Master Thesis

Improving the scheduling of aircraft service tasks at KLM Aircraft Services

Author/Student F.J.J. Politiek

13 November, 2015

KLM

UNIVERSITY OF TWENTE.

Improving the scheduling of aircraft service tasks at KLM Aircraft Services

by

F.J.J. Politiek

Amsterdam, 13 November 2015

Master program	Industrial Engineering and Management	
Specialization track	Production and Logistics Management	
Faculty	Behavioural, Management and Social Sciences (BMS)	
Institute	University of Twente	
Company	KLM Royal Dutch Airlines	
Department	KLM Aircraft Services	
Committee Members	Dr.ir. J.M.J. Schutten	University of Twente
	Dr.ir. L.L.M. van der Wegen	University of Twente
	Ir. M.H.G. Bovenkerk	KLM Aircraft Services
	Ir. M. Vos	KLM Aircraft Services

"The airplane stays up because it doesn't have time to fall." – Orville Wright

Preface

In the past eight months I have been a fuel truck operator, catering truck operator, aircraft towing operator, and water truck operator for one day. My time at KLM was a great experience and I enjoyed it to the fullest. Before I started with the master Industrial Engineering and Management at the University of Twente I studied Aeronautical Engineering. To do my master's research at KLM was thus a great opportunity for me.

KLM is a large company, and large companies are often not known by their fast and simple processes. In a large company it takes often a long time and huge effort to accomplish the goals that someone has at the start of a project. I started with the task to improve the scheduling of aircraft service tasks. This sounds maybe easy, but where to start? During my research I found out that the scheduling of aircraft service tasks are influenced by many parameters, not only mathematically, but also by human factors. However, I found my way through this research and I am more than happy with the results.

A preface without acknowledgements is not a preface. I start with my supervisors from KLM Aircraft Services Mark Bovenkerk and Maarten Vos. The help in finding the right people in the organization, the discussions about the assignment and other KLM related processes, and the arrangement of taster days in the operation made it to a great experience and helped me to finish this thesis. I next thank my supervisors from the University of Twente Marco Schutten and Leo van der Wegen. They always provided me valuable feedback on the research content and I always left Enschede with new ideas, it definitively helped me to improve my thesis. At last I thank my colleagues from KLM AS Tony, Pascal, Elizabeth, Iwi, Marian, Tonnie, Erik, Glen, and Wilfried for the good working atmosphere, the non-work related discussions, and everyday lunches.

Finally, I thank my younger brother Hylke Politiek for sharing his apartment and the great time that we had together in Amsterdam.

Amsterdam, 13th of November 2015.

Feike Politiek

Summary

Introduction

This research focuses on scheduling improvements of aircraft service tasks at KLM Aircraft Services. The scheduling of aircraft service tasks is done by dispatchers (in Dutch: regisseurs). These dispatcher are helped by the coordination and scheduling of aircraft service tasks by CHIP (Communicatie & Hub Indelings Programma). Only aircraft towing, aircraft pushbacks, aqua services, toilet services, and aircraft refueling tasks are scheduled with CHIP.

Problem description

The aviation world is a highly competitive market where every minute counts. As increasing number of flights and shorter turnaround times make the timely completion of aircraft service tasks more and more important. Currently there is a common feeling under AS management that CHIP is underperforming. Instead of scheduling pro-actively and constantly keeping track of future critical events, the dispatchers and CHIP are scheduling re-actively. The underperformance is also caused by limited knowledge and insight in what actually happens inside CHIP. These underperformance tendencies are the basis of the research goal of this thesis. The research goal is to provide insights into the scheduling of aircraft service tasks and to propose schedule improvements.

Approach

We investigate the CHIP scheduling process during the day, describe the scheduling problem, and define a new measurement method to measure the performance in the future. This measurement method consists of a dynamic workload graph tool and a dynamic performance measurement tool. Based on the dynamic workload graph we are able to identify critical scheduling issues and propose scheduling improvements. With the dynamic performance measurement tool we developed a new way for the dispatcher to see critical events in advance. We evaluate the dynamic workload graph and dynamic performance measurement tool by using the operational data of a specific day.

Important results and findings

We provide insights into the scheduling behavior over time and based on these insights we are able to propose scheduling improvements. We also show that the scheduling process can be improved by using the dynamic workload graph and dynamic performance measurement tool within the operation. From the dynamic workload analysis and dynamic performance measurement tool, we present the main findings:

- Dispatchers are able to see critical time windows in advance when using the performance measurement tool and dynamic workload graph.
- We showed that operators are sent home before their shift ends. To facilitate this the dispatcher re-schedules tasks at the end of a shift to the next shift without considering the future workload. This leads to unnecessary increases in the workload and should therefore be avoided.
- CHIP uses many optimization criteria to optimize the assignment of tasks to operators. Due to this complex optimization the dispatcher should make as less changes to the schedule as needed. Since the dispatcher is unable to evaluate all optimization criteria in a short time.
- We showed that breaks are scheduled at the last moment and often on their latest end time. We propose a break schedule that gradually assigns breaks to operators and that is fixed at the start of the day. This leads to a more predictable break schedule and decreases the nervousness of the scheduling system.
- During the shift change between 14:00 and 14:30 on average one task is completed. A shift change schedule that is gradually implemented will lead to more completed tasks and higher resource utilization during the shift change.

Recommendations

We recommend KLM AS to discuss and further improve the performance measurement tool and workload graph together with all KLM AS dispatchers and DMAs. We believe that the scheduling process and schedule can be improved if these tools are used by the dispatcher and DMA. However, a new tool will only be successful if one has the full collaboration and acceptance of the users. KLM AS should invest enough time to demonstrate and explain the importance of the tool to users.

We also recommend KLM AS to discuss the early departure of operators that is facilitated by the dispatchers. In this discussion KLM AS should use the dynamic workload graph to show the effects of re-scheduling tasks to a next shift to facilitate the early departure of operators.

On the long-term KLM AS should convince and learn all dispatchers that CHIP is able to make a better task assignment than dispatchers, even if CHIP is scheduling tasks against the scheduling logic of a dispatcher. This is due to the fact that CHIP is able to optimize the assignment of tasks against multiple optimization criteria.

Contents

List of Abbreviations

Abbreviation	Meaning
A0	Arrival performance
ADC	All Doors Closed
AIBT	Actual In-Block Time
AS	Aircraft Services
CHIP	Communicatie Hub Indelings Programma
D0	Departure performance
DAM	Duty Area Manager
DARP	Dial-a-Ride Problems
DHM	Duty Hub Manager
DMA	Duty Manager Aircraft
EXIT	Estimated Taxi-In Time
EXOT	Estimated Taxi-Out Time
FIRDA	Flight Information Royal Dutch Airlines System
GS	Ground Services
HCC	Hub Control Center
HTO	Human Technological Organization
TOBT	Target Off-Block Time
TSP	Traveling Salesman Problem
VRP	Vehicle Routing Problem
VRPSD	Vehicle Routing Problem Stochastic Demand
VRPSTT	Vehicle Routing Problem Stochastic Travel Times
VRPTW	Vehicle Routing Problem Time Windows

1 Introduction

This thesis is written during an eight month internship at KLM Aircraft Services. We first introduce Aircraft Services with the following question: Have you ever thought while you were sitting in an aircraft which services were needed before your aircraft could leave? Most of the readers of this thesis probably did not. The aircraft must be fueled, cleaned, supplied with water, catered, de-iced if necessary, and in most cases pushed back from the gate. At Amsterdam Schiphol Airport these services are provided by KLM Aircraft Services (AS).

The aircraft services are controlled by a department of KLM at Amsterdam Airport. Within this department dispatchers (in Dutch: 'regisseurs') coordinate the operation of aircraft services separately. The scheduling and coordination of these services is a complex task. During the day there are a lot of disturbances that ruin a predefined schedule of tasks. The dispatcher is helped in the coordination of these processes by the CHIP system (In Dutch: Communicatie & Hub Indelings Programma). CHIP contains all tasks related to a service that an operator needs to do during the working day and automatically dispatches jobs to operators.

The aim of this research is to improve the *control* and *planning* of aircraft services. In this research we analyze the work environment of an AS dispatcher and we perform a literature study to make suggestions for improvement in the coordination of aircraft services.

Section 1.1 introduces KLM and specifically KLM AS. In Section 1.2 we explain the research motivation and in Section 1.3 the research objective and research questions. Section 1.4 presents the outline of this thesis.

1.1 Background

Section 1.1.1 describes the KLM in general. Section 1.1.2 addresses all aircraft services in further detail.

1.1.1 KLM

The KLM is started and established in 1919 as 'Koninklijke Luchtvaart Maatschappij'. Nowadays KLM is the oldest airline in the world that is operating under its original name. The huge growth of KLM is remarkable to mention. In the first operating year KLM transported 345 passengers and 25,000 kilograms freight. The KLM annual report of 2014 records a total of 40 million passengers who travelled

Figure 1: KLM brand mark

with KLM, Air France, or its partners. The merger between KLM and Air France was established in 2004 and KLM is now part of Air France KLM Group.

1.1.2 KLM Aircraft Services

KLM Aircraft Services is part of KLM Ground Services (GS). GS manages all hub operations at Amsterdam Schiphol Airport. An airport is called a hub when an airline is using this airport as a transfer point to get passengers to their intended destination. AS offers the following services:

- De-icing/Anti-icing
- Aircraft Towing
- Aircraft Pushbacks
- Aqua Services
- Toilet services
- Aircraft Refueling
- Cabin quality/cleaning
- Catering services

1.1.3 Hub control center

All activities and processes from the daily flight operations are monitored and controlled by AS dispatchers in the Hub Control Center (HCC). The HCC in general is responsible for all flight operations of KLM and partners and has the following primary tasks:

- Managing the critical resources on the day itself.
- Managing the flights on hub Amsterdam Airport in cooperation with the Operational Control Center (OCC).
- Responding to emergencies and operational crisis situations.
- Preparing and evaluating the hub performance.

Within the HCC there are different functions. Appendix B describes those in further detail. The Duty Manager Aircraft (DMA) and Aircraft Services dispatchers are part of this research. Section 1.1.4 briefly describes the role of the DMA. Section 1.1.5 addresses the role of the Aircraft Service dispatcher.

1.1.4 Duty Manager Aircraft

Within the HCC there is a Duty Manager Aircraft (DMA). The DMA is responsible for the AS dispatchers. The DMA is responsible for the communication between AS dispatchers and informs them about possible calamities and disturbances.

1.1.5 Aircraft Services dispatcher

The dispatchers of AS are responsible for the operational scheduling of aircraft services during the day. Each aircraft service is controlled separately by a dispatcher. The dispatcher is helped with the control of services by the CHIP system. CHIP is a computer program that helps the dispatcher to plan tasks in time. CHIP contains all tasks related to a service that an operator needs to do during the working day and automatically dispatches jobs to operators. This dispatching is based on many different input parameters.

The operational scheduling by dispatchers can be positioned in the planning and control framework of Hans et al. (2011) as follows (see Figure 2).

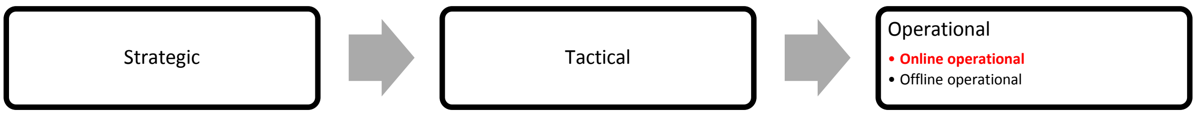


Figure 2: Positioning of dispatcher in the hierarchical planning & control framework, based on Hans et al. (2011)

According to Hans et al. (2011) online operational planning involves "the control mechanisms that deal with monitoring the process and reacting to unforeseen or unanticipated events". This definition can be translated to the tasks of the dispatcher. CHIP together with the AS dispatcher functions as a control mechanism to monitor the aircraft service process. The dispatcher and CHIP should react and anticipate on unforeseen events.

1.2 Research Motivation

Currently there are multiple reasons to start a research into the online operational scheduling of AS services. Shorter turnaround times and an increasing number of flights put high pressure on the planning and control on the day of execution. There is a common feeling under AS management that the CHIP system is underperforming. This underperformance is partly due to the actions and interventions of the dispatcher who is responsible for the coordination of the aircraft services. This underperformance is a direct cause for further analysis into the work environment of the dispatcher. There is also a lack of insight into the working methods of the dispatchers and discussion about the responsibilities and mandate of a dispatcher. Of course what they do is known, but how and based on what are questions that are not fully clear at this moment and that gives potential for improvement.

In an ideal world where all future events are known it is less difficult to construct a 'good' schedule for the aircraft services. The dispatcher can schedule pro-actively and anticipate on disturbances and adjust the schedule if needed. However, in the real world the dispatcher has to deal with a lot of unknown events. These unknown events are often the cause that CHIP cannot schedule automatically. In those cases the dispatcher schedules manually.

The dispatcher often schedules the tasks re-actively instead of pro-actively. When a job is not planned automatically by CHIP, the dispatcher waits until the equipment is available. After it is available the dispatcher gives the job to the operator with the free equipment. This is called scheduling re-actively. Scheduling re-actively often results in a poor schedule.

At this moment it is not possible to assess the quality of a constructed plan at the end of the day. The quality cannot be assessed, because the performance measures that are used to measure the quality are not clear. This research will construct and evaluate performance measures for quality to use and measure the effect of proposed improvements in the scheduling process. Summarizing, there are opportunities for further improvements in the online scheduling.

1.3 Research objective and questions

The goal of this research is to improve the process of online scheduling of aircraft services within the Hub Control Center. The improvement proposals will be gathered by a thorough analysis into the work environment of an AS dispatcher and literature study. The improvements will lead to better *control* and *planning* of aircraft services on the day of execution. These improvements contribute to the priorities and performance goals that are set by the hub Schiphol.

This research focusses on the current working methods and structure within the HCC. It does not assess a different organization structure or the use of other IT programs to control the operation. This results in the following main research question:

"How can KLM Aircraft Services improve the online scheduling of aircraft services, within the current organization and IT structure?"

The first step is to obtain a clear understanding about the current working methods of AS dispatchers within the HCC.

Research question 1

What is the current situation regarding online scheduling of aircraft services?

We divide research question 1 into the following sub questions:

i. What information and systems are used by an AS dispatcher?
ii. What are the current working methods of an AS dispatcher?
iii. What are the responsibilities, duties, and positions within KLM of an AS dispatcher?

To understand the current situation and working methods that are used within the HCC we spend several weeks at the HCC. We closely observe, question, and describe the operation and talk with the DMA and dispatchers about their working day.

Research question 2

What is described in literature that can help to improve the planning and control of aircraft services?

i. How can the scheduling problem of aircraft services be characterized?
ii. Which process information is needed for these scheduling problems?
iii. What are the human and organizational aspects in scheduling?
iv. What is needed to define good performance measures?

We start defining the scheduling problem of aircraft services related to the scientific literature. From this definition we argue and try to find out which information is needed to solve these scheduling problems. Next, the answer to sub question iii describes how scheduling is influenced by the scheduler and the organization. Sub question iv is formulated as basis for research question 3.

Research question 3

How can we measure the performance and quality of a schedule?

i. What is considered as schedule quality for different stakeholders?
ii. How can we assess and measure the scheduling performance over time?

By answering research question 3, we present performance measures that quantitatively describe the quality of a created schedule over time for the aircraft services. We provide insights into schedule changes during a day. To find the data that is needed to construct the performance measures we ask the help from process analysts of KLM AS.

Research question 4

Which improvements can be made in the online scheduling of aircraft services?

i. What are the potential improvements for the decision of assigning tasks to operators?
ii. How can AS and the HCC implement the proposed improvements?

By answering research question 4, we present potential improvements in the online scheduling of aircraft services. We also discuss how the proposed improvements and performance measures can be applied and implemented.

1.4 Outline

The remainder of this thesis is structured as follows:

Chapter 2 describes the current situation of online scheduling of aircraft services. We discuss the current situation, information systems, and the dispatcher in general. In Chapter 3 we review the literature and characterize the scheduling problem. We also present how to define good performance measures, and how to perform a stakeholder analysis. Chapter 4 provides performance measures for the quality of a created schedule for the aircraft services. We construct two tools that provide insights into future schedule performance. In Chapter 5, we combine our knowledge from literature and performance from practice to propose improvements. At last, Chapter 6 presents the conclusion and recommendations for further research.

2 Current situation

This chapter describes the current situation of online scheduling of aircraft services. Section 2.1 presents and briefly explains all aircraft services. Section 2.2 describes the decision support system CHIP. We explain the different types of tasks, the dispatching of tasks, and the optimizer behind CHIP. Section 2.3 presents an overview of the responsibilities and position within the organization of a KLM AS dispatcher. Section 2.4 ends with a conclusion.

2.1 KLM Aircraft services in detail

Each aircraft service has its own dispatcher that controls the daily operation. The dispatchers are sitting together so that they can interact and discuss with each other easily. Section 1.1.2 enumerates all aircraft services. To understand the current working methods of an AS dispatcher we first explain each aircraft service in more detail. Only aircraft towing, aircraft pushbacks, aqua services, toilet services, and aircraft refueling are coordinated with CHIP.

De-icing/Anti-icing

Figure 3: Aircraft receiving de-icing treatment

De-icing is a treatment where de-icing fluid is sprayed onto the aircraft to remove snow and ice from the critical areas of the airplane. Critical areas are for example the wings and the stabilizers. There are two major reasons why de-icing is necessary. The first reason is to ensure the free movement of the steering surfaces of the aircraft. The second reason is that a possible layer of ice on the wings can disrupt the airflow around a surface which can lead to a loss of lift. The de-icing department is located at a remote area of Schiphol. The aircraft taxi towards this position and the engines can stay running while undergoing the treatment. In total there are 24 Safeaero de-icing vehicles available. Figure 3 shows a KLM aircraft that receives a de-icing treatment by de-icing vehicles.

Aircraft towing

Aircraft towing and aircraft pushbacks are operated by the KLM Aircraft Towing & Pushback services department. Towing is needed to move aircraft from and to the buffers and gate positions on Schiphol Airport. A buffer is an outside position on Schiphol Airport where aircraft can be parked. When an aircraft has a long ground time the buffer is used to free up space at the gate positions. Towing is also needed when aircraft are located at the maintenance department at Schiphol East. Figure 4 shows an aircraft towing tug.

Aircraft pushbacks

A pushback is an operation where the aircraft is moved backwards from the gate. An aircraft needs a pushback to leave the gate, because the aircraft is not capable of riding backwards. However, some aircraft are: they can use reverse thrust from its engines, but this can cause severe damage to the terminal and gate. A pushback is performed by a tug (see Figure 4).

Figure 4: KLM tug

Figure 5: KLM Toilet truck

Figure 6: Fuel bowser

Figure 7: Catering services

Aqua services & Toilet services

Aqua service is responsible for the supply of potable water to the aircraft. The water is delivered by small trucks with a water tank. The toilet service is separated from the water service and operated with a different truck. Figure 5 shows a KLM toilet truck. The truck has a platform that can lift so that the toilet drains can be reached.

Aircraft refueling

The refueling department is responsible for the supply of fuel to the aircraft. The department has three large bowsers ($80m^3$) and 15 smaller bowsers ($40m^3$) that refuel the aircraft on remote stands where the fuel hydrant system is not available. Figure 6 shows a bowser that is refueling an aircraft. Many gates at Schiphol have a fuel hydrant point. In those cases it is not necessary to reach the aircraft with a large truck. To use this system 21 dispensers are available. A dispenser is a truck that has the equipment to use the fuel hydrant system, such as hoses, couples, and pumps.

Cabin quality/cleaning

Aircraft cleaning activities are outsourced to Asito and Klüh that are both cleaning companies that are not part of KLM. Both cleaning companies operate autonomously, however they are supervised by a contract manager from KLM. Both companies have a dispatcher to control the daily operation.

Catering services

The catering activities are outsourced to a subsidiary company KLM Catering Services (KCS). KCS is responsible for the supply of meals and non-food items that can be found in the aircraft. Catering trucks are used to supply the aircraft. Figure 7 shows a catering truck. The container is lifted to align the aircraft doors with the container which makes loading easier.

2.2 Information system

This section describes the information system CHIP that is used for dispatching aircraft service tasks to resources.

2.2.1 CHIP

CHIP is used by the dispatcher to control the operation of a specific aircraft service. CHIP is a computer program that is built by INFORM[1]. INFORM is a company based in Aachen Germany which is specialized in intelligent planning and logistics decision-making software. CHIP is a tool for decision support, not for decision take-away. It does not replace the human dispatcher, but helps the dispatcher making the best assignment at that time. We start with an explanation of the basic idea behind CHIP.

Figure 8 shows six operators (A to F), a timeline where t denotes the current time, and several unplanned jobs. With jobs we mean specific aircraft service tasks, for example refueling, potable water supply etc. The jobs are represented by grey blocks. All jobs must be planned and assigned to AS operators. However, there are a lot of restrictions when dispatching jobs to operators. Time windows, shift roster limitations, breaks, flight schedules, and equipment in use for example. If a dispatcher should manually collect all information that is needed to dispatch a job and interpret this information continuously, he or she will be unable to schedule aircraft service jobs.

[1] More information can be found on www.inform-software.com/products/

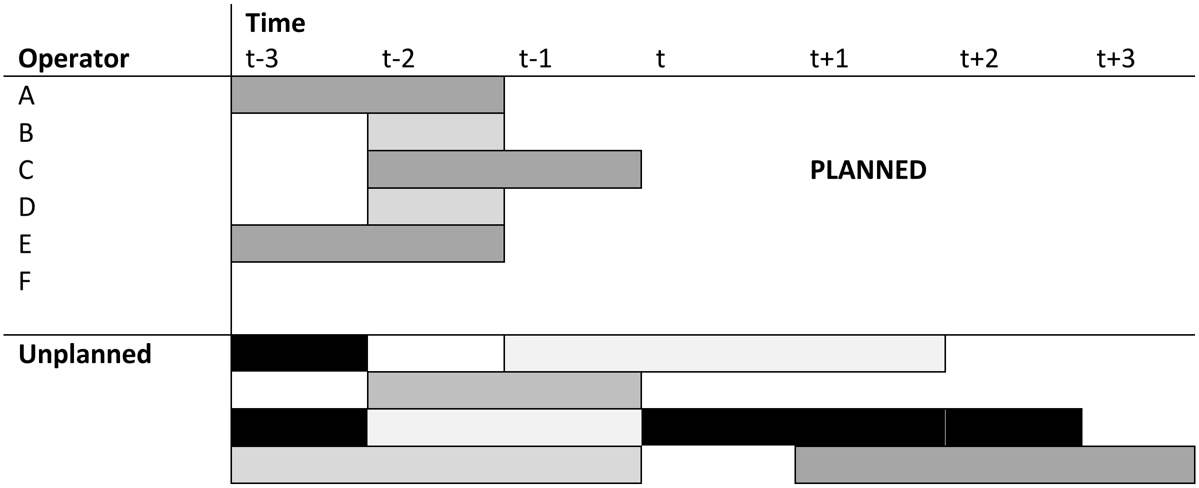


Figure 8: Basic scheduling representation CHIP

Information input CHIP

Figure 9 shows the input sources of CHIP. The input of CHIP consists out of the flight information system, base data, and roster control.

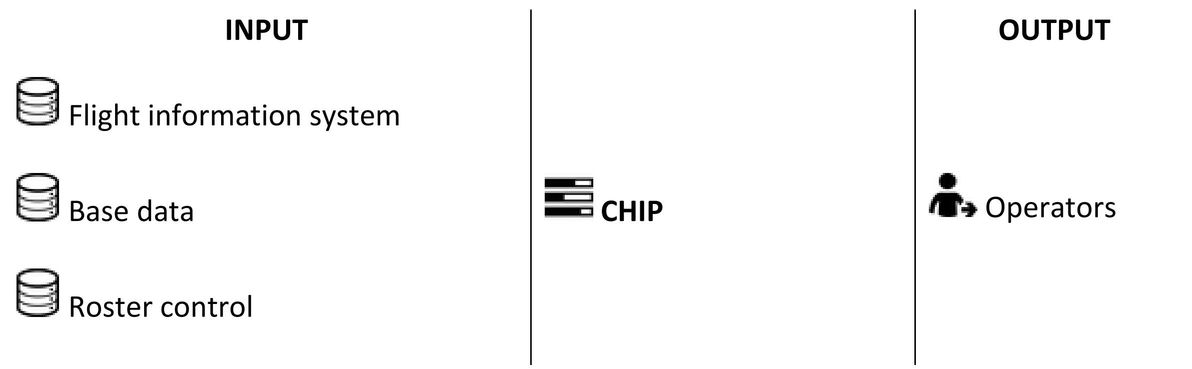


Figure 9: Input information CHIP

The actual flight information is continuously retrieved from the Flight Information Royal Dutch Airlines system (FIRDA). The base data contains data that defines the operative environment. Examples of base data are distances between positions, employee information, shift types, qualifications, airlines, and aircraft types. This base data is not automatically changed or updated by the information from FIRDA. Within the data a distinction can be made between static and dynamic data. The information from FIRDA is dynamic data, because it is updated continuously. Dynamic data describes what takes place in the base data defined environment. Examples of dynamic data are flights, tasks, alerts, and shifts. This dynamic data is updated by FIRDA, but also due to the actions of the dispatcher and optimizer within CHIP. If for example CHIP dispatches a task or the dispatcher dispatches a task manually, there is a change in the dynamic data. The roster control holds information about shifts, breaks, and amount of personnel available. The tasks as displayed in Figure 8 are created by CHIP based on the input information as displayed in Figure 9. Tasks are continuously created, updated, and deleted based on the real time information from its input sources.

Types of tasks

A task can be flight related or non-flight related. In the first case a task is linked to actual flight events, whereas the non-flight related tasks do not have a link with a flight event. This can be for example a standard daily activity. CHIP considers also single-flight and multi-flight related tasks. As the name suggests single-flight tasks are related to one single-flight event, whereas multi-flight tasks are related to multiple flights. For instance a check-in task for multiple flights is considered as a multi-flight task. CHIP also makes a distinction between time-interval-oriented and moment-oriented tasks. A time-interval-oriented task is a task that must be performed during a given time-interval. For example, a Boeing 747-400 must be refueled during its ground time. Moment-oriented tasks are tasks that must be performed at a given moment in time, or start at a given time.

CHIP also considers main tasks and sub tasks. Sub tasks that belong to the same operation are grouped together in a main task. Important to mention is that only sub tasks can be assigned to resources. Main tasks are considered as a structure within the data. When tasks are created, the next step is to dispatch the tasks to operators.

Dispatching tasks

CHIP creates tasks for the following day during the night. It receives the initial flight data of the next day and based on that information it creates the tasks for the next day. During the day these tasks are updated due to the real time flight information which can lead to the generation of new tasks, changing tasks or subtasks, or deleting tasks and subtasks.

Starting from Figure 8, the unplanned tasks must be assigned to one of the operators. The operators are considered as resources. Each task has a set of different qualifications that are needed to perform this task. For example for the refueling of a Boeing 747-300 an operator needs a specific license. This can be considered as a qualification for the task. The qualifications are split up in mandatory and non-mandatory qualifications. When a task is assigned to a resource, it must comply with all mandatory qualifications and to some degree with the non-mandatory. The resources have also specific qualifications, for instance a tank operator with a specific license, or a bowser with a capacity of $40m^3$. When allocating a task to a resource, the qualifications of both are compared to each other; when there is a match between these two, the task can be assigned to the resource.

However, if there are more resources that match with the qualifications of a specific task the system has to make a decision which resource to assign. This assignment is performed by the optimizer within CHIP. Table 1 shows which data is used by the optimizer when making an assignment.

Tasks	• Time windows • Duration • Travel time • Priority • Task type	• Workload • Work area • Task requirements • Teaming
Shifts	• Start and end time of a shift	

Table 1: Optimizer data

Time windows

CHIP considers time windows when a certain task has to be planned. The time window is based on the earliest start and latest end time requirement. These requirements are related to the actual landing time and scheduled departure time. Figure 10 schematically presents the time window when a job has to be planned. The EXIT time is the Estimated Taxi-In Time. It is the time that an aircraft spends taxiing between the run way and parking place. The actual in-block time (AIBT) is the actual time and date when the parking brakes of the aircraft have been engaged at the parking position (EUROCONTROL, 2009). The end of the task time window is marked with the target off-block time (TOBT). At that point in time the ground handling process is concluded and the aircraft is ready to start-up and pushed backed from the parking position. The TOBT is a forecasted value. The Estimated Taxi-Out Time (EXOT) is the outbound taxi time. If a task is planned later than its latest end time requirement, the aircraft will be delayed.

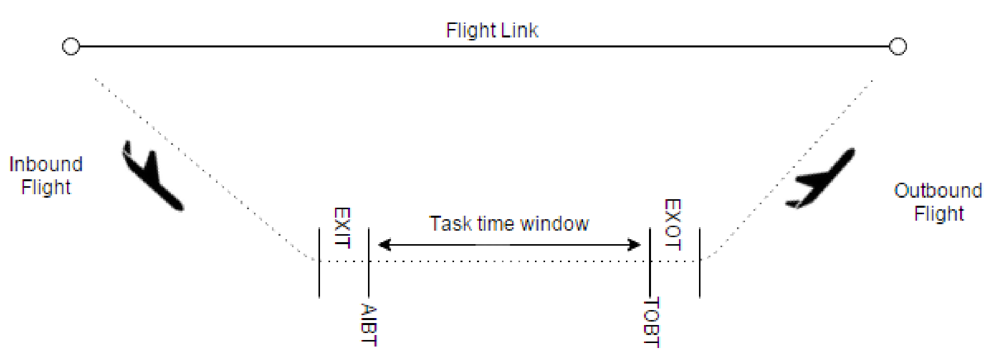


Figure 10: Earliest start and end time based on Harmsen (2012)

The optimizer in more detail

CHIP considers and creates a schedule for the next four-hour time window. Before task allocation, the optimizer starts with a search for all combinations of tasks and resources that are allowed based on the qualifications of the resource and task. Second, a cost value is associated with each task and resource combination based on a cost function. The cost function consists of different parameters. The exact parameters of this cost function are not given by INFORM. The optimizer tries to minimize the total costs of all task allocations to resources. The cost function is configurable by the system owner with predefined so called alpha parameters.

2.3 The dispatcher

In Section 2.2.1 we discussed and explained the working of CHIP in detail. We explained the basic idea of CHIP, the information input sources, types of tasks, dispatching tasks, and the optimizer. In this section we discuss the dispatcher, who is responsible for the schedule output of CHIP. Currently[2] 26 dispatchers are working within the HCC. There are also back-up dispatchers. A back-up dispatcher is someone who is normally working in the operation as an operator, but if there are not enough dispatchers available at a specific time he or she can also play the role of a dispatcher. We do not count the dispatchers from the cleaning companies Klüh and Asito, simply, because they are not part of KLM. Another important note is that the catering KCS is not using CHIP, but a different system.

