This project has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no 609611.

D24.2

Business Modelling

Project number:	609611
Project acronym:	PRACTICE
Project title:	PRACTICE: Privacy-Preserving Computation in the Cloud
Start date of the project:	1st November, 2013
Duration:	36 months
Programme:	FP7/2007-2013

Deliverable type:	Report
Deliverable reference number:	ICT-609611 / D24.2/ FINAL | 1.0
Activity and Work package contributing to the deliverable:	Activity 2 / WP 24
Due date:	April 2015 – M18
Actual submission date:	30th April 2015

Responsible organisation:	DTA/CCII-UNISA
Editor:	Antonio Zilli
Dissemination level:	PU
Revision:	Final version 1.0

Abstract:	New business models for MRO segment of the aeronautic industry and for the consumer goods industry. Mathematical models and algorithm for collaborative forecasting and planning in the aeronautic MRO process and for implementing Vendor managed inventory business model. Process-related data disclosures events and impact evaluation models.
Keywords:	Aeronautic Fleet management, MRO process, vendor managed inventory, secure algorithms.

Editor

Antonio Zilli - UNISA/CCII (third party of DTA)

Contributors (ordered according to beneficiary numbers)

Contributors	Sections (sub-sections are included)
TEC - Mario Münzer	Section 2.4.2 – Data model for the aeronautic case
SAP - Florian Hahn	Section 3.1.2.2.3 – Data sensitivity assessment Section 3.2.2.1 – Secure aggregation protocol Chapter 5 – PRACTICE architecture and tools
ARC - Cem Kazan, Elif Ozdogan, Buket Serper	2.1.2 New business models to supply new products in the consumer goods 2.3 Consumer goods industry
UWUERZ - Richard Pibernik, Julian Kurz, Fabian Taigel, Jan Meller, Fabian Diehm	Chapter 3 – Models and algorithms for cloud-based supply chain collaboration, Section 3.1 – Secure cloud-based collaborative maintenance management in the aeronautic industry, Section 3.2 – Secure cloud-based Vendor Managed Inventory in the consumer goods industry, Section 3.3 – Conclusion and outlook
UNISA/CCII (third party of DTA) - Antonio Zilli, Marianna Lezzi	Chapter 1 – Introduction, Chapter 2 – Industrial process model Section 2.1 – Business modelling, Section 2.2 – Aerofleet management, Section 2.4.1 – Functional and security requirements for the aeronautic case, Section 2.4.2 – Data model for the aeronautic case Chapter 6 – Conclusions
UMIL - Stelvio Cimato, Ernesto Damiani	Chapter 4 – Risk assessment in supply chains
PAR - Kurt Nielsen	Section 3.3 – Auctions and supply chain coordination

Disclaimer

The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 609611 (PRACTICE).

Executive Summary

This report focuses the changes in supply chain business models enabled by the development of closer and closer relationships among partners. The aeronautic and consumer goods supply chain are explored, forecasting and planning models are defined, and implementation approach with respect to other PRACTICE WPs are considered.

The fleet management process (that is the aero-engine after sale service) is designed with a supply chain perspective. Bottlenecks of the current situation are highlighted and data available in the supply chain, that can be leveraged to improve process performance, are identified. The analysis is aimed at showing business risks related to the management of confidential data on a collaborative cloud system, and at preparing the data model for a collaborative service forecasting and planning application.

The consumer goods industry is analysed by a leading Turkish actor (ARCELIK). The target of the analysis is the identification of the issues facing the current situation and the improvements bought by a business model between retailers and the producer based on 'vendor managed inventory' approach. In particular, the more detailed topics are the business innovations required to improve the accuracy of forecasts and a more effective use of the resources available in the supply chain.

The analysis of the industrial cases provides inputs for defining the most suitable forecasting and planning models. Current standard models were updated and tailored to the two cases, moreover they were extended to satisfy requests from the industries. The main motivation to update current models is to leverage private data available in the supply chain.

The mathematical and computing models were also analysed to verify their implementation under the PRACTICE framework.

The main consideration about the analysis of the two industrial supply chain cases and the models developed are the following ones:

1. In order to create benefits, new business models (the fleet management and the Vendor Managed Inventory) need supply chain partners to provide private data to compute jointly highly industry-specific algorithms; on the other hand, bringing private data introduces critical business risks;
2. A collaborative service demand forecasting model was developed to enable fleet management, it is based on the frequent analysis of data reporting actual usage of engines;
3. The collaborative service demand forecasting model provides inputs to improve inventory management in the maintenance (MRO) base;
4. An optimized engine servicing schedule can be calculated integrating the servicing demand forecasting, MRO resource management data and penalties due to unsatisfied service level agreement;
5. Service demand forecasts and resource management information are used also to plan capacity usage and needs in the MRO base;
6. In the case of the consumer goods industry, a secure demand aggregation protocol integrating private data from different clients was developed, it computes demand for specific products by aggregating data from many competing customers;
7. Also a secure model for the Vendor Managed Inventory approach was developed, that is based on the availability of inventory and costs data from resellers and of production and inventory data from production centers;
8. A method to compare data leakage risks with secure cloud systems implementation costs is developed. The method is based on designing the flow of encrypted parts of

data, from which the possibility to reconstruct a complete confidential data on a cloud node can be identified. This event is related to the economic costs of releasing the specific data to other partners of the supply chain.

The main conclusions of the work presented in the report are:

- Business risks associated to data leakage events depend on the couple *data owner-recipient* involved in the event, and are based on the capabilities of one of them to shrink its own profits or to increase costs of the other;
- models and algorithms developed for the aeronautic case result in a complete and innovative collaborative planning system providing unparalleled services to a supply chain; preliminary simulations show that a cost reduction of about 20% can be achieved for the MRO service;
- significant improvements in the resource management can be obtained in the consumer goods case, in particular a more efficient production quantity assigned to suppliers and reduced inventory level in the product assembly side and in the retailer side;
- The method for comparing cloud systems protocols with business risks was tested on the fleet management case and showed that it is able to provide valuable information for selecting the most effective secure implementation with respect to industry features.

Results presented in this report encourage the development of a secure cloud forecasting and planning system. The designed system enables innovative processes for supply chains providing economic benefits to involved partners. Such new processes are necessary to improve the competitiveness of a supply chain considered as a unique economic system.

In the next project period, specific industrial test cases as well as metrics for the prototype assessment will be prepared. At the same time, the implementation of the system will start.

Contents

Chapter 1 Introduction 11

Chapter 2 Industrial process models 15

2.1 Business modelling 15

2.1.1 Business innovation in the aeronautic industry 15

2.1.2 New business models to supply new products in the consumer goods 16

2.2 Aerofleet management 18

2.2.1 Description and design of the Aerofleet management process 19

2.2.2 Confidential data managed in the process 21

2.2.3 Risks of data leakage 22

2.2.4 Trust relationship in the supply chain 24

2.3 Consumer goods industry 25

2.3.1 Description of Consumer Goods Planning & Production Systems 25

2.3.2 Description of Consumer Goods Management Troubles 28

2.3.3 Proposed System and VMI Approach 35

2.4 Summary of functional and security requirements 36

2.4.1 Functional and security requirements for the aeronautic case 36

2.4.2 Data model for the aeronautic case 37

2.4.2.1 Fleet monitoring 37

2.4.2.2 MRO service planning 40

2.4.2.3 Inventory management 42

Chapter 3 Models and algorithms for cloud-based supply chain collaboration 45

3.1 Secure cloud-based collaborative maintenance management in the aeronautic industry 45

3.1.1 Overview and introduction 45

3.1.2 Collaborative maintenance demand forecasting 47

3.1.2.1 Supply chain collaboration in the context of machine learning 48

3.1.2.2 Demand forecasting without condition data 51

3.1.2.2.1 Aggregated service demand forecasting 51

3.1.2.2.2 Implementation 55

3.1.2.2.3 Data sensitivity assessment 56

3.1.2.2.4 Conclusion and further references 58

3.1.2.3 Integrating condition monitoring data 58

3.1.2.3.1 Repair or replace: Forecasting with probabilistic decision trees 60

3.1.2.3.2 Implementation 63

3.1.2.3.3 Data sensitivity assessment 67

3.1.2.3.4 Conclusion and further references 69
3.1.2.4 Secure condition based preventive maintenance 69
3.1.2.4.1 Forecasting maintenance demand with ordinal decision trees 70
3.1.2.4.2 Implementation 71
3.1.2.4.3 Data sensitivity assessment 72
3.1.2.4.4 Conclusion and further references 73
3.1.3 Collaborative maintenance planning 74
3.1.3.1 Spare parts management 74
3.1.3.1.1 A dynamic (r,Q) inventory control policy 74
3.1.3.1.2 Implementation 76
3.1.3.1.3 Data sensitivity assessment 78
3.1.3.1.4 Adjustments and potential extensions of the model 79
3.1.3.1.5 Conclusion and further references 80
3.1.3.2 Maintenance arrival slot scheduling 81
3.1.3.2.1 Current engine arrival characteristics 82
3.1.3.2.2 Cloud-based collaborative scheduling of engine arrival slots 83
3.1.3.2.3 Implementation of scheduling algorithm 89
3.1.3.2.4 Data sensitivity assessment 90
3.1.3.2.5 Conclusion and further references 90
3.1.3.3 Capacity planning 91
3.1.3.3.1 Optimization model definition 92
3.1.3.3.2 Sojourn time approximation for queueing networks 94
3.1.3.3.3 Queueing network parameter computation 96
3.1.3.3.4 Solution procedure 98
3.1.3.3.5 Effects of advanced information and collaboration 103
3.1.3.3.6 Numerical example 108
3.1.3.3.7 Implementation 111
3.1.3.3.8 Data sensitivity assessment 112
3.1.3.3.9 Conclusion and further references 113
3.1.4 Summary 113
3.2 Secure cloud-based Vendor Managed Inventory in the consumer goods industry 115
3.2.1 Overview and introduction 115
3.2.2 Collaborative forecasting for mid-term production planning 117
3.2.2.1 Secure aggregation protocol 117
3.2.2.2 Implementation 118
3.2.2.3 Data sensitivity assessment 119
3.2.2.4 Conclusion and possible extensions 119
3.2.3 Collaborative planning – secure Vendor Managed Inventory 120
3.2.3.1 Notation 121
3.2.3.2 Assumptions 122
3.2.3.3 Model development 123
3.2.3.3.1 Total Vendor Cost (TC_V) 124
3.2.3.3.2 Total Retailers Cost (TC_R) 126
3.2.3.3.3 Total System Cost (TC) 126

3.2.3.4 Model ... 127
3.2.3.5 Solution algorithm ... 129
3.2.3.6 Implementation ... 130
3.2.3.7 Data sensitivity assessment ... 131
3.2.3.8 Conclusion and further references ... 133
3.2.4 Summary ... 134
3.3 Auctions and supply chain coordination ... 135
3.3.1 Auctions in theory and practice ... 136
3.3.2 Truth telling and Secure Multiparty Computation ... 137
3.3.3 Coordination of price and quantity ... 138
3.3.4 Coordination of prices and other attributes ... 140
3.3.5 On the use of auctions in PRACTICE ... 141
3.4 Conclusions and outlook ... 142
Chapter 4 Risk assessment in supply chains ... 143
4.1 Introduction ... 143
4.2 Risk in supply chains: general notions ... 143
4.3 Rationale of the presented approach ... 145
4.3.1 The overall methodology ... 146
4.4 Risk Assessment case study ... 147
4.5 Computing the timeline scheduling ... 148
Chapter 5 PRACTICE architectures and tools ... 154
Chapter 6 Conclusions ... 156
Chapter 7 Bibliography ... 158

List of Figures

Figure 1: Position of this deliverable with respect to the whole PRACTICE project. 14
Figure 2: Complex Network Model of ARC 17
Figure 3: Aero fleet management process. Source: Author's illustration. 20
Figure 4: Modular architecture of a conventional twin-spool turbofan engine. Source: S. Ackert, 2011 21
Figure 5: Data categories into CPS database. Source: Author's illustration 21
Figure 6: Customer Level Process in ARC 26
Figure 7: Supplier and Inventory Level Process in ARC. 27
Figure 8: Forecast accuracy of customers. 30
Figure 9: Relational data model. 44
Figure 10: Collaborative planning system for aircraft engine maintenance. 46
Figure 11: Focus on collaborative maintenance demand forecasting....................................47
Figure 12: Collaborative forecasting for maintenance management. 48
Figure 13 Representation of a partitioned attribute-space as decision tree...........................50
Figure 14: Linear regression model for the #cycles subject to time.......................................53
Figure 15: Demand forecasting with usage data only 55
Figure 16: Example of a binary decision tree of depth 3 62
Figure 17: Secure collaborative forecasting process for repair/replace demand 63
Figure 18: Example of a demand distribution...66
Figure 19: Time structure of the planning horizon H. 67
Figure 20: Deriving binary classification tasks 71
Figure 21: Secure collaborative forecasting process for overhaul demand using condition data 72
Figure 22: Intermittent demand pattern for spare parts 75
Figure 23: Dynamic (r_t,Q) inventory policy [illustration according to Babaï et al. (2009)] 76
Figure 24: Process overview on the Spare Parts Management Module 78
Figure 25: CPS for collaborative maintenance management 81
Figure 26: Illustration of engine arrival slot scheduling...82
Figure 27: Probability of the scheduled arrival slot to coincide with the optimal arrival slot ($\lambda e = 5$)...84
Figure 28: Cumulative distribution function for different time window widths. 85
Figure 29: Probability density function for different time window widths. 87
Figure 30: Updated interarrival time variability in dependence of relative window width........89
Figure 31: Engine Arrival Slot Scheduling module. 90
Figure 32: MRO production network. 92

Figure 33: Schematic production network. 94
Figure 34: One-dimensional illustration of total cost function (—), composed of capacity costs (···) and penalty costs (· – ·). 99
Figure 35: Capacity planning module. 112
Figure 36: Overview collaborative maintenance management implementation. 114
Figure 37: ARC supply chain 115
Figure 38: Forecasting and planning processes in the consumer goods case 116
Figure 40: Secure aggregation protocol 118
Figure 39 Basic concept of secure forecast aggregation 118
Figure 41: Model notation for VMI in the ARC supply chain. 122
Figure 42: Inventory level for vendor (A) and m = 3 retailers (B, C and D). 123
Figure 43: Inventory level over time for vendor 125
Figure 44: Overstock for retailer j 125
Figure 45: Implementation process 131
Figure 46: Secure aggregation protocol. 134
Figure 47: Computational process for one product in VMI setting 135
Figure 48 View of the overhaul management process to compute the timeline scheduling 147
Figure 49 The scheduling maintenance process (two airlines) represented by the Bogdanov model 148
Figure 50 The scheduling maintenance process (two airlines) 150
Figure 51 The secured scheduling maintenance process 151
Figure 52 The secured scheduling maintenance process (two airlines) 152
Figure 53: One possible implementation for some models presented in Chapter 3. 155

List of Tables

Table 1. Input data, business functions of the system and output results. Source: Author's illustration. 22
Table 2. Risks for the actors of the aero engine overhaul supply chain. Source: Author's illustration. 24
Table 3: Customer Demand Parameters 29
Table 4: February Actual sales and Forecast Figures. 30
Table 5: February Forecast Accuracy Ratio 30
Table 6: Material based Availability Schedule 31
Table 7: Product Life Cycle Status. 32
Table 8: Customer Fulfillment 33
Table 9: Inventory Management Activities 33
Table 10: Current Scores of Deviation Stock Ratio 34
Table 11. Engine parameters 39
Table 12. Module parameters 40
Table 13. Customers 40
Table 14. Service status 41
Table 15. Overhaul activities 41
Table 16. Human resources 42
Table 17. Inventory status 43
Table 18. Supplier 43
Table 19: Usage parameters for one engine of one customer; all values are accumulations since the last overhaul 52
Table 20: Thresholds for an engine type e 52
Table 21: Output of linear regressions 53
Table 22: Number of engines expected for service for one customer a 55
Table 23: Data sensitivity assessment without sensor data. 57
Table 24: Structure of a learning dataset D 59
Table 25: Notation used in this chapter 61
Table 26: Data sensitivity assessment without sensor data 67
Table 27: Sample for an encrypted unclassified dataset $Sencr$ 68
Table 28: Differing security requirements for distributed private learning datasets 73
Table 29: Interfaces of the Spare Parts Management Module 77
Table 30: Criticality assessment of the parameters within the Spare Parts Management Module 79
Table 31: Data sensitivity assessment for collaborative engine arrival slot scheduling. 90
Table 32: Numerical example queueing network parameters. 109
Table 33: Numerical test of benefits through collaboration. 110

Table 34: Data sensitivity assessment for capacity planning. 113
Table 35: Sensitivity analysis of vendor inputs..131
Table 36: Sensitivity analysis of retailers input..132
Table 37: Sensitivity analysis of variables calculated through inputs. 132
Table 38: Main features of secure aggregation and sVMI 134

Chapter 1 Introduction

Competitiveness of companies realizing complex products strongly depends on the capability to create valuable and effective collaborations with customers and suppliers, and to manage processes through and in all layers of their supply chain. Indeed, in the last years the collaboration among supply chain partners has been pushed in every phase of the product development process, from product conceptualization to after-sale services provision (Petersen et al., 2005; Flynn et al., 2010; Li et al., 2006).

Moving toward a *supply chain-based perspective of the organization* means updating internal processes of a firm in order to align them with current (and changing) customers' business needs, as well as with the production processes of suppliers. Aligning activities of firms with the business needs of customers is more and more strategic the higher the volatility of market and the capital intensity of the involved products get (Ramanathan & Gunasekaran, 2014).

This perspective change involves radical innovations in the business models deployed by companies and in the IT infrastructures and ICT systems available to the supply chain participants (Nyaga et al., 2010). In particular, business model innovations are focused on identifying and exploiting the actual sources of value in client-supplier relationships. For example, information belonging to customers that have high impact on the effectiveness and quality of suppliers' processes (Prajogo & Olhager, 2012; Koçoğlu et al., 2011; Jammernegg & Reiner, 2007; Rai et al., 2006). IT and ICT innovations are required in order to ensure availability of the identified sources of value (data) to the right user at the right moment (Zhou & Benton, 2007; Wu et al., 2006). Moreover, ICT systems need to support each participant in achieving its business goals, without introducing new and unexpected sources of concerns such as data leakage risks (Tang & Musa, 2011). Indeed, while the development and deployment of collaborative ICT systems is quite diffused, they often provide less benefits than what was initially expected due to misalignment with individual business goals of supply chain participants.

Aeronautic and consumer goods industries are two exemplar cases in which such innovations are pursued by leading global players. They strongly feel the global competition, so they need to play a leading role also in the supply chain management strategies also.

The aeronautic industry that produces high capital intensive and long lifecycle products is pushing the design and development of innovative ICT systems, in order to respond to the need of managing the whole aircraft lifecycle in integrated frameworks (from the product conceptualization and design phases to the maintenance services provided during product usage by customers). This trend is evident in the development of the Product Lifecycle Management (PLM) conceptual frameworks and systems, and in the continuous challenge to integrate product data in the different systems applied during product design, manufacturing and after sales service provision. The main challenge is connecting the most relevant and business valuable data of supply chain partners. This challenge is hard to solve for two main reasons:

1. Coupling different ICT systems used in different layers of the supply chain is not easy since IT capabilities of big companies and SMEs differ a lot and the production tasks they carry out require very different applications, data formats and database structures.
2. Making organizational data available to partners add value to the entire supply chain, but involves high risks to the data owners.

The consumer goods industry, which is characterized by high volatility of the market, is strongly committed to smoothing and increasing the speed of product flows along the supply chain in order to capture the consumer requests without increasing excess inventories that are expensive and quickly become obsolescent. Different business approaches were developed and applied with such aim, and all of them force companies to strengthen IT connections in the supply chain. Business relationships and agreements between client(s) and supplier(s) are changing with the result of moving the responsibility on products availability to the producers (OEM and its suppliers) instead of access to actual demand data. In this case, the highest concern comes from the intrinsic structure of the global consumer goods supply chain. Indeed, it is strongly inter-connected so that players compete in the same local markets (e.g., subsidiaries of OEMS and wholesalers belonging to the same area) could manage the same demand database. Similarly, many suppliers of components provide their products to different and competing global actors, so they could make decisions in favour of their most profitable clients.

Cloud systems and secure multiparty computation are the ingredients enabling the conceptualization and the implementation of new business models in these industries. Cloud systems are an effective solution to make available standardized collaboration services in a supply chain, overcoming the difficulties in creating direct communication channels among IT infrastructure of partners. Cloud services, moreover, standardize collaboration patterns and related contents, reducing complexity of supply chain and efforts for supply chain management. Secure multiparty computation technology is going to provide more effective data protection capabilities to data processing applications, so that also confidential data can be used to shape supply chain production.

This report explores three main topics related to some of the most critical challenges in the mentioned supply chains.

Firstly, the business models that are going to spread in the aeronautic and consumer goods industries are presented. Section 2 describes the dynamics of the supply chain of the after sales service in the aeronautic industry, specifically the business and technological dynamics pushing the new concept of fleet management in the aero engine industrial section, and the dynamics of the consumer goods industry in terms of data flow required to introduce a Vendor Managed Inventory business model. In this part, the business goal for each supply chain is presented and modelled. In the fleet management scenario, the global business goal is to increase the engines' availability, i.e., to reduce servicing time without increasing costs related to stocks of spares or the number of machines and tools. In the consumer goods scenario the business goal is to satisfy the demand of consumers, i.e., to provide products on the sellers' shelf on time with respect to the demand, without saturating inventories and adopting a heavy reverse logistic (necessary to manage unsold products). Introduction of such an ambitious business goal in the supply chain implies each participant re-shapes its business processes by reducing local decision making scope, for example moving from a decision making process based on direct customer priority to a decision making process based on end users' actual needs. Inventory management and purchasing policies, among others, are strongly affected by this approach.

As a collaborative cloud system requires that users provide private data in a common system[1], data leakage and the related business risks need to be modelled and quantified. Using this, it is possible to adapt the security features of the cloud system to the security aspects of the supply chain business model.

[1] Actually, more complex cloud architectures are explored and designed in other WP of the PRACTICE project.

The main result is the recognition that business risks are strongly dependent on the couple *data owner-recipient* involved in the leakage event, and they are based on the capabilities of one of them to shrink its own profits or to increase costs of the other.

Section 2, after discussing the new business models for the aeronautic case (the fleet management process and the Vendor Managed Inventory for consumer goods industry), propose, the functionalities of forecasting and planning systems supporting the inter-related business models and the data protection features.

Section 3 presents formal models and algorithms required for collaborative supply chain management. Models and algorithms computing supply chain wide optimization functions are more complex than the optimizing models at individual node level, firstly because of more variables, parameters and constraints involved, then because of the optimization objectives depending on the specific supply chain structures and industrial models. Indeed, different models and algorithms are developed for the aeronautic and the consumer goods cases.

In the aeronautic case, the service needs can be anticipated by mining product usage data (in general owned by end users), and, at the same time, the service provider having some flexibility in planning the service. Thus, it is possible to plan the engine maintenance, repair and overhaul (MRO) service when the actual resources (in terms of persons, tools and spare parts) are available in the service provider base, thus reducing turnaround-time (TAT). **The innovation of the presented models and algorithms is the capability to incorporate collaborative maintenance demand forecasting, scheduling of the engine arrivals, spare parts management and capacity planning modules in one collaborative planning system.** The difficulties that must be overcome in deploying this computation logic are mainly related to the confidentiality of the involved data, indeed, if data required in these models would be revealed to other partners the data-owner would run risks to increase its costs or to lose part of its profits.

In the consumer goods scenario, the focus of the model faces the short time the supply chain has to react to market evidences. The vendor managed inventory approach is taken into consideration for this case. **The models and the algorithms target the case of a single vendor (the OEM) serving many retailers**. The computation of market forecasts is complicated since it requires confidential data belonging to a number of competing actors.

Section 3 discusses also more innovative techniques available to coordinate activities in supply chains. Auction models and systems are already quite diffused in the B2C scenarios, but auction designs may improve also the B2B supply chain coordination.

An auction is a set of trading rules (or a protocol) that dictates how to select the winners (one or more) of an auction and what the winners buy or sell. The improved coordination may come by ensuring a better coordination of the product flow and between price and other attributes. The auctioneer, the person managing an auction to ensure that everyone follows the protocol and shares information as prescribed, has access to the data provided by participants; thus he is the risky node in the data flow because he could behave opportunistically instead of respecting the logic. **Secure multiparty computation is able to honestly implement the logic of the auction and to compute correct results, while not revealing any input to participants.** Some application cases of auction systems in B2B scenarios are reported for future analysis with respect to the fleet management and consumer goods cases.

Section 4 provides a risk analysis for insiders' attacks, those in which one or more parties involved in the collaborative cloud processes succeed in obtaining more information than what the correct protocol execution prescribes. There are many causes for such events, such as the intentional publishing of supposedly protected information items, the carelessness in the communication protocol implementation and deployment, inadequate data protection practices at the user side (sharing the same terminal with other users able to reconstruct the information items held, and so on). By looking at these cases, a methodology for realizing a risk assessment and evaluating the related risk mitigation strategies is proposed. In particular, the risk mitigation strategies taken into consideration are those implementing secure computation technologies deployed in a cloud infrastructure. In particular, different infrastructure architectures and computation methods can be evaluated by analysing information accumulated in different nodes. **This methodology offers a novel way to evaluate investments in the data protection features of collaborative systems**.

Section 5 provides a brief overview of possible techniques that are introduced in other work packages of the PRACTICE project and can be useful for implementing the models presented in this work. Different approaches are possible and selecting one comes along with a trade-off between functionality, security and efficiency.

In this report, the content structure follows this schema: conceptualization of the business model, recognition of the data conveying business value for organizations and the entire supply chain, preparation of the computing algorithms returning results to the involved parties, towards the design of a cloud-based collaborative planning system presenting an effective security-risks-costs trade-off.

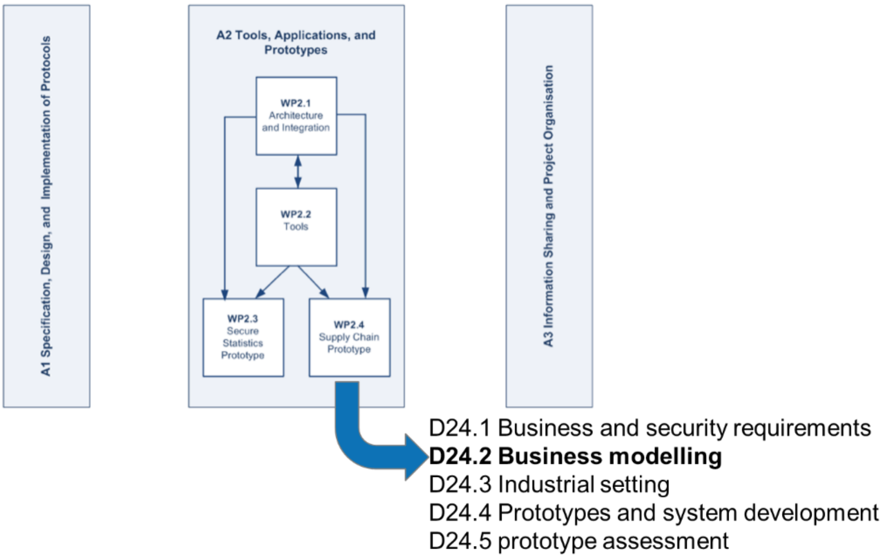


Figure 1: Position of this deliverable with respect to the whole PRACTICE project.

Chapter 2 Industrial process models

2.1 Business modelling

In recent years, business relationships and IT innovations are changing the way in which customers and suppliers cooperate to create value and to respond to the pressure of the global market (Rebolledo and Nollet, 2010; Schubert P. and Leger C., 2011; Khajavi et al., 2013).

The increasing quantity of digital data regarding the entire lifecycle of products and components, for example about the organizational resources usage plan, or about the market needs and opportunities, is a new source of business value (Du et al., 2011; Braradwaj et al., 2013). In order to leverage such value, organizations involved in layered supply chains are reshaping their more strategic relations from an 'Arm's Length' model to a relationship model (Hoyt and Huq, 2000).

Long term relationships have been creating opportunities for cooperative and complementary development of competences and capabilities, thus increasing individual as well as global value proposition through which addressing the market as a whole system. Significant greater value can be created by leveraging data available in local (organizational) databases in an integrated and supply-chain based framework. In order to apply a global data management framework it is necessary that all actors give away part of their decision power to the centralized system (even if it should be transparently governed through the data management framework) instead of an improved margin of the additional business value created (Hamlen and Thuraisingham, 2012; Demirkan H., Delen D., 2012).

Such trend is prone to the creation of lock-in traps for SMEs, and, as sharing organizational data is also sharing business sources and drivers, large and more capable partners can exploit the value without returning the fair value to owners. For this reason, the diffusion of such supply chain innovative system is quite slow and regards very few partners, the most strategic ones mainly. Nevertheless, this business challenge is pursued by every firm involved in supply chains (Shih et al., 2012).

In order to enable supply chain participants to apply new supply chain collaborative schemes it is required that a global business model is developed. It is generally composed of the recognition of the value sources, the way they have to be managed and processed, the risks associated to managing them in a collaborative framework, and the value created. In this report changes characterizing the aeronautic and consumer goods industries are presented, with a focus on the after-sale service in the aero engine business segment and the new product supply chain of a large multinational firm in consumer good industry. In particular the collaborative business scenarios, the data model enabling the collaborative data processing, and the benefits provided will be discussed.

2.1.1 Business innovation in the aeronautic industry

The aeronautic industry is experiencing a main innovation change in the supply chain management. The last two most important products were developed and produced by leveraging more tied relationships among supply chain partners, now it is expected that better supply chain performance can be achieve by sharing data and information by engaging supply chain partner in the same business objectives. The diffusion of 'risk sharing' relationship among clients and module suppliers (Alfalla-Luque et al., 2013) and 'power by hour' servicing contracts[2] between airlines/air forces and MRO service providers (Smith, 2013) corroborate this evidence.

Until last decade, business relationships between airline/airforce (that are the engine owners) and the Maintenance, Repair and Overhaul (MRO) service provider were mainly based on 'time and material' contract types, business goals of the service provider were to service out-of-service engines. In the last decade, instead, the 'power by hour' contract types have started to diffuse. It strategic change is that MRO service provider is paid for the efficient engines ready to fly instead of for the maintenance activities or spares. In this contract framework, the business goal of the service provider is to maintain the efficiency level of the whole customers' engine fleet, allowing airlines (or air forces) to reduce costs, mainly those related to spares, and to increase profits, thanks to a higher efficiency of their engine (Guajardo et al., 2012).

The aeronautical industry is capital intensive: inventory is a relevant cost and strongly determines the quality of the service (i.e. in terms of time delivery); resources (employees and machineries) also are very expensive, thus a high usage efficiency is mandatory. The profitability of the fleet management, in the 'power by hour' framework, depends on the accuracy of forecasts, which lead to lower safety stocks and higher servicing capacity. Thus, the possibility that MRO service providers are able to achieve profits (or reduce its fees) in the new business scenario is related to the capability to directly involve clients in the service forecasting and planning process.

The trouble is that aeronautic players are very reluctant to use a system shared among customers and service provider due to confidentiality reasons. The result is poor planning capabilities as the service plan at the MRO base is realized by 'client managers' based on face to face negotiations about customer priorities rather than on information on actual and planned resource usage and status.

In the following sections, the aero fleet management will be described in order to show the data required to compute service forecasts and plans, and the related business risks, associated with such a collaborative system.

2.1.2 New business models to supply new products in the consumer goods

For years Consumer Goods' business is mainly aimed to satisfy personal requirements and also focused on the effectiveness and efficiency of separate business functions, like planning, controlling and designing a supply chain as a whole.

Personal requirements are the trigger for consumer goods industry, and they are all dynamically changeable. Requirements are affected by people's disposable incomes, consumer attitudes toward product and brands, different geographical conditions, different ages, genders and so on, that companies have a key role on being dynamic, innovative and open minded, which bring us to creating new business models to supply new products in the consumer goods. Those differentiations of personal requirements cause great fragmentation on products variety, which creates complex business models, categorized production portfolios, and also huge and much more confidential database models, because of the increased competition of both customers and companies. There are mainly two kinds of flow throughout the chain: flow of goods and flow of information. Most of the documents are created and transferred through the chain manually. Delays in the information transfer and

[2] 'Power by hour' is a general name for a type of contracts in the aeronautic MRO business sector characterized by fact that the supplier is paid for the availability of a product at the customer base (http://knowledge.wharton.upenn.edu/article/power-by-the-hour-can-paying-only-for-performance-redefine-how-products-are-sold-and-serviced/). In this way, supplier is strongly committed to reduce its servicing time.

data reliability problems are mostly due to one-to-one communication, manual communication and data processing with a high risk of human errors. The key functionalities for such a cloud based planning system relate to aspects of security. The security is extremely important because highly sensible data (such as strategic decisions about production and distribution, info on supply chain partners and their tariffs etc.) will be stored and shared through the cloud.

In a supply chain for a typical consumer product, even when consumer sales may not seem to vary much, there is a huge variability on products. Orders to the manufacturer and to the manufacturers' supplier increase even more rapidly. Consumer goods supply chain networks are differentiated by several different dimensions; by the nature of the markets, by product ranges and also by sourcing types etc.

On the other hand, a wide range of products and sourcing modes generates complexity. As it seems on the Picture 1, production facilities, warehouses and customers are distributed in different cities and countries. In addition, all production plants have different production and material flexibility. Refrigerators and washing machines have more than one production plant in the supply network. Large home appliances are mainly supplied by the production facilities of the company whereas small appliances are generally outsourced from suppliers. In total, production deals with over thousand suppliers that are distributed mostly across Europe and Asia. Even though all products are assembly manufacturing products, inventory management strategies or monitoring requirements are different for each product group as well.

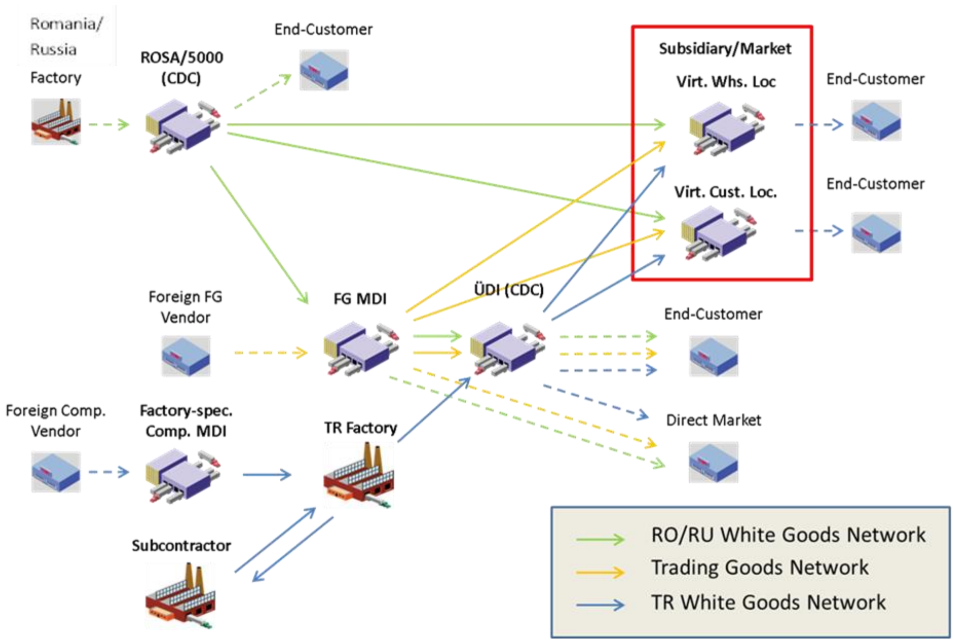


Figure 2: Complex Network Model of ARC

The product range contains white goods as well as consumer electronics. Since consumer electronics have a relatively short product life, critical components/materials become rapidly obsolete. For instance, given the share of a display in the total material cost of a LED TV (around 60%) and given the supply lead times, the incentives in monitoring and managing material and work-in-process inventories needs to be more aggressive in consumer electronics than all white goods.

