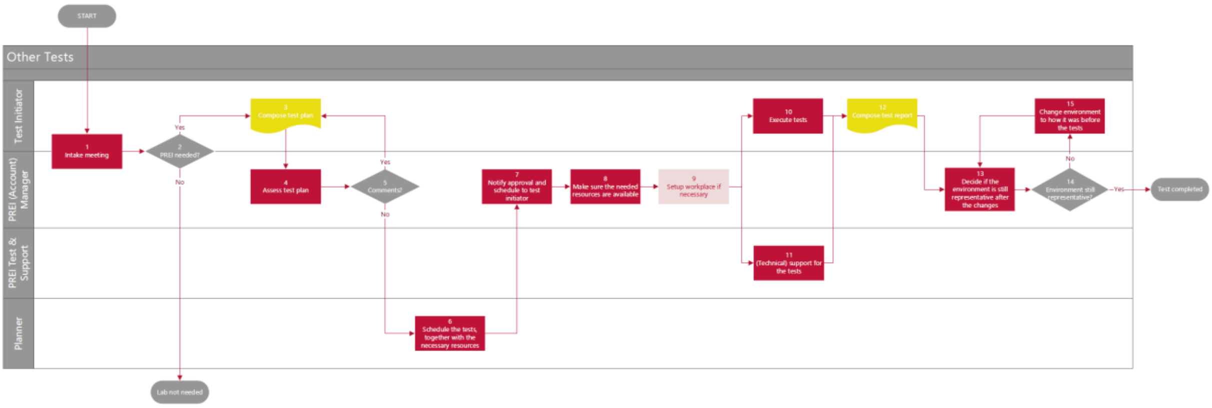

Figure 4.5: Flowchart of the process of the group other tests

4.2.2 Demonstrations

The second core process of PREI is demonstrations. Demonstrations are given to show for example how ERTMS works in the Netherlands. The process of giving the demonstrations involves especially internal stakeholders. Of course, the participants of the demonstration can be external as well. The roles of the stakeholders are described and the process is captured with a flowchart.

4.2.2.1 Roles

The following roles are distinguished for the demonstration process:

Initiator – The party that initiates the demonstration.

Planner – The employee of PREI who is responsible for the schedule of the lab.

PREI Test & Support – The employee of PREI that can execute demonstrations in the lab.

4.2.2.2 Flowchart

The flowchart of the demonstration process can be found in Figure 4.6. This flowchart is also included on a larger scale in Appendix C.

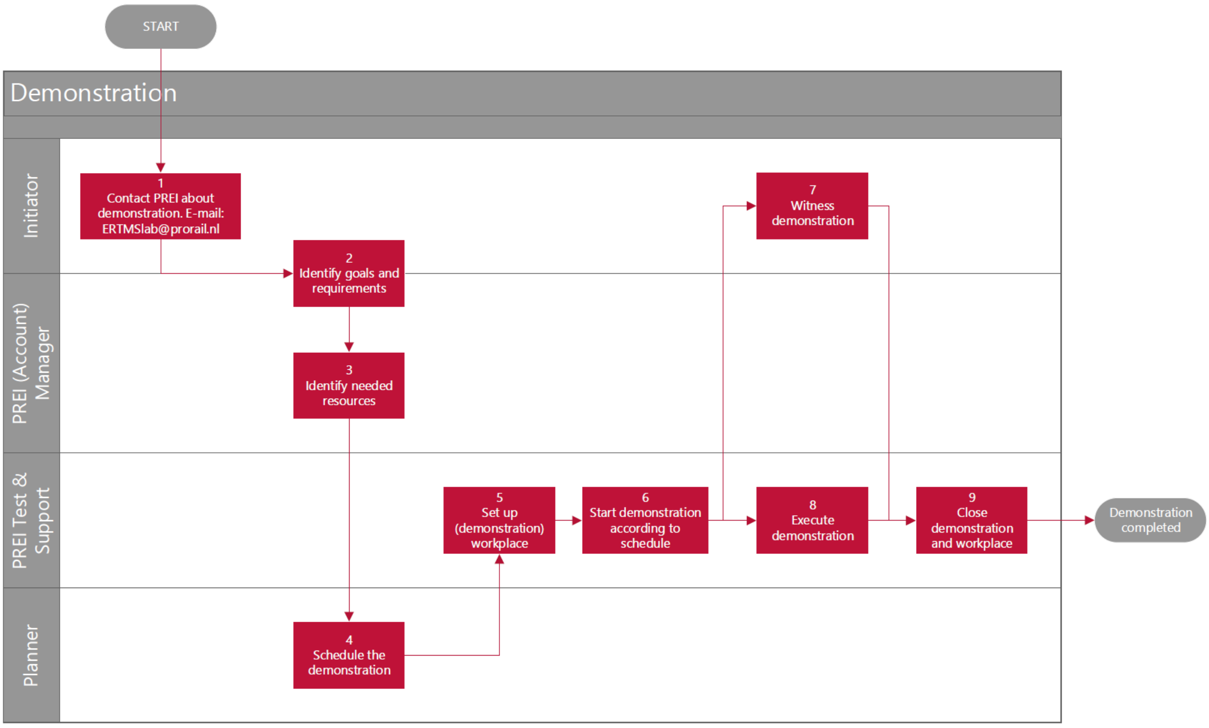

Figure 4.6: Flowchart of the demonstration process

4.2.3 Trainings

The last core process of PREI includes the trainings. Trainings are given at the lab to educate and re-educate people. The most important stakeholders are the Railcenter, who initiate the trainings, and some internal stakeholders of ProRail. The roles are described and the process is designed with a flowchart.

4.2.3.1 Roles

The following roles are distinguished for the training process:

Planner – The employee of PREI who is responsible for the schedule of the lab.

PREI Test Facility Manager – The employee of PREI who is responsible for (the management of) the test facility.

PREI Test & Support – The employee of PREI that can execute and support tests in the lab.

Trainer Railcenter – The trainer from the Railcenter who initiates and leads the training.

4.2.3.2 Flowchart

The flowchart of the training process can be found in Figure 4.7. This flowchart is also included on a larger scale in Appendix C.

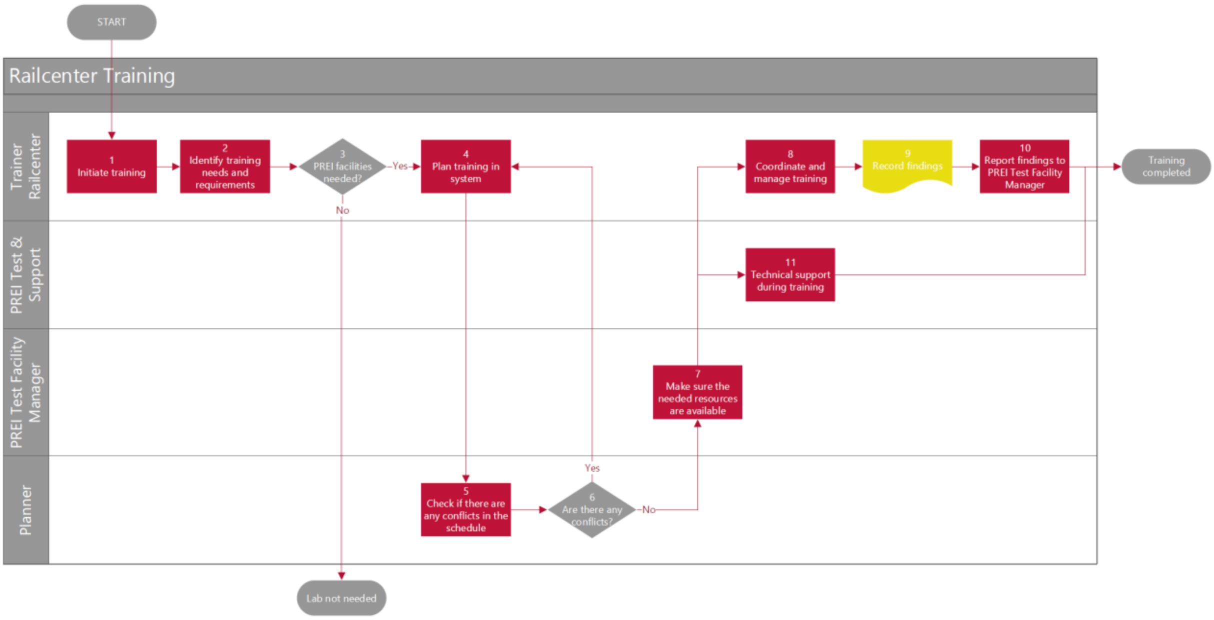

Figure 4.7: Flowchart of the training process

4.3 Supporting Processes

The last type of processes are supporting processes. We know from the literature that the supporting processes support the core and management processes of an organisation. Six supporting processes are distinguished that are important for PREI: *operational scheduling*, *human resources*, *indirect procurement*, *maintenance*, *facility management* and *IT management*.

These processes are elaborated at the same level of detail as the management processes, for the same reasons. The form proposed for the management processes can be used to capture these processes as well.

4.3.1 Operational Scheduling

The first supporting process that is distinguished is operational scheduling. In Chapter 1 and 2 is already discussed why a schedule is important for the activities at PREI. This schedule is also called the operational schedule, as the activities on the operational level are scheduled. The operational level is the level where the execution is controlled, on the short-term (Zijm, 2000). This process ensures that the operational schedule is made, updated and followed, which results in a better overview of the activities, and a more efficient way of working.

At this moment, the operational schedule is composed manually. This is still going well, but we expect that this will cause problems in the future. In the second part of the research, we propose a model to solve this scheduling problem. The aim of the model is to create a schedule for the activities at the lab, where the schedule is visualised with a graph. When the model is implemented, the scheduling process will change. The schedules do not have to be composed by hand anymore, which results in a more efficient and future proof process.

4.3.2 Human Resources

The term human resources is used for two concepts. At first, the people that work for an organisation are described with the term human resources. The other concept is the department of an organisation that is responsible for managing everything related to the employees (Human Resources Edu, 2021). The second concept is also referred to as human resource management. Human resource management covers human resource flow (e.g.

recruitment and training), work systems (e.g. supervisory style), employee influence (e.g. involvement in decision making) and reward management (e.g. benefits) (Boddy, 2014).

Several people are working for and at the lab, so there are human resources involved that should be controlled. But as PREI is part of ProRail, the human resource management tasks (e.g. recruitment) are done by the HR department of ProRail. However, it is important the human resources are not forgotten at PREI. Everyone should be and stay involved. Also, it is still important at PREI that it is monitored how much capacity is needed.

4.3.3 Indirect Procurement

Procurement or purchasing is the concept of buying the necessary materials, products or services from suppliers (Slack, Brandon-Jones, & Johnston, 2013). Indirect procurement refers to the purchasing of materials, products or services that are necessary to keep the business running. This can be buying office supplies, but also acquiring services (Loi, 2021).

Indirect procurement is an important process for PREI. ProRail has to ensure that the testing environment is always representable. Different products and services are necessary to ensure this and the daily business of the lab. Therefore, it is important that this process is elaborated.

4.3.4 Maintenance

Maintenance within an organisation is avoiding failures by taking care of the physical resources (Slack, Brandon-Jones, & Johnston, 2013). PREI is a lab facility where other parties can test their products. As already mentioned, ProRail is responsible that the testing environment of the lab is always representable. This can only be ensured when maintenance is done. Maintenance is also necessary for other resources, which do not have to be directly linked to the core activities. Some maintenance is done by third parties. This is captured within the SLAs as mentioned in Section 4.1.4. Three types of maintenance can be determined (Slack, Brandon-Jones, & Johnston, 2013):

- Run to breakdown maintenance: Maintenance is done after a failure.
- Preventive maintenance: Trying to reduce or eliminate failures by servicing (e.g. checking) at planned intervals.
- Condition-based maintenance: Maintenance is only done when necessary (based on monitoring).

An organisation does not have to stick to one of these types of maintenance: a mixture can be used as well. A computer monitor can for example be replaced when it breaks down, but for other resources, it is better to reduce or eliminate failures. With this process, a maintenance plan can be determined, including the maintenance types for all resources. This will result in effective maintenance (Slack, Brandon-Jones, & Johnston, 2013). The two most important benefits of effective maintenance at PREI are higher quality and increased reliability.

4.3.5 Facility Management

ISO (2018b) describes facility management as follows: 'Facility management (FM) integrates multiple disciplines in order to have an influence on the efficiency and productivity of economies of societies, communities and organizations, as well as the manner in which individuals interact with the built environment'. This description is quite broad, but the impact of facility management is broad as well (International Organization for Standardization, 2018b).

Facility management can be seen as the process that delivers efficient and effective support to the organisation. This is an important process for many organisations or even departments.

Facility management is of added value for PREI. Several benefits can be identified, including (International Organization for Standardization, 2018b):

- A more cost-effective working process,
- Increased efficiency and effectiveness of the organisation (or lab in this case),
- Improved service consistency,
- Improvements in managing workforce productivity, safety and health.

An important remark is that facility management is already a department of ProRail. Therefore, tasks that are included in this process may be the responsibility of this department. However, it is still important to think about the process, so the responsibilities are clear.

4.3.6 IT Management

IT management is the process where all information technology (IT) operations and resources are monitored and managed. These IT operations and resources include hardware, software and networks. IT management makes sure that the information systems work efficiently and add value to the organisation (IBM, 2021).

The activities and operations of PREI are depending on different IT resources. Therefore, it is of great importance to implement IT management correctly, so the lab can stay up and running.

4.4 Storage of the Processes

The processes should be stored properly once designed. When not everyone who should be able to access the processes can access them, the designs are not important anymore as they cannot be used. Document management is an important aspect that should be considered. Document management is not only important for and within ProRail itself, but also to the other stakeholders in especially the ESC testing process. In the remainder of this section, the storage and distribution of the processes are discussed. The storage of the processes is discussed from the view of two different groups of stakeholders: ProRail and external stakeholders. The requirements and document management are slightly different for these two groups.

4.4.1 ProRail

As already mentioned, Microsoft software is frequently used within ProRail. Microsoft has a web-based platform for document management, process management and online collaborations that integrates with Microsoft Office: SharePoint. ProRail uses SharePoint to unlock information sources and manage document flows. Also, collaboration portals are introduced in which the documents can be shared and distributed, and to be able to collaborate in documents due to the integration with Microsoft Office products. SharePoint keeps track of and manages versions automatically (ProRail, 2020c; Microsoft, 2020).

One of the SharePoint portals is the *KMS Techniek* portal. This portal includes all defined processes of the department Techniek. Once the processes are determined and captured, they are stored at this portal, where all internal stakeholders that need to access the processes can find them.

4.4.2 External Stakeholders

Most process descriptions are especially important within ProRail. However, some processes also involve external stakeholders. The most important process that does involve external stakeholders is the ESC testing process. In Section 4.2.1.1.1, fourteen different roles are distinguished. These roles are both internal and external. When different parties are involved in a process, it is of great importance that the process is clear to everyone and that everyone is on the same page, as already pointed out in the problem description in Section 1.2. This clarifies who is responsible for what and results in less finger-pointing. The process can be adjusted to specific cases, but only by mutual agreement. Every involved party must agree with the adjustments.

Parallel to this research, the possibilities for an ERTMS website of the lab are examined. On this website, the processes can be stored for external stakeholders. This website will be public, which means that everyone can access the processes if they need to.

4.5 Validation

The process design started with multiple meetings with various stakeholders to clarify the processes of PREI. Different processes are discussed during these meetings. After the first meetings, a list is composed of possibly suitable processes of the lab. This list is discussed with the stakeholders.

After the list of processes was established and validated, the processes are elaborated. The level of elaboration depends on the importance. It became clear during the meetings that the designs of the core processes are of greater importance because external stakeholders are involved in these processes. These processes are therefore elaborated in more detail than the management and supporting processes.

The processes are validated multiple times during the design. Meetings were planned to discuss the processes with the concerned stakeholders. Also, the end results are shared and final reviews are given. The remarks of the final reviews are applied to the processes. Therefore, it can be concluded that the elaborated processes are valid.

An important remark is that the lab is new and that some activities are not often performed yet. Therefore, some process descriptions are rather based on expectations than on experiences. When the processes are followed with real activities, it can be that some parts of the process do not work well in the designed way. The process should be changed when that is the case. All documents will be available for and owned by ProRail, so it is possible to change processes if necessary.

4.6 Conclusion

Based on literature and meetings with the stakeholders, processes are determined that are important for PREI. The processes are categorised on the three process types found in the literature. The management and supporting processes are determined and elaborated. The elaboration does not include many details, as it became clear that the importance is relatively low at this moment. We proposed a form to design the management and supporting processes.

All core processes are captured with flowcharts. The stakeholders are defined and their roles are described. The ESC testing process is complicated and does involve different external stakeholders. Five phases are determined. A flowchart is composed for every phase to keep a clear overview. Also, all steps are elaborated in a table with additional remarks.

The captured processes should be stored to allow the involved stakeholders to access them. ProRail uses SharePoint for document and process management. The processes can also be stored on a public website, so external stakeholders can access the processes as well.

We can conclude that the processes are valid, as they are composed based on literature and validated with the stakeholders. Also, the processes and progress of the designs are discussed multiple times. An important remark is that it is possible that (parts of) the processes do not work the way they are designed. When that is the case, they should be changed.

PART 2

SCHEDULING

5 Literature Review: Scheduling

The second part of the research covers the PREI scheduling or resource allocation problem. Resource allocation problems are widely discussed problems in literature, which we will discuss in this chapter. First, it is important to define what resource allocation is and how this can be applied to this research. This is done in Section 5.1. The scheduling problem is introduced in this section as well. The Resource-Constrained Project Scheduling Problem (RCPSP) is discussed in Section 5.2. We start with a general discussion, followed by specific models within the RCPSP. This literature review is concluded in Section 5.3.

5.1 Resource Allocation

Most organisations have to deal with limited resources. Resource allocation is about how a set of limited resources can be shared among various processes (Lister, 1979). Important to remember is that *one* set of *limited* resources is available for *all* processes. At this level of control, there is hardly to no flexibility in the capacity of the resources.

Scheduling is closely related to resource allocation, and can be seen as a subtopic of resource allocation. A schedule can be influenced by the way resources are allocated (Lister, 1979). The main goal of a schedule is to indicate which activity should be executed at what time and where. The terms *planning* and *scheduling* are often mixed up. With *planning* we indicate strategic and tactical control, with *scheduling* we indicate the operational control (Hans E. W., 2019). Strategic control is focussed on the long-term planning, tactical control is focussed on mid-term plans (e.g. allocation of production) and operational control is focussed on the daily schedule of the activities (Zijm, 2000; Brunaud & Grossmann, 2017). To summarise: planning is used to lay the groundwork, and scheduling will help to get the activities done.

In this research, we focus on the scheduling of the activities at PREI. There is a known, limited number of available resources. We aim for a scheduling method that is able to create a schedule for the activities at the lab.

5.1.1 Scheduling Problem

The scheduling problem is an optimisation problem within operations research and is widely discussed in the literature. The problem can become very complex, especially when the number of activities that need to be scheduled increases. As already stated, scheduling is a process that takes place on the operational level. Short-term schedules are created to schedule the activities (Zijm, 2000; Brunaud & Grossmann, 2017).

The aim of scheduling is to schedule all work within the given time window, given a set of activities and the fixed number of available resources (people, machines and tools). The objective is the performance measure of the schedule and depends on the situation (Baker & Trietsch, 2007). Possible objectives are to minimise the total flowtime, lateness, tardiness or makespan, or to maximise the utilisation (Hans E. W., 2019; Baker & Trietsch, 2007). In the scheduling theory, three main objectives are prominent: the turnaround time, the due-date performance and the throughput time (Baker & Trietsch, 2007).

The scheduling problem should also take different kinds of constraints into account. Not all these constraints apply to the basic deterministic single-machine model but are necessary for more complex models. The first group of constraints are time constraints. An activity can for example only be scheduled after the release date and should be finished before the due date. The second group are the capacity constraints. These constraints include for example the number of machines that can be used. The last group includes technological constraints (Hans

E. W., 2019). It can for example be important that activity i is scheduled before activity j (precedence relations).

5.2 Resource-Constrained Project Scheduling Problem

The scheduling problem can become more complex when resources are necessary that have limited availability. Then, it is the question how the resources should be assigned to the activities, optimising a defined objective. Literature shows that the Resource-Constrained Project Scheduling Problem (RCPSP) is a powerful framework for the description of scheduling problems (Jannach & Friedrich, 2009). The scheduling problems solved by the RCPSP show similarities with our problem description. For example, at PREI, also limited resources are available to perform the activities and our aim is to create an operational schedule and an overview of the resource allocation over a defined time period. Because of the similarities we have found in both solution methods, we elaborate further on this RCPSP.

The RCPSP does not represent an isolated research area (Özdamar & Ulusoy, 1995). Many well-known optimisation problems are special cases of the more general RCPSP (Brucker, Drexl, Möhring, Neumann, & Pesch, 1999; Özdamar & Ulusoy, 1995; Jannach & Friedrich, 2009). Examples are the Operating Room (OR) scheduling problem and the job shop scheduling problem. These special cases will be introduced later in this section.

The models in this field are rich and can be difficult to solve (Brucker, Drexl, Möhring, Neumann, & Pesch, 1999). The RCPSP is a combinatorial optimisation problem, which means that the problem is defined by a solution space with a subset of feasible solutions with an objective function. Also, it is known from the literature that the RCPSP is NP-hard, which means that the problem cannot be solved to optimality in polynomial time (Jannach & Friedrich, 2009). Literature suggests many efficient algorithms that can provide good results in a short time. Examples are simulated annealing, local search and tabu search, which are meta-heuristics (Jannach & Friedrich, 2009; Özdamar & Ulusoy, 1995).

Although the RCPSP is NP-hard, we start creating an integer linear program (ILP) of the PREI scheduling problem as this helps us define the problem description more concrete. From this ILP we obtain a clear description and overview of the problem, which can be the first step in optimising the schedules. When we have a clear ILP, we can expand this model in order to fit better to reality. We can do this by for example including components of the RCPSP that match our problem. As the literature suggests multiple different mathematical problems in order to solve scheduling problems, we will search for similarities and differences between the problems proposed in literature (and their solution method) and from there, we will create a model that fits our problem.

In the remainder of this section, four cases of the RCPSP are discussed. To get a better idea of the RCPSP, we start with the basic model, which is the basis of the other models. Then, we expand the basic model with so-called modes, which results in the multi-mode RCPSP (MRCPSP). In this model, an extra index is added to the decision variable compared to the basic model. We can use this notation for our scheduling problem, as we want to assign a starting time and workplace to all activities. Then, we discuss two special cases of the RCPSP: the OR scheduling problem and the job shop scheduling problem. Many resource constraints defined in the OR scheduling problem are suitable for the PREI scheduling problem. For the job shop scheduling problem, various objectives are discussed in the literature that we can use. For all cases of the RCPSP, the mathematical formulation, based on ILP, is given.

5.2.1 Basic Model

Hartmann (1999) introduces a basic, standard model for the RCPSP that is rather simple. They state that many real-world aspects are covered with this model. The model is composed for the scheduling problem of a project that consists of activities that are related to two kinds of constraints: precedence constraints and resource constraints. Given are the duration, resource requirements and precedence relations for all activities, and the availability of all resources. All information is assumed to be known in advance and deterministic.

They consider a project with activities $j = 1, \ldots, J$. Activity j has processing time p_j. The planning horizon is divided into periods of equal lengths, e.g. days. The processing times are defined as a multiple of a period (e.g. 5 days). An activity is executed continuously and cannot be interrupted. Activity $j = 0$ represents the beginning of the project and activity $j = J + 1$ represents the end of the project. Activity 0 is called the source of the network, and activity $J + 1$ is called the sink. These activities are no real activities but are indications of the start and end of the project. Therefore, their processing times are $p_0 = p_{J+1} = 0$. The set of all activities, including the start and end activity, is denoted by J^+.

Precedence relations are indicated with the sets P_j. All predecessors $h \in P_j$ should be executed and thus completed before activity j can start. The sets S_j include the successors of activity j. The activities in these sets can only be started when activity j is completed.

Resources are necessary to be able to execute the activities (except for the source and sink activity). In this case, the resources are renewable. This means that they are fully available in every time period. The set of renewable resources k is denoted by K^ρ, where $k \in K^\rho$. The availability of the resources is denoted by R_k^ρ and is assumed to be constant for every time period. r_{jk} units of resource k are required for activity j, every time period the activity is executed. As already said, the source and sink activity do not require resources. Therefore, $r_{0k} = r_{J+1,k} = 0, \forall k \in K^\rho$.

All parameters are assumed to be nonnegative integers. The objective is to minimise the makespan. This means that the aim is to find the earliest possible end time of the project. The schedule assigns a start time s_j (or a finish time f_j) to every activity j. The precedence and resource constraints may not be violated.

Hartmann (1999) defines time instants and periods, where period t starts at time instant $t - 1$ and ends at time instant t. The upper bound T on the makespan is computed by $T := \sum_{j=1}^{J} p_j$. Given the planning horizon T, the set of time instants is denoted by $\mathcal{T} = \{0, \ldots, T\}$ and the set of periods is denoted by $\mathcal{T}' = \{1, \ldots, T\}$.

They determine the earliest possible start time ES_j and earliest possible finish time EF_j for every activity $j \in J^+$ by forward recursion. The latest possible start time LS_j and latest possible finish time LF_j for every activity $j \in J^+$ are determined with backward recursion. Time windows are created for every activity. Every activity $j \in J^+$ should start in the time window $\{ES_j, \ldots, LS_j\}$ and finish in the time window $\{EF_j, \ldots, LF_j\}$. They state that time windows can be useful for two reasons. First, the number of variables can be reduced in the mathematical formulation. Second, time windows allow evaluating partial schedules of whole scheduling procedures.

5.2.1.1 Mathematical Programming Formulation

Hartmann (1999) also proposes a mathematical programming formulation, based on the above-described problem and parameters. A mathematical programming formulation can help to translate the problem description. For this problem, an ILP formulation is composed. A binary decision variable x_{jt} is introduced to indicate whether the activities $j \in J^+$ are finished at $t \in \mathcal{T}$ or not, where:

$$x_{jt} = \begin{cases} 1, & if\ activity\ j\ is\ finished\ at\ time\ t \\ 0, & otherwise. \end{cases}$$

The aim is to schedule the set of activities subject to the precedence and resource constraints, at minimal duration. So, the makespan is minimised. The following ILP follows:

$$minimise \sum_{t=EF_{J+1}}^{LF_{J+1}} t \cdot x_{J+1,t} \tag{5.1}$$

Subject to

$$\sum_{t=EF_j}^{LF_j} x_{jt} = 1\,, \qquad j \in J^+ \tag{5.2}$$

$$\sum_{t=EF_h}^{LF_h} t \cdot x_{ht} \leq \sum_{t=EF_j}^{LF_j} (t - p_j) \cdot x_{jt}\,, \qquad j \in J^+, h \in P_j \tag{5.3}$$

$$\sum_{j=1}^{J} r_{jk} \sum_{b=t}^{t+p_j-1} x_{jb} \leq R_k^{\rho}\,, \qquad k \in K^{\rho}, t \in \mathcal{T} \tag{5.4}$$

$$x_{jt} \in \{0,1\}, \qquad j \in J^+, t \in \mathcal{T} \tag{5.5}$$

The objective (5.1) minimises the finish time of the last activity (the sink), and thus the makespan. With constraints (5.2) is ensured that each activity is executed only once. Constraints (5.3) are precedence constraints. These constraints ensure that when activity j has to be executed after activity h, this is the case. The resource constraints are observed by constraints (5.4). The binary decision variable is defined by constraints (5.5).

5.2.1.2 Release Dates and Due Dates

In the proposed model, release dates and due dates are not considered. The release date is the time period from which an activity may be started. The due date is the time period an activity should be finished. Hartmann (1999) introduces the release date and due date and suggests two related constraints. They denote the release date of activity j by $\underline{\delta}_j$ and the due date of activity j by $\overline{\delta}_j$. Constraints (5.6) are related to the release date and constraints (5.7) to the due date. These constraints can be added to the basic model to ensure that all activities are started not earlier than the release date and that all activities are completed at their due date.

$$\sum_{t=EF_j}^{LF_j} t \cdot x_{jt} - p_j \geq \underline{\delta}_j, \qquad j \in J^+ \tag{5.6}$$

$$\sum_{t=EF_j}^{LF_j} t \cdot x_{jt} \leq \overline{\delta}_j, \qquad j \in J^+ \tag{5.7}$$

5.2.2 Multi-Mode RCPSP

The multi-mode resource-constrained project scheduling problem (MRCPSP) is an extension of the basic RCPSP. The MRCPSP model allows different alternatives (called modes) in which an activity can be executed. A mode is a combination of the duration and the resource requirements that allow the activity to be completed. For example, an activity can be executed in four time periods by two workers, or in eight time periods by one worker (Hartmann, 1999).

Hartmann (1999) introduces the MRCPSP. They denote the number of modes in which activity j can be executed by M_j. The set of modes is denoted by $\mathcal{M}_j = \{1, \dots, M_j\}$. The processing time of activity j in mode m is denoted by p_{jm}.

Renewable resources were already introduced in the basic model. The set of renewable resources k is denoted by K^ρ. Non-renewable resources have limited availability over the whole time horizon. An example of a non-renewable resource is money. A budget is often set for the whole project rather than for one time period. The set of non-renewable resources is denoted by K^ϑ. The availability of renewable resources is denoted by R_k^ρ and is assumed to be constant for every time period. The capacity of the non-renewable resources is R_k^ϑ units for the whole project. r_{jmk} units of resource k are required for activity j in mode m.

5.2.2.1 Mathematical Programming Formulation

Hartmann (1999) proposes a mathematical programming formulation for the MRCPSP. With the RCPSP, only the finish time (and with that the start time) of an activity has to be determined. Here, also the mode has to be determined. Therefore, the binary decision variable is expended with the mode. This results in the binary decision variable x_{jmt} to indicate whether the activities $j \in J^+$ are executed in mode $m \in \mathcal{M}_j$ and finished at $t \in \mathcal{T}$ or not, where:

$$x_{jmt} = \begin{cases} 1, & if\ activity\ j\ is\ executed\ in\ mode\ m\ and\ finished\ at\ time\ t \\ 0, & otherwise. \end{cases}$$

The aim is to schedule the set of activities subject to the precedence and resource constraints, at minimal duration. This results in the following ILP:

$$minimise \sum_{t=EF_{J+1}}^{LF_{J+1}} t \cdot x_{J+1,1,t} \tag{5.8}$$

Subject to

$$\sum_{m=1}^{M_j} \sum_{t=EF_j}^{LF_j} x_{jmt} = 1, \qquad j \in J^+ \tag{5.9}$$

$$\sum_{m=1}^{M_h} \sum_{t=EF_h}^{LF_h} t \cdot x_{hmt} \le \sum_{m=1}^{M_j} \sum_{t=EF_j}^{LF_j} (t - p_{jm}) \cdot x_{jmt}\,, \qquad j \in J^+, h \in P_j \tag{5.10}$$

$$\sum_{j=1}^{J} \sum_{m=1}^{M_j} r_{jmk} \sum_{b=t}^{t+p_{jm}-1} x_{jmb} \le R_k^\rho\,, \qquad k \in K^\rho, t \in \mathcal{T} \tag{5.11}$$

$$\sum_{j=1}^{J} \sum_{m=1}^{M_j} r_{jmk} \sum_{t=EF_j}^{LF_j} x_{jmt} \le R_k^\vartheta\,, \qquad k \in K^\vartheta \tag{5.12}$$

$$x_{jmt} \in \{0,1\}, \qquad j \in J^+, m \in \mathcal{M}_j, t \in \mathcal{T} \tag{5.13}$$

The objective (5.8) minimises the makespan. With constraints (5.9) is ensured that each activity is executed only once, in one mode. Constraints (5.10) are precedence constraints. The constraints ensure that when activity j has to be executed after activity h, this is the case.

Renewable and non-renewable resource constraints are observed by constraints (5.11) and (5.12), respectively. The binary decision variable is defined by constraints (5.13).

5.2.3 OR Scheduling Problem

The Operating Room (OR) scheduling problem is one of the known special cases of the RCPSP. The ORs of a hospital consume many resources, which makes efficient planning of great interest. This allows to efficiently use the available resources and reduce costs (Roland, Di Martinelly, & Riane, 2006; Hans, Wullink, van Houdenhoven, & Kazemier, 2008).

Roland et al. (2006) suggest a mathematical model that is based on the MRCPSP, but without precedence constraints. The model is presented to combine the surgeries planning and scheduling over a short time horizon. First, the model assigns an operating day to each operation. This is the planning part and is done over several days. The scheduling part assigns a starting hour over the day for each operation. Renewable (e.g. surgeons) and non-renewable (e.g. pharmaceuticals) resource availabilities are considered. The objective of this model is to minimise the opening costs of the ORs and the overtime costs.

The time horizon is short: typically one week. The set of opening days is denoted by D, where each opening day $d = 1, \dots, D$. Each day has T time periods, with time period $t = 1, \dots, T$. T is the upper bound of periods in a day. The set of activities J includes all surgical operations j, with $j = 1, \dots, J$. Only elective patients are considered, no emergencies. The processing time p_j of all activities j are assumed to be deterministic.

The set of renewable resources k is denoted by K^{ρ} and the set of non-renewable resources is denoted by K^{ϑ}. Activity j requires r_{jk}^{ρ} renewable resources and r_{jk}^{ϑ} non-renewable resources. The availability of renewable resources is denoted by R_{kd}^{ρ} for day d, and is assumed to be constant for every time period at day d. The capacity of the non-renewable resources is R_{kd}^{ϑ} units for the entire day d. Every activity j has to be executed between the earliest starting day ES_j and the latest starting day LS_j.