Table 2 displays all 26 dispatchers and their qualifications. In total there are 18 dispatchers certified for aqua and refueling and 20 for towing and push-back.

# Dispatchers	Aqua	Towing	Push-Back	Refuel
4	✓	✓	✓	✓
8	✓	✓	✓	
6	✓			✓
8		✓	✓	✓

Table 2: Number of dispatchers with qualifications

The dispatchers are working within shifts. The first shift is from 6:00 AM to 2:30 PM. The next shift is the day shift which starts at 7:00 AM and ends at 3:30 PM. The late shift starts at 2:00 PM and ends at 10:30 PM. The night shift is from 10:00 PM to 6:30 AM.

2.3.1 Position, duties, and responsibilities

The goal of the dispatcher is that all aircraft service processes are finished on time. The dispatcher is responsible for the effective dispatching of jobs to the available resources that

[2] Numbers from May 2015

day. When dispatching the jobs the dispatcher should in some cases communicate with the other aircraft services. This communication is needed when a task of a different service conflicts with another task of another service. The towing department has the highest priority in CHIP compared to the other services. So other service tasks are planned around the towing task. Figure 11 displays a task conflict. The red sign at the end of the task indicates this conflict. The conflicting task of the other aircraft service is not visible for this dispatcher, only the start time of the task is given. There is a conflict between a towing task and pre-fuel task. It is namely not possible to tow an aircraft when the fuel truck is fueling. This conflict occurs when the pre-fuel task has a longer duration than expected or the pre-fuel is planned manually for a certain reason. In those cases it is necessary that a dispatcher communicates with the dispatcher of another service.

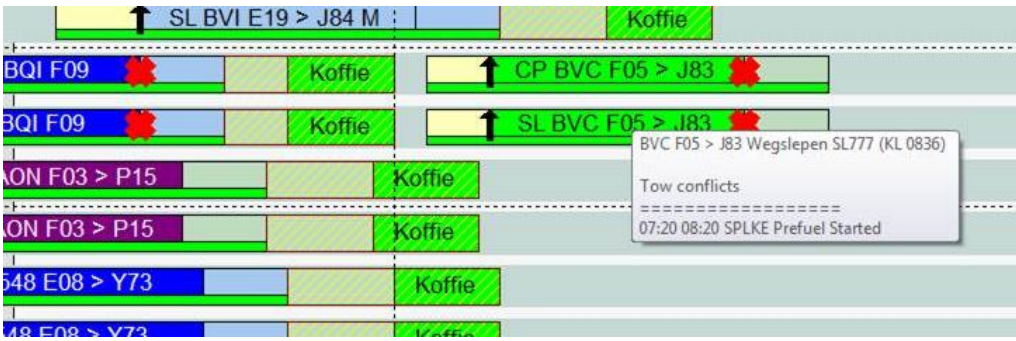


Figure 11: Task conflict with other service (Date: 25/06/2015)

The dispatcher should also act according to the priorities set by the duty hub manager (DHM) and the duty area manager (DAM). The DHM is the highest responsible person for the operational control of the handling processes of KLM and third parties at Hub Schiphol Airport. The goal of this function is to realize the flight schedule and to minimize the impact of disturbances. The DAM supports the DHM and is responsible for the daily operation and prioritization of services to ensure the timely departure of aircraft. The DMA coordinates and supervises the AS dispatchers. The shift leader of a specific aircraft service is responsible for the operators on the floor. This function is positioned between the AS dispatcher and the operators, but the shift leader is not responsible for dispatching of tasks. Summarizing the dispatcher is influenced by several people as displayed in Figure 12.

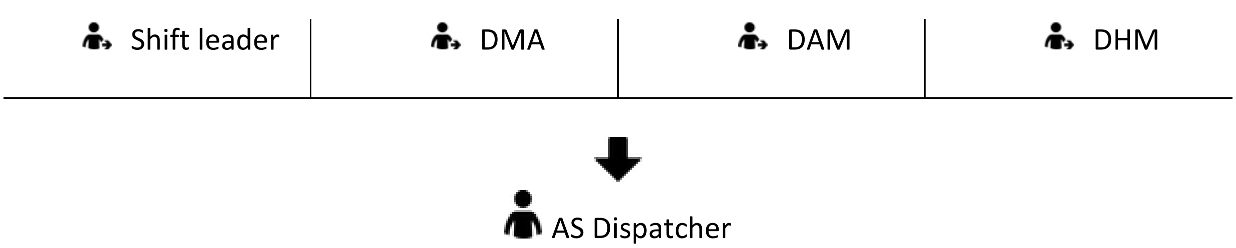


Figure 12: Dispatcher work environment

2.4 Conclusion

In this chapter we explained the current working methods of the AS dispatcher and responsibilities, duties, and position within KLM of an AS dispatcher. We started with a description of all aircraft services in more detail. Important is that only pushback/towing, aircraft refueling, and water/toilet services use CHIP as decision support system. CHIP is built by INFORM and uses a Gantt chart as main front-end that displays all resources and tasks. In total there are 26 dispatchers available that can work with CHIP. The information in CHIP is retrieved from three input sources, the flight information system FIRDA, the base data, and roster control. FIRDA provides real time flight information, the base data contains data that defines the operative environment, and the roster control holds information related to the operators. When dispatching tasks it is obligatory that a task matches the qualifications of the resource. When dispatching tasks the tasks must match with the qualifications of the resource, in order to ensure that a certain task is operated by an operator that has the right qualifications and certifications. Another important aspect that we considered in this chapter is the time window in which a task can be planned. This time window is bounded by the AIBT and the TOBT.

The dispatcher is responsible for the effective dispatching of jobs to the available resources that day. The dispatcher is influenced by the DHM, DAM, DMA, and the shift leader, where the shift leader is positioned between the operators on the floor and the dispatcher. For a dispatcher it is not always easy to act according to the interests of all four functions. Often there are entangled interests of all parties. In Chapter 3 we start with a literature study and discuss the scheduling problem, the scheduling process, robustness, scheduling information, and the performance measures.

3 Literature review

This chapter reviews the current literature on scheduling problems related to the scheduling problem of AS. In Section 3.1 we position the scheduling problem of AS within the scheduling literature and describe the scheduling problem according to the current knowledge. Section 3.2 discusses the scheduling process and related issues and techniques that are used to schedule against uncertainty. Section 3.3 explains the importance of the quality and timeliness of information in managing uncertainty. Section 3.4 discusses the human aspects of scheduling which are of great importance in an efficient scheduling process. At last, we discuss the construction of performance measures and stakeholder analysis in Section 3.5.

3.1 The scheduling problem

All aircraft services share common properties with respect to scheduling. All aircraft service tasks must be performed within a certain time window. Each aircraft service has limited resources available. Cost reduction is the shared optimization criterion, together with the on time departure. Each aircraft service has a process time that in most cases is stochastic. A good example of this process time stochasticity is the water service. The amount of water left in the airplane is not known beforehand when doing a refill. Therefore it is difficult to calculate the processing time upfront.

We consider the aircraft as customers of the aircraft service providers. The aircraft undergo a certain service treatment. The customers are at fixed places at the airport, the aircraft service is visiting the customers at their parking place. The former indicates that the aircraft service scheduling can be characterized as a vehicle routing problem (VRP). De Man (2014) considers the scheduling of refueling tasks also as a VRP problem, and mentions the fact that a VRP does not consider time-windows. Within the VRP the objective is to find a set of routes whose travel costs is minimized, taking into account that each customer is visited, the depot is used as start and end point, and that the demand of all customers does not exceed the capacity of the vehicles. This all complies with the scheduling problem of aircraft services. However, De Man (2014) already mentioned that time windows are not considered in the basic VRP problem.

The vehicle routing problem with time windows (VRPTW) is an important generalization of the basic VRP (Cordeau et al., 2007). Each service has to be performed in a given time interval $[a_i, b_i]$. For a service it is allowed to arrive before a_i , but arrivals after b_i are not allowed. If a service arrives before a_i it has to wait until a_i. In the aircraft service case it is allowed to arrive before the aircraft is at its parking place. Different from the VRPTW is that it is allowed to arrive later than b_i. This results in a late departure which is not desirable. However, the VRPTW can be modified with a penalty cost when arriving later than b_i, but this is not considered in the original VRPTW. The task time window start time a_i and task time window end time b_i are not precisely known upfront, the flight schedule is known beforehand, but gives no certainty on exact landing and departure times of aircraft beforehand. So the VRPTW has some similarities with the aircraft service case, but does not cover the exact problem.

The aircraft service scheduling problem has also properties of the Capacitated VRP. The capacity of each vehicle is known in advance, and it is not allowed to load the vehicle more than its capacity. Not all aircraft services share this property. For example the push-back service has no capacity constraints, simply because it does not carry any load. However, the refueling service has capacity constraints, it can supply a set of aircraft of fuel and after multiple hours it needs to refill the truck. A property of the capacitated VRP problem is that the demand of each customer is known beforehand (Daneshzand, 2011). This is certainly not the case in the aircraft service case where for most services the exact demand is not known upfront.

The time-window of arrival and departure of aircraft is considered as stochastic. This stochastic aspect is covered within the stochastic vehicle routing problems (SVRPs) and can deal with random components, stochastic customers, stochastic demands, and stochastic times (Cordeau et al., 2007). Stochastic customer means whether a customer is present or not. Stochastic demand means that the demand of customer *i* is a random variable. Stochastic service times means that the service time s_i and the travel time t_{ij} are random variables. De Man (2014) argues that the problem of refuel scheduling can be characterized as the stochastic demand VRP (VRPSD). The problem could also be treated as the stochastic customer VRP. A flight schedule does not give full certainty whether an aircraft arrives or not. An important note is that the opposite is also true, an aircraft that was not expected shows up. This situation can be translated in the show up probability p_i which is used within the stochastic customer VRP. We consider the travel times from and to the aircraft as deterministic, the travel times are estimated based on the distances and the vehicle speed. However, in the 'real' world travel times are not completely deterministic. Therefore, the vehicle routing problem with stochastic travel times (VRPSTT) fits the aircraft service scheduling problem. Nevertheless, we consider the travel times as deterministic, because the travel times are estimated by CHIP based on the vehicle speed and distances on the platform Therefore we do not use the VRPSTT to describe the aircraft service scheduling problem.

Another problem that has similarities with the scheduling of aircraft services are the dial-a-ride problems (DARP). In a dial-a-ride problem the objective is to fulfill as many requests as possible against minimum vehicle route costs (Cordeau & Laporte, 2007). There are two types of dial-a-ride problems: static or dynamic problems. In a static DARP all request are known beforehand. In a dynamic DARP requests are not known beforehand and become available throughout the day. Feuerstein and Stougie (2001) study the dial-a-ride problem in an on-line setting where the calls for rides come in time while the driver is travelling. However, they consider the single-server problem. If we had only one truck and one type of aircraft service this would be the case in the aircraft service scheduling, but this does not hold. Literature about the on-line multi-server dial-a-ride problems is limited. Bonifaci and Stougie (2009) study the online multi-server routing problems in which they propose several algorithms and proof lower bounds.

Pillac et al. (2013) argue that there are two important dimensions when considering vehicle routing problems within the real-world. The first dimension is evolution. Evolution is used in the sense that information gradually becomes available during the execution of routes. The second dimension is quality. Quality reflects the uncertainty that lies within the data. Another important distinction that Pillac et al. (2013) address and also De Man (2014) mentions is the difference between static and dynamic VRP. In a static VRP, all information for the construction of routes is available beforehand. In a dynamic VRP, information becomes gradually available and routes are created online. The scheduling problem of aircraft services is a dynamic or online scheduling problem. According to Pillac et al. (2013) the online arrival of customers is the most common source of uncertainty; this precisely matches the aircraft service case.

In this section we positioned the scheduling problem of aircraft service within the existing literature. However, the scheduling algorithm is an important part of scheduling process of aircraft services, but not the only part. It is part of the scheduling process. Kuhn and Loth (2009) did research towards algorithms for scheduling airport service vehicles. The objective was to minimize the fuel costs and air carrier delays for the service provider. Several algorithms were proposed and tested by using simulation data of Hamburg and Dallas-Fort Worth Airports. Figure 13 gives the framework that is developed by Kuhn and Loth (2009). This framework has many similarities with

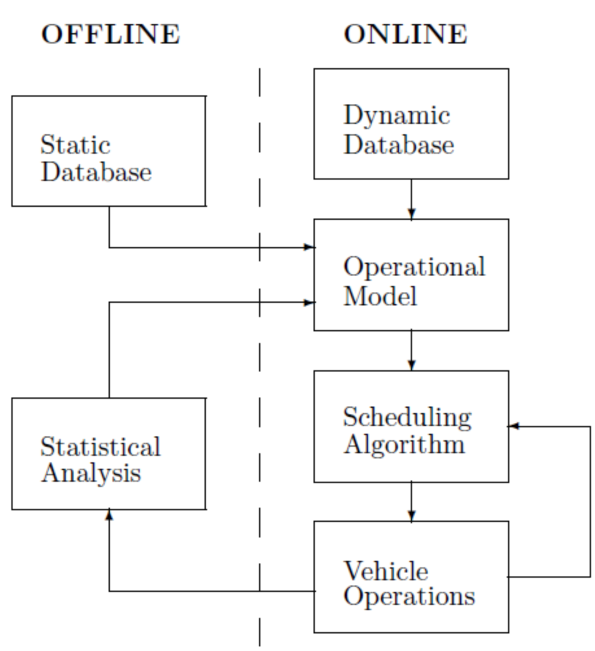


Figure 13: Framework Kuhn and Loth (2009)

the aircraft service case in this research. They end their paper with the remark that further research is needed to study the impact of uncertainty in airport arrival and departure onto the scheduling of service vehicles. Another important note to conclude this section is the fact that the scheduling of services includes three optimization problems according to Ip et al. (2013). The first one is the optimal assignment of jobs to operating crew. The second includes the optimization for obliging the flight schedule. The third encompasses the optimization of travelling time. These optimization problems cannot be seen separately. Next, we continue with the scheduling process.

3.2 Scheduling process

In Section 3.1 we outlined the difficulty of capturing the stochastic influence and uncertainty of given parameters within the scheduling problem of aircraft services. Most of the VRP models are rather static than dynamic. Uncertainty definitely increases the complexity of the assignment of jobs to aircraft service resources. Infeasibilities and process disturbances are often caused by uncertainty and are therefore considered as very important in production scheduling (Li & Ierapetritou, 2008). Pistikopoulos (1995) divides uncertainty in four categories:

- Model-inherent uncertainty
- Process-inherent uncertainty
- External-uncertainty
- Discrete uncertainty

Model-inherent uncertainty could be kinetic constants, physical properties, and so on. Process-inherent uncertainty could be uncertain processing times and equipment availability. External-uncertainty is uncertainty that is caused by factors outside the model such as prices and product demands. Finally discrete uncertainties are random discrete events such as personnel absence or broken equipment. Within the coordination and control of aircraft services we consider all types of uncertainty as proposed by Pistikopoulos (1995).

Verderame et al. (2010) describes how uncertainty factors can be characterized and how this uncertainty can be expressed in numbers. The preferred option to describe the uncertainty factor is to construct an estimation of the parameter's distribution (Verderame et al. (2010)). This is only possible if there is enough data available about this factor. However, in many practical cases there is not enough information to construct an accurate estimation of the distribution. Li and Ierapetritou (2008) suggests in those cases to use the bounded form. The uncertainty is then described with a calculated mathematical interval. This interval describes all possible values of the uncertain parameters. Another option is the fuzzy description, where the parameter is described with a value between 0 and 1. A high value implies a high possibility and a low value a poor possibility (Li & Ierapetritou, 2008).

To place uncertainty in the broader concept of the scheduling process we use the performance matrix of De Snoo et al. (2011). Figure 14 shows the performance matrix. In this matrix uncertainty is placed on the x-axis. The product performance on the left y-axis represents the mathematical side of the scheduling process. The process performance on the right y-axis encompasses the softer side of the scheduling process. This matrix shows that when uncertainty is low the mathematical side of scheduling is more important and when uncertainty is high the 'soft' side is more important. In that case the communication of dispatchers is more important.

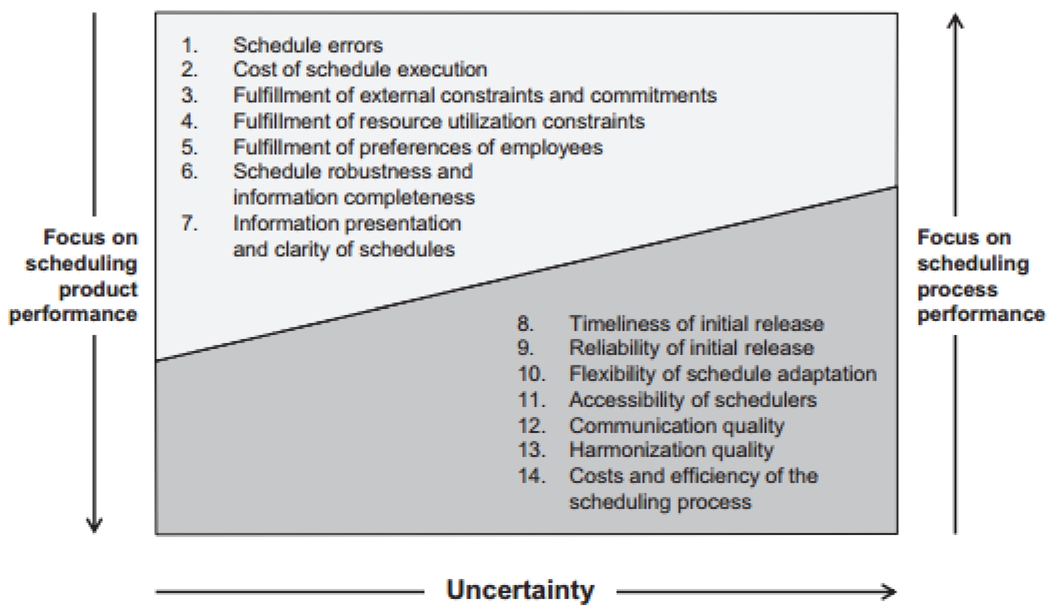


Figure 14: Scheduling performance matrix De Snoo et al. (2011).

De Snoo et al. (2011) performed three studies towards scheduling performance. The authors conclude that scheduling is not only a 'production process' of constructing a schedule. It is also a 'service process'. The authors argue that within this service process "information is collected and delivered, interests and trade-offs are discussed, and constraints or commitments are negotiated".

Within the literature several key elements are described to identify the steps in scheduling. Sabuncuoglu and Kizilisik (2003) consider the schedule generation and schedule control, where the schedule generation is considered as planning module and the schedule control as reactive mechanism. Li and Ierapetritou (2008) make a similar distinction between generation and control. The most important part in the aircraft service scheduling is to cope with uncertainty that influences the schedule on the day of execution. A way of coping with uncertainty within a schedule is to use the robustness principle. We discuss this in Section 3.2.1.

3.2.1 Robustness during scheduling

De Man (2014) describes the importance of creating a robust schedule. He describes robustness as a buffer against variability. We use and interpret the explanation of Herroelen and Leus (2004): if we can cope and protect the schedule of aircraft services against uncertainty we are able to create a robust schedule that is protected against anticipated schedule disruptions.

Li and Ierapetritou (2008) make a distinction between reactive scheduling and preventive scheduling. In handling uncertainty, we start with this distinction. The starting point in reactive scheduling is the baseline schedule. This schedule is constructed prior to the operation and modified in time to cope with or handle uncertainties. Preventive scheduling tries to deal with uncertainties prior to schedule execution, but it also uses a baseline schedule. We explain the difference between preventive and reactive with an example. Suppose we have an aircraft that is parked at the gate. The next task is to supply the aircraft with potable water, but suddenly this task is disturbed, because the water truck is broken. When scheduling reactively, this event is handled by changing the assignment of the water truck. This event could not have been captured into the schedule (preventive) prior to schedule execution.

Within the literature different approaches are described for preventive scheduling: Stochastic based approaches, robust optimization, fuzzy programming methods, sensitivity analysis, and parametric programming methods (Li & Ierapetritou, 2008). In a stochastic based approach, the uncertain variables are treated as random variables. Robust optimization tries to formulate the problem in such a way that the solution is robust with respect to the uncertainty in the data. A measure that is often used to measure the performance of a deterministic schedule is the standard deviation. The problem instance is solved for different settings of the random variables so that different results are realized (scenarios). The standard deviation is then given by Equation 1.

$$SD = \sqrt{\sum_{k} \frac{(H_k - H_{avg})^2}{p_{tot} - 1}}$$

Equation 1: Standard deviation of make span (Li & Ierapetritou, 2008)

Here H_{avg} represents the average makespan over all scenarios k, H_k the measurement value, and p_{tot} shows the total number of scenarios.

If the information to describe a probability distribution is not available another option is to use fuzzy programming. Sensitivity analysis can be used to describe how the output of a model depends upon the random variable input. It is then possible to evaluate for which values of the variables the output remains the same and which parameters are important for the model output.

De Man (2014) discusses the use of time slack to buffer against process time variability. The total planned time is described as the average process time plus a factor of the variance. However, slack considers only process time variability whereas in the scheduling of aircraft services more different uncertainties arise. The use of slack can be a way to absorb the uncertainty in processing times. A key issue in the use of slack is that if more slack is added to a task's processing time the robustness of the schedule increases, but the quality of the schedule decreases (Davenport et al., 2001). In Section 3.3 we continue with the explanation of the value of information.

3.3 Information

Information can be described by the quality of information or the timeliness of information. The higher the quality of information, the more precise the information of a certain parameter is. When information about a parameter is provided earlier it is less difficult to anticipate on this parameter or event. De Man (2014) mentions that most information that is known by AS is based on forecasts. The flight schedule can be considered as a reliable forecast. However, not all tasks are related to the flight schedule, those tasks are more difficult to plan according to De Man (2014). Most of the tasks that are scheduled within CHIP are flight related. The biggest non-flight related tasks are the lunch breaks of the operators.

Jaillet and Wagner (2006) introduces the notion of disclosure dates and release dates. If for example a flight operator or system says 'I will arrive in about 20 minutes', it is considered as a disclosure date. However, if the flight operator says 'I would like to be serviced now' it is considered as release date. Jaillet and Wagner (2006) argues that disclosure and release information increases the power of the online scheduler or player in handling uncertainty. The authors introduce a new sort of TSP namely the online TSP with disclosure dates. By varying the disclosure dates they can vary the "online-ness" of the problem. They use the competitive ratio as measure of the quality of a tested algorithm. This ratio is the ratio between the outcomes of the online TSP with disclosure dates against the optimal value of the TSP with release dates. The authors show that the existence and use of disclosure dates leads to improved competitiveness in comparison to the TSP with release dates. This finding contributes to the fact that the timely release of information is of high importance in scheduling.

Cowling and Johansson (2002) argue that if there are well-defined procedures for handling real time information, the nervousness of the system can be decreased and schedule improvement can be realized. For handling real time information they propose a four stage model given in Figure 15: detection, classification, identification, and diagnosis. The first step is detection of the information. For example, there is an arriving aircraft that needs service immediately, but was not listed on the flight schedule prior to execution. This information is detected by the system CHIP and the dispatcher that is responsible for the aircraft service. Next, this event must be classified by the system or dispatcher. This depends on the type of information whether it can be recognized by CHIP or the dispatcher. Often there is a need for more information about the event, the identification step. This step is proposed by Cowling and Johansson (2002) for prevention and improved prediction when a certain similar event occurs in the future. The last step a decision or action is taken to respond to the event, this is called the diagnosis step.

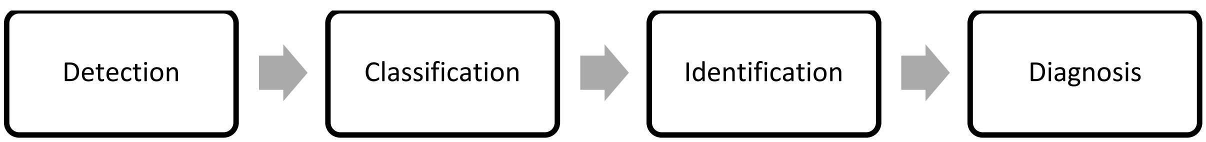


Figure 15: Four stage model Cowling and Johansson (2002)

The decision on a certain event is taken either by the system or by the dispatcher. An important aspect is that the dispatcher is not a computer that handles according to given rules. The dispatcher is part of the scheduling process and the organization. The decisions that the dispatcher makes related to the schedule are not only influenced by the scheduling problem, but also due to external effects. It is therefore important to assess the human and organizational aspects in Section 3.4.

3.4 Human and organizational aspects

CHIP is a decision support system that supports the dispatcher in the scheduling of aircraft service tasks. At the end the dispatcher is responsible for the schedule that is created. This schedule outcome is not only influenced by the technical scheduling process, but also by the dispatcher's informal authority and the activities between the different organizational groups where the dispatcher is working (Berglund & Karltun, 2007).

The main research area is on the mathematical side of the scheduling processes. The human and organizational aspects in the scheduling chain are often not considered. However, in this research the dispatcher plays an important role and the human and organizational aspects cannot be totally excluded. MacCarthy et al. (2001) argues that also in most manufacturing organizations scheduling still require human support to ensure the establishment of a schedule. The literature describes that scheduling is influenced by roles, trust, respect, and interpersonal interaction (McKay et al., 1992).

From Figure 14 we argued that when facing high uncertainty the need for high quality communication is larger. The dispatcher must not only be capable of creating a schedule, but needs also skills in communication and negotiation. De Snoo et al. (2011) explicitly describes when appointing schedulers the communication skills of the scheduler need to be assessed. This invigorates the finding that scheduling is more than only the mathematical construction of a feasible schedule, the human aspects cannot be omitted.

Berglund and Karltun (2007) consider an HTO-concept that can be used for the analysis and understanding of highly complex work activities. The H stands for human. It addresses the contribution of the individual to the process. In this research the main 'human' is the dispatcher which is responsible for the scheduling. The T in 'HTO' denotes the technical system. In this research the system CHIP is considered as the technical aspect. The O stands for the organizational aspects. H means the organization which is the complete set of humans. We believe that we need to consider all three aspects in this thesis to improve the online scheduling of aircraft services.

3.5 Performance measurement

In this section we review the literature about performance measurement. We review this literature, because in paragraph 4.3.3 we start with the construction of performance measures. Section 3.5.1 gives a general introduction about performance measurement. Section 3.5.2 describes the construction of performance measures in detail.

3.5.1 Performance measurement in general

Performance measurement is a broad topic. Therefore we start with the description of three terms: performance measurement, a performance measure, and a performance measurement system (Neely et al., 2005). Performance measurement is the process of measuring and quantifying activities. A performance measure is a specific metric or measure that quantifies the activity. Another definition comes from Lohman et al. (2004): performance measures "provide management with a tool to compare actual results with a pre-set target and to measure the extent of any deviation". The performance measurement system (PMS) can be seen as a tool to describe and quantify the efficiency and effectiveness of actions (Neely et al., 2005). In other words, the PMS is a tool to indicate how well a company is performing according to their actions, strategies, and processes. The PMS is thus more than only a set of performance indicators.

Performance measurement plays a central role in this research. Currently there is no performance measurement system with performance indicators that describe the quality of a schedule that is created by CHIP. However, there are performance indicators that indicate the overall performance of aircraft services, for example the on time departure performance according to the customer norm times, but there are no indicators that specifically describe the quality of a plan. In this research we are mainly interested in performance measurement rather than the construction of a performance measurement system.

3.5.2 Performance measures

Performance measures are used to quantify an activity. De Snoo et al. (2011) developed a scheduling performance criteria framework, which is displayed in Figure 16. In this framework, scheduling performance is divided into four different performance areas.

Scheduling performance criteria

Criteria focused on the scheduling product	Criteria focused on the scheduling process	Indirect scheduling performance criteria	Influencing factors
1. Schedule errors	1. Timeliness of initial release	1. Realized performance	1. Organizational planning structure
2. Cost of execution of the schedule	2. Reliability of initial release	2. Complaints and feedback from schedule users	2. Scheduler knowledge/skills
3. Fulfillment of constraints and commitments made to external parties	3. Flexibility of schedule adaptation		3. Information technology
4. Fulfillment of resource utilization constraints	4. Accessibility of schedulers		4. Complexity & uncertainty
5. Fulfillment of preferences and wishes of employees using the schedules	5. Communication quality		
6. Schedule robustness/information completeness	6. Harmonization quality		
7. Information presentation and clarity	7. Cost and efficiency of the scheduling process		

Figure 16: Scheduling performance criteria framework based on De Snoo et al. (2011).

The first criteria group is focused on the scheduling product. It encompasses the output of the scheduling process, the schedule itself. The second group includes the performance indicators related to the scheduling process. Measures that can only be influenced indirectly by schedulers are grouped in the third group. An example that De Snoo et al. (2011) describes is the end-customer satisfaction indicator. The end-customer satisfaction is often influenced by for example waiting times which are often a result of the schedule. The last group comprises all indicators that influence scheduling performance. For instance, the scheduler's knowledge and skills, the availability of information technology, and the level of complexity and uncertainty in the scheduling environment.

Several studies (Neely et al. (1997); Lohman et al. (2004)) recommend several aspects when constructing performance measures:

- Performance measures should be well-defined, understandable, and simple.
- When presenting the performance indicators the target must also be displayed.
- Performance indicators have to be used in combination with each other so as to cover all relevant aspects of an activity.

However, it is not only the construction of a performance measure that must be taken into account, also the purpose of the measure, frequency of measurement, and the source of data have to be considered (Neely et al., 2005). A stakeholder analysis is a helpful tool to relate performance measures to the end customers. Mitchell et al. (1997) developed a useful stakeholder typology which we explain in the next paragraph.

3.5.3 Stakeholder analysis

The perceived quality of a schedule during execution depends heavily on the stakeholder. A schedule that contains a lot of coffee breaks will probably be considered as 'high' quality from the perspective of an operator. However, a schedule with a lot of coffee breaks is considered as 'low' quality from the perspective of a dispatcher, simply for the reason that there is less scheduling flexibility. The difference in opinions between stakeholders invigorates the need for a decent stakeholder analysis before we start to measure the performance.