The interaction with the business partners is still limited to manual, means such as e-mail, phone, and fax or only partially supported by ICT solutions which creates a high risk of data leakage and security problems.

When conducting business in such highly networked, often border-crossing, dynamic and competitive environments, it becomes crucial for the involved actors to collaborate in an efficient, secure and trustworthy manner. Linking demand with the supply throughout the entire supply chain is required for implementing tailor-made supply chain strategies in order to increase reliability and responsiveness to customer with a cost efficient and high quality.

To enable strategic visions, keep up with the changing consumer requirements and booming technology, consumer goods executives are focusing on operational efficiencies, technological updates, novel solutions and cost reduction initiatives. Key element for managing the wide variety of needs is having a dynamic, efficient and confidential business model which covers the supply chain network as a whole.

In an environment where data available in local databases are leveraged and combined within a global data management framework which enables more secure and faster internal communication, more accurate information flow will be observed and information updates (like revised requested date, demand quantity or production lead time..etc) will be immediately shared from one source with the whole network through secure channels.

In today's increasingly competitive, consumer-driven marketplace, being aware of customer requirements, handling successfully with the bullwhip effects, minimizing the risks and the limitations about both information and production flows of complex network models in supply chain and also creating successful product innovation ideas are more important than ever before. New strategy development will consume more energy and time. To continue to drive profitable and sustainable growth, companies need to develop strategies to appeal to today's consumers and leverage technology to enhance customer experience. Consumer goods companies need to focus on tapping into new idea sources, new business models and more agile engineering processes. With these new idea sources they need to apply innovative strategies of customer demands.

2.2 Aerofleet management

According to Schauenburg and Sinha (2008), fleet management addresses the operational and support issues for effective vehicle life-cycle management. It is possible to define it as a network of systems that monitors the usage status of aircraft and components and provides the required support. A poor fleet management can entail an outstanding increase of the costs due to excess inventory, misuse of resources (e.g. manpower and tools), time delays, loss of data, missed or improper maintenance and low asset operational availability (Operational Availability Handbook, 2004, by Reliability Analysis Center).

In aeronautics, the necessity for the MRO service provider to adopt efficient IT systems, able to manage the fleet of its customers and, in general, the great amount of information related to the supply chain, is expressed in different papers of professional services firms (some of them are Accenture, 2005; Capgemini, 2009; IBM, 2008; Deloitte, 2012). However, in the aero fleet management, confidentiality of data has to be taken into consideration. Data are related to components performance, the way in which the overhaul activities are performed in the MRO plants (in terms of resources involved, turn-around-time and inventory management), the capacity of the suppliers of MRO service providers and the penalties defined for unsatisfied service level agreement between the client and the MRO service provider. Such data is considered confidential since diffusion to competitors or other supply chain participants can entail loss of competitive advantages.

The Cloud Planning System (CPS), whose functional requirements and use cases are detailed in D24.1, is required to monitor the aero engine modules status during their operating phase in order to forecast the demand of MRO services, and to gather data describing production plan of the entire overhaul supply chain. In particular, the CPS adoption will make the MRO service able to:

- Plan shop maintenance slots;
- Optimise materials provisioning;
- Avoid aircraft on ground (AOG) due to lack of engine replacement;
- Ensure adequate capacity by service providers for subcontracted work;
- Optimise engine time on the wing within an adequate trade-off with risk of in-service failure and consequential losses.

The cornerstone in the CPS is innovative data security technology - Secure Multiparty Computation, which allows all supply chain participants to provide confidential information to the collaborative system without fearing other partners (or external actors) being able to read them. The secure multiparty computation technology uses cryptographic primitives and protocols to ensure that none private data can be learned by any party involved in the computation, nor any other attacking party.

2.2.1 Description and design of the Aerofleet management process

In the Figure 3, the aero engine overhaul process is represented, and the relations among airlines, MRO service provider and suppliers are highlighted. In this analysis, the trigger of the MRO process is the storage of the shipped (to-be-overhauled) engine into the MRO warehouse (Store A) where it will remain until the service activities start.
The first MRO activity is the general inspection of the engine; it is composed by the following tasks:

- The engine is disassembled in its modules and components;
- Each module is cleaned;
- All modules are inspected through non-destructive (NDT) testing;
- An accurate inspection is carried out on each module.

During the general inspection, there are some activities (disassembling and inspection) that can be performed concurrently on different engine models, using different human resources and tools, while other activities (cleaning and testing) are performed in sequence, independently of the engine model, applying the same machines and human resources (i.e. the disassembling of EM1 is parallel with disassembling of EM2, but the two engine models have to be cleaned sequentially, see Figure 3). In the latter case, the service policy FCFS (First Come First Served) is usually adopted.
At the end of the general inspection phase, the single components can be:

- Repaired and then allocated in the store B;
- Replaced by new components already available in the MRO warehouse or purchased from suppliers;
- Directly stored in store B since no repair is required.

Hence, suppliers can be involved into the overhaul process in two cases:

- In the first one, if components have to be replaced by new ones that are not available in the MRO warehouse;
- In the second case, if MRO service provider stocks safety spare parts to have them available when required.

Business relations between MRO service provider and parts' suppliers are regulated with agreements and usually there is a lag between the request of parts and their delivery.

The next phase includes the engine kitting: all modules and components required to assemble the engine are taken from the store B, and the engine is assembled.
After assembling the engine, performance is checked. Positive tests conclude with certification in accordance with regulation, negative tests are followed by new (specific) maintenance activities. Certified engines are delivered to airlines (or air forces) and installed on the aircraft.

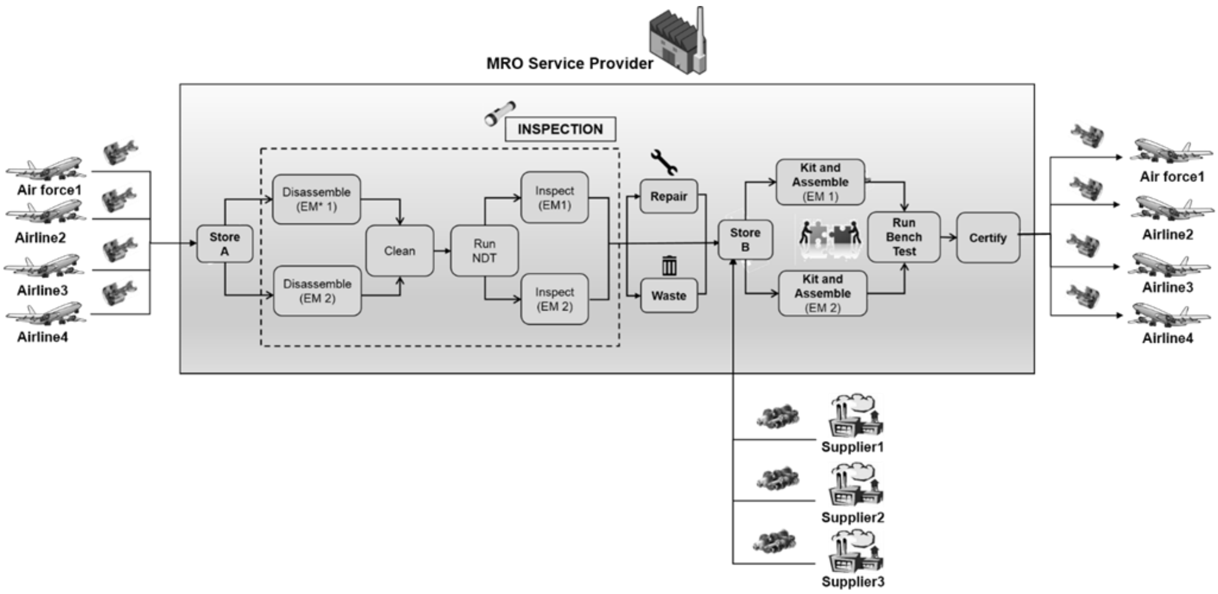


Figure 3: Aero fleet management process. Source: Author's illustration.

In order to complete the analysis of the overhaul process, the aero engine structure is highlighted. In order to simplify the planning computation algorithm without losing process and product particularities, the engine is considered as composed of seven major modules (Figure 4):

1. *Fan module*, that has the main task to draw air into the engine, compressing the bypass stream to produce part of the engine's thrust, and feeding air to the gas turbine core;
2. *Low Pressure Compressor (LPC)* whose purpose is to increase the pressure of the air in the next components of the engine;
3. *High Pressure Compressor (HPC)* module, that has the main function to raise the pressure of the air supplied directly to the combustion chamber;
4. *Combustor module*, in which fuel is added to the air in order to create thermal energy;
5. *High Pressure Turbine (HPT)* module, that acts in order to extract the combustion thermal energy for driving the high-pressure compressor and accessory gearbox;
6. *Low Pressure Turbine (LPT)* module whose role is to extract the remaining combustion thermal energy to drive the fan and low-pressure compressor rotor assembly;
7. *Accessory Drive module*, usually attached to the engine core or fan case, transfers mechanical energy from engine to drive the basic engine and aircraft accessory (such as generators and hydraulic pumps).

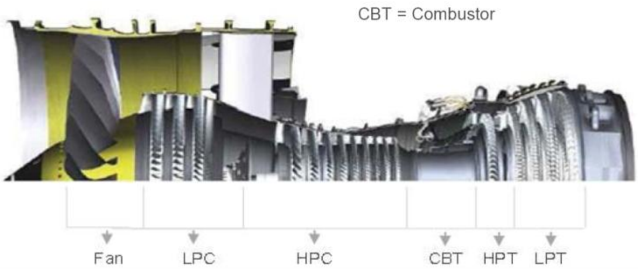


Figure 4: Modular architecture of a conventional twin-spool turbofan engine. Source: S. Ackert, 2011

2.2.2 Confidential data managed in the process

Within the supply chain of the aero engine overhaul, sharing confidential data with supply chain partners is a key enabler for improving economic performance, by efficient use of resources, reduction of Turn Around Time (TAT) and inventory optimization. In Figure 5, data categories that each partner has to share into the CPS collaborative system are reported (for more details see deliverable 24.1).

Figure 5: Data categories into CPS database. Source: Author's illustration

In the Table 1 the confidential input, business functions available on the CPS and the valuable results are reported for each partner.

Airlines/air forces have to share engine working parameters (I.e. flight hours, cycles numbers, and, in more advanced scenarios data gathered by engine sensors[3]) in order to have monitored their own fleet status. Such input data is useful for the entire overhaul supply chain since the MRO service provider needs to forecast demand.

MRO service providers have to share their current production plan, inventory levels, and penalties conditions, in order to have in return:

- Forecasts of the engines in arrival and the service plan;
- Inventory plans;
- Penalties management;
- Resources management.

Hence, the output results for the overall supply chain concern the possibility to:

- Satisfy aggregated MRO needs in terms of demand forecasts (on long term period);
- Plan customer-based MRO services (on short term period);

[3] A deeper analysis of the individual data to be shared is reported in the section 2.4.2 – Data model.

- Define the penalties risk and their amount (both for each customer and the aggregated one);
- Forecast/Plan material purchasing requirements;
- Plan inventory levels;
- Establish the resources usage (current, planned and forecast).

Parts' suppliers have to share their current production plan and inventory levels, in order to forecast demand and define production plans.

Table 1. Input data, business functions of the system and output results. Source: Author's illustration.

	INPUT	FUNCTIONS	RESULTS
AIRLINE/AIR FORCE	✓ Engine working parameters	✓ Fleet status monitoring	✓ Fleet and engine working status ✓ MRO needs (demand forecast)
MRO SERVICE PROVIDER	✓ Current production plan ✓ Inventory levels ✓ Penalties conditions	✓ Forecast and plan of MRO services ✓ Inventory plan ✓ Penalties management ✓ Resource management	✓ (Long term) aggregated MRO needs (demand forecast) ✓ (Short term) customer-based MRO service plan ✓ Planned inventory levels ✓ Penalties risks and amounts (aggregated and per customer) ✓ Material purchasing requirements (forecast and plans) ✓ Resource usage (current, planned, forecast)
PARTS' SUPPLIERS	✓ Current production plan ✓ Inventory levels	✓ Parts' demand forecast and plan	✓ Aggregated part's needs (parts' demand forecast) ✓ Material/parts purchasing orders

2.2.3 Risks of data leakage

Information sharing within supply chains allows all involved actors to have better performance, but it is not costless: making information available to a common computing system requires investment in information technologies (ITs) and leads to risks of leaking confidential data (Jason Shiu Kong, 2007).

In the following analysis, we consider risk related to the probability of confidential data leakage to supply chain partners or IT service provider.

Firstly, divergent interests and opportunistic behaviours of supply chain partners, and informational asymmetries across the supply chain can affect the quality of information shared (Lee and Whang, 2000). Groves and Loeb (1979) suggest that organizations deliberately distort information that can potentially reach not only their competitors, but also their own suppliers and customers, in order to preserve its own economic plans or increase their benefits. Hence, many firms are reluctant to share private information since they fear that their confidential data can be divulged to competitors or used for opportunistic bargaining (Xu and Dong, 2004).

Manatsa and McLaren, in their research work (2008), speak of three main categories of risks (or costs) that firms have to face during information sharing practices:

- *Coordination costs* for gathering, collating, and sharing information with other firms;

- *Operational risk costs* related to the shirking or reneging on previous agreements;
- *Opportunistic risk costs* linked to information used inappropriately to gain bargaining advantages or provide information to competitors.

Here, the third risks typology is taken into consideration. Applying a collaborative system, the opportunistic risk occurs when an insider actor (i.e. a supply chain participant or the IT service provider) can access input data provided by other participants, but also data managed during the computation, through which inferring more information than what can be inferred from computation results addressed to him.

In the case of the aero engine overhaul supply chain, risks for the appropriation of confidential data are different depending on the victim (the actor whose data are accessed) and the attacker (the actor who access private data). In the following, such risks are matched with risks identified in a Eurich et al. (2010) study, such as:

1. L*oss of information advantage.* Disclosing sales information to other organizations may change relationships among partners in a supply chain, since it may reduce the “information rents” effect for which especially weak partners profit;
2. *Reconstruction of strategic decisions.* Sharing of data increases the visibility of operations for all companies involved; thus if confidentiality is not preserved, a competitor could anticipate company’s future plans;
3. *Development of a competitive product/service.* With the knowledge of suppliers, competitors can develop new competitive products or offer more competitive services;
4. We*akening of the bargaining power after disclosure of purchase or supply volume.* A customer may compare its own purchase volume to that of other customers in order to calculate its share; such information can be used to strengthen its bargaining power*.* At the same way, suppliers can calculate their share of overall supply; this information increases their power over the customer in a business negotiation.

Airline companies risk reducing their market share and profit with respect to competitors: if fleet status is visible to another airline, it can apply an aggressive market strategy targeted on the less efficient routes/aircrafts (*reconstruction of strategic decisions*). The risk of the MRO service provider is related to the reduction of bargaining power since MRO service provider needs and profitability can be better evaluated (weakening *of the bargaining power after disclosure of purchase volume)*. Lastly, the risk run by airline/air force against parts’ suppliers is related to the probability that suppliers renegotiate the delivery deadlines according to their production needs (*weakening of the bargaining power after disclosure of purchase volume)*, as well as they offer airline/air force information to outsiders (l*oss of information advantage)*.

The same way, the MRO service provider could have its confidential data attacked from airlines, other MRO service providers or suppliers. In the first case, the service provider runs the risk to lose bargaining power during trading relationships with the airline/air force (*weakening of the bargaining power after disclosure of supply volume)*. In fact, from information on resources availability, inventory management and Turn Around Time, effective service quality level can be inferred (and higher penalties can be imposed). In the second case, data of the MRO service provider are attacked by a direct competitor (also user of the cloud system)[4]; here the risk is a reduction in competitive power and resulting market share decrease (l*oss of information advantage* and *development of a competitive product/service)*. The competitor, who accesses data without authority, can adopt a production and/or market strategy focused on stealing customers from the data owner. Finally, in the case a supplier

[4] In the considered supply chain, there is only one MRO service provider; on the contrary, in real cases, airlines are served by different after sale service providers, in example depending on on/off-wing services. Moreover, we are designing a cloud system that can be used by different MRO supply chains, that means a number of MRO service providers will be users of the system.

accesses service provider data, it can negotiate product delivery deadlines threatening the manufacturing plan (*weakening of the bargaining power after disclosure of purchase volume)*, or can sell data to other service providers' or to airlines (l*oss of information advantage)*. In this way, risks are related to the loss of control on manufacturing plans and service quality, and to the decrease of market share.

To conclude, suppliers risk a reduction of their market share against competing suppliers (*development of a competitive product/service)*. Against MRO service providers, they risk losing its bargaining power and profits, i.e. due to higher penalties (*weakening of the bargaining power after disclosure of supply volume)*; while they risk their private data being sold if other participants (i.e. airlines/air forces) access them (l*oss of information advantage)*.

It is also important to consider that if confidential data is acquired by an external actor to a specific supply chain, the victim's competitive position will be affected since the attacker could offer private data to another supply chain participant in exchange of economic benefits (l*oss of information advantage)*. In general, increase in violated information quantity grows the risk the victim reduces its competitive power.

Table 2 sums up risks for each actor of the aero engine supply chain in relation to the attackers.

Table 2. Risks for the actors of the aero engine overhaul supply chain. Source: Author's illustration.

VS	**Airline/Air force**	**MRO service provider**	**Supplier**	**External actor**
Airline	✓ *Reconstruction of strategic decisions*	✓ Weakening *of the bargaining power after disclosure of purchase volume*	✓ *Weakening of the bargaining power after disclosure of purchase volume* ✓ *Loss of information advantage*	✓ *Loss of information advantage*
MRO service provider	✓ *Weakening of the bargaining power after disclosure of supply volume*	✓ *Loss of information advantage* ✓ *Development of a competitive product/service*	✓ *Weakening of the bargaining power after disclosure of purchase volume* ✓ *Loss of information advantage*	✓ *Loss of information advantage*
Supplier	✓ *Loss of information advantage*	✓ *Weakening of the bargaining power after disclosure of supply volume*	✓ *Development of a competitive product/service*	✓ *Loss of information advantage*

2.2.4 Trust relationship in the supply chain

The aero engine overhaul supply chain is characterized by companies of various size, with very different culture and level of maturity in partnerships management. However, they need to collaborate to achieve common objectives and the related benefits.

Trust between partners represents the basis of collaboration; it is defined as "the firm's belief that another company will perform actions that will result in positive actions for the firm, as well as not take unexpected actions that would result in negative outcomes for the firm" (Anderson and Narus, 1990). Nevertheless, in overhaul supply chains, inter-organizational data sharing is often a hostile process (partners are reluctant to share confidential data) since competitors can be tied to supply chain participants or directly involved in the supply chain; partners can abuse shared information to have all the benefits of information sharing or can lack care for protecting them in the future.

Depending on the effort spent and the technique applied to attack the system in order to access private data, attackers can be classified as semi-honest, covert and malicious (Aumann and Lindell, 2009).

In particular, the CPS service provider can be considered a *semi-honest adversary*: if it "correctly follows the protocol specification, yet may attempt to learn additional information by analysing the transcript of messages received during the execution" (Miyaji and Rahman, 2010) and makes use of the data against other platform users. Actually, the service provider doesn't participate in the supply chain but only furnishes the ICT resources (hardware and software) to the supply chain participants, so its business position will not change if participants' private data would be disclosed to him. On the contrary, all supply chain participants (the airline or air force, the MRO service provider and its suppliers) are the actual users of the CPS, since they need to provide their data in order to compute the overhaul plans. Due to their behaviour they can be defined as *covert adversaries*: they may deviate from steps of the protocol in an attempt to cheat, but will not misbehave if they know they will be caught with significant probability (Goyal et al., 2008). In other words, they can either put in the CPS data overestimating their business condition in order to improve their business position untruthfully (an event that's not possible to cope with), or attack the system aiming to acquire partners confidential data. Anyway, these events have low probability because, if detected, their business relationships with other participants will be definitely broken.

At last, in the case the MRO service provider provides also the cloud system, the risks for the other participants are higher: the MRO service provider could attack the system itself to access confidential business data of its suppliers and customers or to correct results. In such a case the attacker operates as a 'malicious' actor against its business partners and the probability to be detected is quite low.

2.3 Consumer goods industry

2.3.1 Description of Consumer Goods Planning & Production Systems

The increasing complexity and dynamic changes in consumer good markets' demand has a greater speed of adjustment and flexibility from companies throughout the entire supply chain (suppliers, OEMs and distributors) all the way to the consumer. Quickly recognizing and evaluating changes and adapting organizations to accommodate these changes will become even greater critical success factors than they are today. These dynamic changes can be managed only with good, sharp and innovative decision making strategies. Strategically decision making needs to be supported by planning, by mathematical results and regressions, by reference points, which can be observed by good organized feasibility models. These planning tools are instruments for preparing decisions in a situation of uncertainty. Decisions need to be concerned within different planning horizons. Horizons can be distinguished in three different planning levels: long-term planning, mid-term planning and short-term planning.

Long-term planning covers strategic decisions about the future development of the supply chain. This includes decisions about the design of the supply chain network. The planned and implemented measures usually affect several years. On the other hand, the main task of mid-term planning is to make regular operations based on the strategic decisions definite. Therefore, strict forecast quantities need to be received and after forecasting period, the material flow has to be determined. Short-term planning has to specify detailed and definite instructions for the execution and control of operations.

However planning decisions not only differ with respect to their horizon, but they can also be assigned to different functional areas: procurement (supplier side), production (plant side), distribution (inventory side), and sales (customer side). Consumer goods management planning process in ARC is detailed for customer, supplier and inventory level as below.

Customer Level – One-sided Demand Planning:

In ARC, there are two types of customers that finished goods are sold to: subsidiaries and direct sales markets.

Subsidiaries are companies which use their own identity in their local area like ARC in Turkey. However, they are also companies that financially dependent on ARC. They have all their own supply chain teams which are connected to central supply chain in Turkey. For every subsidiary in different countries, it has been used different methodologies according to their local conditions.

On the other hand, direct sales markets are not served via local companies. Sales teams, which are located in ARC headquarters, are responsible for sales forecasting according to their analysis on the customer needs of the specific market and they are getting in contact with the customers during the demand forecasting process. After receiving information about requested sales figures, for short term planning, pre-orders are submitted based on requested delivery date and lead time and they might be changed between 4 to 8 weeks due to differentiated transportation times for each sales market.

Another type of direct market is OEMs (Own Equipment Manufacturer) which are also production companies; by the way they used ARC factories for some of private label productions.

Beside the short term planning level, demand planning has to forecast the potential sales in specific regions for the mid-term planning level. Based on the given network and these sales forecasts, a definite capacity planning needs to be executed and a production schedule specified. This schedule, which may concern all plants simultaneously, determines the netted quantities which have to be produced in certain periods, in order to deal with time-varying demands.

Both order preparation flow and order netting process has been figured within the formulas by the Figure 6, below.

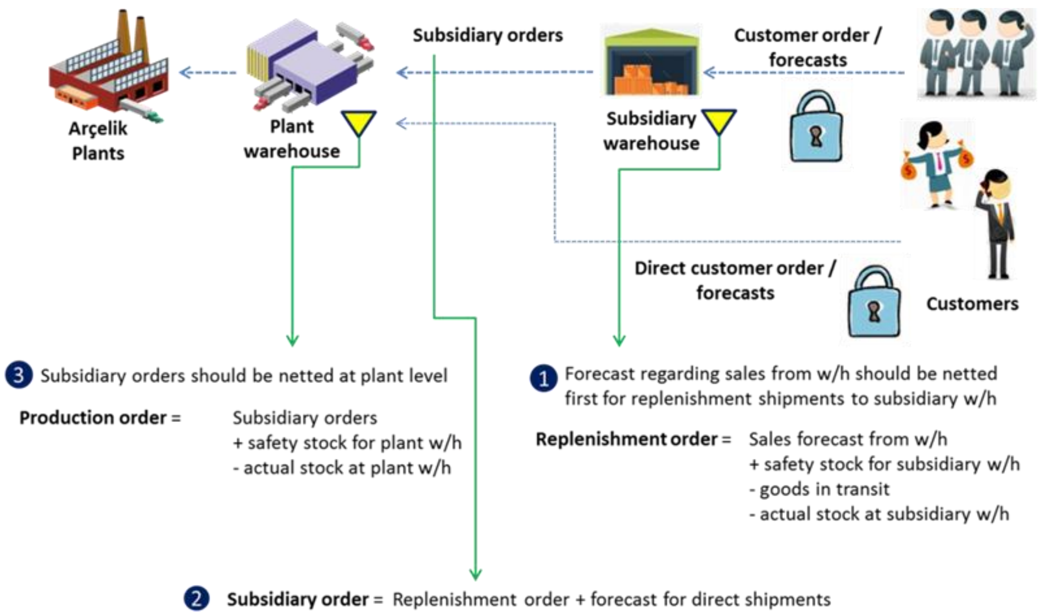


Figure 6: Customer Level Process in ARC

Supplier Level – One-sided Production Planning:

Based on customer requirements, productions are planned for each plant in ARC. With the help of capacity planning, the required material as well as the required personnel has to be calculated and the appropriate contracts need to be signed and in this case, plants informed suppliers about their future production plans to make them indicate their own material production plans. For each supplier, ARC has different agreement based on material production lead time, material life cycle... etc.

As today more and more components are purchased, the suppliers have to be selected carefully in terms of product quality, service, and further properties. Another topic is to build cooperation with suppliers. By sharing information about final customer demands and production capacities (among other things), such a collaboration may help to reduce, e.g., inventories, and thus costs, on both sides. Moreover, the decisions about the product and material program are also directly connected to decisions about the location of plants and the corresponding production systems, as they have an effect on how a customer can be reached.

The supplier sees a much smoother demand signal at the factory. This reduces costs by permitting better resource utilization for production and transportation; it also reduces the need for large buffer stocks.

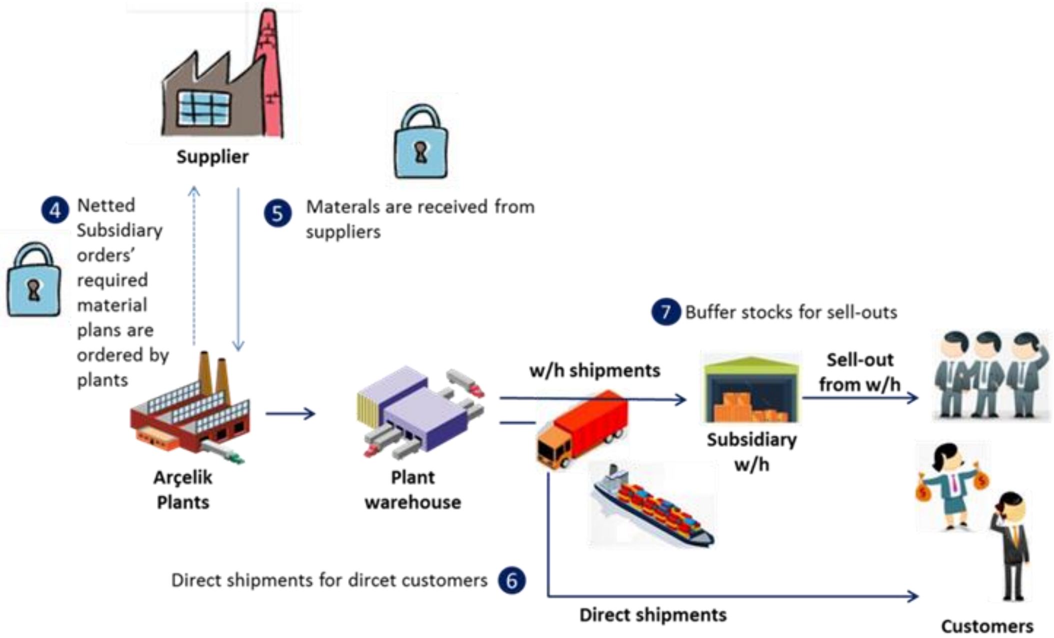


Figure 7: Supplier and Inventory Level Process in ARC.

Inventory Level – Two Sided Demand, Production and Inventory Planning:

Inventory is one of the most important indicator for ARC. For correct production netting, inventory figures should be accurate and visible, that affects also supplier requirements directly.

Inventory is the only concrete reference point for improvements for companies. Higher inventory levels can explain lots of troubles that the company experienced and needs to be aware of the problem. However when the inventory level is enough higher than it should be, there emerged a thought that the customer level will be increased within a positive manner, because inventory is a concrete and the only visible guarantee, which also covers the bigger underlying troubles like an iceberg.

Supply chain has been applying a replenishment system in order to achieve a higher customer service level by obtaining a buffer against uncertainties throughout Arçelik supply chain, both demand and supply sides and to keep stock for right model so as to decrease inventory on hand (days), decrease inventory fluctuation.

Firm orders and forecasts are given by subsidiaries, while safety stocks are determined by Supply Chain. Special occasions such as deals, promotions etc. are open to discussion with customers in mid-term planning horizon. Safety stocks are planned to be kept in Turkey especially for make-to-stock (MTS) “direct” shipments for A Class products, based on the theoretical ABC analysis. Calculated safety stocks are kept also at plant warehouses in Turkey, in order to accommodate demand uncertainty.

Since the location and capacity of a plant influence the physical distribution structure, they are also usually planned together. Depending on the production schedule and the sales forecast, the number of warehouses has to be determined or the decision for a logistic service provider has to be determined.

On the short-term level, demand planning is responsible for detailed forecasts and demand fulfillment. Sales team has to check whether a customer order can be satisfied with the products already available or whether a new production order has to be created. Distribution planning determines the detailed schedule for transport to replenish the stocks in the warehouses or to deliver them to the customers. Based on the short-term production schedule, a time-efficient shift schedule (extra shifts, overtime, capabilities of personnel, etc.) has to be determined.

2.3.2 Description of Consumer Goods Management Troubles

Customer Level:

Customers behave individually. They can reach any information they want to see, even other customers’ sales, stock and production plans. In order to get their complete requests on time in full, even they are not fully accurate; they try to get enough capacity. However there is no infinite capacity, so that there are some prioritization rules for production and stock allocation planning.

Then the problems occur accordingly.

- Decreasing Forecast Accuracy
- Lack of capacity usage
- Unplanned buffer stocks, accordingly increasing aged and excessive stocks,
- Uncertain real demand fluctuation, accordingly wrong theoretical future inventory planning
- Increasing Inventory day, with increasing consolidated stock figures.
- Decreasing OTIF (on time in full) ratios

Customers can reach every one of sales plans that received by central supply chain from every customer. Firstly, if there are pool products, that for example three customers want to have at the same period, in order to get the first production they may create the order and enter the request date into the system before they want to request just to be prior than other two customers and they may also enter their forecasts with buffer quantities, which central supply chain has no idea that those quantities are entered with buffer quantities into the system, just to get higher and definite stock allocation. They reflect the risks, which are observed according to some limitations of production (like MOQ), without giving any information to central supply chain. Because of that buffered forecasts and orders, sales results are observed within even more deviations.

Pre-orders are used by customers (subsidiaries and direct markets) as a strategic business management tool, to reduce the risk of being out of stock. (With respect to competitors and OEM suppliers)

However because subsidiaries can reach the data of other sales groups without any cloud based security system and they can use the data to get more capacity portion from plants, which can create excess stocks and unfulfilled production. With cloud based system they can share their sales requests without seeing others' requests. Only in plant and central supply chain level whole requests will be collected and netted within the available stocks on hand, and production plans and stock ownerships will be done according to production lead times and shipment dates beside their involvements, so that there won't be any inefficient usage of capacity.

Regarding to demands, production schedule is announced daily based. Every plant announces the figures separately. In Table 3, demand creation date, requirement date and lead time figures are main inputs. Daily production figures are indicated according to requirement dates, which are also gathered within sales figures from subsidiaries and direct sales groups. Lead time data is also taken into consideration for the determination of production dates/availability dates.

Customer	Plant	Demand Creation Date	Requirment Date	Lead Time / wk	Available Date
Beko PLC	REF	01.11.2014	25.11.2014	3	30.11.2014
	DWA	05.11.2014	17.12.2014	6	02.12.2014
	WMA	28.10.2014	15.11.2014	3	15.11.2014
	DRY	15.10.2014	27.11.2014	6	03.12.2014
	CKG	12.10.2014	26.11.2014	6	28.11.2014
Beko Deutschland GmbH	REF	15.10.2014	28.11.2014	6	26.11.2014
	DWA	25.10.2014	06.12.2014	6	05.12.2014
	WMA	05.11.2014	18.11.2014	2	on stock
	DRY	12.09.2014	27.11.2014	11	27.11.2014
	CKG	18.11.2014	12.12.2014	3	on stock
Middle Africa	REF	09.11.2014	21.12.2014	6	on stock
	DWA	15.10.2014	22.11.2014	5	02.12.2014
	WMA	12.11.2014	28.12.2014	7	27.12.2014
	DRY	05.11.2014	27.11.2014	3	15.12.2014
	CKG	06.11.2014	26.11.2014	3	26.11.2014
South America	REF	02.11.2014	14.12.2014	6	06.12.2014
	DWA	15.10.2014	26.11.2014	6	15.11.2014
	WMA	07.11.2014	19.12.2014	6	02.12.2014
	DRY	25.10.2014	26.11.2014	5	26.11.2014
	CKG	01.11.2014	13.12.2014	6	13.12.2014

Table 3: Customer Demand Parameters

However production schedules are frozen only for one week. Within a month future planning is a hard business to manage in this case. Forecasting is not future information for ARC customers. They can even share current month "forecast" figures and they can significantly differ from actual demands, which are also indicated by customers in a short period of time. So that production schedules can be fluctuated also within month, which affects material planning and inventory projections negatively.

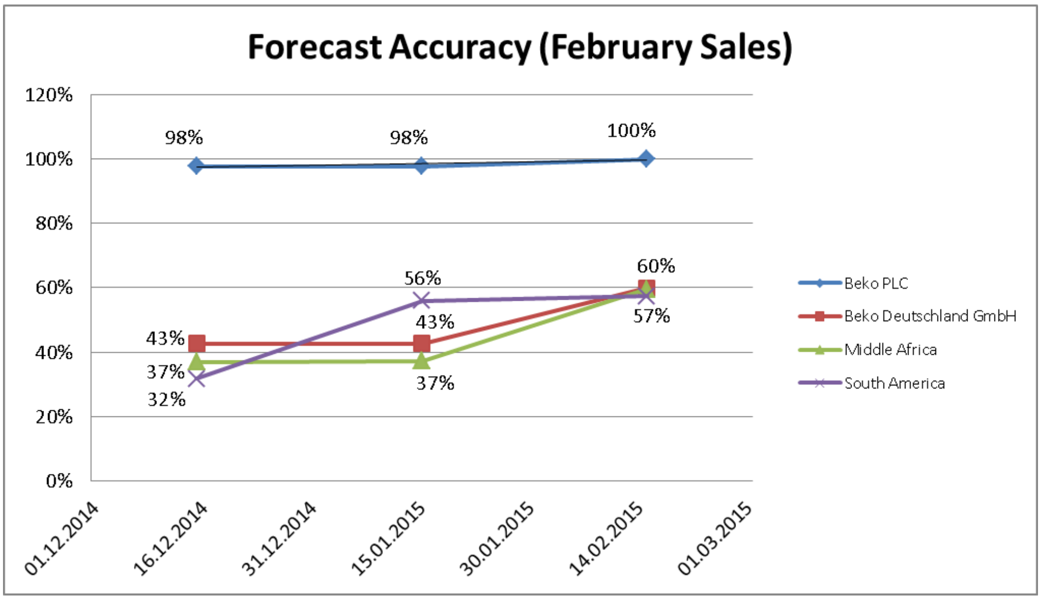


Figure 8: Forecast accuracy of customers.