The set of surgeons is denoted by C, with surgeon $c = 1, \dots, C$ and its availability is defined by R_{cd}^{C}. There are S identical operating rooms, with operating room $s = 1, \dots, S$. The availability of the operating rooms for day d is denoted by D_{sd}^{N}. This variable represents the normal opening hours of the operating rooms. Overtime is allowed. Therefore, an extra parameter is introduced: the maximum availability of an operating room for day d is denoted by D_{sd}^{M}. This variable represents the total number of available time periods, so the opening hours plus the allowed overtime. When an operating room is open, a fixed amount of C^{open} euros is charged. Overtime costs are denoted by C^{over}.

5.2.3.1 Mathematical Programming Formulation

Roland et al. (2006) propose a mathematical programming formulation for the above described OR scheduling problem. First, two binary decision variables are introduced. The first one is the binary decision variable x_{jsdt} to indicate whether activity j starts in operating room s at day d and at time period t, or not, where:

$$x_{jsdt} = \begin{cases} 1, & \text{if activity } j \text{ starts in room } s, \text{at day } d \text{ and time period } t \\ 0, & \text{otherwise.} \end{cases}$$

The second binary decision variable is z_{sd}. This variable is introduced to indicate if operating room s is opened on day d, where:

$$z_{sd} = \begin{cases} 1, & \text{if operating room } s \text{ is opened on day } d \\ 0, & \text{otherwise.} \end{cases}$$

The overtime in operating room s on day d is represented by the variable l_{sd}, and should be 0 if there is no overtime.

The mathematical formulation, including the objective and the constraints, follows from the above description:

$$\min_{x,z} \sum_s \sum_d [C^{open}\, z_{sd} + C^{over}\, l_{sd}] \tag{5.14}$$

Subject to

$$l_{sd} \geq (t + p_j)x_{jsdt} - D_{sd}^N, \qquad \forall j, \forall s, \forall d, \forall t \tag{5.15}$$

$$ES_j \leq \sum_s \sum_d \sum_t d \cdot x_{jsdt} \leq LS_j, \qquad \forall j \tag{5.16}$$

$$\sum_s \sum_d \sum_t x_{jsdt} = 1, \qquad \forall j \tag{5.17}$$

$$\sum_s \sum_j r_{jk}^{\rho} \sum_{\tau=t-p_j+1}^{t} x_{jsd\tau} \leq R_{kd}^{\rho}, \qquad \forall d, \forall t, \forall k \in K^{\rho} \tag{5.18}$$

$$\sum_j \sum_s \sum_t r_{jk}^{\vartheta} \cdot x_{jsdt} \leq R_{kd}^{\vartheta}, \qquad \forall d, \forall k \in K^{\vartheta} \tag{5.19}$$

$$\sum_s \sum_{j \in O(c)} \sum_{\tau=t-p_j+1}^{t} x_{jsd\tau} \leq R_{cd}^{C}, \qquad \forall d, \forall t, \forall c \tag{5.20}$$

$$\sum_j (t + p_j)x_{jsdt} \leq D_{sd}^M \cdot z_{sd}, \qquad \forall s, \forall d, \forall t \tag{5.21}$$

$$\sum_j \sum_{\tau=t-p_j+1}^{t} x_{jsd\tau} \leq 1, \qquad \forall s, \forall d, \forall t \tag{5.22}$$

$$x_{jsdt}, z_{sd} \in \{0, 1\}, \qquad \forall j, \forall s, \forall d, \forall t \tag{5.23}$$

$$l_{sd} \geq 0, \qquad \forall s, \forall d \tag{5.24}$$

The objective (5.14) minimises the total costs. The overtime of every operating room is determined with constraints (5.15). Constraints (5.16) ensure that all activities are started in the available time window of the activity, which is between the earliest start day and the latest start day. Constraints (5.17) ensure that all activities are executed only once. Constraints (5.18), (5.19) and (5.20) are the resource availability constraints, where constraints (5.18) deal with the renewable resources, constraints (5.19) with the non-renewable resources and constraints (5.20) with the availability of the surgeons. The set $O(s)$ is the set of all activities j that are dedicated to surgeon c. Constraints (5.21) ensure that the maximum capacity of the operating rooms is not exceeded. Constraints (5.22) ensure that there are no overlapping activities in operating rooms, and constraints (5.23) define the binary decision variables.

Because the problem is NP-hard, there is no efficient method to solve the problem to optimality. Therefore, Roland et al. (2006) suggest solving the problem with a meta-heuristic: the genetic algorithm approach.

5.2.3.2 Adding Slack to the Schedule

Hans et al. (2008) propose several constructive heuristics and local search methods to solve the robust surgery loading problem. This problem is concerned with the assignment of elective surgeries to the operating rooms and sufficient planned slack to the operating room days, with the objective of maximising the capacity utilisation and minimising the risk of overtime, and therefore the number of cancelled patients. They assign planned slack to the planned surgeries on each operating day to make the schedule more robust. The slack is determined based on historical statistical data of the duration of the different surgeries.

They denote the operating rooms by K, with operating room $k = 1, \ldots, K$. There are S specialties, with specialty $s = 1, \ldots, S$. The set of surgeries assigned to specialty s in operating room k on day t is denoted by N_{skt}. The planned slack is based on the expected variance of the duration of the planned surgeries, where the expected duration of the planned surgeries for specialty s in operating room k on day t is:

$$\mu_{skt} = \sum_{i \in N_{skt}} \mu_i \tag{5.25}$$

The variance is calculated by:

$$\sigma_{skt}^2 = \sum_{i \in N_{skt}} \sigma_i^2 \tag{5.26}$$

They assume that the surgery durations are mutually independent. The planned slack size δ_{skt} is then calculated by:

$$\delta_{skt} = \beta \cdot \sqrt{\sum_{i \in N_{skt}} \sigma_i^2} \tag{5.27}$$

β $(\beta \geq 0)$ influences the probability that no overtime occurs, so surgeries are completed on time. Hans et al. (2008) assume the sum of the duration of the surgeries to be normally distributed with mean μ_{skt} and standard deviation σ_{skt}. With $\beta = 0.5$, this results in a probability of 69.15% that the surgeries will be finished on time.

If slack is added to the schedule, the capacity constraints of the operating rooms should be:

$$\sum_{i \in N_{skt}} \mu_i + \delta_{skt} \leq c_{kt} + O_{skt}, \qquad \forall s, \forall k, \forall t \tag{5.28}$$

c_{kt} $(c_{kt} \geq 0)$ is defined as the capacity of operating room k on day t, and O_{skt} $(O_{skt} \geq 0)$ is defined as the overtime of operating room k on day t for specialty s. In the constraints (5.28) can be seen that the planned slack after the regular opening times is considered as overtime.

5.2.4 Job Shop Scheduling Problem

The job shop scheduling problem is also a well-known special case of the RCPSP. In this problem, there are n jobs that need to visit various machines in a predetermined sequence or route. Different models are proposed for this problem, as different situations can be distinguished. It can for example be that a job may visit every machine at most once, whereas in other cases, a job may visit the machines more often (Pinedo, 2005). The single machine model is the simplest model for this problem. Here, only one machine is available for the execution of the operations. The parallel machine model can be seen as a flexible job shop with only one work centre. The flexible job shop is a generalisation of the basic job shop. The flexible job shop has several work centres, each containing a number of parallel identical machines. This allows for the processing of operations on any machine out of a set of identical machines (Pinedo, 2005). Different objective functions can be used to solve the job shop scheduling problem, such as:

- Minimise the makespan
  - $\min C_{max}$
- Minimise the total weighted completion time
  - $\min \sum w_j C_j$
- Minimise the maximum lateness
  - $\min L_{max}$
- Minimise the total tardiness
  - $\min \sum T_j$

5.2.4.1 Mathematical Programming Formulation

Pinedo (2005) proposes a mathematical programming formulation for the job shop scheduling problem, where n jobs and m machines are considered. Operation (i,j) is the operation where job j has to be processed on machine i. The processing time is denoted by p_{ij}. Each job has to be processed on multiple machines. The order of the machines is given and there is no recirculation. The objective is to minimise the makespan, which is denoted by C_{max}.

The starting time of operation (i,j) is denoted by the variable y_{ij}. The set N contains all operations (i,j) and the set A contains all precedence relations $(i,j) \rightarrow (h,j)$, where job j has to be processed on machine i before it can be processed on machine h. The mathematical programming formulation of the above-described problem is as follows:

$$minimise\ C_{max} \tag{5.29}$$

Subject to

$$y_{hj} - y_{ij} \geq p_{ij}, \qquad \forall (i,j) \rightarrow (h,j) \in A \tag{5.30}$$

$$C_{max} - y_{ij} \geq p_{ij}, \qquad \forall (i,j) \in N \tag{5.31}$$

$$y_{ij} - y_{ik} \geq p_{ik}\ or\ y_{ik} - y_{ij} \geq p_{ij}, \qquad \forall (i,k), \forall (i,j), \forall\, i \tag{5.32}$$

$$y_{ij} \geq 0, \qquad \forall (i,j) \in N \tag{5.33}$$

The objective (5.29) minimises the makespan. Constraints (5.30) are precedence constraints. They ensure that job j is processed on machine i first and then on machine h. The maximum makespan is defined with constraints (5.31). These constraints ensure that the operations can be processed completely between the starting time and the maximum makespan, and thus set the maximum makespan. Constraints (5.32) are called disjunctive constraints. With these constraints, it is ensured that an order is generated among the operations that have to be processed on the same machine. Constraints (5.33) ensure that the variables y_{ij} are positively valued.

The job shop scheduling problem is NP-hard and is therefore hard to solve to optimality, especially when the number of jobs increases. Many heuristics are proposed in the literature to solve the problem, such as the branch and bound approach (Schutten, 2019), the shifting bottleneck heuristic (Schutten, n.d.; Pinedo, 2005), the neighbourhood-based genetic algorithm (Nouri, Belkahla Driss, & Ghédira, 2018), local search methods (Nouri, Belkahla Driss, & Ghédira, 2018) and the adaptive algorithm (Gholami & Sotskov, 2014).

5.3 Conclusion

Resources are rarely unlimited within organisations. This is also the case at PREI. Resource allocation deals with the question of how the resources can be shared among the different processes. Scheduling is closely related to resource allocation. The main goal of scheduling is to create a schedule that indicates which activity should be executed at what time and where. The scheduling problem is an optimisation problem within operations research. In the literature, models are introduced that represent practical scheduling problems. The objective is the performance measure of the schedule and can be expressed in different ways. Also, different kinds of constraints have to be considered, such as time constraints, capacity constraints and technological constraints.

The RCPSP is introduced. This problem describes how the resources should be assigned to the activities, optimising a defined objective. The RCPSP is a combinatorial optimisation problem and is NP-hard. Many well-known optimisation problems are special cases of the RCPSP. The RCPSP describes many scheduling problems that are, at some points, similar to the distinguished scheduling problem at PREI. Therefore, the RCPSP is used as the beginning point to solve the scheduling problem in this research.

Four cases of the RCPSP are introduced: the basic model for the RCPSP, the MRCPSP, the OR scheduling problem and the job shop scheduling problem. All problems are elaborated, including their parameters and variables. Also, mathematical programming formulations are given. Table 5.1 includes an overview of the discussed models and why these models are useful in the remainder of this research.

Table 5.1: Overview discussed cases RCPSP

Model	Purpose	Why is this model useful?
Basic RCPSP	Determine the finishing times of all activities	Many real-world problems are covered with this model. Also, the notation of many parameters and variables can be used. This model determines the finishing times of the activities. However, we do not use this formulation for the PREI scheduling problem, as it is more important to know the starting times of the activities.
MRCPSP	Extension of the RCPSP, so modes can be assigned to activities	The addition of the modes in this model results in an extra index for the decision variable. We can use this notation to be able to assign the activities to a workplace. An important remark is that this model determines the finishing times of the activities, so this has to be taken into account when using the model.
OR Scheduling Problem	Assign an operating day, starting hour and operating room to all surgeries	This model is based on the MRCPSP, and is useful in different ways. At first, the resource constraints of this model can be used. Also, the model assigns a starting time to the surgeries. So, these notations and formulations can be useful. Another aspect we can use is the addition of planned slack to the schedule.
Job Shop Scheduling Problem	Assign a starting time to all operations, where an operation is defined as a job that has to be processed on a certain machine	In literature, various objectives are defined for this model, that can be useful for this research. Also, weights are introduced to indicate the importance of activities.

6 Proposed Scheduling Model

In this chapter, we propose a model for the PREI scheduling problem. The model is based on the literature review in Chapter 5. We start with the integer linear programming in Section 6.1. This section includes the problem description and an elaboration of the mathematical formulation. In Section 6.2 we introduce a second objective that can be suitable for the problem. We also propose some additions on the introduced model in Section 6.3. The collection of inputs is discussed in Section 6.4 and the chapter is concluded in Section 6.5.

6.1 Integer Linear Programming

The PREI scheduling problem is considered and modelled as an ILP. The model should generate a schedule, where a workplace and starting time is assigned to all activities. The MRCPSP is used as the starting point of this model, where the modes are exchanged for the workplaces of PREI. Also, there are no precedence relations at the lab, so the precedence constraints are not included. In the remainder of this section, the different aspects of the model are elaborated. An overview of the model can be found in Appendix D.

6.1.1 Assumptions

During the modelling of the problem, some assumptions are made:

1. Human resources are unlimited. It is assumed that the schedule is not influenced by human resources and that there are always enough employees available for the execution of the activities.
2. The activities are continuously executed. Once the activities are started, they are not interrupted.
3. The processing times of the activities are deterministic and known in advance. Also, any setup times and completion times are included in or added to the processing times.
4. A working week consists of five working days. Weekends, holidays and days off where the whole lab is closed are not included.
5. There are no precedence relations. All activities can be executed individually, there are no relations between activities.

6.1.2 Problem Description

The model should generate a schedule for all activities $j \in J$. The activities are assigned to a workplace w (or w'). There are eight workplaces in the lab: seven regular workplaces and one demonstration workplace. We also introduce a fictional workplace to schedule activities such as service windows that do not require a workplace. When there is no fictional workplace, some activities cannot be scheduled, or they reserve a workplace while they do not need it. The set of workplaces is $W = \{0, \dots, 8\}$, where $w = 8$ is the fictional workplace. The binary parameter z_j is introduced to indicate whether activity j requires the fictional workplace (so, the activity does not require a real workplace), where:

$$z_j = \begin{cases} 1, & \text{if activity } j \text{ requires the fictional workplace} \\ 0, & \text{otherwise.} \end{cases}$$

The workplaces are designed in such a way that usually only one (fictional or real) workplace is required for the activities. However, some activities do require two workplaces. The binary parameter α_j is introduced to indicate whether activity j requires two workplaces or not, where:

$$\alpha_j = \begin{cases} 1, & \text{if activity } j \text{ requires two workplaces} \\ 0, & \text{if activity } j \text{ requires one (fictional) workplace.} \end{cases}$$

The regular workplaces are all situated in rooms in the lab. Some rooms are larger than the others. There is also one room where two workplaces are situated. Some activities that require two workplaces prefer to be scheduled in this room ($s_j = 1$), so everything happens in only one room. However, some activities (e.g. trainings) require separate rooms ($s_j = 0$). To indicate whether activity j is preferably scheduled in the 'double' room based on the type of activity or not, the binary parameter s_j is introduced, where:

$$s_j = \begin{cases} 1, & \mathit{if\ activity\ j\ is\ preferably\ scheduled\ in\ the\ room\ with\ two\ workplaces} \\ 0, & \mathit{if\ activity\ j\ requires\ seperate\ rooms}. \end{cases}$$

We already mentioned that some rooms are larger than the others. The group sizes of the activities vary as well. A large group cannot perform its activity in the smallest room. A distinction is made, where very large groups (more than six people) require the 'double' room (the room with two workplaces), large groups (four up to six people) require at least a large room, and small groups (up to three people) can use all rooms. The group size of activity j is denoted by g_j. Based on the group size g_j it is determined which rooms are suitable for the activity. To ensure that the activities are assigned to suitable rooms based on the group sizes, we introduce two binary parameters $double_j$ and $large_j$, where:

$$double_j = \begin{cases} 1, & \mathit{if\ activity\ j\ requires\ the\ room\ with\ two\ workplaces\ (double\ room)} \\ 0, & \mathit{otherwise}. \end{cases}$$

$$large_j = \begin{cases} 1, & \mathit{if\ activity\ j\ requires\ at\ least\ a\ large\ room} \\ 0, & \mathit{otherwise}. \end{cases}$$

The demonstration workplace is indicated with $w = 7$. It is assumed that all demonstrations are executed at the demonstration workplace and that other activities are preferably not. The binary parameter o_j is introduced to help modelling this, where:

$$o_j = \begin{cases} 1, & \mathit{if\ activity\ j\ is\ a\ demonstration} \\ 0, & \mathit{otherwise}. \end{cases}$$

A line is denoted by ℓ and the set of all lines that are present at the lab is denoted by L. The activities at PREI are executed on a predefined line. Some activities do not require a line, but it is also possible that an activity requires more than one line. The binary parameter $b_{j\ell}$ is introduced to indicate whether line ℓ is required for activity j, where:

$$b_{j\ell} = \begin{cases} 1, & \mathit{if\ activity\ j\ requires\ line\ \ell} \\ 0, & \mathit{otherwise}. \end{cases}$$

The hand-held terminals (HHTs) and Post21 Functionality are the resources at the lab. These resources are renewable (see Section 5.2.1), as the availability is constant at every time period. The resources are denoted by k, and the set of resources by K. The availability of the resources is denoted by R_k. An activity j can require resources. The required number of resources k for activity j is denoted by the parameter r_{jk}.

The time horizon of the schedule is denoted by T, and consists of time periods t (or t'). The time periods are set to half working days for this problem. It is assumed that a working week contains five working days. The processing time of activity j is denoted by p_j and is expressed in a multiple of the defined time period. The testing and training activities can be expressed in working days, but demonstrations do require only two hours. However, we assume that a demonstration does require half a day because of for example group movements. When we express the time periods in working days, the problem cannot be formulated as an ILP anymore, as the processing times of the demonstrations are no integers (0.5 time periods).

Expressing the time periods as half working days does not cause any problems in this research. However, when the definition does cause problems, the scheduling problem can be formulated as a Mixed Integer Linear Programming (MILP) model. Then, not all parameters and variables have to be expressed as integers. For example, the starting times and processing times can be continuous. One drawback is that the starting times are determined by the model, and can therefore get undesired values, such as 4.45. This can be solved by defining the variable properly and adding constraints to ensure for example that demonstrations can start in the morning and the afternoon ($t = 1, 1.5, 2, \ldots$), and other activities only in the morning ($t = 1, 2, 3, \ldots$).

The release date $\underline{\delta}_j$ of activity j is the time period in which the activity may start and is expressed in a multiple of the defined time period. The due date $\overline{\delta}_j$ is the time period in which the activity should be finished and is also expressed in a multiple of the defined time period.

6.1.2.1 Overview Sets, Indices and Parameters

By now, different sets, indices and parameters are defined. An overview of the sets and indices can be found in Table 6.1.

Table 6.1: Overview of sets and indices ILP

Set	Description	Index	Description
J	All activities	j	Activity
W	All workplaces	w, w'	Workplace
L	All present lines	ℓ	Line
K	All resources	k	Resource
T	Time horizon	t, t'	Time period

We defined integer and binary parameters. The integer parameters are:

- Processing time of activity j — p_j
- Group size of activity j — g_j
- Availability of resources type k — R_k
- Number of resources type k needed for activity j — r_{jk}
- Release date of activity j — $\underline{\delta}_j$
- Due date of activity j — $\overline{\delta}_j$

The binary parameters are:

- $z_j = \begin{cases} 1, & \text{if activity } j \text{ requires the fictional workplace} \\ 0, & \text{otherwise.} \end{cases}$
- $\alpha_j = \begin{cases} 1, & \text{if activity } j \text{ requires two workplaces} \\ 0, & \text{if activity } j \text{ requires one workplace.} \end{cases}$
- $s_j = \begin{cases} 1, & \text{if activity } j \text{ is preferably scheduled in the room with two workplaces} \\ 0, & \text{if activity } j \text{ requires seperate rooms.} \end{cases}$
- $double_j = \begin{cases} 1, & \text{if activity } j \text{ requires the room with two workplaces (double room)} \\ 0, & \text{otherwise.} \end{cases}$
- $large_j = \begin{cases} 1, & \text{if activity } j \text{ requires at least a large room} \\ 0, & \text{otherwise.} \end{cases}$
- $o_j = \begin{cases} 1, & \text{if activity } j \text{ is a demonstration} \\ 0, & \text{otherwise.} \end{cases}$
- $b_{j\ell} = \begin{cases} 1, & \text{if activity } j \text{ requires line } \ell \\ 0, & \text{otherwise.} \end{cases}$

6.1.3 Decision Variables

The model has to assign a workplace and a starting time to all activities. These assignments will form the schedule. The decision variable x_{jwt} is introduced to be able to assign workplaces and starting times to all activities, where:

$$x_{jwt} = \begin{cases} 1, & if\ activity\ j\ is\ executed\ at\ workplace\ w\ and\ started\ at\ time\ t \\ 0, & otherwise. \end{cases}$$

The maximum lateness of the schedule is denoted by the variable L_{max}.

6.1.4 Objective Function

Different parties consult the lab for their activities. Most activities performed at the lab are performed for those parties. The activities are often part of a project. Usually, a due date is agreed on with the stakeholders. It is therefore desired for ProRail that the activities are finished before the due date. When that is the case, one of the suggestions in literature is to minimise the maximum lateness of the schedule, denoted by L_{max}. The objective is then:

$$\min L_{max} \tag{6.1}$$

6.1.5 Constraints

Various constraints are necessary to generate feasible schedules. In this section, the constraints are introduced. We introduce constraints regarding the workplaces, lines, resources and release dates, and constraints to define the maximum lateness.

6.1.5.1 Workplaces

An activity is defined as a test, demonstration or training that has to be scheduled and that requires a workplace in the lab. We also assumed that the availability of the resources is the same for every time period. However, we already know that, for example, the Post21 Functionality is not always available due to service windows. This is an 'activity' that does not require a workplace. We introduced the binary parameter z_j, where $z_j = 1$ means that activity j does not require a real workplace and $z_j = 0$ means that activity j does require a real workplace. We add constraints (6.2) to ensure that all activities that do not require a workplace, are assigned to the fictive workplace ($w = 8$).

$$\sum_t x_{j,8,t} = z_j, \qquad \forall j \tag{6.2}$$

In the general RCPSP literature, all activities are scheduled exactly once. However, we have to adjust these constraints a bit, as there are activities at PREI that require two workplaces. We want to schedule these activities twice, as we assign the activities to a workplace and a time period. We use the introduced parameter α_j that indicates whether an activity requires one ($\alpha_j = 0$) or two ($\alpha_j = 1$) workplaces. This results in constraints (6.3), that ensure that all activities j are scheduled exactly as often as the number of required workplaces.

$$\sum_w \sum_t x_{jwt} = (\alpha_j + 1), \qquad \forall j \tag{6.3}$$

We also want to ensure that the activity uses two workplaces at the same time, and thus start twice, at the same time period but different workplaces. This is ensured with constraints (6.4). Here we state that when activity j is scheduled at a given workplace w and at a given time period t, we want that for this activity and at this time, the sum of the scheduled activities over all workplaces is two. This ensures that the activity is always scheduled twice at the same time period when necessary. When an activity does require only one workplace, $\alpha_j = 0$ and so these constraints do not hold.

$$\alpha_j \cdot \sum_{w'} x_{jw't} \geq 2 \cdot \alpha_j \cdot x_{jwt}, \qquad \forall j, w, t \tag{6.4}$$

We also need to ensure that a workplace can only be in use by one activity at the same time. It is for example not possible for activity 2 to start at $t = 2$ when activity 1 started at $t = 1$ and is still in progress. This occurs at the OR scheduling problem as well, as an OR can only be used for one surgery at the same time. The constraints suggested by the literature are used to introduce the constraints for this scheduling problem. This results in constraints (6.5). These constraints ensure that there is no overlap between the activities on a workplace. However, overlap is allowed on the fictive workplace, as this workplace does not exist. Therefore, these constraints do not have to hold for $w = 8$. The set of real (existing) workplaces is denoted by E, where $E = \{0, \ldots, 7\}$.

$$\sum_j \sum_{t'=\max(0, t-p_j+1)}^{t} x_{jwt'} \leq 1, \qquad \forall w \in E, t \tag{6.5}$$

We want the demonstrations to be executed at the representative demonstration workplace ($w = 7$). We introduced the parameter o_j that indicates whether activity j is a demonstration $(o_j = 1)$ or not $(o_j = 0)$. When the activity is not a demonstration, it may be scheduled on this workplace, but that is not necessary. This results in constraints (6.6).

$$\sum_t x_{j,7,t} \geq o_j, \qquad \forall j \tag{6.6}$$

The activities other than demonstrations are preferably executed at the general workplaces. As it is a preference, we can add this as a so-called soft constraint. We can add soft constraints to the model by adding a penalty to the objective: when an activity that is not a demonstration is scheduled on the demonstration workplace, a penalty is added to the objective value. The penalty is calculated with constraints (6.7).

$$Penalty_1 = \sum_j (1 - o_j) \cdot \sum_t x_{j,7,t} \tag{6.7}$$

Now, the activities other than demonstrations that are scheduled on the demonstration workplace ($w = 7$) are penalised. We add this penalty to the objective function, but we want to minimise the maximum lateness (L_{max}) as well. It is important to define what is more important: minimising L_{max} or that the penalty is minimised. To indicate the importance of the terms in the objective function, a weight can be added to the penalty: β_1. This results in objective function (6.8). When the weight is increased, the penalty becomes more important, and vice versa. The weight of the penalty still has to be determined.

$$\min L_{max} + \beta_1 \cdot Penalty_1 \tag{6.8}$$

Now, all demonstrations are scheduled at the demonstration workplace, and the other activities are preferably scheduled at the general workplaces. But the general workplaces have differences as well. There are for example two workplaces situated in one room at PREI, and the other workplaces are situated individually in a room, and the sizes of the individual rooms differ as well. The characteristics of the activity influence the most suitable workplace. Some activities that require two workplaces are preferably performed in the room with the two workplaces, whereas others (e.g. trainings) do require two different rooms. Also, the group size of the involved people in the activity is important to determine which room is suitable. Large groups do for example not fit in the small rooms. Table 6.2 gives an overview of the room types. Also, a reference is made to the room numbers as used at PREI and the (preferred) activities for these room types are defined.

Table 6.2: Room types with indication which activities should be scheduled or are preferred

Room type	Room number PREI	w	(Preferred) activities
Large	A301 A302	0 1	Activities with medium groups (4-6 persons)
Double	A305a A305b	2 3	Activities with large groups (> 6 persons) Some activities require two workplaces, whereas others must be scheduled to two different rooms
Small	A307 A308 A310	4 5 6	Activities with small groups (1-3 persons)
Demon-stration	Demo	7	Demonstrations
Fictive	-	8	The activities that do not require a workplace

The set W is already introduced as the set of all workplaces. We introduce four subsets of W that we need for the model. The subsets can be found in Table 6.3.

Table 6.3: Subsets of W

Set	Workplaces	Description
E	0, 1, 2, 3, 4, 5, 6, 7	All real, existing workplaces
S	0, 1, 4, 5, 6	All workplaces in small and large rooms
D	2, 3	All workplaces in the double room
L	2, 3, 4, 5, 6	All workplaces in the large and double rooms

As already stated, some of the activities that require two workplaces must be scheduled in two different rooms. This is a hard constraint, as it is required that two different rooms are assigned. The parameter s_j is used to indicate whether the activity must be scheduled on workplaces in two different rooms ($s_j = 0$) or that it is preferred that the activity is scheduled on two workplaces in the same room ($s_j = 1$), which is only possible in the double room. Constraints (6.9) are introduced to ensure that when an activity does require two rooms, the activity is at least once assigned to a workplace in a nondouble room, so a small or large room ($w \in S$).

$$(1 - s_j) \cdot \alpha_j \cdot \sum_{w \in S} \sum_{t} x_{jwt} \geq (1 - s_j) \cdot \alpha_j, \qquad \forall j \tag{6.9}$$

Some activities are preferably scheduled on two workplaces in the same room (room A305, $w = 2\ and\ 3$). We add these soft constraints as a penalty, just as we did before in constraints (6.7). This results in constraints (6.10) to calculate the penalty and an update of the objective as displayed with (6.11).

$$Penalty_2 = \sum_j s_j \cdot \alpha_j \cdot \sum_{w \in S} \sum_t x_{jwt} \tag{6.10}$$

$$\min L_{max} + \beta_1 \cdot Penalty_1 + \beta_2 \cdot Penalty_2 \tag{6.11}$$

There is one group of constraints left regarding the workplaces: we do not look at the group sizes of the activities yet. We already distinguished that the small rooms are not suitable for large groups. To indicate whether the group needs the double room (group size of more than 6 people) or at least a large room (group size of four up to six people), we use the introduced binary parameters $double_j$ and $large_j$. To ensure that the activities where $double_j = 1$ are scheduled on the workplaces in the double room, we introduce constraints (6.12). We see that these constraints only hold when the activity is no demonstration ($o_j = 0$), the activity requires a workplace ($z_j = 0$) and the activity requires a workplace in the double room ($double_j = 1$).

$$\begin{aligned}(1 - o_j) \cdot (1 - z_j) \cdot double_j \cdot \sum_{w \in D} \sum_t x_{jwt} \geq \\ (1 - o_j) \cdot (1 - z_j) \cdot double_j, \qquad \forall j\end{aligned} \tag{6.12}$$

For activities with big group sizes ($large_j = 1$), at least a large room is needed. The workplaces of the double room are also suitable. This results in constraints (6.13).

$$\begin{aligned}(1 - o_j) \cdot (1 - z_j) \cdot large_j \cdot \sum_{w \in L} \sum_t x_{jwt} \geq \\ (1 - o_j) \cdot (1 - z_j) \cdot large_j, \qquad \forall j\end{aligned} \tag{6.13}$$

6.1.5.2 Lines

All present lines at the lab can only be used for one activity at the same time. These constraints look like constraints (6.5) where we can only use one workplace at the same time. However, we first have to indicate whether the activity requires the specific line or not. When we look for example at the Hanzelijn, only one activity that requires the Hanzelijn ($b_{j,Hanze} = 1$) can be executed at the same time. When the activity does not require the line, it can be scheduled at the same time (note: other constraints may still hold!).

As we discussed, we ensure that activities that require two workplaces are scheduled twice (constraints (6.3)). But we stated above that only one activity can use a line at the same time. This means that the activities that require two workplaces cannot be scheduled, as the model thinks that the line is used twice (although this is not the case). This is solved by dividing the result by $(a_j + 1)$.

Constraints (6.14) are introduced to ensure that every line can be used for at most one activity at the same time. When an activity requires two workplaces ($\alpha_j = 1$), a line can be assigned to two combinations of activity, workplace and time (x_{jwt}), when it is the same activity j.

$$\sum_j \frac{b_{j\ell} \cdot \sum_{t'=t-p_j+1}^{t} \sum_w x_{jwt'}}{a_j + 1} \leq 1, \qquad \forall t, \ell \tag{6.14}$$

6.1.5.3 Resources

The resources are the HHTs and Post21 Functionality. The constraints regarding the resources look like constraints (6.14). The differences are that we now need to know how much resources are required by the activity (r_{jk}) instead of just an indication of whether the resource is needed or not, and that the availability of the resources is resource specific. In the current situation, for example, there are three HHTs and the Post21 Functionality can be used only once. However, this may change in the future. Constraints (6.15) are introduced to ensure that for every time period no more resources are used than the maximum capacity of the resources. When the resource availability changes in the future, it will be easy to change the model as only the parameter R_k has to be changed.

$$\sum_{j} \frac{r_{jk} \cdot \sum_{t'=t-p_j+1}^{t} \sum_{w} x_{jwt'}}{a_j + 1} \leq R_k, \qquad \forall t, k \tag{6.15}$$

6.1.5.4 Release Dates and Maximum Lateness

The activities can only start after they are released. Hartmann (1999) introduced constraints concerning the release dates and due dates as an addition to the basic model. However, they define the decision variable as the finishing time of the activity, whereas we assign a starting time to the activities. Therefore, the constraints have to be adjusted. Hartmann (1999) introduced the constraints concerning the release dates where the finishing time minus the processing time (which is the starting time) should be equal to or higher than the release date. The processing times are not included in our constraints, as the starting time is already suggested by the decision variable. Also, we added an index to the decision variable as we assign a workplace to the activities. The starting times of the activities that require two workplaces will be counted twice, so we add the term $(\alpha_j + 1)$ as we did before. This results in constraints (6.16).

$$\sum_{w} \sum_{t} x_{jwt} \cdot t \geq \underline{\delta}_j (\alpha_j + 1), \qquad \forall j \tag{6.16}$$

We do not have constraints regarding the due dates of the activities. However, we do have one hard due date, as the time horizon T is set. We assign a starting time to all activities that lies within this time horizon. So, all activities should start within this time horizon. When that is not possible, the solution will be infeasible. The due dates of the activities are part of the objective function, and we want to minimise the exceedance of these due dates. In literature, the lateness of an activity j is defined in two different ways:

- L_{max} can only be positive:
  - When an activity is finished before its due date, $L_{max} = 0$,
  - When an activity is finished after its due date, L_{max} is the finishing time minus the due date,
- L_{max} can be both positive and negative:
  - L_{max} is always the finishing time minus the due date.