According to the Oxford English Dictionary a stakeholder is "a person, company, etc., with a concern or (esp. financial) interest in ensuring the success of an organization, business, system etc." Mitchell et al. (1997) have constructed a very useful classification or typology to address different stakeholders. Their model is based on three different stakeholder typologies: power, legitimacy, and urgency. These classes overlap, resulting in different types of

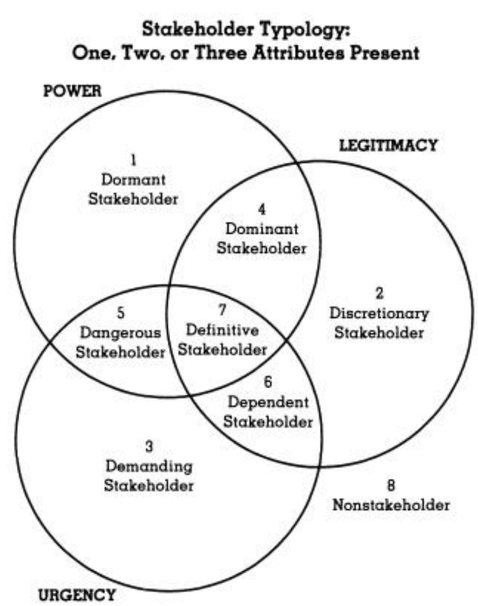


Figure 17: Stakeholder typology (Mitchell et al., 1997)

stakeholders. Figure 17 displays the stakeholder typology. The first circle represents power. Mitchell et al. (1997) use the definition of Weber (1947) power is “the probability that one actor within a social relationship would be in a position to carry out his own will despite resistance”. Legitimacy refers to socially accepted or the acceptance of authority. The combination of legitimacy and power creates authority (Mitchell et al., 1997).

Another helpful definition comes from the Oxford English Dictionary where legitimacy is described as: “conformity to rule or principle; lawfulness”. The last circle represents urgency. Urgency is added to the model, because power and legitimacy does not capture the dynamics of stakeholder-manager interactions (Mitchell et al., 1997). The authors conclude that adding urgency to the typology moves the model from rather static to dynamic. The Oxford English Dictionary describes urgency as: “the state, condition, or fact of being urgent; pressing importance; imperativeness”. Those three typologies are the basis of the stakeholder typology. Within the model there are eight different types of stakeholders: dormant stakeholder, discretionary stakeholder, demanding stakeholder, dominant stakeholder, dangerous stakeholder, dependent stakeholder, definitive stakeholder, and non-stakeholder.

3.6 Conclusion

In this chapter we positioned the AS scheduling problem within the current literature. We also discussed the scheduling process and uncertainty in scheduling. Furthermore, the chapter described the robustness of a schedule and the quality of information. The chapter ended with a discussion about the human and organizational aspects in scheduling and the construction of performance measures.

Scheduling problem

We argued that the scheduling problem of AS tasks can be described as a VRP. However, a shortcoming of the VRP is that it does not consider time windows. The VRP with time windows does, but in the basic VRPTW it is not allowed to arrive later than the given time window. This shortcoming can be modeled by a penalty cost which penalizes a late arrival of a certain service. The AS scheduling problem has also properties of the capacitated VRP. The key problem with the capacitated VRP is that the demand of each customer must be known beforehand. This is certainly not the case for aircraft, where service demand is often not predictable. The types of VRP problems that can deal with random components are called stochastic VRPs. We argued that the scheduling problem can be modeled as a stochastic customer VRP and stochastic demand VRP. The VRP with stochastic travel times VRPSTT can be used, but we consider the travel times of aircraft service vehicles as deterministic. The travel times are calculated based on the distances on the platform and vehicle speeds of the aircraft service vehicles. The dial-a-ride problem where service requests become available throughout the day does fit the scheduling problem of AS, where service requests are also not completely known beforehand. However, literature on multi-server dial-a-ride problems in an online fashion is limited.

Uncertainty increases the complexity of the assignment of AS tasks to AS resources. Four uncertainty categories can be identified: model-inherent, process-inherent, external, and discrete uncertainty. Those four categories are all present in the AS scheduling process. To describe the parameter uncertainty in numbers we can use the parameter's distribution, mathematical interval, or the fuzzy description. When we place the AS scheduling problem in the scheduling performance matrix of De Snoo et al. (2011) we conclude that the 'soft' aspects of the scheduling process are important. This is due to the high uncertainty that is involved in the AS scheduling problem. We also argued that if we can cope and protect the schedule of AS against uncertainty we are able to create a robust schedule that is protected against unanticipated schedule disruptions. A way to achieve this is to add slack as a buffer against process time variability. The main weakness with this slack is that it considers only process time variability whereas in the AS scheduling more uncertainties arise.

Process information and Human and Organizational aspects

Quality and timeliness are two important aspects of information. The higher the quality of information the more precise the information of a certain parameter is. Timeliness describes that when information is presented earlier it is less difficult to anticipate on this parameter or event. Cowling and Johansson (2002) developed a four stage model for handling real time information. This model consists out of four stages: detection, classification, identification, and diagnosis. The authors conclude that if there are well-defined procedures for handling real time information the nervousness of the system can be decreased and schedule improvement can be realized. Finally, we argued that the human aspects are of high importance in scheduling performance. This also fits the finding from the scheduling performance matrix, where we identified the importance of the 'soft' side of the scheduling process.

Performance measures and stakeholder analysis

When constructing performance measure we should define performance measures that are well-defined, understandable, and simple. When we display the performance measure we should also include the performance target value. Furthermore, the performance measures should be used in combination with each other to cover all relevant aspects of an activity. The stakeholder typology of Mitchell et al. (1997) is a helpful tool to identify and categorize stakeholders according to their, power, legitimacy, and urgency. It is a good starting point for the defintion of performance measures.

4 Performance and quality of online scheduling

In this chapter we provide insight into the scheduling performance and schedule quality during a day. Section 4.1 contains a stakeholder analysis that is used as basis for the performance measures. Section 4.2 discusses the importance of time in online scheduling and presents the measurement methods that we use to describe the scheduling performance. Section 4.3 presents the results from the performance analysis and outcomes. We conclude this chapter in Section 4.4 with an extensive discussion and conclusion about the results from the data analysis.

4.1 Stakeholder analysis

The dispatcher is responsible for scheduling aircraft service tasks. The output is a schedule that is continuously updated. We start with a list of all stakeholders that are involved or affected by a schedule. Next, we categorize them according to the stakeholder typology of Mitchell et al. (1997). Table 3 presents the results of the stakeholder analysis.

Stakeholder	Type
KLM Aircraft Service operators	Demanding
KLM Aircraft Services	Definitive
Pilots	Dependent
Crew	Dependent
Dispatchers	Definitive
KLM Aircraft Service shift leaders	Dominant
DAM (Duty Area Manager)	Definitive
DHM (Duty Hub Manager)	Definitive
DMA (Duty Manager Aircraft)	Definitive
Passengers	Dependent

Table 3: CHIP stakeholders and type

The pilots, crew, and passengers are all dependent stakeholders. These stakeholders have both urgency and legitimacy, but are dependent on the power of other stakeholders. This power lies at the dispatchers. The dispatchers have the power to change a schedule. The dispatchers are part of KLM Aircraft Services and therefore also marked as definitive stakeholder. The KLM AS operators are demanding stakeholders. They often have urgent claims, but having neither power nor legitimacy to act to those claims. A KLM AS operator is dependent on the power of the dispatcher to follow up on his claim. The DAM and DHM are both definitive stakeholders, because they also have power, legitimacy, and urgency to affect a schedule. Both are involved and responsible for the timely departure of aircraft. However, they are also dependent on the actions of the dispatcher, but if we reconsider Section 2.3.1 we see that the dispatcher should act according to the priorities set by the DAM and DHM and we therefore mark this stakeholder still as definitive. The same holds for the DMA who is responsible for the dispatchers. The KLM AS shift leaders are dominant stakeholders. They have power and legitimacy, but do not have the urgency to change the schedule.

The timely completion of aircraft service tasks is not the only important aspect of a timely departure. There are more processes that influence the departure time, for example the boarding process. These processes are not coordinated and scheduled in CHIP. The timely completion of aircraft service tasks within their corresponding time window does not guarantee that the aircraft can leave on its intended departure time. If the aircraft is delayed for a certain reason, the task time windows are shifted to this new departure time. However, the objective is still to schedule all tasks within their time windows.

Besides the timely completion of AS tasks within their corresponding time windows. there are more objectives that CHIP should pursue. These objectives are formulated by the project team that is responsible for the CHIP system within KLM Aircraft Services. We do not know which objectives are prioritized by the project team and CHIP.

The main objectives are:

- As much as possible tasks are planned within their corresponding time windows.
- Tasks with a high priority should be planned first.
- A task should be planned as early as possible within its time window.
- Driving times should be minimized.
- CHIP should support the dispatcher in all situations (for example: weather conditions, and understaffing)

4.2 Schedule performance over time

Time is an important aspect in online scheduling, a schedule evolves over time. Aircraft service tasks are changing in time, but also the numbers of aircraft service operators are changing. Within this section we assess the quality of a schedule and provide insights into the schedule behavior and changes over time. With this analysis we identify potential improvement areas into the scheduling process. We start in Section 4.2.1 with an explanation of the two different analyses. In Section 4.2.2 we describe the data gathering method.

4.2.1 Measurement methods

Our objective is to propose improvements for the online scheduling of aircraft service tasks. The optimizer within CHIP optimizes the assignment of tasks to resources against multiple optimization criteria. If we want to find improvement areas in the schedule optimization we need to know how the schedule evolves over time. In this manner we provide insights into the role of the optimizer. We can distinguish four types of changes in a schedule: tasks are created or deleted, and operators/resources are coming and going. We consider these four basic changes separately in our analysis. The first analysis provides insights into the number of resources during a day compared to the number of tasks. The second analysis gives a detailed view of the schedule performance over time. We present a new performance measurement method in which we analyze the performance for different time intervals. In our analysis we use the Aqua & Toilet service department for several reasons:

- The duration of aqua service tasks are relatively short compared with other aircraft service tasks. Due to this short duration there are more tasks to schedule during a day. In that way the optimizer has to schedule more tasks.
- The aqua service tasks are relatively straightforward. There are no multiple options per task such as pre-fuel or storm-fuel. This reduces the analysis complexity.
- CHIP is later implemented at the Aqua & Toilet service than at other aircraft services. For this reason, there is less research performed to the working of CHIP at the Aqua & Toilet service department. This makes the contribution of this research more valuable.

We explained the Aqua & Toilet service department in Section 1.1.2. However, we did not present all type of tasks that are obtained from the data of the Aqua & Toilet service department. To get a clear idea about the occurrences and type of tasks we gathered all CHIP tasks that have a start date between Monday 27 July 2015 and Sunday 2 August 2015. We have chosen this week and month, because it is the busiest time of the year (summer season). This summer season is clearly visible in the total number of tasks per month see Table 4.

Month (2015)	**Total number of tasks**
April	32,501
May	35,411
June	35,070
July	37,614
August	37,536

Table 4: Total number of CHIP tasks at the Aqua & Toilet service department 2015

From Table 4 we see that for the months July and August more tasks are handled compared to April, May, and June. In the week of Monday 27 July 2015, 8,509 tasks are handled by the Aqua & Toilet service department. Table 5 shows all aircraft service tasks and their corresponding occurrence in percentage for this specific week. The toilet service task and water fill tasks are most common, 33.94% and 27.90% respectively.

Type of task	%	Type of task	%
Airco	1.46%	Refill	6.11%
Airco WB[3]	2.94%	Special Task	0.80%
Airco APUloos[4]	0.01%	Standing Order	0.31%
Airco ARR	0.01%	ToiletCheck	0.01%
Airco removal	4.03%	ToiletService	33.94%
Airco removal wegsleep	0.04%	ToiletService Ex Hangar	1.10%
Airco WB opsleep	1.39%	ToiletService KBX	0.01%
Break	7.23%	WaterDrain	2.08%
Einde Dienst	3.88%	WaterFill	27.90%
Jetstarter	1.08%	WaterFill B737	0.07%
PCA[5]_APU_START	0.40%	WaterRefresh	3.43%
PCA_VERPLAATSEN	0.26%	WindowCleaning	0.08%
		WS + TS oke	1.43%

Table 5: Aqua service tasks and their occurrence in %, date: 27 July 2015 – 2 August 2015

[3] WB means a wide body aircraft. A wide body aircraft has two walking aisles.
[4] An APU is an auxiliary power unit, and is used to provide energy for the aircraft.
[5] PCA is a preconditioned air unit that is used to cool the aircraft on the ground.

The number of resources against the number of tasks over time

Workload is defined as the total number of tasks that are performed at a certain moment in time. A **workload graph** is a representation of workload over time, representing the workload across a 24-hour day for 1 minute time intervals (De Man (2014)).

Definition 1: Workload and workload graph definition

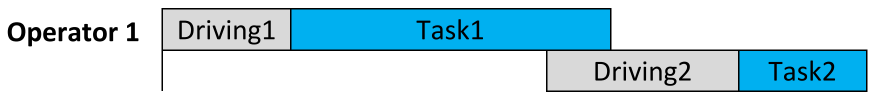
In the first analysis we construct multiple workload graphs that display the amount of tasks and number of active resources at a certain time. Such a graph can be constructed at the end of a day when all tasks and the exact number of resources are precisely known, because these events have already occurred. However, we are interested in the effects and schedule behavior over time, so we construct this workload graph for every 15 minutes during a day.

When constructing a workload graph we make several important assumptions:

1. We construct a workload graph based on data that contains past, present, and future events. The start time information of a resource that has been started in the past is known from the data, however, the real end time is not known. In that case we use the planned end time of a resource.
2. For a resource that starts in the future we use the planned start and end time.
3. For a resource that has already finished the shift, we use the real start and end time.
4. If a resource ends at 8:15, we count his presence until 8:14. This is caused by the fact that we construct the workload graph for every minute, so 8:14 means that the resource or task is active until 8:14 and 59 seconds.
5. We calculate the workload for every minute, because the data contains only information in minutes.
6. When a task is unplanned it does not have a driving time yet, because the previous task location is not known. In this case CHIP assigns a standard driving time of 60 seconds to this task.
7. We do not display coffee and lunch breaks of operators into the workload graphs, because we do not have the complete break information data. This results into a few gaps into the workload graphs at certain times. At those times it looks as if there is resource capacity left, but in fact there is not. However, the corresponding scheduling time windows (earliest start and latest end) of breaks are very narrow, this results into the fact that break times are almost fixed in advance. So we are able to easily detect break times within the workload graphs.

8. For the calculation of driving times we use the theoretical driving times that are used in CHIP. This driving time is based on the distance and vehicle speed. It can occur that driving times (light grey bars) exceed the number of resources. This is caused by the fact that we use the theoretical driving times and not the actual driving times.

Example:

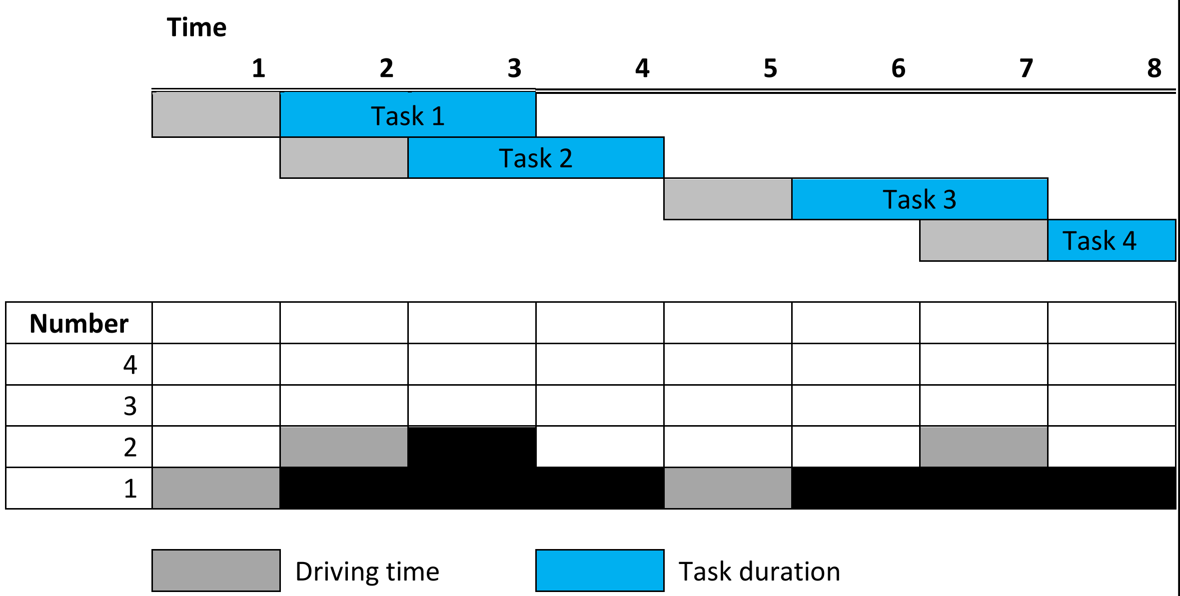


The calculated driving time of task 2 overlaps the duration of task 1, in this case the driving time exceeds the number of active operators. In fact two active operations (driving + task). However, in a practical setting this could not have occurred, because an operator cannot drive while he is performing a task.

Example 1: Driving time exceeds the number of active operators

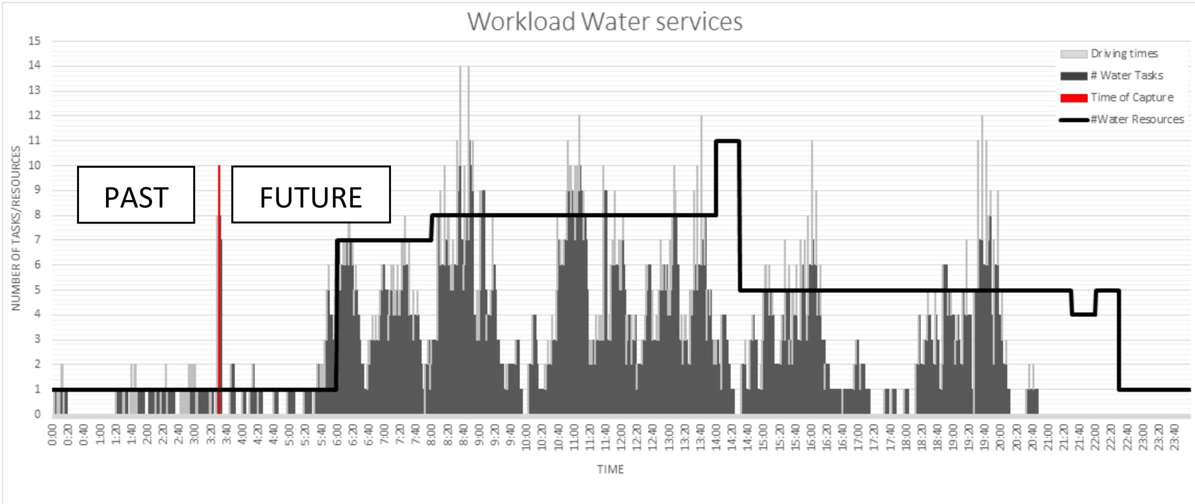
Example 2 graphically demonstrates the construction of workload graphs.

Example:



Example 2: Construction of workload graphs

In example Graph 1 we explain the workload graph in detail.



Graph 1: Workload example graph, date: 28 July 2015 3:30

Graph 1 displays the total number of tasks and resources on 28 July 2015 from 0:00 to 23:59. The number of resources is represented by the black line, the number of tasks by the dark grey bars, the driving times by the light grey bars, and the data capture time by the red vertical line (see Example 2). The data capture time is the time at which all data is retrieved from the server. This graph is made on 28 July at 3:30 (see red vertical line) so all tasks and resource changes before this time have occurred already. Time later than 3:30 is considered as future where tasks and resources are not fixed and can change. For example, resources do not show up, or tasks are deleted or created. The driving times are stacked with the tasks durations, because at those times the operators are not free. If we retrieve the data with task information and resource information every 15 minutes, from 0:00 to 23:59 we are able to construct a dynamic workload graph and show how the number of tasks and resources evolve over time. In this way we are able to provide insights into the working methods of the optimizer, because we visually can show how tasks are shifted and planned in time. From these graphs we are able to find improvements of the scheduling of water service tasks. These dynamic workload graphs can also be used as a helpful tool within the operations. With a dynamic workload graph the dispatcher is able to identify critical time windows in advance.

Time interval performance measurement

In the second analysis, we build a performance measurement sheet that measures the performance in the future. Instead of measuring the performance at the end of the day we measure the performance during a day. With this measurement technique we are able to provide insights into the scheduling performance in the future and identify possible critical time windows in advance. We construct performance measures during the day for the intervals in Table 6, where 0 represents the current time, or time of data capture.

0-15 min.	15-30 min.	30-60 min.	60-120 min.	120-180 min.	180-240 min.

Table 6: Performance measure intervals

The last interval is bounded by 240 minutes, because the optimizer within CHIP considers an optimization window of 4 hours. The first two intervals are both 15 minutes wide. The first two time intervals are smaller than the other time intervals, because we like to have more detailed performance measure data closer to the current time.

For the calculation of the performance measures we use the same method as used for the dynamic workload graphs. Namely, we calculate all performance measures every 15 minutes during a day. With this method we are able to evaluate the performance measures during a day and identify improvements. The performance measurement sheet will also be very helpful for the DMA and dispatchers, because they are able to look into the future. When using this performance measurement sheet they are able to identify possible critical situations in advance and take appropriate actions to prevent and solve these critical situations. The performance measurement sheet consists of two separate output sheets:

- A sheet that contains all performance measures[6] in numbers.
- A sheet that graphically displays the performance measures.

4.2.2 Data gathering method

In our analysis we capture data from the live environment of CHIP. This live environment is currently running and used in the operation. All information is stored in a database and we gather this data by using Oracle Hyperion which can be used as a query tool. We programmed this query tool in such a way that two comma separated values files are created for every 15 minutes which store the shift/resource information and task

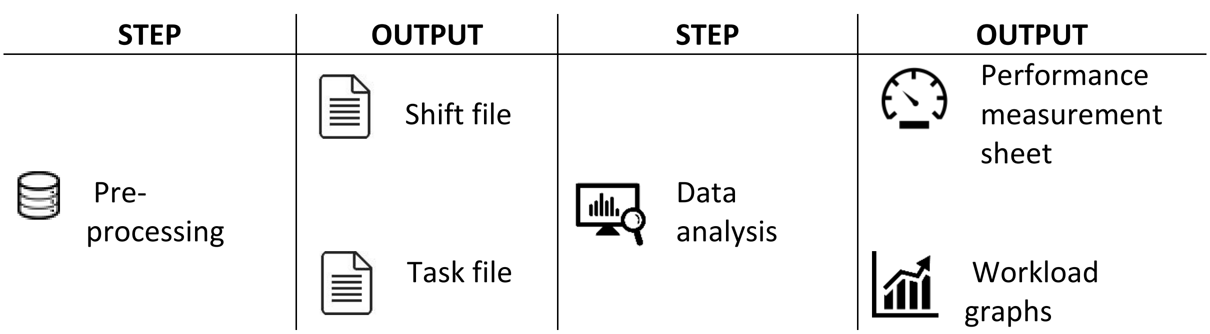


Figure 18: Data analysis steps

[6] We define all performance measures in Section 4.3.3.

information. Table 7 displays all data fields that we store in the shift file. The real start and end time are the clocking times of the resources. So if we retrieve the shift file at 8:00 am, resources that started before 8:00 do have a real start time, but resources that started later do not have a start time yet. The same holds for the real resource end time. Table 8 gives all data fields for the task file.

Shift file			
Resource name	Real start time	Real end time	Date
Department	Scheduled start	Scheduled end	

Table 7: Data fields shift file

Task file			
Department	Task type	Aircraft registration	Aircraft type
Start time	End time	Earliest start	Latest end
Task duration	Resource name	Task status	Setup duration
Task priority			

Table 8: Data fields task file

Each day we retrieve 96 task files and 96 shift files (every 15 minutes two files). Before we can use the task and shift file in our analysis we need to pre-process the files. This pre-processing is needed, because the task and shift file both contain data errors which occur during the data gathering from the live environment. The pre-processing step encompasses the following:

- Correct date format should be applied to columns that contain date fields.
- The raw files contain lines with tasks that are from another day than the day for which we are looking. These lines are removed.
- In some columns both dots and commas are used as decimal sign. For a correct calculation we should use either commas or dots. This is solved in the pre-processing step.
- The raw files contain illogical column names that do not represent the column content. We rename these columns in the pre-processing step.

Due to the large amount of data files we programmed a script in Python. A clean task and shift file are the output of this script. The two clean data files are used as input of two Excel documents, the workload analysis and performance measurement sheet. Appendix A gives the Python pre-processing code. Figure 18 graphically summarizes the data analysis steps and related output.

4.3 Analysis

In Section 4.3.1 we start with a general explanation about workload. Section 4.3.2 describes the first analysis and presents dynamic workload graphs. Section 4.3.3 presents the results from the performance measurement sheet.

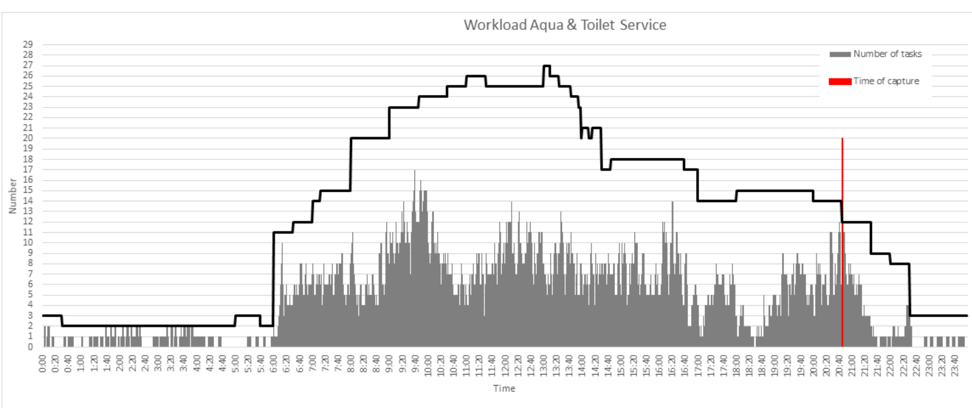
4.3.1 Workload analysis

Within the Aqua & Toilet service department there are three groups of tasks:

1. Jet starter, Airco, PCA cooling
2. Water drain, Water fill, Water refresh
3. Toilet service, Toilet Service EX hangar, Toilet Service KBX

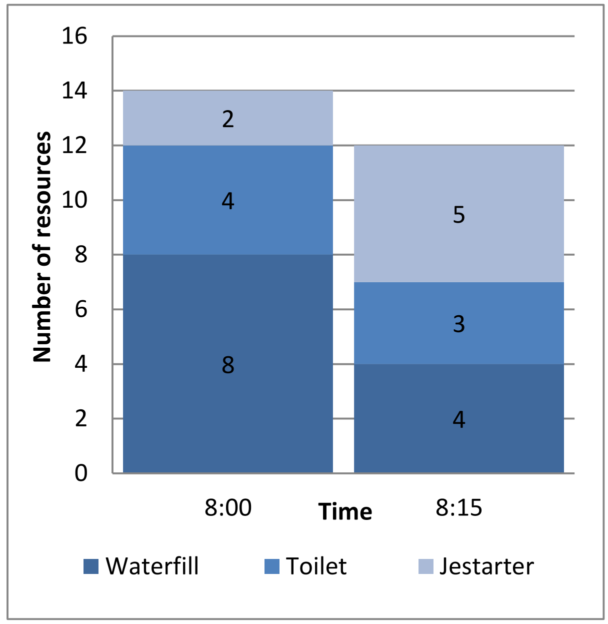
To obtain the first insights into the workload distribution over time, we construct a workload graph for one day with all task groups. At 8:00 we need 2 operators that are qualified to operate a jet starter, however at 8:15 we need 5 operators for the jet starter equipment. Graph 2 displays the workload graph for 28 July 2015, based on the data that was retrieved on 20:45, see the red vertical line. This workload distribution is based on all Aqua & Toilet service resources and task types displayed in Table 5. The first thing that we notice is the big gap between the number of resources and number of tasks from 6:00 to 17:00. From this graph one could say: the operation can probably be run by less people and probably less costs.

However, the discrepancy can be clarified by the resource qualification. An operator that can operate a toilet truck does not automatically have the right qualifications to also operate a water truck. And an operator that operates the Jet starter, Airco, and PCA is often not qualified to operate a water or toilet truck. Example Graph 3 displays the number of operators needed at two given times 8:00 and 8:15. This graph does not use the number of operators from Graph 2, but uses fictitious numbers to clarify the discrepancy.



Graph 2: Workload distribution Aqua & Toilet service, data capture at: 28 July 2015, 20:45

At 8:00 we need 2 operators that are qualified to operate a jet starter, however at 8:15 we need 5 operators for the jet starter equipment. When evaluating the workload distribution before determining the number of resources to hire, we probably selected 5 jet starter operators to handle the peak in tasks. This results in the fact that we have too many operators at a certain moment in time. The same effect holds for the water and toilet services. Important to note is that we do not investigate the processes to determine the number of resources needed. The big difference in number of tasks and resources for all tasks and resources within Aqua & Toilet services leads to the next step to analyze the groups separately. We do not analyze the Jet starter, Airco and PCA group in this step, because they mostly use only temporary workers to operate these tasks. This gives the shift leaders and resource planners far more freedom in sending operators home early or hiring extra operators on the day itself.

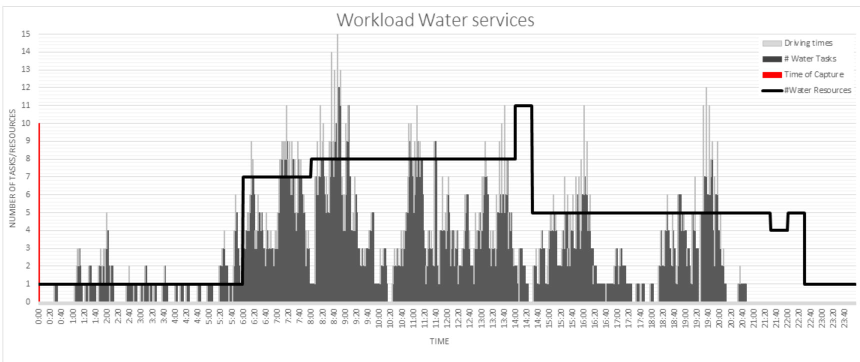


Graph 3: Example number of operators needed

The difference in number of tasks and resources for all tasks and resources with Aqua & Toilet services is not the only reason to study both services separately. The other reasons are: to decrease the analysis and computation complexity we only study the water service related tasks, and due to research time limitations we do not have enough time to study both services in full detail.