Figure 8 is the reflection of the results of forecast accuracies of customers. For example South America region has a 57% of forecast accuracy even at 15th of February compared to February actual sales

Customers	Feb Actual Sales	15.02.2015	15.01.2015	15.12.2014
Beko PLC	176.649	176.649	180.526	180.526
Beko Deutschland GmbH	33.851	56.478	79.516	79.516
Middle Africa	13.109	22.028	35.236	35.497
South America	15.386	26.805	27.477	48.596

Table 4: February Actual sales and Forecast Figures.

Customers	15.02.2015	15.01.2015	15.12.2014
Beko PLC	100%	98%	98%
Beko Deutschland GmbH	60%	43%	43%
Middle Africa	60%	37%	37%
South America	57%	56%	32%

Table 5: February Forecast Accuracy Ratio

This is the main problem that ARC faced for optimization of the process. Actual sales figures information is quite critical for whole ARC network. Central supply chain needs to have the data keep confidential in between the customers. If cloud system provides us safe information flow, each customer shares actual forecast figures for actual requirement dates, not with additional buffer quantities or early requirement dates. That will help to decrease the deviation.

Based on production dates/availability dates, can be followed from Table 6, confirmation dates are determined customer based and also with the announcement of production quantities, for each order, fulfillment ratio can be easily observed.

Estimated time of departure is also followed up by a logistical schedule. Depending on lead time for each subsidiaries and direct sales locations, estimated time of arrival changed accordingly.

Main focus on customer and material based availability schedule is the loaded quantity information, which gives us an idea about customer fulfillment ratio. The whole procedure completion brings us to the OTIF (on time –in full) results are presented to subsidiaries and direct sales groups.

Customer	Material	Confirmation Date	Fulfillment Ratio	ETD	Loaded Qty	ETA
Beko PLC	7100541100	30.11.2014	95%	03.12.2014	177	15.12.2014
	7100642400	17.12.2014	95%	24.12.2014	144	05.01.2015
	7100841100	15.11.2014	97%	19.11.2014	177	01.12.2014
	7182681400	03.12.2014	90%	10.12.2014	207	22.12.2014
	7101141300	28.11.2014	95%	03.12.2014	159	15.12.2014
Beko Deutschland GmbH	7755582951	28.11.2014	95%	01.12.2014	108	21.12.2014
	7185281300	06.12.2014	95%	08.12.2014	159	28.12.2014
	7100841100	18.11.2014	95%	18.11.2014	177	08.12.2014
	7182681400	27.11.2014	97%	01.12.2014	207	21.12.2014
	7178981300	12.12.2014	100%	12.12.2014	175	01.01.2015
Middle Africa	7101341100	21.12.2014	100%	21.12.2014	159	20.01.2015
	7103241500	02.12.2014	92%	02.12.2014	152	01.01.2015
	7100841100	28.12.2014	100%	28.12.2014	177	27.01.2015
	7104241500	15.12.2014	88%	15.12.2014	152	14.01.2015
	7107342100	26.11.2014	100%	26.11.2014	175	26.12.2014
South America	7176781400	14.12.2014	100%	14.12.2014	207	13.01.2015
	7100841100	26.11.2014	100%	26.11.2014	177	26.12.2014
	7181581500	19.12.2014	100%	19.12.2014	177	18.01.2015
	7291548092	26.11.2014	100%	26.11.2014	35	26.12.2014
	7615349042	13.12.2014	100%	13.12.2014	159	12.01.2015

Table 6: Material based Availability Schedule

Supplier Level

Main problems of supplier level process:

- OEM Supplier: The time gap between pre-orders and orders impacts on costs of purchased components and raw materials. (for example, if the firm request has sent very late to plant, the material could be transferred by plane, which increases the transfer price)
- Part Suppliers: They serve many competing OEMs, so there is a high risk, that they can leverage that information to support one customer against others, supplier can share the request data of one OEM to a second one in order to get advantage by that second one. This is a great trust issue between the company and the supplier.
- Infrequent large orders from consuming organizations force manufactures to maintain surplus capacity or excess finished goods inventory, which are very expensive solutions, to ensure responsive customer service.

Inventory Level:

Main problems of inventory level process:

Because of the deviated demand figures, capacities are used for unnecessary and also unplanned buffer stocks, which we call deviation stocks. When those stocks are understood that they are excessive from the real demand, the demand quantity has changed immediately by customers from the system, however because the production has already done (because they also entered an earlier requirement date), those stock will be free stocks which we can lose the ownership forever if we don't have any allocation methodology for free stocks. Free stocks are unpegged stocks, and without any stock ownership methodology for free stocks they can be unseen from customers and will became aged stocks, which makes them unprofitable due to interest rates or useless due to regulation changes.

The bullwhip effect occurs when the demand order variabilities in the supply chain are amplified as they moved up the supply chain. Distorted information from one end of a supply chain to the other can lead to inefficiencies. (excessive inventory investment, poor customer service, lost revenues, misguided capacity plans, inactive transportation, and missed production schedules) Companies can effectively counteract the bullwhip effect by thoroughly understanding its underlying causes. Industry leaders are implementing innovative strategies that pose new challenges:

- integrating new information systems,
- defining new organizational relationships, and
- Implementing new incentive and measurement systems.

Another problem is that the customers can make a product inactive in their market zone, without giving any information to central supply chain. Without noticing that they won't sell those products, inventory planning team could calculate a long term planning for those products also.

Product life cycle status is a really important topic about material specification for inventory planning. (Table 7)

Firstly product is introduced to the system then bill of material list is emerged. After trial production period, validated products are transferred to Active status, which is required for mass production.

Production management team determines the phase out models according to regulation situations. Those models are transferred to inactive status until stocks are finished in whole system warehouses. The models without any production and stock are called obsolete models.

Material	T1: Introduction	T2: BOM	T3: Trial Production	T4: Active	T5 : Inactive	T6: Obsolete
7790220235						X
7786588611					X	
7768320255					X	
7306730003			X			
7182481200				X		
7121982300						X
7118741400				X		
7107591500	X					
7103441300		X				
7101141400		X				
7100141500	X					

Table 7: Product Life Cycle Status

Since the inventory level is the level that makes the problems visible as an improvement reference point only if we measure the results with KPIs, otherwise it will just stay as an iceberg.

In Table 8 and Table 9, KPI results are followed up to define the process' weak points , that needs to be improved . The results are announced to every subsidiaries and direct sales groups.

Demand quantities, total supply and actual shipments data are used for calculations for Fixed Order Fulfillment, Shipment Plan Consistency.

Customer	Plant	Demand Qty	Total Supply	Actual Shipments	Fixed Order Fullfilment	Target	Shipment Plan Consistancy	Target
Beko PLC	REF	62.658	59.588	54.024	86,2%	93,0%	95,1%	90,0%
	DWA	28.731	28.052	26.816	93,3%	93,0%	96,6%	90,0%
	WMA	55.926	44.372	41.569	74,3%	93,0%	97,3%	90,0%
	DRY	49.457	28.972	27.736	56,1%	93,0%	85,6%	90,0%
	CKG	58.588	54.976	53.740	91,7%	93,0%	93,8%	90,0%
Beko Deutschland GmbH	REF	8.419	8.080	7.205	85,6%	93,0%	96,0%	90,0%
	DWA	10.788	10.687	9.582	88,8%	93,0%	99,1%	90,0%
	WMA	20.474	14.249	19.728	96,4%	93,0%	96,8%	90,0%
	DRY	13.321	9.997	13.186	99,0%	93,0%	75,0%	90,0%
	CKG	24.650	21.133	20.258	82,2%	93,0%	85,7%	90,0%
Mıddle Africa	REF	8.219	9.382	8.084	98,4%	93,0%	88,6%	90,0%
	DWA	9.306	9.697	8.822	94,8%	93,0%	94,2%	90,0%
	WMA	12.596	13.697	12.461	98,9%	93,0%	98,7%	90,0%
	DRY	2.350	2.214	2.215	94,3%	93,0%	94,2%	90,0%
	CKG	1.560	547	1.425	91,3%	93,0%	85,1%	90,0%
South America	REF	12.914	12.476	12.779	99,0%	93,0%	96,6%	90,0%
	DWA	2.745	3.100	2.610	95,1%	93,0%	92,9%	90,0%
	WMA	4.548	4.323	3.448	75,8%	93,0%	95,1%	90,0%
	DRY	4.400	4.214	4.265	96,9%	93,0%	95,8%	90,0%
	CKG	2.500	1.647	1.258	50,3%	93,0%	86,9%	90,0%

Table 8: Customer Fulfillment

Deviation Stock Ratio, Inventory days and Aged stock ratio KPIs are the inventory results of the whole process. As it seems on Table 9, figures are all calculated material based. For each material detailed analyses has been made in order to indicate a target point and also to achieve the targets throughout the year.

Customer	Material	Demand Qty	Total Supply	Average Sales	Actual Sales	Total Stock Qty	Aged Stock Qty	Inventory Days	Deviation Stock Ratio
Beko PLC	7100541100	40	48	13	24	24	17	40,77	20%
	7100642400	930	930	650	800	130	27	42,92	14%
	7100841100	1.020	809	720	570	239	44	33,71	30%
	7182681400	445	260	208	260	0	132	37,50	0%
	7101141300	1.250	1.127	830	1.000	127	49	40,73	11%
Beko Deutschland GmbH	7755582951	625	600	480	507	93	26	37,50	16%
	7185281300	1.750	1.750	1.890	1.750	0	533	27,78	0%
	7100841100	750	520	650	400	120	23	24,00	23%
	7182681400	1.375	1.030	1.200	1.030	0	313	25,75	0%
	7178981300	930	800	690	800	0	35	34,78	0%
Mıddle Africa	7101341100	1.510	1.724	1.380	1.000	724	525	37,48	34%
	7103241500	240	250	300	170	80	11	25,00	29%
	7100841100	375	407	570	350	57	18	21,42	7%
	7104241500	300	282	600	200	82	86	14,10	29%
	7107342100	250	250	300	207	43	76	25,00	17%
South America	7176781400	750	720	900	507	213	31	24,00	30%
	7100841100	24	27	16	0	27	1	50,63	100%
	7181581500	558	530	650	480	50	161	24,46	9%
	7291548092	612	600	720	600	0	26	25,00	0%
	7615349042	267	300	208	280	20	91	43,27	0%

Table 9: Inventory Management Activities

Deviation Stock Ratio KPI results are shown in Table 10. To hit the target, it is quite critical to minimize deviation of the sales forecast vs. actual loading quantities. Beko Deutschland GmbH is the only subsidiary that hit the target. The other figures are also between upper and lower limits.

SALES ORGANISATION	2013 ACTUAL	2014 ACTUAL	2015 TARGET	UPPER LIMIT	LOWER LIMIT	JANUARY'15	FEBRUARY'15	ROLLING	SCORE
Beko PLC	16,8%	15,0%	13,5%	11,5%	16,9%	7,6%	6,2%	13,7%	93,3
Central Africa	30,2%	40,7%	30,0%	25,5%	37,5%	40,3%	33,9%	36,6%	12,0
Southern America	34,7%	24,6%	22,1%	18,8%	27,6%	0,9%	81,8%	26,4%	22,1
Beko Deutschland GmbH	23,3%	14,0%	12,6%	10,7%	15,8%	7,7%	8,7%	12,2%	109,5

Table 10: Current Scores of Deviation Stock Ratio

This KPI has been calculated with the formula, below. Total Supply is the total quantity of end of last month stocks and current month productions. M0 Supply request means the first requested quantity within the current month for the current month. Every 3rd working day of every month customers send us their revised current month requests, which are called as M0 Supply request. Actual shipments are the actual sales of the end of the month.

$$\text{Aged Stock Ratio} = \frac{\textit{Aged stock}\ \text{value}}{\textit{Cost of}\ \text{all}\ \textit{goods sold}\ \text{(consolidated)}}$$

$$DSR = \frac{[min(S_m\,;\, x_{m0}) - TS - x_m]}{[min\,(S_m\,;\, x_{m0}) - TS]}$$

S_m «Total Supply»

x_{m0} «M0 Supply Request»

x_m «Actual Shipments»

TS «Target Stock»

DSR «Deviation Stock Ratio»

Aged stocks are also the result of a forecast deviation, of leak usage of capacity uninformed product life cycle data, not followed regulation changes.

Inventory day is the main result of whole supply chain inventory level. For every team this final result shows the big picture of the company's unused money in warehouses.

• For monitoring monthly performance, monthly inventory days are calculated and reported by central supply chain.

$$\textit{Inventory Days for a specific mont} = \frac{\textit{Average inventory value carried within the month}}{\textit{Cost of goods sold within the month}} * 30$$

• "Rolling average inventory days" are taken into account, whereas;

$$\textit{Rolling average inventory days} = \frac{\textit{Average inventory value carried within the year}}{\textit{Annual cost of goods sold average}} * 30$$

2.3.3 Proposed System and VMI Approach

In the current system, confidentiality is satisfied manually and abuse of any user endangers the confidentiality of the data. Whole sales figures collected from each subsidiaries and direct sales markets individually. In order to avoid misuse of the production and a better theoretical inventory planning approach, it is really critical to get correct/relevant figures. Now there is no limitation and each customer can access others forecasts and order details. To be on safe side and avoid nonfulfillment, customers usually add buffer stock to their forecast figures. Those cause deviation and keeping stocks/production for one customer also loss of sales for other one which really require stock/production at that time. Also these stocks, which has the actual production requirement date is later than it was submitted into the system, can be aged and maybe due to product life cycle limitations or energy regulation changes, there might be no possible market to sell the goods to end customers. If sales markets have limited access and central supply chain of ARC can collect whole information from system, whole process will be under control, the negative manual effects on operations will be decreased and that will help the capacity usage to utilize the whole network, inventory costs will be decreased and customer requirements will be supplied on time as required.

Confidentiality is another risky part for supplier side. Suppliers can work with two competitors. Communications with suppliers are needed to be limited because of the possible data leakages. It also affects the financial parts of the supply chain processes. The agreed rates and production costs are also needed to be confidential.

End-to-end collaborative supply chain planning is essential however is mainly blocked by the deficient information which is being shared between different actors that are focusing on their internal business activities and resulting in a distortion on information flow which is also known as the bullwhip effect. To solve the problem of deficient information, it needs to be understood clearly what creates this bullwhip effect and trust issues, which are also need to be solved, with customers and suppliers as well in order to minimize the risk of data leakage.

The bullwhip effect can be explained as an occurrence detected by the supply chain where orders sent to the manufacturer and supplier create larger variance then the sales to the end customer. These irregular orders in the lower part of the supply chain develop to be more distinct higher up in the supply chain. This variance interrupt the smoothness of the supply chain process as each link in the supply chain over or underestimates the product demand resulting in exaggerated fluctuations.

In the proposed system, VMI approach will be implemented according to ARC use case. Potentially VMI offers sources of bullwhip reduction. Mainly, there will be an elimination of information flow deficiencies. It is obvious that better sight and understanding of both information flow and material flow should lead to better business performance. This approach would demonstrate an effective information technology to assess all supply chain parties' true needs. New algorithm will lock the confidential data and these data can be seen and used only by the program itself when optimizing the plans. As a result, effect of misuse of any user can be eliminated from the system. Furthermore, customers safely will share their purchase capacities, inventory levels and free balances in their accounts. With this proposed system, we would observe the effect and the adaptation of VMI in environments with different levels of demand variability with limited production capacity in a complex network model.

2.4 Summary of functional and security requirements

2.4.1 Functional and security requirements for the aeronautic case

In D24.1, all input data that supply chain partners have to put into the Cloud Planning System, as well as all output services received in return by the System have been defined in detail.

In general, two main business-optimizing services have to be guaranteed in the aero engine overhaul supply chain: collaborative demand forecasting, and collaborative planning and scheduling of the overhaul activities. The first one allows the MRO service provider to obtain demand forecasts from all customers based on on-condition engine status observations, reducing so overall costs thanks to a more accurate capacity planning. The collaborative planning and scheduling, instead, guarantees better supply chain performance since an ideal receipt point for each engine can be computed (this is a way to decrease costs and TAT).

Here, a general view on the main CPS functional requirements is provided:

1. *Monitor the engines working status.*
   Airlines and air forces can have performances of their engines constantly measured with the aim to know the status of the fleet and to adopt the right strategic behaviour; while MRO service providers can track engines performance in order to forecast the time of future overhaul visits, smoothing so the the peaks and troughs of its service capacity.
2. *Integrate the service plan of the MRO service provider.*
   The MRO service provider, putting into the System its service data in terms of human resources availability and time necessary to perform activities, can have conveniently managed overhaul services through an integrated prospective (it considers both the demand and supply side).
3. *Integrate the delivery plan of the parts' supplier.*
   The parts' supplier shares with the CPS data about times to deliver each engine module to the MRO plant. Such information is useful for the service provider since it makes it possible to define a good plan to manage the spare parts inventory and to avoid blocks during the execution of overhaul activities (all spare parts will be always available when they are requested).
4. *Compute forecast on overhaul services demand.*
   Such functional requirement is linked to the first one: the monitoring of the engines working status. Thus, the knowledge of forecasts about the overhaul services demand allows MRO service provider to define its service plans in advance, avoiding periods of over and under capacity.
5. *Plan overhaul activities for the next 3 (or 6) months.*
   The MRO service provider can have its overhaul activities planned: for example, it will be possible to know the best day in which an engine has to be shipped, or which human resources will be necessary to perform activities, as well as the time necessary to satisfy the customer demand.
6. *Compute penalties for delays in the overhaul process.*
   The MRO service provider can have knowledge of the penalties related to possible delays in the customer demand satisfaction. Hence, it will be able to organize its service plan adopting a priority-based strategy.

7. *Compute parts needs and propose purchasing orders.*

   The continuous monitoring of the service provider inventory (in terms of number of items in the warehouse) makes it possible to define which parts are needed (in function of customer demand) and to highlight the necessary purchasing orders.

8. *Compute forecast on parts requests.*
   The CPS provides to parts suppliers forecasts on which and when parts will be required by a certain MRO service provider. Such information is important for suppliers since, in this way, they can estimate in advance the necessary resources to face the service provider demand.

9. *Plan production activities for parts' supplier.*
   Parts suppliers can have their production activities planned: for example, it will be possible to know the period in which parts will be required, or which human resources will be necessary to perform activities, as well as the time necessary to satisfy the customer demand.

10. *Simulate demand satisfaction capability both for MRO service providers and for parts' supplier*
    The CPS allows to simulate the MRO service provider and parts supplier capability in the demand satisfaction. In such way, they will be able to know their performances in function of changes in some parameters values, such as the number of human and instrumental resources.

2.4.2 Data model for the aeronautic case

In order to provide the above mentioned functions to the user community, the following data model is provided. It is composed of eight tables collecting data coming from airlines/air forces, MRO service providers and parts' suppliers; in particular, data will be organized in the following three different areas:

- Fleet monitoring;
- MRO service planning;
- Inventory management.

The required data is usually stored in organizational database, for example the structure of engines is available in Configuration Management systems, a standard aeronautic information system through which the lifecycle of each serial number is managed. Similarly, data about service plans is stored in Enterprise Resource Planning systems, while data about inventory are available in specific inventory databases.

Overcoming the segregation of these data sources, even in each organization, is the first and perhaps most relevant benefit of the Cloud Planning System, as managing them in an integrated framework will free new business value for all involved actors.

Here, it is important to underline that all data reported in the tables are entered as an example.

2.4.2.1 Fleet monitoring

In the Table 11 and Table 12, data about engines and engine modules are collected. These two tables aim to compare the threshold working conditions of each engine model (Table 11) and of engine modules (Table 12), stated by the engine producer, with the current usage conditions stated by the engine owner. For this reason parameters frequently used by MRO service providers to monitor the fleet and carry out on-condition maintenance are considered here.

In particular, in the Table 11, the following columns are characterized:

- *Engine Serial Number*: is a unique code/number assigned for identification of a specific engine.
- *Engine Part Number:* is an identifier of a particular engine model.
- *Flight Hours limit:* flight hours concern the amount of time a particular engine spends in the air from the moment the aircraft leaves the runway to the moment it touches down on the same or a different runway. Here, such limit value is given by the engine producer according to technical valuations performed at the beginning of the engine life.
- *Flight cycles limit:* flight cycles are the amount of take offs and landings that a particular engine performs. Here, such limit value is given by the engine producer according to technical valuations performed at the beginning of the engine life.
- *Time Between Shop Visits:* that is the expected time (in flight hours) between two different overhaul events on the engine, given by the engine producer.
- *Flight Hours Since New:* the total number of flight hours performed by the engine. Such value is usually updated every ten days and depends on the engine usage.
- *Flight Cycles Since New:* the total number of flight cycles performed by the engine. Such value is usually updated every ten days and depends on the engine usage.
- *Time Since Last Shop Visit:* real time (in flight hours) since last overhaul event on the engine.

Similarly, in the Table 12 it is possible to collect the following data:

- *Module Serial Number:* is a unique code/number assigned for identification of a specific engine module.
- *Module Part Number*: is an identifier of a particular module type of the engine (seven different module types are here mentioned).
- *Engine Serial Number:* is a unique code/number assigned for identification of a specific engine (this data is necessary to associate a module serial number with an engine serial number).
- *Production Date:* it is the date in which the engine module was built. Such data is useful to monitor the necessity of future updates on the module.
- *Flight Hours limit:* flight hours are the amount of time a particular engine module spends in the air from the moment the aircraft leaves the runway to the moment it touches down on the same or a different runway. Here, such limit value is given by the engine producer according to technical valuations performed at the beginning of the engine module life.
- *Flight Cycles limit:* flight cycles are the amount of take offs and landings that a particular engine module performs. Here, such limit value is given by the engine producer according to technical valuations performed at the beginning of the engine module life.
- *Time Between Shop Visits:* that is the expected time (in flight hours) between two different overhaul events on the engine module, given by the engine producer.

- *Flight Hours Since New:* the total number of flight hours performed by the engine module. Such value is usually updated every ten days and depends on the engine usage.
- *Flight Cycles Since New:* the total number of flight cycles performed by the engine module. Such value is usually updated every ten days and depends on the engine usage.
- *Time Since Last Repair:* real time (in flight hours) since last repair event on the engine module.

Thanks to all this data it is possible to monitor the usage of the engines and modules in order to plan overhaul activities to be performed into the MRO plant, with a certain precision. Maintenance activities on the other hand cannot be planned in advance since they depend on unexpected events.

ENGINE							
Engine SN	Engine PN	FH limit	FC limit	TBSV (flight hours)	FHSN	FCSN	TSLSV (flight hours)
1	x	39,200	28,000	5,500	18,500	13,215	5,500
2	x	39,200	28,000	5,500	20,000	14,286	5,500
3	x	39,200	28,000	5,500	15,500	11,071	5,000
4	y	36,200	25,000	5,000	14,000	9,669	4,250
5	y	36,200	25,000	5,000	4,800	3,315	4,800
6	y	36,200	25,000	5,000	13,000	8,978	6,200
7	z	42,000	31,000	6,500	21,200	15,648	5,270
8	z	42,000	31,000	6,500	19,000	14,024	6,000
9	z	42,000	31,000	6,500	6,450	4,800	6,250

Table 11. Engine parameters

MODULE									
Module SN	Module PN	Engine SN	Production Date	FH limit	FC limit	TBSV (flight hours)	FHSN	FCSN	TSLSV (flight hours)
1	Fan	1	03-2000	39,000	27,500				
2	Low pressure compressor	1	02-2000	28,000	20,000				
3	High pressure compressor	1	02-2000	28,000	20,000				
4	Combustor	1	04-2000						
5	High pressure turbine	1	03-2000	28,000	20,000				
6	Low pressure turbine	1	02-2000	35,000	25,000				
7	Accessory drive	1	04-2000						
8	Fan	2	05-2001	38,000	26,000				
9	Low pressure compressor	2	06-2001	30,000	23,000				
10	High pressure compressor	2	05-2001	30,000	23,000				
11	Combustor	2	07-2001						
12	High pressure turbine	2	05-2001	30,000	23,000				
13	Low pressure turbine	2	06-2001	36,000	25,000				
14	Accessory drive	2	06-2001						
15	Fan	3	05-2004	37,000	27,000				

MODULE									
Module SN	**Module PN**	**Engine SN**	**Production Date**	**FH limit**	**FC limit**	**TBSV (flight hours)**	**FHSN**	**FCSN**	**TSLSV (flight hours)**
16	Low pressure compressor	3	07-2004	28,000	22,000				
17	High pressure compressor	3	02-2004	28,000	22,000				
18	Combustor	3	03-2004						
19	High pressure turbine	3	07-2004	28,000	22,000				
20	Low pressure turbine	3	05-2004	34,000	25,000				
21	Accessory drive	3	04-2004						

Table 12. Module parameters

2.4.2.2 MRO service planning

The following tables contain data specifically provided by the MRO service provider.

In the Table 13, the aero engines owners are declared (in the columns named *Customer ID* and *Customer Name*), and for each of them, one or more *Engine Serial Number* is associated. Furthermore, it is useful to take note of the *Contract Type* and the *Contract Life Span* of customers in order to define a good activities plan on short and long period (in such way, it will be possible to order the service activities as a function of customer priorities).

CUSTOMER				
Customer ID	**Customer Name**	**Engine SN**	**Contract Type**	**Contract Life Span**
1	A	1	Power by hours	8 years
2	B	2	Power by hours	3 years
3	C	3	Power by hours	5 years
4	E	4	Power by hours	10 years
5	F	5	Power by hours	10 years
6	G	6	Power by hours	8 years
7	H	7	Power by hours	5 years
8	I	8	Power by hours	5 years
9	L	9	Power by hours	8 years

Table 13. Customers

Hence, in the Table 14, for each *Engine Serial Number* the following data are indicated:

- *Date in*: the date in which the engine was ready to be overhauled;
- *Date in plan:* the date in which the engine overhauling was planned;
- *Date out*: the date in which the engine overhauling was completed;
- *Date out plan: t*he date in which the engine overhauling was planned to be completed;
- *ID of activities performed: t*he activities ID that were carried out on the engine after overhauling.

All this information is useful to evaluate MRO service provider performance according to an historical perspective. Furthermore, activities ID are required to keep track of specific activities performed on the engine in order to have a picture on the future ones.

SERVICE STATUS					
Engine SN	**Date in**	**Date in plan**	**Date out**	**Date out plan**	**ID of activities performed**
1	2-02-2005	12-02-2005	4-05-2005	8-05-2005	1,2,3,4,5,8,9,10,11,12
2	10-04-2006	30-03-2006	12-09-2006	27-08-2006	1,2,3,4,5,6,8,9,10,11,12
3	25-05-2008	27-05-2008	4-09-2008	5-09-2008	1,2,3,4,5,7,8,9,10,11,12
4	22-02-2005	28-02-2005	18-05-2005	16-05-2005	1,2,3,4,5,8,9,10,11,12
5	12-06-2005	5-06-2005	28-09-2005	15-09-2005	1,2,3,4,5,8,9,10,11,12
6	9-09-2006	9-09-2006	23-12-2006	10-12-2006	1,2,3,4,5,8,9,10,11,12
7	15-01-2008	19-01-2008	17-04-2008	17-04-2008	1,2,3,4,5,7,8,9,10,11,12
8	13-09-2006	21-09-2006	15-12-2006	21-09-2006	1,2,3,4,5,6,8,9,10,11,12
9	30-01-2008	21-01-2008	6-05-2008	30-04-2008	1,2,3,4,5,6,8,9,10,11,12

Table 14. Service status

In the Table 15 and Table 16, respectively, a general view of overhaul activities performed and of available human resources within the MRO plant are provided. It is considered that all activities are performed at the engine level, with the exception of *repair, wasting* and *storage in warehouse B* activities for which we speak at module level.

In particular, in Table 15 each overhaul activity (see columns named *Activity ID* and *Activities Name*) is characterized with the *Turn Around Time* (in days) and with the number of human resources necessary to perform it. Such table is linked to the Table 16 in which information about human resources are defined. Here, it is important to understand that not only different skilled human resources are necessary to overhaul different engine models, but also for each activity of an engine model it is possible that different resources are required. Hence, in the Table 16, for each human resource (mentioned in the column named *Resource ID* and *Resource Name)* is specified if he/she is specialized (on specific activity on the specific engine model) or generic technician (*Specialized/Generic*), and the overhaul activity associated with a specific engine model for which he/she has competences (*Engine PN* and *ID Activities)*.

OVERHAUL ACTIVITIES			
Activity ID	**Activity Name**	**TAT (days)**	**# Human Resources**
1	Store in warehouse A	1	2
2	Disassemble	1÷4	3
3	Clean	1÷3	2
4	Run NDT	2÷5	3
5	Inspect	1÷4	2
6	Repair	Variable	Variable
7	Waste	1	1
8	Store in warehouse B	1	1
9	Kit	17÷55	2
10	Assemble	3÷8	3
11	Run Bench Test	5	2
12	Certify	2	2

Table 15. Overhaul activities

Table 16. Human resources

HUMAN RESOURCES				
Resource ID	**Resource Name**	**Specialized/Generic**	**Engine PN**	**ID Activities**
1	A	Generic	x-y-z	1
2	B	Generic	x-y-z	1
3	C	Generic	x-y-z	3
4	D	Generic	x-y-z	3
5	E	Generic	x-y-z	7
6	F	Generic	x-y-z	7
7	G	Generic	x-y-z	8
8	H	Generic	x-y-z	8
9	I	Specialized	x	2
10	L	Specialized	x	2
11	M	Specialized	y	2
12	N	Specialized	x	4
13	O	Specialized	z	4
14	P	Specialized	y	4
15	Q	Specialized	y	5
16	R	Specialized	z	5
17	S	Specialized	x	6
18	T	Specialized	x	6
19	U	Specialized	y	6
20	V	Specialized	z	9
21	Z	Specialized	z	9
22	X	Specialized	x	10
23	Y	Specialized	y	10
24	W	Specialized	z	10
25	A'	Specialized	y	11
26	B'	Specialized	z	11
27	C'	Specialized	x	12

2.4.2.3 Inventory management

The following Table 17 and Table 18 are designed with the aim to manage the MRO service provider inventory and to properly plan the purchasing orders for modules' suppliers.

In the Table 17, for each engine module (*Module Serial Number*), the following parameters are periodically monitored and tracked:

- *# Items in Warehouse: that is t*he number of items available in the warehouse;
- *Threshold for Purchasing*: that is the threshold level triggering the purchasing procedure;
- *Batch Size*: that is the minimum number of items to be purchased in a single purchasing order.

In the table 10, instead, the modules' suppliers are shown, and for each of them, the modules they provide (*Module Part Number*) and the time necessary to deliver them (*Lead Time*) are specified. This information is mandatory to plan the purchasing order.

Table 17. Inventory status

INVENTORY STATUS				
Module SN	**Module PN**	**# Items in Warehouse**	**Threshold for Purchasing**	**Batch Size**
1	Fan	2	2	4
2	Low pressure compressor	1	0	2
3	High pressure compressor	1	0	2
4	Combustor module	0	1	3
5	High pressure turbine	2	2	3
6	Low pressure turbine	3	1	2
7	Accessory drive module	0	0	2
8	Fan	3	2	4
9	Low pressure compressor	1	0	2
10	High pressure compressor	1	0	2
11	Combustor module	2	1	3
12	High pressure turbine	0	2	3
13	Low pressure turbine	0	1	2
14	Accessory drive module	2	0	2
15	Fan	3	2	4
16	Low pressure compressor	2	0	2
17	High pressure compressor	1	0	2
18	Combustor module	0	1	3
19	High pressure turbine	0	2	3
20	Low pressure turbine	2	1	2
21	Accessory drive module	1	0	2

Table 18. Supplier

SUPPLIER			
Supplier ID	**Supplier Name**	**Module PN**	**Lead Time (days between PO and delivery)**
1	A	Fan	5
2	B	Low pressure compressor	10
3	C	High pressure compressor	10
4	D	Combustor module	7
5	E	High pressure turbine	5
6	F	Low pressure turbine	5
7	G	Accessory drive module	10

The following figure illustrates the database scheme as an entity relationship model (ERM). The tables "*ENGINE*" and "*MODULE*" consists a composite primary key (short PK). This means that the unique identifier will be composed out of two or more columns. In the case of the "ENGINE"-table the primary key will be composed by the "*EngineSN*"- and "*EnginePN*"-columns. The interpretation of the relations, etc. can be found within the figure (see "Legend").

Figure 9: Relational data model.

Chapter 3 Models and algorithms for cloud-based supply chain collaboration

In this chapter formal models and algorithms are developed for the secure cloud-based implementation of the aerospace and the consumer goods collaborative supply chain management scenarios. Thereby, entirely new models where developed in some cases (Sections 3.1.2, 3.1.3.2, 3.1.3.3) in order to explicitly match the requirements of the industries introduced in the previous sections. In other sections (3.1.3.1, 0), existing models were adapted in order to meet the scenario requirements (references are provided accordingly where necessary) of the aeronautic and consumer goods industry.

In Section 3.1 a collaborative planning system for aircraft engine maintenance is developed, incorporating collaborative maintenance demand forecasting, spare parts management, arrival slot scheduling and capacity planning modules. While the respective models are developed independently in the subchapters, we also provide guidelines for the implementation of the entire system that consider the interdependencies of the individual modules. The actual secure cloud-based implementation will be the focus of the next deliverable D24.3.

In Section 0 a collaborative inventory planning system for the consumer goods scenario is developed. Therein, a model for collaborative forecasting of future demand is presented as well as a single-vendor multi-retailer vendor managed inventory model incorporating highly sensitive data from all parties involved. Again, an overview of a potential secure cloud-based implementation guideline is provided.

In Section 3.3 a short introduction to auctions and opportunities for supply chain coordination in the consumer goods industry is given.

Finally, Section 0 provides a conclusion and an outlook regarding future research activities, implementation and benefit assessment.

3.1 Secure cloud-based collaborative maintenance management in the aeronautic industry

3.1.1 Overview and introduction

In the past, technical service providers and contract manufacturers ensured cost-wise competitiveness through lean management and rigorous spend control. Nowadays, with the rise of cloud technologies and advanced cryptographic methods, new opportunities for cost savings are available through collaboration between supply chain partners. This has been highlighted in deliverable D24.1

As a wide range of parameters of technical equipment, e.g., vibrations levels and oil pressure/temperature of aircraft engines can be monitored online (for newer engine models), sophisticated predictive maintenance mechanisms can be employed in order to not only determine the ideal maintenance timing but also forecast the types of services that need to be carried out. E.g., depending on the parameters and lifetimes of individual parts or modules, one can predict if it should be repaired or replaced. Therefore, supply chain activities such as spare parts management or preparation of the service can be optimized.