We decide to use the second definition, so L_{max} can be both positive and negative. Therefore, for this problem, L_{max} is defined as the starting time plus the processing time, minus the due date. The maximum lateness of the schedule is calculated with constraints (6.17), where the lateness of all activities is calculated and L_{max} is set as the largest lateness.

$$L_{max} \geq x_{jwt} \cdot t + p_j - \overline{\delta}_j, \qquad \forall j, w, t \tag{6.17}$$

When we add hard constraints on the due dates of the activities (e.g. the due dates can never be exceeded), L_{max} and thus that part of the objective will always be zero and the solution can be infeasible more easily. When we decide to add the due dates as hard constraints nevertheless, a change in the objective is required (e.g. minimising the penalties only as L_{max} will always be zero) and we need to add the possibility of overtime to prevent infeasible solutions. However, overtime is not desired, so we decide not to implement this.

6.2 Second Objective

As we already discussed, most activities involve various parties and a due date of the activities can be agreed on. It is desired that the activities are finished before this due date. In the proposed model, the objective is to minimise the maximum lateness. Another possible objective that is discussed in literature and suitable for this problem is to minimise the total tardiness, where the tardiness of activity j is defined as:

$$T_j = \max\left\{0,\ \frac{\left(\sum_w \sum_t x_{jwt} \cdot t\right)}{\left(\alpha_j + 1\right)} + p_j - \overline{\delta}_j\right\} \tag{6.18}$$

The tardiness of the activity is always positive and indicates the lateness of the activity. When the activity is finished before its due date, the tardiness is zero. The total tardiness of the schedule is defined as the sum of the tardiness of all activities: $\sum_j T_j$. Formula (6.19) shows the objective function when the objective is to minimise the total tardiness.

$$\min \sum_j T_j + \beta_1 \cdot Penalty_1 + \beta_2 \cdot Penalty_2 \tag{6.19}$$

Constraints (6.17) of the model that determine L_{max} become obsolete and should be replaced by constraints that determine the tardiness T_j of every activity j. These constraints look like the constraints of L_{max}, but now we determine a value for every activity j instead of the maximum value. This results in constraints (6.20), where $T_j \in \{0, 1, \ldots\}$.

$$T_j \geq \frac{\left(\sum_w \sum_t x_{jwt} \cdot t\right)}{\left(\alpha_j + 1\right)} + p_j - \overline{\delta}_j, \qquad \forall j \tag{6.20}$$

6.3 Proposed Additions

In this section, two additions to the model are proposed: dealing with uncertainty in processing times and prioritising activities. The additions are elaborated, including a way of adding them to the model. It is also indicated why these additions cannot be added to the model yet and what is necessary to be able to do so.

6.3.1 Dealing with Uncertainty

In the proposed model, it is assumed that the processing times of the activities are deterministic and known in advance. However, there is some uncertainty in the processing times. The processing times do include the setup times, and the setup times can vary as discussed earlier in Section 2.3.2.2. The connection between the OBU and the TLC is not always easy and can take some time. Also, uncertainty can be distinguished in the processing times themselves. It can be that problems are experienced during the execution of an activity. Or, that the activity is completed earlier than expected. This results in uncertainty in the processing times. This is not included in the proposed model.

When we look at the OR scheduling problem as introduced in Chapter 5, we see that Hans et al. (2008) add slack to the planned surgeries to make the schedules more robust. Slack can also be added to the planned activities at PREI to make the schedule more robust. Hans et al. (2008) determine the slack based on the planned surgeries of a specific specialty on a specific day. For PREI, the slack for every activity has to be determined. The planning horizon is way bigger than one day and when we determine the slack over the whole planning horizon and add the slack at the end of the time horizon, this will still result in many activities that cannot be started or finished at the desired time.

When we include slack in the proposed model, the outcomes will be more robust against the uncertainties in the processing times. To be able to include slack, statistical data of the durations should be collected first. It is important to group the activities based on their characteristics. Here, we propose a more detailed grouping than just ESC test, other test, training or demonstration. The ESC tests can, for example, be divided into groups based on the line and the suppliers of the needed resources.

Once historical statistical data is collected, slack can be added to the model. Some adjustments have to be made to the model. The slack is based on the expected variance of the processing times of the activities. The mean μ_j and the standard deviation σ_j (which is the square-root of the variance) of all activities j can be determined based on the statistical data. Then, the planned slack size θ_j can be calculated:

$$\theta_j = \beta \cdot \sigma_j \tag{6.21}$$

β $(\beta \geq 0)$ influences the probability that the activities are completed on time. We assume that the processing times are normally distributed. With $\beta = 0.5$, this results in a probability of 69.15% that the activity will be finished on time.

Inspired by Hans et al. (2008), we change the processing time to the mean of the activity plus the planned slack size $(\mu_i + \theta_j)$ to add the planned slack size to the schedule. So, we schedule extra time for the activities to include some of the uncertainty of the processing times. This will result in more robust schedules. To do so, we need to change constraints (6.17) of the proposed model. Constraints (6.17) should be changed to:

$$L_{max} \geq x_{jwt} \cdot t + \left(\mu_i + \theta_j\right) - \overline{\delta}_j, \qquad \forall j, w, t \tag{6.22}$$

When the objective is to minimise the total tardiness, constraints (6.23) should be added to the model, together with constraints (6.21), and the constraints for L_{max} become obsolete as already discussed.

$$T_j \geq \frac{\left(\sum_w \sum_t x_{jwt} \cdot t\right)}{\left(\alpha_j + 1\right)} + (\mu_i + \theta_j) - \overline{\delta}_j, \qquad \forall j \tag{6.23}$$

6.3.2 Prioritising Activities

In the proposed model, it is assumed that the importance of all activities is equal. However, we can distinguish activities that are of greater importance than others. A demonstration for the minister is, for example, of greater importance than a demonstration for a school class. Also, an activity that is mentioned to PREI far in advance can get priority over an activity that is mentioned last minute. However, there are no policies yet that indicate which activity is more important than others. To be able to add priorities to the model, such a policy has to be composed first.

We advise making a scale to indicate the importance of the activities. It can for example be decided to use a scale from 1 to 10. The activities with the greatest importance should always get the highest weight γ_j (e.g. 1 is not important and 10 is of great importance). However, there is no priority policy yet, so it is not yet possible to determine an appropriate scale. When there is a policy, it can become clear that there are for example six different priorities. Then, the scale should be from 1 to 6. The activity with the highest priority should always get the highest weight γ_j, which is 6 in this example. Also, the scale should start at 1. When an activity has weight $\gamma_j = 0$, it does not matter for the model that the activity is completed after its due date, while that is never desired.

In the literature, we already introduced the objective to minimise the total weighted completion time: $\min \sum w_j C_j$. We see that when we multiply the completion time C_j of the activity with its weight w_j, we get the weighted completion time of activity j. However, the objective of the proposed model is to minimise the maximum lateness L_{max}. It makes no sense to multiply this objective with the weights of the activities, as the objective is the maximum lateness of all activities. Therefore, we advise adding the weights (γ_j as we already denote the workplaces by w) in constraints (6.17) where we determine L_{max}. This results in constraints (6.24). We multiply the lateness of the activity by the weight of the activity.

$$L_{max} \geq \gamma_j \cdot \left(x_{jwt} \cdot t + p_j - \overline{\delta}_j\right), \qquad \forall j, w, t \tag{6.24}$$

When the objective is to minimise the total tardiness, the weight can be added to the objective function in the way we saw in the literature. This results in the objective function (6.25).

$$\min \sum_j \gamma_j \cdot T_j + \beta_1 \cdot Penalty_1 + \beta_2 \cdot Penalty_2 \tag{6.25}$$

6.3.3 Both Additions

It is also possible to include both proposed additions in the model. The constraints that determine the maximum lateness L_{max} should then be adjusted to constraints (6.26) and the additional constraints (6.21) should be added to calculate the planned slack size θ_j.

$$L_{max} \geq \gamma_j \cdot \left(x_{jwt} \cdot t + (\mu_i + \theta_j) - \overline{\delta}_j\right), \qquad \forall j, w, t \tag{6.26}$$

When the objective is to minimise the total tardiness, constraints (6.23) should still be added, together with constraints (6.21). Also, objective function (6.25) should be used.

6.4 Collection of Inputs

The first set of inputs was already created for the toy-sized problem. This data can be used to test the model, but is not a good representation of real life, as the data set is too small and not all parameters are present.

Inputs are created when an activity is introduced. The parameters of every activity can be determined. A standard form (e.g. in Microsoft Excel) can be used to collect all parameters. This results in a list of activities with their parameters. However, at this point, there is no historical data available, and the lab is not fully operational yet. Therefore, the list of activities (data set) will be more of a prediction. ProRail is consulted to draw up a list of activities that is based on future (known) activities.

6.5 Conclusion

We proposed an ILP to solve the scheduling problem at PREI. The objective is to minimise the maximum lateness of the activities, and different constraints are introduced that apply to the problem at the lab. We also propose a second objective of minimising the total tardiness, and two additions to the model. First, we proposed an addition to deal with uncertainties in the processing times. The second addition is proposed to be able to prioritise the activities.

7 Model Performance

In this chapter, we test and validate the proposed model for the scheduling problem. At first, we discuss the setup we use to test the proposed model in Section 7.1. The objective function of the model contains two weighted penalties. Before we can start the experiments to test the model, we have to determine the weights. This is done in Section 7.2. We compose several experiments in Section 7.3, and the performance of the model is discussed in Section 7.4. The chapter is concluded in Section 7.5.

7.1 Setup Model Testing

The proposed model is programmed in the Python 3 language. Different packages are used:

- Itertools,
- Math,
- Matplotlib.patches,
- Matplotlib.pyplot,
- Mip,
- Numpy,
- Pandas,
- Time.

We choose to program the proposed model in the Python language, as this is an open-source language. The solver we use is the Gurobi solver. Gurobi is known for solving (mixed) integer linear programs and is free for academic use. All results are collected with the use of the same computer with an Intel Core i7-8550 CPU and 8GB RAM.

7.2 Determine Importance of Soft Constraints

We determined two soft constraints in Section 6.1.5, and added these constraints to the model as penalties. When the preference is not adhered to, a penalty is added to the objective value. We also introduced weights for the penalties, β_1 and β_2, to indicate how important the penalties, and thus the soft constraints, are. However, the weights of the penalties are not determined yet.

For the activities at PREI, the most important thing is that they are completed on time. Most activities are part of a project or other schedule. Therefore, the activities should be completed within the agreed time window. The soft constraints are composed as model preferences, and we do not give these preferences a high priority. Also, both preferences are equally important, which means that the weights will be equal.

It is acceptable for an activity to have a delay of one working day if that results in an inclusion of the preferences. The time periods we use in the model are half working days, which means that an activity may be delayed by two time periods. Therefore, we set the weights β_1 and β_2 to $\beta_1 = \beta_2 = 2$, so the preferences are included in the schedule when that results in a delay of not more than two time periods (one working day). When the inclusion of the preferences results in larger delays, the preferences are not included.

7.3 Composed Experiments

We compose 7 experiments to test the model and its robustness. The experiments are done with the following data sets, where the expansions (data sets 3-7) are expansions of the provided data set (data set 2) and based on possible growth scenarios of PREI:

1. **Toy-sized data.** This data set is introduced in Section 2.5.1 and contains fictive activities.
2. **Provided data.** This data set is composed by ProRail. An overview of the activities can be found in Appendix E.
3. **Expansion 1: Duplicate All Tests.** This data set is the first expansion of the provided data set. We duplicate all tests (ESC tests and other tests), and change the release dates and due dates.
4. **Expansion 2: Duplicate All Tests and Demonstrations.** This data set is the second expansion of the provided data set, where we duplicate the tests and demonstrations and change their release dates and due dates.
5. **Expansion 3: Duplicate All Tests and Trainings.** This data set contains duplicates of the tests and trainings compared to the provided data set. Note: it is not the most likely scenario that the trainings are expanded.
6. **Expansion 4: Duplicate All Tests, Demonstrations and Trainings.** In this data set, we duplicate all tests, demonstrations and trainings, compared to the provided data set.
7. **Expansion 5: Duplicate 'Test Other' Activities and Add 'Test ESC' Activities Three Times.** This expansion is also based on the provided data set. We expect more ESC tests in the future, so the 'Test Other' activities are still only duplicated, whereas the 'Test ESC' activities are added three times to the data set. The release dates and due dates are changed again.

Every data set is used for one experiment. The characteristics of the data sets of the different experiments can be found in Table 7.1. We observe that the number of activities depends on the experiment.

Also, the time horizons are not always the same. We assume that a working week contains five working days and we set the time periods in all experiments to half working days. In experiment 1, the activities should be performed in a time horizon of eight working weeks, as defined in Section 2.5.1. This results in 80 time periods, where $T = \{1, 2, \dots, 80\}$ and $t \in T$ is the beginning of a time period. The length of the time window of the other experiments is half a year, as this is in line with the by ProRail provided list of activities. All activities should be completed within a time horizon of half a year, from 1 January 2021 until 1 July 2021. The time horizon consists of 258 time periods, where $T = \{1, 2, \dots, 258\}$.

Another difference can be seen in the used workplaces. In Section 2.5.1, only workplaces one, two, three and the demonstration workplace were used. This will be the same in experiment 1. The other experiments use all workplaces, including the fictional workplace as introduced in Section 6.1.

In Table 7.1, we see that there are some unknown parameters. In all data sets, the planned slack and the weights of the activities are unknown. Therefore, these parameters are not included in the results. In the first experiment, the group sizes, availability of the resources and the number of resources needed for all activities is unknown. These parameters are therefore not included in the results of this experiment. Next to that, there are no activities included in this data set that require two workplaces, so in this experiment, we cannot conclude whether the model is able to hold the constraints regarding the activities that require two workplaces.

Table 7.1: Characteristics of the data sets of the seven experiments

Exp.	Name	Acti-vities	Time horizon	Used workpl.	Unknown parameters
1	Toy-sized	10	$T = \{1, 2, \ldots, 80\}$	1, 2, 3, demo	Group size (g_j) Availability of resources (R_k) Number of resources needed (r_{jk}) Planned slack (θ_j) Weights of activities (γ_j)
2	Provided data	35	$T = \{1, 2, \ldots, 258\}$	All	Planned slack (θ_j) Weights of activities (γ_j)
3	Expansion 1: Duplicate All Tests	51	$T = \{1, 2, \ldots, 258\}$	All	Planned slack (θ_j) Weights of activities (γ_j)
4	Expansion 2: Duplicate All Tests and Demonstrations	57	$T = \{1, 2, \ldots, 258\}$	All	Planned slack (θ_j) Weights of activities (γ_j)
5	Expansion 3: Duplicate All Tests and Trainings	58	$T = \{1, 2, \ldots, 258\}$	All	Planned slack (θ_j) Weights of activities (γ_j)
6	Expansion 4: Duplicate All Tests, Demonstrations and Trainings	64	$T = \{1, 2, \ldots, 258\}$	All	Planned slack (θ_j) Weights of activities (γ_j)
7	Expansion 5: Duplicate 'Test Other' Activities and Add 'Test ESC' Activities Three Times	63	$T = \{1, 2, \ldots, 258\}$	All	Planned slack (θ_j) Weights of activities (γ_j)

For every experiment, the model is run with both distinguished objectives: minimising the maximum lateness and minimising the total tardiness, to determine which objective is most suitable. In every experiment, we first run the model for 300 seconds (five minutes). When the model is not able to find an optimal solution, we run the model for 3600 seconds (one hour), and eventually for 7200 seconds (two hours). For every run, we determine the optimisation status, possible gap, objective value, maximum lateness, total tardiness, penalties and computation time. Based on the found solution, an operational schedule and a resource allocation graph is composed. The operational schedule shows which activity starts at what time and at what workplace. We can say that this graph visualises the decisions the model makes. The second graph shows the resource allocation. We display per resource when that resource is used, by what activity and on which workplace. This graph visualises the occupation of the resources, which gives an overview of the usage of the resources. A possible bottleneck can be determined with this overview.

7.3.1 Explanation of the Graphs

All operational schedules (see for example Figure 7.1) in this research are constructed in the same way. The time horizon can be found on the x-axis and the workplaces on the y-axis. We use the PREI names of the workplaces. The activities are added as bars. The length of the bar indicates the processing time of the activity: the left side of the bar indicates the starting time and the right side the finishing time. The number of the activity can be found at the left of the bar. The colour indicates which line is used for the activity, as can be seen in the legend.

The resource allocation graphs (see for example Figure 7.3) are constructed as follows. The time horizon can be found on the x-axis again, but the y-axis is different compared to the operational schedule. We want to see the resource allocation in this graph, so we include all resources on the y-axis. The activities are again indicated with bars. The colours now indicate which workplace is used, rather than which line (as the line is also one of the resources).

7.4 Results of the Experiments

In this section, the results are discussed of the 7 experiments as defined in Section 7.3. All computation times are rounded to three decimal places.

7.4.1 Toy-Sized Data

The PREI scheduling problem is introduced with a toy-sized data set in Section 2.5.1. In this experiment, we test the model with the same data set. Table 7.2 includes the results. We see that the model is able to find optimal solutions in both cases, in a very short amount of time. For the objective maximum lateness, the operational schedule is depicted in Figure 7.1 and the resource allocation in Figure 7.3. The operational schedule of the total tardiness is depicted in Figure 7.2 and the resource allocation in Figure 7.4.

Table 7.2: Outcomes experiment 1 (toy-sized data set)

Objective	Maximum lateness	Total tardiness
Optimisation status	Optimal	Optimal
Objective value	0	0
Maximum lateness	0	0
Total tardiness	0	0
Penalty 1	0	0
Penalty 2	0	0
Computation time	0.424 seconds	0.360 seconds

In the operational schedules (Figure 7.1 and Figure 7.2), we see that all activities are started after their release dates and completed before their due dates. The objective value, maximum lateness and total tardiness are therefore zero in both cases. We also see that all activities that are no demonstration are scheduled at another workplace than the demonstration workplace. Therefore, penalty 1 is zero in both cases. The data set of this experiment does not include activities that require two workplaces, causing penalty 2 to be automatically zero. In Table 7.2, we see that only the computation times differ. The model is slightly faster when the objective is to minimise the total tardiness. This is probably because there are more possible solutions with the objective maximum lateness.

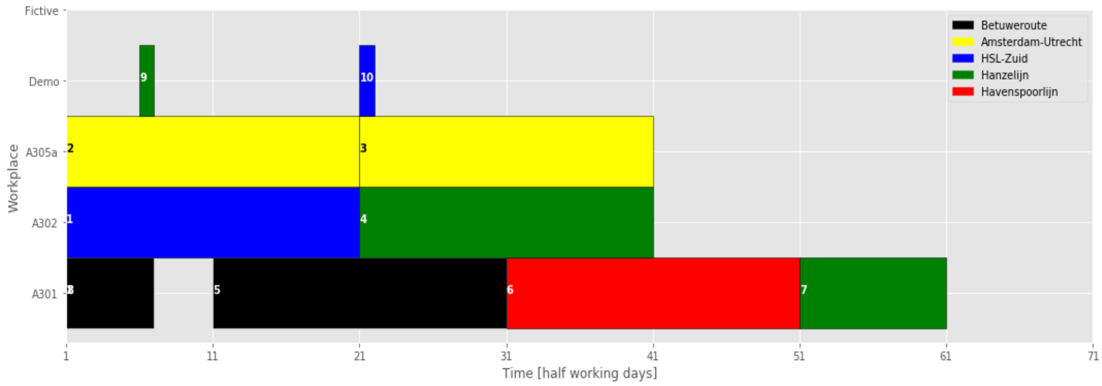

Figure 7.1: Operational schedule with toy-sized data set and the objective to minimise the maximum lateness

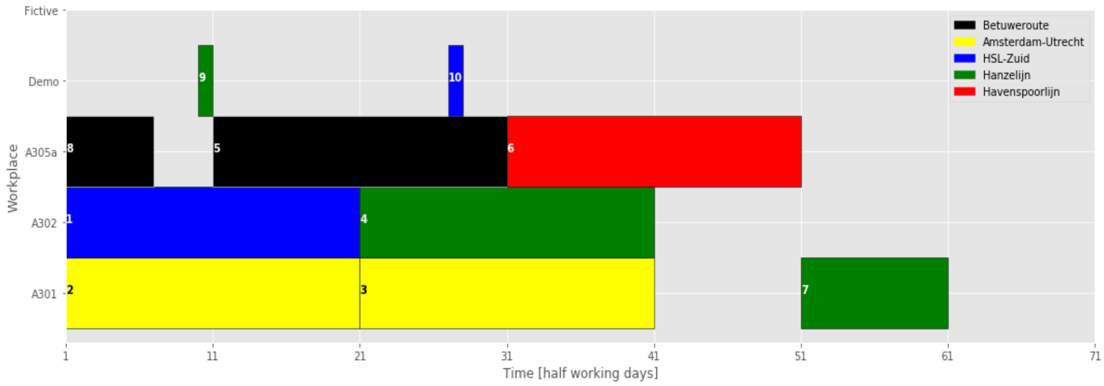

Figure 7.2: Operational schedule with toy-sized data set and the objective to minimise the total tardiness

When we look at the operational schedules in Figure 7.1 and Figure 7.2, we see that every workplace is used only by one activity at the same time. There is no overlap in the activities on a workplace. We also observe that every line is used at most once at the same time and that all activities are started after their release dates. Therefore, we can conclude that these constraints are taken into account. When we compare the schedules, we observe small differences. Some activities that are no demonstrations are executed on another workplace, and the demonstrations have other starting times. Therefore, we can conclude that the model made different decisions, but we cannot say which schedule is better, as both objective values are zero. When we compare the operational schedules determined by the model with the operational schedule as introduced in Section 2.5.1, we see that the schedules are not identical. All schedules are based on optimal solutions, but different decisions are made.

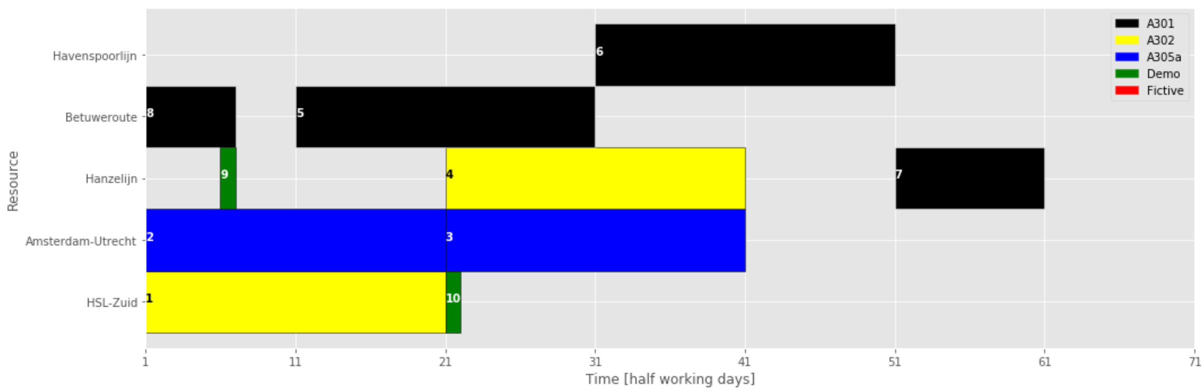

Figure 7.3: Resource allocation with toy-sized data set and the objective to minimise the maximum lateness

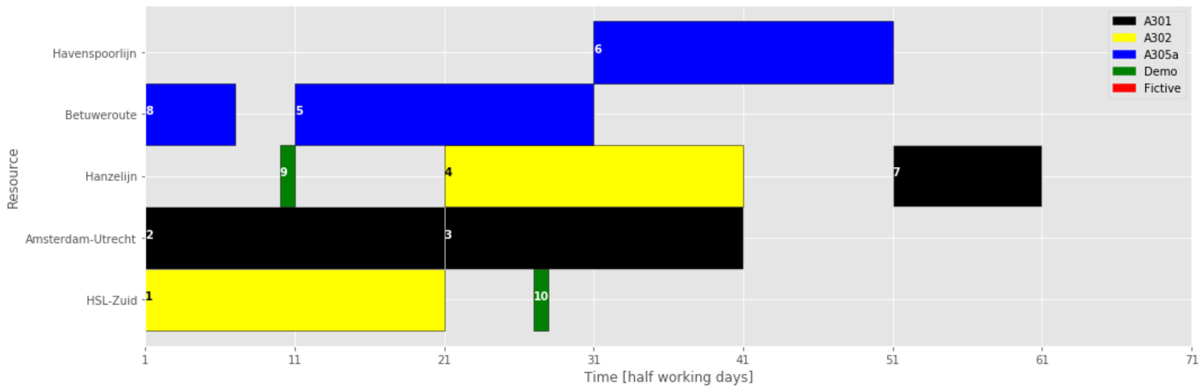

Figure 7.4: Resource allocation with toy-sized data set and the objective to minimise the total tardiness

The resource allocation graphs (Figure 7.3 and Figure 7.4) show that all resources still have some room left for other activities. Therefore, we can say that no resource acts as a bottleneck in this schedule.

7.4.2 Provided Data

The model is able to find optimal solutions in both cases within a short amount of time (< 300 seconds). The results can be found in Table 7.3. For the objective maximum lateness, the operational schedule is depicted in Figure 7.5 and the resource allocation in Figure 7.7. For the total tardiness, the operational schedule is depicted in Figure 7.6 and the resource allocation in Figure 7.8.

Table 7.3: Outcomes experiment 2 (data set provided by ProRail)

Objective	Maximum lateness	Total tardiness
Optimisation status	Optimal	Optimal
Objective value	3	3
Maximum lateness	3	3
Total tardiness	3	3
Penalty 1	0	0
Penalty 2	0	0
Computation time	20.940 seconds	61.749 seconds

When we look at the operational schedules (Figure 7.5 and Figure 7.6), we see that all activities are scheduled and started after their release date. However, we also see that not all activities are completed before their due date. This explains the increase in the maximum lateness and total tardiness. In both cases, the objective value, maximum lateness and total tardiness are 3 time periods (= 1½ working days). This means that all activities are completed not more than 3 time periods after their due date (maximum lateness) and that the tardiness of the whole schedules is 3 time periods as well. We see that this means that only one activity is completed after its due date: activity 32 in both cases. For both objectives, the average tardiness of the activities is $3/35 \approx 0.09 \;\, time\ periods.$

The values of both penalties are zero. The demonstration workplace is only used for demonstrations. Activity 16 is preferably executed in the double room (A305a and A305b). In both Figure 7.5 and Figure 7.6, we see that this is the case. From these figures, we can conclude that the other constraints hold as well. The activities that require two workplaces in two different rooms (activities 23-29) are scheduled in different rooms. Also, the service window activities (activities 1-5) are all scheduled on the fictional workplace, and no other activities are scheduled here.

For the other constraints, we look at the resource allocation graphs in Figure 7.7 and Figure 7.8. We observe that there is no overlap in the activities for any resource. Therefore, we can conclude the constraints regarding the lines and resources hold as well. We do see that the Post21 Functionality resource is often in use. The only activity that is completed after its due date is activity 32. This activity is started at $t = 153$ and requires the Hanzelijn and Post21 Functionality. We see that the activity is started immediately after activity 24 is finished. Activity 24 requires the Hanzelijn and the Post21 Functionality as well. So, activity 32 cannot be started earlier as both the Hanzelijn and the Post21 Functionality are already in use. In this case, the Hanzelijn and the Post21 Functionality are the resources with restricting capacity

7.4.3 Expansion 1: Duplicate All Tests

The results of this experiment can be found in Table 7.4, and the operational schedule and resource allocation graphs in Appendix F. The model is able to schedule the extra tests and found an optimal solution with both objectives within 300 seconds. When we compare the outcomes with the outcomes of the previous data set (the provided data in Section 7.4.2), we see that the objective values, the maximum lateness and the total tardiness are the same in both experiments. With the expanded data set, the model requires some extra computation time to find the optimal solution due to an increase in the number of activities and therefore in the number of possible solutions.

Table 7.4: Outcomes experiment 3 (expansion 1)

Objective	Maximum lateness	Total tardiness
Optimisation status	Optimal	Optimal
Objective value	3	3
Maximum lateness	3	3
Total tardiness	3	3
Penalty 1	0	0
Penalty 2	0	0
Computation time	44.512 seconds	144.857 seconds

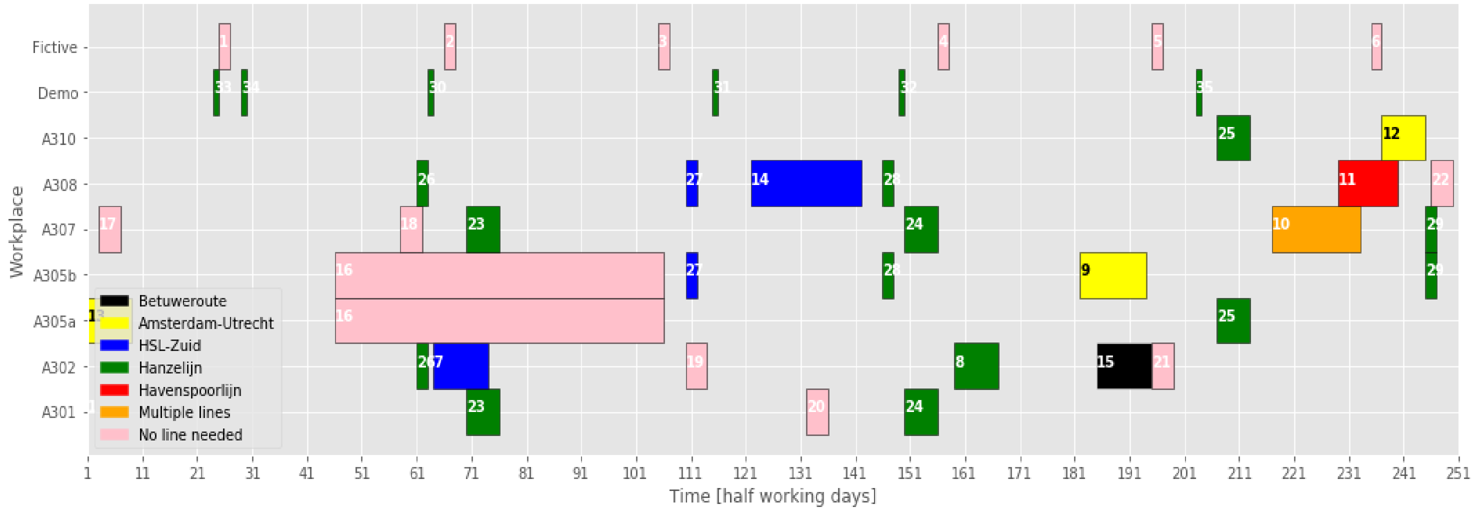

Figure 7.5: Operational schedule with data set ProRail and the objective to minimise the maximum lateness

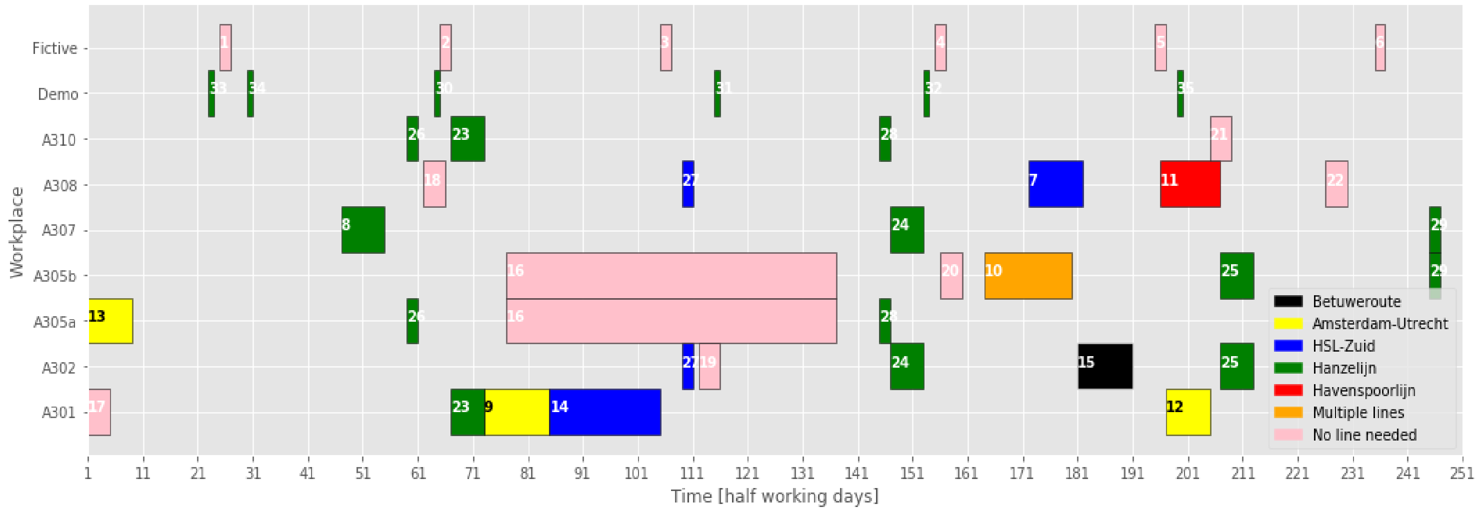

Figure 7.6: Operational schedule with data set ProRail and the objective to minimise the total tardiness