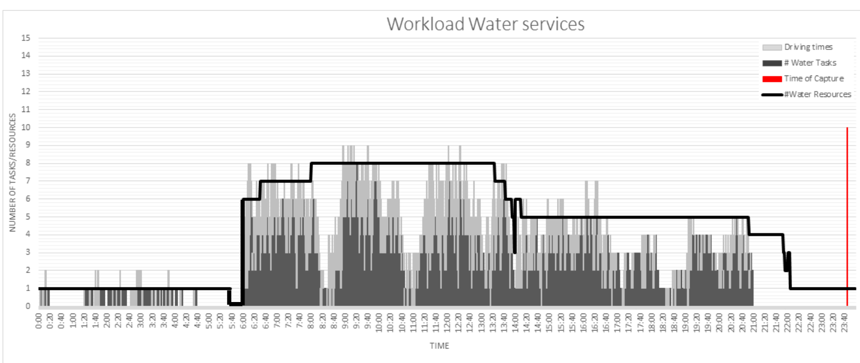
4.3.2 Dynamic workload graphs

We analyze the dynamic graphs by using animated gifs, in that way we can clearly identify the changes in tasks and resource numbers over the day visually. One major drawback of animated gifs is that we cannot show them within this report. However, we present the results of our analysis by presenting the separate graphs. Graph 4 shows the workload graph at the start of the day, where all tasks and resources are not fixed yet. In Graph 4 it is visible that at certain times the number of tasks and driving times exceeds the number of active resources. During the day the optimizer in CHIP and the dispatcher have to solve the problems within critical time windows. Graph 5 shows the workload graph at the end of the day, where all tasks are finished. In Graph 5 tasks never exceed the number of active resources, because this is not possible. This graph presents the events that already occurred during the day, and it is not possible to handle more tasks then active resources at a certain moment. In Section 4.2.1 we explained why the driving times can exceed the number of active resources. The next step is to analyze the scheduling process between Graph 4 and Graph 5 and present all findings. We do this by creating a list in Table 10 with all observations from the dynamic workload graphs. We use all observations from Table 9 as basis for the scheduling improvements in Chapter 5. We made and evaluated the observations with AS process analysts, resource planners, and operators from the Aqua & Toilet service department.



Start of the day

Graph 4: Workload distribution Aqua service tasks, data capture date: 28 July 2015, 0:00



End of the day

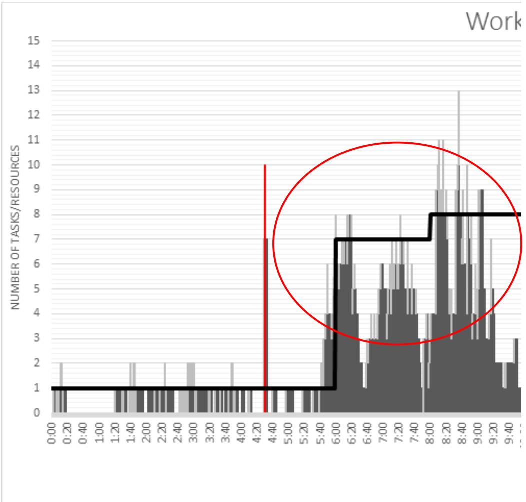
Graph 5: Workload distribution Aqua service tasks, data capture date: 28 July 2015, 23:45

All corresponding graphs are displayed on page 42 and 43. The red circles in the graphs mark the interesting points in time. All workload graphs from every hour are given in Appendix C.

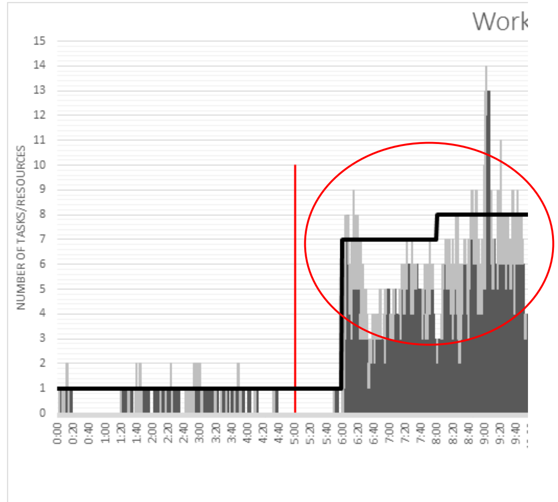
Observation	Graph
At the start of the day (0:00) almost all time windows that have a high task density and critical task/resource ratio are known.	4, 5
Driving times become longer closer to the current time (red line), this is caused by the fact that when a task is assigned to an operator CHIP calculates the theoretical driving time based on the previous task. This means that for the majority of cases the standard driving time that is added to tasks is too short.	6,7
During the night shift from 22:00 to 6:00 there is one operator active. At 5:45 this operator is suddenly clocked out while he should have finished his shift at 6:00. To facilitate his early departure the task with a start time of 5:30 is shifted to 6:00. This is probably done by the dispatcher, because at 6:00 there is a high task density and high workload, the optimizer would not have scheduled a tasks in this time window while having completely free time before 6:00.	8,9
We do not display breaks within the workload graph. However, the break times are clearly visible in the workload graph. At those times there are less scheduled tasks. Remarkable is that at the start of the shift it is known that operators should have a break, but it seems that the break is scheduled at the last moment by scheduling tasks to a later time in their time window.	10,11
The optimizer considers a four hour optimization time window. In this time window we clearly see that tasks are shifted, deleted, and created.	-
When a shift is almost finished a so called 'finishing shift'[7] task is created for each operator. We do not display this task in the workload graphs. Between 14:00 and 14:30 there is an increase in the number of operators due to a shift change. In this time window there are finishing shift tasks created, so we do not expect planned tasks in this time window. However, the optimizer thinks multiple times that the increase in operators can be used to schedule more tasks. At the last moment this increase in tasks is repaired by shifting tasks to a later time window. The result is that in this time window (14:00 – 14:30) only one task is completed.	12,13
After 15:00 there are multiple critical time windows that were known in advance, at those times the number of tasks exceeds the number of available resources. At 16:15 the current time bumps into this critical time window without repairing this critical task cluster in advance. This results in several jobs unplanned, because there were not enough operators. It seems that the dispatcher tried to schedule as much as possible tasks by hand without carefully looking at the corresponding task time windows.	14,15
At the shift change from 22:00 all operators are leaving earlier then their planned shift scheduled end time.	16,17

Table 9: Observation table

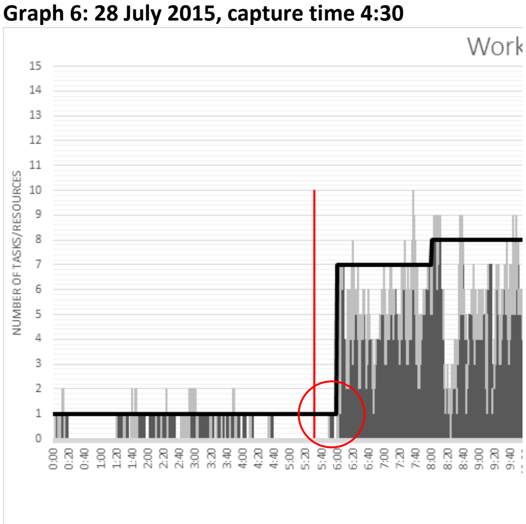
[7] A finishing shift task is created to notify the operator that his or her shift finishes and to reserve capacity, so that the optimizer does not plan tasks till the end of a shift.



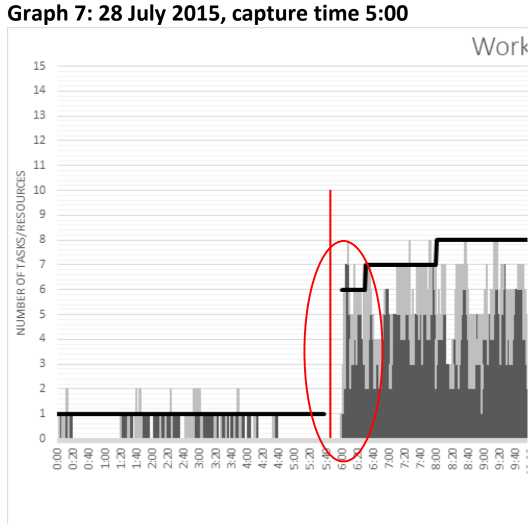
Graph 6: 28 July 2015, capture time 4:30



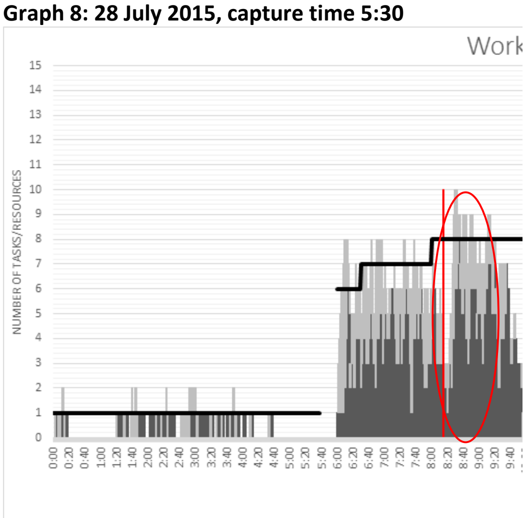
Graph 7: 28 July 2015, capture time 5:00



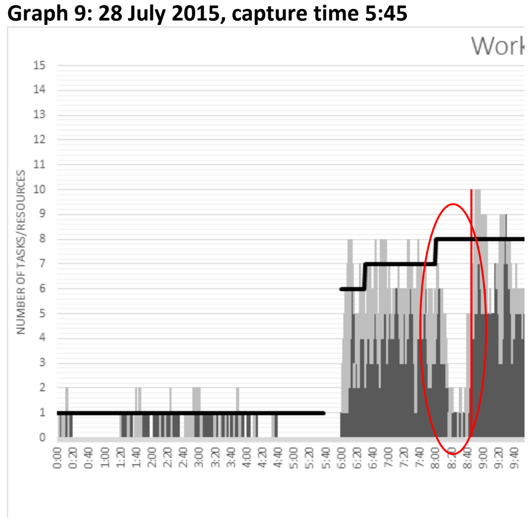
Graph 8: 28 July 2015, capture time 5:30



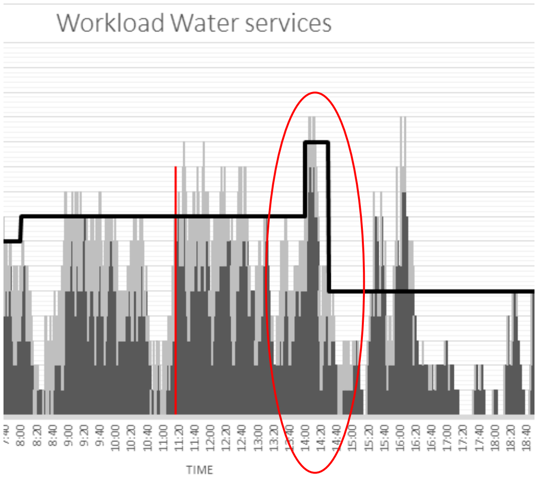
Graph 9: 28 July 2015, capture time 5:45



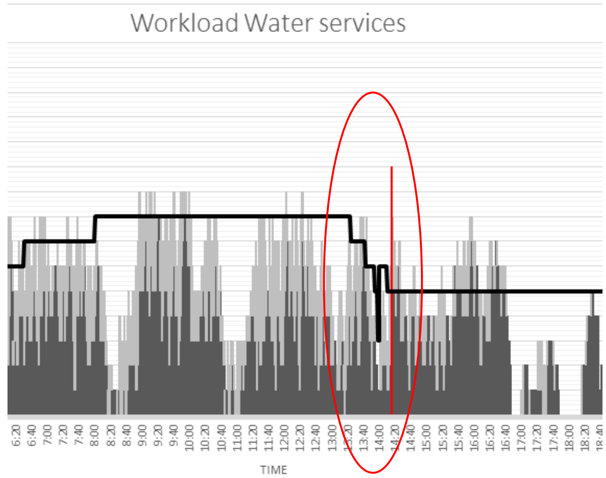
Graph 10: 28 July 2015, capture time 8:15



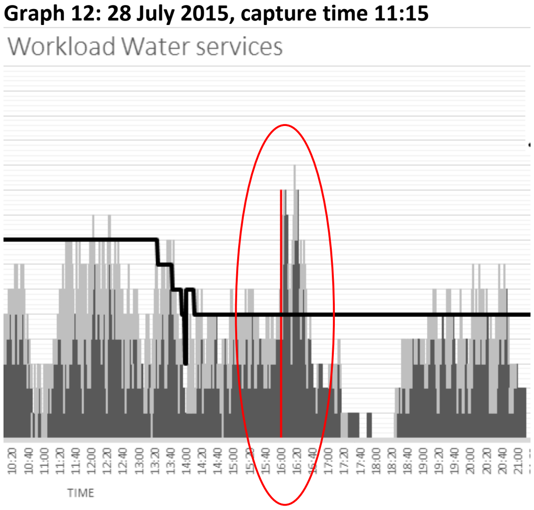
Graph 11: 28 July 2015, capture time 8:45



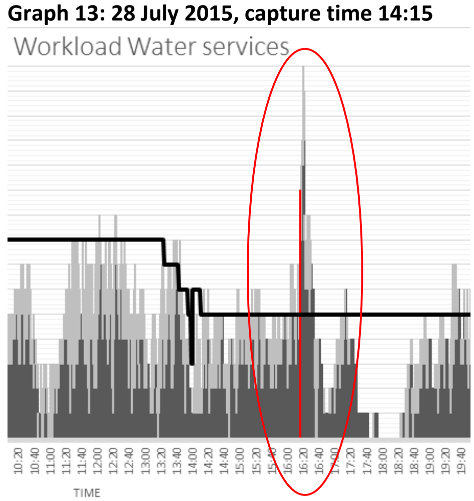
Graph 12: 28 July 2015, capture time 11:15



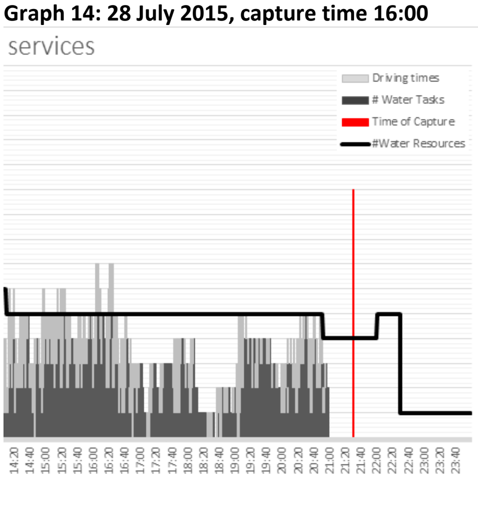
Graph 13: 28 July 2015, capture time 14:15



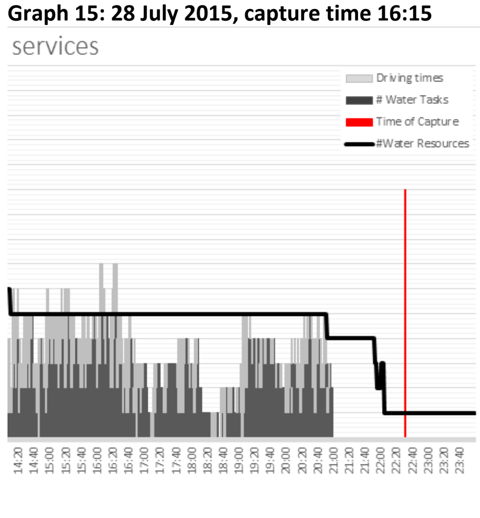
Graph 14: 28 July 2015, capture time 16:00



Graph 15: 28 July 2015, capture time 16:15



Graph 16: 28 July 2015, capture time 21:30



Graph 17: 28 July 2015, capture time 22:30

4.3.3 Performance measurement sheet analysis

We build a performance measurement sheet to measure and display performance measures related to the scheduling of aircraft service tasks. As explained in Section 4.2.1 we calculate and present the performance measures for future time windows. The future time windows can be changed in the performance measurement sheet, but the default is the same as we used for the dynamic workload graph analysis, see Table 10.

0-15 min.	15-30 min.	30-60 min.	60-120 min.	120-180 min.	180-240 min.

Table 10: Performance measurement sheet time windows

We use and construct the performance measurement sheet in two ways:

- Static performance measurement sheet. This performance measurement sheet can be used by the DMA or dispatcher to see critical time windows in advance. It shows the performance measures that we define in this section within their time windows, both graphically and numerically.
- Dynamic performance measurement sheet. This performance measurement is solely used within our analysis to evaluate the performance measures and to find possible areas for scheduling improvement. (See Section 4.2.1)

The performance measures that we construct are based on the first column of the scheduling performance criteria framework of De Snoo et al. (2011), see Figure 14 in Section 3.5.1. The first column is focused on the scheduling product, or the schedule itself. The performance measures are also based on the CHIP objectives listed in Section 4.1 and the stakeholder analysis. We start with the performance measure definitions and detailed explanations. After the definitions we link the performance measures to the stakeholders and CHIP objectives. We use the following notations:

<table>
<tr><th>Notation</th><th colspan="2">Description</th></tr>
<tr><td>i</td><td colspan="2">Tasks</td></tr>
<tr><td rowspan="7">j</td><td>j</td><td>time window (min.)</td></tr>
<tr><td>1</td><td>0-15</td></tr>
<tr><td>2</td><td>15-30</td></tr>
<tr><td>3</td><td>30-60</td></tr>
<tr><td>4</td><td>60-120</td></tr>
<tr><td>5</td><td>120-180</td></tr>
<tr><td>6</td><td>180-240</td></tr>
<tr><td>n_j</td><td colspan="2">All tasks that have a <u>start time</u> within time window j</td></tr>
</table>

Table 11: Notations used for the definition of performance indicators

Performance indicator 1 Planned ratio

$$planned\ ratio(j) = \frac{\sum_{i=1}^{n_j} planned(i,j)}{\sum_{i=1}^{n_j} taskcount(i,j)} * 100\%$$

where,

$$planned(i,j) = \begin{cases} 1, & if\ task\ i\ is\ planned^{8}\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

$$taskcount(i,j) = \begin{cases} 1, & if\ task\ i\ has\ a\ start\ time\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

Performance measure 1 provides insights into the capability of CHIP to plan all tasks with a given number of resources. A value closer to 100% indicates a better performance.

Calculation example: Planned and unplanned ratio

Task	Earliest start time	Task start time	Task end time	Latest end time	Status
1	28-7-2015 19:10	28-7-2015 19:16	28-7-2015 19:25	28-7-2015 20:30	Planned
2	28-7-2015 19:20	28-7-2015 19:20	28-7-2015 19:25	28-7-2015 20:15	Free
3	28-7-2015 19:22	28-7-2015 19:22	28-7-2015 19:28	28-7-2015 20:15	Free

Current time : 19:15

When a task has the status free (unplanned) the task start time is set equal to the earliest start time. The duration of time window 1 (*j=1*) is 15 minutes. So we look at all tasks n_j that have a start time between 19:15 and 19:30. The planned ratio for time window 1 is calculated as follows:

$$planned\ ratio(1) = \frac{\sum_{i=1}^{3} planned(i,j)}{\sum_{i=1}^{3} taskcount(i,j)} * 100\% = \frac{(1+0+0)}{(1+1+1)} * 100\% = 33.3\%$$

$$unplanned\ ratio(1) = 100 - planned\ ratio(1) = 100 - 33.3 = 66.7\%$$

Performance indicator 2 Unplanned ratio

$$unplanned\ ratio(j) = 100 - planned\ ratio(j)$$

Performance measure 2 also provides insight into the capability of CHIP to plan all tasks with a given number of resources. A value closer to 0% indicates a better performance.

[8] Tasks with a task status, planned, confirmed, arrived, or assigned are considered as planned.

Performance indicator 3 Driving time ratio

$$driving\ time\ ratio(j) = \frac{\sum_{i=1}^{n_j} task\ driving\ time(i,j)}{\sum_{i=1}^{n_j} task\ duration(i,j)}$$

where,

$$task\ driving\ time(i,j) = \begin{cases} driving\ time\ (min.)\ task\ i, & if\ task\ i\ is\ planned\ and\ has\ a\ start\ time\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

$$task\ duration(i,j) = \begin{cases} duration\ (min.)\ task\ i, & if\ task\ i\ is\ planned\ and\ has\ a\ start\ time\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

Performance measure 3 provides insights into the ratio between the total driving time and task duration. The objective is to minimize the driving times for resources. To make it comparable we standardize the total driving time in minutes with the task duration in minutes.

Calculation example: Driving time ratio

Task	Driving time (min.)	Task start time	Task end time	Duration (min.)	Status
1	0:03	28-7-2015 19:16	28-7-2015 19:25	0:09	Planned
2	0:01	28-7-2015 19:20	28-7-2015 19:25	0:05	Free
3	0:01	28-7-2015 19:22	28-7-2015 19:28	0:03	Free

Current time : 19:15

The driving time ratio of time window 1 (*j=1*) is calculated as follows:

$$driving\ time\ ratio(1) = \frac{\sum_{i=1}^{3} task\ driving\ time(i,j)}{\sum_{i=1}^{3} task\ duration(i,j)} = \frac{3+0+0}{9+0+0} = \frac{1}{3}$$

Performance indicator 4 Time window duration

$$time\ window\ duration(j) = \frac{\sum_{i=1}^{n_j} time1(i,j) - time2(i,j)}{\sum_{i=1}^{n_j} taskcount(i,j)}$$

where,

$$time1(i,j) = \begin{cases} latest\ end\ time\ task\ i, & if\ task\ i\ is\ planned\ and\ has\ a\ start\ time\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

$$time2(i,j) = \begin{cases} earliest\ start\ time\ task\ i, & if\ task\ i\ is\ planned\ and\ has\ a\ start\ time\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

$$taskcount(i,j) = \begin{cases} 1, & if\ task\ i\ is\ planned\ and\ has\ a\ start\ time\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

Section 2.2.1 discusses the earliest start time and latest end time of a task. The average time window duration gives the difference between the earliest start time of a task and the latest end time of a task. Longer time window duration gives more scheduling flexibility.

Remark 1:

This performance measure cannot be directly influenced by the dispatcher, because the earliest start time and latest end time of a task are based on the flight plan. However, our performance measurement sheet has also the purpose of being used by a dispatcher to see critical events in advance. Shorter time window duration gives less scheduling flexibility for CHIP and the dispatcher. With this performance measure the dispatcher is able to see this in advance, where cannot see this information directly from his Gantt chart.

Calculation example: Time window duration

Task	Earliest start time	Task start time	Task end time	Latest end time	Status
1	28-7-2015 19:10	28-7-2015 19:16	28-7-2015 19:25	28-7-2015 20:30	Planned
2	28-7-2015 19:20	28-7-2015 19:20	28-7-2015 19:25	28-7-2015 20:15	Free
3	28-7-2015 19:22	28-7-2015 19:22	28-7-2015 19:28	28-7-2015 20:15	Free

Current time : 19:15

The duration of time window 1 (*j=1*) is 15 minutes. So we look at all tasks n_j that have a start time between 19:15 and 19:30. The time window duration of time window 1 (*j=1*) is calculated as follows:

$$time\ window\ duration(1) = \frac{\sum_{i=1}^{3} time1(i,j) - time2(i,j)}{\sum_{i=1}^{3} taskcount(i,j)}$$
$$= \frac{(20:30 - 19:10) + (0 - 0) + (0 - 0)}{1 + 0 + 0} = 80\ min.$$

Performance indicator 5 Average priority

$$average\ priority(j) = \frac{\sum_{i=1}^{n_j} priority(i,j)}{\sum_{i=1}^{n_j} taskcount(i,j)}$$

where,

$$priority(i,j) = \begin{cases} priority\ task\ i, & if\ task\ i\ is\ planned\ and\ has\ a\ start\ time\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

$$taskcount(i,j) = \begin{cases} 1, & if\ task\ i\ is\ planned\ and\ has\ a\ start\ time\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

CHIP uses the priority of a task to prioritize tasks with a high priority. Priorities are ranged from 1 – 400, where 400 indicate a high priority and 1 represents a low priority.

Remark 2:

This performance measure cannot be directly influenced by the dispatcher, because the priorities are fixed values. However, with this performance we can evaluate how priorities are handled by the optimizer for future time windows. It also provides the dispatcher extra information that he cannot see directly from his Gantt chart.

Calculation example: Average priority

Task	Earliest start time	Task start time	Task end time	Priority	Status
1	28-7-2015 19:10	28-7-2015 19:16	28-7-2015 19:25	290	Planned
2	28-7-2015 19:20	28-7-2015 19:20	28-7-2015 19:25	100	Free
3	28-7-2015 19:22	28-7-2015 19:22	28-7-2015 19:28	5	Free

Current time : 19:15

The duration of time window 1 (*j=1*) is 15 minutes. So we look at all tasks n_j that have a start time between 19:15 and 19:30. The average priority of time window 1 (*j=1*) is calculated as follows:

$$average\ priority(1) = \frac{\sum_{i=1}^{3} priority(i,j)}{\sum_{i=1}^{3} taskcount(i,j)} = \frac{290 + 0 + 0}{1 + 0 + 0} = 290$$

Performance indicator 6 Earliest start plan

$$earliest\ start\ plan(j) = \frac{\sum_{i=1}^{n_j}(start\ time(i,j) - time2(i,j))}{\sum_{i=1}^{n_j} taskcount(i,j)}$$

where,

$$start\ time(i,j) = \begin{cases} start\ time\ task\ i, & if\ task\ i\ is\ planned\ and\ has\ a\ start\ time\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

$$time2(i,j) = \begin{cases} earliest\ start\ time\ task\ i, & if\ task\ i\ is\ planned\ and\ has\ a\ start\ time\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

$$taskcount(i,j) = \begin{cases} 1, & if\ task\ i\ is\ planned\ and\ has\ a\ start\ time\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

The average earliest start plan measure calculates the difference between the start time of a task and the earliest start time of task. Finally, we take the average over all measurements. A low earliest start plan is preferable, because in that case the task is planned close to the earliest start time.

Calculation example: Earliest start plan

Task	Earliest start time	Task start time	Task end time	Latest end time	Status
1	28-7-2015 19:10	28-7-2015 19:16	28-7-2015 19:25	28-7-2015 20:30	Planned
2	28-7-2015 19:20	28-7-2015 19:20	28-7-2015 19:25	28-7-2015 20:15	Free
3	28-7-2015 19:22	28-7-2015 19:22	28-7-2015 19:28	28-7-2015 20:15	Free

Current time : 19:15

The earliest start plan for time window 1 (*j=1*) is calculated as follows:

$$earliest\ start\ plan(1) = \frac{\sum_{i=1}^{3}\left(start\ time(i,j) - time2(i,j)\right)}{\sum_{i=1}^{3} taskcount(i,j)} = \frac{(19{:}16 - 19{:}10) + 0 + 0}{1 + 0 + 0}$$
$$= 6\ min.$$

Performance indicator 7 Average number of resources available

The average number of active resources is calculated in two steps:

- First we calculate for every minute from 0:00 to 23:59 the number of active resources. A resource is active in a specific minute, if the minute is between the shift start time and shift end time of a resource.
- Next, we calculate the average number of active resources per time window *j*.

Remark 3:

This performance measure cannot be directly influenced by the dispatcher, because the number of resources is mostly fixed at the start of a day. However, this measure gives the dispatcher a helpful tool to precisely see how much resource capacity is scheduled in future time windows.

Calculation example: Average number of resources

Resource	Shift start time	Shift end time
1	28-7-2015 19:10	28-7-2015 20:00
2	28-7-2015 19:20	28-7-2015 20:00
3	28-7-2015 19:22	28-7-2015 20:00

Current time : 19:15

Time	# Resources	Time	# Resources	Time	# Resources
19:15	1	19:21	2	19:27	3
19:16	1	19:22	3	19:28	3
19:17	1	19:23	3	19:29	3
19:18	1	19:24	3	19:30	3
19:19	1	19:25	3		
19:20	2	19:26	3		

Time window 1 is from 19:15 to 19:30. The average number of resources is the average from 19:15 to 19:30 for every minute. **The average number of active resources is : 2.25**

Performance indicator 8 Total average task duration per resource

$$AvTaskDurResource(j) = \frac{\sum_{i=1}^{n_j} task\ duration(i,j)}{average\ number\ resources(j)}$$

where,

$$task\ duration(i,j) = \begin{cases} task\ duration\ task\ i, & if\ task\ i\ is\ planned\ and\ has\ a\ start\ time\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

$$task\ duration\ task\ i = Task\ i\ end\ time - task\ i\ start\ time$$

$$average\ number\ resources(j) \rightarrow see\ performance\ indicator\ 7$$

The total average task duration per resource gives the total average task duration per resource within a given time interval *j*. The optimizer within CHIP tries to equally divide and assign tasks to resources based on the task durations, so that every resource has on average the same workload. With this performance measure we are able to provide insights into the optimization of workload for resources.

Calculation example: Total average task duration per resource

Task	Earliest start time	Task start time	Task end time	Duration (min.)	Status
1	28-7-2015 19:10	28-7-2015 19:16	28-7-2015 19:25	0:09	Planned
2	28-7-2015 19:20	28-7-2015 19:20	28-7-2015 19:25	0:05	Free
3	28-7-2015 19:22	28-7-2015 19:22	28-7-2015 19:28	0:06	Free

Current time : 19:15

The total average task duration per resource for time window 1 (*j=1*) is calculated as follows:

$$AvTaskDurResource(1) = \frac{\sum_{i=1}^{3} task\ duration(i,j)}{average\ number\ resources(1)} = \frac{9+0+0}{2.25} = 4\ min.$$

Performance measure 9 Average task duration

$$Average\ task\ duration(j) = \frac{\sum_{i=1}^{n_j} task\ duration(i,j)}{\sum_{i=1}^{n_j} taskcount(i,j)}$$

where,

$$task\ duration(i,j) = \begin{cases} task\ duration\ task\ i, & if\ task\ i\ is\ planned\ and\ has\ a\ start\ time\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

$$task\ duration\ task\ i = Task\ i\ end\ time - task\ i\ start\ time$$

$$taskcount(i,j) = \begin{cases} 1, & if\ task\ i\ is\ planned\ and\ has\ a\ start\ time\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

Performance measure 9 gives the average task duration per time window *j.* If a time window *j* has a lower average task duration if compared to the other time windows, it has a higher risk of not completing tasks on time. For the reason that a time window with low average task duration tend to have more tasks than a time window with higher average task duration, due this there are extra driving times between jobs that increases the risk.

Remark 4:

This performance measure cannot be directly influenced by the dispatcher, because the task duration is a fixed value. However, this measure gives the dispatcher a helpful tool to assess the risk of future time windows.

Calculation example: Average task duration

Task	Earliest start time	Task start time	Task end time	Duration (min.)	Status
1	28-7-2015 19:10	28-7-2015 19:16	28-7-2015 19:25	0:09	Planned
2	28-7-2015 19:20	28-7-2015 19:20	28-7-2015 19:25	0:05	Free
3	28-7-2015 19:22	28-7-2015 19:22	28-7-2015 19:28	0:06	Free

Current time : 19:15

The total average task duration per resource for time window 1 (*j=1*) is calculated as follows:

$$Average\ task\ duration(j) = \frac{\sum_{i=1}^{3} task\ duration(i,j)}{\sum_{i=1}^{3} taskcount(i,j)} = \frac{9+0+0}{1+0+0} = 9\ min.$$

Performance measure 10 Number of tasks that have an end time in time window j

$$NumberTaskEndTimeWindow(j) = \sum_{i=1}^{n_j} taskend(i,j)$$

where,

$$taskend(i,j) = \begin{cases} 1, & if\ task\ i\ is\ planned\ and\ has\ an\ end\ time\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

Time windows that contain a lot of task endings have a higher risk of task delay than time windows that do not contain task endings.