In this section we specifically focus on a setting where aircraft engines are shipped to an MRO service provider, are processed there and afterwards shipped back to the customer. It should be noted, however, that the approach presented in this deliverable can be adapted to maintenance operations in other industries. We propose the implementation of a collaborative planning system in order to capture benefits through advanced information as described above. Naturally, as different parties are involved in the collaborative planning system, a potential cloud-based implementation needs to be secure, i.e., private data needed for the various optimization models must be protected.

Currently, the airlines send in their engines after a specific workload; the service provider has little or no information regarding the arrival time and lacks any ability to influence it. Additionally, the service provider has no information regarding the engine condition; therefore, both capacity and spare parts requirements are highly uncertain, leading to high capacity and inventory levels and long turnaround times. We show different opportunities for cloud-based collaborative planning of MRO supply chain and production activities that exhibit a large potential to remedy these problems. First, we explain collaborative maintenance demand forecasting methods in section 3.1.2. In the collaborative maintenance planning section we consider optimized spare parts management using collaborative forecasting demand information in Subsection 3.1.3.1. In Subsection 3.1.3.2 we establish scheduling rules for collaborative scheduling of engine arrival time slots. In Subsection 3.1.3.3 we define an optimization mechanism to optimally plan capacity in the maintenance overhaul production network of the MRO. In this section the results of all previous sections are used in order to reduce total cost for the MRO. As these costs are composed of capacity costs and penalty costs for not meeting contractually defined turnaround times, the objectives of all preceding sections are such that either capacity of turnaround time can be reduced. Finally, a conclusion is provided in Section 3.1.4, also summarizing the implementation details from the previous subchapters in order to provide a holistic overview.

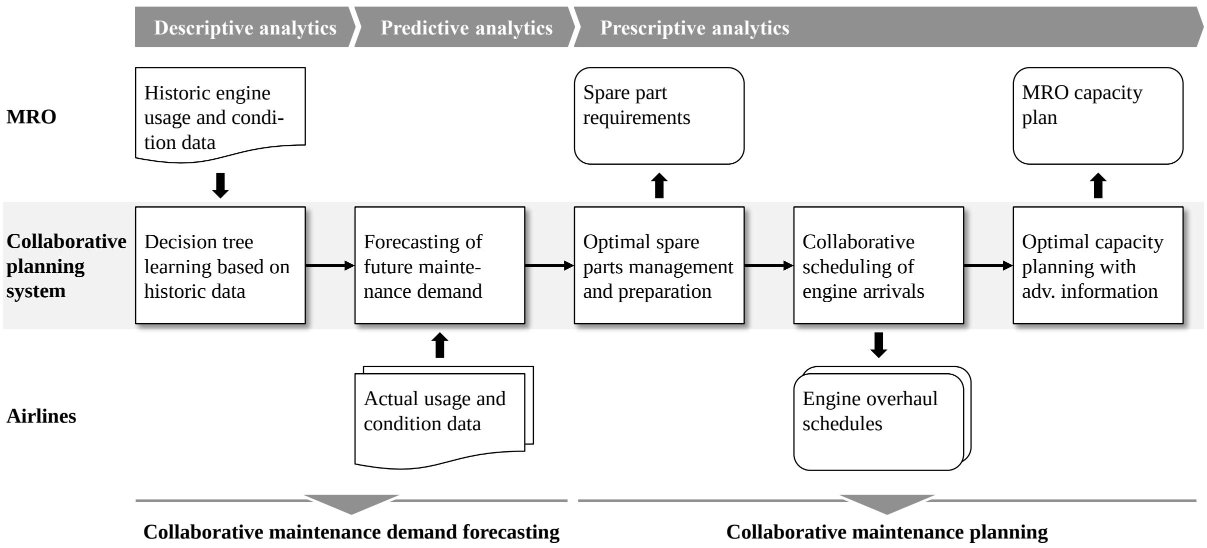


Figure 10: Collaborative planning system for aircraft engine maintenance.

Figure 10 provides an overview of the collaborative planning system and is used throughout this section in order to support the reader in understanding the overall approach and dependencies between the collaborative planning modules. In the following subsections, we detail the individual modules and provide a description of their functionality and formal logic.

3.1.2 Collaborative maintenance demand forecasting[5]

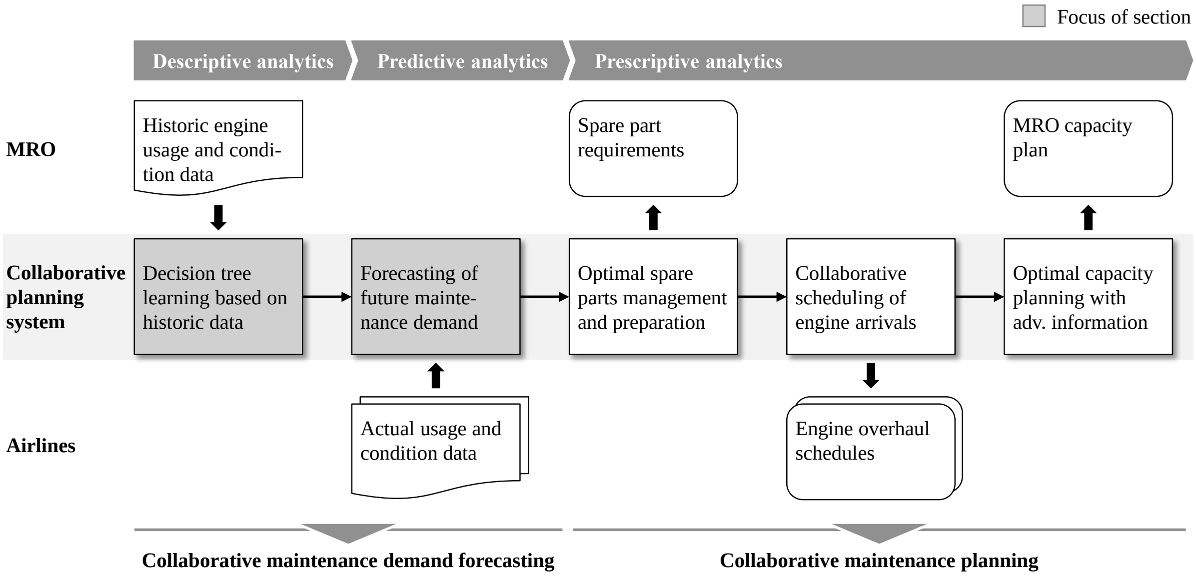


Figure 11: Focus on collaborative maintenance demand forecasting

In this chapter we describe three models for secure collaborative forecasting using different supervised machine learning techniques. We show the application of these machine learning concepts in the context of a MRO[6] for the aerospace industry. Again, we want to emphasize that these tools can easily be adopted to the specific requirements of other industries. The combination with privacy preserving techniques favors the implementation as part of a cloud based platform for secure supply chain collaboration services. The output of the forecasting modules can then be used in subsequent planning operations as depicted in Figure 11.

As highlighted previously, the MRO faces high uncertainty with respect to the order arrival and the condition of the engines. The status quo of the maintenance operations can be described by two main characteristics: first, there is no collaboration between customers and MRO with respect to sharing information about upcoming service demand. Second, the overhauls are scheduled based on thresholds for the usage parameters of the engines (e.g. flight hours and flight cycles). These parameters are only indirectly related to the actual status of an engine. According to our industrial partners this will change in the near future since newer engine models allow tracking of detailed sensor data during flight operations which then can be used to predict overhaul demand.

In this chapter we take this technological change from hard-time towards condition-based maintenance into consideration. In subchapter 3.1.2.2 we show how secure collaboration can be achieved using only the usage parameters of each engine. In subchapter 3.1.2.3 we develop a model that uses the additional sensor data and can be used to predict the type of service that will be necessary and the demand for spare parts. In subchapter 3.1.2.4 we show how the sensor data can be used to predict overhaul demand. What all forecasting models have in common is that they use the distributed data from customers, service provider (MRO) and spare part suppliers as depicted in Figure 12.

---

[5] Main author: Fabian Taigel (UWUERZ)

[6] MRO = maintenance, repair and overhaul service provider

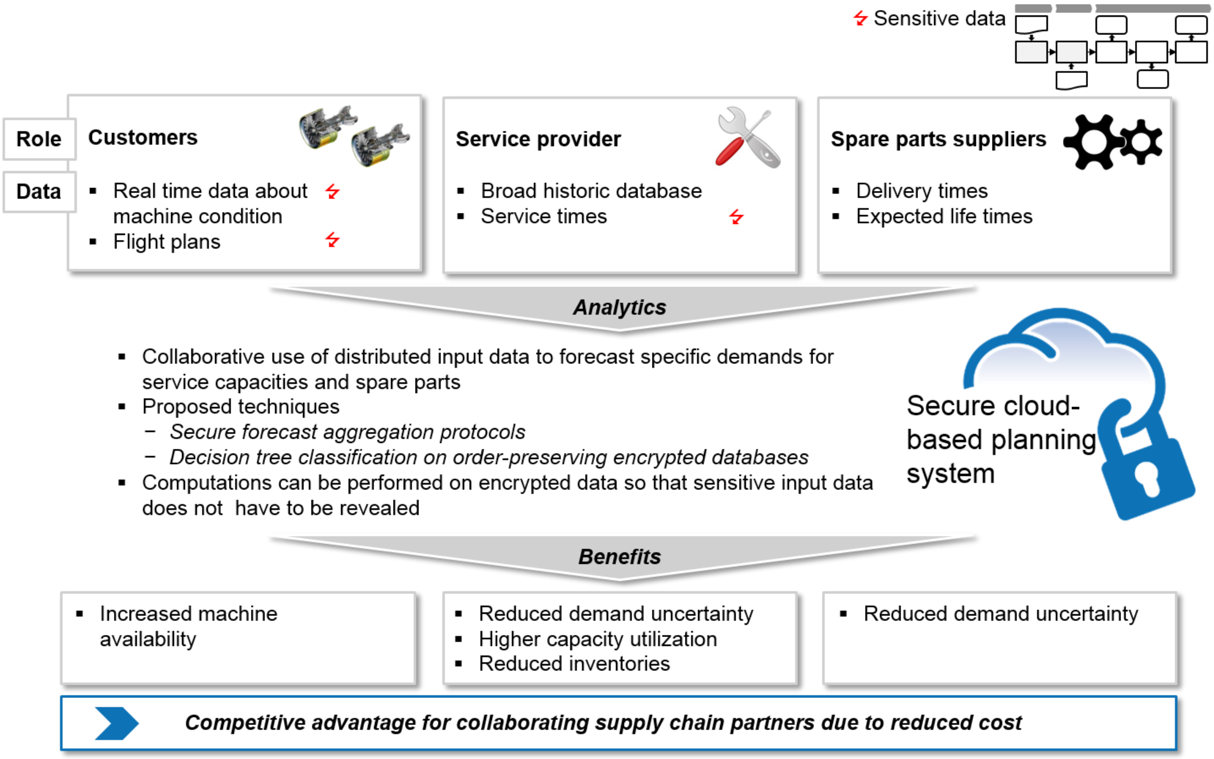


Figure 12: Collaborative forecasting for maintenance management.

The structure of this chapter is as follows: In subchapter 3.1.2.1 we describe the basic concepts of machine learning and its application for supply chain collaboration. The subsequent three subchapters are all divided in four parts: in the first we explain the specific task for supply chain planning and give a detailed description how this task can be tackled using a specific supervised machine learning model. The second part contains algorithms for the implementation. In the third part we analyze the security requirements for the used data and in part four we summarize and contextualize the results.

3.1.2.1 Supply chain collaboration in the context of machine learning

In order to see how supervised machine learning can be employed to support collaboration we first take a look at its basic concept. In supervised machine learning concepts the goal is to find a mapping from a set of attributes $\mathbf{x}$ to a response variable y by assuming that the value of y can be predicted from the values of the other attributes and that this interrelation can be found by analyzing a (preferably large) set of already classified (i.e. labelled) learning data. A classified or labelled instance is a combination of a set of describing attributes and the corresponding response variable. In the context of the MRO the set of attributes $\mathbf{x}$ are various parameters that track specific data for a certain engine and y then represents in some form the status of an engine.

Machine learning can be split up in two phases: first, the learning phase where a learning dataset is used to fit a function to this data and second the prediction phase where the estimated function is applied to new data. For supply chain collaboration this separation is very useful since the data we use for learning and prediction is typically not centrally available. For instance, the MRO has a broad dataset of historic data from past overhaul activities which can be used in the learning phase whereas the airlines have the unclassified real-time data on which we apply the learned models in the second phase.

We now give a more formal description that will be used throughout the following subchapters. Given a learning dataset $\mathrm{D} = \{(y_j, \mathbf{x}_j)\}_{j=1}^{N}$ with N instances of pairs of response variables and set of attributes we assume that there is a unknown function f that explains the relation between attributes and response $y = f(\mathbf{x})$. Based on this assumption we estimate a function $\hat{f}_{\mathrm{D}}$ using the given labelled training set D. As in Murphy (2012, pp. 2–3) this function can be used to make predictions on unknown data by computing $\hat{y} = \hat{f}_{\mathrm{D}}(\mathbf{x})$. (The hat symbol here denotes an estimation.)

There exist various machine learning algorithms that can be used to estimate a function $\hat{f}_{\mathrm{D}}$. One distinctive feature is the form of the response variable y that can be either continuous (such as flight hours) or categorical (such as repair or replace). In maintenance operations we have both, so we have to choose the adequate machine learning techniques. In this part we will use two different forms of supervised machine learning: Linear regressions in subchapter 3.1.2.2 and decision tree learning in subchapters 3.1.2.3 and 3.1.2.4. In the linear regression setting the response variable is approximately continuous (e.g. representing flight hours) and the explaining attribute is the time since the last overhaul. Based on the experience of our industrial partners, we assume a linear relation

$$y(x) = \gamma x$$

and try to estimate the parameter γ. In the decision tree setting the response variable is categorical and we focus on the case with only two categories, i.e. $y \in \{1,2\}$ representing classes (such as repair or replace).[7] The explaining attributes are represented as k-dimensional numeric vectors (although the numbers might again stand for nominal categories, e.g. deviation: high, normal, low). The aim is to partition the k-dimensional space of the explaining attributes by axis parallel splits. I.e. in the learning phase we try to find regions defined by a sequence of splits of the attribute values that contain mostly instances with similar values of the response variable. These regions can be interpreted as the leaves of a tree as depicted in Figure 13.

[7] This is not a limitation since multiple binary decision trees can be used in multi-categorical scenarios as described in subchapter 3.1.2.4.

A leaf can be uniquely described by the set of set of attributes and their corresponding split values from the root of the tree to the leaf. This rather simple representation is the main

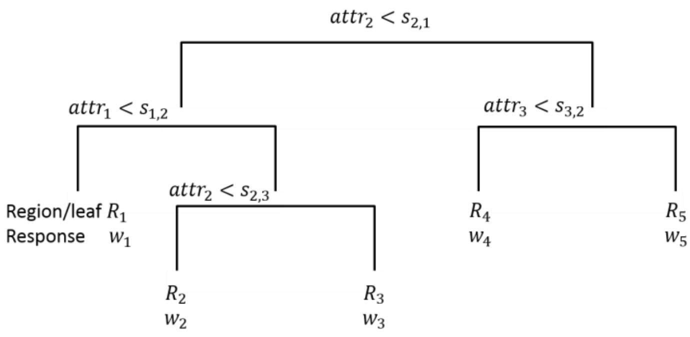


Figure 13 Representation of a partitioned attribute-space as decision tree

reason why decision tree learning is a promising concept for privacy preserving applications: The classification of unknown instances (i.e. the prediction) is done by comparison operations which can be efficiently performed on order-preserving encrypted databases.[8] Predicting a new instance is then done by sorting it into the correct leaf according to the splits and assigning the response value for this region.

As an example consider Figure 13, which depicts a possible result of a learning phase. In the learning phase the order of attributes and the attached split values were determined such that the obtained regions contain only (or at least mostly) instances with similar class attributes.[9] The left branch of each node is the branch where the inequality of the split is fulfilled. We have three attributes, so each leaf can be uniquely described by split values for (at most) each of these attributes. The split value for $attr_i$ on level j is denoted as $s_{i,j}$. For instance, leaf R_3 where $s_{2,3} \leq attr_2 < s_{2,1}$ and $attr_1 < s_{1,2}$.

The classification function can be written in the following form:

$$f(\mathbf{x}) = \sum_{l=1}^{L} w_l \mathrm{I}(\mathbf{x} \in R_l) = \sum_{l=1}^{L} w_l \phi(\mathbf{x}; \mathbf{v}_l)$$

The indicator function $\mathrm{I}(\mathbf{x} \in R_l)$ yields 1 if an instance $\mathbf{x}$ belongs to leaf R_l and 0 if $\boldsymbol{x}$ is not in R_l. By w_l we denote the attached response value for leaf l.The basis function $\phi(\mathbf{x}; \mathbf{v}_l)$ does the same, given the specific parameters of the tree representation denoted as $\mathbf{v}_l$ which encodes the route from the root to leaf l of a tree by giving each variable to split on with the respective threshold value. Hence, the function returns the response w_l

Before we further elaborate the specific models, we here give introduction to each subchapter to depict how linear regression and decision tree learning are applied in the MRO scenario. In subchapter 3.1.2.2 linear regression is applied to model the dependencies between time and flight hours and between time and flight cycles for a specific engine. With these models we can estimate the time when the engine requires overhaul due to a reached threshold for either flight hours or flight cycles. In this scenario we estimate the model parameters on the most recent usage data which requires privacy protection since it contains information about present utilization and planned operations (e.g. flight plans). Only the

[8] We refer to subchapter 3.1.2.3.1 for more details on the secure classification
[9] We refer to subchapter 3.1.2.3.2 for detailed algorithms for the decision tree learning phase

aggregated results from the individual regressions are to be provided to the MRO. This ensures the highest degree of privacy for the customers while giving the MRO much better information to plan its capacities.

In 3.1.2.3 we assume that the MRO has a broad database of historic data that was collected during overhaul operation for its various customers. In particular, the MRO obtains the information about the status of a certain component during its own disassembling and overhaul operations. The decision trees are trained on these historic databases. For prediction the relevant real-time data is generated and provided by the customers. We propose a concept how the classification of this new data can be computed without the need for revealing the sensitive real-time data to the MRO.

In 3.1.2.4 we discuss a scenario where also the historic condition monitoring data is kept private by each customer. The important labeling information (insights into the true status of an engine from hands-on analysis) are in possession of the MRO. Therefore it is necessary to train decision trees on the distributed data while keeping some of these data private. For prediction the relevant real-time data is again under the customers' control.

3.1.2.2 Demand forecasting without condition data

This chapter provides a basic model for secure collaborative forecasting of service demand without any sensor or on-condition data of engines. We offer a solution to forecast overhaul service demand that is triggered by usage parameters which reach specified thresholds. Usage parameters are meta information as flight hours and flight cycles that have no direct relationship with the actual status of an engine. This approach is suitable in case the MRO does not receive condition data because the data is not collected (e.g. for older engines) or not shared by the customer (e.g. due to technical reasons or privacy issues).

The structure of this section is as follows: first, we describe how the MRO's service demand can be predicted. For this, we propose an aggregation of the output from linear regression models on each customer's proprietary usage data. In the second part we describe why privacy preservation is an issue here and how these issues could be overcome by using additively homomorphic encrypted databases and secure multiparty computation.

3.1.2.2.1 Aggregated service demand forecasting

For capacity planning the MRO needs to know the number of engines of a certain model in specified time periods $d_{e,t}$ where $e \in \{1,...,E\}$ refers to the engine type and $t \in \{1,...,T\}$ to a time interval, e.g. month t of a planning period. The MRO serves A different customers (Airlines) each owning a number of engines of the types $e = 1,...,E$. The sum of the individual demands $d^a_{e,t}$ of customers $a \in \{1,\ldots,A\}$ determine the total demand the MRO has to serve in a certain period:

$$d_{e,t} = d^1_{e,t} + d^2_{e,t} + ... + d^A_{e,t}$$

Before determining this aggregate value, we have to estimate the individual demands. As input data we use the usage parameters for each engine. For each flight cycle the airlines record the flight hours and a timestamp when the flight cycle was completed. Hence, for each of these timestamps we know, how many cycles the engine has flown and how many flight hours it has accumulated. An example for an input data table is shown in Table 19. These tables can also contain future data since the flight plans are established some time in advance.

# cycles	time	accumulated flight hours
$y_1 = 1$	x_1	h_1
$\vdots$	$\vdots$	$\vdots$
$y_N = N$	x_N	h_N

Table 19: Usage parameters for one engine of one customer; all values are accumulations since the last overhaul

As we know from our industrial partners in the aerospace engine MRO business, additional condition monitoring data is currently not yet available.[10] One reason for this is that older engine generations are not equipped with sensors so that the data can't be tracked. The second reason is that the data is not available from the airlines because adequate data processing systems are not in place. In this situation, overhaul is due when the first usage parameter reaches a predefined threshold value. There are thresholds for the number of flight cycles, the number of flight hours and the lifetime as listed in Table 20.

# flight cycles:	Y_e
# flight hours:	H_e
Lifetime (in days):	L_e

Table 20: Thresholds for an engine type e

We assume a linear dependency between time and number of cycles respectively between time and the number of accumulated flight hours. This is based on the experience of our industrial partners. In the following steps we show the linear regression for the former case, however the latter works in an analogous way. The given learning dataset consists of the columns '#cycles' and 'time' from the input data table with N entries for the specific engine denoted as $\{(y_j, x_j)\}_{j=1}^{N}$. Formalized the linear dependency between time and the number of cycles is given by:

$$y_j(x_j) = \gamma x_j + \varepsilon_j,$$

where γ can be interpreted as the slope of a line that approximates the data point. ε_j is the residual error between the linear prediction and the true response. According to Murphy (2012) an estimate for γ that minimizes the sum of the squared residual is given by:

[10] We refer to this as condition data meaning any sensor data that allows to infer the current status/condition of an engine

$$\gamma = \frac{\sum_{j=1}^{N} x_j y_j}{\sum_{j=1}^{N} x_j^2} \tag{1}$$

Figure 14 shows the regression line for the numbers of cycles dependent on the time since the last overhaul.

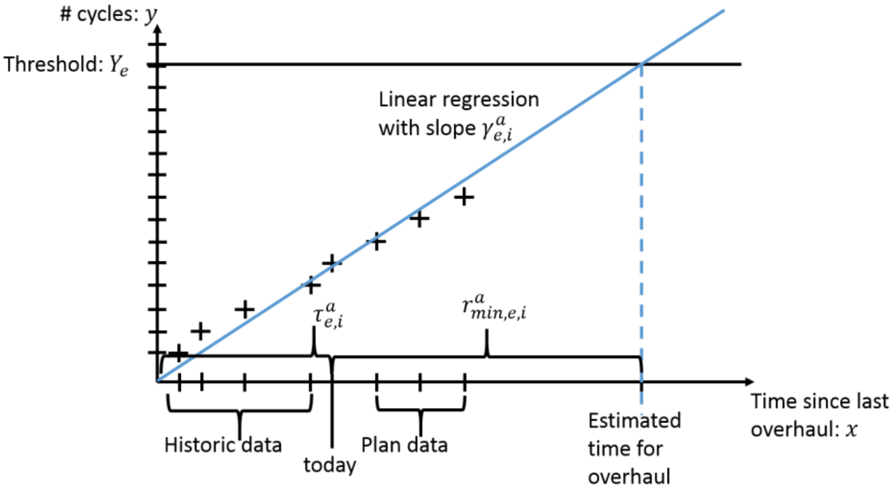


Figure 14: Linear regression model for the #cycles subject to time

Building these linear regression models for each engine of each customer for the number of cycles and for the number of accumulated flight hours gives us the parameters we need for the further computations. Table 21 provides an overview on those outputs. The indices are added to allow exact identification. For example $\gamma_{e,i}^a$ refers to the slope of the regression line for engine i of type e owned by customer a. $\gamma_{e,i}^a$ can be computed using equation (1). For the computation of $\zeta_{e,i}^a$ the y_i in equation (1) have to be replaced by h_i.

Slope of cycles/day:	$\gamma_{e,i}^a$
Slope of flight hours/day:	$\zeta_{e,i}^a$

Table 21: Output of linear regressions

The next step is to compute $r_{min,e,i}^a$, the time until the next overhaul is due for a certain engine i of type e of customer a. Here we assume that for a customer a the engines of a model e can be numbered $1,2,...,I_e^a$; then the combination of the index for the customer a, the type e and the number $i \in \{1,...,I_e^a\}$ identifies any engine that is served by the MRO.

The number of days until each threshold is reached, denoted as $r_{Y,e,i}^a$, $r_{H,e,i}^a$ and $r_{L,e,i}^a$, can then be computed by

$$r_{Y,e,i}^a = \frac{Y_e}{\gamma_{e,i}^a} - \tau_{e,i}^a$$

$$r^a_{H,e,i} = \frac{H_e}{\zeta^a_{e,i}} - \tau^a_{e,i}$$

$$r^a_{L,e,i} = L_e - \tau^a_{e,i}$$

where $\tau^a_{e,i}$ is the time since the last overhaul.

For the first two cases we use the inverse of our regression function. The $r^a_{j,e,i}$ with $j \in Y_e, H_e$ are the solutions to the equations:

$$f_{D_j}(r^a_{j,e,i} + \tau^a_{e,i}) = j$$

where f_{D_j} is the linear regression model that was trained on the corresponding learning data (the flight cycles for $j = Y_m$ and the flight hours for $j = H_m$). The remaining time until the next overhaul is due is the minimum of these three numbers:

$$r^a_{min,e,i} = \min(r^a_{Y,e,i}, r^a_{H,e,i}, r^a_{L,e,i})$$

So far we just looked at the remaining time before a single engine of a single customer is sent to the MRO for overhaul. However, with regard to capacity planning for service demand, it is necessary to look at aggregated numbers in specified planning periods. Therefore we first define a count function:

$$\mathbf{1}(r,t) := \begin{cases} 1 \, if \; r \in t \\ 0 \, else \end{cases}$$

It returns 1 if the critical threshold is reached in interval t and 0 if not. We have I_e engines of model e. The service demand for an engine model e of customer a in period t is:

$$d^a_{e,t} = \sum_{i=1}^{I_e} \mathbf{1}(r^a_{min,e,i}, t)$$

In Table 22 the specific service demand for each model in each period is listed for a customer a. If these results were publicly available from all customers it would be straightforward to aggregate them to obtain the estimated total demand for each engine type in each time period. However, the $d^a_{e,t}$, which are the interim results from the linear regressions, are still considered private for each customer a. Therefore, the aggregation should be performed in a privacy preserving protocol that only reveals the aggregated result and keeps each party's input private. A first concept is outlined in subchapter 3.1.2.2.3.

Engine type	period	1	…	t	…	T
1		$d^a_{1,1}$	…		…	$d^a_{1,T}$
$\vdots$		$\vdots$				

e			$d^a_{e,t}$		
$\vdots$					$\vdots$
E	$d^a_{E,1}$	$\dots$		$\dots$	$d^a_{E,T}$

Table 22: Number of engines expected for service for one customer a

3.1.2.2.2 Implementation

The main functionality of this submodule is the computation of the forecasts for the aggregated overhaul demand for each engine type e in a time period t, denoted as $d_{e,t}$. The process can be broken down in two major steps and is also depicted in Figure 15. In the first step the individual demand forecast for each engine type e for a customer a is estimated using two linear regression models. The calculation for this is formalized, for implementation

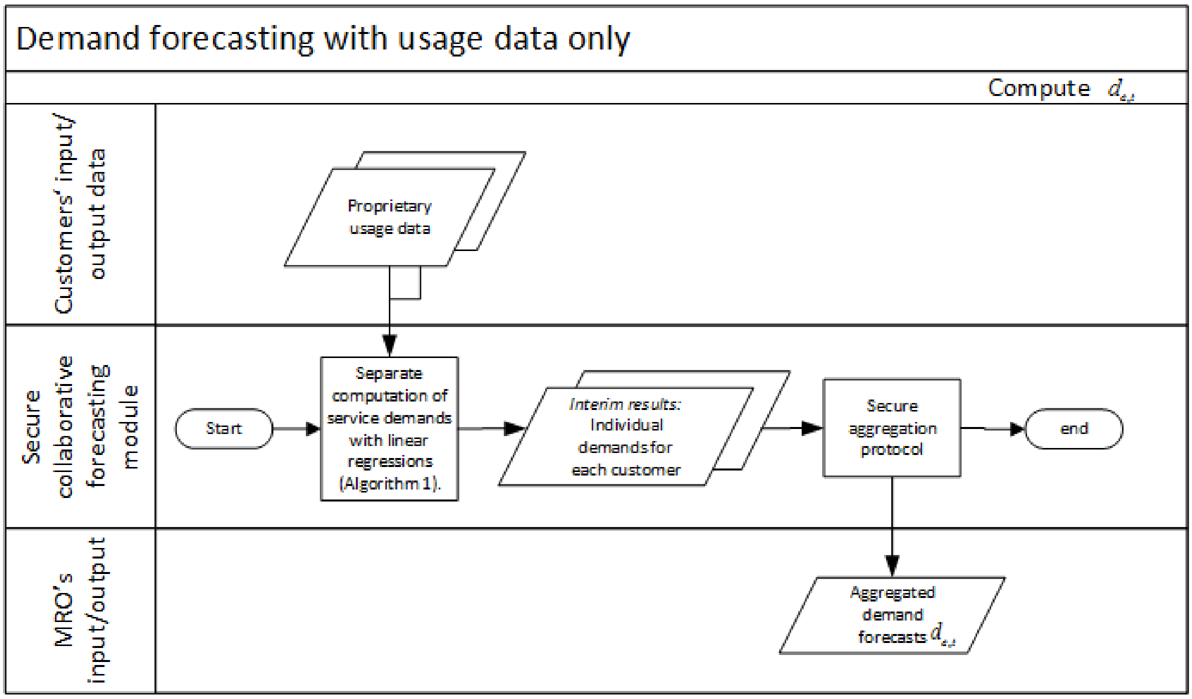


Figure 15: Demand forecasting with usage data only

purposes, in Algorithm 1.

Algorithm 1: individual demand forecasts for engine type e of a customer a

---

1: **For each** $i = 1{:}I_e$ *// For each engine i of type e*

2: $$\gamma^a_{e,i} \leftarrow \frac{\sum_{j=1}^{N} x_j y_j}{\sum_{j=1}^{N} x_j^2}$$ *//slope parameter in model for: #flight cycles:* y_i

3: $\zeta_{e,i}^{a} \leftarrow \frac{\sum_{j=1}^{N} x_j h_j}{\sum_{j=1}^{N} x_j^2}$ // slope parameter in model for flight hours: h_i

4: $r_{Y,e,i}^{a} \leftarrow \frac{Y_e}{\gamma_{e,i}^{a}} - \tau_{e,i}^{a}$ //time until cycle threshold is reached

5: $r_{H,e,i}^{a} \leftarrow \frac{H_e}{\zeta_{e,i}^{a}} - \tau_{e,i}^{a}$ //time until flight hour threshold is reached

6: $r_{L,e,i}^{a} \leftarrow L_e - \tau_{e,i}^{a}$ //time until live time threshold is reached

7: $r_{\min,e,i}^{a} \leftarrow \min(r_{Y,e,i}^{a}, r_{H,e,i}^{a}, r_{L,e,i}^{a})$ //determine the minimum

8: $t_{e,i}^{a} \leftarrow t \mid t \leq r_{\min,e,i}^{a} < t+1$ //time interval when engine needs overhaul

9: **end for**

10: $d_{e,t}^{a} \leftarrow \sum_{i=1}^{I_e} \mathbf{1}(r_{\min,e,i}^{a}, t)$ //compute the demand for engine type e in period t

The second step in the process is the secure aggregation of the interim results. We want to compute

$$d_{e,t} = \sum_{a=1}^{A} d_{e,t}^{a},$$

which would be straightforward if the interim results (i.e. all values $d_{e,t}^{a}$ for each customer) were centrally available in plaintext. They are, however, sensitive data for each customer and we require a secure aggregation protocol. This will be developed in the following subchapter.

3.1.2.2.3 Data sensitivity assessment[11]

As mentioned before, customers may be reluctant to share all or some of their individual input data openly with the MRO. Based on the model developed in the previous section, we can now identify all relevant input and output data and assess to which extent data privacy needs to be ensured. The detailed security requirements for each input variable are listed in Table 23.

Variable	Description	Party concerned	Security requirements
Y_e	Threshold value for #flight cycles of engine type e	OEM/supplier	Low – standard information from OEM/spare parts supplier

[11]Main authors: Florian Hahn (SAP), Fabian Taigel (UWUERZ)

H_e	Threshold value for flight hours of engine type e	OEM/supplier	Low – see above
L_e	Threshold value time between overhauls of engine type e	OEM/supplier	Low – see above
y_i	Usage data: # flight cycles	customer	High – Private data of each airline. Must be protected due to fear of data leakage.
h_i	Usage data: accumulated flight hours	customer	High- see above
$d^a_{e,t}$	Interim result: forecasted demand for engine type e of customer a in time period t	customer	High - Private data of each airline. Must be protected due to fear of data leakage.
$\gamma^a_{e,i}$	slope parameter in regression model for: #flight cycles y_i	customer	Medium – derived from private data of each airline. Should be protected due to fear of data leakage.
$\varsigma^a_{e,i}$	slope parameter in regression model for: flight hours h_i	customer	Medium – see above
$r^a_{Y,e,i}$	Time until threshold for #cycles is reached	customer	Medium – see above
$r^a_{H,e,i}$	Time until threshold for flight hours is reached	customer	Medium – see above
$r^a_{L,e,i}$	Time until threshold for time between overhauls is reached	customer	Medium – see above
$r^a_{min,e,i}$	Time until the first threshold is reached	customer	Medium – see above
$t^a_{e,i}$	Time interval in which first threshold is reached	customer	Medium – see above

Table 23: Data sensitivity assessment without sensor data.

We note that in a first step sharing these values is not necessary since the linear regression can be calculated locally on the customers' site using Algorithm 1. The crucial part is that the obtained interim results $d^a_{e,t}, a \in \{1,...,A\}$ can then be aggregated without the possibility that individual inputs are revealed. In literature there exist concepts that provide this form of secure aggregation, e.g. the additive homomorphic encryption scheme by Paillier (1999). It will be part of the next deliverable how such an encryption scheme could be applied in the given scenario and how it maps with the Practice architecture.

For future work one could investigate how to store all customers' data in a secure way on a centralized cloud based platform, however, enabling this cloud based platform to compute also the individual demand forecasts and their aggregation in a privacy preserving manner.

3.1.2.2.4 Conclusion and further references

In the previous subchapters we provided a basic model for a secure collaborative forecast of service demand given the currently available data. We describe a solution to forecast overhaul service demand that is triggered by usage parameters which reach specified thresholds. The output of the model is the aggregated demand for each type of engine per planning interval.

This information can be used for spare parts management, maintenance scheduling and capacity planning. In section 3.1.2.3 we show how spare parts demand can be forecasted for the engines that will be overhauled in a certain planning interval which then can be used for inventory management as described in Section 3.1.3.1. Also the maintenance scheduling approach in section 3.1.3.2 requires demand forecasts as inputs.