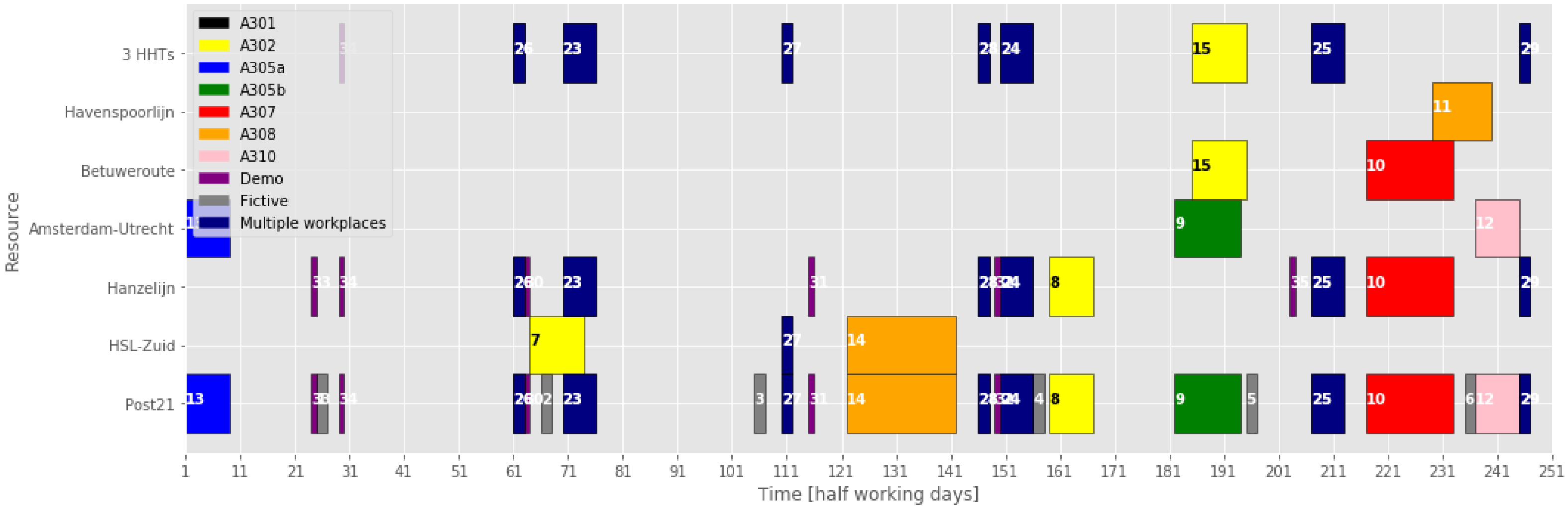

Figure 7.7: Resource allocation with data set ProRail and the objective to minimise the maximum lateness

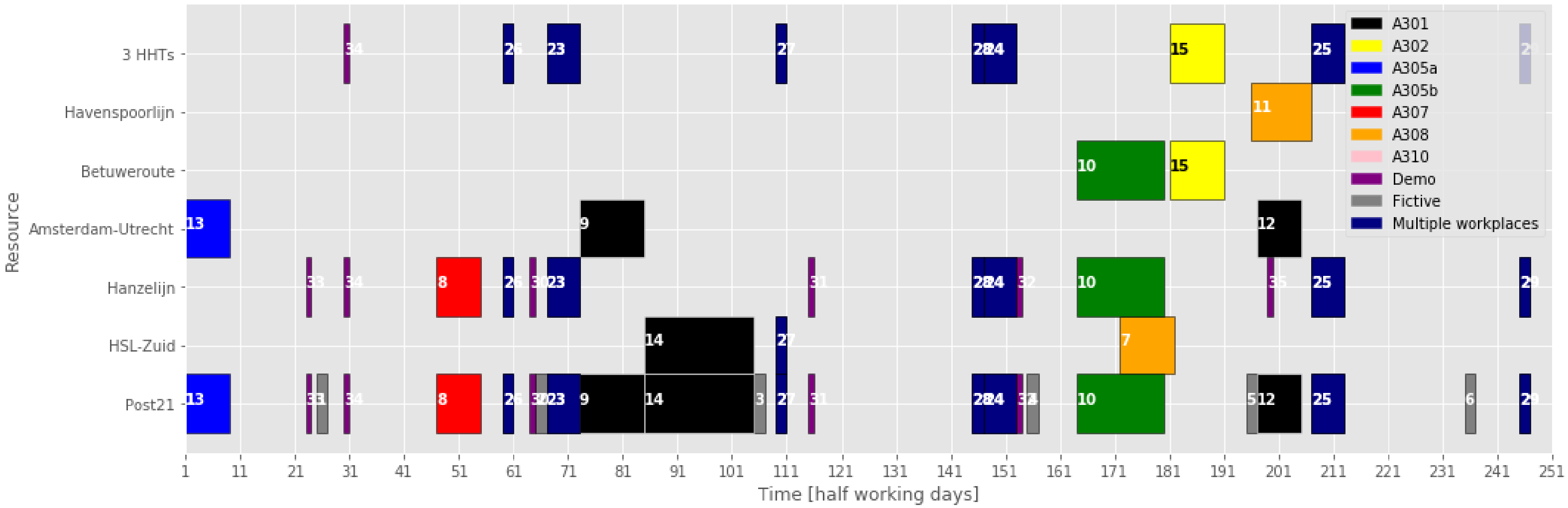

Figure 7.8: Resource allocation with data set ProRail and the objective to minimise the total tardiness

When we compare the results of the different objectives in this experiment, we observe that the objective value, the maximum lateness and the total tardiness is always 3 time periods (average of 0.06 time periods per activity). In both cases, the model was able to find an optimal solution. When we look at the graphs in Appendix F, we see that the schedules differ a bit. However, the schedules are evenly good, as the maximum lateness and total tardiness are in both cases the same. The model is able to find an optimal solution faster with the objective of minimising the maximum lateness. We can explain this by the fact that there are more possible (optimal) solutions when the objective is to minimise the total tardiness.

7.4.4 Expansion 2: Duplicate All Tests and Demonstrations

The results of this experiment can be found in Table 7.5. The operational schedule and resource allocation graphs can again be found in Appendix F. Again, the model is able to find an optimal solution within an acceptable time (less than 300 seconds), and we see that the computation time is higher with the objective of minimising the total tardiness.

In the previous experiment, the objective value, maximum lateness and total tardiness were 3 time periods. In this experiment, we see that this has changed. The maximum lateness is in both cases still 3 time periods, but the total tardiness is increased. We see that in the previous experiment, in both cases, only one activity was completed after its due date. Now, 8 activities are too late in the case of minimising the maximum lateness, and 2 activities (activities 25 and 32) in case of minimising the total tardiness. Activities 25 and 32 both need the Post21 Functionality and the Hanzelijn. These two resources cause the lateness of these activities, as we can see in the graphs in Appendix F, and are therefore the bottlenecks of this schedule. When we compare the computation times of this experiment and the previous experiment, we see an increase of 92.80% and 79.11%, for the maximum lateness and total tardiness, respectively.

Table 7.5: Outcomes experiment 4 (expansion 2)

Objective	Maximum lateness	Total tardiness
Optimisation status	Optimal	Optimal
Objective value	3	6
Maximum lateness	3	3
Total tardiness	14	6
Penalty 1	0	0
Penalty 2	0	0
Computation time	85.819 seconds	259.459 seconds

7.4.5 Expansion 3: Duplicate All Tests and Trainings

After the model was run for 300 seconds, no solution was found in case of minimising the maximum lateness, and a feasible solution was found with a gap of 1.0 (100%) for the total tardiness. Therefore, the model is run with higher maximum run times. The model was able to find optimal solutions in both cases, within about 18 minutes with the maximum lateness and about 1:02 hours with the total tardiness. This is a long time, but still acceptable as the model does not have to be run often as we schedule activities for half a year. The results of the runs in this experiment can be found in Table 7.6, and the operational schedule and resource allocation graphs in Appendix F.

Table 7.6: Outcomes experiment 5 (expansion 3)

Objective	Maximum lateness		Total tardiness		
Max. run time	300 sec.	3600 sec.	300 sec.	3600 sec.	7200 sec.
Optimisation status	No solution found	Optimal	Feasible	Feasible	Optimal
Gap	-	-	1.0 (100%)	0.2333 (23.33%)	-
Objective value	-	7	96	30	30
Maximum lateness	-	7	14	8	8
Total tardiness	-	76	96	30	30
Penalty 1	-	0	0	0	0
Penalty 2	-	0	0	0	0
Computation time	300.112 sec.	1100.370 sec.	300.122 sec.	3600.233 sec.	3727.949 sec.

In both cases, the total tardiness is high and the maximum lateness is relatively low. We see that the objective values, therefore, differ a lot. When we compare the maximum lateness and the total tardiness of both cases, we see that the difference in the maximum lateness is only 1 time period (7 versus 8 time periods), whereas the difference in the total tardiness is 46 time periods (76 versus 30 time periods). The maximum lateness is in both cases acceptable. However, the total tardiness is very high in case of minimising the maximum lateness: 76 time periods (= 38 working days) in total and an average of 1.31 time periods per activity. The total tardiness in case of minimising the total tardiness is acceptable (average of 0.52 time periods per activity). In the graphs in Appendix F, we see that the delays are especially due to tightness in the Post21 capacity.

7.4.6 Expansion 4: Duplicate All Tests, Demonstrations and Trainings

The results of this experiment can be found in Table 7.7, and the operational schedule and resource allocation graphs in Appendix F. Again, we see that the model was not able to find a solution within 300 seconds with the objective maximum lateness. With the objective total tardiness, a feasible solution is found within 300 seconds. When we increase the maximum run time, optimal solutions are found in about 23 minutes and about 1:11 hours, respectively. This is an increase in computation time compared to the previous experiment. We can explain this by the fact that the number of activities is increased, and therefore probably the number of possible solutions.

Table 7.7: Outcomes experiment 6 (expansion 4)

Objective	Maximum lateness		Total tardiness		
Max. run time	300 sec.	3600 sec.	300 sec.	3600 sec.	7200 sec.
Optimisation status	No solution found	Optimal	Feasible	Feasible	Optimal
Gap	-	-	1.0 (100%)	0.6727 (67.27%)	-
Objective value	-	7	102	55	39
Maximum lateness	-	7	59	16	9
Total tardiness	-	125	102	55	39
Penalty 1	-	0	0	0	0
Penalty 2	-	0	0	0	0
Computation time	300.201 sec.	1405.221 sec.	300.395 sec.	3601.330 sec.	4276.254 sec.

With the objective maximum lateness, the objective value is still 7 time periods. The total tardiness increased even further, and is very high with a total tardiness of 125 time periods (= 62½ working days) and an average tardiness of 1.95 time periods per activity. With the objective total tardiness, the objective value and maximum lateness are only slightly higher in this experiment in comparison with the outcomes of the previous experiment, and are still acceptable. We again observe in the graphs in Appendix F that the increase in the objective values, maximum lateness and total tardiness is caused by the tightness in the Post21 Functionality capacity.

7.4.7 Expansion 5: Duplicate ‘Test Other’ Activities and Add ‘Test ESC’ Activities Three Times

In the previous experiment, we scheduled 64 activities. Now, we want to schedule 63 activities, which is one activity less. However, the model is infeasible. In the previous experiments, we saw that especially the Post21 Functionality resource was used very often. Therefore, the capacity of this resource is increased from one unit to two units. The model was able to find optimal solutions in a very short amount of time, with low objective values, maximum lateness and total tardiness. The results of this experiment can be found in Table 7.8, and the operational schedule and resource allocation graphs in Appendix F. In the resource allocation graph we see that some activities are overlapping for the Post21 resource, as this resource is now available twice.

Many ESC testing activities do require the Post21 Functionality (4 out of 6 in the provided data set) for a relatively long period (average processing time is 11 time periods). When we add extra capacity to this resource, an optimal solution is found quickly and the objective value is low. Therefore, we can conclude that the capacity of the Post21 Functionality is a restricting factor to schedule this data set.

Table 7.8: Outcomes experiment 7 (expansion 5)

Objective	Maximum lateness		Total tardiness	
Post21 availability	1	2	1	2
Optimisation status	Infeasible	Optimal	Infeasible	Optimal
Gap	-	-	-	-
Objective value	-	3	-	3
Maximum lateness	-	3	-	3
Total tardiness	-	3	-	3
Penalty 1	-	0	-	0
Penalty 2	-	0	-	0
Computation time	67.141 sec.	69.375 sec.	39.771 sec.	192.592 sec.

7.5 Conclusion

In all experiments, the model was able to find optimal solutions. In experiment 7, a shortage of the Post21 Functionality resource made the model could not find a feasible solution. We increased this resource's capacity to ensure a feasible solution could be found. The results are summarised in Table 7.9. The grey shaded cells are the objective values, as both penalties were zero in all experiments.

The computation time was higher for some experiments than for others. We can explain this by the fact that the number of activities increased, and possibly the number of possible (optimal) solutions as well. A maximum computation time of about 1:11 hours was needed, which is acceptable: the model does not have to be run often as we schedule activities for half a year.

Also, the maximum lateness and total tardiness are acceptable in most cases. The maximum lateness is in all experiments and all cases acceptable: the maximum is 9 time periods, which is 4½ working days. The total tardiness, however, is increasing to a very large number. For example in experiment 6 with the objective maximum lateness, where the total tardiness is 125 time periods. This is 62½ working days or 12½ working weeks, whereas we are scheduling for a time horizon of 258 time periods (= 129 working days). The total tardiness in the same experiment, but with the objective total tardiness, is only 39 time periods (= 19½ working days), which is acceptable. Therefore, we can say that the schedules composed by the model with the objective total tardiness are more suitable for this problem.

Table 7.9: Summary of the results of all experiments, with both objectives. The grey cells are the objective values

Exp.	Number of activities	Objective	Maximum lateness	Total tardiness	Computation time (sec.)
1	10	Maximum lateness	0	0	0.424
		Total tardiness	0	0	0.360
2	35	Maximum lateness	3	3	20.940
		Total tardiness	3	3	61.749
3	51	Maximum lateness	3	3	44.512
		Total tardiness	3	3	144.857
4	57	Maximum lateness	3	14	85.819
		Total tardiness	3	6	259.459
5	58	Maximum lateness	7	76	1100.370
		Total tardiness	8	30	3727.949
6	64	Maximum lateness	7	125	1405.221
		Total tardiness	9	39	4276.254
7	63	Maximum lateness	3	3	69.375
		Total tardiness	3	3	192.592

Although we had to adjust the capacity of one of the resources, we can conclude that the model is able to find good schedules in various growth scenarios, and therefore is suitable to schedule the activities of PREI. The capacities of the resources can be adjusted easily when necessary as these are introduced as parameters. This will, therefore, not cause problems in the future. Based on the outcomes of the experiments (especially experiment 5 and 6), we can say that the most suitable objective for the PREI scheduling problem is to minimise the total tardiness. We saw that the maximum lateness is in all experiments and with both objectives acceptable, but the total tardiness becomes too large when the utilisation of the resources increases and the objective is to minimise the maximum lateness.

The graphs help to visualise the schedule and the occupation of all resources, which can be used to keep an eye on the capacity of the lab and its resources.

An important remark is that there is no historical data present on the processing times and policies regarding priorities are missing. Therefore, the parameters planned slack and priorities of the activities are unknown and are not included in the results.

8 Implementation

In this chapter, the implementation of the proposed model for the scheduling problem is discussed. First, we will determine who should be able to use the model in Section 8.1. Then, it is suggested how the model can be implemented so ProRail can use it in Section 8.2. The requirements for the implementation are discussed in Section 8.3. Here, we look at the requirements on both the harder and the softer side. The chapter is concluded in Section 8.4.

8.1 Users

The purpose of the model is to schedule the activities of the lab. The most important user of the model is the planner of the activities at PREI. The planner (or planners) should be able to use the model, so the schedules for the activities can be composed with the help of the proposed model. It will especially be useful when the workload increases in the lab, as it will be hard to schedule all activities with all constraints when the list of activities becomes longer.

The planner is the person that should be able to use the model by inserting a certain list of activities (the input) and generate schedules (the output). The output can be used by several stakeholders. At first, all employees that are involved in the lab should be able to consult the schedules. In that way, it is clear for everyone which activities are executed at what time and at what workplace. Also, the use of resources is made more transparent. The insights into the use of the resources help employees that are involved in other processes of PREI as well. Think of resource planning and capacity requirements. The schedules can for example help to determine when a resource is used all the time, and the activities start getting delayed.

All involved stakeholders described above are internal stakeholders. External parties, for example the parties involved in the ESC testing process, do not need to consult the schedules. It is important, however, that when an activity is planned, it is communicated to all stakeholders, including external stakeholders.

8.2 Model Implementation

The proposed model is able to generate suitable schedules and is a good starting point for a problem that does not exist yet, but can still be improved. Two additions are already proposed in Section 6.3. Another way to improve the schedules, especially when the number of activities increases, is to use an algorithm or heuristic to compose the schedules. In the current situation, this is not required as we observed during the experiments in Chapter 7 that the proposed model is able to determine optimal solutions within a relatively short amount of time. Another remark is that the model is static. Once a schedule is composed, it is not possible to add extra activities to the schedule without rerunning the model and thus start over with composing a schedule. This can be solved by creating a (more) dynamic model. Further research is necessary on the possibilities.

We programmed the model in the Python language. This is an open-source language, but the software needed is not included in the software package available at ProRail. Adding new software to the package is a long process, with a chance that the software will not be approved. Therefore, we suggest to implement the model differently and use software that is already available. An example is Microsoft Excel. OpenSolver is an open-source optimiser for Microsoft Excel and is able to solve ILPs. Another possibility is to develop new software (inhouse). However, further research is needed on which method is most suitable for PREI.

8.3 Requirements

We can distinguish different requirements, both on the harder side and on the softer side. In the remainder of this section, these requirements are discussed.

8.3.1 Harder Side

The first set of requirements can be distinguished on the harder side. The first requirement is suitable software to implement the model. As already mentioned, it should still be investigated which software is most suitable for PREI to solve the model and generates (visualised) schedules. To be able to run the model once it is implemented, a computer is needed with the right specifications, such as enough memory. Another requirement is a standardised list to collect all parameters of all activities. This will ease the process of collecting the necessary data and the processing of the data as input for the model.

8.3.2 Softer Side

With the requirements on the softer side, we intend the requirements needed on the more human side. The most important requirement on the softer side is the acceptance of the employees. They should work with the proposed model, so they should accept the method. When there is no acceptance, it will be hard to implement a 'new way' of working. Acceptance can be achieved when the 'why' is clear and the method works well.

The proposed model is already introduced to the involved stakeholders. The first responses are positive. Although the problem is not encountered as a problem yet, the planner is already thinking more about the way of scheduling and what is important within the schedules. New insights are gained and the model is a good beginning to solve the future expected problem.

8.4 Conclusion

The most important user of the model is the planner of the activities at PREI, who should be able to generate the schedules. Other employees of ProRail that are involved in the lab should be able to consult the schedules. External parties have no direct interests in the schedules. The model can, however, still be improved. This can especially be important in the future when the number of activities increases. Also, we did experiments in the Python language, but ProRail is not able to use this. Further research is needed to determine a way (e.g. OpenSolver in Microsoft Excel or developing new software) in which ProRail will be able to use the model.

Requirements of the implementation are distinguished on the harder and softer side. On the harder side, suitable software and a computer with the right specifications are necessary to run the model. Also, a standardised list helps to collect all parameters. On the softer side, the most important requirement is acceptance of the employees. Acceptance can be achieved when the 'why' is clear and the method works well.

9 Conclusion and Recommendations

This chapter concludes the research. We start with a discussion in Section 9.1. Limitations of the research are discussed for both parts. The main research question is answered in the conclusion in Section 9.2. At last, some recommendations are provided in Section 9.3.

9.1 Discussion

We made some assumptions and interpretations during the research. In this section, we elaborate on these assumptions and interpretations for both parts of the research.

9.1.1 Process Design

Different processes are distinguished at PREI. Some designs contain more details than others. The limitations that we distinguished are:

- Some processes are designed in detail, where especially the ESC Testing process contains many details. Although the processes and designs are discussed multiple times with most internal stakeholders, it can occur that some steps or explanations are still missing. This will become clear when the processes are implemented.
- Due to the pandemic, all meetings were online. Although not expected, some aspects of the processes or the designs may be missed, especially because it was somewhat harder to communicate.
- Another discussion point is that a list of possibly important processes is composed. This list is composed based on literature and discussed with the stakeholders, but this list may lack some processes that are maybe less important.

9.1.2 Scheduling

The proposed model can be used to schedule a list of activities, whereby an overview of the usage of all resources is provided as well. We were able to generate optimal solutions in reasonable time. Some limitations can be distinguished as well:

- The most important limitation is that the model is static. Once the model is run, the schedule is composed and is fixed for the upcoming time horizon. During the experiments, we chose a time horizon of half a year. However, we know that more activities can be added to the list of activities for this half a year. When the model is run again with the updated list, the results will probably differ a lot compared to the old schedule. This means that the activities are rescheduled, which is not desired.
- One of the assumptions of the model is that all parameters are deterministic and known in advance. However, from practice, we know that there is some uncertainty in especially the processing times of the activities. We already proposed an addition to the model to include this uncertainty, but no data on the involved parameters was available, and therefore the effects of this addition are not included in the results.
- We already discussed that some activities are more important than other activities. We did propose an addition to include the priorities of the activities. In that way, the activities with the highest priorities are more likely to be completed before their due dates. However, the effects of this addition are not included in the results, as a priority policy is still lacking.

- Another limitation is that it is assumed that the human resources of PREI are unlimited. Therefore, human resources are not included in the model. We see in practice, however, that human resources are limited. We also know that some activities require more support, and thus human resources of PREI, compared to other activities. When the human resources were taken into account, the results could differ, as some activities cannot be executed at the same time due to human resources constraints. However, since ESC testing is mainly related to projects, it can be assumed that these projects will deliver additional resources for those projects.

9.2 Conclusion

PREI was established in 2018 and is still in a development phase. The processes are not fully designed yet, which results in a lack of clarity about the lab. Also, the activities of PREI that require resources of the lab are not optimally scheduled. In the current situation, the activities are manually scheduled without a specific method. This is still going well, as the lab capacity is not fully utilised yet. However, it is known that the workload of the lab will increase. It will be harder to schedule all activities manually, as they have many requirements. The goal of this research was to design the processes of the lab and to propose a method to schedule the activities. Therefore, the following main research question was defined in the introduction:

> ***How can the processes of the ERTMS Integration Lab be designed based on a suitable typology, and how can the resources be allocated to the activities under various growth scenarios?***

Various kinds of processes can be distinguished at PREI. Based on literature, we categorised the processes into three types: *management processes*, *core processes* and *supporting processes*. The three defined core processes are testing, demonstrations and trainings. They are designed with in total eight flowcharts including additional remarks. We selected the flowchart method with the addition of swimlanes based on existing literature, the requirements of ProRail and already by PREI designed processes. We also defined fifteen management processes and six supporting processes. These processes are not yet designed in detail, but we proposed a form based on the forms-based approach which can help further specifying the processes. The processes will be implemented at ProRail and included in the quality management system.

We proposed an ILP model to determine how the resources can be allocated to the activities under various growth scenarios. The model is based on the RCPSP. The RCPSP addresses how limited resources should be assigned to activities, optimising a defined objective. Many well-known optimisation problems, such as the OR scheduling problem, are special cases of the RCPSP. We defined two objective functions for our model, based on literature and meetings with ProRail: minimising the maximum lateness and minimising the total tardiness. Various constraints are introduced and taken into account. They define the use of the workplaces, lines and resources, and the release dates of the activities.

We did seven experiments with different data sets. The first experiment is with a toy-sized data set. The time horizon in this experiment has a length of 8 working weeks. The second experiment is done with a data set composed by ProRail. In the following experiments, we expanded the provided data set. The expansions are based on growth scenarios of the lab. The time horizon of experiments 2-7 is set to half a year (1 January 2021 – 30 June 2021). The time periods are defined as half working days. We used the Gurobi solver in Python 3, and a computer with an Intel Core i7-8550 CPU and 8GB RAM. We were able to validate the model and we found optimal solutions. The decisions of the model are visualised in an operational schedule that shows which activity is started at what time and what workplace, and a resource allocation graph that shows the occupation of all resources over time.

The proposed model is able to generate optimal schedules for the PREI scheduling problem, under various growth scenarios. We observe that the computation times increase when more activities are scheduled, but an optimal solution was found within 1½ hours. However, we increased the capacity of the Post21 Functionality from 1 to 2 in experiment 7 to be able to find a solution. This shows this resource is scarce and should be monitored well, as this resource is the bottleneck for the schedules. The maximum lateness and total tardiness of all experiments with both objectives can be found in Table 9.1. We observed and see in Table 9.1 that the maximum lateness is in all experiments and with both objectives acceptable, but the total tardiness becomes too large when the utilisation of the resources increases and the objective is to minimise the maximum lateness. Therefore, we can conclude that the most suitable objective for the PREI scheduling problem is to minimise the total tardiness.

Table 9.1: Maximum lateness (ML) and total tardiness (TT) of all experiments, with both objectives. The objective value is bold

Experiment	1	2	3	4	5	6	7
Data set / **Objective**	Toy-sized	Provided	Tests 2x	Tests and demos 2x	Tests and trainings 2x	Tests, demos and trainings 2x	ESC tests 4x, other tests 2x
Maximum lateness	**ML: 0** TT: 0	**ML: 3** TT: 3	**ML: 3** TT: 3	**ML: 3** TT: 14	**ML: 7** TT: 76	**ML: 7** TT: 125	**ML: 3** TT: 3
Total tardiness	ML: 0 **TT: 0**	ML: 3 **TT: 3**	ML: 3 **TT: 3**	ML: 3 **TT: 6**	ML: 8 **TT: 30**	ML: 9 **TT: 39**	ML: 3 **TT: 3**

In the current situation, the activities are scheduled manually. The experiments are done with data sets that are possible scenarios for the future. Therefore, it is difficult to compare the outcomes of the model with the current situation. We do not know whether the manually composed schedules in the current situation are optimal. We can say that the use of the model eases the scheduling process and generates at least as good schedules, if not better schedules. The planner of PREI supports the method and the results.

The recommended steps for the implementation of the model can be found in the roadmap in Figure 9.1.

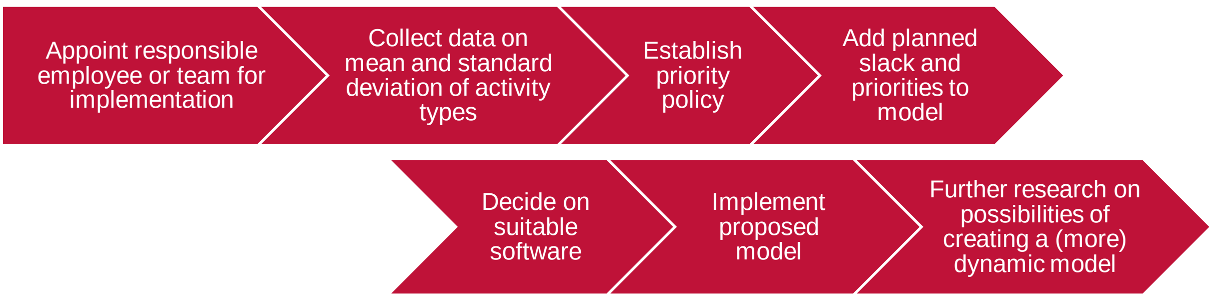

Figure 9.1: Roadmap implementation of the scheduling model

9.3 Recommendations

We have some recommendations for ProRail. First, we elaborate on the recommendations for the process design:

- The first recommendation is to implement the process designs. A roadmap for the implementation can be found in Figure 9.2. The processes should be included in the quality management system of ProRail (KMS). We already mentioned that the management and supporting processes still need to be further specified. We recommend to do this in the implementation phase. It is important that the processes are well elaborated and designed. This will result in a clear way of working and will prevent possible problems.

  Once the processes are added to the KMS, the processes are reviewed every five years. Every five years, it is determined whether the processes are still valid, which is an important task.

- Some of the processes that we designed are not yet fully performed. One example is the ESC testing process, which is a very complex process. We designed the processes with all available information and validated the process many times with some of the stakeholders. However, it can occur that the designs of (parts of) the processes are not completely sound. When that is the case, we recommend changing the design, so the process design does work well.

- It is also possible that the process itself is changing over time. It is important to change the design accordingly. We recommend to do this as fast as the change is noticed. Also, as already discussed, the processes are reviewed every five years. When a change is noticed in any process, the design should be changed as soon as possible.

Figure 9.2: Roadmap implementation of the processes

The proposed model is a good starting point when scheduling the activities will become harder, but can be improved on several aspects. We have some recommendations and suggestions for further research:

- The model is programmed in the Python 3 language, but ProRail is not able to use this language. Therefore, we recommend to determine a way in which ProRail will be able to use the model. First, suitable software should be determined. An example can be OpenSolver in Microsoft Excel, which is an open-source optimiser and can solve ILPs. Another possibility is to develop new software (inhouse). Thereafter, specific requirements should be distinguished, such as computer specifications.

- We proposed two additions for the model in Section 6.3:
  - The first addition is proposed to deal with uncertainty in the processing times. We already indicated in the discussion in Section 9.1 that one of the assumptions of the model is that all parameters are deterministic and known in advance, but that we know that there is some uncertainty in the processing

times. However, we were not able to implement this addition, as there is no historical data available.

The activities should be grouped based on their characteristics. An example of a group can be ESC tests with the use of the line Betuweroute. Also, we recommend to collect more data on the activities in the lab that require a workplace. It is especially important to collect the actual processing times, as this is used to calculate the mean and standard deviation. When sufficient data is collected and the activities are grouped, the mean and standard deviation of the activity group can be calculated, whereafter the slack can be determined. We recommend to add the proposed addition once it is possible to determine the slack.

- The second addition is proposed to be able to prioritise the activities. We know that some activities are more important than other activities. However, as we already discussed in Section 9.1, we were not able to implement this in the model, as there is no priority policy yet. When there is no policy, it is not exactly known which activities are more important than others, and it is also possible that the priorities are estimated differently by various people.

  We recommend to elaborate on a priority policy, and then add the proposed addition to the model. The priority policy should contain a point scale (e.g. 1 to 10) to score the activities, where the most important activity should be weighted with the highest number (e.g. 10).

- In the discussion in Section 9.1, we already stated that the model is static. When a schedule is composed, it is fixed for the upcoming time horizon. The time horizon was set to half a year during the experiments, which means that the schedule is already known for the upcoming half a year. However, new activities can be announced that should be executed within that time period. When the model is run again, the results will probably differ strongly compared to the old schedule, which is not desired.

  We propose some directions for further research to create a (more) dynamic model, so the model will be able to deal with newly announced activities after an initial schedule is already proposed:

  - The first possibility is to shorten the time horizon. When the time horizon is set to for example three months, the probability that a new activity has to be added to the schedule is lowered. However, the effectiveness of this method still has to be examined, as there are also activities that should be scheduled on even shorter notice.
  - Another possibility is implementing a rolling horizon approach, where the input parameters are allowed to update or modify. Therefore, it is possible to optimise the problem with the currently available information. This approach is applied to many scheduling problems with uncertainty. Kopanos & Pistikopoulos (2014) show how this approach can be used.
  - We already introduced a way to prioritise activities. This method can be suitable to make the model more dynamic. When a schedule is already composed, and an extra activity should be added, the already scheduled activities can get a very high weight. Also, the release dates and due dates of the already scheduled activities should be set to the already scheduled starting and ending times. When the model is run with the new list of activities, the model will only reschedule an activity when it is its only option, as the objective will increase

with the weight times the delay of the activity, which is the number of time periods the activity is shifted.

- We also recommend to look into the possibilities of generating schedules with heuristics. We observed that the proposed model is able to generate optimal solutions within short times, but when the list of activities was increased, the computation time increased as well. When it is not possible to find optimal solutions or even feasible solutions within a reasonable time, heuristics are preferred.

  As we start with an empty schedule, we first need a constructive heuristic. Here, activities are repeatedly added to the schedule until a complete solution is constructed. This can be done in different ways, e.g. random or by scheduling the activities with the longest processing times first. Thereafter, an improvement heuristic can be used to improve the initial solution. Examples of heuristics that are already proposed in the literature review in Chapter 5 are simulated annealing, local search and tabu search. We recommend to do further research into these heuristics, and how they can be implemented.

- Based on the current situation, we assumed that activities require one or two workplaces. We introduced the parameter α_j to indicate the number of required workplaces. When activities require more workplaces in the future (three or more), the model is not able to handle this and should be changed. We recommend changing the definition of the binary parameter α_j to:

$$\alpha_j = \begin{cases} 1, & if\ activity\ j\ requires\ two\ or\ more\ workplaces \\ 0, & if\ activity\ j\ requires\ one\ (fictional)\ workplace. \end{cases}$$

  Also, a new integer parameter should be introduced to indicate how many workplaces are needed:

Number of workplaces needed to perform activity j	n_j

  When we look at the overview of the ILP in Appendix D, some constraints should be changed, based on the changes in the parameters:

  - At first, all terms $(\alpha_j + 1)$ should be replaced by n_j to ensure that the number of required workplaces is used. This results in changes in the constraints (D.6), (D.13), (D.14) and (D.15) in the model with the objective maximum lateness, and constraints (D.17), (D.21), (D.28), (D.29) and (D.30) with the objective total tardiness.
  - In the model, we ensure that the activities that require two workplaces are assigned to two workplaces with constraints (D.7) and (D.22). When more workplaces are required, these constraints should be changed as well. The term $2 \cdot \alpha_j$ should be replaced by $n_j \cdot \alpha_j$, so the number of required workplaces is used instead of 2.