Performance measure 11 Time loading

$$Time\ loading(j) = \frac{\sum_{i=1}^{n_j} task\ duration(i,j)}{\sum_{j=1}^{j} timewindowduration(j) * average\ number\ resources(j)}$$

where,

$$task\ duration(i,j) = \begin{cases} task\ duration\ task\ i, & if\ task\ i\ is\ planned\ and\ has\ a\ start\ time\ in\ time\ window\ j \\ 0, & otherwise \end{cases}$$

$$timewindowduration(j) = time\ window\ end\ time(j) - time\ window\ start\ time\ (j)$$

$$average\ number\ resources(j) \rightarrow see\ performance\ indicator\ 7$$

The time loading describes the ratio between the total task duration in time window *j* and the total available resource capacity in time window *j*. A high time loading results in a lower scheduling flexibility, the time workload should not exceed 1.

Calculation example: Time loading

Task	Earliest start time	Task start time	Task end time	Duration (min.)	Status
1	28-7-2015 19:10	28-7-2015 19:16	28-7-2015 19:25	0:09	Planned
2	28-7-2015 19:20	28-7-2015 19:20	28-7-2015 19:25	0:05	Free
3	28-7-2015 19:22	28-7-2015 19:22	28-7-2015 19:28	0:06	Free

Current time : 19:15

The time workload for time window 1 (*j=1*) is calculated as follows:

$$average\ number\ resources(1) = 2.25$$

$$timewindowduration(1) = 15 - 0 = 15\ min.$$

$$Time\ loading(1) = \frac{\sum_{i=1}^{n_j} task\ duration(i,j)}{\sum_{j=1}^{j} timewindowduration(j) * average\ number\ resources(j)} = \frac{9+0+0}{15 * 2.25} = 0.27$$

Important to mention is that time loading is different from the workload used in the workload graphs.

Performance measure relation with stakeholders and CHIP objectives

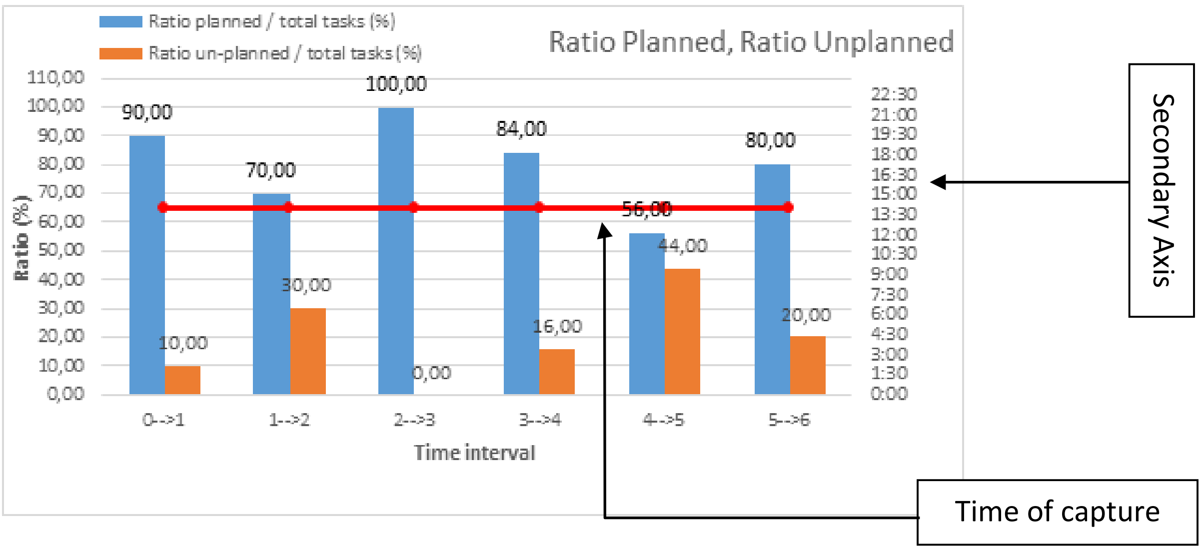
The performance measures that we explained and discussed are all based on the stakeholder analysis and CHIP objectives. Table 12 shows the relation between the performance measures, stakeholders, and CHIP objectives.

Stakeholder	Performance measure
Pilots, crew, and passengers	1,2
AS operators	3, 9, 11
KLM Aircraft Services	1, 2, 3, 5, 6 ,7
AS shift leaders	1, 2, 8
DAM, DHM	5
Dispatchers, DMA	4, 8, 10
CHIP objective	
As much as possible tasks are planned	1, 2
High priority should be planned first	5
Planned as early as possible in time window	4, 6, 7
Driving times should be minimized	3
CHIP supports the dispatcher in all situations	-

Table 12: Performance measure relation

The performance measures in detail

In our analysis we use the dynamic performance measurement sheet and graphs to evaluate the scheduling process. The dynamic performance measurement sheet output consists of ten graphical representations of the performance measures. We use performance measure 1 in Graph 18 as an example to explain how we evaluate the performance measures to find scheduling improvements.



Graph 18: Performance measure 1, data capture date: 28 July 2015 14:00

All graphs have a secondary axis that displays the time of capture (right axis). The other axis displays the performance measure value. The horizontal axis displays numbers that represent the time windows. We use this numbering, because the user of the performance measurement tool can change the time window durations. Table 13 displays the settings that we use in the performance measurement analysis.

0-15 min.	15-30 min.	30-60 min.	60-120 min.	120-180 min.	180-240 min.
0->1	1->2	2->3	3->4	4->5	5->6

Table 13: Time window duration settings

All graphs are constructed dynamically and processed into an animated gif (see Figure 19). In that way the red horizontal line is moving upwards and we can analyze and evaluate the performance measure.

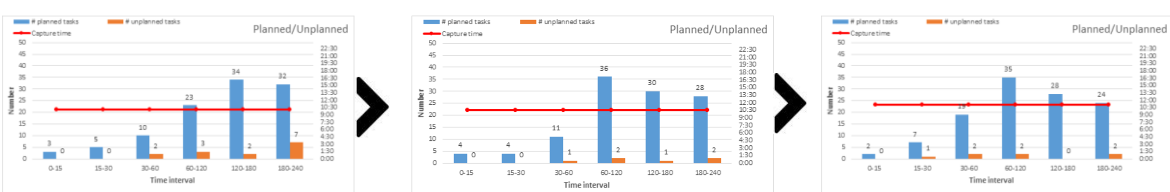


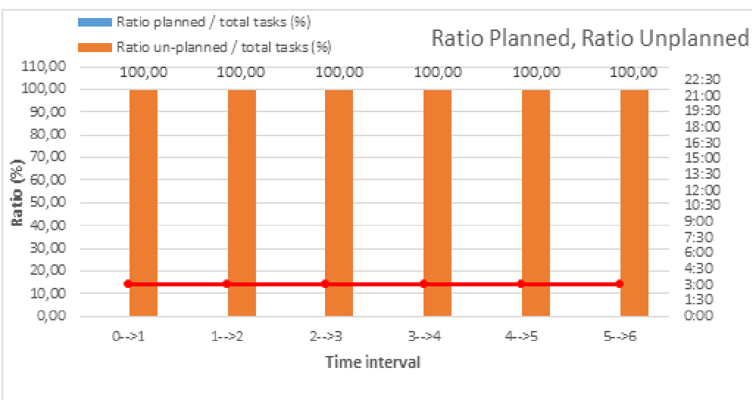
Figure 19: Dynamic performance measurement representation

There are two important aspects in the analysis of the dynamic performance measurement graphs:

- Some performance measures are designed to be used by the dispatcher as a tool to identify future (critical) events. With these dynamic graphs we are able to analyze and evaluate precisely what the dispatcher will see during the day. These performance measure are not directly usable in finding scheduling improvements.
- Other performance measures describe the scheduling behavior during the day. With these performance measures we identify possible scheduling improvements.

Performance measure 1, 2 Planned ratio and Unplanned performance

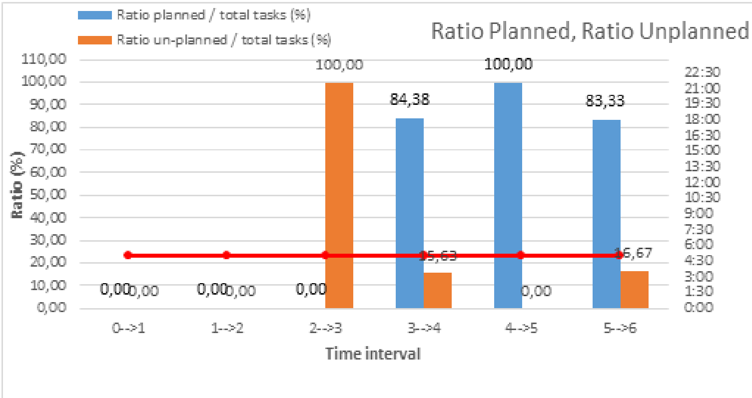
Performance measure 1 and 2 are combined into one graph. These performance measures calculate the ratio between the number of planned and unplanned tasks against the total number of tasks within the considered time window. During the night shift between 22:00 and 6:00 we expect to see no planned tasks, because in this mode all tasks stay unplanned and the operator receives all tasks on a list. During the night shift the operator is completely free to choose which task to perform first.



Graph 19: Performance measure 1,2, capture time: 3:00

Graph 19 shows the performance graph during the night at 3:00. The effect of the CHIP night modus is clearly visible, because the ratio between the number of unplanned jobs and total jobs equals 100%. The ratio planned is in this case 0%.

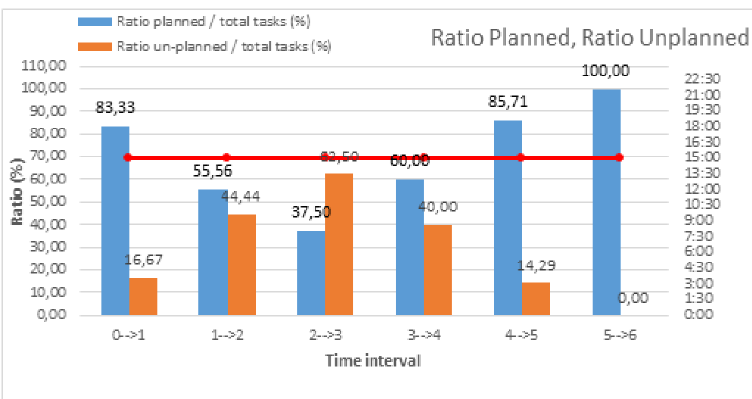
Interesting to analyze is for which future time windows the first tasks are planned by CHIP.



Graph 20: Performance measure 1,2, capture time: 5:00

Graph 20 shows the performance graph from 5:00. At this time the first planned jobs are visible in time window 3-4, where 3-4 represents the time between 1 and 2 hours in advance. This means that the first jobs are planned at the end of the CHIP night modus from 6:00.

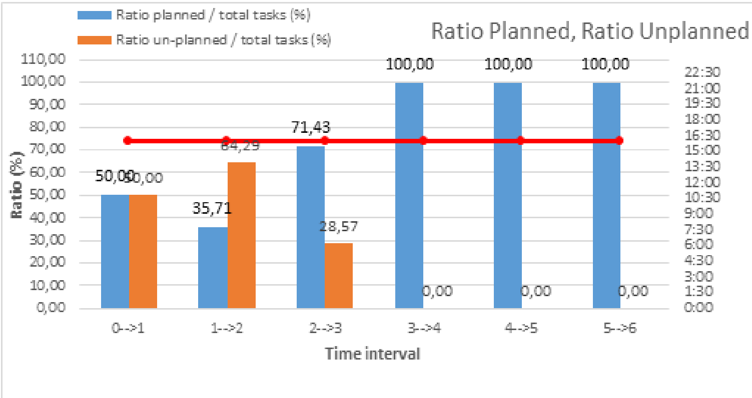
In Section 4.3.2 we identified a critical time window, where the number of tasks exceeds the number of available resources. Graph 14 shows this critical time window around 16:00. With the performance measurement sheet we are able to evaluate the number of unplanned and planned tasks for this specific time. We can also evaluate if this critical time could have be seen in advance.



Graph 21: Performance measure 1,2, capture time: 15:00

Graph 21 shows the performance graph from 15:00. The critical time window exists at 16:00. So, in this graph we have to look at time interval 2-3, or 1 hour in advance. The ratio of unplanned jobs exceeds the ratio of planned jobs (62,5% vs. 37,5%) in this time window.

In this case, the dispatcher was able to see that there existed a critical time window at 16:00 from both the workload graph and performance graph. If we look at the performance graph from 16:00 we are able to evaluate if the dispatcher and CHIP were able to solve this critical time window.

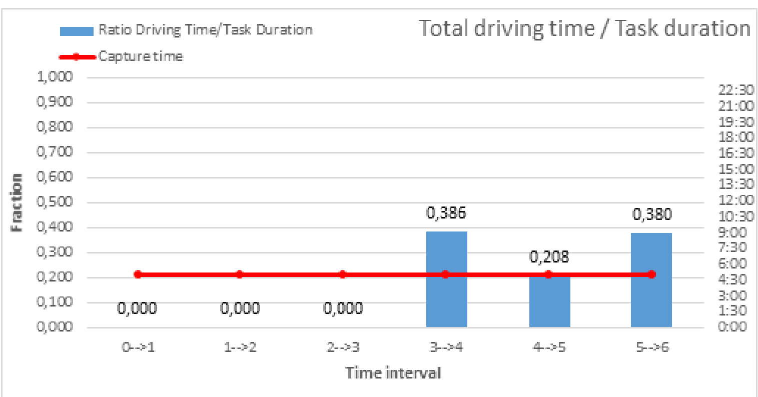


Graph 22: Performance measure 1,2, capture time: 16:00

From Graph 21 can be seen that still 50% of the jobs are unplanned for the next 15 minutes. This means that the dispatcher and CHIP were not able to re-schedule the tasks in such a manner that all tasks were planned.

Performance measure 3 Driving time duration ratio

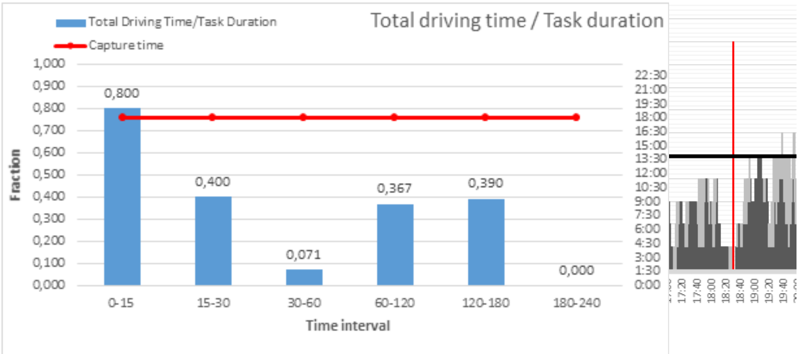
Performance measure 3 describes the ratio between the driving time and task duration per time window. We only consider driving times if tasks are planned, because only in those cases the optimizer within CHIP tries to minimize the driving times for resources. This results in the fact that during the CHIP night modus, the performance graph does not show any results, because all tasks have the status of unplanned. From performance measure 1 and 2 we have seen that the first tasks are planned at 5:00 for the 6 o'clock time window.



Graph 23: Performance measure 3, capture time: 5:00

In Graph 23 it is clearly visible that the first ratios are calculated for 6:00. See time window 3->4 (60 to 120 minutes in advance).

We expected to see smaller ratios closer to the current time. This for the reason that the optimizer continuously changes the assignment of tasks to minimize the driving time for resources. If a task is planned for example 3 hours in advance the optimizer has enough time to change the assignment of this task to optimize the driving time. However, this trend is not visible into the performance measurement graph that we constructed. This could mean that the optimizer makes a fairly good first assignment of tasks to resources, based on the driving time. Another reason could be that other optimization criteria have a higher priority within the optimization algorithm. As introduced in Section 2.2 we do not have enough insights into the optimization algorithm that is built by Inform to verify the exact optimization criteria priorities.

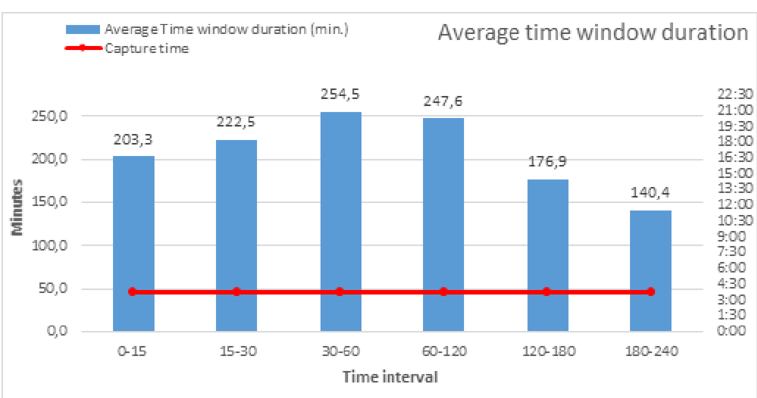


Graph 24: Driving time/task duration, capture time 18:00

The general tendency is that the ratio is lower in periods with a high workload ratio. Intuitively this is correct, because the chance of a long driving time when only a few tasks have to be completed is higher than when there are a lot of tasks. This is visible in Graph 24.

Performance measure 4 Time window duration

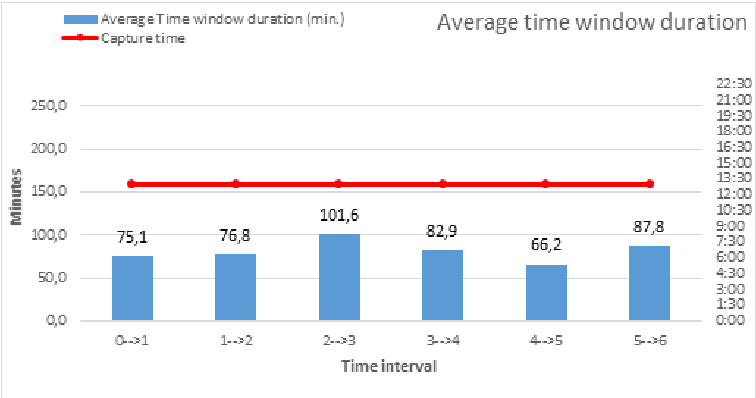
The time window duration gives the average difference between the latest end time and earliest start time of a task per time window. This performance measures cannot be influenced by the dispatcher directly, but gives information that can help to improve the scheduling of tasks. If the dispatcher can see shorter time window durations in advance, he or she can take in account that there is less scheduling flexibility. We now analyze how the average time window durations change during a day. Again the average time window duration is only important for the dispatcher when tasks are planned, because only then they reserve capacity. However, in this case we are also interested in the average time window duration during the night shift. During the night shift the operator receives a list with all tasks that must be performed. If there are a lot of tasks with small time window durations during the night, it is highly difficult for the operator to make a good task order to finish all tasks within the task windows. To see if this is true we also look at the unplanned tasks during the night shift.



Graph 25: Average time window duration, capture time 3:45

The average time window duration is higher during the night until 6:00 than for the rest of the day. Graph 25 shows the average time window duration calculated on 3:45. There is thus more scheduling time flexibility during the night.

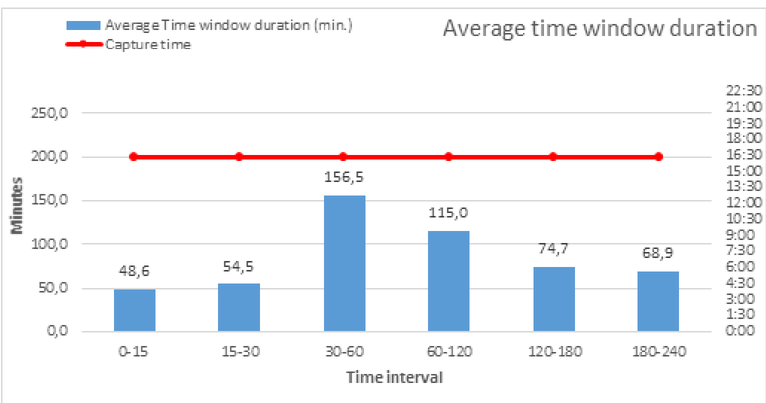
We showed that there is more scheduling flexibility during the night. However, from the graph we are not able to evaluate if there is enough scheduling flexibility so that the operator can choose his or her own task order.



Graph 26: Average time window duration, capture time 13:00

The average time window duration is much lower in peak times during the day than at night, see Graph 26.

During peak times many aircraft do have a very short available ground time. In those cases a fast turnaround is needed to facilitate an on-time departure. This clarifies the finding from Graph 26. Graph 27 shows that the decrease in average time window duration is visible well in advance. This means that the dispatcher is able to see a decrease in the scheduling flexibility from the performance graph in advance.

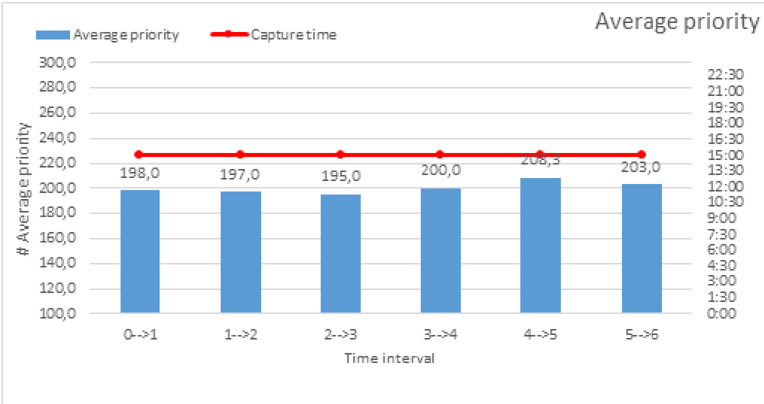


Graph 27: Average time window duration, capture time 16:15

If we look into the future for time window 120-180, and 160-240 minutes the decrease in average time window duration is clearly visible. This information is thus available well ahead of time. Graph 27 is made on 16:15 and shows shorter average time window durations.

Performance measure 5 Average priority

The average priority cannot be influenced by the dispatcher, because priorities are fixed values based on the type of task. We only consider task priorities if tasks are planned, because the optimizer considers planned tasks only. We expect to see differences in priorities during the day, because of the flight schedule. Priorities of ICA flights are higher than Europe flights.



Graph 28: Average priority, capture time 15:00

We did not see large differences in average priorities during the day. We also did not find differences in priorities for different times during the day.

This means that the assumption of different priorities during the day based on the difference in priorities of ICA and Europe flights does not hold. This means that there are no large differences in priorities between the Europe and ICA flights during the day.

Performance measure 6 Earliest start plan

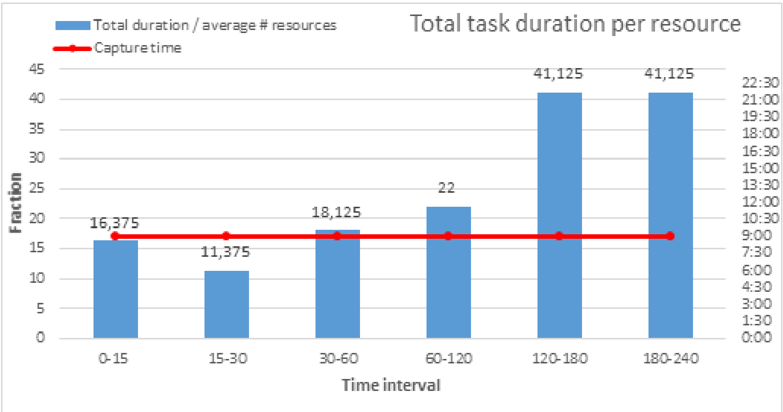
The average earliest start is the difference between the start time and the earliest start time of a task for each time window. The objective is to schedule tasks as early as possible within their time windows. We expect to see larger averages during peak hours, because then there is less scheduling flexibility to plan all tasks on their earliest start time. However, this trend was not visible in the earliest start plan graphs. This means that CHIP does not schedule tasks earlier in their corresponding time windows during off-peak hours than at peak hours. This means that the scheduling in CHIP can be improved during off-peak hours by scheduling tasks earlier in their corresponding time windows. With this performance measure the dispatcher is able to directly evaluate certain schedule changes that he makes.

Performance measure 7 Average number of resources

The dynamic graph of the average number of resources does not provide new insights. The graph is a helpful tool for the dispatcher to numerically see the changes in number of operator in the future.

Performance measure 8 Total average task duration per resource

We expected to see longer total task durations in peak times. For the reason that in peak times more tasks are handled by the same number of operators this automatically leads to a higher total task duration if compared to off-peak hours. The total task duration differs per time window, but there are no big differences between peak and off-peak hours. However, there are some time windows with a low total task duration, but we consider these as outliers. We consider these as outliers, because they occur in off-peak hours. The assumption of a higher total task duration in peak hours was not visible from this performance graph this means that the workload is relatively stable during the day.

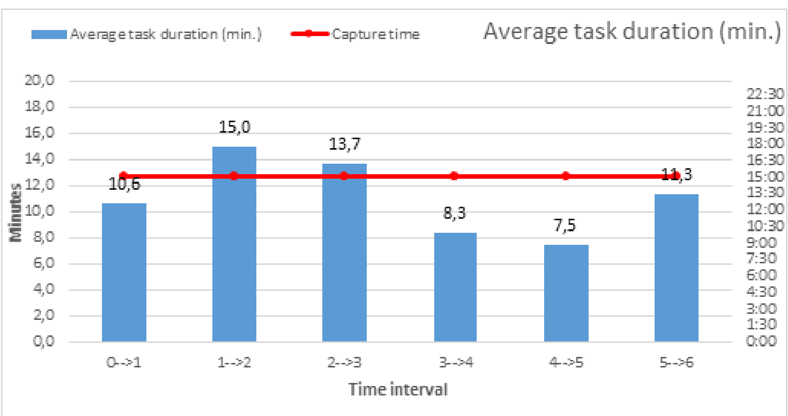


Graph 29: Average task duration per resource, capture time 9:00

When analyzing this performance graph, one should keep in mind that the time window durations of, 60-120, 120-180, and 180-240 are longer then the first three time windows. Automatically this leads to longer total task durations per resource, see Graph 29.

Performance measure 9 Average task duration per time window

The task durations of different aqua service tasks are relatively the same. For this reason we do not expect to see many differences in the average task durations during the day.



Graph 30: Average task duration, capture time 15:00

We did not expected to see large differences in average task durations during the day. However, at 15:00 there is a notable increase in the average task duration to 15 minutes.

To clarify the sudden increase we looked at the dynamic workload graph from 15:15. However, this graph does not provide us insight to clarify the sudden increase. We therefore conclude that there is probably a special task that has a long duration at this moment, which does not occur often. The assumption of minor differences in the average task duration does not hold. During the day there are fluctuations in the average task duration per time window. This means that there exists a variation in the task duration per type of task that is larger than we expected.

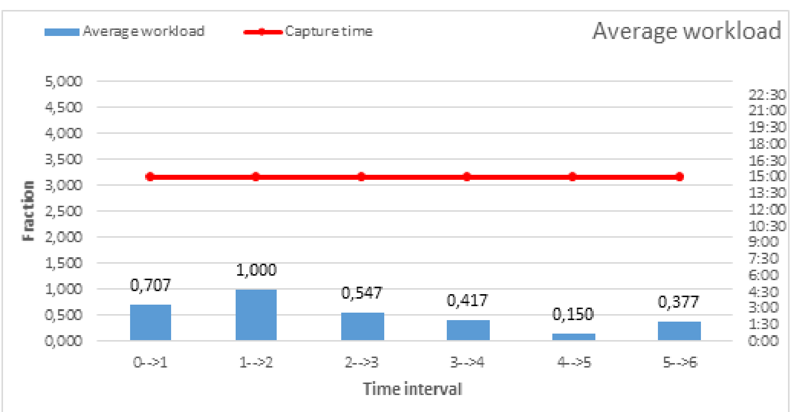
Performance measure 10 Number of tasks that end in time window

The number of tasks that have an end time in a time window is a very helpful measure for the dispatcher so see when the risks of task delay are highest. We expected to see that the number of tasks that end in a time window would become more stable during peak times, because at those times there are more tasks. However, the expectation was not visible in the performance measurement graph. The total numbers of ending tasks are fluctuating constantly during the day. We did not find any other patterns or noteworthy events.

This does not mean that this performance measure is not useful. The dispatcher is constantly able to see in advance when the risks of delay are highest.

Performance measure 11 Time loading

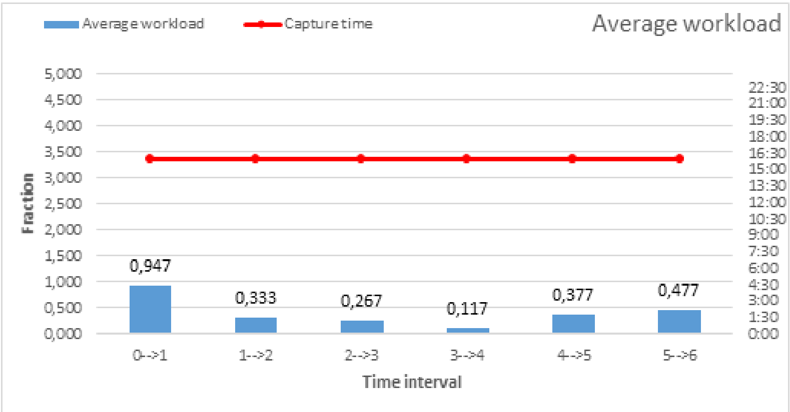
We expect to see that we see the same results with respect to the workload as from the dynamic workload graphs. From the dynamic workload graphs we found that after 15:00 there are some critical time windows, where the number of tasks exceeds the number of operators. We argued that these time windows could be seen in advance by the dispatcher based on the information that is provided by the dynamic workload graph. Interesting to see is whether these critical time windows can also be seen from the performance measurement graph. The time loading should not exceed one, because in that case there are more tasks than available resource capacity. This occurs only when the dispatcher manually changes the assignment of tasks, because the optimizer within CHIP cannot schedule tasks that do not fit into a current schedule.



Graph 31: Average workload, capture time 15:00

In the dynamic workload graphs in Section 4.3.2 we saw a sudden increase in the number of tasks, which exceeded the number of resources at 16:00. Graph 31 shows the performance graph from 15:00, from this graph we cannot see that the critical time window would arise around 16:00.

From Graph 31 is not directly visible that a critical time window would arise around 16:00. However for the second time window (15 minutes in advance) there is a time loading of 1.00. This means that full resource capacity is needed and that no more tasks can be planned by the optimizer. The risks of task delay and a delay in task start time in this time window are very high. This should be a signal for the dispatcher to closely examine which tasks are planned in this time window and if there are tasks that could be scheduled to a later time if problems arise.