In literature, e.g. Hall et al. (2011) and Fang et al. (2013) used homomorphic encryption for privacy preserving linear regression. However, in their setting they aimed for an overall regression model estimated from distributed private datasets, whereas we need one model for each private dataset and a protocol to securely aggregate the results from each model.

3.1.2.3 Integrating condition monitoring data

As we know from our industrial partner, newer engine generations collect detailed sensor data as for example oil pressure, temperature and vibration levels. This is a development that can be observed in a broad range of industries as machines tend to get smarter (Neely 2008). They are equipped with sensors which are connected to the databases of their owners. This condition data can reflect the status of the engine much more accurately than only the usage data (meta information such as flight hours, etc.) we used in the chapter before. To keep our model applicable for various constellations of input data we model these as abstract attributes. However, it is important to note that these attributes now contain condition data in addition to the usage parameters used in the previous chapter. The necessary pre-processing of raw data is assumed to be completed beforehand. In Table 24 each row represents one instance with values for the k different attributes $Attr_1$, ..., $Attr_k$ and the class label y .

In this section we consider a scenario in which condition monitoring data based on the sensor data is available and used for forecasting in two areas that are highly relevant for the MRO's capacity planning problems:

- forecasting the type of service that will be necessary
- forecasting spare parts demand

The task of forecasting the actual need for overhaul services is deferred to subchapter 3.1.2.4. This reflects the most likely order of implementation since a change from the hard-time threshold model as described in the previous subchapter towards a condition-based preventive model as in 3.1.2.4 contains some risks for the customers and requires the highest quality of data and prediction as well as some changes of regulations.

Both forecasting tasks above can be condensed to the problem of whether a certain component can be reused or has to be replaced. If a component is reused there will be demand for specific service capacity (e.g. for cleaning, inspection, repair by specialized technician). If a component has to be replaced there will be a demand for a corresponding spare part. To predict the capacity and spare part demands, we use a probabilistic binary decision tree that was 'learned' on a historic dataset D. The choice of the instances that can be used in a learning dataset has to be related to the forecast horizon. An instance of learning data consists of a set of attributes taken at time t_{attr} and the class label that is determined during overhaul operations, let's say at time t_{class}. If we want to learn a decision tree that predicts time t_{pred} ahead we can only use learning instances where $t_{class} - t_{attr} \approx t_{pred}$.

$Attr_1$	$Attr_2$	$\ldots$	$Attr_k$	**Class label** y
x_{11}	x_{12}	$\ldots$	x_{1k}	y_1
$\vdots$			$\vdots$	$\vdots$
$x_{\|D\|1}$	$\ldots$		$x_{\|D\|k}$	$y_{\|D\|}$

Table 24: Structure of a learning dataset D

The structure of a learning dataset is depicted in Table 24. Each row represents one instance with values for the different attributes and the attached class label that we would want to predict for unclassified data. However, we slightly differ from the standard classification model since we are looking for the probability of a class label and not only for the mean response. This provides us with additional information about the reliability of the output which can then be considered by the subsequent spare parts planning module. Therefore we define our function for the probability of a replacement by:

$$f_D(\mathbf{x}) = p(replace \mid \mathbf{x}) = \sum_{l=1}^{L} \pi_l \phi(\mathbf{x}; \mathbf{v}_l)$$

where π_l is the probability for a replacement of an instance that is classified in leaf $l \in \{1, ..., L\}$ and $\phi(\boldsymbol{x}, \boldsymbol{v}_l)$ yields 1 if an instance $\boldsymbol{x}$ is classified in leaf l, described by the matrix $\boldsymbol{v}_l$. The estimated overall spare part demand F_{spares} for this component is then found by summing over all customer (airlines) $a \in \{1, ..., A\}$ and over all entries in their real-time datasets S_a for the specific component:

$$F_{spares} = \sum_{a=1}^{A} \sum_{\mathbf{x} \in S_a} f_D(\mathbf{x}) = \sum_{a=1}^{A} \sum_{\mathbf{x} \in S_a} \sum_{l=1}^{L} \pi_l \phi(\mathbf{x}; \mathbf{v}_l)$$

Loosely speaking, $f_D(\boldsymbol{x})$ yields the probability that a component with the attributes $\boldsymbol{x}$ needs to be replaced. Summing over the probabilities of all components of this type gives the expected demand for this particular type of component. The structure of a real-time dataset of a customer is similar to the learning dataset in Table 24, except for the missing column of the class label. Each row of a real-time data set represents the current attribute values of one component of that customer. We only consider one certain type of component, so all entries in a real-time dataset represent different components of the certain type.

In the following, we focus on F_{spares} as the demand for spare parts. The same logic can be applied when forecasting the number of parts that require service capacity. The two numbers are directly related since they add up to the overall number of components that are to be serviced in a certain period. This total number is given as a multiple of the number of expected engines in this period as forecasted by the models in sections 3.1.2.3 and 3.1.2.4.

In the following subsection 3.1.2.3.1 we describe how to learn the tree, i.e. how to find the $\mathbf{v}_l$ and how to estimate the probabilities, i.e. how to find the π_l. In this subchapter we assume that the MRO has a broad historic database including time series of historic condition data that was obtained during past service operations. In particular, the MRO obtains the information about the status of a certain component during disassembling and inspection. We model this response value as $y=0$ if a component can be reused or $y=1$ if it has to be replaced. The decision trees are learned on this historic database.

However, the sensitive real-time data we want to take as a basis for our predictions (i.e. classify) is under the control of the different customers. In subsection3.1.2.3.3 we provide a more detailed sensitivity analys and outline a concept how the established decision tree can be used to classify encrypted data and therefore predict the relevant parameters for the MRO's capacity and spare part planning.

3.1.2.3.1 Repair or replace: Forecasting with probabilistic decision trees

In this subchapter we describe how decision trees are learned and how an established tree is used to predict the probability that a specific component needs to be replaced. The sum of these probabilities represents the estimated overall spare part demand for this kind of component.

For convenience, Table 25 summarizes the notation used in this chapter:

F_{spares}:	MRO's estimated overall spare part demand for a component
F_{repair}::	MRO's estimated overall number of components that need to be repaired
D; D_l:	learning dataset, subset in leaf l
S_a:	real-time dataset of customer a we want to classify
A:	number of different customers
l; L:	leaf index, number of leaves
$\mathbf{v}_l$:	matrix that encodes the route from the root to leaf l
π_D; π_l:	probability for class 1 in dataset D; respectively leaf l
$1-\pi_l$:	complementary probability
$\boldsymbol{\bar{\pi}}$:	result vector that contains one entry for each classified component with the probability π_l when the component was classified in leaf l

j^*:	index of the best split attribute
s^*:	best split value for the according attribute
$\mathrm{H}(D)$:	deviance in a dataset D (measure for the purity)
$\mathbf{x}_i$:	Vector containing usage **and** condition parameters; a pair $(\mathbf{x}_i, y_i)$ is one row from a learning dataset D
y_i:	Class of an instance; in this chapter either 1 or 0 representing *replace* or *repair*
N:	Number of components of a certain type in a certain time period that need to be overhauled (and therefore either replaced or repaired)

Table 25: Notation used in this chapter

There are numerous variants of tree learning algorithms; in general they have in common that they use a greedy approach and build a tree by recursively partitioning the learning dataset, i.e. they start with the full set and find the attribute along with a split value that allows the best $split_{D_L,D_R}$ of a dataset D in two disjoint parts D_L and D_R. This is recursively continued in each child node until a stopping criterion is reached. The process is described in pseudo code in Algorithm 2 in the following subchapter. Similar to the one of Murphy (2012, p. 547) our split function determines the index j^* of an attribute along with the best split value t^* for this attribute by minimizing the sum over the deviance $\mathrm{H}(D)$ in the separated parts:

$$(j^*, t^*) = arg \min_{j \in \{1,\ldots,|\mathbf{x}|\}} \min_{t \in \mathrm{T}_j} \mathrm{H}(\{(\mathbf{x}_i, y_i) \in D \mid x_{ij} \leq t\}) + \mathrm{H}(\{(\mathbf{x}_i, y_i) \in D \mid x_{ij} > t\})$$

where $|\boldsymbol{x}|$ denotes the number of attributes in the parameter vectors $\mathbf{x}_i$.

Here we assume that all entries in the datasets are numerical so that the comparison operations make sense. This can easily be achieved by a bijective mapping of nominal values to numeric ones. The cardinality of the set of possible values for the threshold value T_j is equal to the number of distinct values of an attribute because any additional splits would be superfluous. Typical stopping criteria are

- the depth of the tree
- the number of instances that are available in a certain node
- the potential improvement achievable in the next split.

The measure for the quality of a split is the purity in the disjoint parts. A perfect split would lead to two parts that only contain instances of the same class. Then the estimated probability for an instance that is classified in the leaf would be close to 1. Similar to Murphy (2012, p. 549) we measure the deviance from this purity by:

$$\mathrm{H}(D) = -\pi_D \log \pi_D - (1-\pi_D)\log(1-\pi_D)$$

where

$$\pi_D = \frac{1 + \sum_{(\mathbf{x}_i, y_i) \in D} \mathbf{1}(y_i = 1)}{2 + |D|}$$

is the Laplace corrected estimate for the class-conditional probability as proposed by Provost (2003, p. 209). The characteristic function $\mathbf{1}(y_i = 1)$ yields 1 if an instance is labelled as 1, i.e., if the component needs to be replaced. In our setting π_D is just the rate of instances in a dataset labeled as 1. The Laplace correction smoothes the estimation by adding 1 in the numerator and the number of different classes (2 in our case) in the denominator.

Given these functions, we can determine the best split in each node. The split function is called recursively in each of both partitioned datasets, which results in a tree like structure as depicted in Figure 16. In each step we store the attribute to split on, the split value and the direction (i.e. left or right branch) as a row in a matrix with three columns. When a stopping criterion is reached, the matrix $\boldsymbol{v}_l$ contains one row for each split and therefore encodes the route from the first node (i.e. the root) to this leaf l. A matrix $\boldsymbol{v}_l$ is denoted by:

$$\mathbf{v}_l = \begin{bmatrix} j_1^* & s_1^* & dir_1 \\ \vdots & \vdots & \vdots \\ j_{depth}^* & s_{depth}^* & dir_{depth} \end{bmatrix}$$

where j_i^* is the index of the ith split attribute, s_i^* the according split value and dir_i^* is the direction, where $dir_i = 0$ stands for $attr\ j_i^* \le s_i^*$ and $dir_i = 1$ for the information that $attr\ j_i^* > s_i^*$. A formalized algorithm for the process of decision tree learning is given in Algorithm 2 in the following subchapter.

The output of such a tree learning algorithm can be displayed as a tree like the example in Figure 16. In this example the parameters that define the second leaf (with the assigned probability π_2) would be $\boldsymbol{v}_2 = \begin{bmatrix} j_1^* & s_1^* & 0 \\ j_2^* & s_2^* & 1 \end{bmatrix}$.

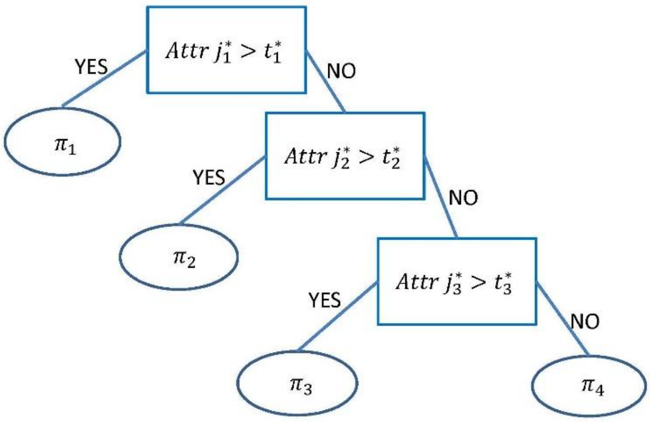


Figure 16: Example of a binary decision tree of depth 3

After learning a decision tree on a learning dataset, it can be used to classify a new instance (i.e. a set of attributes with the current parameters of a component without the information whether it needs to be replaced). We formalize the classification of an instance by:

$$f_D(\mathbf{x}) = p(replace \mid \mathbf{x}) = \sum_{l=1}^{L} \pi_l \phi(\mathbf{x}; \mathbf{v}_l)$$

where π_l is the probability for a replacement of an instance that is classified in leaf $l \in \{1,...,L\}$ and $\phi(\boldsymbol{x}, \boldsymbol{v}_l)$ yields 1 if an instance $\boldsymbol{x}$ is classified in leaf l, described by the matrix $\boldsymbol{v}_l$. In the example above this means that $\phi(\boldsymbol{x}, \boldsymbol{v}_2) = 1$ if $Attr\ j_1^* \leq s_1^*$ and $Attr\ j_2^* > s_2^*$.

To obtain the overall demand for spare parts of a certain type of component we need to classify all N instances of this type of component that possibly need to be replaced. We denote the result as a vector containing the probability for each component as:

$$\overline{\boldsymbol{\pi}} = (f_D(\boldsymbol{x_1}), f_D(\boldsymbol{x_2}), \ldots, f_D(\boldsymbol{x_N}))$$

The expected number of components that need to be replaced F_{spares} can simply be computed by summing over all individual replacement probabilities for each part:

$$F_{spares} = \sum_{i=1}^{N} \overline{\boldsymbol{\pi}}[i]$$

However, the mean does not convey sufficient information for successful spare parts (or capacity) management. We also need a measure for how likely a deviation from this mean is. Therefore we need a probability distribution over all possible outcomes. Algorithm 4 in the subchapter 0 gives a detailed description of the computation.

3.1.2.3.2 Implementation

The overall process for forecasting spare parts or repair capacity demand is depicted in Figure 17. The difference between these two forecasting objectives is only between classification and the computation of mean and probability distribution and is described further below.

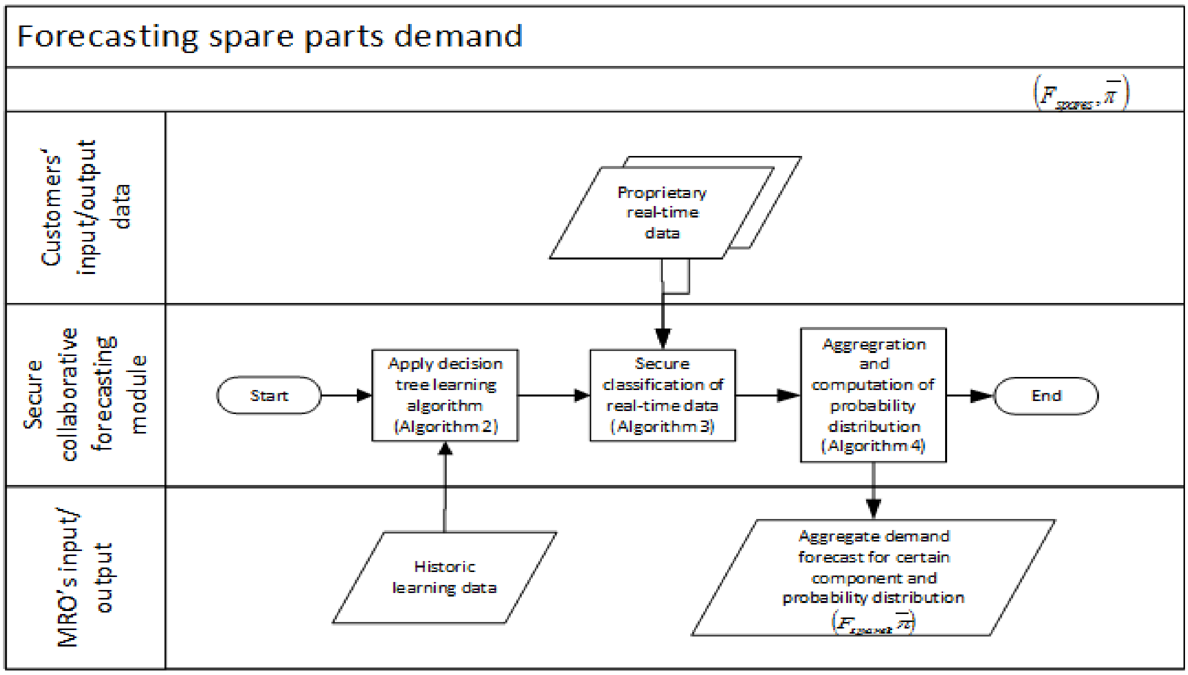


Figure 17: Secure collaborative forecasting process for repair/replace demand

Three separable task can be identified for the secure computation platform which are described in detail in this subchapter:

1. Decision tree learning algorithm
2. Secure classification
3. Computation of mean and probability distribution

The decision tree learning algorithm works as follows:

Algorithm 2: Recursive procedure to grow a binary classification tree[12]

function fitTree(node, D, depth)

node.$\pi_D \leftarrow \dfrac{1+\sum_{y_i \in D} \mathbf{1}(y_i = replace)}{2+|D|}$ //empiric probability for replacement

$(j^*, s^*, D_{left}, D_{right}) = split(D)$; //split attribute j^* and split value s^*

if stopping criterion is reached **then**

return node

else

node.left $\leftarrow$ fitTree(node, D_{left}, depth$+1$) //recursive call

node.left. $\mathbf{v}[depth] \leftarrow (j^*, s^*, 1)$ // store the split attribute and value

node.right = fitTree(node, D_{right}, depth$+1$)

node.right. $\mathbf{v}[depth] \leftarrow (j^*, s^*, 0)$

The returned nodes are the leaves of our decision tree. Given $\boldsymbol{V} := (\boldsymbol{v}_1, \dots, \boldsymbol{v}_L)$, where each $\boldsymbol{v}_l$ describes one of these leaves we can classify all instances in the encrypted data sets $S_{a,encr}$ of all customers $a = 1, \dots A$.

Algorithm 3: Secure classification

function classify($\boldsymbol{V}$, $S_{a,encr}$)

for l in $1{:}L$

$n_l \leftarrow \sum_{\mathbf{x} \in S_{encr,a}} \phi(\mathbf{x}; \mathbf{v}_l)$ //SelectCount queries

attach n_l entries at the result vector $\overline{\boldsymbol{\pi}_{\boldsymbol{a}}}$ each with the value π_l

end for

return $\overline{\boldsymbol{\pi}_{\boldsymbol{a}}}$

[12] Adaption of Murphy (2012, p. 548)

The final result vector is then given as a concatenation of all $\overline{\boldsymbol{\pi}}_{\boldsymbol{a}}$ as

$$\overline{\boldsymbol{\pi}} = (\overline{\boldsymbol{\pi}}_{\boldsymbol{1}}, \overline{\boldsymbol{\pi}}_{\boldsymbol{2}}, \dots, \overline{\boldsymbol{\pi}}_{\boldsymbol{A}})$$

The expected spare part demand is then:

$$F_{spares} = \sum_{i=1}^{N} \overline{\boldsymbol{\pi}}[i]$$

In order to obtain the expected demand for parts that have to be repaired, denoted by F_{repair}, and hence the required repair capacity we just use the contrary probabilities, which yields:

$$F_{repair} = \sum_{i=1}^{N} (1 - \overline{\boldsymbol{\pi}}[i])$$

The following algorithm describes how to obtain a probability distribution from the result vector $\overline{\boldsymbol{\pi}}$, which contains one entry for each component with its probability for replacement:

Algorithm 4: Computing the probability distribution

function distribution($\overline{\boldsymbol{\pi}}$)
  $N \leftarrow length(\overline{\boldsymbol{\pi}})$
  $\mathbf{p} \leftarrow$ vector with $length = N+1$ *//initialize the distribution vector*
  $\mathbf{p}[1]=1$; //*set the first entry to* 1
  for $(i \ in \ 2{:}N+1)$ *//in each step we add one component*
    for $(j \ in \ i{:}2)$ *// we iterate backwards since we use* $\mathbf{p}[i-1]$
      $\mathbf{p}[j] = \mathbf{p}[j-1]*(1-\overline{\boldsymbol{\pi}}[i-1]) + \mathbf{p}[j]*\overline{\boldsymbol{\pi}}[i-1]$
    end for
    $\mathbf{p}[1] = \mathbf{p}[1]*\overline{\boldsymbol{\pi}}[i-1]$ *//this is the probability that all components have to be replaced*
  end for
  return$(\mathbf{p})$ *//now* $\mathbf{p}[i]$ *contains the probability for* $N+1-i$ *replacements*

As an example consider the case with only 3 components with the individual probabilities for replacement given by $\overline{\boldsymbol{\pi}} = (0.8;0.2;0.1)$. Then the initial distribution vector has length 4 and is given by $\mathbf{p} = (1;0;0;0)$. This vector already fulfils all necessary criteria for a probability distribution since:

$0 \le \mathbf{p}[i] \le 1$ for $i = 1,\dots,length(\mathbf{p})$

and $\sum_{i=1}^{length(\mathbf{p})} \mathbf{p}[i] = 1$

We will show that one iteration of the inner loop in Algorithm 4 maintains these properties. This gives the inductive proof that the return of algorithm 4 is in fact a probability distribution. In the next step the algorithm sets $\mathbf{p}[2] = \mathbf{p}[1]*(1-\bar{\boldsymbol{\pi}}[1]) + \mathbf{p}[2]*\bar{\boldsymbol{\pi}}[1] = 1*(1-0.8) + 0*0.8 = 0.2$ and $\mathbf{p}[1] = \mathbf{p}[1]*\bar{\boldsymbol{\pi}}[i-1] = 1*0.8 = 0.8$. We now have the distribution for one component where $\mathbf{p}[1]$ is the probability that this component needs replacement and $\mathbf{p}[2]$ the counter probability and thus $\mathbf{p} = (0.8;0.2;0;0)$. Adding another component results in three possible events: Either both parts need to be replaced or only one or none at all. The idea for the algorithm is that the probability for j replacements with an additional component consists of two parts:

- the probability for j replacements without the additional component times the probability that the additional one does not have to be replaced
- the probability for j-1 replacements without the additional component times the probability that the additional one does have to be replaced

Therefore, we cover all possible combinations of repair/replace. With the same formulas as above in the next step we obtain $\mathbf{p} = (0.16;0.68;0.16;0)$ and in the last step as final result $\mathbf{p} = (0.016;0.212;0.628;0.144)$

Now F_{spares} (the expected number of components that need to be replaced) can be computed in an additional way by calculating the mean of the constructed distribution:

$$F_{spares} = \sum_{i=1}^{N+1} \mathbf{p}[i]*(N+1-i)$$

Figure 18 illustrates the result of another illustrative application of Algorithm 4. It shows the distribution function for hundred components. Sixty of these components had replacement probabilities equally distributed between 0.75 and 0.95. The other forty were equally distributed between 0.05 and 0.25. The mean for this example is $F_{spares} = 64.12$.

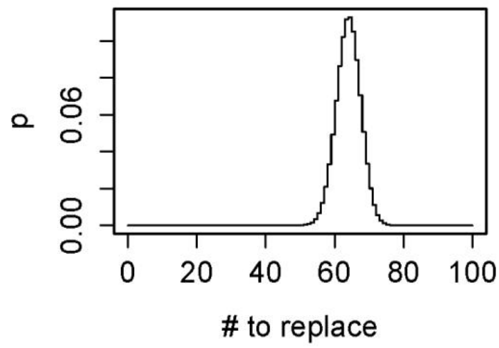


Figure 18: Example of a demand distribution

Given a distribution function $\boldsymbol{p}$ we can easily define the cumulative distribution function $\Phi(\delta)$, i.e. the function that yields the probability that the realized demand does not exceed a certain value δ:

$$\Phi(\delta) = \sum_{i=1}^{\delta} \boldsymbol{p}[i]$$

So far we focused on the task to predict spare parts demand for a certain period. However for profound spare parts management as described in section 3.1.3.1 we need the aggregated demands over longer time periods. First, we need an estimate for the entire planning horizon H. Second, we need the demand for the protection period R, which is the first time interval that covers the lead time $\mathcal{L}$ of a component. The time structure is also depicted in Figure 19. To deliver these forecasts we need to establish a decision tree for each time period $t = 1, \dots, H$. Then we can aggregate the outputs of the respective periods.

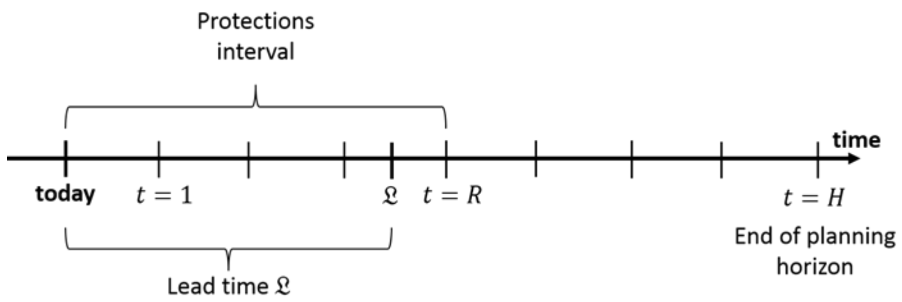


Figure 19: Time structure of the planning horizon H.

To keep the computational effort within reasonable limits, we propose to utilize the characteristic of our machine learning concept that the entire process can be separated in two main steps: Learning phase and classification (or prediction) phase. The learning of all H decision tree models that are necessary for one component could be performed once every planning period with the additional learning data obtained during the most recent period. Whereas the classification of the real-time data should be updated on a daily basis or even in real time, i.e., whenever an instance of the dataset changes.

3.1.2.3.3 Data sensitivity assessment

In Table 26 we first provide an overview over the used input and process data with the respective security requirements. Then we outline an approach for privacy preserving computation of the demand forecasts from order-preserving encrypted real-time data.

Variable	Description	Party concerned	Security requirements
D; D_l	leaning dataset, subset in leaf l	MRO	Medium – data is collected by MRO during overhaul operations and therefor centrally available for learning phase. However, it contains information about engines of different (competing) customers.
$\mathbf{v}_l$	matrix that encodes the route from the root to leaf l	MRO	High – derived from sensitive learning data set.
π_D; π_l	probability for class 1 in dataset D; respectively leaf l	MRO	Medium – only valuable in combination with $\mathbf{v}_l$
S_a	real-time dataset of customer a we want to classify	customer	Highest – sensitive data from each customer. Includes usage and condition data. No access to full plaintext possible
$\boldsymbol{\pi}$	Result vector that contains one entry for each component with its probability for replacement:	MRO and customer	Medium – the entries are not retraceable to a specific component, so only medium criticality

Table 26: Data sensitivity assessment without sensor data

As mentioned before, the concept of decision tree learning can be separated into two phases: 1) The learning phase in which the tree is built using a given set of labelled data and 2) the phase in which the established tree is used to classify new data. Because in each phase we have different computational requirements and a different distribution of data, we consider both phases separately.

For the learning phase we assumed that the MRO has one central database that stores all historic condition and usage data, that is, all the data required to grow the tree.

In the classification phase privacy requirements are high since we want to classify datasets of different (competing) customers. In the aerospace example these datasets include current flight plans and real-time condition data. This is sensitive data, not only for passenger or cargo airlines, but more so for national air forces. In this scenario we propose to store the customers' data in encrypted databases using an order preserving encryption scheme (OPES) as introduced by Agrawal (2004, pp. 367ff.). The present chapter focuses on the more general rationale of the concept rather than on all of the technical details. Details for an implementation in the specific secure implementation in the aerospace MRO context and for the integration in the PRACTHCE architecture will be part of the next deliverable.

In combination with OPES the main advantage of decision tree classification comes into effect: in the classification phase the computational requirements are rather low since we only need comparison operations $\{>;<;=\}$ to classify an instance. OPES allows us to perform exactly these comparison operations on encrypted data, satisfying:

$$encr(a) < encr(b) \Leftrightarrow a < b,$$

where $encr(a)$ means the encryption of data a. Main features of an order preserving encryption scheme (OPES) are:

- Ordering and comparison operations can be directly applied to encrypted data, including equality and range queries, MAX, MIN and COUNT queries.
- Updates on the encrypted database like adding a value or modifying an already existent one can be handled without changing the encryption of other values
- Time and space requirements are reasonable enough to allow deployment in real systems

To classify an instance by comparing its attributes' values with all splits of a given decision tree, comparison operations are sufficient. The COUNT operation then allows us to obtain the number of instances in each class. Consider the sample database in Table 27 for illustration purposes. It contains encrypted real-time data that is to be classified by the corresponding tree like the one provided in Figure 16. For each engine the most recent set of data is stored.

$Attr_1$	$Attr_2$	$\ldots$	$Attr_k$
$encr(x_{11})$	$encr(x_{12})$	$\ldots$	$encr(x_{1k})$
$\vdots$			$\vdots$
$encr(x_{\|S\|1})$	$\ldots$		$encr(x_{\|S\|k})$

Table 27: Sample for an encrypted unclassified dataset S_{encr}

The *MRO* has read access to the encrypted database via a cloud-based platform, though he does not have the key to decrypt the data. Still he can calculate the necessary forecasts by running the classification of the developed tree via SQL queries. For each leaf of the tree it is one query which counts how many engines fall into this leaf. For the example in Table 19 and the second leaf of the tree in Figure 16, which is represented by the matrix $\boldsymbol{v}_2 = \begin{bmatrix} j_1^* & s_1^* & 0 \\ j_2^* & s_2^* & 1 \end{bmatrix}$. The SQL query would look as follows:

SELECT COUNT	ID
FROM	S_{encr}
WHERE	$Attr\ j_1^* \leq s_1^*$
AND	$Attr\ j_2^* > s_2^*$

This query returns the number of engines which are assigned to the leaf with $(\pi_2;1-\pi_2)$ as the probabilistic distribution over the two classes (replace; repair). All leaves can be evaluated through similar queries. This yields the same result as $\sum_{\mathbf{x}\in S_{encr,a}} \phi(\mathbf{x};\mathbf{v}_2)$.

3.1.2.3.4 Conclusion and further references

In this subchapter we consider only two possible classes: repair and replace. However, for more accurate capacity planning it might be necessary to differentiate additional classes that come with different capacity requirements. For example, an additional class ‘reuse’ for a component that does not require expensive repair capacities. In subchapter 3.1.2.4.1 we describe how decision trees can be applied in such a multi-class scenario.

Relevant related work in the field of airline maintenance is e.g. Letourneau et al. (1999) where different machine learning techniques (including decision tree learning) are compared with regard to their ability to predict the need for replacement of specific aircraft component from condition data. Their goal was to set up an alarm system that could indicate whether a component should be replaced within a predefined upcoming time period. The main focus of their work was the (cumbersome) pre-processing of data and the comparison of different model. In contrast to our work, they did not show how such techniques could be used to obtain forecasts for aggregate spare parts demand. Furthermore, they did not consider any issues concerning the privacy of the required data.

3.1.2.4 Secure condition based preventive maintenance

In the previous subchapter we showed how additional condition data can be used to derive information about the status of specific components at the time of their predicted overhaul event. We now assume that the available condition data allows to predict the overhaul period. In the context of machine learning this means that we can infer a functional relationship between the condition of an engine, now described by multiple attributes, and the need for overhaul at some point in the future by analyzing a broad historic dataset for this correlation. The advantage of this approach is that an engine is overhauled when it is necessary due to its actual condition. Currently the airlines use fixed thresholds for the usage parameters to determine the dates for overhaul operations as described in subchapter 3.1.2.2. Since an engine is a highly complex technical entity, our approach is to break it down into several components. Overhaul for the entire engine is due when the first of these components requires overhaul. So in the following subchapter we describe a model for predicting the time period when overhaul is necessary for a component.

Here the classes represent ordered time intervals $t_1,...,t_N$. For example each t_i could stand for a month i months ahead. Again we will learn a function $f_D(\boldsymbol{x})$ on a dataset D, the only difference to the model in the preceding subchapter is that the function now returns not only the probability for the first of two possible classes but a distribution over N classes.

$$\mathrm{f_D}\,(\mathbf{x}) = (\Pr(t_1),\Pr(t_2),..., \Pr(t_N)\,|\,\mathbf{x}) = \sum_{l=1}^{L} \vec{\pi}_l \phi(\mathbf{x};\mathbf{v}_l)$$

Where $\vec{\pi_l}$ now is a vector with a probability distribution for leaf l.

3.1.2.4.1 Forecasting maintenance demand with ordinal decision trees

In this subchapter we want to predict the time period until overhaul is required for a component. Here the classes represent ordered time intervals $t_1,...,t_N$. For example each t_i could stand for a month i months ahead. Hence, the main difference to the concept described in subchapter 3.1.2.3 is that we now have to deal with more than two classes.

The procedure as described in Algorithm 2 could be easily adapted to handle multiple classes, by just considering multiple splits and minimizing an adapted deviance function. However, this would cause a loss of information since the measure for the quality of a split only considers the deviance between all classes and not the concentration of probability mass around one class. For clarification consider an example where we have five classes $t_1,...,t_5$ where t_i stands for a month i months ahead. Then the adapted learning algorithm would yield for each leaf of the tree a vector of probabilities for each of the five classes. For example $\pi := (0.4;0.3;0.1;0.1;0.1)$ with the deviance $\mathrm{H}(\pi) = \sum_{i=1}^{5} -\pi[i]\log \pi[i]$ where $\pi[i]$ is the i th entry of the vector π. For a component that is classified in this leaf we have quite a high chance that it requires overhaul in the next two months ($0.4+0.3=0.7$). Intuitively, such a distribution of probabilities makes more sense than for example $\tilde{\pi} := (0.4;0.1;0.1;0.3;0.1)$, where we have a higher chance for month 1 and for month 4. However, the deviance of both distributions is the same so the algorithm would not be able to consider the additional information that a leaf with a higher concentration around one class is to be preferred. Standard decision tree learning algorithms ignore this additional information.

The better solution to handle ordinal N-class classification problems is to divide the problem in $N-1$ binary classification problems as proposed by Frank and Hall (2001). The first step is to derive $N-1$ binary learning datasets from the original one with N different ordered classes. Therefore we copy the attribute values and replace in set D_{t_i} the class value by 1 if the original value was greater than the threshold t_i for this set and by 0 otherwise. Now we can grow a binary tree in each of these learning sets with the methods described in subchapter 3.1.2.3. This gives us classification functions $f_{D_{t_i}}(x) = Pr(Target > t_i \mid x)$ for each set. This process is depicted in Figure 20.

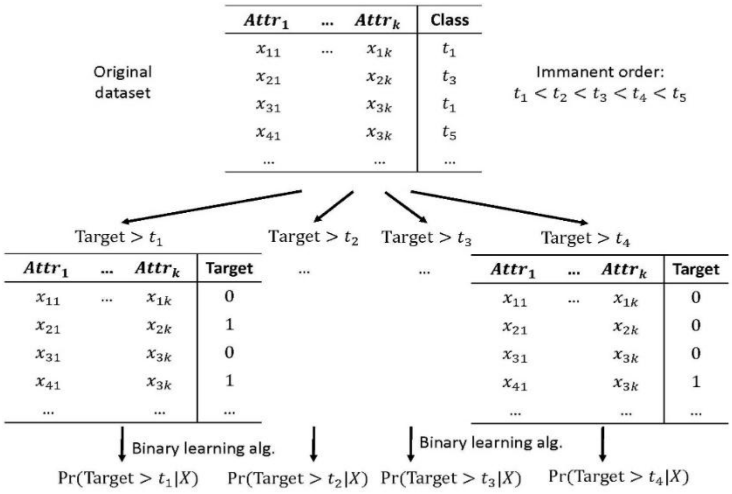


Figure 20: Deriving binary classification tasks

To classify an instance X we now just compute the result from each binary classification $f_{D_t}(\boldsymbol{x})$ and then assemble the class probabilities in the following way:

$$\boldsymbol{\pi}[1] = Pr(t_1) = 1 - f_{D_{t_1}}(x) = 1 - Pr(Target > t_1)$$

$$\boldsymbol{\pi}[i] = Pr(t_i) = f_{D_{t_{i-1}}}(x) - f_{D_{t_{i-2}}}(x) = Pr(Target > t_{i-1}) - Pr(Target > t_i)$$

, for $1 < i < N$

$$\boldsymbol{\pi}[N] = Pr(t_N) = f_{D_{t_{N-1}}}(x) = Pr(Target > t_{N-1})$$

Hence, we obtain

$$f_D(\boldsymbol{x}) = \boldsymbol{\pi} = (\boldsymbol{\pi}[1], \dots, \boldsymbol{\pi}[N])$$

where $\boldsymbol{\pi}$ is the demand distribution over the possible classes (i.e. time intervals).