10 References

Adenmark, M., Deter, M., & Schulte, T. (2006). Testing Networked ECUs in a HIL Based Integration Lab. *SAE Technical Paper*.

Aguilar-Savén, R. (2004). Business process modelling: Review and framework. *International Journal of Production Economics 90*, 129-149.

Appian. (2021). *What is Process Management?* Retrieved January 7, 2021, from Appian: https://www.appian.com/bpm/process-management/#:~:text=Process%20Management%20refers%20to%20aligning,they%20will%20manage%20processes%20effectively.

Applied Dynamics International, Inc. (2007). *Understanding Advanced System Integration Labs: Simulation-Centric System Integration.* Ann Arbor, USA.

Association for Project Management. (2021). *What is project management?* Retrieved January 7, 2021, from APM: https://www.apm.org.uk/resources/what-is-project-management/#:~:text=Project%20management%20is%20the%20application,a%20finite%20timescale%20and%20budget.

Baker, K. R., & Trietsch, D. (2007). Safe Scheduling. *INFORMS TutORials in Operations Research*, 79-101.

Barone, A. (2020, July 28). *Quality Management*. Retrieved January 11, 2021, from Investopedia: https://www.investopedia.com/terms/q/quality-management.asp#:~:text=Quality%20management%20is%20the%20act,quality%20control%20and%20quality%20improvement.

Boddy, D. (2014). *Management: An Introduction.* Harlow: Pearson.

Brucker, P., Drexl, A., Möhring, R., Neumann, K., & Pesch, E. (1999). Resource-constrained project scheduling: Notation, classification, models, and methods. *European Journal of Operational Research, 112*, 3-41.

Brunaud, B., & Grossmann, I. E. (2017). Perspectives in multilevel decision-making in the process industry. *Frontiers of Engineering Management, 4*(3), 256-270.

Cambridge Dictionary. (2021). *Information*. Retrieved January 11, 2021, from Cambridge Dictionary: https://dictionary.cambridge.org/dictionary/english/information

ConceptDraw. (2020). *Flow Chart Symbols*. Retrieved September 30, 2020, from ConceptDraw: https://www.conceptdraw.com/How-To-Guide/diagram-software-flow-chart-symbols

Dufresne, T., & Martin, J. (2003). Process Modeling for E-Business. *INFS 770 - Methods for Information Systems Engineering: Knowledge Management and E-Business*.

ERA * UNISIG * EEIG ERTMS USERS GROUP. (2016). *Glossary of Terms and Abbreviations.*

ERSA by CLEARSY. (2021). *ETCS Driver Machine Interface*. Retrieved 03 15, 2021, from ERSA: https://ersa.clearsy.com/products/etcs-driver-machine-interface/?lang=en

ERTMS | The European Rail Traffic Management System. (2020). *ERTMS in brief*. Retrieved September 21, 2020, from ERTMS | The European Rail Traffic Management System: https://www.ertms.net/?page_id=40

European Commission. (2020a). *ERTMS - European Rail Traffic Management System*. Retrieved September 8, 2020, from European Commission - Mobility and Transport: https://ec.europa.eu/transport/modes/rail/ertms_en

European Commission. (2020b). *ERTMS - Subsystems and Constituents of the ERTMS*. Retrieved September 22, 2020, from European Commission - Mobility and Transport: https://ec.europa.eu/transport/modes/rail/ertms/what-is-ertms_en

European Commission. (2020c). *ERTMS - What is ERTMS?* Retrieved September 22, 2020, from European Commission - Mobility and Transport: https://ec.europa.eu/transport/modes/rail/ertms/what-is-ertms_en

European Union Agency for Railways. (2017). Training on Interoperability and Safety ERTMS. Retrieved November 30, 2020, from https://www.era.europa.eu/sites/default/files/events-news/docs/2-ertms_presentation_en.pdf

European Union Agency for Railways. (2020). *Guide for the application of the CCS TSI.*

Gholami, O., & Sotskov, Y. N. (2014). Solving parallel machines job-shop scheduling problems by an adaptive algorithm. *International Journal of Production Research, 52*(13), 3888-3904.

Grapholite. (2020). *Swimlanes*. Retrieved September 30, 2020, from Grapholite: https://grapholite.com/diagrams/swimlane

Hans, E. W. (2019). Planning & Scheduling: Lecture 1 and 2.

Hans, E., Wullink, G., van Houdenhoven, M., & Kazemier, G. (2008). Robust surgery loading. *European Journal of Operational Research, 185*(3), 1038-1050.

Hartmann, S. (1999). *Project Scheduling under Limited Resources: Models, Methods, and Applications.* Berlin: Springer.

Haughey, D. (2021). *RACI Matrix*. Retrieved January 7, 2021, from Project Smart: https://www.projectsmart.co.uk/raci-matrix.php

Heerkens, H., & Van Winden, A. (2012). *Geen Probleem - Een aanpak voor alle bedrijfskundige vragen en mysteries.* Nieuwegein: Van Winden Communicatie.

Human Resources Edu. (2021). *What is Human Resource?* Retrieved January 13, 2021, from Human Resources Edu: https://www.humanresourcesedu.org/what-is-human-resources/

IBM. (2021). *IT management*. Retrieved January 13, 2021, from IBM: https://www.ibm.com/topics/it-management

ifex. (2021). *What is an Annual Plan or Workplan?* Retrieved January 07, 2021, from ifex: https://ifex.org/toolkits/organisational-toolkit/strategic-planning-toolkit/what-is-an-annual-plan-or-workplan/

IGI Global. (2020). *What is Function Model*. Retrieved October 27, 2020, from IGI Global: https://www.igi-global.com/dictionary/function-model/74661

International Organization for Standardization. (1985). *ISO 5807:1985 Information processing - Documentation symbols and conventions for data, program and system flowcharts, program network charts and system resources charts.* Geneva: ISO.

International Organization for Standardization. (2018a). *ISO 31000:2018 Risk management.* Geneva: ISO.

International Organization for Standardization. (2018b). *ISO 41001:2018 Facility management - Management systems - Requirements with guidance for use.* Geneva: ISO.

Jannach, D., & Friedrich, G. (2009). Extending the RCPSP for modeling and solving disruption management problems. *Applied Intelligence*.

Juneja, P. (2021). *Financial Management - Meaning, Objectives and Functions*. Retrieved January 7, 2021, from Management Study Guide: https://www.managementstudyguide.com/financial-management.htm

Kinney, G., & Wiruth, A. (1976). Practical Risk Analysis for Safety Management.

Kopanos, G. M., & Pistikopoulos, E. N. (2014). Reactive Scheduling by a Multiparametric Programming Rolling Horizon Framework: A Case of a Network of Combined Heat and Power Units. *Industrial & Engineering Chemistry Research, 53*(11), 4366-4396.

Lansdaal, M., & Lewis, L. (2000, September). Boeing's 777 Systems Integration Lab. *IEEE Instrumentation & Measurement Magazine*, 13-18.

Lister, A. M. (1979). Resource Allocation and Scheduling. *Fundamentals of Operating Systems*, 91-113.

Liu, K., Alderson, A., & Qureshi, Z. (1999). Requirements Recovery from Legacy Systems by Analysing and Modelling Behaviour. *Proceedings IEEE International Conference on Software Maintenance*, 3-12.

Loi, K. (2021). *Direct vs Indirect Procurement: What's the Difference?* Retrieved January 13, 2021, from Procurify: https://blog.procurify.com/2013/09/25/whats-the-difference-between-indirect-and-direct-procurement/#:~:text=Indirect%20procurement%20is%20the%20act,office%20supplies%20or%20acquiring%20services.

Meier, K. (2020, June 18). *A Project Manager's Guide to Resource Planning*. Retrieved January 7, 2021, from Float: https://www.float.com/blog/the-ultimate-guide-to-resource-planning-for-project-management/#:~:text=Resource%20planning%20is%20a%20process,to%20keep%20projects%20on%20budget.

Microsoft. (2020). *SharePoint*. Retrieved December 14, 2020, from Microsoft: https://www.microsoft.com/nl-nl/microsoft-365/sharepoint/collaboration

Ministerie van Infrastructuur en Waterstaat. (2020). *13e voortgangsrapportage van het Programma ERTMS.*

Nouri, H. E., Belkahla Driss, O., & Ghédira, K. (2018). Solving the flexible job shop problem by hybrid metaheuristics-based multiagent model. *Journal of Industrial Engineering International, 14*, 1-14.

NVBS. (2020). *Reizigersvervoer*. Retrieved November 24, 2020, from NVBS: https://www.nvbs.com/kennis/railsystemen-nederland/spoorwegen-reizigersvervoer/

O'Neal, K. (2017, November 22). *What's the Difference Between Information Management and Data Management?* Retrieved January 11, 2021, from First San Francisco Partners: https://www.firstsanfranciscopartners.com/blog/difference-between-information-management-data-management/?cn-reloaded=1

Özdamar, L., & Ulusoy, G. (1995). A Survey on the Resource-Constrained Project Scheduling Problem. *IIE Transactions*(27), 574-586.

Pieters, M. (2019, March 30). Retrieved September 8, 2020, from Nederlands enige Treinbeveiliging-site: http://www.marcrpieters.nl/

Pilskalns, O., Andrews, A., Ghosh, S., & France, R. (2003). Rigorous Testing by Merging Structural and Behavioral UML Representations. *International conference on the unified modeling language*, 234-248.

Pinedo, M. L. (2005). *Planning and Scheduling in Manufacturing and Services.* New York: Springer.

Porter, M. E. (1985). *The Competitive Advantage: Creating and Sustaining Superior Performance.* New York: Free Press.

Programma ERTMS. (2020). Retrieved September 8, 2020, from ERTMS: https://www.ertms-nl.nl/home/default.aspx

ProRail (Director). (2019). *ProRail ERTMS: onze grootste klus ooit* [Motion Picture]. Retrieved from https://www.youtube.com/watch?v=S4Xt2Z8LUgg

ProRail. (2020a). *Onze klanten*. Retrieved November 24, 2020, from ProRail: https://www.prorail.nl/vervoerders/onze-klanten

ProRail. (2020b). *Geschiedenis*. Retrieved September 8, 2020, from ProRail: https://www.prorail.nl/wie-zijn-we/geschiedenis

ProRail. (2020c). Internal Source.

ProRail. (2020d). *Jaarverslag 2019.*

ProRail. (2020e). *Omwonenden*. Retrieved September 8, 2020, from ProRail: https://www.prorail.nl/omwonenden

ProRail. (2020f, July). ProRail ERTMS Integratie Lab - Amersfoort.

RISK-ACADEMY. (2019, April 8). *Finally! An alternative to risk matrices*. Retrieved January 11, 2021, from RISK-ACADEMY Blog: https://riskacademy.blog/finally-an-alternative-to-risk-matrices/

Roland, B., Di Martinelly, C., & Riane, F. (2006). Operating Theatre Optimization: A Resource-Constrained Based Solving Approach. *International Conference on Service Systems and Service Management*, 443-446.

Schuster, H. (2019). *Principles for the demonstration of ETCS system compatibility: Version 1.1.*

Schutten, J. (2019). Job shop scheduling: Branch and bound. Enschede.

Schutten, J. (n.d.). Practical job shop scheduling.

Slack, N., Brandon-Jones, A., & Johnston, R. (2013). *Operations Management.* Harlow: Pearson.

Smartsheet. (2021). *Information Management Strategies: From Punch Cards to Data Warehouses, and Looking to the Future with Big Data and AI*. Retrieved January 11, 2021, from Smartsheet: https://www.smartsheet.com/information-management

Theisens, H. C., Harborne, F., & Verreijt, R. (2018). *Lean Six Sigma Green Belt.* Amstelveen: Lean Six Sigma Academy.

von Rosing, M., Kemp, N., Hove, M., & Ross, J. W. (2015). Process Tagging - A Process Classification and Categorization Concept. *The Complete Business Process Handbook: Body of Knowledge From Process Modeling to BPM*(1), 123-171.

Weske, M. (2012). *Business Process Management: Concepts, Languages, Architectures.* Springer.

Wikipedia. (2020). *Gantt chart*. Retrieved October 27, 2020, from Wikipedia: https://en.wikipedia.org/wiki/Gantt_chart

Zijm, W. (2000). Towards intelligent manufacturing planning and control systems. *OR-Spektrum*(22), 313-345.

Zwanenburg, M. I. (2018). *Opzetten methodiek rondom Kritieke Prestatie Indicatoren.*

Appendix A Form Annual Planning

This appendix includes a completed form for the management process: annual planning.

Table A.1: Example completed form for the process annual planning

Process group	Lab management
Process name	Annual planning
Last modified	25-01-2021
Description	Compose an annual planning. This planning contains the goals and objectives for PREI (ProRail ERTMS Integration Lab), and a detailed plan of which activities must be accomplished, when and by whom.

Process manager	Manager PREI
Responsibilities	• Planning is drawn up annually with all those involved. • Monitoring the progress of the activities in the planning over the year. • Planning is analysed and discussed at the end of the year.

Input	Strategy PREI (long-term goals)
Output and results	An annual planning with goals, activities, activity managers, time indication for activities and required resources. Insights into the goals and activities.

Relations with other processes	The annual planning is related to many other processes. However, this depends mainly on the specific elaboration of the planning. For example, the annual planning can be related to the process *resource planning* when resources are discussed in the annual planning. It is important to keep relations with the other processes in mind. Composing a list of related processes when the annual planning is made, can help to give an overview of the processes.

Details	The annual planning should be in line with the long-term goals and objectives (strategic). The necessary activities that contribute to achieving the goals and objectives are recorded. It is also recorded who is responsible for the activities, when the activity takes place and what resources are needed. Think of systems, and human, financial and facility resources. The planning is monitored throughout the year. When the year is over, the schedule is analysed.

Appendix B Roles and Responsibilities ESC Testing Process

The following roles and their responsibilities are distinguished for the ESC testing process:

ERA – European Union Agency for Railways.

The ERA should:

- Support the process in case of IOP issues revealed during a Test Campaign or disagreement on the test conclusions (Issue Handling).

ESC Test Applicant – Or Entity applying for ESC Demonstrations. The party that initiates the ESC Test Campaign. This will typically but not necessarily be the OBU Supplier (e.g. vehicle manufacturer, railway undertaking, Infrastructure Manager, vehicle owner).

The ESC Test Applicant/Entity applying for ESC Demonstration should:

- Ask ProRail Vehicle Authorisation to initiate/organise a Test Campaign according to the terms laid down in this document.
- Compose a list of required ESC Checks.
- Appoint the ESC Test Manager in consultation with the ProRail ESC Manager.
- Provide the certified OBU hardware and software with the corresponding ETCS Baseline and preferably in a Representative Configuration during a defined test period according to the exact conditions defined in the process as described in Section 4.2.1.1 and equip the OBU Test Bench for tests in laboratory environments with interfaces corresponding to the technical concept describe in the Subsets (e.g. agreed specification for the interface between OBU Test Bench and ESC Test Facility).
- Inform ProRail Vehicle Authorisation of all constraints, limitations, (non)implemented CRs, error corrections and the associated onboard behaviour.
- Support the activities according to Section 4.2.1.1 in terms of lab integration, test execution, maintenance of the OBU Test Bench and analysis of findings during the test period.
    - And ensure the Notified Body is hired.

ESC Test Facility Manager – The employee of PREI who is responsible for (the management of) the test facility.

The ESC Test Facility Manager should:

- Be responsible for managing the ESC Test Facility.
- Provide the ESC Test Facility for ESC Tests.
- Ensure in case of lab tests the integration of the OBU Test Bench with the ESC Test Facility and coordinate the provision corresponding to the technical concept described in the interface specifications (e.g. agreed specification for the interface between OBU Test Bench and ESC Test Facility).

- Operate the ESC Test Facility during the activities according to Section 4.2.1.1 and ensure a smooth running of the Test Campaign.
- Support the process in case of Test facility or OBU Test Bench issues revealed during a Test Campaign, if necessary (Issue Handling).
- Ensure the maintenance/update of the ESC Test Facility.

ESC Test Manager – The party who is responsible for managing the Test Campaign. Can be someone from ProRail or an external person.

The ESC Test Manager should:

- Be competent for the job.
- Not be influenced by the Involved Parties in his decision making.
- Organise and lead the execution of and act as 'master' for the activities according to Section 4.2.1.1, whereas the OBU acts as 'slave'.
- Ensure the OBU Supplier is involved.
- Ensure the ESC Tester is appointed and informed.
- Provide a Check Report prepared according to the process described in Section 4.2.1.1.
- Agree with the OBU Supplier, the PREI (Account) Manager, the ProRail ESC Manager and if required the Trackside Supplier on the content of the preliminary test report and issue it.

ESC Tester – The party that is involved in the execution of the test campaign. The ESC Tester is appointed and informed by the ESC Test Manager. Can be someone from ProRail or an external person.

The ESC Tester should:

- Be appointed and informed by the ESC Test Manager.
- Execute the Test Campaign according to the test plan and record the findings.

Notified Body (NoBo) – A body that has been notified by a Member State of the European Union to be responsible for assessing the conformity or suitability for use of the interoperability constituents or for appraising the 'EC' procedure for verification of the subsystems.

The roles and responsibilities of the Notified Body related to ESC are included in Section 6.3.3.1 of the CCS TSI. The Notified Body should check:

- That the technical compatibility checks have been performed under the technical document published by the Agency.
- Based on the Check Report, that the technical compatibility checks results indicate all the incompatibilities and errors encountered during the technical compatibility checks.

OBU Supplier – A party responsible for the design and implementation of the OBU.

The OBU Supplier should:

- Support the Entity applying for ESC Demonstration by providing mandatory evidence, such as the EC declaration of conformity and the limitations against the requirements of the CCS TSI.
- Define a Representative Configuration, if ESC Checks are performed on the product level.
- Agree with the ESC Test Manager, the PREI (Account) Manager, the ProRail ESC Manager and if required the Trackside Supplier on the content of the preliminary test report and issue it.
- Support the process in case of OBU issues revealed during a Test Campaign (Issue Handling).
- Support the process in case of Test facility or OBU Test Bench issues revealed during a Test Campaign, if necessary (Issue Handling).

Planner – The employee of PREI who is responsible for the schedule of the lab.

The Planner should:

- Schedule the test campaign, taking the required resources into account.
- Schedule re-execution of tests if necessary, taking the required resources into account.

PREI (Account) Manager – The employee of PREI who the contact person for all parties for all PREI related affairs is. This person takes care of all the coordination that is required on the ProRail side.

The PREI (Account) Manager should:

- Be the contact person for all parties and operations where PREI is involved.
- Coordinate all that is required on the ProRail side (PREI related affairs).
- Agree with the OBU Supplier, the ESC Test Manager, the ProRail ESC Manager and if required the Trackside Supplier on the preliminary test report (testing part prepared by the ESC Test Manager).
- Lead the process in case of Trackside Implementation issues revealed during a Test Campaign (Issue Handling). Other parties may be consulted to solve the issues.
- If ESC Checks encompass ESC Tests:
    - Make sure that an ESC Test Facility representing their reference trackside for the ESC Types is available and accessible to an Entity applying for ESC Demonstration upon request for ESC Tests and compliant with the conditions of the Principles (e.g. processes, technical compliance to agreed specification for the interface between OBU Test Bench and ESC Test Facility in case of laboratory environments, e.g. Subset-110/111/112).
    - Deliver the necessary information to the ESC Test Facility Manager to continuously maintain the ESC Test Facility according to the ESC Types and its modifications.

    - Make the specific commercial and technical conditions for access to the ESC Test Facility for their ESC Types publicly available.
    - Make sure that the ESC Tests are performed in a non-discriminatory manner; priority rules for parallel demands should be defined on a case by case basis, involving all Entities for ETCS System Compatibility Testing, who initiated the parallel demands and the concerned ESC Test Facility Managers.

PREI Test & Support – The employee of PREI that can execute tests in the lab.

PREI Test & Support should:

- Witness and facilitate tests to the test plan and record the observations in a test observations report.
- Solve small issues revealed during a Test Campaign. These issues are solvable on demand (e.g. another setting).

ProRail ESC Manager – The employee of ProRail who the contact person for all parties for the Test Campaign is. This person takes care of all the coordination that is required on the ProRail side.

The ProRail ESC Manager should:

- Be the contact person for all parties and operations of the Test Campaign.
- Coordinate all that is required on the ProRail side.
- Agree with the OBU Supplier, the ESC Test Facility Manager, the PREI (Account) Manager and if required the Trackside Supplier on the preliminary test report (testing part prepared by the ESC Test Manager). Other parties within ProRail may be consulted for this (e.g. ProRail EG ERTMS).
- Hand-over the final ESC Check Report to ProRail Vehicle Authorisation once the report is finished, and contact the ESC Test Manager when the ESC Check Report is not approved.
- Appoint the ESC Test Manager in consultation with the ESC Test Applicant.
- If ESC Checks encompass ESC Tests:
    - Appoint the PREI (Account) Manager.

ProRail EG ERTMS – The ERTMS expert group within ProRail.

ProRail EG ERTMS should:

- Assign their trackside to one or more ESC Types.
- Manage Engineering Rules to reduce technical variability of trackside solutions.
- Set up a generic database of ESC Tests for execution in the ESC Test Facility with relation to the activities according to Section 4.2.1.1.
- If ESC Checks encompass ESC Tests:

- In case of an infrastructure change, assess the functional changes to ESC Types and should consider the impact analysis concerning an already existing ESC Statement for an OBU.

ProRail Vehicle Authorisation – The department of ProRail responsible for the vehicle authorisation.

ProRail Vehicle Authorisation should:

- Initiate/organise a Test Campaign according to the terms laid down in this document.
- Check the infrastructure compatibility.
  - Once the infrastructure compatibility is checked and the list of required ESC Checks is approved, this is captured with a 'Visie'.
- Appoint the ProRail ESC Manager.
- If ESC Checks do not encompass ESC Tests, prepare the Check Report and agree with the Entity Applying for ESC Demonstration on the final Check Report. Can be done in collaboration with ProRail EG ERTMS.
- If ESC Checks encompass ESC Tests:
  - Support test analysis with their operational knowledge and confirm the acceptability of any exported constraints to the ESC Types and its operation on request of the ESC Test Manager.
- Approve final ESC Check Report with a 'Visie'

Trackside Supplier – A party responsible for the design and implementation of ETCS trackside products (e.g. the RBC).

The Trackside Supplier should:

- Support the Infrastructure Manager, ESC Test Facility Manager and ESC Test Manager with knowledge about the Trackside Implementation and the relevant ESC Tests.
- Support the process in case of Trackside Implementation issues revealed during a Test Campaign (Issue Handling).
- If required by the Parties, agree with the ESC Test Manager, OBU Supplier and the PREI (Account) Manager to issue the preliminary test report.

Appendix C Design Core Processes

This appendix includes the tables with additional remarks on the ESC testing process and all flowcharts of the core processes. First, the tables with additional remarks are included. The flowcharts follow: first the testing processes, then the demonstration process and at last the training process.

Tables with additional remarks

Table C.1: Table with all steps of preparation phase of ESC testing process, with remarks and indication of the steps of the ERA process

A: ESC Test Campaign – Preparation			
Step	**Activity**	**Remark**	**Step ERA process**
A1	Contact ProRail Vehicle Authorisation about test campaign (e-mail: inzet.spoorvoertuigen@prorail.nl), including all constraints, limitations, (non)implemented CRs, error corrections and the associated onboard behaviour	This e-mail should contain: • The contact details for the test; • The involved train; • The ERTMS onboard equipment; • The involved software; • A time schedule; • All constraints, limitations, (non)implemented CRs, error corrections and the associated onboard behaviour	1
A2	Determine the scope of the test campaign	This step is about the extent of ESC tests and the conditions for their execution. The OBU Supplier and the Trackside Supplier may be consulted if necessary.	1
A3	Discuss the scope of the test campaign	This step is about the extent of ESC tests and the conditions for their execution. The OBU Supplier and the Trackside Supplier may be consulted if necessary.	1
A4	Check infrastructure compatibility	The constraints, limitations, (non)implemented CRs, error corrections and the associated onboard behaviour that are included in the e-mail in step A1 are checked and assessed.	
A5	Compose list of required ESC Checks	The ESC Test Applicant should check which ESC Checks are applicable with respect to the trackside configurations or operational conditions (e.g. speed, location) and the special characteristics of the OBU (e.g. product limitations and maturity (e.g. based on existing ESC evidence)). The decision about	2

		whether ESC Checks need to be performed or repeated should be made at this stage and be justified in the final Check Report (step D5).	
A6	Assess list of required ESC Checks	In this step, the list of required ESC Checks as composed by the ESC Test Applicant is assessed by ProRail Vehicle Authorisation. Any comments are recorded.	2
A7	Comments?	If there are comments added to the list of required ESC Checks, the list should be adjusted.	
A8	Approve list of required ESC Checks (Visie)	If there are no comments left on the list of required ESC Checks, the list is approved through a document called Visie.	2
A9	Appoint ProRail ESC Manager	-	
A10	Appoint ESC Test Manager	-	5
A11	Tests required?	When it is decided not to perform ESC tests, a final ESC Check Report still must be composed. This is done in flowchart D, Completion.	
A12	PREI facilities needed?	The required ESC tests that can be executed in the lab facility, are done at PREI. When the required ESC tests cannot be performed at PREI, field tests are done. This is done in flowchart C, Field Execution.	
A13	Can all tests be done at PREI?	The required ESC tests that can be executed in the lab facility, are done at PREI. The required ESC tests that cannot be executed at PREI are performed as field tests. This is done in flowchart C, Field Execution.	
A14	Inform ProRail ESC Manager about the required tests at PREI	-	
A15	Appoint PREI (Account) Manager	-	
A16	Request required ESC tests to PREI (Account) Manager	-	3
A17	Request required ESC tests to PREI	-	3
A18	Inform ESC Test Applicant about the test facility and conditions	-	4
A19	Compose test plan, including ESC tests and order	The test plan includes a script. The script describes the whole test campaign and ensures that it is clear who is doing what. Also, the specific	

		ESC tests and order should be present.	
A20	Assess test plan on feasibility	The PREI Test Facility Manager assesses the test plan only on the technical feasibility. Practical matters are considered, such as the time estimates and whether a test is executable.	
A21	Comments?	If there are comments added to the test plan, the test plan should be adjusted.	
A22	Schedule the test campaign	Taking the test plan into account.	

Table C.2: Table with all steps of the PREI execution phase of ESC testing process, with remarks and indication of the steps of the ERA process

B: ESC Test Campaign – PREI Execution			
Step	**Activity**	**Remark**	**Step ERA process**
B1	Ensure the OBU Supplier is involved	The ESC Test Manager ensures that the OBU Supplier is involved in the test campaign.	
B2	Start testing campaign according to schedule	-	
B3	Identify applicable Subsets-110/111/112 versions	It is assumed that the OBU Supplier has to be involved in case of tests in a laboratory environment, especially if the OBU Test Bench is integrated with the PREI Test Facility for the first time.	6
B4	Check functionality of communication link between test facility and OBU test bench	It is assumed that the OBU Supplier has to be involved in case of tests in a laboratory environment, especially if the OBU Test Bench is integrated with the PREI Test Facility for the first time.	7
B5	Connection succeeded?	-	
B6	Request support of TCL supplier	When the connection between the test facility and the OBU test bench is not succeeded, the process cannot proceed. When the connection failed, support is requested of the supplier of the TCL in the PREI Test Facility.	
B7	Start re-execution according to schedule	Based on the conclusions at Issue Handling (flowchart E), tests might be re-executed. It is the responsibility of the ESC Test Manager that this re-execution is started.	
B8	Manage test execution	-	9a
B9	Witness test execution	-	

B10	Execute tests according to test plan	The ESC tester is responsible for the execution of the tests.	
B11	Witness and facilitate tests according to test plan	PREI test & support is present during the execution of the tests and has a witness role. Facilitating tests includes technical support of PREI facilities.	
B12	Record findings	-	
B13	Record observations in Test observation report	-	
B14	Could all tests be performed?	-	
B15	Can issues be solved during the test campaign?	In this step is determined whether the issues are small (e.g. wrong setting used) and can be solved during the test campaign. Large (technical) issues cannot be solved during the test campaign.	
B16	Solve issues	Dependent on the issue(s), the responsible party should solve the small issue(s).	9b

Table C.3: Table with all steps of the field execution phase of ESC testing process, with remarks and indication of the steps of the ERA process

C: ESC Test Campaign – Field Execution			
Step	**Activity**	**Remark**	**Step ERA process**
C1	Make sure all field tests are done and completed	It is the responsibility of the ESC Test Manager that all field tests are done and completed.	