Graph 32: Average workload, capture time 16:00

Graph 32 shows the performance graph from 16:00. At 16:00 there is a critical time window. The time loading for this time window is 0,947. This high workload occurred at the last moment and was not visible in advance from the performance graph.

The time loading graphs are a helpful tool for the dispatcher to see critical time windows in advance. However, there still arise critical time windows at the latest moment which are not predictable of foreseeable with this performance graph.

4.3.4 Actual performance

In the stakeholder analysis in Chapter 4.1 we identified the pilots, crew, and passengers as dependent stakeholders. They are dependent on the timely completion of all aircraft service tasks so that their aircraft can leave on time. This research is focused on the scheduling performance of KLM AS departments, in specific the Aqua & Toilet service department, and not on the actual departure performance. However, there is a link between this research and the actual departure performance. It is interesting to show how this research contributes to the current research[9] that is done at KLM to the departure non-performance.

Currently there is a research going on that studies the contributions of separate (sub) processes to the departure non-performance. These (sub) processes are for example AS processes. In that research a turnaround is considered as a project that has a critical path of activities. Any delay of an activity on the critical path directly impacts the planned project completion date. The project team tries to use the dependencies between activities to determine the causal relationships between activities on the critical path. The management of KLM AS estimates that in 1% of the cases the aqua service tasks are the cause of an actual departure delay. This research contributes thus to the optimization of the separate (sub) processes.

[9] Research project is called APK

4.4 Conclusion

This chapter described the stakeholder analysis, schedule performance over time, dynamic workload graphs, and the dynamic performance graphs. The performance measures were defined according to the stakeholder typology of Mitchell et al. (1997). This stakeholder typology uses legitimacy, urgency, and power to classify the different types of stakeholders. We described all stakeholders that are responsible and affected by a schedule created in CHIP. The most important stakeholders are KLM Aircraft Services, dispatchers, DAM, DHM, and DMA. These are all definitive stakeholders. A definitive stakeholder has the power, legitimacy, and urgency to afffect a schedule. We used the stakeholder analysis as a basis for the performance measures.

The most important part of this research is the schedule performance over time. If we can assess the schedule performance over time we can find schedule improvements, because we are able to see what happens during schedule creation and execution. To find schedule improvements was not the only objective that we had. We were also interested if it is possible for the dispatcher to see critical events and schedule changes in advance. In this way we can improve the scheduling process.

We used and developed two measurement techniques specifically to assess the schedule performance over time. We developed a dynamic workload graph that displays the number of resources against the number of tasks over time for a specific day. The other measurement technique that we used is a dynamic performance measurement sheet that shows all different kinds of performance measures based on the stakeholder analysis.

In our analysis we used the data from the Aqua & Toilet service department and specifically the water related tasks. We only used the data from aqua service tasks, for several reasons: aqua service tasks are relatively short which makes it a very dynamic scheduling problem and aqua service tasks do not use multiple options or configurations for the same task which reduces the analysis complexity.

The input data that is used for the two analyses are gathered from the live or operational environment of CHIP. For every 15 minutes we saved all data related to the tasks and shifts that were available at that time. Before we could use this data we programmed a script in Python to clean both data files.

We analyzed the dynamic workload graphs by using animated gifs. From the animated graphs we identified and described different kind of striking schedule changes. All these observations and corresponding workload graphs can be found in Table 9. We found that:

- Critical time windows in which the number of tasks exceeds the number of resources are well known in advance.
- The early departure of operators in the night and day shift is facilitated by the dispatcher. At those time tasks are shifted to a later time which is often not desirable.
- The optimizer time window of four hours is clearly visible from the tasks that are shifted and planned in their corresponding time windows.
- The effects of dispatcher handling are clearly visible when tasks are fixed in their time window. At those times the number of tasks exceeds the number of resources and the optimizer cannot change the assignment of those tasks, because they are fixed by the dispatcher.

The second analysis encompasses the performance measurement sheet. This performance measurement sheet can be used by the dispatcher to look into the future. We constructed performance measures that can help the dispatcher to detect possible critical scheduling issues in advance. To evaluate and find scheduling improvements we developed a measurement technique that is also based on dynamic graphs. For every 15 minutes we constructed the performance measurement graphs and turned them into animated gifs. We searched for critical scheduling issues and if these issues could have been known in advance. In that way the dispatcher could have solved these issues well in advance. We found that:

- The dispatcher is able to see critical time windows with a substantial number of unplanned jobs well in advance.
- Most of the performance measurement graphs do not give results during the CHIP night modus, because in this modus tasks stay unplanned.
- We expected to see that driving times would become smaller closer to the current time. However, this was not visible from the performance graph.
- Driving times are smaller during peak hours.
- Time window durations tend to be longer during off-peak hours including the night.
- A decrease in the time window duration is visible from the performance graph well in advance.
- The variation in the average earliest start time of tasks is minor during the day. This means that there are no differences between peak hours and off-peak hours.
- The total task duration per resource is relatively the same during the day. This means that operators are almost facing the same workload during the day.
- The average task duration of aqua service tasks is stable during the day and night.

- The performance graph of the number of tasks that end in a time window does not provide us schedule improvements. However, this graph is a perfect tool for the dispatcher to assess the risk of task delay per time window.
- A time loading higher than 1 indicates that the dispatcher did change the schedule.
- Critical time windows are often foreseeable in advance.

To conclude, the dynamic workload graphs and the dynamic performance measurement sheet are a helpful tool for the dispatcher to see critical scheduling issues in advance. Both sheets are constructed in Excel and VBA and are therefore easy to use on every computer. The dynamic graphs are also a helpful tool for those who want to study the scheduling behavior over time as we did in this research. In the next chapter we study how all scheduling observations and findings can help to improve the scheduling of aqua service tasks.

5 Scheduling improvements

In this chapter we discuss the potential schedule improvements that we gather from the data analysis, stakeholder analysis, and literature review. We do not only consider schedule improvements, but also scheduling improvements that need to be considered by the dispatcher. Section 5.1 discusses the improvements gathered from the dynamic workload graphs. Section 5.2 provides improvements from the dynamic performance measurement sheets. Section 5.3 discusses how the analysis tools can be used by the dispatcher to improve the scheduling of aircraft service tasks. Section 5.4 describes the implementation.

5.1 Schedule improvements based on dynamic workload graphs

From the dynamic workload graphs in Chapter 4, we found all kinds of different schedule changes. In this section we relate these observations and changes with the scheduling theory and discuss if these changes are desirable from a scheduling perspective. In this way we are able to present schedule improvements.

We found that almost all critical time windows with a high task density are known in advance. These tasks clusters occur mainly due to the flight schedule which holds certain peak departure times. The peak times are mostly the same for every weekday, so our findings are applicable for every weekday. The front end screen of CHIP holds a Gantt chart for the coming hours (see Section 2.2.1). One major disadvantage of this Gantt chart is that it is highly difficult to see peak times and critical time windows in advance. We can help the dispatcher with this insight by dynamically presenting the dynamic workload and performance graphs besides the Gantt chart. The dispatcher is then able to see peak times and critical time windows in advance. This is completely in line with literature where we found that information can be described by the quality of information and the timeliness of information. If we can provide information earlier to the dispatcher it is less difficult to anticipate on this event. We also discussed the four stage model of Cowling and Johansson (2002) in handling real time information. The dynamic workload graph and performance graphs provide real time information that can help the dispatcher in the detection and classification step of this model (see Figure 20).

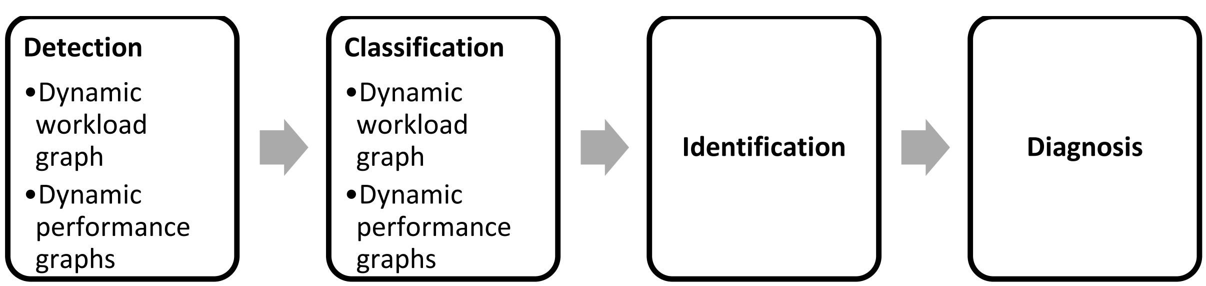


Figure 20: Workload and performance graph positioned in to the model of Cowling and Johansson (2002)

However, for the identification step, more information is needed that is not provided by the workload graph or performance graphs. For example, the dispatcher sees a critical time window in the coming hours from the dynamic workload graph. The dynamic workload graph does not give exact task information. For this identification step the dispatcher should use another information source. This other information source could be the Gantt chart. For example, the dispatcher notifies a critical time window one hour in advance from the dynamic workload graph. The dispatcher should then search in his Gantt chart to this specific time and see which tasks are planned. The next step is to set the right solution, re-scheduling some tasks for example.

We found that there is a trend to send operators home early before the end time of their shift. To facilitate the early departure of operators tasks are shifted to a later time window without anticipating on the results of this task shift. An operator should never be sent home early if therefor a task must be shifted to a time window with a higher task density, since this leads to an unnecessary increase in the workload which could become critical.

From literature, we found that scheduling is more than only the mathematical construction of a schedule. The human aspects cannot be omitted. Facilitating the early departure of operators is a good example of that. The dispatcher helps the operator so that the operator can leave earlier than his or her planned shift end time. An exception to the early departure are the temporary workers. Temporary workers are often hired on an hourly rate basis. If the schedule facilitates an early leave of these temporary workers, it saves money to send them home. However, this should only be done if there are no more tasks planned in the current shift. If tasks can be re-scheduled to the next shift without drastically increasing the workload, an early leave of temporary workers can also be facilitated.

From the dynamic workload graphs we found that it seems that breaks are scheduled at the last moment, hereby omitting the high workload at certain times. Breaks are scheduled by CHIP before a shift starts, but not yet fixed and assigned to a specific time. Breaks have a low priority if compared to other service tasks. Due to this low priority CHIP is not likely to schedule a break. To force CHIP to schedule breaks, the corresponding time window (earliest start and latest end time) is set very narrow to 60 minutes. At this moment there is more research needed to the specific break task behavior. From a scheduling perspective it would not be favorable that break tasks are planned on their latest end time. This would mean that CHIP in fact has limited time space for a break, but it must schedule a break due to labor legislation. This could result in the effect that we identified with our dynamic workload graphs. Figure 21 gives a desirable break schedule where breaks are not scheduled on their latest end times (vertical lines) and not all breaks are scheduled at the same time. So there is still resource capacity at that moment, and all breaks are scheduled within their corresponding time window.

Operator/Time	t	t+1	t+2	t+3	t+4
1			break		
2		break			
3			break		
4				break	

Figure 21: Break scheduling

During the shift change between 14:00 and 14:30, only one task is handled by the operators, while there are 11 active operators. The change between operators on a truck takes time, but this cannot result in a performance loss this big. The shift change uses a finishing task that indicates the shift end for an operator. This finishing task is planned approximately 20 minutes prior to the shift end of an operator, so at 14:10 (for the day shift). All operators that are active in this shift receive the finishing tasks almost at the same time. Of course, there could be some differences, because some operators are still active with a task. It would be better to gradually assign the finishing tasks to operators. This results in lower capacity drop during the shift change and a smoother shift change. Also the change of trucks will be easier, because there is more free parking space which also results in a time saving.

The dispatcher should minimize his or her actions and changes to the schedule. The dynamic schedule changes and many different optimization criteria make it to a highly difficult and dynamic scheduling system. This scheduling system makes it highly difficult for a dispatcher to make smart changes to a schedule. In practice we saw many times that dispatchers make schedule changes. They argued for example that it is smarter to assign two tasks that are located closely to each other to the same operator. Yes, this is true if we only have driving time as optimization criteria. Many times the human aspects play a role in these schedule changes. Operators do not like it if they are driving to a gate position and they meet another operator on the way back, they immediately think: why must I drive to this position while there was already another operator? A dispatcher should only make changes to a schedule if he or she has information that is not available for CHIP, and that this information leads to a smarter schedule.

5.2 Schedule improvements based on dynamic performance measurement sheet

The performance measurement sheet had two objectives: first to help the dispatcher to look into the future, and second to evaluate the scheduling process. All performance measures are based on the stakeholder analysis and objectives set by the CHIP project team (see Section 4.1). In this section we discuss the improvements and observations from the dynamic performance measurement graphs.

The first important note is that the performance measurement graphs do not provide extra helpful information when the CHIP night modus is active. This is for the reason that we only consider tasks that are planned by the optimizer and unplanned tasks are not considered by the optimizer yet. From performance measure 1 and 2 we saw that the first tasks are planned for the first shift that starts at 6:00. These planned tasks are visible around 5:00 in the performance graphs. The performance graphs can thus be used by the dispatcher from 5:00. Again also here, the sooner information is provided to the dispatcher the better. Performance measure 1 and 2 do not give extra information that can be used to improve the schedule.

Driving times tend to be shorter in time windows with a high workload. This means that CHIP is able to optimize the driving times to tasks. However, we also expected that the ratio would become smaller closer to the current time, because the optimizer is continuously changing the assignment of tasks to optimize the driving time. This trend was not directly visible from the performance graph. If KLM AS wants to steer more on driving times, there is more research needed to the specific behavior of driving times with different optimizer settings. From the performance measurement graph analysis we are not able to evaluate how good or bad the optimizer optimizes the driving times, because we do not have a direct reference. We do not have a reference, because we did our analysis on the current CHIP system that is active within the operation. With this system it is not possible to simulate the same day again with different settings.

From the dynamic graph of the average time window duration we learned that time windows in the night are longer than time windows during the day. This longer time window duration gives more scheduling flexibility, because it is easier to shift a task to an earlier or later time. Experienced dispatchers will know precisely when time windows become smaller, which is mostly during peak times. However, with our performance graph we are able to visualize information for all dispatchers.

From the average priority performance graph we did not find how the priority of tasks influence the schedule of aircraft service tasks. This means that at this moment we are not able to evaluate the effects on the schedule of high and low priorities with the performance graph. The average earliest start is a measure that is perfectly usable by the dispatcher to see if tasks are planned as early as possible. However, the performance graph does not give us new schedule improvements.

The average number of resources available graph does not provide us new insights to improve the scheduling process, but it does offer insights into the exact number of operators that are active and become active within the next hours. At this moment this information can only be gathered by the dispatcher from the Gantt chart, but he has to count the number of operators from the screen. If the dispatcher knows exactly how much resource capacity exists in future time windows, he or she can use this information when re-scheduling tasks if necessary.

From the performance graph of the total task duration per resource we saw that the total average task duration per resource is relatively the same for the whole day. This means that the workload for operators in the morning is almost the same as for the afternoon. The performance graph does not provide schedule improvements. The same holds for the average task duration per time window.

The number of tasks that have an end time in a time window performance graph is a very helpful tool for the dispatcher to evaluate the risk of delay in certain time windows. We cannot use this performance graph to propose schedule improvements.

The performance measurement graph of the workload is a perfect tool to evaluate the handlings of a dispatcher. For the reason that CHIP is not able to schedule more tasks at a certain moment than available resource capacity. This can only be done by the dispatcher. So if the workload exceeds 1.0 we can conclude that this happened due to the changes in a schedule made by the dispatcher. The workload performance graph does not give us new ideas of schedule improvements.

5.3 Dispatcher's tool

De Man (2014) mentioned that it would be beneficial if CHIP shows the utilization of personnel so that the dispatcher has an idea about the free capacity for a given moment. He also discussed the possibility of showing this value for 15-minute time intervals, so that the dispatcher knows how critical a schedule is. De Man (2014) studied the aircraft refueling department specifically, but his statement is also applicable for the Aqua & Toilet service department. In this research we transformed this idea into the workload and performance measurement method.

In this section we discuss how the performance measurement sheet and dynamic workload graphs can be used by the dispatcher to improve the scheduling process in practice. Important to note is that we only studied the theoretical working of the performance and workload graphs by using the data from one day (28 July 2015). This gives a few practical shortcomings that need to be considered when testing and using the graphs in a practical setting:

- The data that is gathered from the live environment is pre-processed by a script. This script must be run manually and should be automated when using the tools in practice.
- The pre-processed data must be imported into excel by an import tool that is written in VBA. The import tool must be loaded by hand, in the future this should be automated.

We already mentioned that the dispatcher only sees a Gantt chart with all operators and aircraft service tasks as in Figure 22. From this graph it is highly difficult to obtain an idea and feeling about the current and future workload. An idea about the future operator capacity is not visible. Our dynamic workload graph would be a very helpful tool for the dispatcher to obtain a good idea about the past and future workload, future operator capacity, and critical task density.

The dynamic workload graph can be constructed at the start of the day at 0:00 and updated every 15 minutes. As said before, at this moment it is not yet possible to automatically load the data into the excel worksheet, but this is something that should be solved in the future. The dispatcher should check before he starts working how the tasks and resource capacity are distributed during his shift. Besides the dynamic workload graphs the dispatcher should also check the performance measurement sheet. In that way he can combine the information of the workload graph and numbers from the performance sheet.

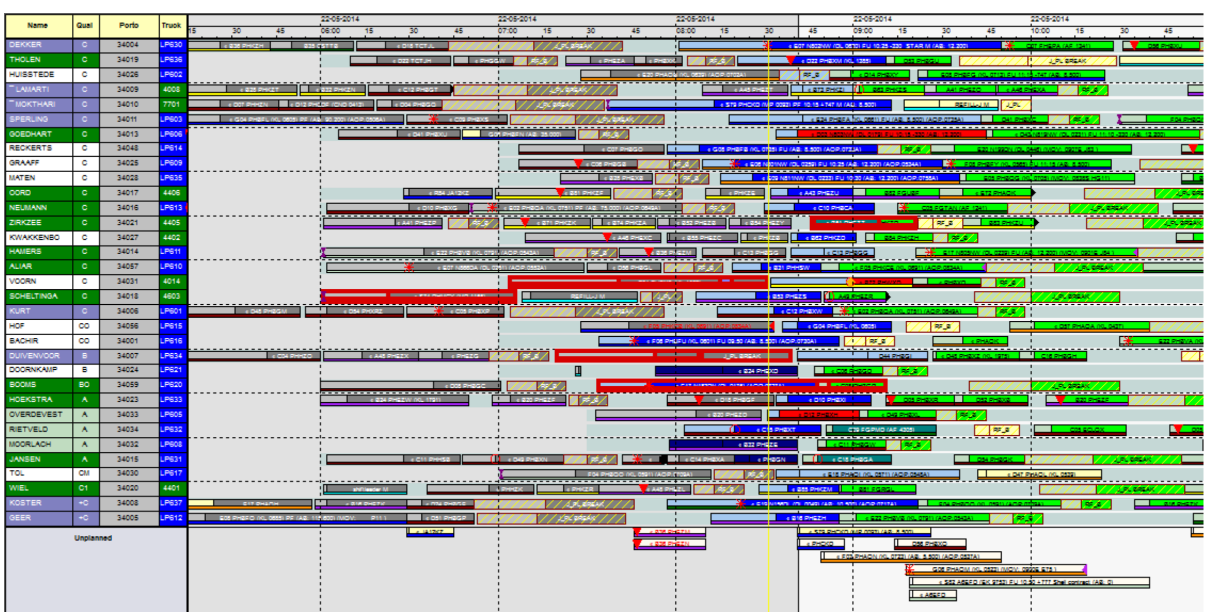


Figure 22: Chip screen shot

If the dispatcher sees a critical time window where the number of tasks exceed the number of operators from the dynamic workload graph he should focus on this time window. If this critical time window is not solved by CHIP, and he or she sees that tasks are unplanned from the performance measurement sheet he should take appropriate actions.

In Section 2.3 we discussed the position and duties of a KLM AS dispatcher. We explained the information sources that a dispatcher uses and the position within the organization. Table 14 shows the position of a dispatcher within the (new) information sources. The dispatcher should use the performance graphs, the workload graphs, and CHIP as a decision support system that helps the dispatcher in making the best assignment of tasks according to the optimization criteria.

INPUT INFORMATION		
Workload graphs	CHIP	Performance measurement
The dispatcher		

Table 14: Dispatcher's role position and information provision

The dispatcher should:

- Only make changes to the schedule if CHIP cannot schedule a task automatically. In almost all cases CHIP can schedule a task much better than a dispatcher can.
- Constantly keep track of critical time windows in advance, which are visible from the workload graph. In those cases he should take appropriate actions, because CHIP is not able to schedule for this non-performance.
- Use the performance measurement graphs and workload graph as a learning mechanism to see critical schedule events in advance.

5.4 Implementation

Implementation of a new tool will only be successful if someone has the full collaboration and acceptance of the person who has to work with the new tool. In this research we focused on the construction of the workload graphs and performance measurement tool rather than investing time in the acceptance of the tool by dispatchers.

We recommend collaborating closely with the dispatchers in the further development of the tool. Important aspects are:

- Closely examine and discuss the performance measures that we thought were important and helpful for the dispatcher in creating a schedule.

- Evaluate the use of future time windows that we have chosen: 0-15, 15-30, 30-60, 60-120, 120-180, and 180-240.
- Most of the dispatchers have their own 'way' of scheduling and idea about good scheduling practices. In many cases they only focus on one optimization criteria. If one is able to make all those ways visible, one is able to evaluate those ways with the performance measures and graphs. In that way one is able to provide evidence in why some scheduling ideas are good or not that good. This gives a better discussion with the dispatcher, than rather saying "this is not good".

5.5 Conclusion

In this chapter we evaluated the scheduling improvements from the workload graph, the performance measurement graphs, the dispatcher's tool, and the implementation of the dispatcher's tool.

The findings from the dynamic workload graphs:

- Almost all critical time windows with a high task density are known well in advance.
- Peak times are closely related to the flight schedule. The flight schedule is often the same for normal weekdays in a specific period this means that the results are also applicable for other week days.
- The workload graph can help the dispatcher with the detection and classification step of the model of Cowling and Johansson (2002). However, for the identification step the dispatcher needs the current Gantt chart to search for specific tasks.
- Operators should never be sent home early if therefore a task must be shifted to a time window with a high workload. This leads to unnecessary increases in the future workload which is or can become critical. Temporary workers are an exception for this, because sending them home early saves money due to hourly payment construction.
- Breaks are often scheduled on their latest end time. This is not favorable and AS should consider to examine the break scheduling that we propose in Figure 21.
- The scheduling of aircraft service tasks is highly dynamic. A human is unable to evaluate all optimization criteria in once. Therefore, the dispatcher should minimize his or her changes to the schedule.

The performance measurement sheet had two objectives: first to propose scheduling improvements and second to provide the dispatcher with future scheduling information.

The findings from the performance measurement graphs are:

- The performance measurement sheet does not present results during the CHIP night modus, because we only consider tasks that are planned.
- The first tasks are planned at 6:00. The dispatcher is able to see this from the performance measurement sheet around 5:00. This means that the first tasks are planned one hour in advance.
- Driving times tend to be shorter during time windows with a high workload.
- The scheduling flexibility is larger during the night due to longer time window durations.
- Presenting the average number of resources besides the Gantt chart is a helpful tool for the dispatcher to see future resource capacity.
- The average total task duration per resource tends to be the same for the whole day.

The main findings concerning the dispatcher's tool are:

- The current setting of the dispatcher's tool can be used in practice, but the import of data should be done by hand.
- Currently, the dispatcher sees only a Gantt chart as the main information source. The dispatcher's tool will provide the dispatcher a lot of extra information that he should use to monitor the scheduling process.
- CHIP is unable to schedule tasks when the workload exceeds 1.0. If this occurs tasks are unplanned or remain unplanned often this is visible from the dispatcher's tool. If the dispatcher sees this event in advance he is able to take appropriate actions.

The main findings about the implementation of the dispatcher's tool are:

- KLM AS should closely collaborate with the dispatchers and DMAs to further improve the dispatcher's tool.
- We showed that many dispatchers do have a own idea about 'good' scheduling practices. In many cases they only focus on one optimization criteria. The dispatcher's tool can be used to evaluate certain scheduling choices of the dispatcher and show the dispatcher why a certain choice is not 'good'.

6 Conclusion and recommendations

Section 6.1 addresses the conclusion and discussion of this research. In Section 6.2 we give recommendations for KLM AS and for further research.

6.1 Conclusion and discussion

This research contributes to the scheduling process optimization of aircraft service tasks at KLM AS. The timely departure of aircraft heavily depends on the on-time performances of all processes that are needed before an aircraft can leave. This on-time performance can only be realized if the scheduling processes of aircraft service tasks are efficient and optimized according to the priorities of all stakeholders. Aircraft service operations are controlled by dispatchers. They use CHIP as decision support system to schedule aircraft service tasks. Before this research there was a common feeling under AS management that CHIP was underperforming. This underperformance is caused by a multifaceted set of issues, but mainly due to limited insights into the scheduling process and actions of the dispatcher. This led to the following main research question:

How can KLM Aircraft Services improve the online scheduling of aircraft services, within the current organization and IT structure?

Due to research time limitations we narrowed this research towards the scheduling of aqua service tasks. To provide insights into the scheduling behavior we developed a dynamic workload graph and a dynamic performance measurement tool. The dynamic workload graph presents past, current, and future workload for a given day and is updated every 15 minutes. The task duration and driving times are incorporated in this workload graph.

The dynamic performance measurement tool presents performance measures related to the scheduling process and schedule itself. Some performance measures are used to evaluate the schedule and others are solely constructed to increase the quality of future information for the dispatcher.

KLM AS can improve the online scheduling of aircraft service tasks by implementing the dynamic workload graph and dynamic performance graphs in practice. In this way we provide the dispatcher with extra helpful information about future events that is not available at this moment. This future information can be used by the dispatcher to make changes to the schedule pro-actively instead of re-actively. Important is that the dispatcher should only make changes to the schedule if CHIP is not able to schedule a task. This occurs when there are more tasks at the same moment than available resources.

KLM AS should also consider the early departure of operators and the scheduling of break tasks at the last moment. We showed that operators are sent home before their shift ends.

Per se, this is not a problem as long as all task are completed at the end of a shift. However, dispatchers re-schedule tasks to a next shift, resulting in unnecessary increases in the workload that may become critical. We also showed that breaks are scheduled at the last moment on their latest end times. This results in a resource capacity drop during break periods. We propose a break schedule that is gradually implemented, which results in a more stable resource availability.

On the long-term AS should create awareness and learn the dispatchers that CHIP is a decision support system. A decision support system that can schedule tasks according to multiple optimization criteria. If there are no critical events during a day, a dispatcher should minimize his or her changes to a schedule, because a human is not able to weight all optimization criteria at once. In practice we were facing many times that dispatchers did not agree with the schedule changes that CHIP made. However, a schedule change can look illogical on the first hand, but a dispatcher should learn that there is always a certain reason for this schedule change by CHIP. KLM AS should use the performance measurement graphs and dynamic workload graph as a tool to create this awareness and increase the trust in CHIP by the dispatcher.

This research has several limitations:

- We did not test and discuss the dynamic workload tool and dynamic performance measurement sheet with dispatchers in practice.
- We only focused on the aqua service tasks from the Aqua & Toilet service department.
- CHIP should support the dispatcher in all situations. In this research we evaluated the tools on one specific day in the summer period. This summer period is the busiest period of the year with respect to the number of aircraft service tasks. However, we did not study the effects of the tools for other periods.

The results of this research are the first steps in improving the scheduling process of aircraft services. In the future KLM AS should closely collaborate with the dispatchers and DMAs to further improve the performance measurement sheet and workload graph to their specific needs. It also should start to categorize and list all sort of critical events that can occur during a day. Next, these events should be simulated in the performance measurement sheet and workload graph so that they can learn how to detect and solve these events. All these events can be used as a learning tool for dispatchers to detect how to solve critical events that are known. This will result in a continuously learning curve for both the DMAs and dispatchers.

6.2 Recommendations

This section provides recommendations for AS and for further research. We divide the recommendations in three sub categories general, further research, and other. The general sub part describes recommendations that AS should make according to this research. The further research sub part provides ideas for a next researcher on this research subject.

General

The first step should be a pilot together with the dispatchers. For this pilot the performance measurement sheet can be loaded by hand. It is Important that someone can run the python script that changes and cleans the data that is gathered from the live environment. If this data cleaning step is skipped, the performance measurement sheet will give errors and a correct outcome is not guaranteed. With this pilot AS can show that dispatchers can be helped with the scheduling of aircraft service tasks and subsequently improving the scheduling process.

We showed that performance measurement in the future is a good tool and way to assess the quality of scheduling by the optimizer. This measurement technique can also be used to tune the optimizer settings. The optimizer tuning is done by the CHIP project team. They use a CHIP test environment that can simulate a given day. With future performance measurement they are able to evaluate the scheduling behavior of the optimizer and compare these afterwards with different settings. The most important aspect is to first discuss the changes that the researcher is expecting to see and precisely stating what the researcher is trying to optimize. Only then one is able to make useful changes to CHIP.

Further research

We showed that the dynamic performance measurement sheet can be a helpful tool for the dispatcher to improve the scheduling of aircraft service tasks. Before this measurement sheet can be applied in practice there is more research needed. The data coming from the live environment is now loaded into a python script and Excel sheet by hand, when using the sheets in practice this should be done automatically.

We only constructed the dynamic workload graphs and dynamic performance measurement sheet for aqua service tasks (water fill, water drain, water refresh, and refill). We believe that this measurement technique can also be used at other aircraft services, but a next researcher should specify this further.

We showed that a dispatcher can be helped with a tool that provides future schedule information. In a next step it is important to study the practical side, most important is to establish goodwill by the dispatchers so that they are going to use the tools. We recommend a next researcher to invest enough time to discuss the workload and performance measurement tool in full detail with the dispatchers.

When programming the sheets we made several important assumptions, we recommend a next researcher to discuss and read these assumptions carefully.

Bibliography

Berglund, M., & Karltun, J. (2007). Human, technological and organizational aspects influencing the production scheduling process. *International Journal of Production Economics, 110*(1–2), 160-174. doi: http://dx.doi.org/10.1016/j.ijpe.2007.02.024

Bonifaci, Vincenzo, & Stougie, Leen. (2009). Online k-Server Routing Problems. *Theory of Computing Systems, 45*(3), 470-485. doi: 10.1007/s00224-008-9103-4

Cordeau, Jean-François, & Laporte, Gilbert. (2007). The dial-a-ride problem: models and algorithms. *Annals of Operations Research, 153*(1), 29-46. doi: 10.1007/s10479-007-0170-8

Cordeau, Jean-François, Laporte, Gilbert, Savelsbergh, Martin W. P., & Vigo, Daniele. (2007). Chapter 6 Vehicle Routing. In B. Cynthia & L. Gilbert (Eds.), *Handbooks in Operations Research and Management Science* (Vol. Volume 14, pp. 367-428): Elsevier.