3.1.2.4.2 Implementation

The basic concept is quite similar to the one of subchapter **Fehler! Verweisquelle konnte nicht gefunden werden.** and described in Figure 21. The main difference is that we now have to derive binary learning datasets as described in Figure 20 before we can apply the learning algorithm given by Algorithm 2. Since we break down the multi-class classification problem to multiple binary classification tasks we can again apply Algorithm 3 for the secure classification step. In Algorithm 5 we describe how the results from the classification are aggregated to generate the demand forecasts for each time period.

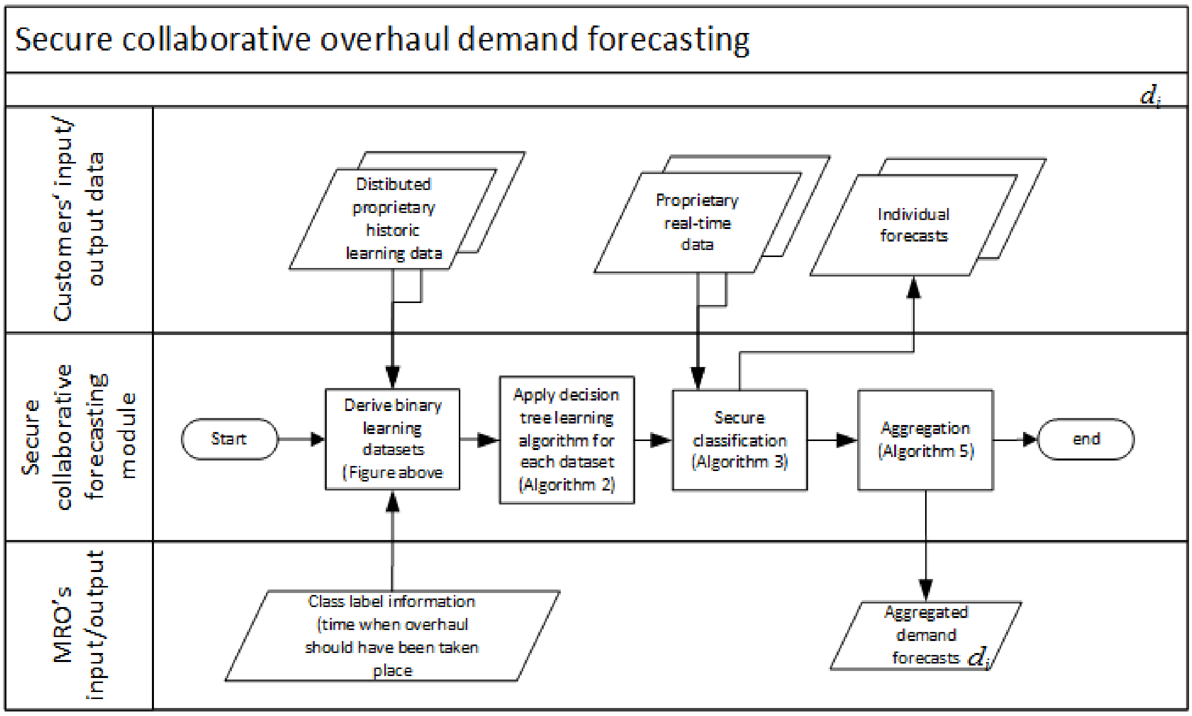


Figure 21: Secure collaborative forecasting process for overhaul demand using condition data

The following algorithm illustrates a possible implementation approach for the aggregation step:

Algorithm 5: Aggregation over multiple classification results to get forecast for period t_i

$d_i \leftarrow 0$ *// Initialize output parameter: the demand for a certain engine type in period t_i*

For $a = 1{:}A$ *// For each customer a*

  For each $\boldsymbol{x} \in S_a$ *// For each instance in the real-time data set of customer a*

    $d_i \leftarrow d_i + f_D(\boldsymbol{x})[i]$

  end for

end for

3.1.2.4.3 Data sensitivity assessment

Since we transferred the forecasting scenario with multiple classes back to a problem of multiple binary classification trees the privacy-preserving concept is quite similar to the one extensively described in subchapter 3.1.2.3.3. As one aspect for future work we will investigate how to drop the assumption that the historic usage and condition data is centrally available to the MRO for the model learning phase. Instead we will consider a scenario with distributed learning data. In this scenario the historic time series of usage and sensor data are kept private by each customer and the MRO is in possession of information about the true status of an engine and actually optimal time for overhaul (i.e. the class label) obtained during overhaul operations.

Variable	Description	Party concerned	Security requirements
D	Distributed learning dataset	MRO	High – historic usage and condition data is considered private by customers. Privacy preservation required

Table 28: Differing security requirements for distributed private learning datasets

In this extended scenario a promising approach could be to use secure multiparty computation protocols for decision tree learning on vertically partitioned data similar to the ones of Vaidya (2008) , de Hoogh (2014) and Lindell and Pinkas (2009, p. 88). They describe approaches how to grow a tree when data is spread between separated databases. Each database contains similar attributes, but the values are arbitrarily partitioned between these databases. The class attribute is public but the values of the attributes must not be shared. These approaches would be applicable if we assume the MRO shares his values for the class attribute (the actual best time for overhaul) with his customers.

In the classification phase order preserving encryption of real-time data could again be applied to maintain privacy while allowing the necessary count and compare operations to obtain the forecasts. The difference compared to the process in subchapter 3.1.2.3 is that we have to use multiple trees to obtain one forecast.

A remark on the value of privacy preservation in this context: if the machine learning model is used to forecast the demand for overhaul services and does so with sufficient quality it becomes a major competitive advantage for the entire supply chain. This increases the importance of keeping these models private as they contain all the information on how to derive forecasts for overhaul demand from condition data. The main benefits were described at the beginning of this chapter.

3.1.2.4.4 Conclusion and further references

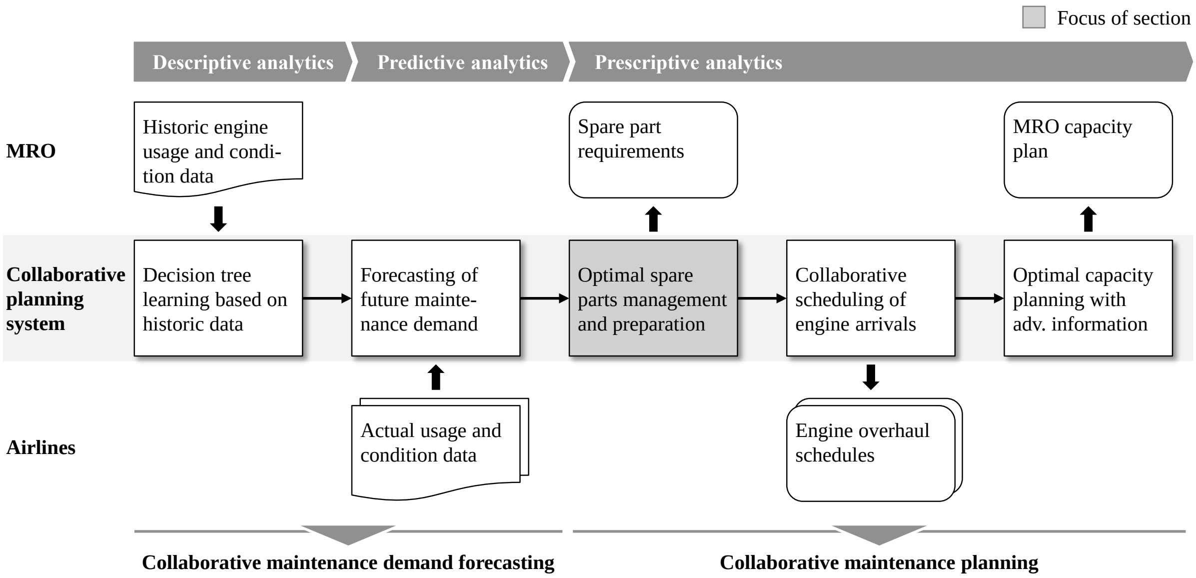
In this subchapter we introduced concepts to predict the time period, after which overhaul is required for a component. The prediction was based on condition data that reflects the actual status of the component in addition to usage parameters. We proposed decision tree learning as method to derive the relationship between parameter values and the probabilities for overhaul service demand in future time periods. This technique is especially valuable due to its compatibility with order preserving encrypted databases in the classification phase.

The final outputs of this section are the aggregated numbers for each type of engine that are expected for overhaul in each planning period. This information can be used for the MRO's capacity planning. As interim result we obtain a probability distribution for the service demand over all possible planning periods for a specific engine. This result can be used for maintenance scheduling as described in section 3.1.3.2.

3.1.3 Collaborative maintenance planning

3.1.3.1 Spare parts management[13]

Due to the lack of information about incoming engines and the resulting demand of service parts, the effective control of spare parts inventory is a major challenge for the MRO. With the introduction of a forecasting and planning system, the maintenance service provider can manage its service parts much more effectively and can achieve higher service levels with constant or even decreasing inventory costs.



In this subchapter, we propose an inventory steering approach for high-valued items based on estimations of future spare parts demand. We assume an intermittent, non-stationary demand pattern which is an appropriate assumption in the case of spare parts for the aviation sector. The policy is characterized by a re-order point and the order quantity, for which a sequential solution procedure to calculate the parameters is presented.

3.1.3.1.1 A dynamic (r,Q) inventory control policy

Forecasting the demand of service parts is an extremely difficult task. The spare parts demand for the MRO depends on factors such as the condition of the delivered engines, age and wear of the installed components and their previous usage profiles, and is in general not free from trends and seasonality.. Hence, we presume that demand for spare parts does not follow a stationary demand distribution. Instead, especially for high-value and critical service parts, we assume that demand follows a non-stationary, intermittent pattern with periods of zero demand between demand intervals. Figure 22 illustrates such a demand pattern.

[13] Main author: Jan Meller (UWUERZ)

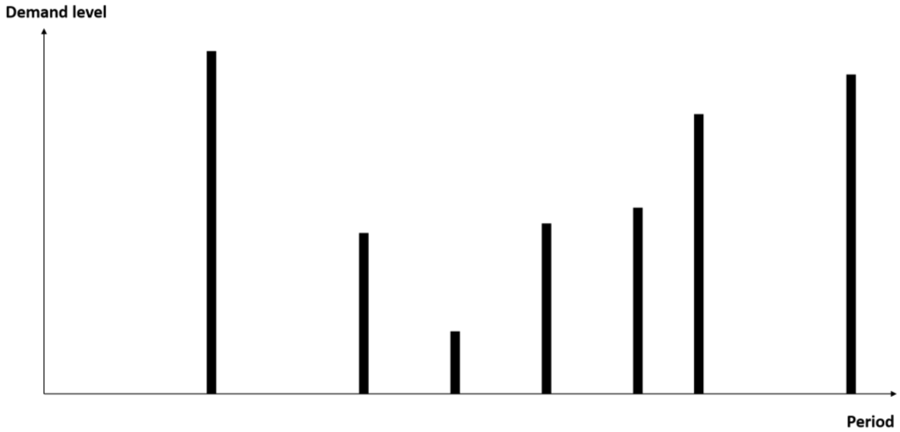


Figure 22: Intermittent demand pattern for spare parts

Furthermore, no specific forecasting procedure is proposed. Instead, in order to focus on efficient inventory control, we assume that demand is forecasted within the dedicated *Secure Collaborative Forecasting Module* which is exogenous to the inventory control model. For this reason, the model is compatible to a wide range of demand forecasting models or procedures, increasing its relevance for changing market and demand conditions of the MRO.

Key output of the collaborative forecast module and input to our inventory management approach are:

- A point estimate of all demands during the forecasting horizon H denoted as $F_{Spares,H}$
- A cumulative distribution function (cdf) of the forecasted demand over H in order to incorporate information about the corresponding forecast uncertainty, denoted as $\Phi_{CFD_H}(\cdot)$

Furthermore, replenishment lead times from the supplier as well as the current inventory level are needed in order to determine dynamic parameters for the inventory control model.

For the remainder of this chapter, the following notation is used:

F_H	Point forecast of the cumulative demand over the next H periods
CFD_H:	Cumulative forecasted demand over interval *H*
$\Phi_{CFD_H}(\cdot)$:	Probability distribution function of *CFD_H*
I_t:	Inventory at the end of period *t*
L	Replenishment lead time
K:	Fixed costs incurred per order
h:	Inventory holding cost per unit
SL:	Service level

In the following, the sequence of events within the *Spare Parts Management Module* is presented. We assume a discrete time model in which inventory is controlled at the end of each review period. At the beginning of a period t_k, starting inventory equals the inventory position at the end of the previous period t_{k-1} plus the received order quantity Q which was ordered in period t_{k-L}. Demand is realized throughout the period. At the end of the period, the inventory is controlled and if the inventory position has fallen below the re-order point r_{t_k}, Q units are reordered and received after L periods. Finally, the forecasts are updated and the re-order point for the subsequent period is determined. Figure 23 illustrates the evolution of the inventory level over a whole planning horizon H, consisting of several periods and replenishment cycles.

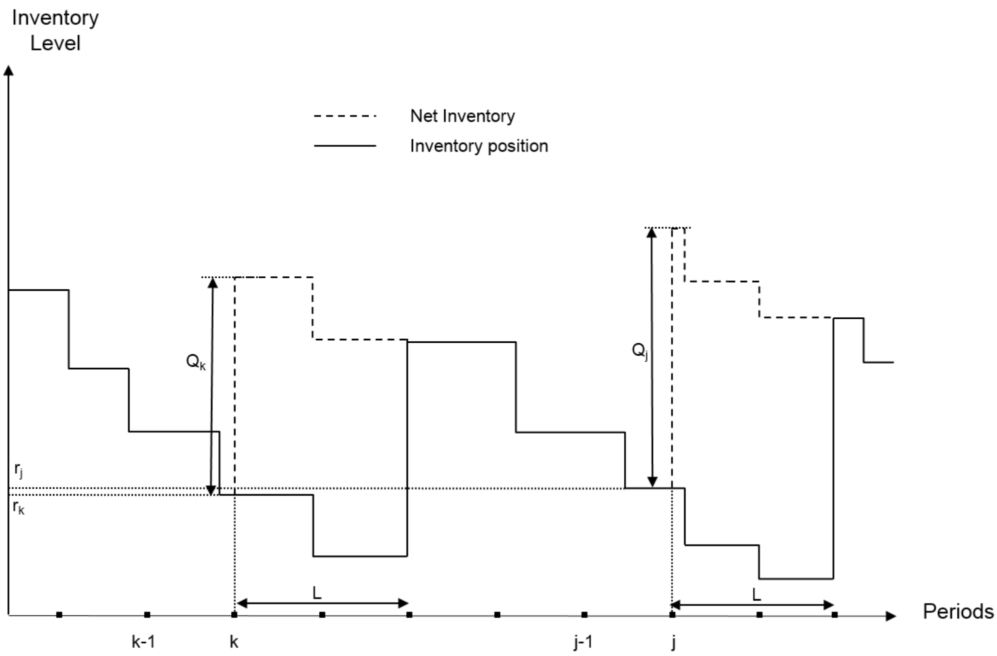


Figure 23: Dynamic (r_t,Q) inventory policy [illustration according to Babaï et al. (2009)]

The decision when to order is based on the expected demand over the supplier's lead time plus the review period, hence over *R = L+1* due to the characteristic that inventory is controlled at the end of a period. As we also update forecasts periodically, the parameters of the policy are dynamically calculated and updated, adjusting inventory control to current demand expectations.

Taking demand uncertainty into account via the cumulative distribution function of the forecasted demand, the reorder point r_t for period t can be determined to:

$$r_t = \Phi^{-1}_{CFD_R}(SL)$$

In order to compute the optimal order quantity Q, Babaï et al. (2009, p. 2469) propose to employ Wilson's formula,

$$Q = \sqrt{\frac{2KF_H}{hH}},$$

with F_H denoting the forecasted demand over the whole planning horizon H.

3.1.3.1.2 Implementation

In this section, a potential implementation framework is described in order to establish an inventory control procedure in the cloud-based planning system.

As described above, the *Secure Collaborative Forecasting Module* has to determine two important characteristics about future demand for each component c:

- Estimate for the total demand over the planning horizon
- For each period t, the cumulative distribution function of demand over the protection interval R

This data is needed as an input to the *Spare Parts Management Module*. With the forecasted total demand over the planning horizon H and $\Phi_{CFD_R}(\cdot)$, we calculate the efficient re-order point per component as well as the order quantity Q_c following the procedures described above. Table 29 presents an overview on the interfaces of the *Spare Parts Management Module*.

Input parameters	Forecasted demand for the whole planning horizon H per component c	$F_{H,c}$
	Cumulative distribution function over forecasted demand for protection period R per component c	$\Phi_{CFD_{R,c}}(\cdot)$
	Supplier lead time per component	L_c
	Inventory level per component	$I_{t,c}$
Output parameters	Re-order point r_t for period t per component c	$r_{t,c}$
	Quantity Q to order in period t if demand is below r_t per component c	Q_c
	Actually ordered quantity in period t per component c	$Q_{a,t,c}$

Table 29: Interfaces of the Spare Parts Management Module

The following algorithm illustrates a possible implementation approach.

Algorithm 6: Inventory control algorithm

For $c = 1{:}C$ *// For each component c, calculate Q_c for the whole planning horizon H*
  $F_{H,c} \leftarrow getExpDemand(H, c)$
  $Q_c \leftarrow \sqrt{\frac{2KF_{H,c}}{h_c H}}$
end
for $c = 1{:}C$ *// at the end of each period t, calculate $r_{t+1,c}$ for each component c*
  for $t = 1{:}H$
    $I_{t,c} \leftarrow getInventory(c)$
    If $I_{t,c} < r_{t,c}$ **then** *// re-order Q_c if inventory has fallen below $r_{t,c}$*
      order($Q_{a,t,c}$, c)
    end
    $L_c \leftarrow getLeadTime(c)$
    $\Phi_{CFD_{R,c}} \leftarrow getExpDemandDistribution(R, c)$

$$r_{t+1,c} \leftarrow \Phi^{-1}_{CFD_{R,c}}(SL)$$

end

end

First, the aggregated forecast over the whole planning horizon H is required. Based on the fixed order and unit holding costs, the optimal order quantity for each component Q_c is determined for the planning horizon. Consequently, at the end of each period, the current values for the replenishment lead times and the inventory level per component are updated. If the inventory level has fallen below $r_{t,c}$, an order $Q_{a,t,c}$ is released directly to the supplier. Next, the cumulative distribution function for the demand is updated by the *Secure Collaborative Forecasting Module*. Based on the new forecasts, the optimal re-order point $r_{t+1,c}$ for the subsequent period is calculated for each component as described in section 3.1.3.1.1. Figure 24 provides an overview on the interactions between *Secure Collaborative Forecasting Module* and the *Spare Parts Management Module* for one component.

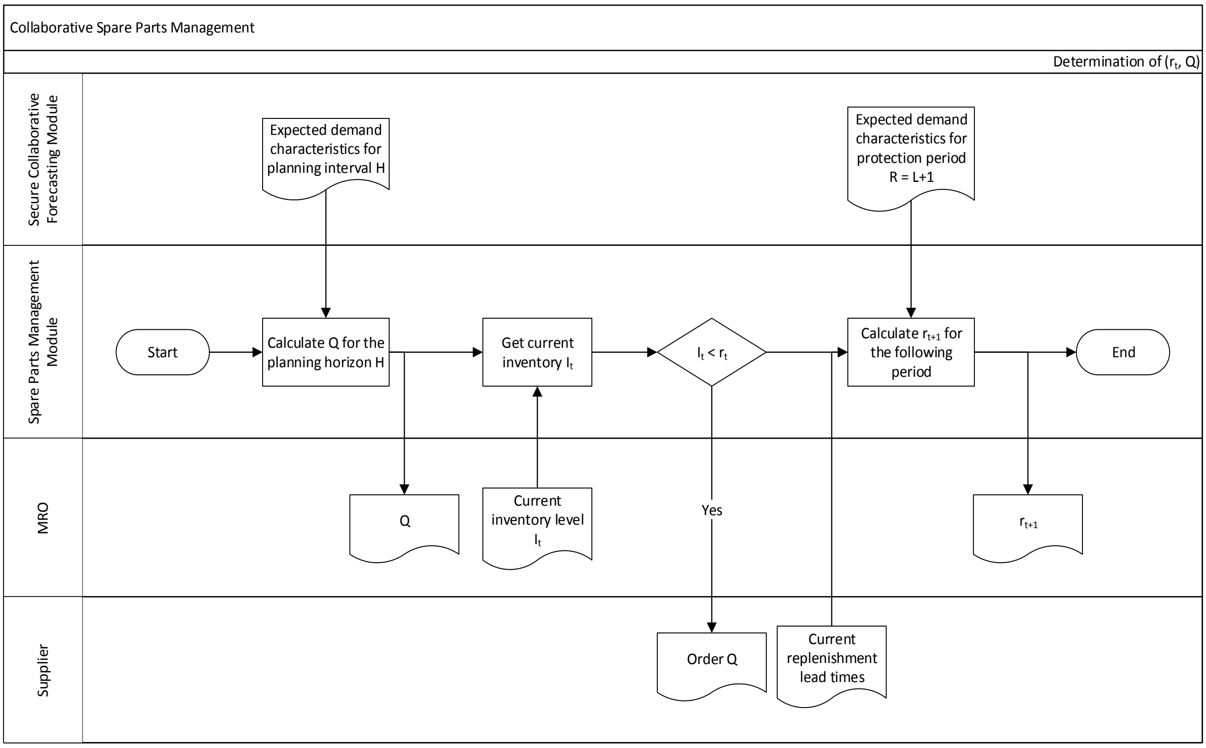


Figure 24: Process overview on the Spare Parts Management Module

3.1.3.1.3 Data sensitivity assessment

An important aspect for a cloud-based *Spare Parts Management Module* is the assessment of the required protection levels for critical information that is used in order to steer the MRO's inventories. Whereas parameters such as the replenishment lead times might not reveal information that could have a negative impact on the respective party, several parameters are highly confidential as they might disclose information about the MRO's internal processes and therefore require high protection levels via strong encryption. Table 30 provides a general assessment of the different parameters that are implemented within the *Spare Parts Management Module*.

Variable	Description	Party concerned	Security requirements
K_c	Fixed costs incurred per order of a component	MRO	High security requirements – Cost data is very sensitive to the MRO due to its direct impact on the MRO's bargaining position with customers and potential information for competitors
h_c	Unit-variable holding costs per component	MRO	High security requirements – Cost data is very sensitive to the MRO due to its direct impact on the MRO's bargaining position with customers and potential information for competitors
$F_{R,c}$	Cumulated forecasted demand over R per component	MRO	High security requirements – Spare parts demand data is sensitive to the airlines as it might reveal information about their resource planning and scheduling to competitors
$CFD_{R,c}$	Cumulated distribution function of the forecasted demand over R per component	Customers	High security requirements – Spare parts demand data is sensitive to the airlines as it might reveal information about their resource planning and scheduling to competitors
$I_{t,c}$	Current inventory level per component	MRO	High security requirements – Spare parts inventory data is very sensitive to the MRO
$r_{k,c}$	Re-order point per component and period	MRO	Medium security requirements – The re-order point discloses information about inventory steering of the MRO
Q_c	Order quantity per component	MRO	Medium security requirements – The order quantity discloses information about inventory steering of the MRO
L_c	Fixed replenishment cycle time per component	MRO/ supplier	Low security requirements – The revelation of the replenishment cycle times should not entail a considerable negative impact on the MRO

Table 30: Criticality assessment of the parameters within the Spare Parts Management Module

3.1.3.1.4 Adjustments and potential extensions of the model

Especially for the case of intermittent demand, the employment of order-up-to-level (s,S) policies is widely spread in practice. However, the determination of optimal parameters for such a policy depends on restrictive assumptions concerning demand distributions. For this reason, we employ a (r_t,Q) policy that is dynamically adaptable to the actual forecasts over the planning horizon. However, in order to account for undershooting of the re-order point r_t due to the periodic review characteristic of our policy, several ways exist to transform the model into an approximated equivalent (s,S) policy. A practical way to set parameters under the (s,S) inventory policy is to set s:= r and S:= s + Q. A more elaborate way of calculating the parameters is Naddor's Heuristic (Naddor 1975) which was proven to achieve near-optimal results compared to the optimal parameters.

A further characteristic of inventory management deals with stochastic replenishment cycle times. A considerable body of literature on inventory control treats lead times as deterministic constants such as in our model. However, in the spare parts management context, replenishment lead times for complex and expensive parts that are re-ordered in irregular cycles at the supplier might be long and can therefore underlie uncertainty. Due to the practical concerns that are related to lead times, a possible extension to the proposed model is the integration of stochastic replenishment lead times. For the case of a stochastic lead time with mean μ_L and standard deviation σ_L, the optimal values of r_t can then be computed numerically by solving:

$$\sum_i P(L = L_i) \cdot \Phi_{CFD_{L_i+1}}(r_t) = SL$$

If we consider the case of very expensive and slow-moving parts, high unit holding costs and long inter-demand cycles entail very high costs of capital commitment for overstocked units. Although the dynamic inventory model adjusts safety stocks to updated information about future demands and therefore outperforms traditional, static policies, its overall efficiency still relies heavily on the forecast accuracy. If no information about future demands is available or if the forecasts show a strong bias, we would recommend to procure those expensive, slow-moving parts rather on a part-by-part basis. However, based on the assumption of sufficiently high forecasting accuracy and a targeted customer service level, our proposed model would efficiently provide stocks in order to secure this service level for both fast-moving and slow-moving parts.

3.1.3.1.5 Conclusion and further references

We presented a concept for a dynamic inventory management model that can be implemented securely within the cloud. Based on a given forecast as well as a corresponding demand distribution probability function, optimal policy parameters are determined that consider underlying non-stationary demand patterns. In order to support the implementation in a cloud setting, a relatively inexpensive numerical solution procedure is employed for the calculation of the policy parameters.

Dynamic inventory control models were first developed by a stream in the late 1950s (Scarf 1959; Karlin 1960). However, their models focused on optimal solutions entailing very restrictive underlying assumptions or computationally expensive calculations which make them unattractive for implementation in a practitioner's context. Furthermore, they minimize total inventory costs including unit holding and backlog costs whereas in most business settings, a pre-determined service level is more appropriate.

During the last two decades, a body of research was devoted to dynamic inventory models based on forecasted demand (Graves 1999; Chen et al. 2000). However, they develop order-up-to-level policies on very restrictive assumptions. Furthermore, practical issues such as lead time uncertainty are neglected in those models.

Most of the literature in inventory control focuses on securing a service level over a pre-known, deterministic lead time. However, as mentioned above, lead time uncertainty might become an important issue**.** Eppen und Martin (1988) developed a model incorporating both non-stationary demand and lead time uncertainty in order to compute optimal safety stocks under a fill rate service level constraint.

Our model is based on a simplified version of the work of Babaï et al. (2009). They propose a model incorporating stochastic lead times with non-stationary demand under a service level constraint and numerical parameter determination. However, in order to account for an efficient solution procedure, we ignored stochastic lead times that were considered in the original model. Compared to most of the authors mentioned before, their model assumes forecast data and demand probability distribution as exogenous data as presented in 3.1.3.1.1. Babaï et al. (2009) have shown that their dynamic model with service level constraints can achieve comparable results to a static inventory model in terms of achieved service level whilst achieving significant reductions in inventory costs.

3.1.3.2 Maintenance arrival slot scheduling[14]

Currently, the MRO service provider has no information regarding future engine arrivals for overhaul. With the collaborative maintenance demand forecasting mechanism implemented in the cloud-based collaborative planning system (see Figure 25), we can use information regarding engine arrival times to schedule their arrivals such that the variability of their interarrival times is minimized.

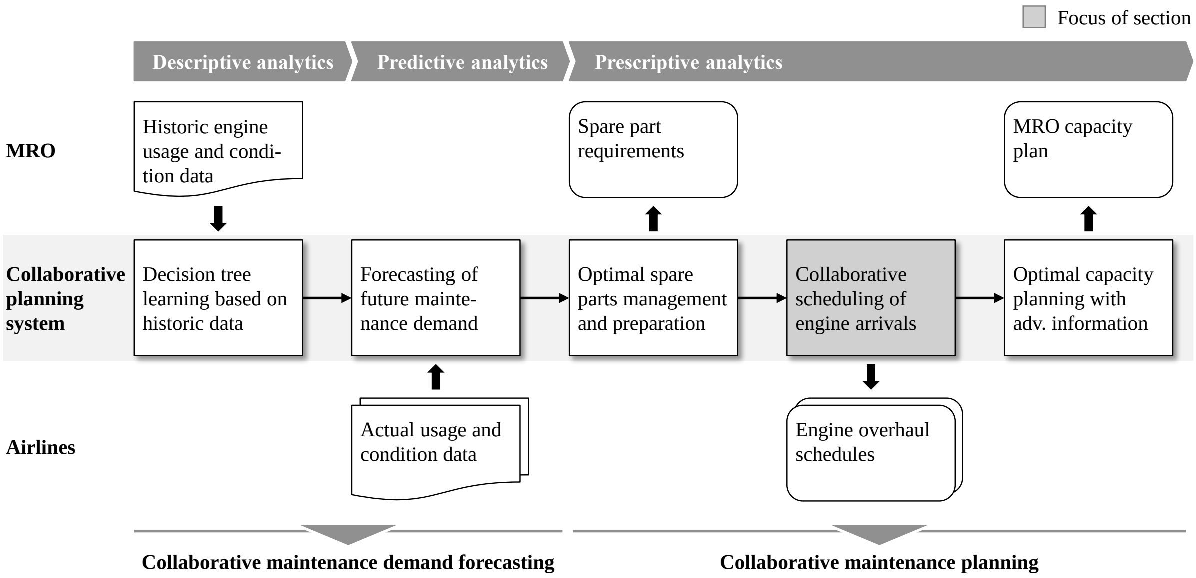


Figure 25: CPS for collaborative maintenance management

As explained in Section 3.1.3.3, this reduction in variability will drastically reduce capacity costs in the engine overhaul production network due.

Figure 26 illustrates the scheduling mechanism. The first timeline shows the stochastic interarrival times as they are observed in the current situation. The second timeline illustrates the scheduling of the arrival slots such that the time between two arrivals is constant. The original arrivals can be moved within the time window Δt in order to reduce interarrival time variability.

[14] Main author: Julian Kurz (UWUERZ)

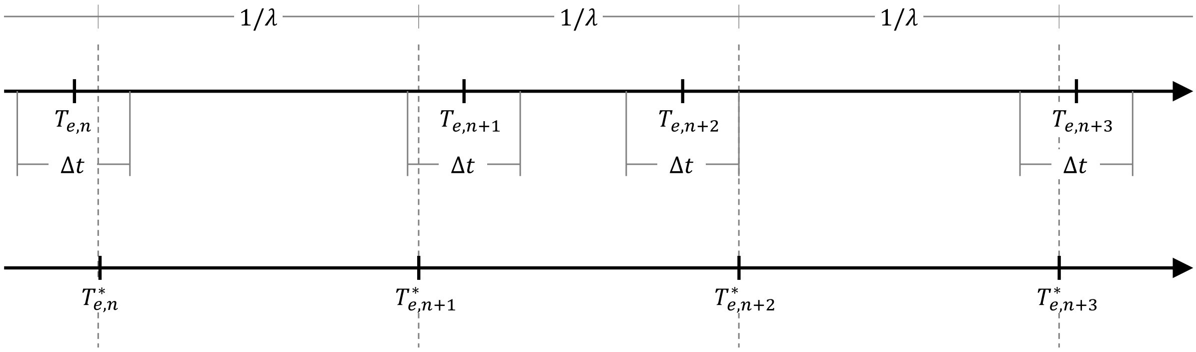


Figure 26: Illustration of engine arrival slot scheduling.

For the remainder of this chapter, the following notation is used:

$e, \mathcal{E}, E$:	Engine types, set of engine types $\mathcal{E} = \{1, \ldots, E\}$
p_e:	Probability that an engine of type e needs to be overhauled
n_e:	Total number of engines of type e
X_e:	Random variable representing the number of arrivals per time period
λ_e:	Arrival rate of type e engines
t_e:	Time between two subsequent arrivals of engines of type e
Y_e:	Random variable representing the interarrival times
$T_{e,n}$:	Arrival time slot of nth engine of type e
$T^*_{e,n}$:	Scheduled arrival time slot of nth engine of type e
Δt_e:	Symmetric scheduling time window for engines of type e
Y^*_e:	Random variable representing the scheduled interarrival times
$\delta(x)$:	Dirac delta function
μ_e	Mean of the interarrival times of engine type e
ca_e:	Squared coefficient of variation (SCV) of type e engines
$\widetilde{ca}_e$:	Updated SCV after scheduling for type e engines

3.1.3.2.1 Current engine arrival characteristics

For each engine type $e \in \mathcal{E} = \{1, \ldots, E\}$ there is a finite total number of engines n_e used by all customers. The probability that an engine of a type needs to be overhauled is given by p_e. Therefore, the number of engines from one engine type k_e arriving for overhaul within one week can be represented by a binomially distributed random variable X_e.

$$f(k_e; n_e, p_e) = \Pr(X_e = k_e) = \binom{n_e}{k_e} p_e^{k_e} (1 - p_e)^{n-k}$$

For large n_e and small p_e (a rule of thumb suggests $n_e \geq 20$ and $p_e \leq 0.05$ or $n \geq 100$ and $n_e p_e \leq 10$) as it is the case in the engine overhaul scenario, the binomial distribution can be approximated by a Poisson distribution with mean $\lambda_e = n_e p_e$.

$$f(k_e; \lambda_e) = \Pr(X_e = k_e) = \frac{\lambda_e^k \exp\{-\lambda_e\}}{k_e!}$$

Therefore, engine arrivals can be described by a homogeneous Poisson process with parameter λ_e. This means that the time t_e between two subsequent arrivals of two engines of the same type, i.e., the interarrival time, can be described by an exponentially distributed random variable Y_e.

$$f(t_e; \lambda_e) = \begin{cases} \lambda_e \exp\{-\lambda_e t_e\} & t_e \geq 0 \\ 0 & t_e < 0 \end{cases}$$

The variance of an exponentially distributed random variable Y_e with mean $1/\lambda_e$ is given by $\mathrm{Var}(Y_e) = \frac{1}{\lambda_e^2}$. Therefore, the squared coefficient of variation which is the relevant parameter when it comes to capacity planning for the production network (see Section 3.1.3.3) results to

$$\mathrm{SCV}(Y_e) = ca_e = \frac{\mathrm{Var}(Y_e)}{\mathrm{E}(Y_e)^2} = \frac{\lambda_e^2}{\lambda_e^2} = 1.$$

The purpose of this section is to determine a mechanism to reduce this ratio.