Table C.4: Table with all steps of completion phase of ESC testing process, with remarks and indication of the steps of the ERA process

D: ESC Test Campaign – Completion			
Step	**Activity**	**Remark**	**Step ERA process**
D1	Create preliminary test report, including the Visie (step A8)	The preliminary test report should contain the test result categorised by ‘OK’, ‘NOK’, ‘possible IOP issue’. Possible IOP issues are the ones that could be due to an error as defined in CCS TSI Section 6.5. The Visie composed in the Preparation phase (step A8) is added to the preliminary test report.	9c
D2	Review test report	The preliminary test report is especially reviewed on the contents (e.g. is the report complete).	9d

D3	Agreement?	-	11
D4	Is the report fit for purpose?	In case (technical) issues are determined in the preliminary test report, and the report does not fit for its purpose, the determined issues should be solved before the process can continue. See flowchart E, Issue Handling. When the report is fit for its purpose, the technical issues do not have to be solved before the final ESC Check Report can be created.	
D5	Create final ESC Check Report	The final ESC Check Report must always show the complete result on all required checks, even if it was decided not to fully execute them.	12-15
D6	Hand-over final ESC Check Report to ProRail Vehicle Authorisation	-	
D7	Approve final ESC Check Report (Visie)	The final ESC Check Report should be approved by the Infrastructure Manager (ProRail Vehicle Authorisation). This is done through a Visie. The ESC Check Report is assessed on the findings within the report. The content of the report (e.g. is the report complete) is not reviewed, this is already done in step D2.	12-15
D8	Final ESC Check Report approved?	If the final ESC Check Report is not approved, the final ESC Check Report should be adjusted.	12-15
D9	Inform ESC Test Manager that final ESC Check Report is not approved	-	
D10	Inform ESC Test Manager and ProRail ESC Manager that final ESC Check Report is approved	-	
D11	Close out conditions of report (if any)	-	12-15
D12	Hire Notified Body and hand-over ESC Check Report	-	16
D13	Assess ESC Check Report	The Notified Body checks: • that the report gives reference to the necessary checks according to the technical document published by ERA; • that ESC checks have been performed and the results indicate for every ESC Check whether the ESC Check was passed as specified or not;	17

		• that for every ESC Check which was not passed as specified, the incompatibilities and errors encountered during ESC Checks are stated; • that for every ESC Check which was not passed as specified, an analysis of the effects on ESC has been performed in accordance with steps D1-D3.	
D14	ESC Check Report confirmed?	If the result of the assessment requires rework of the ESC Check Report, the ESC Check Report should be adjusted accordingly.	18
D15	Inform ESC Test Manager that ESC Check Report is not confirmed	The Notified Body sends the results back to the ESC Test Applicant. The ESC Test Applicant informs the ESC Test Manager about the rework of the ESC Check Report.	
D16	ESC Check Report confirmation (positive assessment report)	If the assessment of the Notified Body end with a positive result, he should confirm that in an Assessment Report.	19
D17	Inform OBU supplier to draw up ESC IC Statement	-	
D18	Draw up ESC IC Statement	-	20

Table C.5: Table with all steps of the issue handling phase of ESC testing process, with remarks and indication of the steps of the ERA process

E: ESC Test Campaign – Issue Handling			
Step	**Activity**	**Remark**	**Step ERA process**
E1	What kind of issue?	-	
E2	Decide which actions are required	For the remaining issues, the responsible party will decide on consequences on operational, product, engineering or interoperability issues. This might require raising the issue to other stakeholders (e.g. the European Union Agency for Railways (ERA) in case of possible interoperability issues).	10a-d
E3	Take required actions	For the remaining issues, the responsible party will decide on consequences on operational, product, engineering or interoperability issues. This might require raising the issue to other stakeholders (e.g. the European Union Agency for Railways (ERA) in	10a-d

		case of possible interoperability issues).	
E4	Re-execution of tests needed?	Based on the conclusions of the previous steps in this flowchart, tests might be re-executed.	10e
E5	Re-evaluation of test results needed?	Based on the conclusions of the previous steps in this flowchart, test results might be re-evaluated.	10e
E6	Decide which actions are required	-	
E7	Schedule re-execution of tests	-	

Flowcharts

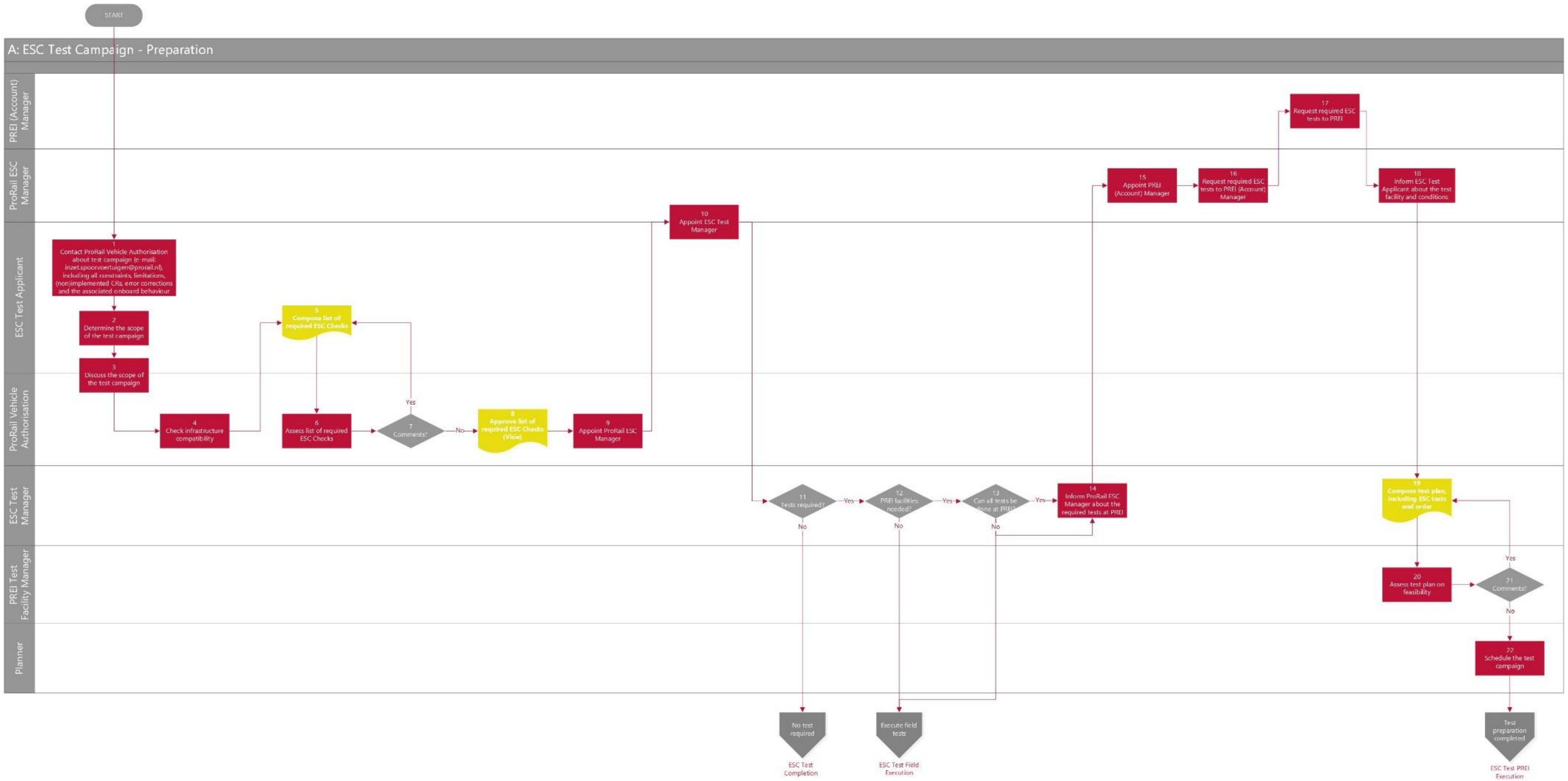

Figure C.1: Flowchart of the preparation phase of the ESC test process

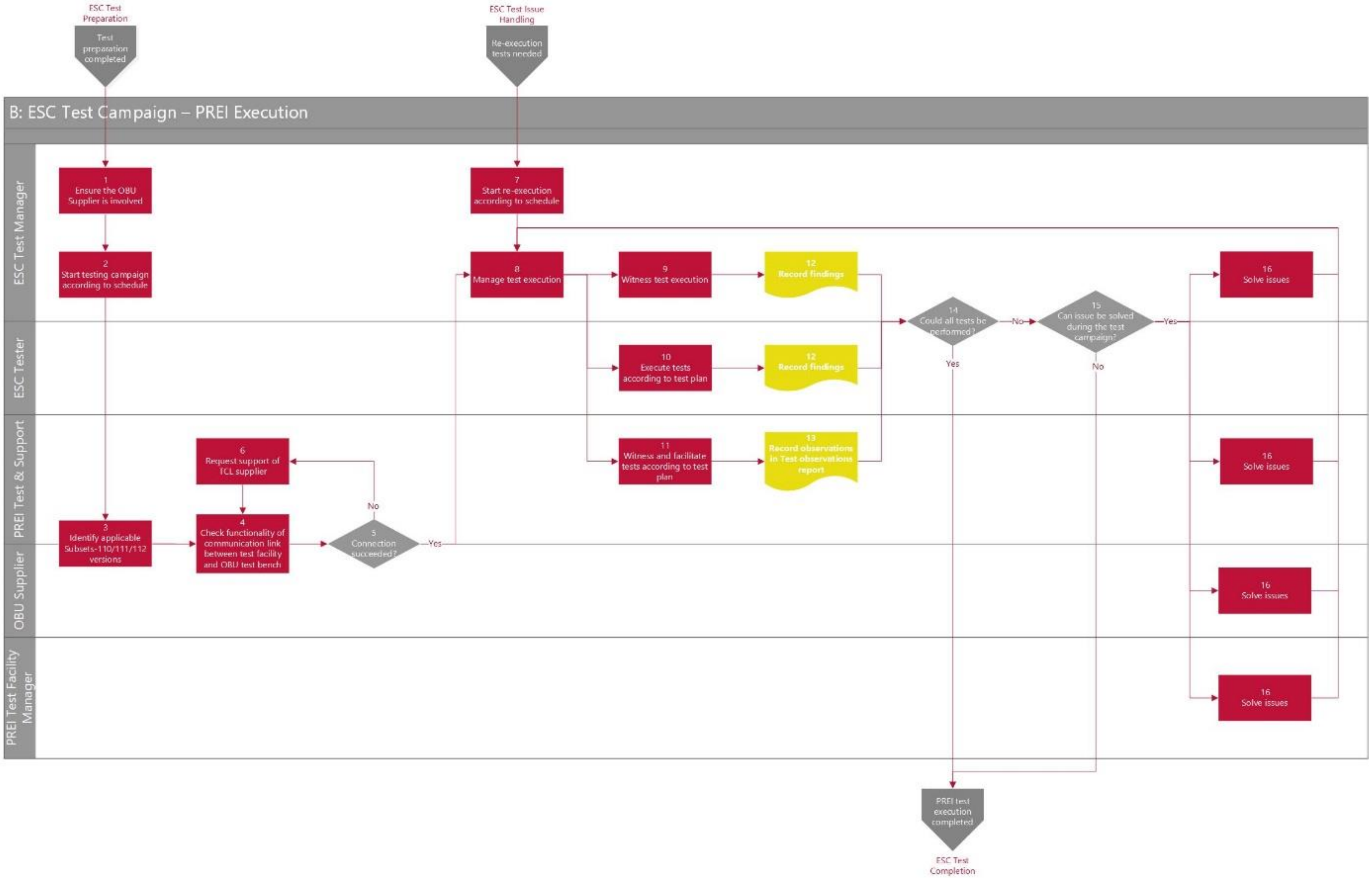

Figure C.2: Flowchart of the PREI execution phase of the ESC test process

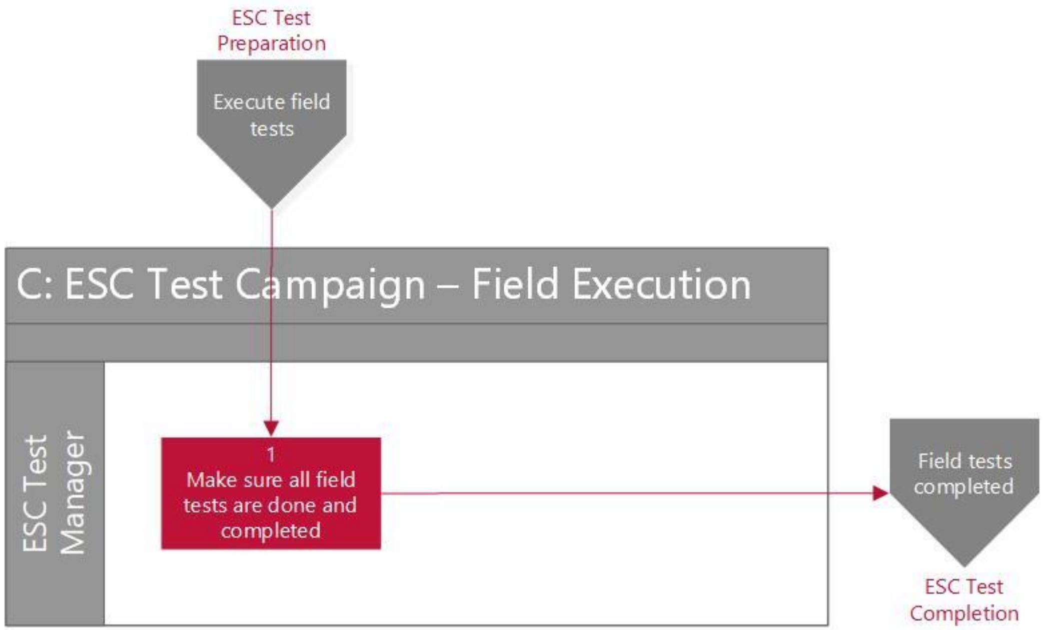

Figure C.3: Flowchart of the field execution phase of the ESC test process

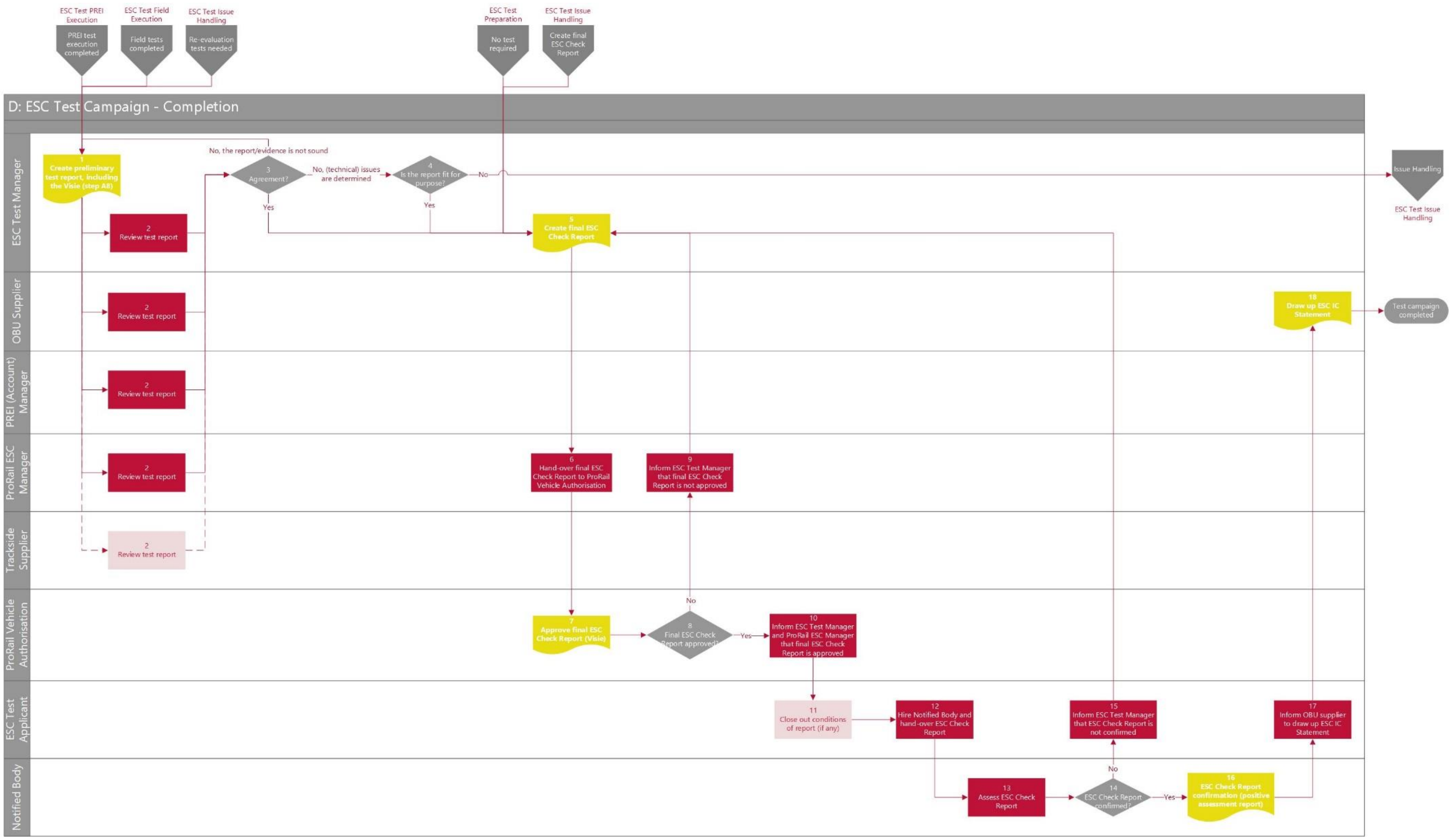

Figure C.4: Flowchart of the completion phase of the ESC test process

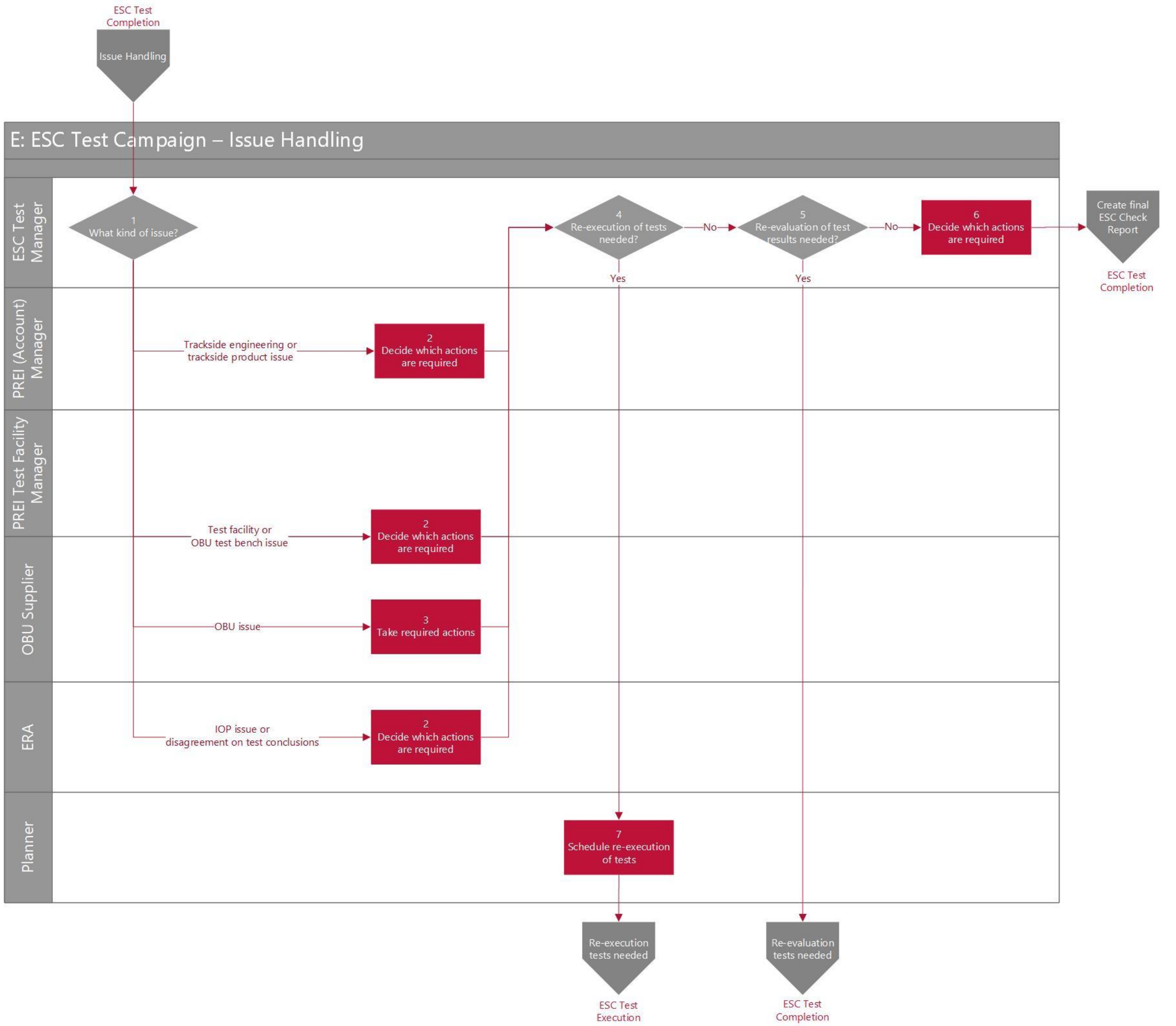

Figure C.5: Flowchart of the issue handling phase of the ESC test process

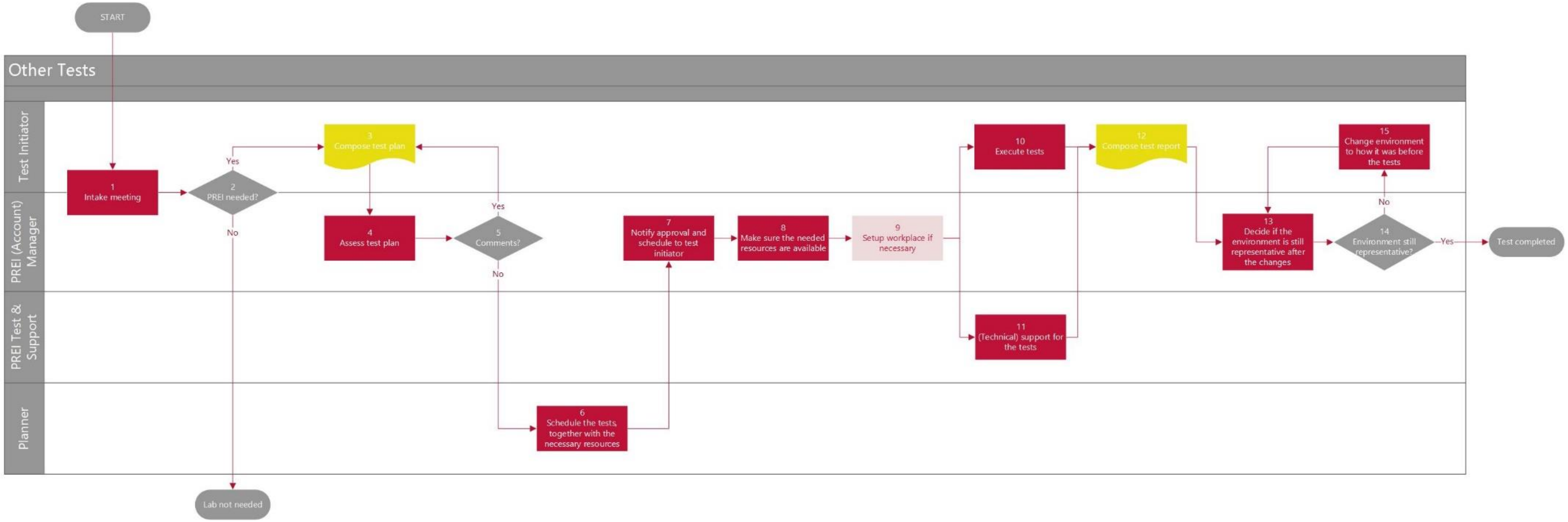

Figure C.6: Flowchart of the process of the group other tests

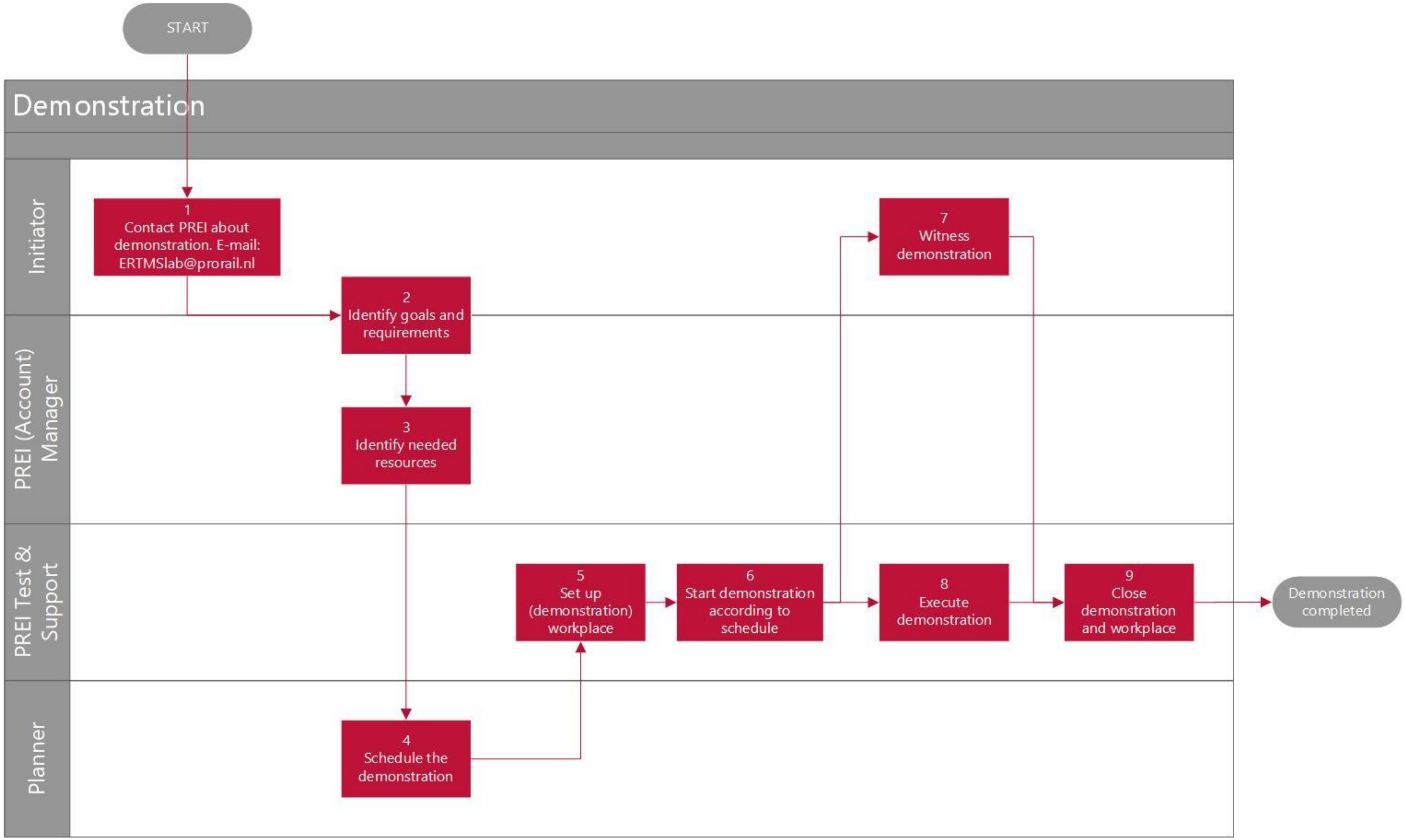

Figure C.7: Flowchart of the demonstration process

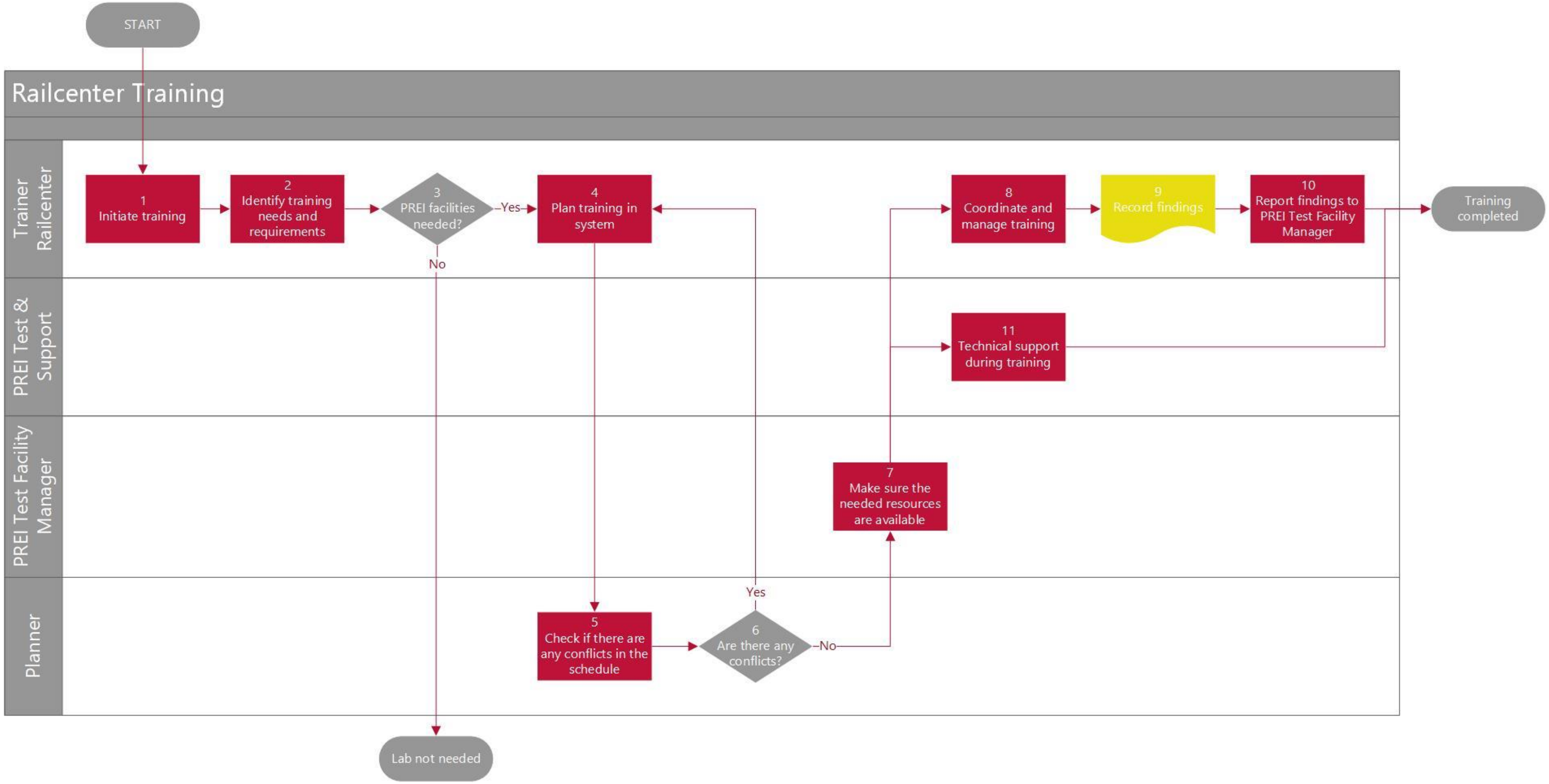

Figure C.8: Flowchart of the training process

Appendix D Overview ILP

An overview of the ILP can be found in this appendix. First, the sets and indices are given. Then, the parameters and variables are elaborated. Thereafter, the complete ILP is given, starting with the objective and followed by the constraints. The complete ILP is given with the objective of minimising the maximum lateness, and with the objective of minimising the total tardiness.

Sets

Activities	$J = \{1, 2, \dots\}$
Workplaces	$W = \{0, \dots, 8\}$
Lines	$L = \{Amsterdam - Utrecht, HSL - Zuid, Hanzelijn, Betuweroute, Havenspoorlijn\}$
Resources	$K = \{HHT, Post21\}$
Time horizon	$T = \{1, 2, \dots\}$

Table D.1: Subsets of W

Set	Workplaces	Description
E	0, 1, 2, 3, 4, 5, 6, 7	All real, existing workplaces
S	0, 1, 4, 5, 6	All workplaces in small and large rooms
D	2, 3	All workplaces in the double room
L	2, 3, 4, 5, 6	All workplaces in the large and double rooms

Indices

Activity	$j \in J$
Workplace	$w, w' \in W$
Line	$\ell \in L$
Resource	$k \in K$
Time period	$t, t' \in T$

Parameters

Integer

Processing time of activity j	p_j
Group size of activity j	g_j
Availability of resources type k	R_k
Number of resources type k needed for activity j	r_{jk}
Release date of activity j	$\underline{\delta}_j$

Due date of activity j $\bar{\delta}_j$

Binary

$$z_j = \begin{cases} 1, & if\ activity\ j\ requires\ the\ fictional\ workplace \\ 0, & otherwise. \end{cases}$$

$$\alpha_j = \begin{cases} 1, & if\ activity\ j\ requires\ two\ workplaces \\ 0, & if\ activity\ j\ requires\ one\ (fictional)\ workplace. \end{cases}$$

$$s_j = \begin{cases} 1, & if\ activity\ j\ is\ preferably\ scheduled\ in\ the\ room\ with\ two\ workplaces \\ 0, & otherwise. \end{cases}$$

$$double_j = \begin{cases} 1, & if\ activity\ j\ requires\ the\ room\ with\ two\ workplaces\ (double\ room) \\ 0, & otherwise. \end{cases}$$

$$large_j = \begin{cases} 1, & if\ activity\ j\ requires\ at\ least\ a\ large\ room \\ 0, & otherwise. \end{cases}$$

$$o_j = \begin{cases} 1, & if\ activity\ j\ is\ a\ demonstration \\ 0, & otherwise. \end{cases}$$

$$b_{j\ell} = \begin{cases} 1, & if\ activity\ j\ requires\ line\ \ell \\ 0, & otherwise. \end{cases}$$

Variables

Integer

Maximum lateness L_{max}

Tardiness of activity j T_j

Binary

$$x_{jwt} = \begin{cases} 1, & if\ activity\ j\ is\ executed\ at\ workplace\ w\ and\ finished\ at\ time\ t \\ 0, & otherwise. \end{cases}$$

Mathematical Programming Formulation

Minimise Maximum Lateness

$$\min L_{max} + \beta_1 \cdot Penalty_1 + \beta_2 \cdot Penalty_2 \tag{D.1}$$

Subject to

$$L_{max} \geq x_{jwt} \cdot t + p_j - \bar{\delta}_j, \qquad \forall j, w, t \tag{D.2}$$

$$Penalty_1 = \sum_j (1 - o_j) \cdot \sum_t x_{j,7,t} \tag{D.3}$$

$$Penalty_2 = \sum_j s_j \cdot \alpha_j \cdot \sum_{w \in S} \sum_t x_{jwt} \tag{D.4}$$

$$\sum_t x_{j,8,t} = z_j, \qquad \forall j \tag{D.5}$$

$$\sum_w \sum_t x_{jwt} = (\alpha_j + 1), \qquad \forall j \tag{D.6}$$

$$\alpha_j \cdot \sum_{w'} x_{jw't} \geq 2 \cdot \alpha_j \cdot x_{jwt}, \qquad \forall j, w, t \tag{D.7}$$

$$\sum_j \sum_{t'=\max(0, t-p_j+1)}^{t} x_{jwt'} \leq 1, \qquad \forall w \in E, t \tag{D.8}$$

$$\sum_t x_{j,7,t} \geq o_j, \qquad \forall j \tag{D.9}$$

$$(1 - s_j) \cdot \alpha_j \cdot \sum_{w \in S} \sum_t x_{jwt} \geq (1 - s_j) \cdot \alpha_j, \qquad \forall j \tag{D.10}$$

$$(1 - o_j) \cdot (1 - z_j) \cdot double_j \cdot \sum_{w \in D} \sum_t x_{jwt} \geq (1 - o_j) \cdot (1 - z_j) \cdot double_j, \qquad \forall j \tag{D.11}$$

$$(1 - o_j) \cdot (1 - z_j) \cdot large_j \cdot \sum_{w \in L} \sum_t x_{jwt} \geq (1 - o_j) \cdot (1 - z_j) \cdot large_j, \qquad \forall j \tag{D.12}$$

$$\sum_j \frac{b_{j\ell} \cdot \sum_{t'=\max(0, t-p_j+1)}^{t} \sum_w x_{jwt'}}{a_j + 1} \leq 1, \qquad \forall t, \ell \tag{D.13}$$

$$\sum_j \frac{r_{jk} \cdot \sum_{t'=\max(0, t-p_j+1)}^{t} \sum_w x_{jwt'}}{a_j + 1} \leq R_k, \qquad \forall t, k \tag{D.14}$$

$$\sum_w \sum_t x_{jwt} \cdot t \geq \underline{\delta}_j (\alpha_j + 1), \qquad \forall j \tag{D.15}$$

Minimise Total Tardiness

$$\min \sum_j T_j + \beta_1 \cdot Penalty_1 + \beta_2 \cdot Penalty_2 \tag{D.16}$$