Cowling, Peter, & Johansson, Marcus. (2002). Using real time information for effective dynamic scheduling. *European Journal of Operational Research, 139*(2), 230-244. doi: http://dx.doi.org/10.1016/S0377-2217(01)00355-1

Daneshzand, F. (2011). The Vehicle-Routing Problem *Logistics Operations and Management* (pp. 127-153).

Davenport, Andrew J., Gefflot, Christophe, & Beck, J. Christopher. (2001). *Slack-based Techniques for Robust Schedules*. Paper presented at the European Workshop on Planning.

De Man, J. C. . (2014). Potential improvements in the planning and scheduling chain at KLM Aircraft Services.

De Snoo, C., Van Wezel, W., & Jorna, R. J. (2011). An empirical investigation of scheduling performance criteria. *Journal of Operations Management, 29*, 181-193. doi: 10.1016/j.jom.2010.12.006

EUROCONTROL. (2009). ATM Airport Performance (ATMAP) Framework. In R. c. b. t. P. R. Commission (Ed.).

Feuerstein, Esteban, & Stougie, Leen. (2001). On-line single-server dial-a-ride problems. *Theoretical Computer Science, 268*(1), 91-105. doi: http://dx.doi.org/10.1016/S0304-3975(00)00261-9

Hans, E.W., Houdenhoven, Mark van, & Hulshof, Peter J.H. (2011). A Framework for Health Care Planning and Control (pp. 23). Enschede: University of Twente.

Herroelen, Willy, & Leus, Roel. (2004). Robust and reactive project scheduling: a review and classification of procedures. *International Journal of Production Research, 42*(8), 1599-1620.

Ip, W. H., Wang, Dingwei, & Cho, Vincent. (2013). Aircraft ground service scheduling problems and their genetic algorithm with hybrid assignment and sequence encoding scheme. *Systems Journal, IEEE, 7*(4), 649-657.

Jaillet, Patrick, & Wagner, Michael R. (2006). Online routing problems: Value of advanced information as improved competitive ratios. *Transportation Science, 40*(2), 200-210.

Kuhn, Kenneth, & Loth, Steffen. (2009). Airport service vehicle scheduling.

Li, Zukui, & Ierapetritou, Marianthi. (2008). Process scheduling under uncertainty: Review and challenges. *Computers & Chemical Engineering, 32*(4–5), 715-727. doi: http://dx.doi.org/10.1016/j.compchemeng.2007.03.001

Lohman, Clemens, Fortuin, Leonard, & Wouters, Marc. (2004). Designing a performance measurement system: a case study. *European Journal of Operational Research, 156*(2), 267-286.

MacCarthy, B. L., Wilson, J. R., & Crawford, S. (2001). Human performance in industrial scheduling: A framework for understanding. *Human Factors and Ergonomics in Manufacturing & Service Industries, 11*(4), 299-320. doi: 10.1002/hfm.1016

McKay, K. N., Buzacott, J. A., Charness, N., & Safayeni, F. R. (1992). The scheduler's predictive expertise: an interdisciplinary perspective. *Artificial intelligence in operational research*, 139-150.

Mitchell, Ronald K, Agle, Bradley R, & Wood, Donna J. (1997). Toward a theory of stakeholder identification and salience: Defining the principle of who and what really counts. *Academy of management review, 22*(4), 853-886.

Neely, Andy, Gregory, Mike, & Platts, Ken. (2005). Performance measurement system design. *International Journal of Operations & Production Management, 25*(12), 1228-1263. doi: 10.1108/01443570510633639

Neely, Andy, Richards, Huw, Mills, John, Platss, Ken, & Bourne, Mike. (1997). Designing performance measures: a structured approach. *International Journal of Operations & Production Management, 17*(11), 1131-1152. doi: 10.1108/01443579710177888

Pillac, V., Gendreau, M., Guéret, C., & Medaglia, A. L. (2013). A review of dynamic vehicle routing problems. *European Journal of Operational Research, 225*, 1-11. doi: 10.1016/j.ejor.2012.08.015

Pistikopoulos, E. N. (1995). *UNCERTAINTY IN-PROCESS DESIGN AND OPERATIONS*. http://dx.doi.org/10.1016/0098-1354(95)00119-M

http://gateway.webofknowledge.com/gateway/Gateway.cgi?GWVersion=2&SrcApp=PARTNER_APP&SrcAuth=LinksAMR&KeyUT=WOS:A1995RM17900091&DestLinkType=FullRecord&DestApp=ALL_WOS&UsrCustomerID=1ba7043ffcc86c417c072aa74d649202

Sabuncuoglu, Ihsan, & Kizilisik, Omer Batuhan. (2003). Reactive scheduling in a dynamic and stochastic FMS environment. *International Journal of Production Research, 41*(17), 4211-4231.

Verderame, Peter M., Elia, Josephine A., Li, Jie, & Floudas, Christodoulos A. (2010). Planning and Scheduling under Uncertainty: A Review Across Multiple Sectors. *Industrial & Engineering Chemistry Research, 49*(9), 3993-4017. doi: 10.1021/ie902009k

Weber, Max. (1947). *The Theory of Social and Economic Organization*.

Appendix A Python data pre-processing code

```
__author__ = 'Feike Politiek'

from pandas import DataFrame, read_csv

import matplotlib.pyplot as plt
import pandas as pd
import matplotlib.dates as dates
import datetime
import numpy as np
import sys

print 'python version' + sys.version
print 'Pandas version' + pd.__version__


######################Read data files from folder

path1 = r'D:\Thesis Files\Data Analysis\Raw data\Shift_data'
path2 = r'D:\Thesis Files\Data Analysis\Raw data\Task_data'

import os

files_in_folder_1 = [os.path.join(path1, f) for f in os.listdir(path1) if
os.path.isfile(os.path.join(path1, f))]

files_in_folder_2 = [os.path.join(path2, f) for f in os.listdir(path2) if
os.path.isfile(os.path.join(path2, f))]

for file1, file2 in zip(files_in_folder_1, files_in_folder_2):
    with open(file1) as f1, open(file2) as f2:

        dftask = pd.read_csv(file2)
        dfresource = pd.read_csv(file1)

        CurrentDay = 2807                      # Hard coded current month and day

        dftask = dftask.rename(columns={'NU': 'CurrentDate', 'Earliest Start':
'EarliestStart', 'Start Time': 'StartTime',          # Rename columns
                                'End Time': 'EndTime', 'Latest End': 'LatestEnd',
'TAAK STATUS ': 'TaskStatus',
                                'Setup Duration1': 'SetupDuration1', 'Setup
Duration2': 'SetupDuration2',
                                'Delta (start - ES)': 'StartEarliestDifference',
'Delta (End - LE)':'LatestEndDifference'})

        dfresource = dfresource.rename(columns={'Name': 'ResourceName', 'Start
Time': 'ShiftStartTime', 'End Time': 'ShiftEndTime'
                               , 'Scheduled End': 'ShiftScheduledEnd', 'Scheduled
Start': 'ShiftScheduledStart'})


######################Data pre-processing

        dftask['CurrentDate'] = pd.to_datetime(pd.Series(dftask['CurrentDate']))
        dftask['EndTime'] = pd.to_datetime(pd.Series(dftask['EndTime']))
        dftask['EarliestStart'] =
pd.to_datetime(pd.Series(dftask['EarliestStart']))      # String to datetime
        dftask['StartTime'] = pd.to_datetime(pd.Series(dftask['StartTime']))
        dftask['LatestEnd'] = pd.to_datetime(pd.Series(dftask['LatestEnd']))
```

```
# String to datetime
        dftask['EndTime'] = pd.to_datetime(pd.Series(dftask['EndTime']))

        dftask['LatestEndDifference'] =
dftask['LatestEndDifference'].convert_objects(convert_numeric=True) #Convert object
to float64
        dftask['SetupDuration1'] =
dftask['SetupDuration1'].convert_objects(convert_numeric=True)
        dftask['SetupDuration2'] =
dftask['SetupDuration2'].convert_objects(convert_numeric=True)
        dftask['SetupDuration2'] = dftask['SetupDuration2'].abs()           #Turn
negative numbers into positive numbers in SetupDuration 2 Column

        dftask['Duration'] = dftask['Duration'].apply(lambda x:
float(str(x).replace(',', '.')) if ',' in str(x) else float(x))  # Remove , and
place dot .
        dftask['Duration'] = np.where(dftask['Duration'] -
np.floor(dftask['Duration']) > 0, dftask['Duration']*1000, dftask['Duration'])  #
Change the value of 1.8 to 1800 seconds
        dftask = dftask[dftask['DATE'] == CurrentDay]              # remove all
tasks that are smaller or larger than the current date

        dftask['DrivingStart1'] = dftask['StartTime'] -
pd.TimedeltaIndex(dftask['SetupDuration1'], unit='s')
        dftask['DrivingStart2'] = dftask['StartTime'] -
pd.TimedeltaIndex(dftask['SetupDuration2'], unit='s')

        dfresource['ShiftStartTime'] =
pd.to_datetime(pd.Series(dfresource['ShiftStartTime']))
        dfresource['ShiftEndTime'] =
pd.to_datetime(pd.Series(dfresource['ShiftEndTime']))
        dfresource['ShiftScheduledEnd'] =
pd.to_datetime(pd.Series(dfresource['ShiftScheduledEnd']))
        dfresource['ShiftScheduledStart'] =
pd.to_datetime(pd.Series(dfresource['ShiftScheduledStart']))

        dfresource = dfresource[(dfresource['ResourceName']) != 'Watertruck large
inh.']            #Remove rows that containt a specific value
        dfresource = dfresource[(dfresource['ResourceName']) != 'Toilettruck small
inh.']           #These rows are no resources, we are interested in the operators
        dfresource = dfresource[(dfresource['ResourceName']) != 'Bike 02']
        dfresource = dfresource[(dfresource['ResourceName']) != 'Bike 01']
        dfresource = dfresource[(dfresource['ResourceName']) != 'Watertruck drain
inh.']
        dfresource = dfresource[(dfresource['ResourceName']) != 'Watertruck small
inh.']
        dfresource = dfresource[(dfresource['ResourceName']) != 'Toilettruck large
inh.']
        dfresource = dfresource[~dfresource['ResourceName'].str.contains('\d')]
#Remove all resource names that contain numbers, in that case only operator names
are listed


################################# Write files to new csv file

        import os.path, time
        print "created: %s" % time.ctime(os.path.getctime(file1))

        created_at = int(os.path.getctime(file1))
        filename = datetime.datetime.fromtimestamp(created_at).strftime('%H%M')
        name, extension = os.path.splitext(file1)
        new_filename = filename + extension
```

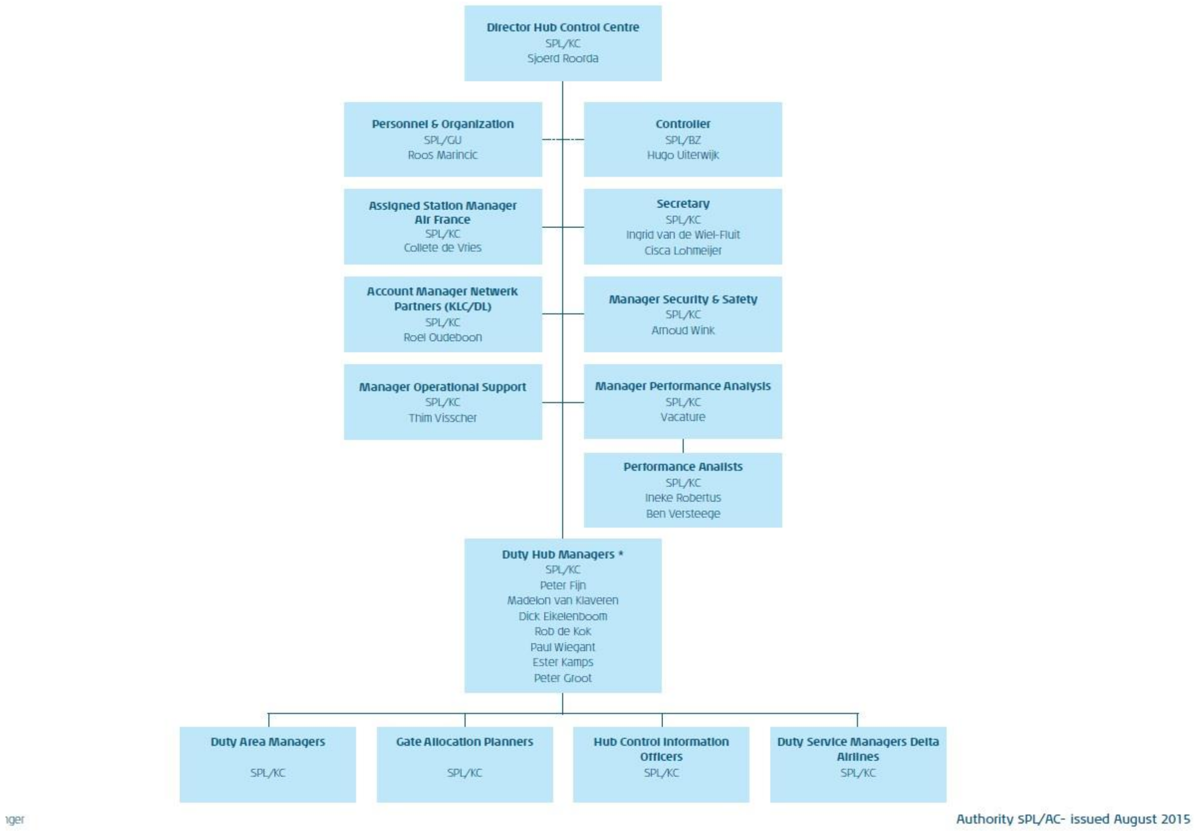
```
created_at1 = int(os.path.getctime(file2))
filename1 = datetime.datetime.fromtimestamp(created_at1).strftime('%H%M')
name1, extension1 = os.path.splitext(file2)
new_filename1 = filename1 + extension1

new_dir = 'D:\Thesis Files\Data Analysis\Processed data/Output_task'
new_dir1 = 'D:\Thesis Files\Data Analysis\Processed data/Output_shift'

print os.path.join(new_dir1, new_filename)
print os.path.join(new_dir, new_filename1)

dfresource.to_csv(os.path.join(new_dir1,new_filename))
dftask.to_csv(os.path.join(new_dir,new_filename1))
```