3.1.3.2.2 Cloud-based collaborative scheduling of engine arrival slots

From the collaborative forecasting mechanism we obtain a sequence of arrival times for each engine type (without information regarding the customer who the engine belongs to).

$$\ldots, T_{e,n-1}, T_{e,n}, T_{e,n+1}, \ldots$$

Let's now define a time window Δt_e symmetrically distributed around each forecasted arrival time. Then, based on the previous engine's arrival time $T_{e,n}$ we can determine a new time slot $T^*_{e,n+1} \in [T_{e,n+1} - \frac{\Delta t_e}{2}, T_{e,n+1} + \frac{\Delta t_e}{2}]$ for the $(n+1)$th engine arrival. Ideally, in order to minimize interarrival time variability, we choose the scheduled time slot such that the interarrival time $T^*_{e,n+1} - T_{e,n}$ is as close as possible to the mean interarrival time $\frac{1}{\lambda_e}$. The following algorithm shows how this can be implemented practically.

Algorithm 1: Engine arrival scheduling

1: **If** $T_{e,n+1} + \frac{\Delta t_e}{2} < T_{e,n} + \frac{1}{\lambda_e}$

2: $\quad T^*_{e,n+1} \leftarrow T_{e,n+1} + \frac{\Delta t_e}{2}$

3: **else if** $T_{e,n+1} - \frac{\Delta t_e}{2} > T_{e,n} + \frac{1}{\lambda_e}$

4: $\quad T^*_{e,n+1} \leftarrow T_{e,n+1} - \frac{\Delta t_e}{2}$

5: **else**

6: $\quad T^*_{e,n+1} \leftarrow T_{e,n} + \frac{1}{\lambda_e}$

7: **end**

Proposition 1. Depending on the width of the time window Δt_e, the probability that the scheduled arrival slot $T^*_{e,n+1}$ coincides with the optimal arrival time $T_{e,n} + \frac{1}{\lambda_e}$ is given by

$$\Pr\left(T^*_{e,n+1} = T_{e,n} + \frac{1}{\lambda_e}\right) = \exp\left\{-\lambda_e \max\left[0; \frac{1}{\lambda_e} - \frac{\Delta t_e}{2}\right]\right\} - \exp\left\{-\frac{\lambda_e \Delta t_e}{2} - 1\right\}.$$

Proof. The case that $T^*_{e,n+1} = T_{e,n} + \frac{1}{\lambda_e}$ is given if $T_{e,n+1} \in [T_{e,n} + \frac{1}{\lambda_e} - \frac{\Delta t_e}{2}, T_{e,n} + \frac{1}{\lambda_e} + \frac{\Delta t_e}{2}]$. Therefore, we know that

$$\Pr\left(T^*_{e,n+1} = T_{e,n} + \frac{1}{\lambda_e}\right) = \Pr\left(t_e \leq \frac{1}{\lambda_e} + \frac{\Delta t_e}{2}\right) - \Pr\left(t_e \leq \frac{1}{\lambda_e} - \frac{\Delta t_e}{2}\right).$$

If we evaluate this we obtain

$$\begin{aligned}
\Pr\left(T^*_{e,n+1} = T_{e,n} + \frac{1}{\lambda_e}\right) &= \Pr\left(t_e \leq \frac{1}{\lambda_e} + \frac{\Delta t_e}{2}\right) - \Pr\left(t_e \leq \frac{1}{\lambda_e} - \frac{\Delta t_e}{2}\right) \\
&= \int_0^{\frac{1}{\lambda_e} + \frac{\Delta t_e}{2}} \lambda_e \exp\{-\lambda_e t_e\}\, \mathrm{d}t_e - \int_0^{\max\left[0; \frac{1}{\lambda_e} - \frac{\Delta t_e}{2}\right]} \lambda_e \exp\{-\lambda_e t_e\}\, \mathrm{d}t_e \\
&= \left[-\exp\{-\lambda_e t_e\}\right]_0^{\frac{1}{\lambda_e} + \frac{\Delta t_e}{2}} - \left[-\exp\{-\lambda_e t_e\}\right]_0^{\max\left[0; \frac{1}{\lambda_e} - \frac{\Delta t_e}{2}\right]} \\
&= -\exp\left\{-\lambda_e \left(\frac{1}{\lambda_e} + \frac{\Delta t_e}{2}\right)\right\} + 1 + \exp\left\{-\lambda_e \max\left[0; \frac{1}{\lambda_e} - \frac{\Delta t_e}{2}\right]\right\} - 1 \\
&= \exp\left\{-\lambda_e \max\left[0; \frac{1}{\lambda_e} - \frac{\Delta t_e}{2}\right]\right\} - \exp\left\{-\frac{\lambda_e \Delta t_e}{2} - 1\right\}.
\end{aligned}$$

The $\max[0;\cdot]$ operator is needed to ensure that $t_e \geq 0$. This concludes the proof. □

As we see in Figure 27, the probability that the optimal interarrival time is chosen increases with the time window width.

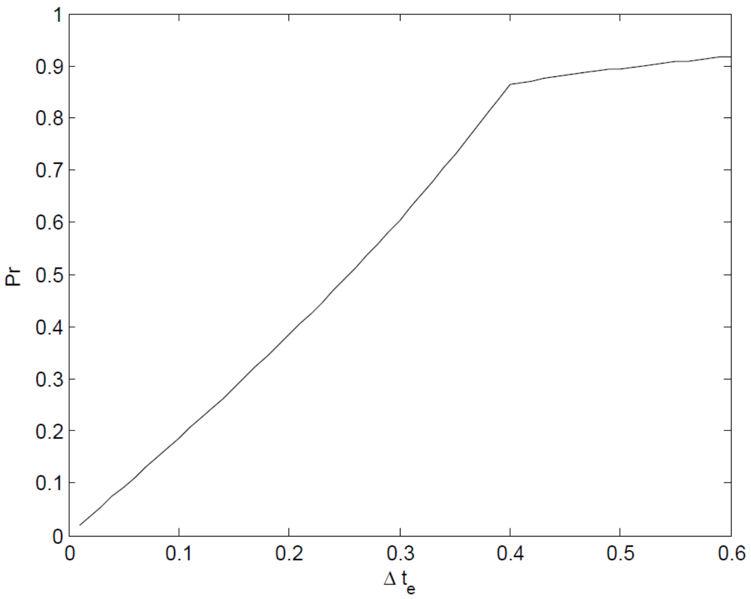


Figure 27: Probability of the scheduled arrival slot to coincide with the optimal arrival slot ($\lambda_e = 5$).

We also see that it there is a kink and it increases less if $\Delta t_e > 0.4$ or $\frac{\Delta t_e}{2} > \frac{1}{\lambda_e}$. This is due to the $\max[0;\cdot]$ operator.

In the next step we want to compute the improved interarrival time squared coefficient of variation in dependence of the time window width, $\widetilde{ca}_e(\Delta t_e) < ca_e$. Therefore, we need to find the probability density function (pdf) and cumulative distribution function (cdf) in order to compute the variance of the interarrival times of the scheduled arrival slots.

We start by determining the cdf of the scheduled interarrival time random variable Y_e^*. It is clear that the cdf will be 0 for $t_e \in \left[0, \frac{\Delta t_e}{2}\right]$. This means, the original cdf of the exponentially distributed random variable representing the original interarrival times Y_e is translated by $-\frac{\Delta t_e}{2}$. On the other hand, for $t_e > \frac{1}{\lambda_e}$ the cdf is translated by $+\frac{\Delta t_e}{2}$. I.e., we obtain for the cdf of Y_e^*:

$$F_{Y_e^*}(t_e;\lambda_e) = \begin{cases} 1 - \exp\left\{-\lambda_e\left(t_e + \frac{\Delta t_e}{2}\right)\right\}, & t_e > \frac{1}{\lambda_e} \\ 1 - \exp\left\{-\lambda_e\left(t_e - \frac{\Delta t_e}{2}\right)\right\}, & \frac{\Delta t_e}{2} \le t_e \le \frac{1}{\lambda_e} \\ 0, & t_e < \frac{\Delta t_e}{2} \end{cases}$$

The cdf is illustrated for different Δt_e in Figure 28. The blue, green, red and turquoise colors show the cdf's for $\Delta t_e = \frac{1}{4\lambda_e}$, $\Delta t_e = \frac{1}{2\lambda_e}$, $\Delta t_e = \frac{3}{4\lambda_e}$ and $\Delta t_e = \frac{1}{\lambda_e}$ with $\lambda_e = 5$, respectively.

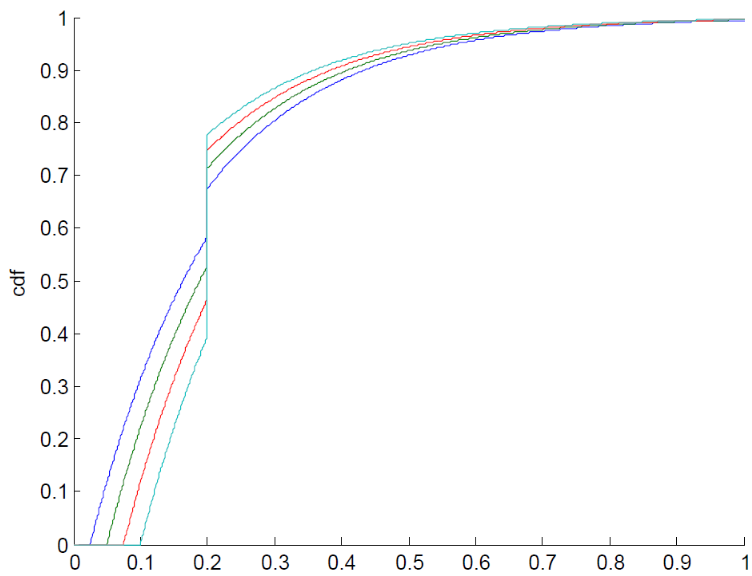


Figure 28: Cumulative distribution function for different time window widths.

As we see in Figure 28, there is a step in the cdf at $t_e = \frac{1}{\lambda_e}$. This is clear if we remember that $T_{e,n+1}^* = T_{e,n} + \frac{1}{\lambda_e}$ if $T_{e,n+1} \in [T_{e,n} + \frac{1}{\lambda_e} - \frac{\Delta t_e}{2}, T_{e,n} + \frac{1}{\lambda_e} + \frac{\Delta t_e}{2}]$. Therefore, the height of the step corresponds exactly to the probability that the scheduled arrival slot coincides with the optimal arrival slot, see Proposition 1.

On the other hand, as

$$f_{Y_e^*}(t_e;\lambda_e) = \frac{\mathrm{d}}{\mathrm{d}x} F_{Y_e^*}(x)$$

we know that there must be scaled Dirac delta function in the pdf of Y_e^* at $\frac{1}{\lambda_e}$. The Dirac delta function is defined as

$$\delta(x) = \begin{cases} \infty, & x = 0 \\ 0, & x \neq 0 \end{cases}$$

with

$$\int_{-\infty}^{\infty} \delta(x)\mathrm{d}x = 1.$$

The scaling property tells us that

$$\delta(\alpha x) = \frac{1}{|\alpha|}\delta(x)$$

and accordingly

$$\int_{-\infty}^{\infty} \delta(\alpha x)\mathrm{d}x = \frac{1}{|\alpha|}.$$

Using this definition, we can write the pdf as

$$f_{Y_e^*}(t_e,\lambda_e) = \begin{cases} \lambda_\mathrm{e}\exp\left\{-\lambda_\mathrm{e}\left(t_e + \frac{\Delta t_e}{2}\right)\right\}, & t_e > \frac{1}{\lambda_e} \\ \delta\left[\alpha\left(t_e - \frac{1}{\lambda_e}\right)\right], & t_e = \frac{1}{\lambda_e} \\ \lambda_\mathrm{e}\exp\left\{-\lambda_\mathrm{e}\left(t_e - \frac{\Delta t_e}{2}\right)\right\}, & \frac{\Delta t_e}{2} \leq t_e < \frac{1}{\lambda_e} \\ 0, & t_e < \frac{\Delta t_e}{2} \end{cases}$$

where

$$\alpha = \exp\left\{-\lambda_e \max\left[0; \frac{1}{\lambda_\mathrm{e}} - \frac{\Delta t_e}{2}\right]\right\} - \exp\left\{-\frac{\lambda_e \Delta t_e}{2} - 1\right\}.$$

The pdf is illustrated for different Δt_e in Figure 28. The blue, green, red and turquoise colors show the cdf's for $\Delta t_e = \frac{1}{4\lambda_e}$, $\Delta t_e = \frac{1}{2\lambda_e}$, $\Delta t_e = \frac{3}{4\lambda_e}$ and $\Delta t_e = \frac{1}{\lambda_e}$ with $\lambda_e = 5$, respectively.

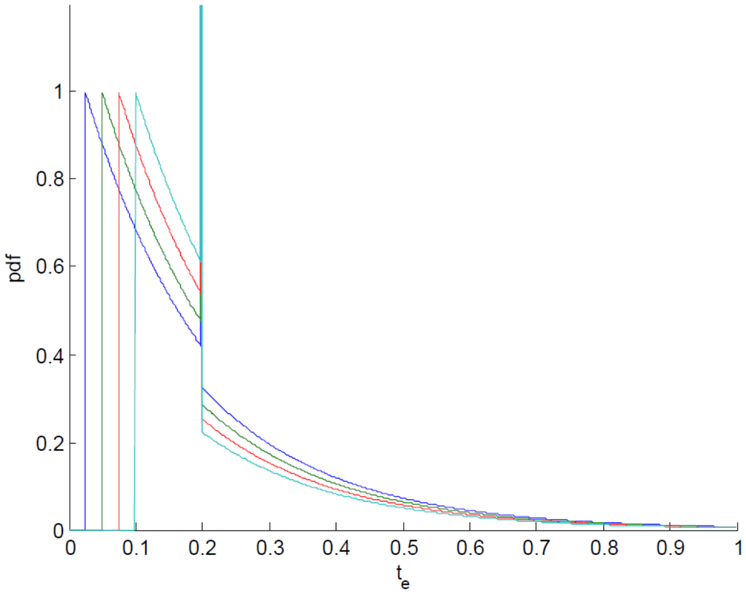


Figure 29: Probability density function for different time window widths.

Knowing the pdf we can now compute the variance and therefore the squared coefficient of variation.

Assumption 1. *The half width of the time window Δt_e is smaller than the mean of the exponentially distributed random variable Y_e, $\frac{\Delta t_e}{2} < \frac{1}{\lambda_e}$.*

With this assumption we can conclude that $\mathrm{E}(Y_e^*) = \mathrm{E}(Y_e)$. Also we find the following analytical insight.

Proposition 2. *The variance of the random variable representing the interarrival times for the scheduled engine arrivals is given by*

$$\mathrm{Var}(Y_e^*) = \frac{1}{\lambda_e^2} + \frac{\Delta t_e^2}{4} + \frac{2}{\lambda_e^2}\left[\exp\left\{-\frac{\lambda_e \Delta t_e}{2} - 1\right\} - \exp\left\{\frac{\lambda_e \Delta t_e}{2} - 1\right\}\right].$$

Proof. The variance of Y_e^* is computed as $\mathrm{Var}(Y_e^*) = \mathrm{E}[(Y_e^* - \mu_e)^2]$ with $\mu_e = \frac{1}{\lambda_e}$.

$$\mathrm{Var}(Y_e^*) = \int_{-\infty}^{\infty} \left(t_e - \frac{1}{\lambda_e}\right)^2 f_{Y_e^*} \mathrm{d}t_e = \int_{-\infty}^{\infty} \left(t_e^2 - \frac{2t_e}{\lambda_e} + \frac{1}{\lambda_e^2}\right) f_{Y_e^*} \mathrm{d}t_e$$

$$= \underbrace{\int_{\frac{\Delta t_e}{2}}^{\frac{1}{\lambda_e}} \left(t_e^2 - \frac{2t_e}{\lambda_e} + \frac{1}{\lambda_e^2}\right) \lambda_e \exp\left\{-\lambda_e \left(t_e - \frac{\Delta t_e}{2}\right)\right\} dt_e}_{(i)}$$

$$+ \underbrace{\int_{-\infty}^{\infty} \left(t_e - \frac{1}{\lambda_e}\right)^2 \delta\left[\alpha\left(t_e - \frac{1}{\lambda_e}\right)\right] dt_e}_{(ii)} + \underbrace{\int_{\frac{1}{\lambda_e}}^{\infty} \left(t_e^2 - \frac{2t_e}{\lambda_e} + \frac{1}{\lambda_e^2}\right) \lambda_e \exp\left\{-\lambda_e \left(t_e + \frac{\Delta t_e}{2}\right)\right\} dt_e}_{(iii)}$$

In order to optimize readability we evaluate the three integrals consecutively.

(i)

$$\int_{\frac{\Delta t_e}{2}}^{\frac{1}{\lambda_e}} \left(t_e^2 - \frac{2t_e}{\lambda_e} + \frac{1}{\lambda_e^2}\right) \lambda_e \exp\left\{-\lambda_e\left(t_e - \frac{\Delta t_e}{2}\right)\right\} dt_e$$

$$= \lambda_e \int_{\frac{\Delta t_e}{2}}^{\frac{1}{\lambda_e}} t_e^2 \exp\left\{-\lambda_e\left(t_e - \frac{\Delta t_e}{2}\right)\right\} \mathrm{d}t_e - 2\int_{\frac{\Delta t_e}{2}}^{\frac{1}{\lambda_e}} t_e \exp\left\{-\lambda_e\left(t_e - \frac{\Delta t_e}{2}\right)\right\} \mathrm{d}t_e$$

$$+ \frac{1}{\lambda_e} \int_{\frac{\Delta t_e}{2}}^{\frac{1}{\lambda_e}} \exp\left\{-\lambda_e\left(t_e - \frac{\Delta t_e}{2}\right)\right\} \mathrm{d}t_e$$

$$= \left[-t_e^2 \exp\left\{-\lambda_e\left(t_e - \frac{\Delta t_e}{2}\right)\right\}\right]_{\frac{\Delta t_e}{2}}^{\frac{1}{\lambda_e}} + \frac{1}{\lambda_e}\left[-\frac{1}{\lambda_e}\exp\left\{-\lambda_e\left(t_e - \frac{\Delta t_e}{2}\right)\right\}\right]_{\frac{\Delta t_e}{2}}^{\frac{1}{\lambda_e}}$$

$$= \frac{1}{\lambda_e^2} + \frac{\Delta t_e^2}{4} - \frac{2}{\lambda_e^2}\exp\left\{-1 + \frac{\lambda_{e\Delta t_e}}{2}\right\}$$

(ii)

$$\int_{-\infty}^{\infty} \left(t_e - \frac{1}{\lambda_e}\right)^2 \delta\left[\alpha\left(t_e - \frac{1}{\lambda_e}\right)\right] dt_e$$

$$= \left(\frac{1}{\lambda_e} - \frac{1}{\lambda_e}\right)^2 = 0$$

Since

$$\int_{-\infty}^{\infty} f(x)\delta[\alpha(x-b)]\mathrm{d}x = \frac{1}{|\alpha|} f(b)$$

(iii)

$$\int_{\frac{1}{\lambda_e}}^{\infty} \left(t_e^2 - \frac{2t_e}{\lambda_e} + \frac{1}{\lambda_e^2}\right) \lambda_e \exp\left\{-\lambda_e\left(t_e + \frac{\Delta t_e}{2}\right)\right\} dt_e$$

$$= \lambda_e \int_{\frac{1}{\lambda_e}}^{\infty} t_e^2 \exp\left\{-\lambda_e\left(t_e + \frac{\Delta t_e}{2}\right)\right\} \mathrm{d}t_e - 2\int_{\frac{1}{\lambda_e}}^{\infty} t_e \exp\left\{-\lambda_e\left(t_e + \frac{\Delta t_e}{2}\right)\right\} \mathrm{d}t_e$$

$$+ \frac{1}{\lambda_e} \int_{\frac{1}{\lambda_e}}^{\infty} \exp\left\{-\lambda_e\left(t_e + \frac{\Delta t_e}{2}\right)\right\} \mathrm{d}t_e$$

$$= \left[-t_e^2 \exp\left\{-\lambda_e\left(t_e + \frac{\Delta t_e}{2}\right)\right\}\right]_{\frac{1}{\lambda_e}}^{\infty} + \frac{1}{\lambda_e}\left[-\frac{1}{\lambda_e}\exp\left\{-\lambda_e\left(t_e + \frac{\Delta t_e}{2}\right)\right\}\right]_{\frac{1}{\lambda_e}}^{\infty}$$

$$= \frac{2}{\lambda_e}\exp\left\{-1 - \frac{\lambda_e \Delta t_e}{2}\right\}$$

Therefore, combining the three intermediate results we obtain

$$\mathrm{Var}(Y_e^*) = (i) + (ii) + (iii)$$
$$= \frac{1}{\lambda_e^2} + \frac{\Delta t_e^2}{4} + \frac{2}{\lambda_e^2}\left[\exp\left\{-\frac{\lambda_e \Delta t_e}{2} - 1\right\} - \exp\left\{\frac{\lambda_e \Delta t_e}{2} - 1\right\}\right].$$

This concludes the proof. □

Using the definition of the variance of the interarrival time distribution for the scheduled arrival slots we can compute the updated squared coefficient of variation $\widetilde{ca}_e$. Figure 30 illustrates the slope of $\widetilde{ca}_e$ for $\lambda_e = 5$ and $\Delta t_e \in \left(0, \frac{2}{\lambda_e}\right)$ which is strictly monotonically decreasing with increasing Δt_e.

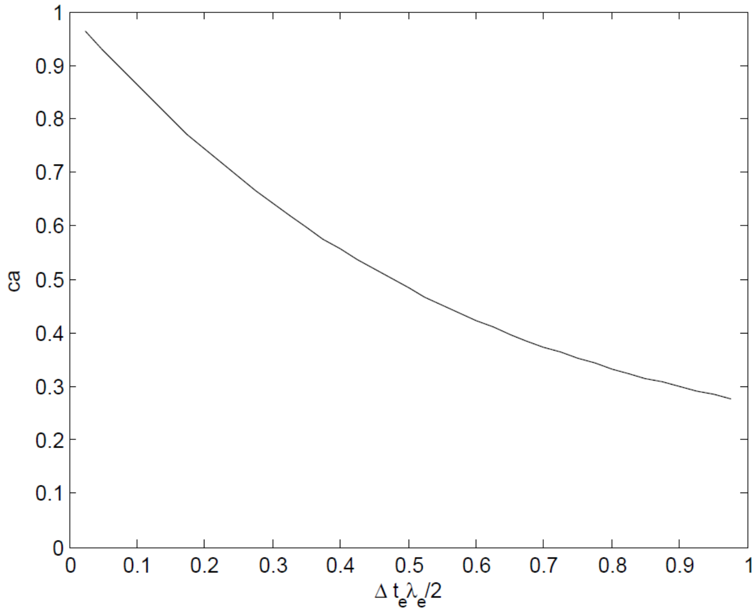


Figure 30: Updated interarrival time variability in dependence of relative window width.

3.1.3.2.3 Implementation of scheduling algorithm

The following figure illustrates the scheduling process in order to provide guidance regarding implementation in the cloud and interfaces to the involved parties.

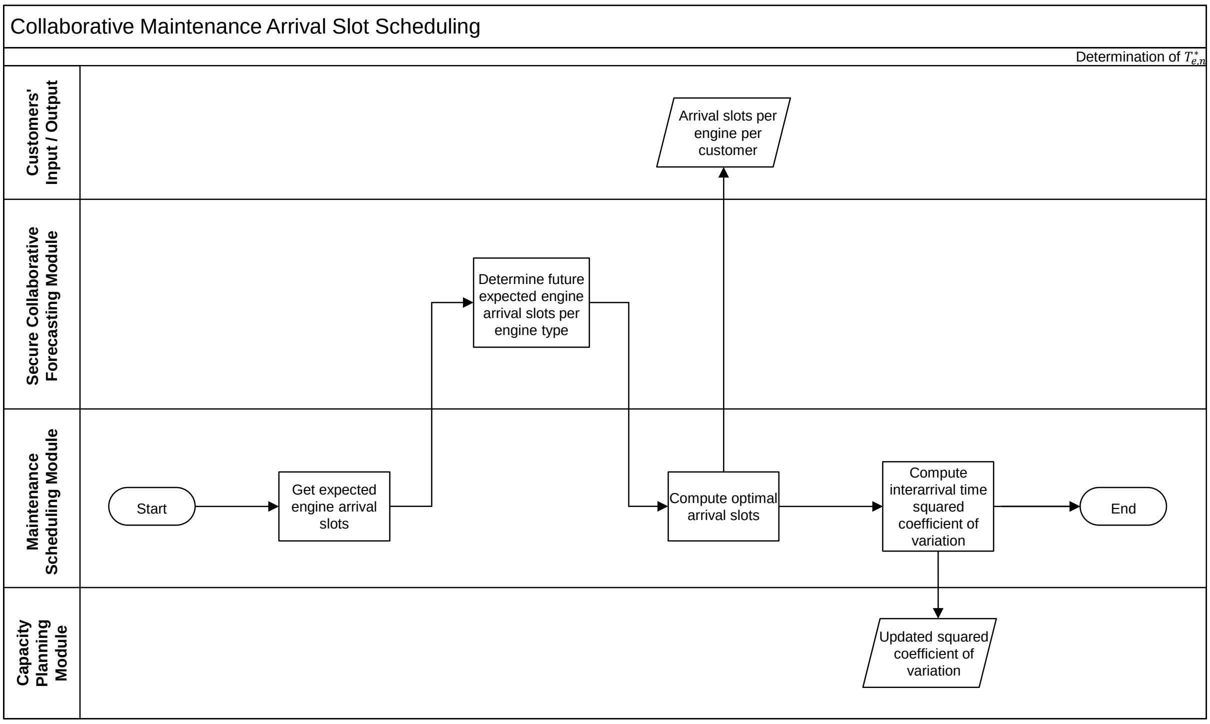


Figure 31: Engine Arrival Slot Scheduling module.

3.1.3.2.4 Data sensitivity assessment

Variable	Description	Party concerned	Security requirements
$T_{e,n+1}$	Time point of $(n+1)$th engine delivery of engine type e	Customers	High security requirements – engine arrival times are private data of the airlines and need to be protected
$\widetilde{ca}_e$	Updated SCV of the interarrival times	customers	The resulting updated squared coefficient of the interarrival times is not critical and can be shared with the MRO service provider

Table 31: Data sensitivity assessment for collaborative engine arrival slot scheduling.

3.1.3.2.5 Conclusion and further references

We have developed a mechanism to schedule aircraft engine arrivals such that the squared coefficient of variation of the interarrival times is reduced.

Of course, assuming a symmetric and continuous time window around $T_{e,n+1}$ is a rather strong assumption. Nevertheless, it enables us to show the effects on interarrival time variability. Furthermore, it is now straight forward to update Algorithm 1 to suit cases where not a continuous time window is given but a discrete set of alternative arrival slots. If we assume a set of M possible arrival slots for engine $n+1$, $\{T_{e,n+1,m}\}_{m=1}^{M}$, we can define the discrete set engine arrival scheduling algorithm as follows.

Algorithm 2: Discrete set engine arrival scheduling

1: **For** $m = 1{:}M$

$$dist(m) \leftarrow \left| T_{e,n} + \frac{1}{\lambda_e} - T_{e,n+1,m} \right|$$

end

$m^* \leftarrow \arg\min_m dist(m)$

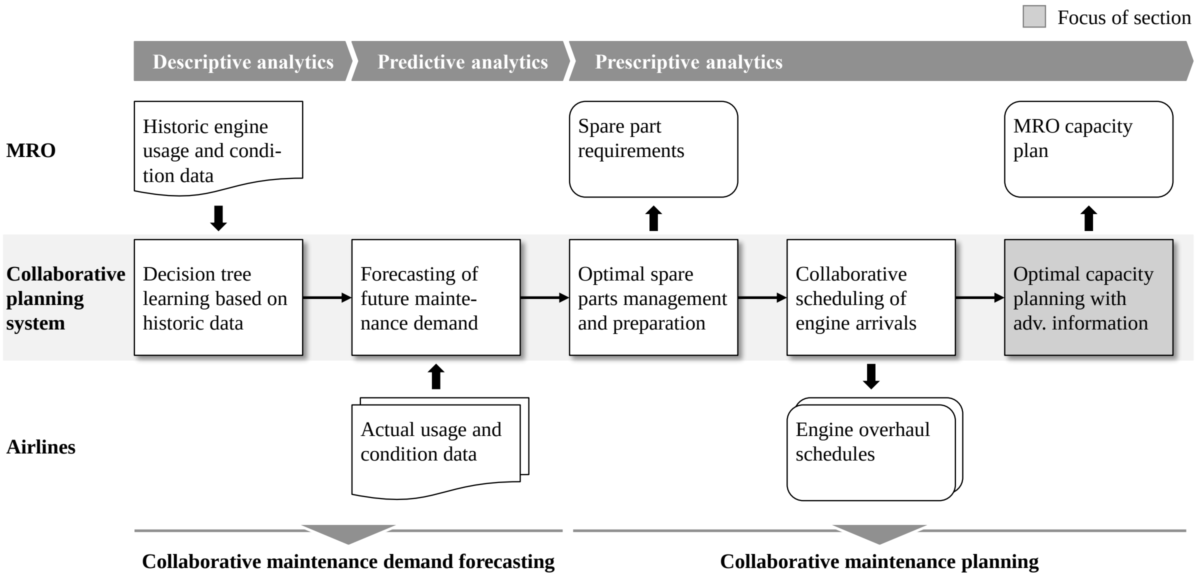
$T^*_{e,n+1} \leftarrow T_{e,n+1,m^*}$

Additional complexity could occur if there are "no-shows", i.e., if engines scheduled for overhaul don't actually arrive. Also, there could exist different priority classes, e.g., for different customers. As we so far don't know whether these cases are relevant for the scenario we restrict ourselves to provide some references.

Zacharias and Pinedo (2014) study overbooking models for scheduling arrivals for a medical facility with patients with no-show behavior. Thereby, the scheduling is carried out such that patients' waiting times and the doctor's idle and overtime are minimized. Jouini et al. (2010) schedule a call center with regular and premium impatient customers by defining policies such that a target ratio constraint on the abandonment probabilities of premium customers to regular ones is met. Finally, Mak et al. (2014) not only consider appointment scheduling, they also determine the sequence, i.e., the order of the appointments in order to minimize expected server idle time and job late-start penalty costs.

3.1.3.3 Capacity planning[15]

In this section we define a capacity planning model. We model the overhaul process as a queueing network and minimize overall costs by computing optimal capacities. With the model we are able to compute the capacity needed in the engine overhaul production network without and with advanced information and collaboration. I.e., we are able to quantify the impact of the collaborative forecasting, spare parts management and maintenance scheduling modules described previously.



[15] Main author: Julian Kurz (UWUERZ)

The optimization model is tailored to represent the MRO production network depicted in Figure 32. Detailed capacity allocation per work station is especially critical for the MRO since specific engines need specifically trained technicians for each station.

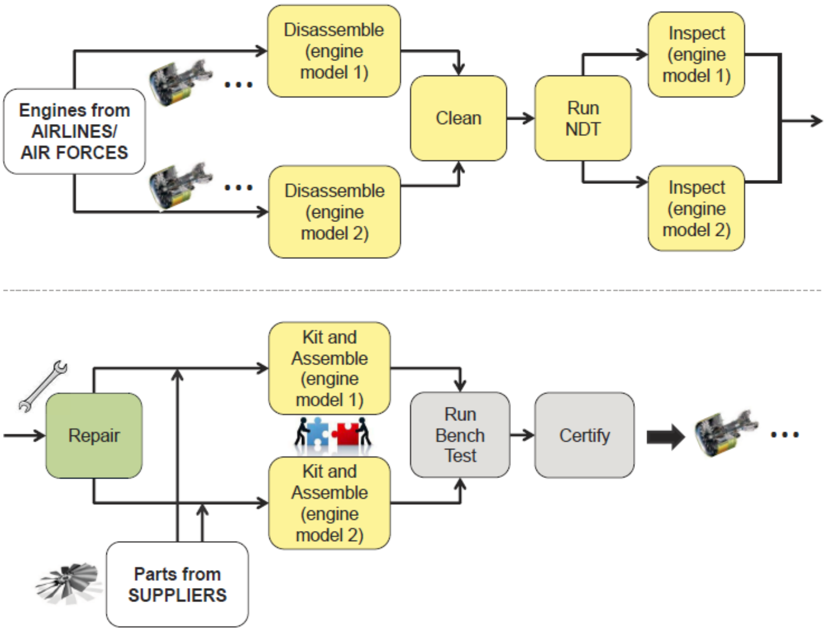


Figure 32: MRO production network.

3.1.3.3.1 Optimization model definition

During the engine overhaul process, aircraft engines pass subsequent work stations under a FCFS[16] policy. In practice, service providers or contract manufacturers and their customers' often have contracts specifying maximum lead times (also referred to as sojourn or turnaround times) for service or production processes. Over periods of time, the actual mean lead time is computed and compared to the contractually defined maximum lead time per product family. If this time is exceeded, a penalty is incurred which increases with increasing mean lead time. The objective of our capacity allocation model is to determine the capacity per work station such that total cost is minimized. Thereby, it is important to note that total cost is comprised of capacity costs associated with the service rate and penalty costs for not meeting contractually defined maximum lead times. Before developing the mathematical model to optimally determine production capacities, we summarize the notation used throughout the next subsections.

The production network consists of J work stations, numbered with the index j. E product families are processed in the network, numbered with the index e. $\mathrm{E} = \{1, \ldots, E\}$ denotes the set of product families. Each product family follows a predetermined route through the network, $\mathrm{J}_e \subseteq \mathrm{J}$, where J and J_e denote the set of work stations and the subset of work stations passed by product family e, respectively.

The decision variables of the optimization model are defined as μ_j, representing the capacity or mean service rate at each work station $j \in \{1,..., J\}$. The associated capacity cost

[16] First come, first served

terms are defined as c_j. The approximate mean lead or sojourn times at the stations are denoted as S_j, while S_e^T denotes the maximum allowed total sojourn time for product family e on path J_e after which a penalty occurs[17]. The product family-specific penalty cost is defined as γ_e.

As we will see in the next sections, we need some additional parameters in order to compute the sojourn times S_j. We define λ_j as the mean arrival rate of products at work station j (sum of the individual arrival rates of all product families served at work station j). ca_j denotes the squared coefficient of variation (ratio of the variance to the squared mean of a random variable) of product interarrival times at work station j. Service time variability at the work stations is captured as the squared coefficient of variation of service times cs_j. Finally, $L_j(\mu_j)$ denotes the approximate expected number of products at work station j depending on the service rate.