Subject to

$$T_j \geq \frac{\left(\sum_w \sum_t x_{jwt} \cdot t\right)}{\left(\alpha_j + 1\right)} + p_j - \overline{\delta}_j, \qquad \forall j \tag{D.17}$$

$$Penalty_1 = \sum_j \left(1 - o_j\right) \cdot \sum_t x_{j,7,t} \tag{D.18}$$

$$Penalty_2 = \sum_j s_j \cdot \alpha_j \cdot \sum_{w \in S} \sum_t x_{jwt} \tag{D.19}$$

$$\sum_t x_{j,8,t} = z_j, \qquad \forall j \tag{D.20}$$

$$\sum_w \sum_t x_{jwt} = \left(\alpha_j + 1\right), \qquad \forall j \tag{D.21}$$

$$\alpha_j \cdot \sum_{w'} x_{jw't} \geq 2 \cdot \alpha_j \cdot x_{jwt}, \qquad \forall j, w, t \tag{D.22}$$

$$\sum_j \sum_{t'=\max(0, t-p_j+1)}^{t} x_{jwt'} \leq 1, \qquad \forall w \in E, t \tag{D.23}$$

$$\sum_t x_{j,7,t} \geq o_j, \qquad \forall j \tag{D.24}$$

$$\left(1 - s_j\right) \cdot \alpha_j \cdot \sum_{w \in S} \sum_t x_{jwt} \geq \left(1 - s_j\right) \cdot \alpha_j, \qquad \forall j \tag{D.25}$$

$$\left(1 - o_j\right) \cdot \left(1 - z_j\right) \cdot double_j \cdot \sum_{w \in D} \sum_t x_{jwt} \geq \left(1 - o_j\right) \cdot \left(1 - z_j\right) \cdot double_j, \qquad \forall j \tag{D.26}$$

$$\left(1 - o_j\right) \cdot \left(1 - z_j\right) \cdot large_j \cdot \sum_{w \in L} \sum_t x_{jwt} \geq \left(1 - o_j\right) \cdot \left(1 - z_j\right) \cdot large_j, \qquad \forall j \tag{D.27}$$

$$\sum_j \frac{b_{j\ell} \cdot \sum_{t'=\max(0, t-p_j+1)}^{t} \sum_w x_{jwt'}}{a_j + 1} \leq 1, \qquad \forall t, \ell \tag{D.28}$$

$$\sum_j \frac{r_{jk} \cdot \sum_{t'=\max(0, t-p_j+1)}^{t} \sum_w x_{jwt'}}{a_j + 1} \leq R_k, \qquad \forall t, k \tag{D.29}$$

$$\sum_w \sum_t x_{jwt} \cdot t \geq \underline{\delta}_j \left(\alpha_j + 1\right), \qquad \forall j \tag{D.30}$$

Appendix E Overview Provided Data

Table E.1: Overview of the provided data used for the experiments

#	Name	Activity (Test ESC, Test Other, Training, Demonstration, Service)	Line (HSLZ, Asd-Ut, HZL, BR A15, Havenspoorlijn)	Post21 (1=yes, 0=no)	Number of HHTs needed	Number of workplaces needed	Workplaces in same room?	Number of involved people (per workplace)	Processing time	Release date	Due date
1	Service window Post21	Service	-	1	0	0	-	0	2	25	27
2	Service window Post21	Service	-	1	0	0	-	0	2	65	67
3	Service window Post21	Service	-	1	0	0	-	0	2	105	107
4	Service window Post21	Service	-	1	0	0	-	0	2	155	157
5	Service window Post21	Service	-	1	0	0	-	0	2	195	197
6	Service window Post21	Service	-	1	0	0	-	0	2	235	237
7	ESC test ICNG op HSLZ	Test ESC	HSLZ	0	0	1	-	2	10	43	216
8	ESC test ICNG op HZL	Test ESC	HZL	1	0	1	-	2	8	43	216
9	ESC test ICNG op Asd-Ut	Test ESC	Asd-Ut	1	0	1	-	2	12	43	216
10	ESC test LOC AXL124 op BR-A15 en HZL	Test ESC	BR A15, HZL	1	0	1	-	2	16	129	259
11	ESC test LOC AXL124 op Havenspoorlijn	Test ESC	Havenspoorlijn	0	0	1	-	2	11	129	259
12	ESC test LOC AXL124 op Asd-Ut	Test ESC	Asd-Ut	1	0	1	-	2	8	129	259
13	Test Asd-Ut patch Zomer/wintertijd probleem	Test Other	Asd-Ut	1	0	1	-	8	8	1	32
14	Test HSLZ keten KEVNL	Test Other	HSLZ	1	0	1	-	3	20	83	173
15	Test Cybersecurity	Test Other	BR A15	0	3	1	-	3	10	173	192
16	TEST KMC	Test Other	-	0	0	2	Yes	3	60	43	173
17	ERTMS-programma PEIL-leveringen (post21)	Test Other	-	0	0	1	-	3	4	1	44
18	ERTMS-programma PEIL-leveringen (post21)	Test Other	-	0	0	1	-	3	4	43	84
19	ERTMS-programma PEIL-leveringen (post21)	Test Other	-	0	0	1	-	3	4	83	130
20	ERTMS-programma PEIL-leveringen (post21)	Test Other	-	0	0	1	-	3	4	129	173

#	Name	Activity (Test ESC, Test Other, Training, Demonstration, Service)	Line (HSLZ, Asd-Ut, HZL, BR A15, Havenspoorlijn)	Post21 (1=yes, 0=no)	Number of HHTs needed	Number of workplaces needed	Workplaces in same room?	Number of involved people (per workplace)	Processing time	Release date	Due date
21	ERTMS-programma PEIL-leveringen (post21)	Test Other	-	0	0	1	-	3	4	173	216
22	ERTMS-programma PEIL-leveringen (post21)	Test Other	-	0	0	1	-	3	4	215	259
23	Opleiding HHT Initieel	Training	HZL	1	3	2	No	4	6	67	73
24	Opleiding HHT Initieel	Training	HZL	1	3	2	No	4	6	147	153
25	Opleiding HHT Initieel	Training	HZL	1	3	2	No	8	6	207	213
26	Opleiding HHT Heractivering	Training	HZL	1	3	2	No	4	2	59	61
27	Opleiding HHT Heractivering	Training	HSLZ	1	3	2	No	4	2	109	111
28	Opleiding HHT Heractivering	Training	HZL	1	3	2	No	4	2	145	147
29	Opleiding HHT Heractivering	Training	HZL	1	3	2	No	4	2	245	247
30	Demo A algemeen 1 uur, groep <= 8	Demonstration	HZL	1	0	1	-	8	1	63	65
31	Demo B carrousel algemeen 3x 1 uur, groep <= 8	Demonstration	HZL	1	0	1	-	8	1	115	117
32	Demo C algemeen 1 uur, groep <= 8	Demonstration	HZL	1	0	1	-	8	1	149	151
33	Demo algemeen 2 uur groep <= 8	Demonstration	HZL	1	0	1	-	8	1	23	25
34	Demo werkzones op HZL of BRA15	Demonstration	HZL	1	3	1	-	8	1	29	31
35	Demo specifiek HSLZ	Demonstration	HZL	0	0	1	-	8	1	199	201

Appendix F Results Model Validation Provided Data Expanded

This appendix includes all visual views of the solutions for the various expansions of the provided data.

Expansion 1: Duplicate All Tests

Objective of Minimising the Maximum Lateness

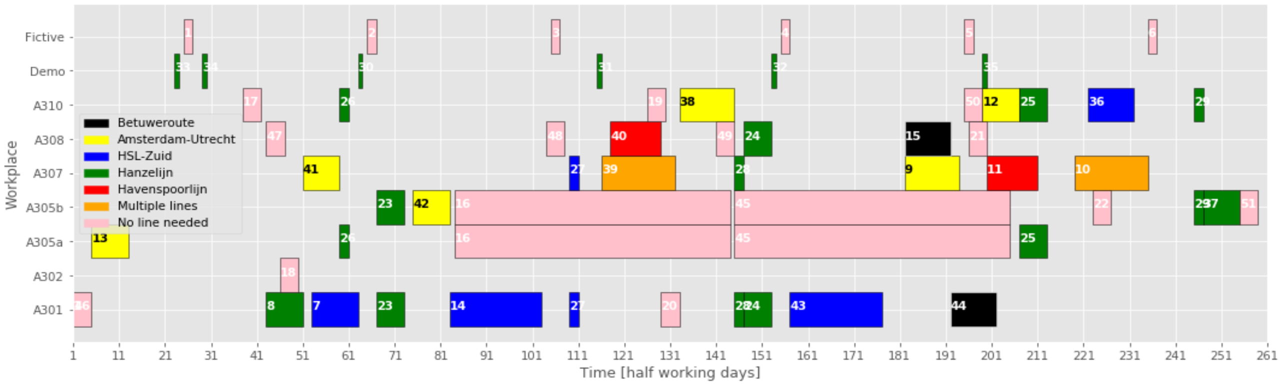

Figure F.1: Operational schedule of data set expanded with duplicates of all tests and objective to minimise the maximum lateness

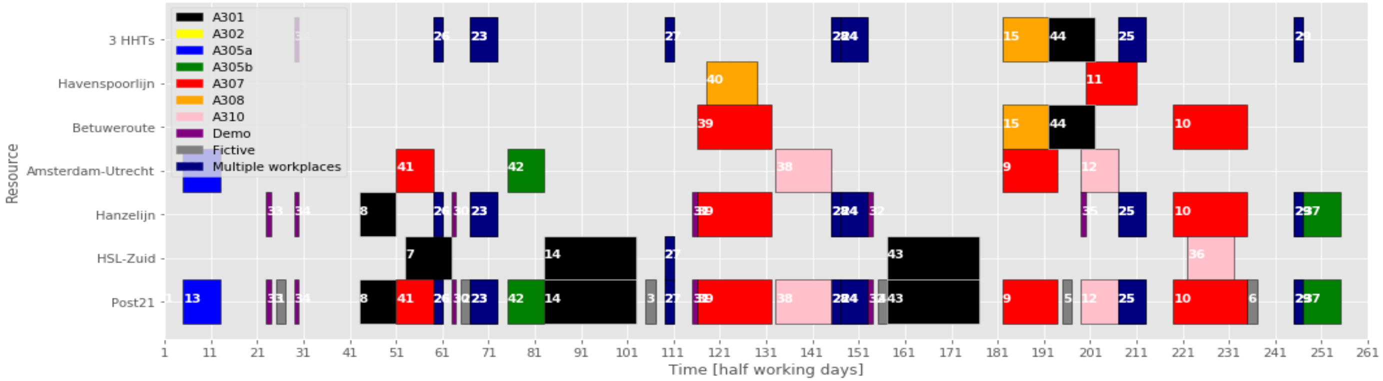

Figure F.2: Resource allocation of data set expanded with duplicates of all tests and objective to minimise the maximum lateness

Objective of Minimising the Total Tardiness

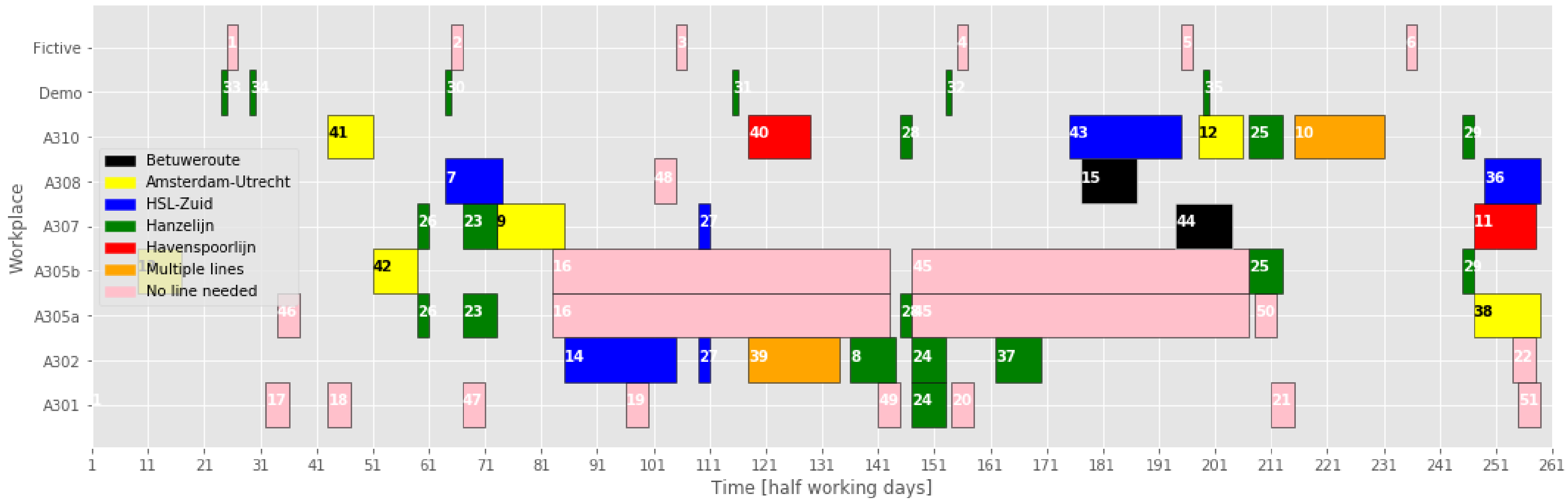

Figure F.3: Operational schedule of data set expanded with duplicates of all tests and objective to minimise the total tardiness

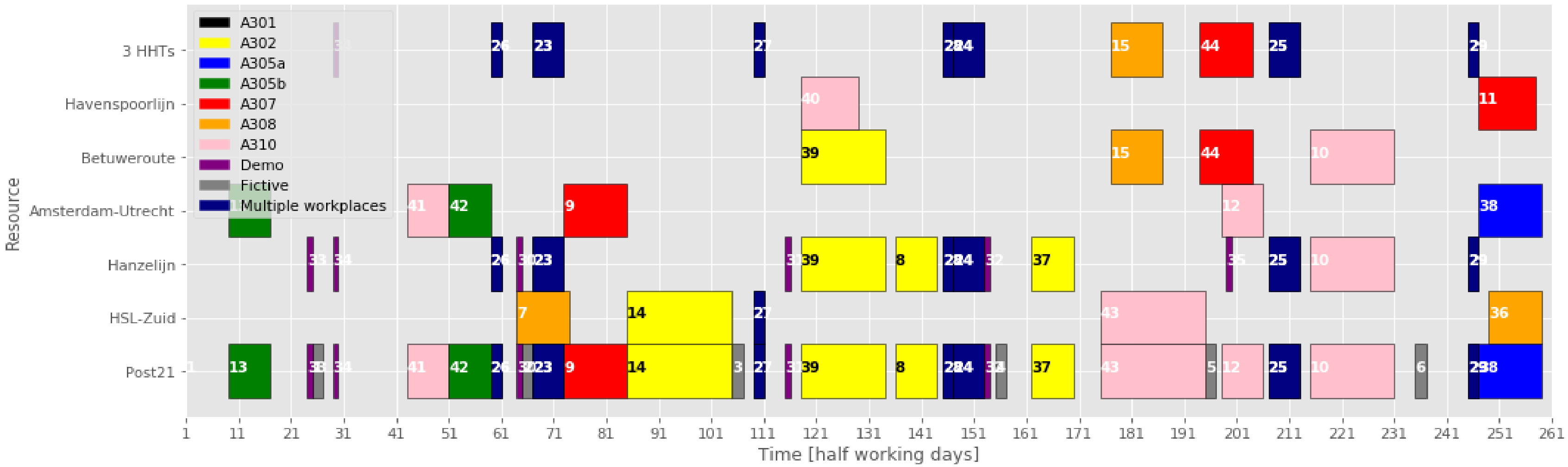

Figure F.4: Resource allocation of data set expanded with duplicates of all tests and objective to minimise the total tardiness

Expansion 2: Duplicate All Tests and Demonstrations

Objective of Minimising the Maximum Lateness

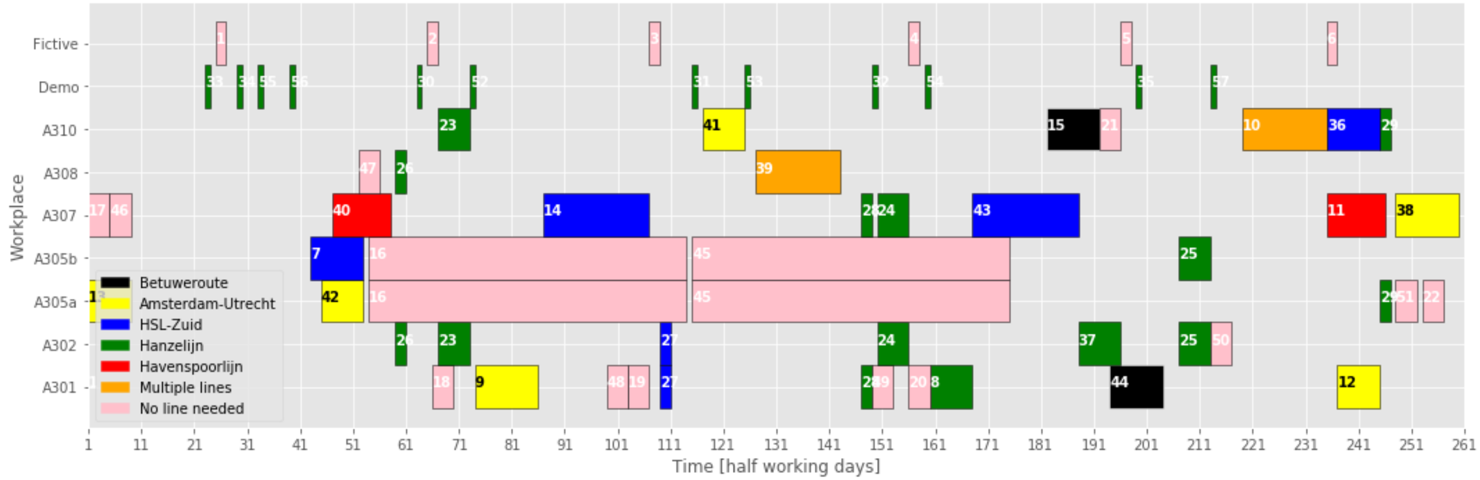

Figure F.5: Operational schedule of data set expanded with duplicates of all tests and demonstrations, and objective to minimise the maximum lateness

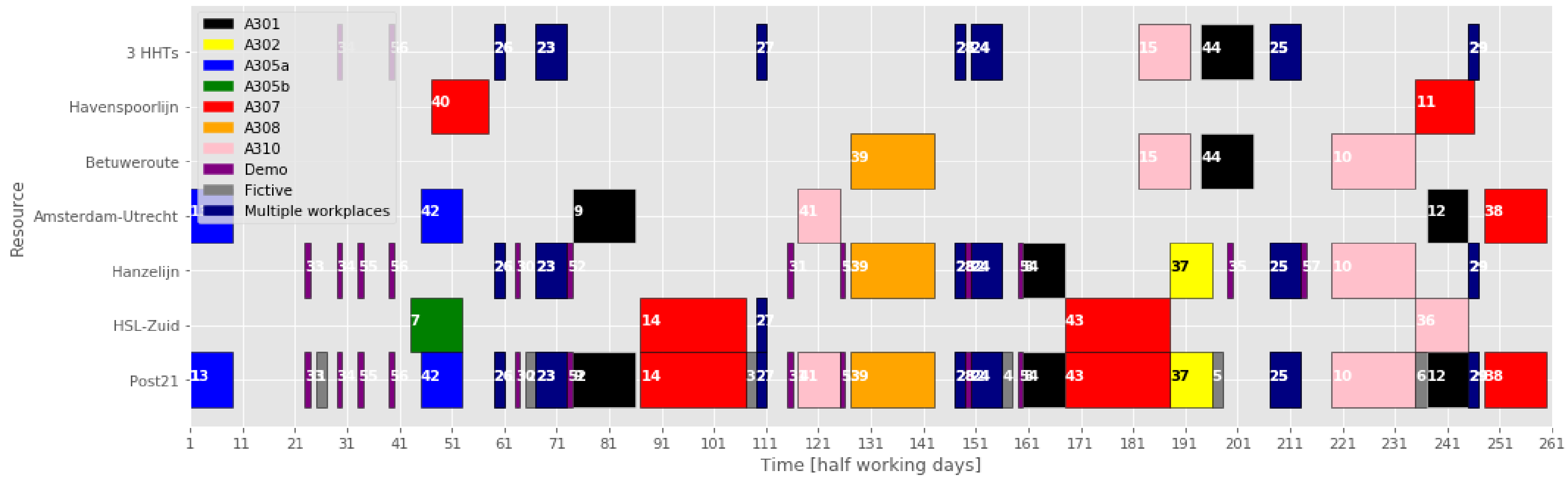

Figure F.6: Resource allocation of data set expanded with duplicates of all tests and demonstrations, and objective to minimise the maximum lateness

Objective of Minimising the Total Tardiness

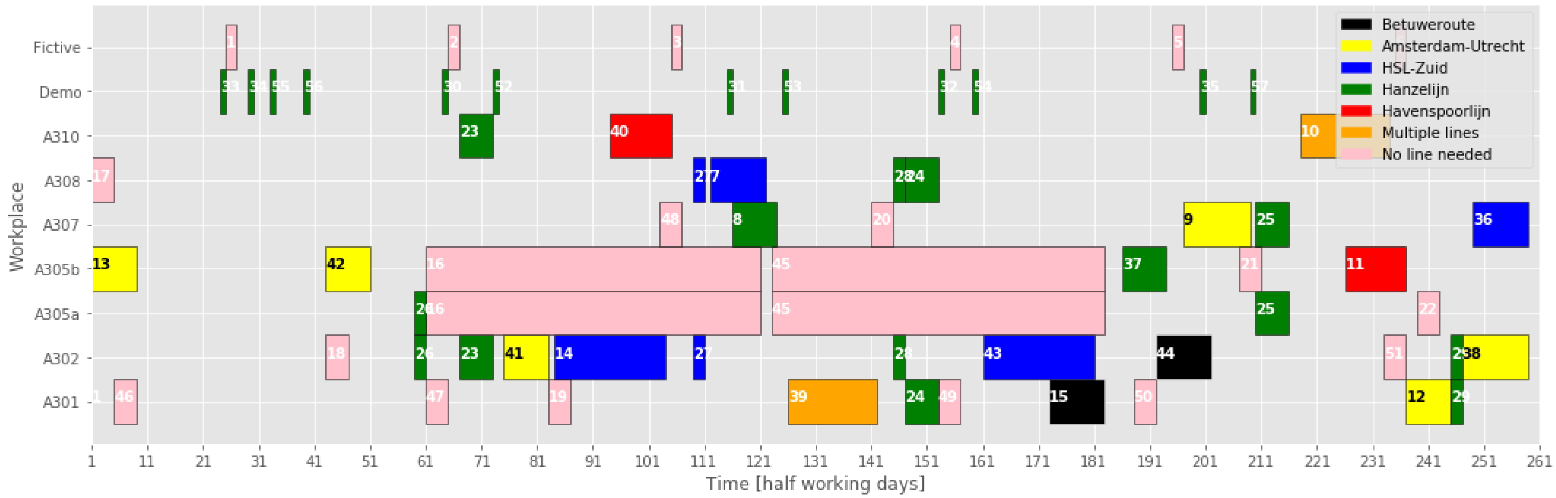

Figure F.7: Operational schedule of data set expanded with duplicates of all tests and demonstrations, and objective to minimise the total tardiness

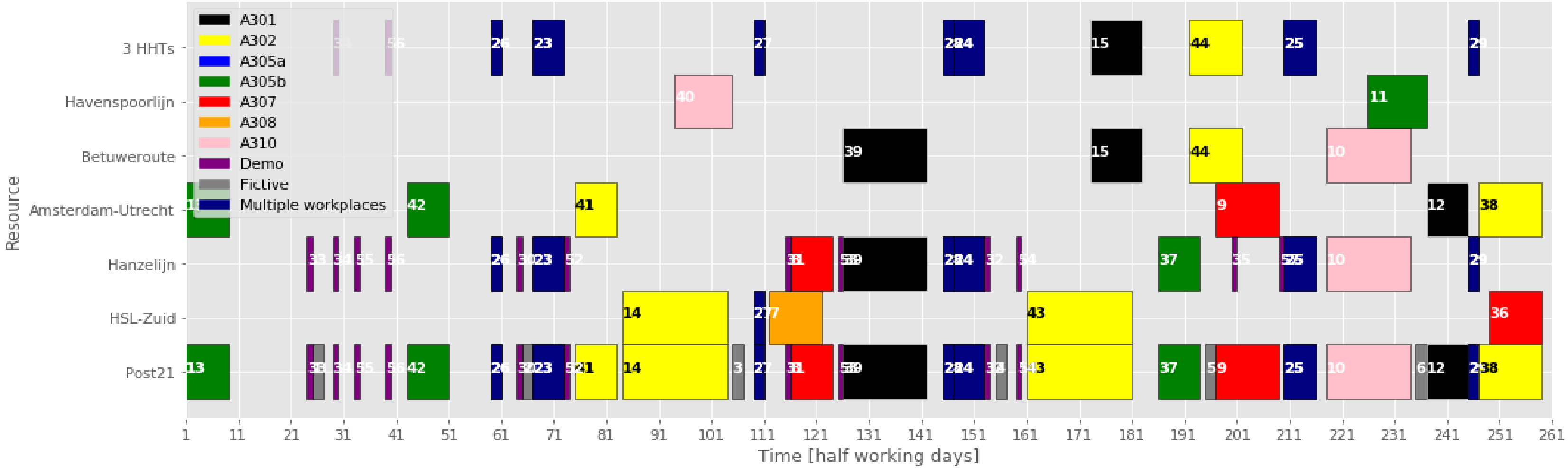

Figure F.8: Resource allocation of data set expanded with duplicates of all tests and demonstrations, and objective to minimise the total tardiness

Expansion 3: Duplicate All Tests and Trainings

Objective of Minimising the Maximum Lateness

Maximum Run Time of 3600 Seconds

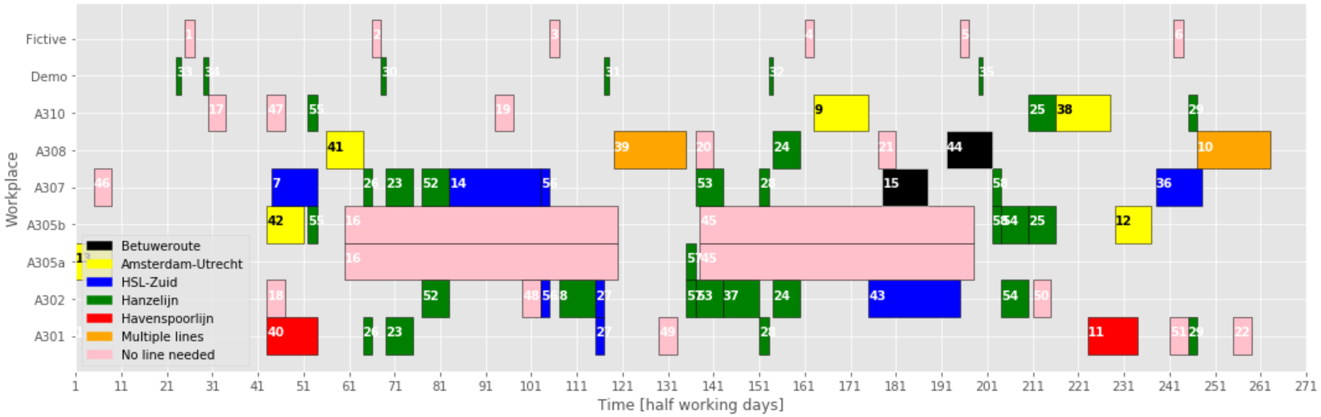

Figure F.9: Operational schedule of data set expanded with duplicates of all tests and trainings, objective to minimise the maximum lateness and maximum run time = 3600 seconds

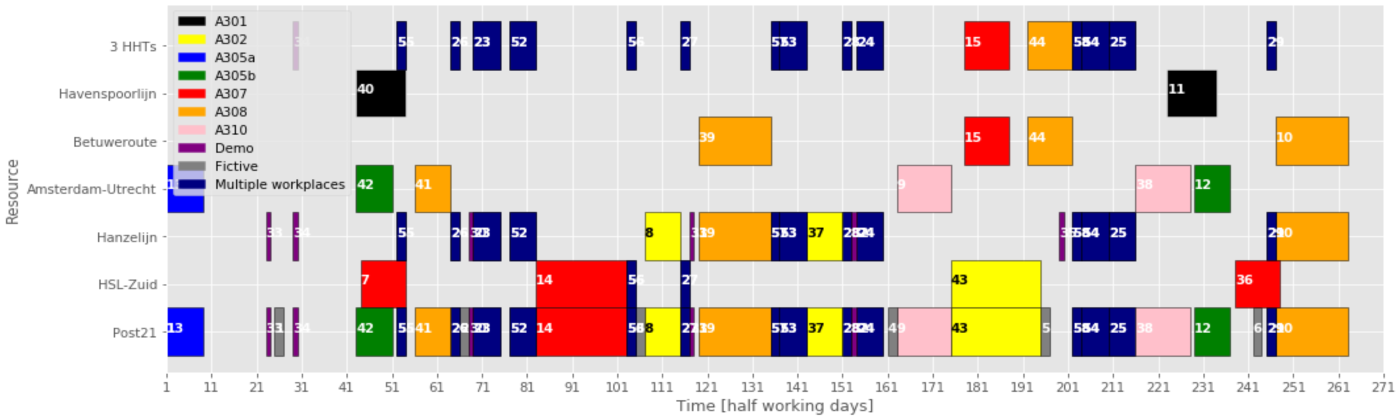

Figure F.10: Resource allocation of data set expanded with duplicates of all tests and trainings, objective to minimise the maximum lateness and maximum run time = 3600 seconds

Objective of Minimising the Total Tardiness

Maximum Run Time of 300 Seconds

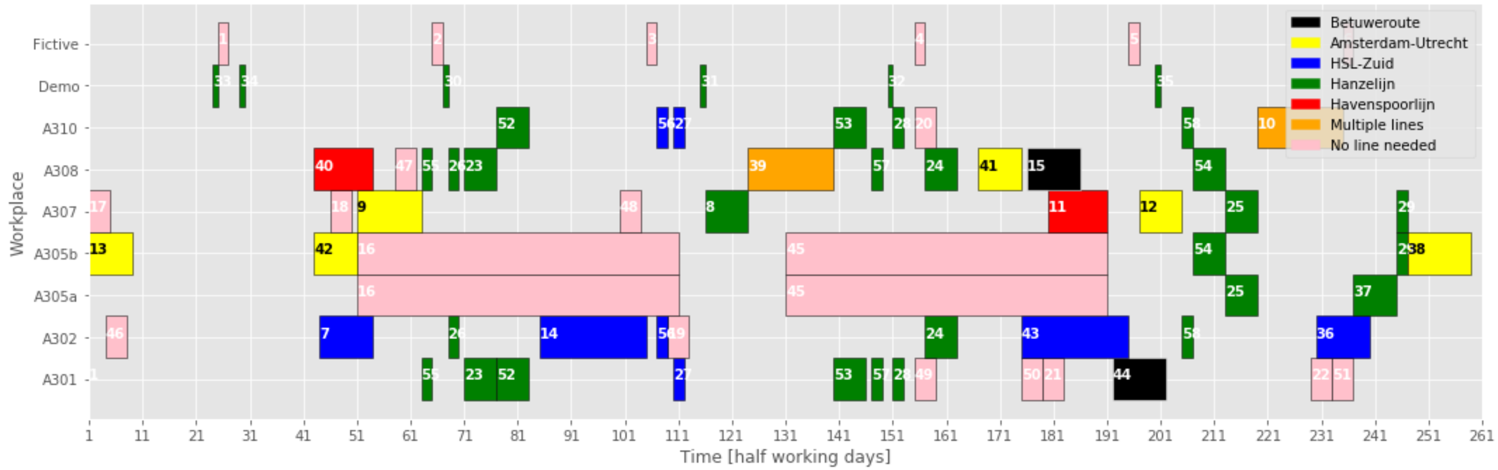

Figure F.11: Operational schedule of data set expanded with duplicates of all tests and trainings, objective to minimise the total tardiness and maximum run time = 300 seconds

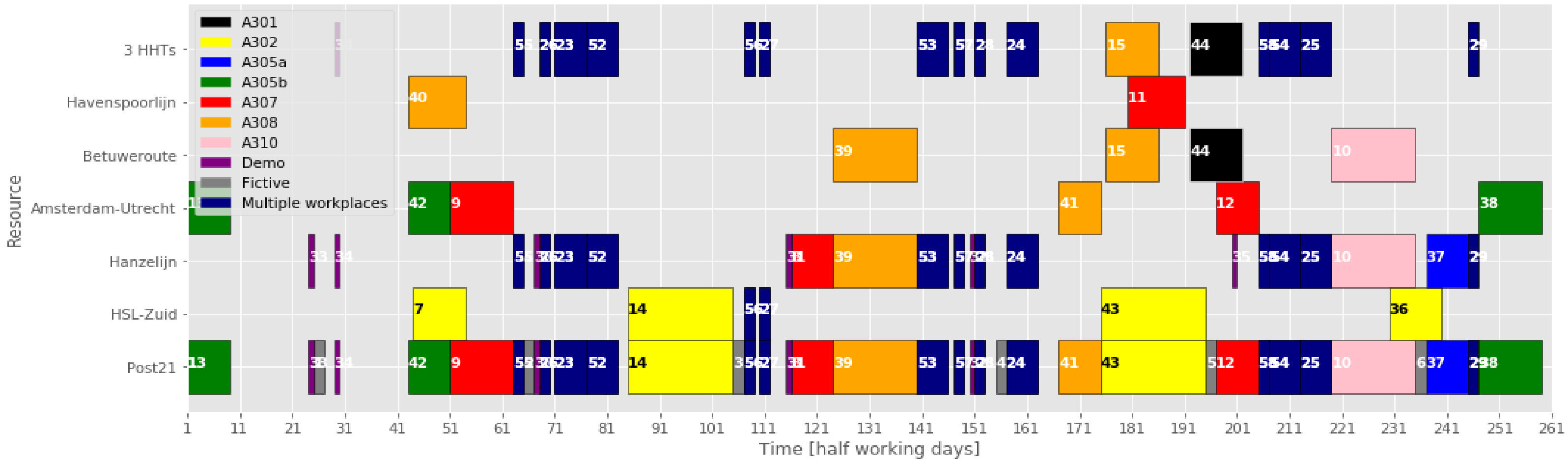

Figure F.12: Resource allocation of data set expanded with duplicates of all tests and trainings, objective to minimise the total tardiness and maximum run time = 300 seconds