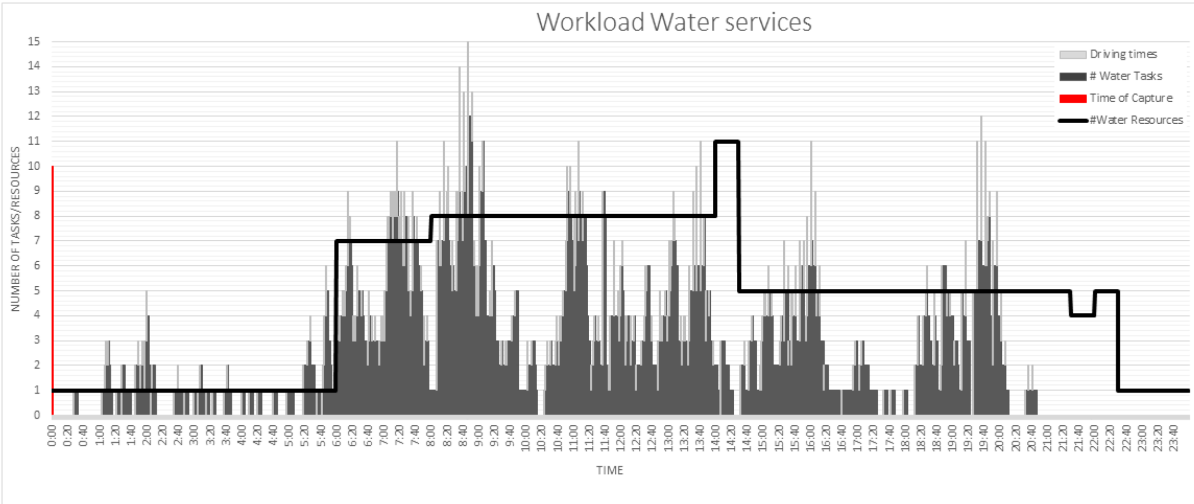
Appendix B HCC Organogram

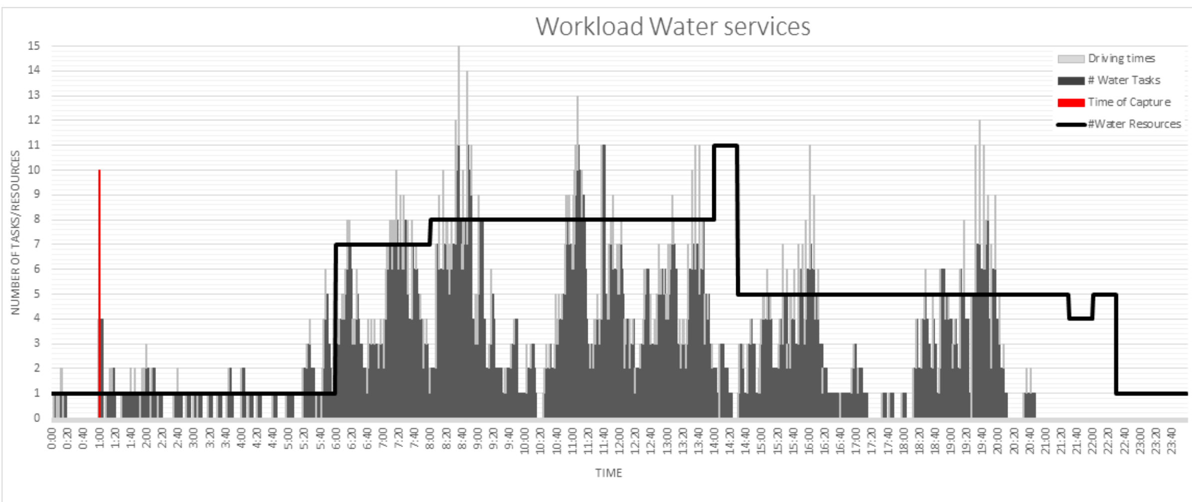


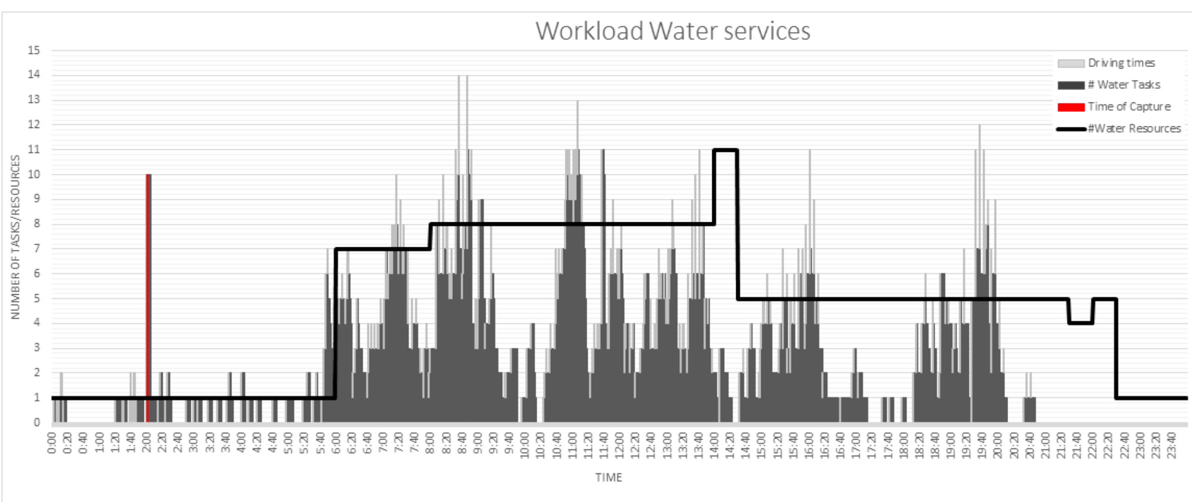
10

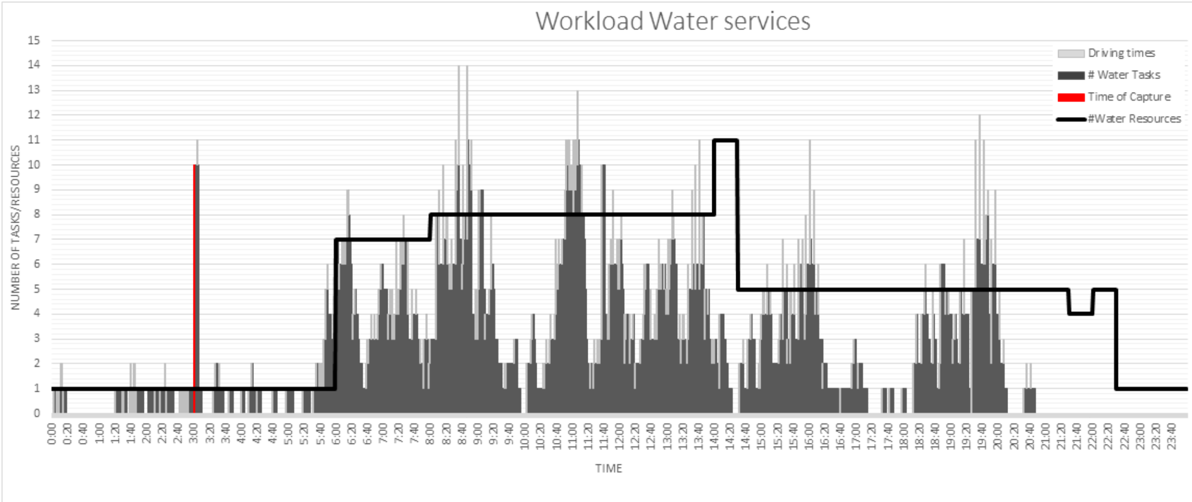
[10] Retrieved at 10 September 2015

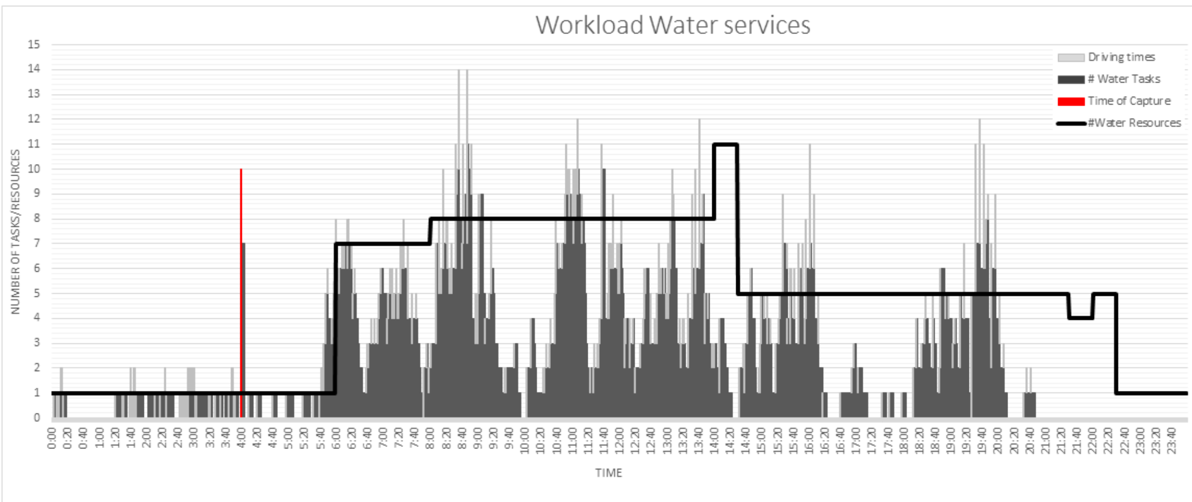
Appendix C Dynamic workload graphs

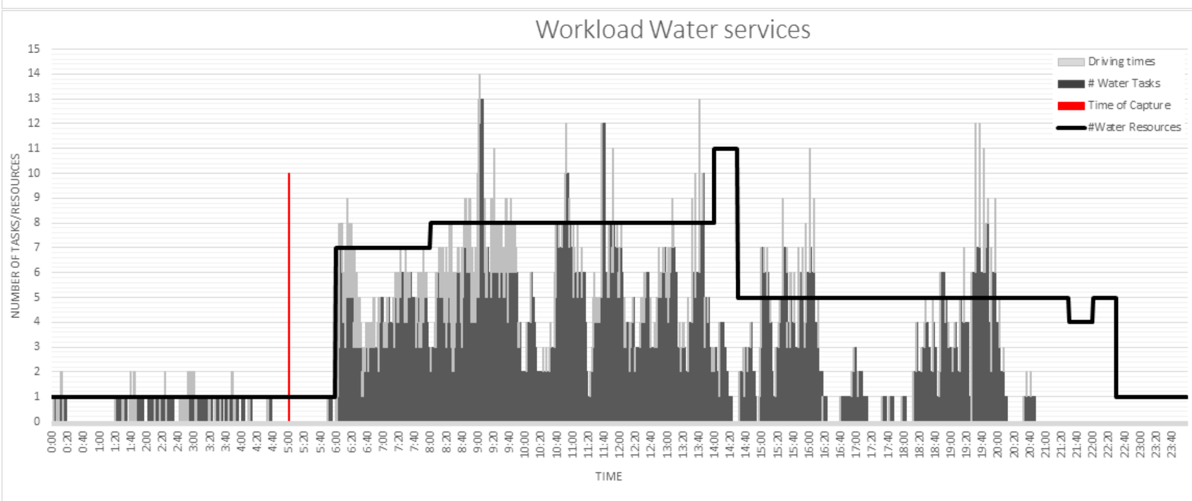


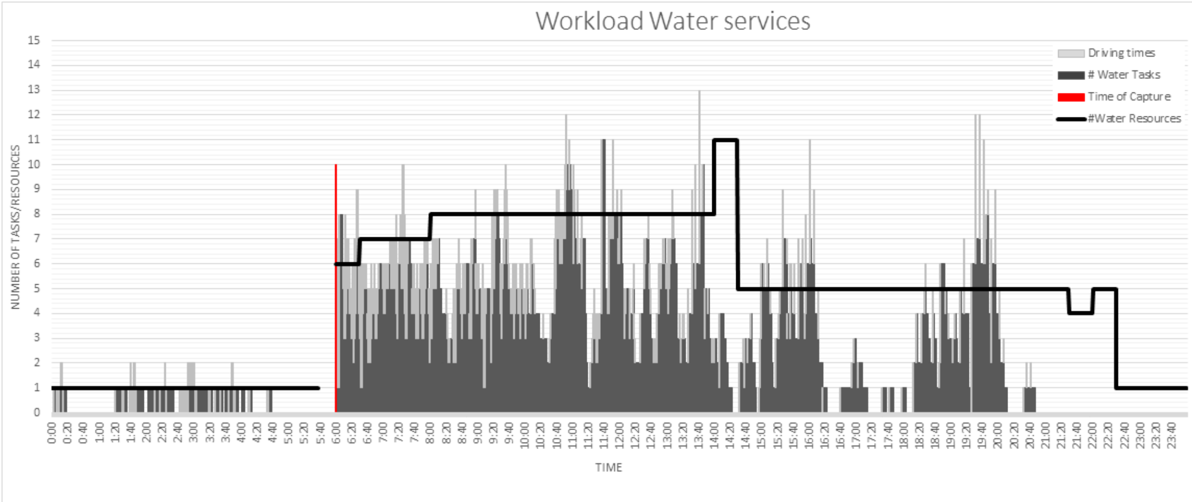


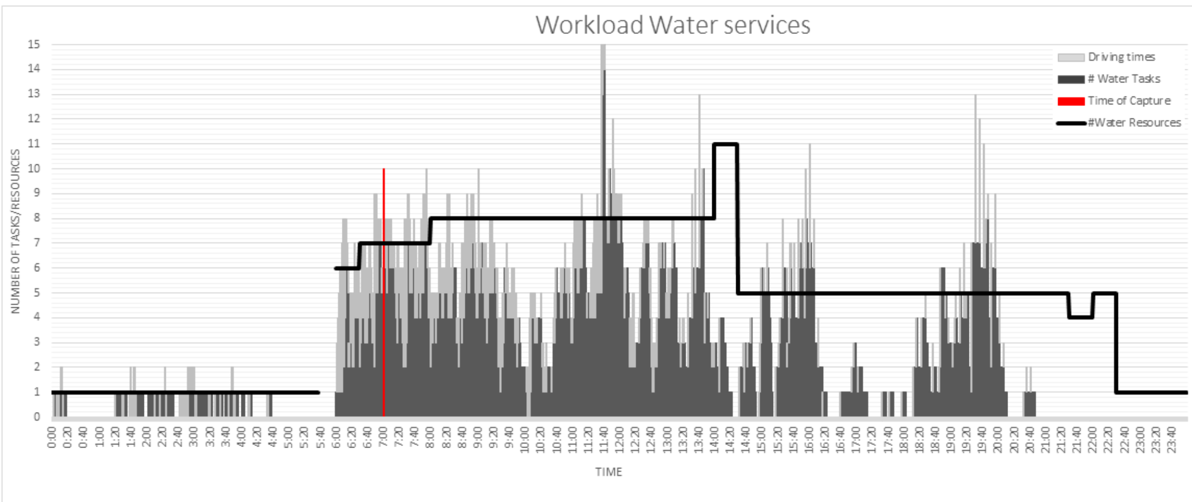


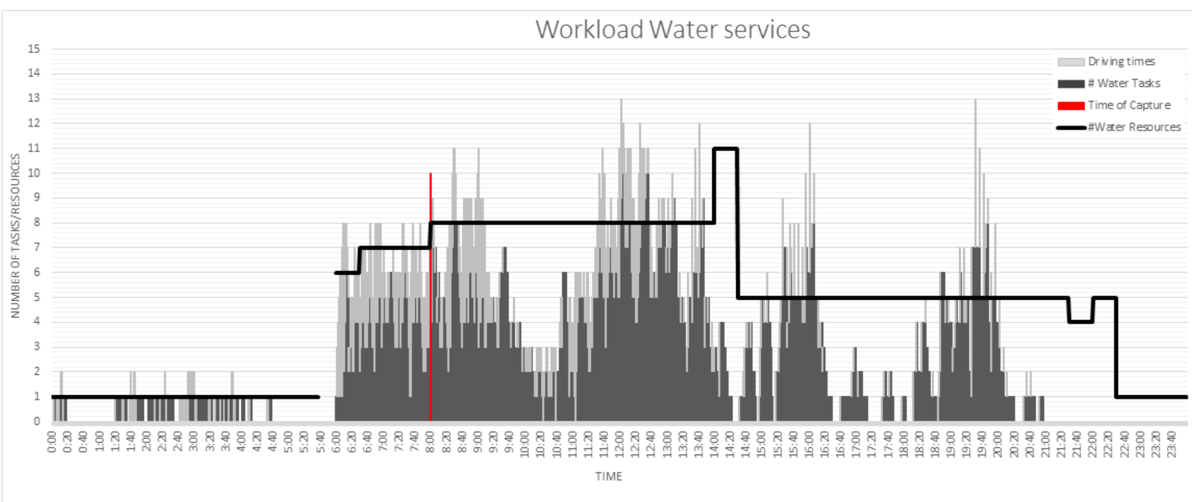

Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME

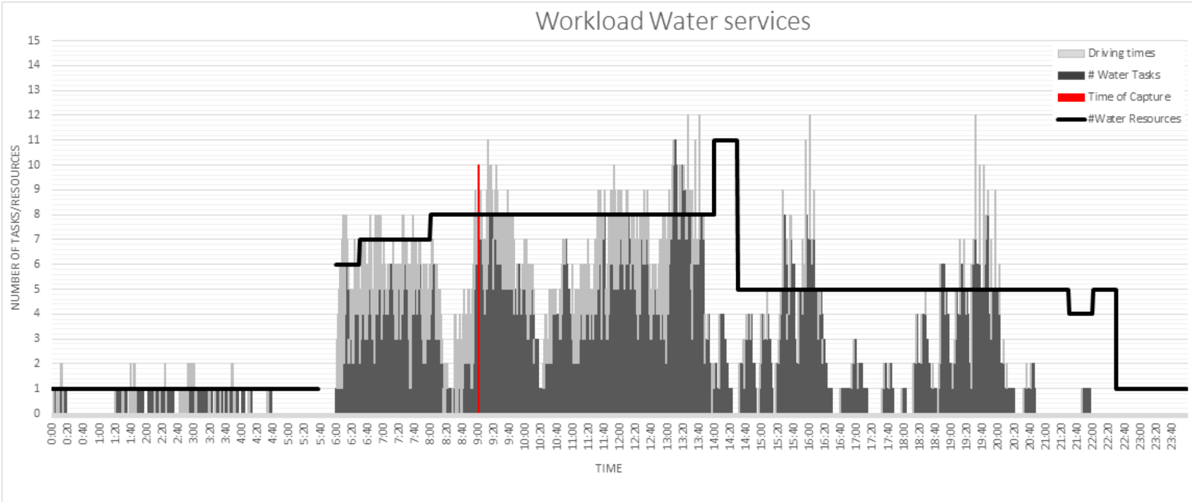

Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME

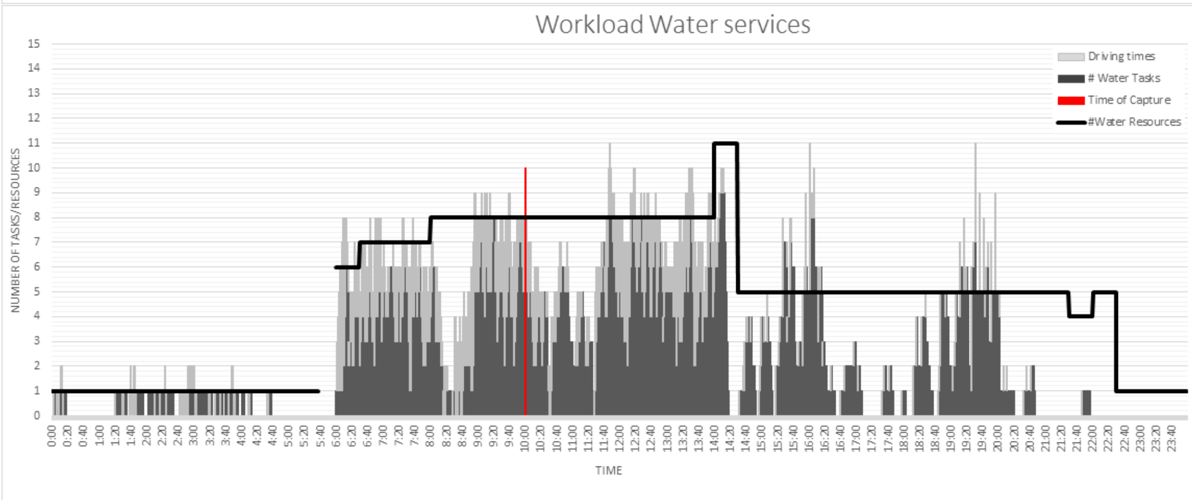

Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME

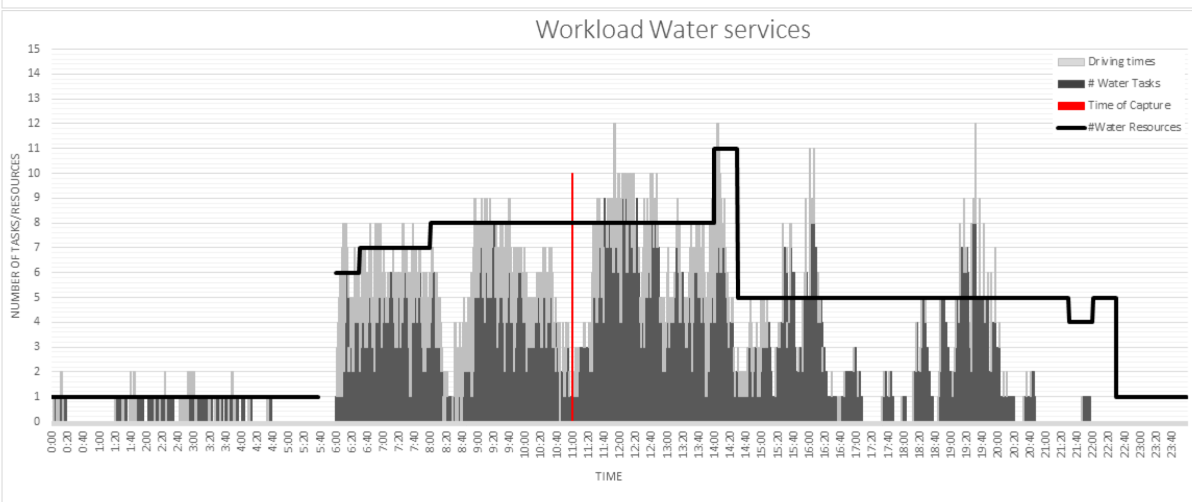

Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME

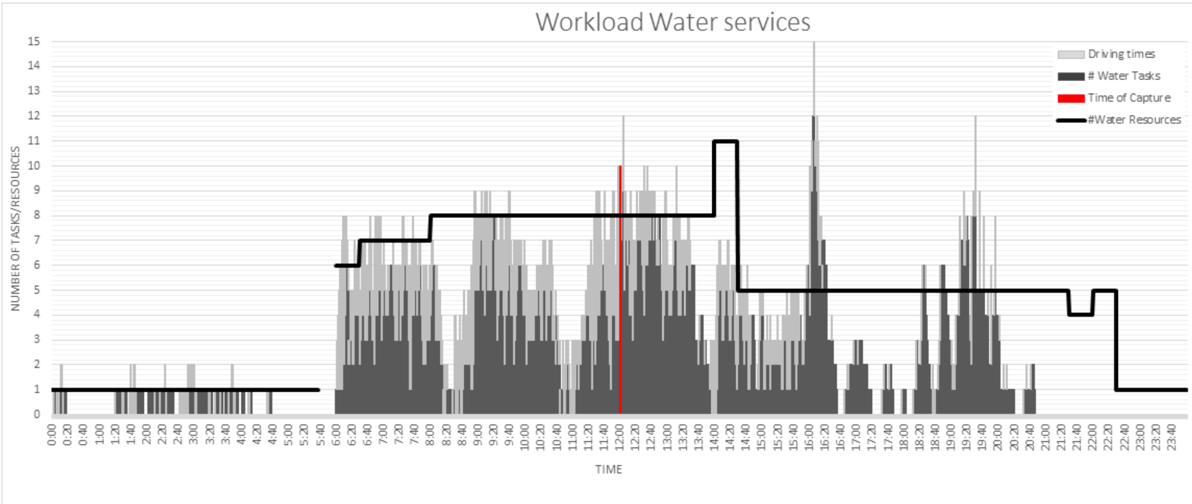

Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME

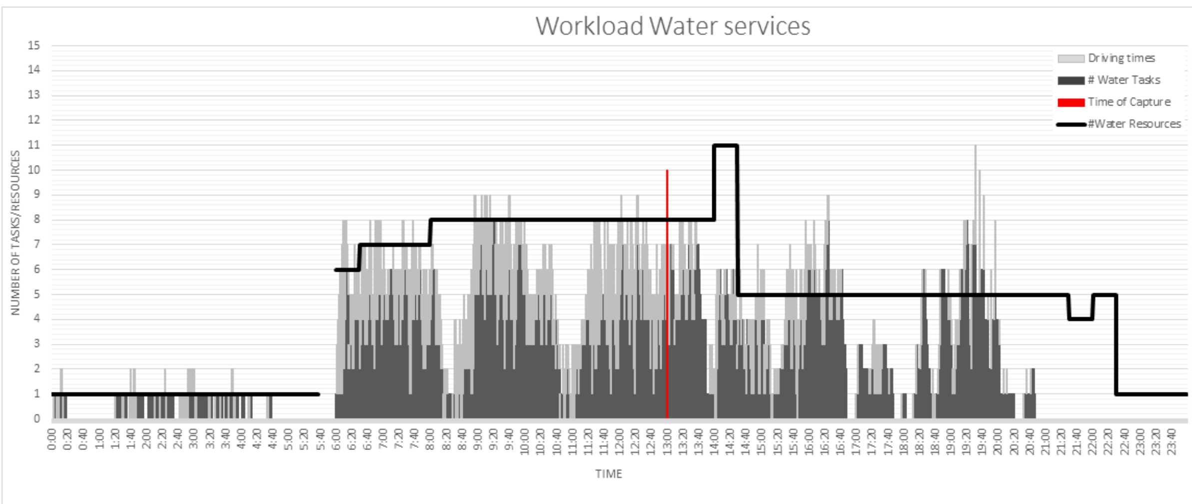

Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME

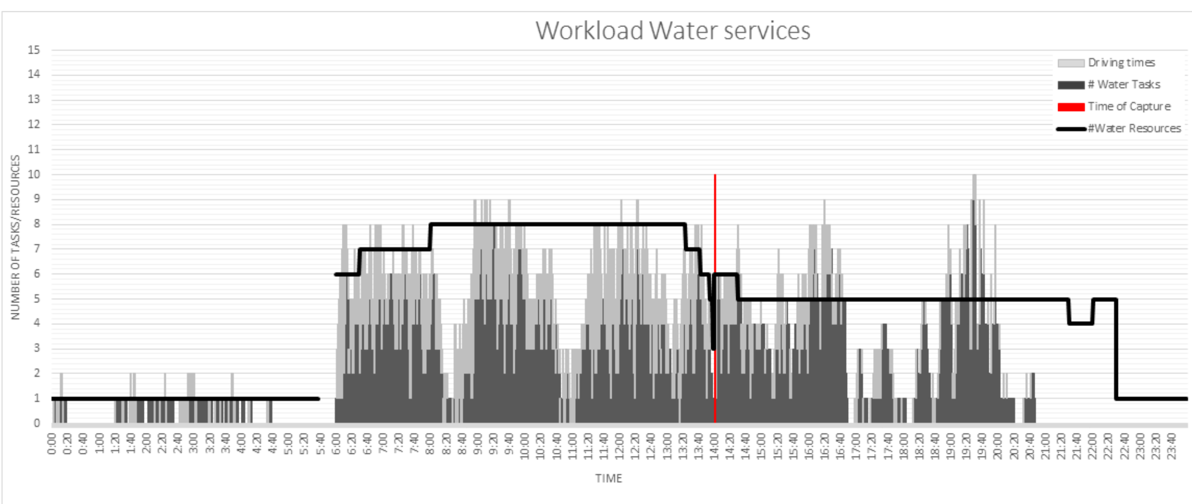

Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME

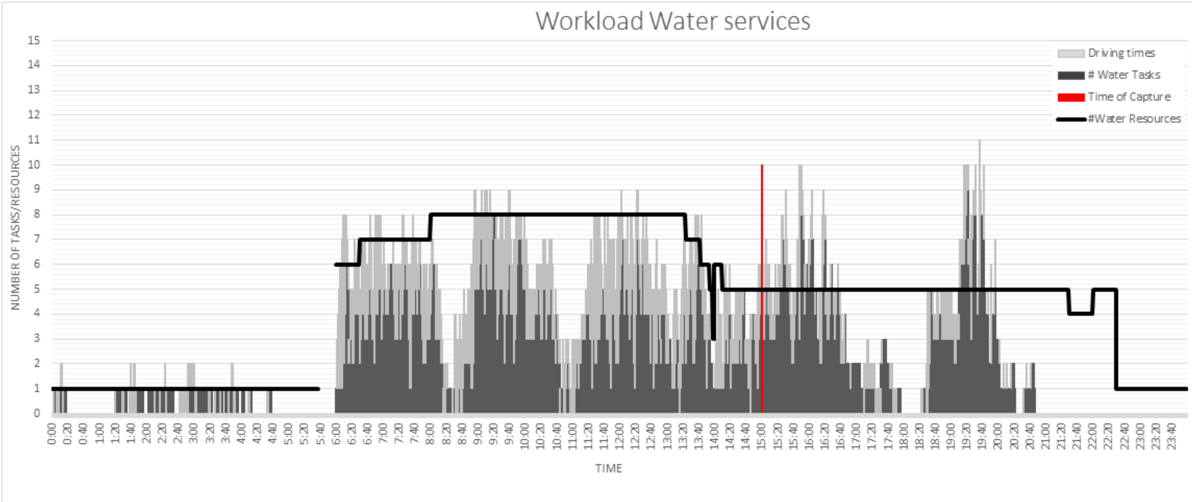

Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME

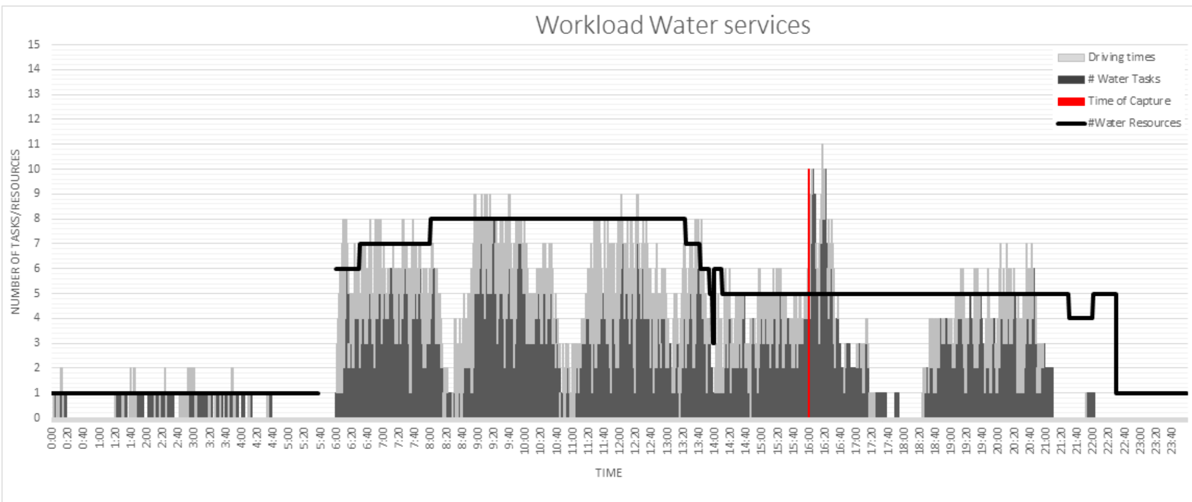

Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME

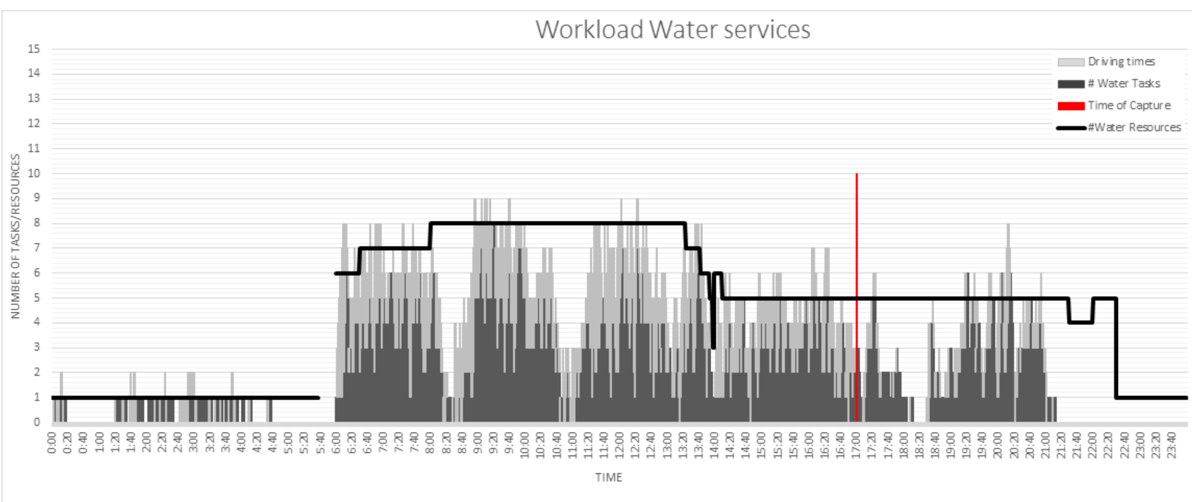

Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME

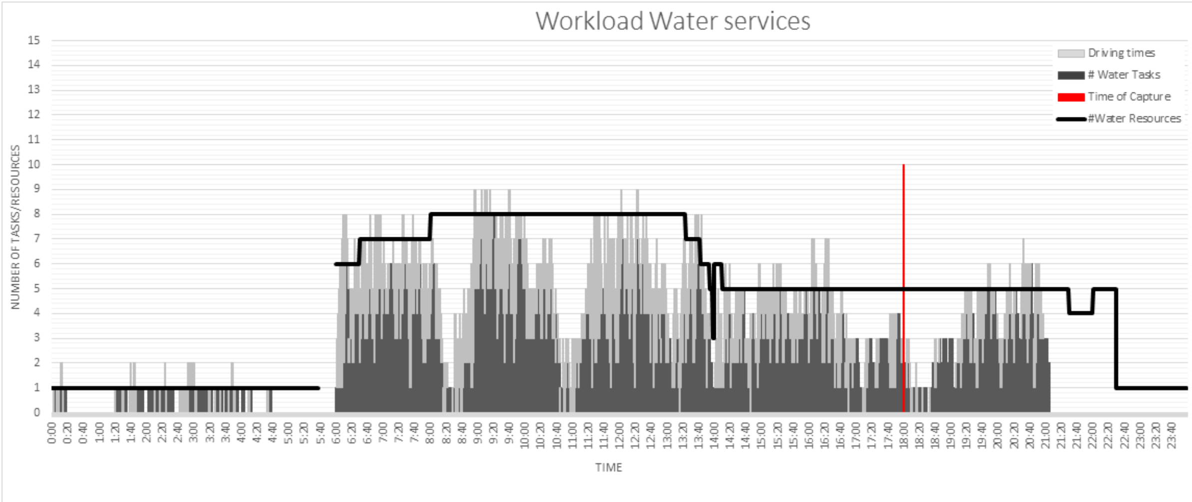

Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME

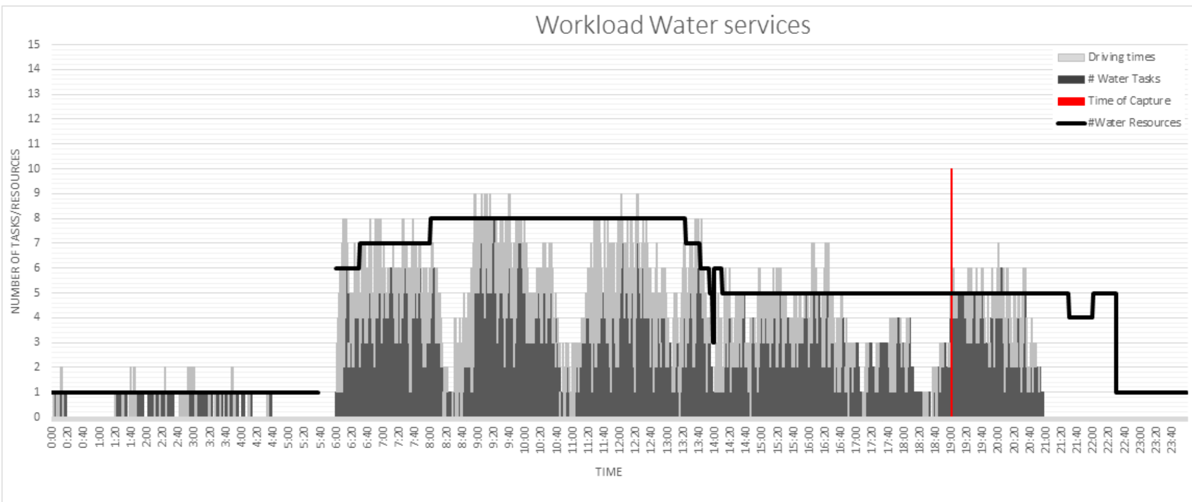

Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME

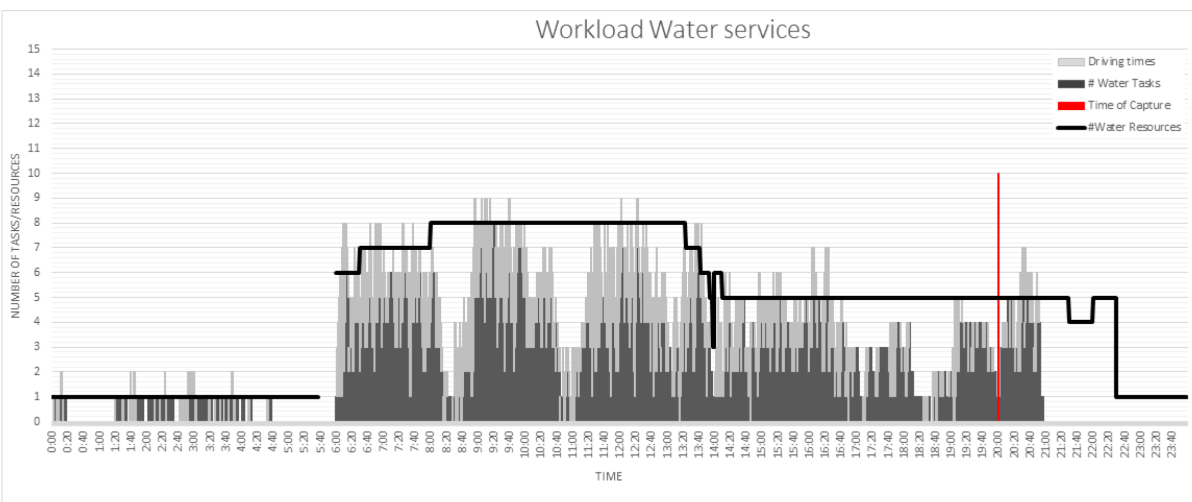

Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
0:00 0:20 0:40 1:00 1:20 1:40 2:00 2:20 2:40 3:00 3:20 3:40 4:00 4:20 4:40 5:00 5:20 5:40 6:00 6:20 6:40 7:00 7:20 7:40 8:00 8:20 8:40 9:00 9:20 9:40 10:00 10:20 10:40 11:00 11:20 11:40 12:00 12:20 12:40 13:00 13:20 13:40 14:00 14:20 14:40 15:00 15:20 15:40 16:00 16:20 16:40 17:00 17:20 17:40 18:00 18:20 18:40 19:00 19:20 19:40 20:00 20:20 20:40 21:00 21:20 21:40 22:00 22:20 22:40 23:00 23:20 23:40
TIME

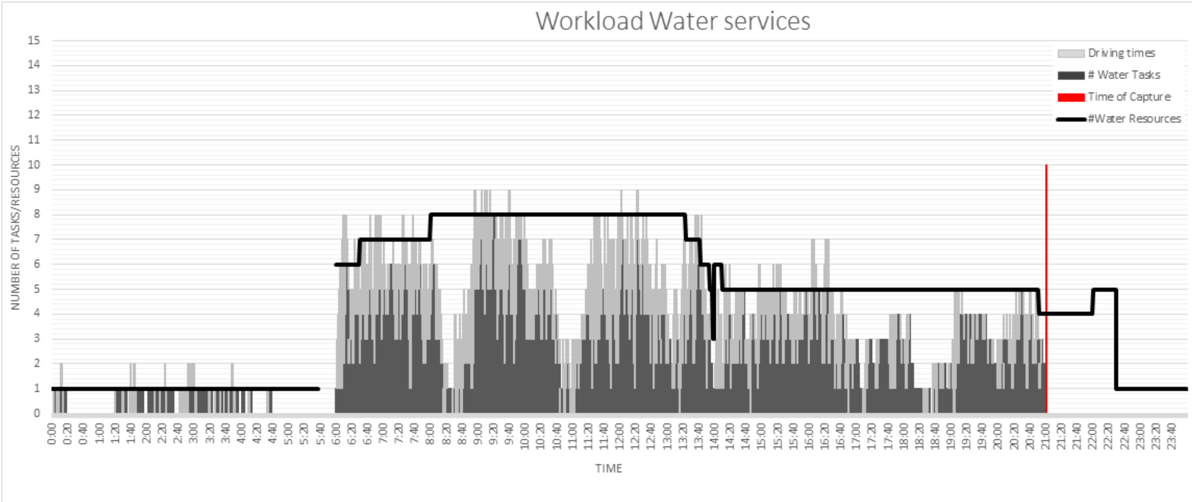

Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
0:00 0:20 0:40 1:00 1:20 1:40 2:00 2:20 2:40 3:00 3:20 3:40 4:00 4:20 4:40 5:00 5:20 5:40 6:00 6:20 6:40 7:00 7:20 7:40 8:00 8:20 8:40 9:00 9:20 9:40 10:00 10:20 10:40 11:00 11:20 11:40 12:00 12:20 12:40 13:00 13:20 13:40 14:00 14:20 14:40 15:00 15:20 15:40 16:00 16:20 16:40 17:00 17:20 17:40 18:00 18:20 18:40 19:00 19:20 19:40 20:00 20:20 20:40 21:00 21:20 21:40 22:00 22:20 22:40 23:00 23:20 23:40
TIME

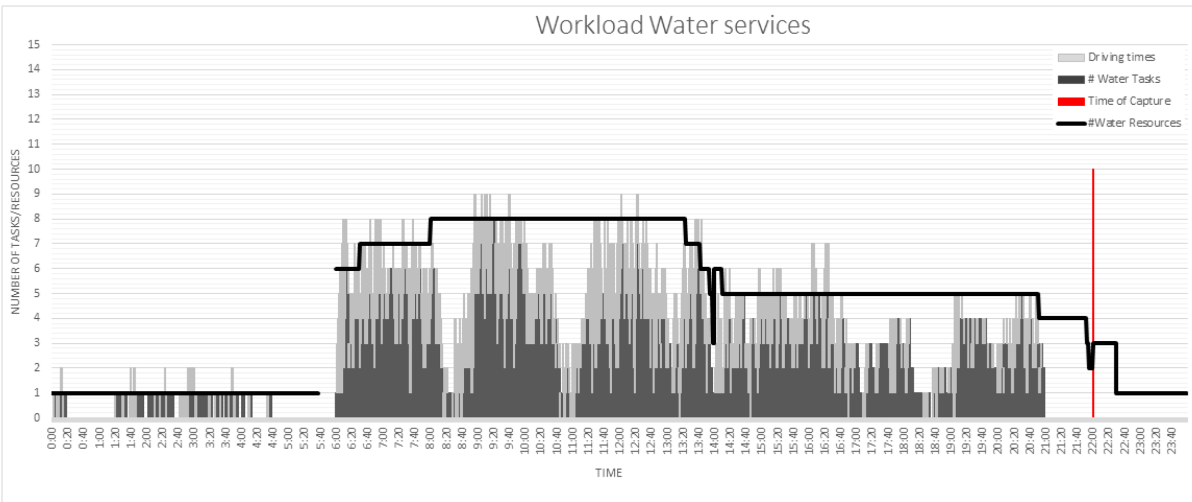

Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
0:00 0:20 0:40 1:00 1:20 1:40 2:00 2:20 2:40 3:00 3:20 3:40 4:00 4:20 4:40 5:00 5:20 5:40 6:00 6:20 6:40 7:00 7:20 7:40 8:00 8:20 8:40 9:00 9:20 9:40 10:00 10:20 10:40 11:00 11:20 11:40 12:00 12:20 12:40 13:00 13:20 13:40 14:00 14:20 14:40 15:00 15:20 15:40 16:00 16:20 16:40 17:00 17:20 17:40 18:00 18:20 18:40 19:00 19:20 19:40 20:00 20:20 20:40 21:00 21:20 21:40 22:00 22:20 22:40 23:00 23:20 23:40
TIME

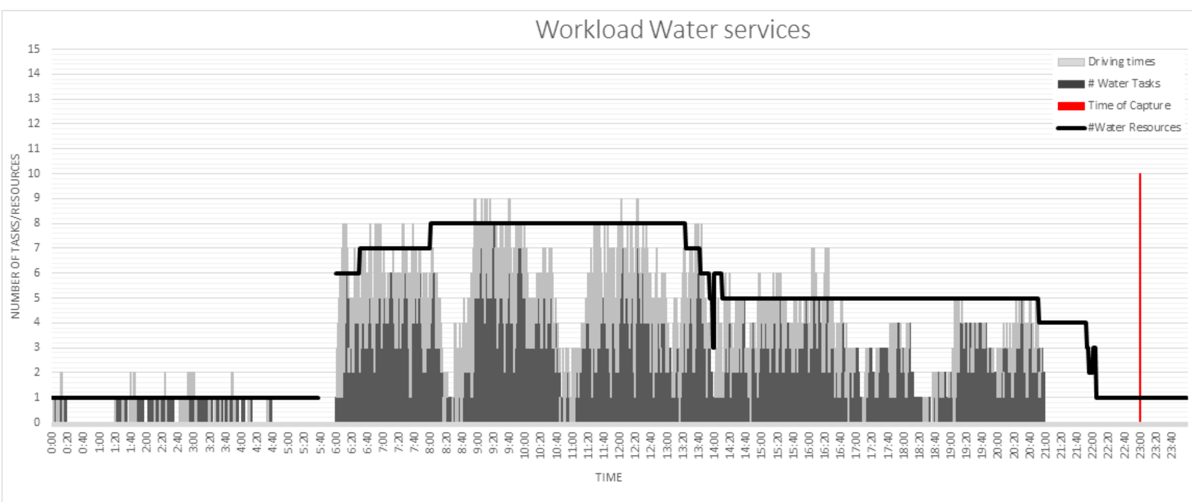

Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME


Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME


Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME


Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME


Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME


Workload Water services
Driving times
Water Tasks
Time of Capture
#Water Resources
NUMBER OF TASKS/RESOURCES
TIME