Maximum Run Time of 3600 Seconds, or 7200 Seconds

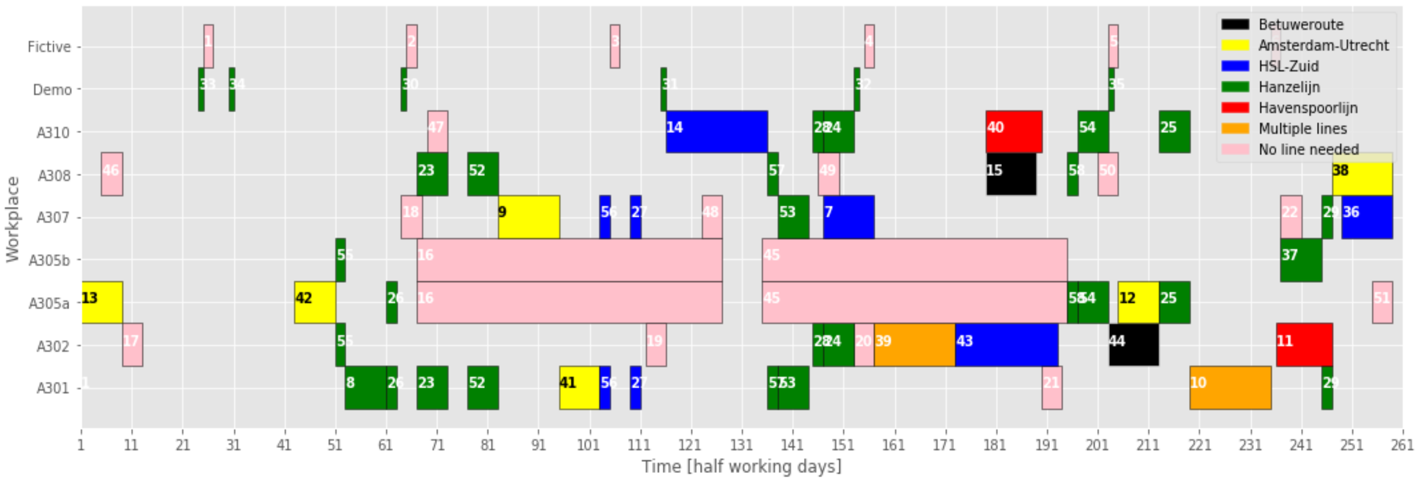

Figure F.13: Operational schedule of data set expanded with duplicates of all tests and trainings, objective to minimise the total tardiness and maximum run time = 3600 seconds, or 7200 seconds

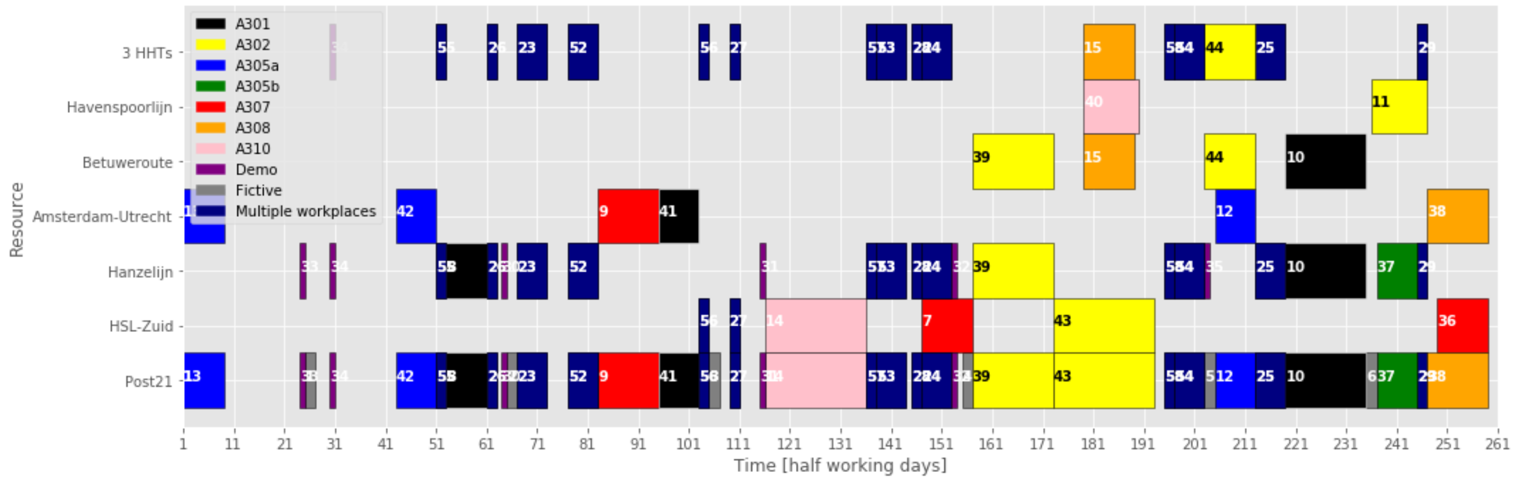

Figure F.14: Resource allocation of data set expanded with duplicates of all tests and trainings, objective to minimise the total tardiness and maximum run time = 3600 seconds, or 7200 seconds

Expansion 4: Duplicate All Tests, Demonstrations and Trainings

Objective of Minimising the Maximum Lateness

Maximum Run Time of 3600 Seconds

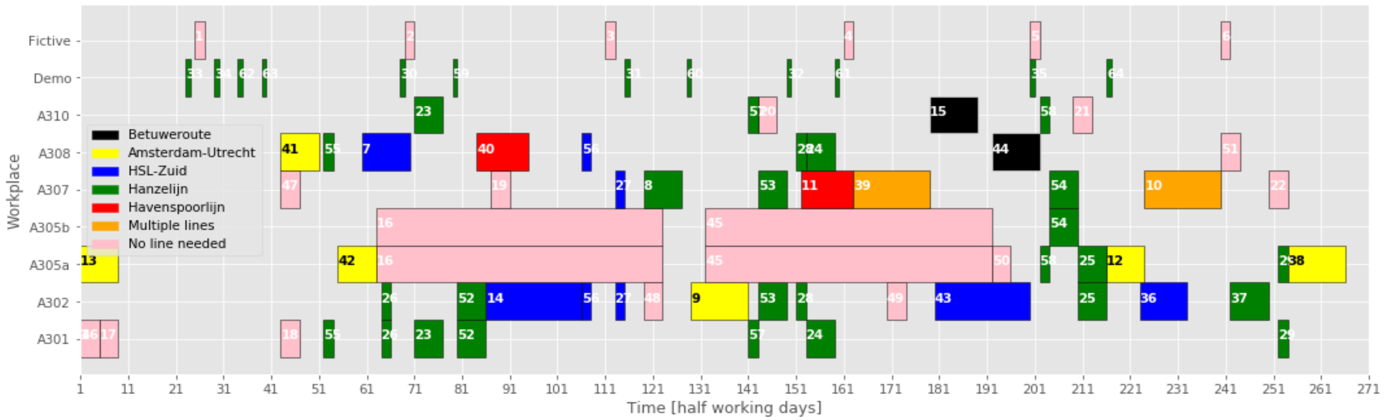

Figure F.15: Operational schedule of data set expanded with duplicates of all tests, demonstrations and trainings, objective to minimise the maximum lateness and maximum run time = 3600 seconds

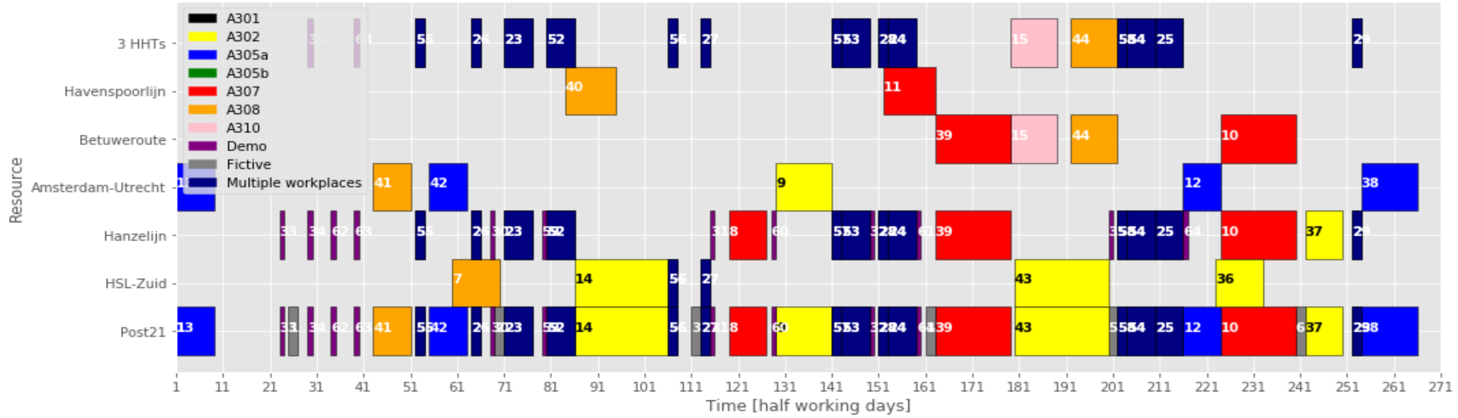

Figure F.16: Resource allocation of data set expanded with duplicates of all tests, demonstrations and trainings, objective to minimise the total tardiness and maximum run time = 3600 seconds

Objective of Minimising the Total Tardiness

Maximum Run Time of 300 Seconds

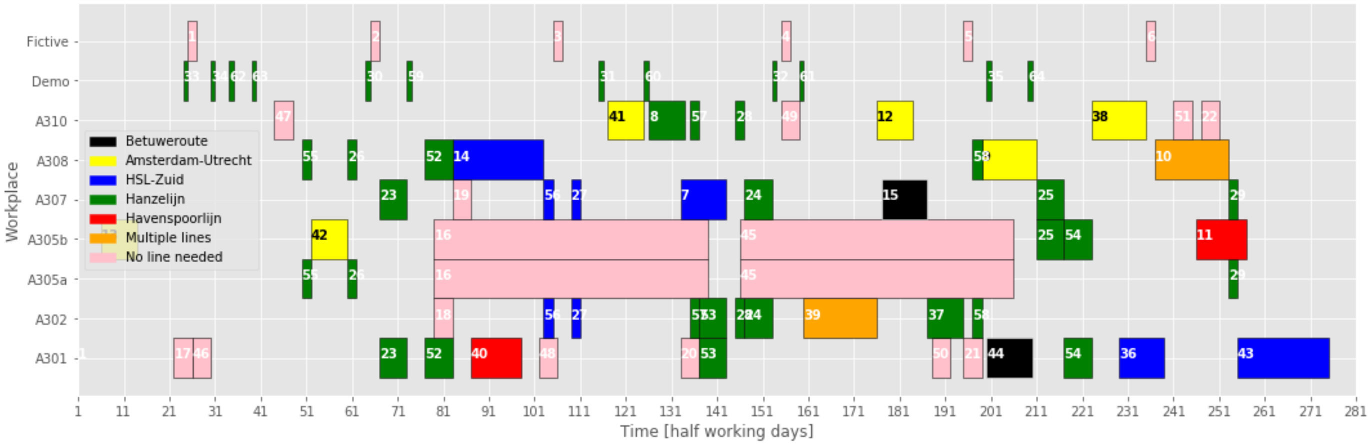

Figure F.17: Operational schedule of data set expanded with duplicates of all tests, demonstrations and trainings, objective to minimise the total tardiness and maximum run time = 300 seconds

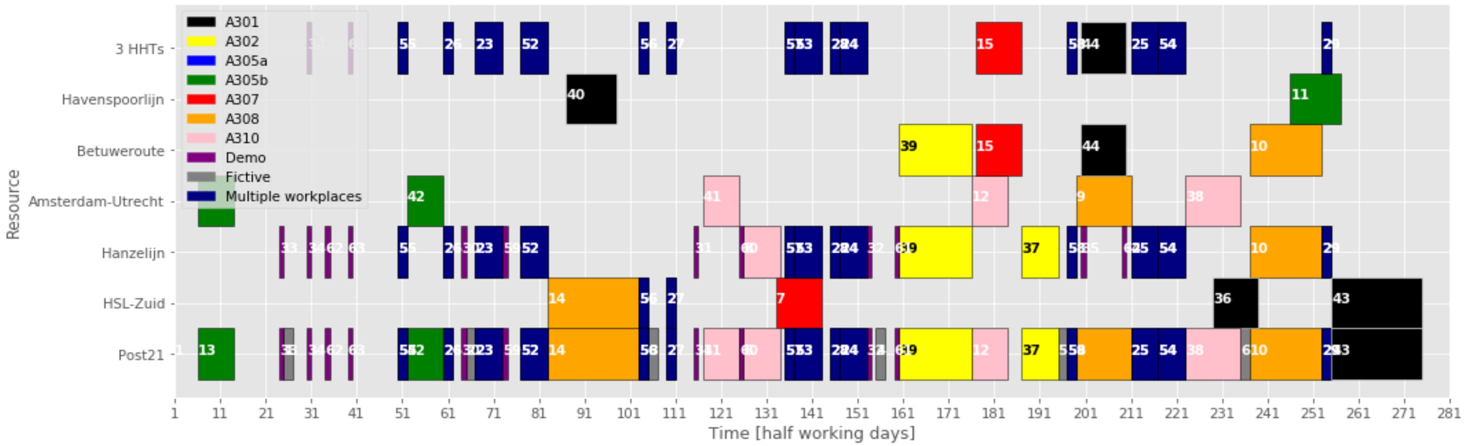

Figure F.18: Resource allocation of data set expanded with duplicates of all tests, demonstrations and trainings, objective to minimise the total tardiness and maximum run time = 300 seconds

Maximum Run Time of 3600 Seconds

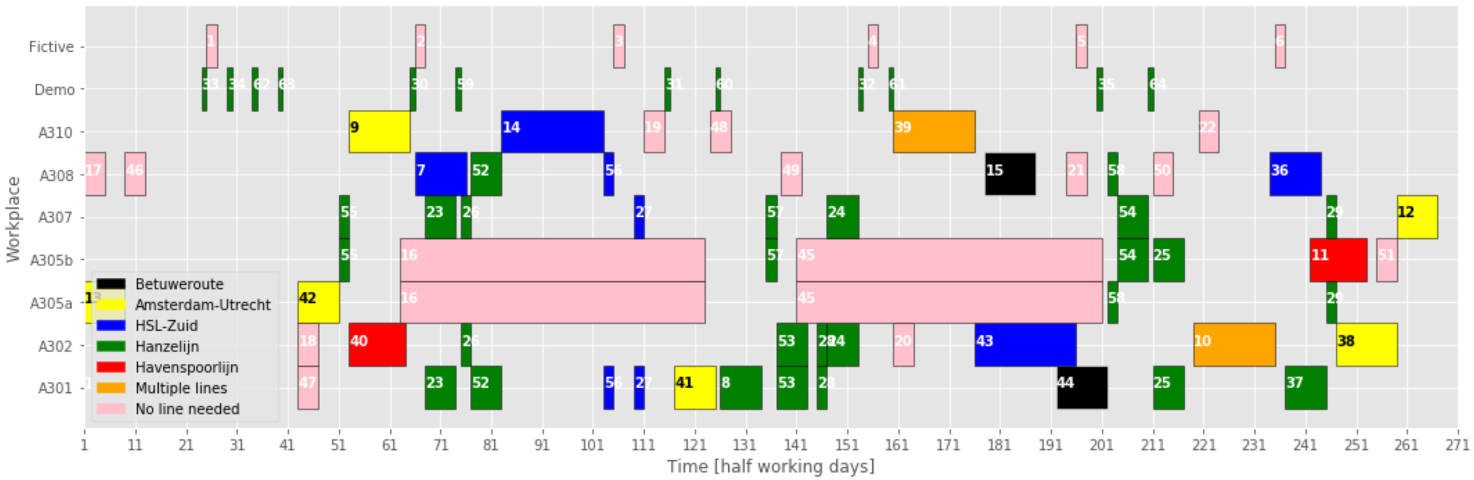

Figure F.19: Operational schedule of data set expanded with duplicates of all tests, demonstrations and trainings, objective to minimise the total tardiness and maximum run time = 3600 seconds

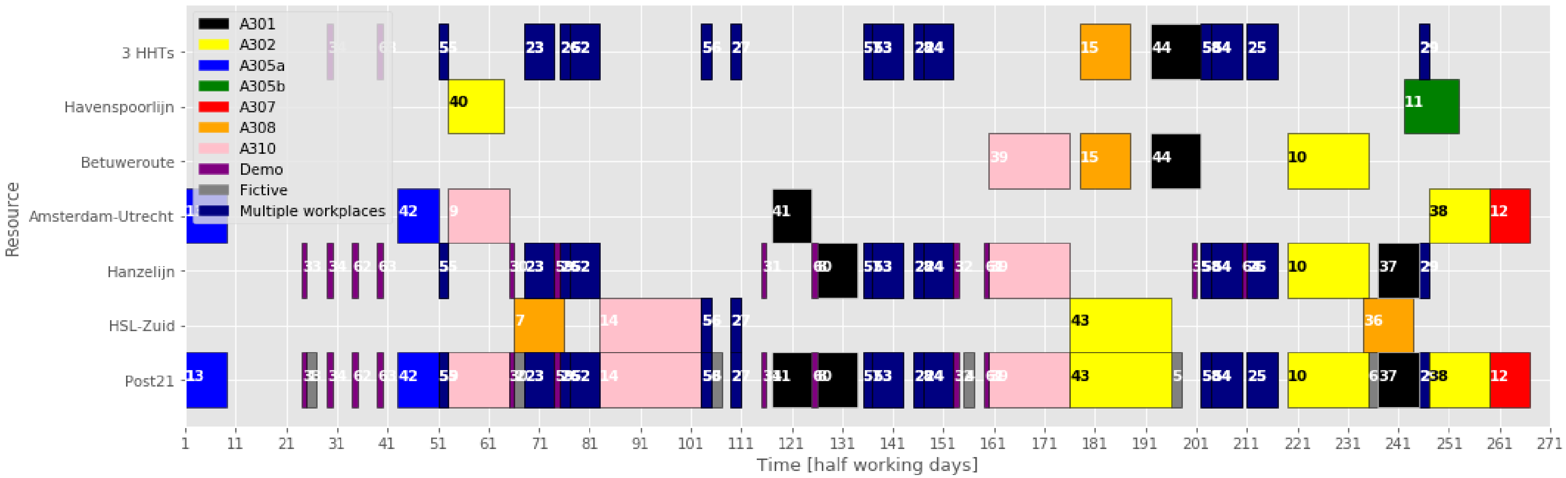

Figure F.20: Resource allocation of data set expanded with duplicates of all tests, demonstrations and trainings, objective to minimise the total tardiness and maximum run time = 3600 seconds

Maximum Run Time of 7200 Seconds

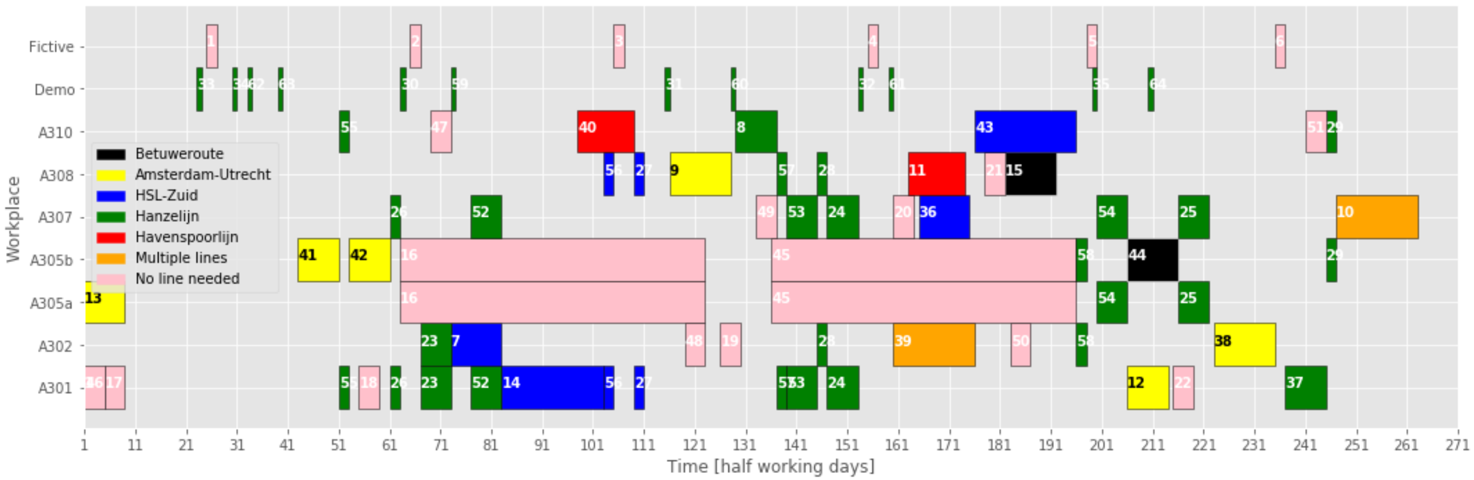

Figure F.21: Operational schedule of data set expanded with duplicates of all tests, demonstrations and trainings, objective to minimise the total tardiness and maximum run time = 7200 seconds

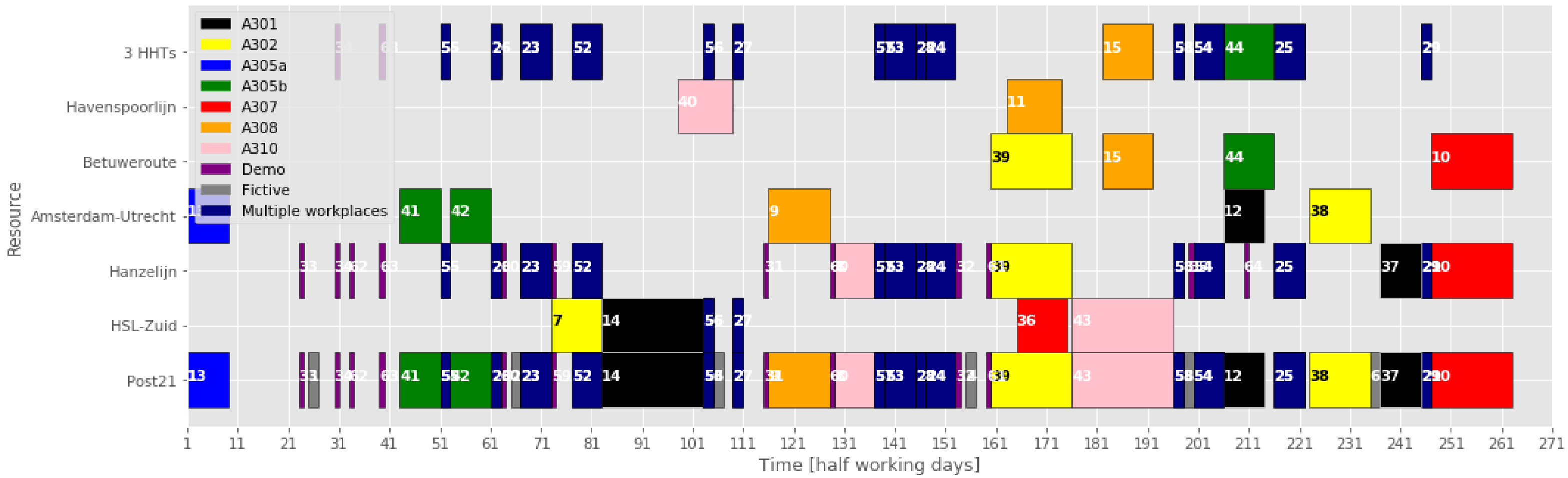

Figure F.22: Resource allocation of data set expanded with duplicates of all tests, demonstrations and trainings, objective to minimise the total tardiness and maximum run time = 7200 seconds

Expansion 5: Duplicate ‘Test Other’ Activities and Add ‘Test ESC’ Activities Three Times

Objective of Minimising the Maximum Lateness

Post21 Functionality Capacity Increased

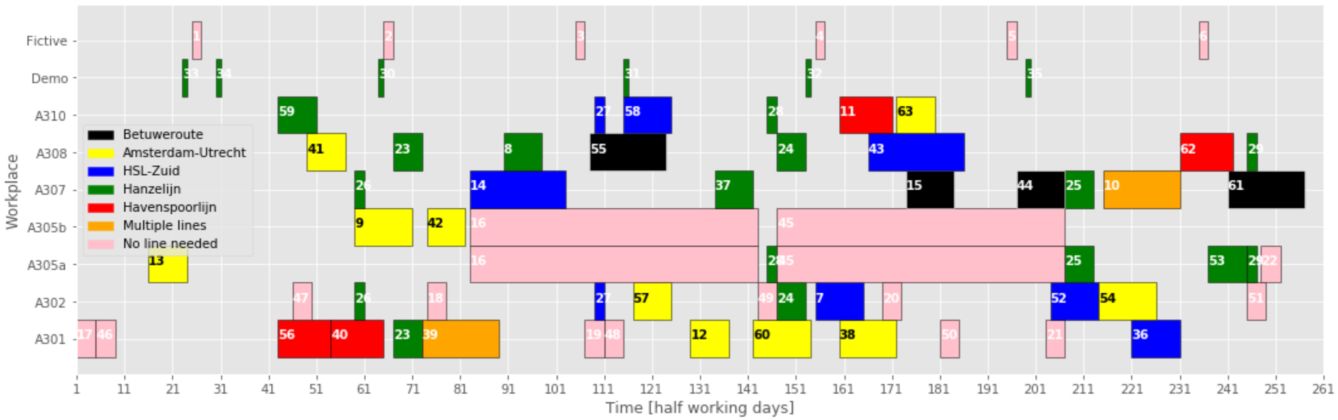

Figure F.23: Operational schedule of data set expanded with duplicates of the ‘Test Other’ activities and three times the ‘Test ESC’ activities, objective to minimise the maximum lateness and the Post21 Functionality capacity is two

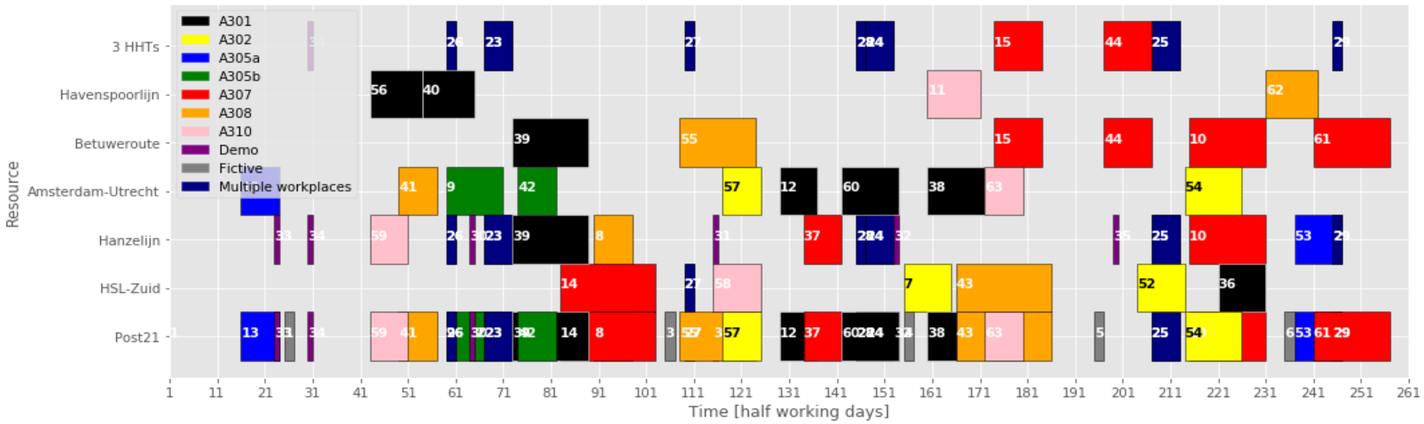

Figure F.24: Resource allocation of data set expanded with duplicates of the ‘Test Other’ activities and three times the ‘Test ESC’ activities, objective to minimise the maximum lateness and the Post21 Functionality capacity is two

Objective of Minimising the Total Tardiness

Post21 Functionality Capacity Increased

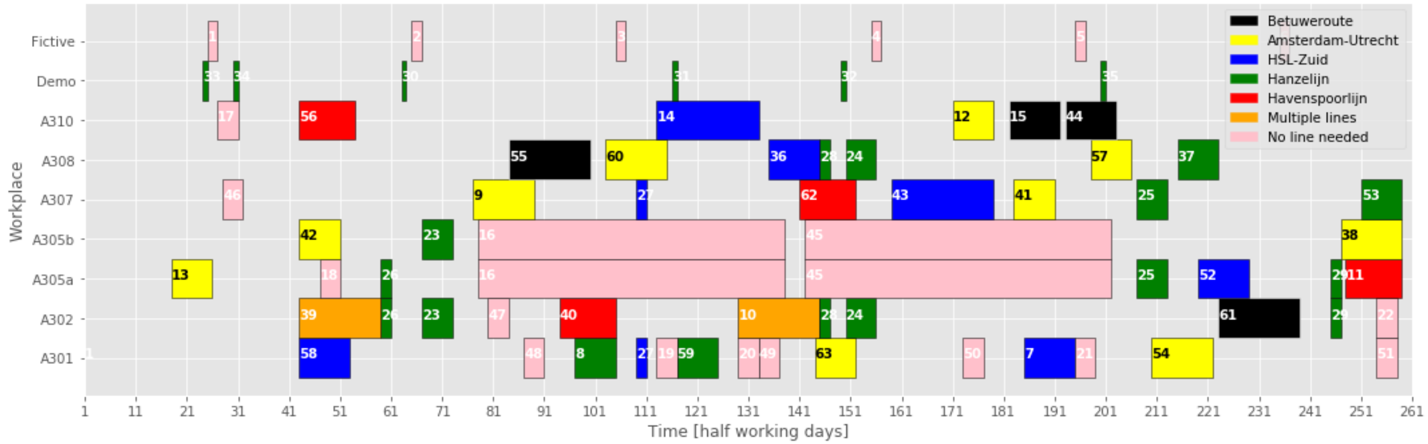

Figure F.25: Operational schedule of data set expanded with duplicates of the 'Test Other' activities and three times the 'Test ESC' activities, objective to minimise the total tardiness and the Post21 Functionality capacity is two

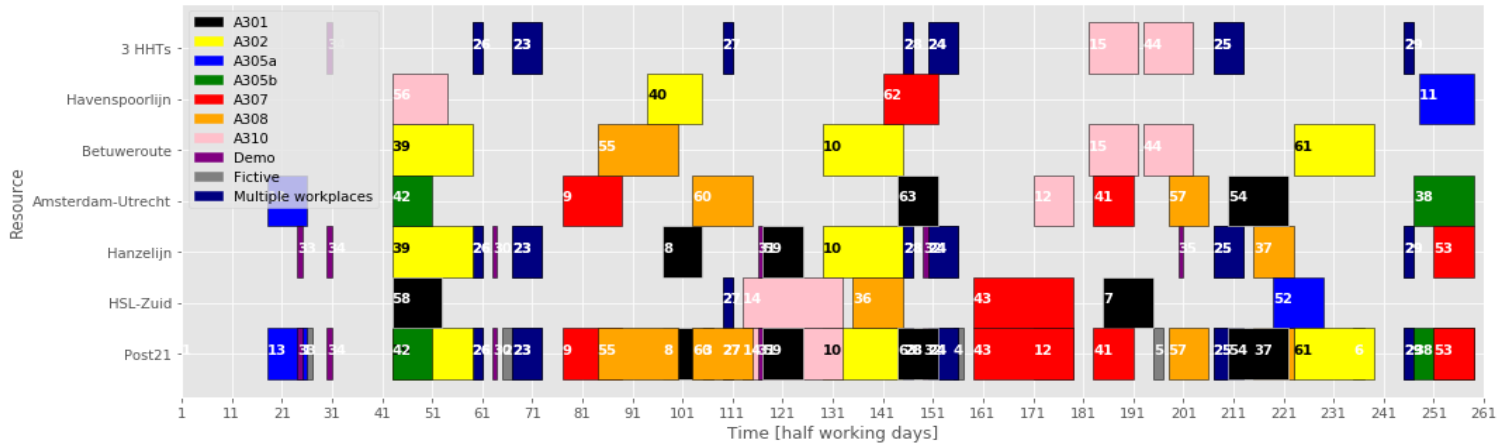

Figure F.26: Resource allocation of data set expanded with duplicates of the 'Test Other' activities and three times the 'Test ESC' activities, objective to minimise the total tardiness and the Post21 Functionality capacity is two