With the notation introduced above we can write the cost-optimal capacity allocation problem (CAP) as the minimization of the total cost function $\mathrm{C}: \mathbf{R}^J \mapsto \mathbf{R}$,

$$\text{minimize} \quad \mathrm{C}(\mu) = c^{\mathrm{T}}\mu + \sum_{e=1}^{E} \gamma_e \max\{0, \sum_{j \in \mathrm{J}_e} S_j(\mu_j) - S_e^T\}, \qquad \text{(CAP)}$$

where the domain of C is the open convex set

$$\mathbf{dom}\ \mathrm{C} = \{\mu \mid \mu_j > \lambda_j,\ j = 1, \ldots, J\} \subset \mathbf{R}_{++}^J.$$

In this nonlinear optimization problem, the left hand side of the objective function represents the total direct capacity costs[18] with the cost parameters $c \in \mathbf{R}_{++}^J$. The right hand side represents the penalty costs for all product families $e \in \mathrm{E}$ with the penalties $\gamma \in \mathbf{R}_{++}^E$ if the contractually defined maximum mean total sojourn time S_e^T is exceeded[19]. The $\max\{0,\cdot\}$ operator ensures that a penalty only incurs for a product family e if

$$\sum_{j \in \mathrm{J}_e} S_j(\mu_j) > S_e^T.$$

Therefore, the service provider or contract manufacturer can allocate capacity such that contractually defined lead times are not met as long as marginal capacity cost reductions exceed increased marginal penalty costs for a given product family.

The domain restriction $\mu_j > \lambda_j, \forall j \in \mathrm{J}$, is such that the network of queues is stable. This means, that the sojourn times $S_j(t)$ (or queue lengths $L_j^q(t)$) at the work stations do not grow towards infinity as time approaches infinity, $\lim_{t\to\infty} L_j^q(t)/t = 0, \forall j \in \mathrm{J}$. Service rates are assumed to be continuous variables which is quite common in the context of production planning, see Bitran and Tirupati (1989) and the references therein.

---

[17] We assume the same penalty costs and parameters S_e^T per product family for all customers.

[18] We assume cost to increase linearly with capacity.

[19] We don't consider fixed cost and material cost as they are not influenced by capacity changes in the presented setting.

Since the mean sojourn time S_j determine the payable penalties they are a key performance measure in the production network. Therefore, we subsequently describe a method to approximate it based on various queueing network parameters.

3.1.3.3.2 Sojourn time approximation for queueing networks

The individual work stations can be modeled as queues with independent and generally distributed interarrival times, generally distributed services times, infinite waiting room and one server. GI/G/1 queues allow us to take into account all possible kinds of probability distribution functions for the interarrival and service times. E.g., without scheduling, interarrival times to the production network are assumed to be exponentially distributed for each product family, resulting in a squared coefficient of variation $ca_{\exp} = \mathrm{Var}[X]/\mathrm{E}[X]^2 = \lambda^2/\lambda^2 = 1$, whereas interarrival times are assumed to be constant for perfectly scheduled arrivals, $ca_{\mathrm{const}} = 0$. A common choice for the description of service times of mainly manual operations is the log-normal distribution, with a squared coefficient of variation given as $cs = e^{\sigma^2} - 1$.

We can use parametric decomposition to approximate the production network as a network of J independent GI/G/1 queues, where each product family e follows a predetermined acyclic path J_e through the network. We assume that the union of all paths is equal to the set of queues in the network, $\bigcup_{e\in\mathsf{E}} \mathsf{J}_e = \mathsf{J}$. For details regarding the decomposition approximation, the reader is referred to Shanthikumar and Buzacott, Whitt, Bitran and Tirupati and Negri da Silva and Morabito as well as the references therein.

The production network of the aerospace case illustrated in Figure 32 for two engine types, $\mathsf{E} = \{1,2\}$, can schematically be described as shown in Figure 33.

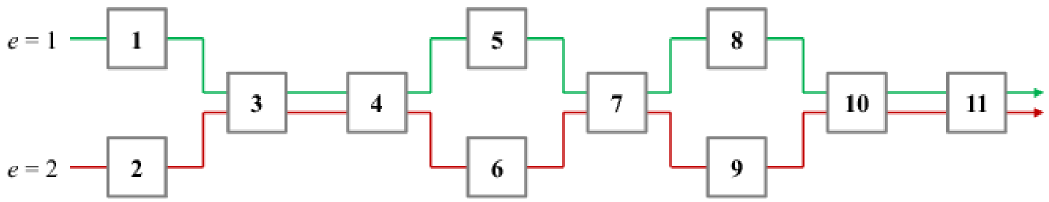


Figure 33: Schematic production network.

With the Krämer-Langenbach-Belz (KLB) approximation for GI/G/1 queues, the steady-state mean number of products L at a given node with a service rate μ can be approximated as

$$\mathsf{E}[N(\infty\,|\,\mu)] = L(\mu) = \frac{\lambda}{\mu} + \frac{(ca+cs)\lambda^2}{2}\,\frac{1}{\mu(\mu-\lambda)}\,g(\mu) \qquad (2)$$

where

$$g(\mu) = \begin{cases} \exp\left\{\dfrac{-2(1-ca)(\mu-\lambda)}{3\lambda(ca+cs)}\right\} & \text{if } ca \le 1 \\ 1 & \text{otherwise.} \end{cases}$$

As mentioned in the previous section, the average service rate needs to exceed the average arrival rate for stability. Therefore, the domain of the approximation function for the mean number of products in the system is given as $\mathbf{dom}\ L = \{\mu \,|\, \mu > \lambda\}$.

With Little's law, $L = \lambda S$, which also applies to GI/G/1 queues, and assuming for all nodes that $ca, cs \in [0,1]$, the KLB approximation for the mean number of products in the system can be translated in the mean sojourn time (sum of expected waiting and service times per customer) at a node,

$$S(\mu) = \frac{L(\mu)}{\lambda} = \frac{1}{\mu} + \frac{(ca+cs)\lambda}{2}\frac{1}{\mu(\mu-\lambda)}\exp\left\{\frac{-2(1-ca)(\mu-\lambda)}{3\lambda(ca+cs)}\right\}, \tag{3}$$

with $\mathbf{dom}\ S = \mathbf{dom}\ L$.

Equation (3) provides a method to approximate the mean sojourn times $S_j(\mu_j)$ at all work stations in the production network. However, not all parameters needed are available off the shelf. We can assume to know the arrival rate per product family to the system and the service time variabilities cs_j for all work stations. Although we assume independence of the queues at the individual nodes, the arrival rates λ_j and their variability parameters ca_j for all work stations need to be determined depending on the network structure. These parameters are determined prior to the optimization based on the approximation algorithms described in Section 3.1.3.3.3. Additionally, since we compute S_j as a univariate function of μ_j, we need the following assumption.

Assumption 1 *The effect of a change in service rate μ_j on the interdeparture time variability cd_j and therefore on the interarrival time variability $ca_i, i > j$, can be neglected.*

We follow the approach proposed by several authors, e.g., by Bitran and Tirupati 1989, and compute λ_j and ca_j for all work stations $j \in \mathrm{J}$ once for a reasonable guess of $\mu \in \mathbf{dom}\ \mathrm{C}$ prior to the optimization routine. During optimization, the parameters are assumed to remain constant.

Finally, with the definition of the approximate mean sojourn times we can derive further insights regarding the capacity allocation problem defined in Section 3.1.3.3.1.

Proposition 1 *With Assumption 1 and $ca, cs \geq 0$, the capacity allocation problem (CAP) is convex in $\mathbf{dom}\ \mathrm{C}$ and a unique optimal solution $\mu^* \in \mathbf{dom}\ \mathrm{C}$ exists.*

Proof. In the first part of the proof, we show that the mean sojourn time approximations functions $S(\mu)$ as defined in (3) are convex and strictly monotonically decreasing in $\mathbf{dom}\ S = \{\mu \mid \mu > \lambda\} \subset \mathbf{R}_{++}$. We start by showing convexity of $g(\mu)$ as defined in (2). For $ca > 1, g(\mu) = 1$, convexity follows instantly. For $ca \in [0,1]$, since $g(\mu) = \exp\{h(\mu)\}$ is convex if $h(\mu)$ is convex, we need to show that

$$h(\mu) = \frac{-2(1-ca)(\mu-\lambda)}{3\lambda(ca+cs)}$$

is convex. With some reformulation, $h(\mu)$ can be rewritten as an affine function $h(\mu) = a\mu + b$ with

$$a = \frac{2(ca-1)}{3\lambda(ca+cs)} \quad \text{and} \quad b = \frac{2(1-ca)}{3(ca+cs)}.$$

Since affine functions are always convex, $h(\mu)$ and therefore $g(\mu)$ are convex for $ca \in [0,1]$. Furthermore, since $h(\mu), h'(\mu) \leq 0$ (h' denotes the first-order derivative of h) for

$ca \in [0,1]$ and $g(\mu)=1$ for $ca > 1$, $g(\mu)$ is non-increasing (or monotonically decreasing) in $\textbf{dom } S$.

Next, since the product of two convex, non-increasing functions on an interval in $\mathbf{R}$ is convex, we show that $f(\mu) = 1/(\mu(\mu-\lambda))$ is convex and non-increasing in $\textbf{dom } S$. Convexity can easily be shown by checking the second-order condition, $\nabla^2 f(\mu) \pm 0, \forall \mu \in \textbf{dom } S$. Since $f(\mu)$ is a composition $f = h \circ g$ of two scalar functions $h(g) = 1/g$ and $g(\mu) = \mu(\mu - \lambda)$ in $\textbf{dom } S \subset \mathbf{R}_{++}$, the second-order condition can be expressed with the first- and second-order derivatives f', g', h' and f'', g'', h'', respectively, as $f''(\mu) = h''(g(\mu))g'(\mu)^2 + h'(g(\mu))g''(\mu) \geq 0$,

$$f''(\mu) = \underbrace{\frac{2(2\mu-\lambda)^2}{(\mu^2-\mu\lambda)^3}}_{>0} - \underbrace{\frac{2}{(\mu^2-\mu\lambda)^2}}_{>0} \overset{?}{\geq} 0.$$

In order for $f''(\mu)$ to be positive, the first term must be larger than the second term for all $\mu \in \textbf{dom } S$. By multiplying with $(\mu^2 - \mu\lambda)^3$ on both sides and some rearranging we end up with the inequality $3\mu^2 - \lambda\mu + \lambda \geq 0$ which strictly holds since $\mu > \lambda$ and $\lambda > 0$. Therefore, $f(\mu)$ is strictly convex. As the first derivative of $f(\mu)$ is negative,

$$f'(\mu) = -\frac{2\mu - \lambda}{(\mu^2 - \mu\lambda)^2} < 0 \qquad \forall \mu \in \textbf{dom } S,$$

the function is strictly monotonically decreasing. Therefore, we can conclude that $1/(\mu(\mu-\lambda))g(\lambda, \mu, ca, cs)$ is convex and strictly monotonically decreasing.

Finally, since the multiplication with a non-negative multiplier $(ca + cs)\lambda/2 > 0$ reveals convexity, the sum of two convex functions is convex and $\partial^2/\partial\mu^2(1/\mu) > 0, \forall \mu \in \textbf{dom } S$, we can conclude that the approximation function for the mean sojourn time S is convex in $\textbf{dom } S$. Since $\partial/\partial\mu(1/\mu) < 0, \forall \mu \in \textbf{dom } S$, it also follows that S is strictly monotonically decreasing. This concludes the first part of the proof.

The sum of separable approximations of the mean sojourn times, $\sum_{j \in \mathrm{J}_e} S_j(\mu_j)$, is convex in $\textbf{dom } \mathrm{C} = \bigcup_{j \in \mathrm{J}} \textbf{dom } S_j$ since all functions $\left(S_j(\cdot)\right)_{j \in \mathrm{J}}$ are convex functions. For all $e \in \mathrm{E}$, subtraction of a constant S_e^T, application of the $\max\{0, \cdot\}$ function and multiplication with a non-negative multiplier $\gamma_e > 0$ retains convexity. Therefore, with the linear convex left hand side of the objective function, $c^{\mathrm{T}}\mu = \sum_{j \in \mathrm{J}} c_j \mu_j$, as well as the convex right hand side, the objective function is convex in $\textbf{dom } \mathrm{C}$.

Finally, since the left hand side of the objective function is strictly monotonically increasing and the right hand side is monotonically decreasing (not strictly due to $\max\{0, \cdot\}$) in $\textbf{dom } \mathrm{C}$, at least one optimal vector $\mu^* \in \textbf{dom } \mathrm{C}$ exists which concludes the proof. □

3.1.3.3.3 Queueing network parameter computation

The algorithm described here is a variation of the Queueing Network Analyzer (QNA) algorithm described by by Bitran et al. 1996, modified to fit the requirements of the DTA scenario.

Whereas some input parameters such as $\lambda_e, ca_e, \mu_j = \tau_j^{-1}, cs_j$ and q_{ij} are given exogenously, the arrival rates λ_j and the according SCV need to be computed prior to the network analysis. Since we assume deterministic routing, the arrival rates λ_j can simply be calculated as the sum of the external expected arrival rates of all engine or customer classes arriving at node j,

$$\lambda_j = \sum_{e=1}^{E} \sum_{l=1}^{\max_i n_{ei}} \lambda_e 1\{(e,l) : n_{el} = j\}$$

where $n_{e,l}$ denotes the lth workstation visited by engines of type e.

In order to determine the variability parameters of the arrivals, we need to consider three different processes: superposition of arrivals, departures from nodes and (deterministic) splitting of departures.

Superposition of arrivals

With $\lambda_{ij} = \lambda_i q_{ij}$, $\rho_j = \lambda_j \tau_j$ and with the interarrival time variability at node j from node i, ca_{ij}, the variability parameters of a superposition of arrivals can be approximated as

$$ca_j = w_j \sum_{i=1}^{J} \frac{\lambda_{ij}}{\lambda_j} ca_{ij} + 1 - w_j,$$

where

$$w_j = \frac{1}{1 + 4(1-\rho_j)^2 (v_j - 1)}$$

$$v_j = \frac{1}{\sum_{i=1}^{J} (\frac{\lambda_{ij}}{\lambda_j})^2}.$$

Departures

The variability of the departure process from a node depends on the variability of the arrivals and the service times. It can be approximated as

$$cd_j = \rho_j^2 cs_j + (1-\rho_j^2) ca_j.$$

Deterministic splitting of departures

For the deterministic splitting process, we do not use the approximation developed by Whitt in 1983 for Markovian routing,

$$cd_i = p_i cd + 1 - p_i,$$

but the convex combination developed by Whitt in 1994 which is an approximation of the complex Erlang numerical procedure proposed by Bitran (1988) for $ca_i \leq 1$,

$$cd_1 = p_1 cd + p_1(1-p_1) ca_2 + (1-p_1)^2 ca_1.$$

Note that the subscript 1 denotes the currently observed customer class $e_1 \in \mathrm{E}$ while subscript 2 represents the superposition of all other classes present at the node, $\mathrm{E}_2 = \mathrm{E} \setminus e_1$.

With these three approximations, all necessary parameters of the modified KLB equations (3) are known which can then be applied to compute the expected sojourn times in the queueing network.

3.1.3.3.4 Solution procedure

Due to the $\max\{0,\cdot\}$ operator in (CAP), C is generally not differentiable in $\mathbf{dom}\,\mathrm{C}$. Therefore, a point $\mu^* \in \mathbf{dom}\,\mathrm{C}$ is minimizer of C if

$$0 \in \partial \mathrm{C}(\mu^*),$$

where $\partial\mathrm{C}(\mu)$ denotes the subdifferential (or the set of subgradients) of C at point $\mu \in \mathbf{dom}\,\mathrm{C}$. In order to construct $\partial\mathrm{C}(\mu)$, we first note that the subdifferential of the pointwise maximum of differentiable convex functions $f_1,\ldots,f_m$ is given as the convex hull of the union of the gradients of the active functions,

$$\partial f(x) = \partial \max_{i=1,\ldots,m} f_i(x) = \mathbf{Co} \cup \{\nabla f_i(x) \mid f_i(x) = f(x)\}.$$

For the $\max\{0,\cdot\}$ operator as defined in the capacity allocation problem,

$$f_e(\mu) = \max\{0, \sum_{j\in \mathrm{J}_e} S_j(\mu_j) - S_e^T\},$$

the subdifferential is therefore given by

$$\partial f_e(\mu) = \begin{cases} 0 & \text{if } \sum_{j\in \mathrm{J}_e} S_j(\mu_j) < S_e^T \\ \nabla \sum_{j\in \mathrm{J}_e} S_j(\mu_j) & \text{if } \sum_{j\in \mathrm{J}_e} S_j(\mu_j) > S_e^T \\ [0, \nabla \sum_{j\in \mathrm{J}_e} S_j(\mu_j)] & \text{if } \sum_{j\in \mathrm{J}_e} S_j(\mu_j) = S_e^T. \end{cases} \tag{4}$$

Using this we can write the subdifferential of C for $\mu \in \mathbf{dom}\,\mathrm{C}$ as

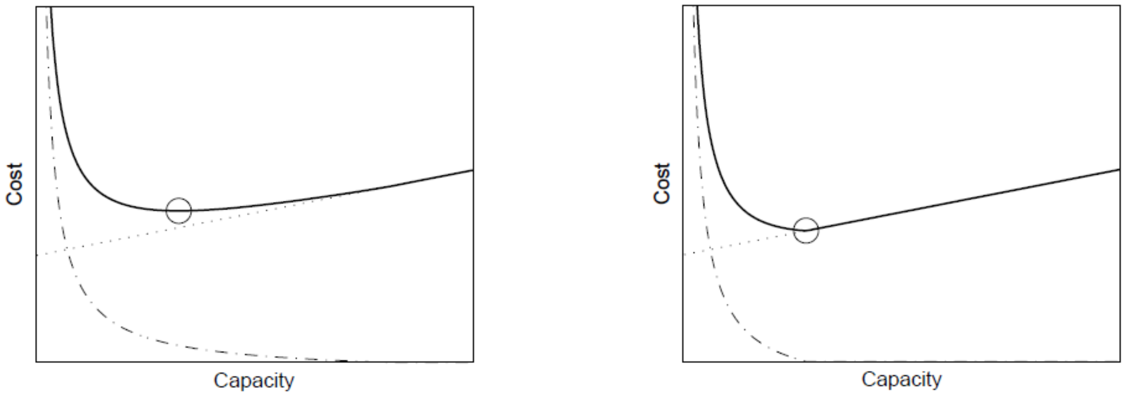
$$\partial \mathrm{C}(\mu) = c^{\mathrm{T}} + \sum_{e\in \mathrm{E}_>} \gamma_e \nabla \sum_{j\in \mathrm{J}_e} S_j(\mu_j) + \sum_{e\in \mathrm{E}_=} \gamma_e [0, \nabla \sum_{j\in \mathrm{J}_e} S_j(\mu_j)], \tag{5}$$

with $\mathrm{E}_>, \mathrm{E}_=, \mathrm{E}_< \subseteq \mathrm{E}$ denoting the disjoint subsets of product families where

$$\sum_{j\in \mathrm{J}_e} S_j(\mu_j) > S_e^T, \qquad \sum_{j\in \mathrm{J}_e} S_j(\mu_j) = S_e^T \qquad \text{and} \qquad \sum_{j\in \mathrm{J}_e} S_j(\mu_j) < S_e^T,$$

respectively. It is easy to see that $\mathrm{E}_> \cup \mathrm{E}_= \neq \varnothing$ for the minimizer μ^*, since (5) reduces to $\partial\mathrm{C}(\mu) = c^{\mathrm{T}} \neq 0$ otherwise. Furthermore, since $\mathrm{C}(\mu) \in \mathbf{R}^J$, there must be enough product families with mean total sojourn time greater or equal to the contractually defined maximum mean total sojourn time such that all queues are encompassed in their paths, $\bigcup_{e\in \mathrm{E}_> \cup \mathrm{E}_=} \mathrm{J}_e = \mathrm{J}$, respectively.

In Figure 34, the total cost function is illustrated as a one-dimensional curve for different values of S_e^T. The optimal point is indicated by the circle. Figure 34 (a) represents the cost curves for subsets $\mathrm{E}_>$, Figure 34 (b) the cost curves for subsets $\mathrm{E}_=$.



(a) Low S_e^T: Mean lead time is higher than S_e^T for the optimal capacity, a penalty is incurred. The curve is differentiable at the minimum.

(b) High S_e^T: Mean lead time equals S_e^T for the optimal capacity, no penalty is incurred. The curve has a kink at the minimum.

Figure 34: One-dimensional illustration of total cost function (—), composed of capacity costs (···) and penalty costs (· – ·).

As we are not able to solve $0 \in \partial C(\mu^*)$ with (5) directly, we distinguish two limiting and directly solvable cases and a general case for any other solution depending on the differentiability of (CAP). The two limiting cases are defined such that either all relevant penalty terms are differentiable (corresponding to Figure 34 (a)) or not differentiable (corresponding to Figure 34 (b)). The general case applies if the cost term is differentiable for some product families and for some not.

Due to the complex structure of the problem, it is a priori not possible to determine which case of the subsequently listed applies for a given set of parameters. Therefore, we propose a solution approach where we solve the optimization problem for all three cases using different methodologies and determine the "true" optimal solution from their respective outcomes. If μ^{*1}, μ^{*2} and $\mu^{(i_{\text{best}}^{(k)})}$ denote the case-optimal capacities for all three cases, the optimal value is determined as

$$\mu^* = *argmin_{\mu}\{C(\mu^{*1}), C(\mu^{*2}), C(\mu^{(i_{\text{best}}^{(k)})})\}. \tag{6}$$

Mathematically, the three cases can be defined as follows.

- *Case 1*: For given parameters S_e^T, the optimal solution $\mu^* \in \mathbf{dom}\ C$ is such that $E_> \cup E_< = E$ and $E_= = \varnothing$, i.e., C is differentiable in $\mathbf{dom}\ C$.
- *Case 2*: For given parameters S_e^T, the optimal solution $\mu^* \in \mathbf{dom}\ C$ is such that $E_> \cup E_< = \varnothing$ and $E_= = E$, i.e., there exists no path $J_e, e \in E$, where the penalty cost term is differentiable in $\mathbf{dom}\ C$.
- *Case 3*: For given parameters S_e^T, the optimal solution $\mu^* \in \mathbf{dom}\ C$ is such that $E_> \neq \varnothing$ and $E_= \neq \varnothing$, which means that there exist paths $J_e, e \in E_> \cup E_<$ where the penalty cost terms are differentiable and paths $J_e, e \in E_=$ where the penalty cost terms are not differentiable in $\mathbf{dom}\ C$.

For *Case 1* and *Case 2*, the solution of the optimization problem (CAP) can be found explicitly from the problem formulation as shown in Proposition 2 and Proposition 3, respectively. For *Case 3*, we use an iterative optimization algorithm that can also be employed to verify the results obtained through the explicit solutions of *Case 1* and *Case 2*.

Case 1

In the following proposition we show that we can explicitly solve the first case directly from the problem statement.

Proposition 2 *With Assumption 1,* $ca, cs \geq 0$ *and if the contractually defined maximum mean total sojourn times* S_e^T *are such that* $\mathsf{C}(\mu^{*1})$ *is differentiable, the Case 1-optimal solution* $\mu^{*1} \in \mathbf{dom}\ \mathsf{C}$ *can be found by solving at most* $J(E^2-1)$ *explicit equations.*

Proof. From the definition of *Case 1* we know that $\mathsf{E}_= = \varnothing$, i.e,

$$\neg\exists e \in \mathsf{E}, \sum_{j \in \mathsf{J}_e} S_j(\mu_j^{*1}) = S_e^T.$$

Taking this and relation (4) into account, the capacity allocation (CAP) can be expressed as

$$\text{minimize} \quad \mathsf{C}^1(\mu) = c^{\mathrm{T}}\mu + \sum_{e \in \mathsf{E}_>} \gamma_e [\sum_{j \in \mathsf{J}_e} S_j(\mu_j) - S_e^T]$$

and the subdifferential (5) of $\mathsf{C}^1(\mu)$ simplifies to the ordinary gradient,

$$\partial \mathsf{C}^1(\mu) = \nabla \mathsf{C}^1(\mu) = c^{\mathrm{T}} + \sum_{e \in \mathsf{E}_>} \gamma_e \nabla \sum_{j \in \mathsf{J}_e} S_j(\mu_j).$$

In order to find potential solutions of the optimization problem, i.e., $\nabla \mathsf{C}^1(\mu^{*1}) = 0$, we need to find vectors $\mu^{*1} \in \mathbf{dom}\ \mathsf{C}$ that solve

$$c^{\mathrm{T}} = -\sum_{e \in \mathsf{E}_>} \gamma_e \nabla \sum_{j \in \mathsf{J}_e} S_j(\mu_j^{*1}) \tag{7}$$

for all possible combinations of $\mathsf{E}_>$. Since $\mathsf{E}_> \neq \varnothing$ and we demand $\bigcup_{e \in \mathsf{E}_>} \mathsf{J}_e = \mathsf{J}$, this are at most $E^2 - 1$ possible combinations, each requiring the solution of J equations. Therefore, we need to solve at most $J(E^2-1)$ equations in total.

Since the solution of Equation (7) is independent of S_e^T, we need to check the conditions

$$\forall e \in \mathsf{E}_>, \sum_{j \in \mathsf{J}_e} S_j(\mu_j^{*1}) > S_e^T$$

and

$$\forall e \in \mathsf{E}_<, \sum_{j \in \mathsf{J}_e} S_j(\mu_j^{*1}) < S_e^T$$

in order to verify that a solution obtained through the approach above is a feasible minimizer of the problem. This concludes the proof. □

For each node in the queueing network, the equation to be solved is explicitly given by

$$c_j = -\sum_{e \in \mathsf{E}_>^j} \gamma_e \frac{\partial S_j(\mu_j^{*1})}{\partial \mu_j} \qquad \forall j \in \mathsf{J} \tag{8}$$

where $\mathrm{E}_{>}^{j} \subseteq \mathrm{E}_{>}$ is the set of product families where node j is contained in all paths, $j \in \mathrm{J}_e, \forall e \in \mathrm{E}_{>}^{j}$ (see equation (23) for the explicit derivative $\partial_{\mu_j} S_j(\mu_j)$). As an example, for the network depicted in Figure 32, we only need to solve J equations as there is only one possible choice of $\mathrm{E}_{>}$, namely $\mathrm{E}_{>} = \mathrm{E}$, due to the condition $\bigcup_{e \in \mathrm{E}_{>}} \mathrm{J}_e = \mathrm{J}$.

Case 2

In the following proposition we show that the second case can also be solved directly by reformulation of the problem and exploitation of duality properties.

Proposition 3 *With Assumption 1,* $ca, cs \geq 0$ *and if the contractually defined maximum mean total sojourn times* S_e^T *are such that there exists no path* J_e, $e \in \mathrm{E}$*, where the penalty cost term is differentiable in* $\mathbf{dom}\ \mathrm{C}$*, the Case 2-optimal solution* $\mu^{*2} \in \mathbf{dom}\ \mathrm{C}$ *can be found by solving a system of* $J + E$ *equations.*

Proof. The proof is based on the Karush-Kuhn-Tucker (KKT) conditions of a reformulation of the optimization problem. If the minimizer of C is such that $\mathrm{E}_{=} = \mathrm{E}$ (i.e., $\mathrm{E}_{>} \cup \mathrm{E}_{<} = \varnothing$), problem (CAP) can be reformulated as the optimization problem

$$\text{minimize}\, c^{\mathrm{T}} \mu \tag{9}$$

$$\text{subject to} \sum_{j \in \mathrm{J}_e} S_j(\mu_j) - S_e^T = 0 \forall e \in \mathrm{E} \tag{10}$$

$$\lambda_j - \mu_j \leq 0 \forall j \in \mathrm{J} \tag{11}$$

If we refer to the equality constraints (10) as $h_e(\mu), e \in \mathrm{E}$, the inequality constraints (11) as $f_j(\mu), j \in \mathrm{J}$, and the objective function as $f_0(\mu) = c^{\mathrm{T}} \mu$ and if we introduce the KKT multipliers η_j and ν_j, the KKT conditions of the optimization problem are defined as

$$f_j(\mu^{*2}) \leq 0, \ \forall j \in J \tag{12}$$

$$h_e(\mu^{*2}) = 0, \forall e \in \mathrm{E} \tag{13}$$

$$\eta_j^* \geq 0, \forall j \in J \tag{14}$$

$$\eta_j^* f_j(\mu^{*2}) = 0, \forall j \in J \tag{15}$$

$$\nabla f_0(\mu^{*2}) + \sum_{j=1}^{J} \eta_j^* \nabla f_j(\mu^{*2}) + \sum_{e=1}^{E} \nu_e^* \nabla h_e(\mu^{*2}) = 0. \tag{16}$$

We instantly see that $\eta_j^* = 0, \forall j \in J$ from the complementary slackness condition (15) and constraint (11). Therefore, condition (16) for the optimal solution of the optimization problem can be expressed as

$$c^{\mathrm{T}} = -\sum_{e=1}^{E} \nu_i^* \nabla \sum_{j \in \mathrm{J}_e} S_j(\mu_j^{*2}). \tag{17}$$

With this equation and condition (13) we get a system of $J+E$ equations which can be solved to obtain the primal and dual optimal points $\mu^{*2} \in \mathbf{dom}\ \mathrm{C}$ and (η^*, ν^*), respectively (where $\eta^* = 0$).

From constraint (13) we know that $\mathrm{E}_> = \varnothing$. Another condition to verify that the obtained solution is optimal is

$$c_j \leq -\sum_{e \in \mathrm{E}^j} \gamma_e \frac{\partial S_j(\mu_j^{*2})}{\partial \mu_j} \qquad \forall j \in \mathrm{J}, \tag{18}$$

where $\mathrm{E}^j \subseteq \mathrm{E}$ is the set of product families where node j is contained in all paths, $j \in \mathrm{J}_e, \forall e \in \mathrm{E}^j$. This condition is necessary since μ^{*2} needs to be such that there is a kink at $\mathrm{C}(\mu^{*2})$ where $0 \in \partial \mathrm{C}(\mu^{*2})$ (also, γ_e is not considered in the optimization problem). This concludes the proof. □

Explicitly, the system of equations to be solved is given by

$$0 = c_j + \sum_{e \in \mathrm{E}^j} \nu_i^* \frac{\partial S_j(\mu_j^{*2})}{\partial \mu_j} \ \forall j \in \mathrm{J}$$

$$0 = \sum_{j \in \mathrm{J}_e} S_j(\mu_j^{*2}) - S_e^T \ \forall e \in \mathrm{E}.$$

As the negative slope of S_j increases with decreasing μ_j, it is obvious that *Case 2* becomes more likely to be the right way to solve (CAP) (as compared to *Case 1*) for increasing S_e^T. In other words, if the contractually defined maximum mean total sojourn times are relatively high we define capacity such that actual and contractually defined sojourn times coincide (incremental penalty cost increases for capacity reduction are higher than incremental capacity cost savings).

Case 3

If $\mathrm{E}_> \neq \varnothing$ and $\mathrm{E}_= \neq \varnothing$ at $\mu^* \in \mathbf{dom}\ \mathrm{C}$ for given parameters S_e^T, i.e., there exist paths where the penalty cost term is differentiable and paths where it is not differentiable, we need to solve the problem with an iterative method for nonlinear and nondifferentiable optimization since we would need to solve Equation (5) which is not possible explicitly. Applicable iterative methods are, e.g., subgradient, cutting plane or trust region methods.

In order to use the subgradient method for unconstrained problems to determine the solution of (CAP), we define the extended-value extension $\tilde{C}$ with $\mathbf{dom}\ \tilde{C} = \mathbf{R}^J$ of the total cost function as $\tilde{C}(\mu) = \infty, \mu \notin \mathbf{dom}\ \mathrm{C}$ and $\tilde{C}(\mu) = \mathrm{C}(\mu), \mu \in \mathbf{dom}\ \mathrm{C}$. Therefore, we can minimize $\tilde{C}$ with the subgradient method which uses the iteration

$$\mu^{(k+1)} = \mu^{(k)} - \alpha_k g^{(k)}. \tag{19}$$

Here, the kth iterate $\mu^{(k)}$ and any subgradient $g^{(k)} \in \partial \tilde{C}(\mu^{(k)})$ are used to determine the $(k+1)$th iteration, where $\alpha_k > 0$ denotes the kth step size. Although Equation (19) looks like the ordinary gradient method for differentiable functions, it is different since it is not a descent method, meaning that $\tilde{C}(\mu^{(k+1)}) > \tilde{C}(\mu^{(k)})$ can happen. Therefore, after each iteration we set

$$\tilde{C}^{(k)}_{\text{best}} = \min\{\tilde{C}^{(k-1)}_{\text{best}}, \tilde{C}(\mu^{(k)})\}, \quad (20)$$

and $i^{(k)}_{\text{best}} = k$ if $\tilde{C}(\mu^{(k)}) = \tilde{C}^{(k-1)}_{\text{best}}$, i.e., if $\mu^{(k)}$ is the best point found so far.

The gradient $g^{(k)}$ can for example be computed as

$$g^{(k)} = c^{\mathrm{T}} + \sum_{e \in \mathrm{E}_>} \gamma_e \nabla \sum_{j \in \mathrm{J}_e} S_j(\mu^{(k)}_j) + \sum_{e \in \mathrm{E}_=} r \gamma_e \nabla \sum_{j \in \mathrm{J}_e} S_j(\mu^{(k)}_j) \in \partial \tilde{C}(\mu^{(k)}),$$

where $r \in [0,1]$ is a random number to reflect the convex hull property in (5).

We have now introduced three ways to compute the solution of the optimization problem (CAP) depending on the input parameters. To determine the "true" optimal solution of the problem, we find the case-optimal solution minimizing C as defined in Equation (6).

3.1.3.3.5 Effects of advanced information and collaboration

Nowadays, as all possible parameters of technical equipment, e.g., vibrations levels and oil pressure/temperature of aircraft engines, can be monitored online, predictive maintenance mechanisms exist in order to not only determine the ideal maintenance timing but also forecast the types of services that need to be carried out. E.g., depending on the parameters and lifetimes of individual parts or modules, one can predict if it should be repaired or replaced. Therefore, through sharing this information with the maintenance service provider, supply chain activities such as spare parts management or preparation of the service can be optimized. In return, this reduces the time needed for the service of an engine at the stations where the additional information allows better preparation. Additionally, as explained in more details in the subsequent section, it reduces the variability of service times since additional information leads to improved supply chain coordination.

The update service rates are given by $A\mu$ where $A \in \mathbf{R}^{J \times J}_+$ is a matrix where the elements on the main diagonal are given by $\xi_j \geq 1, \forall j \in \mathrm{J}$ and all other elements are zero. The factor ξ_j represents the relative mean service rate improvement at node j through collaborative forecasting of maintenance demand and preparation ($\xi_j = 1$ means no improvement, $\xi_j > 1$ indicates an improvement at the respective node). The updated optimization problem is given by

$$\text{minimize} \quad D(\mu) = c^{\mathrm{T}} \mu + \sum_{e \in \mathrm{E}} \gamma_e \max\{0, \sum_{j \in \mathrm{J}_e} S_j(\xi_j \mu_j) - S^T_e\} \quad (21)$$

with

$$\mathbf{dom}\ D = \{\mu \mid \xi_j \mu_j > \lambda_j,\ j = 1, \ldots, J\} \subset \mathbf{R}^J_{++}.$$

If we define the optimal solutions of the original and the updated optimization problems as $\mu^* \in \mathbf{dom}\ \mathrm{C}$ and $\tilde{\mu} \in \mathbf{dom}\ D$, respectively, we can find the following proposition.

Proposition 4 *If there is a mean service rate improvement at any node, i.e.,* $\exists j \in \mathrm{J}, \xi_j > 1$, *we can derive the following insights:*

1. *Penalties will be reduced or remain constant,* $\sum_{j \in \mathrm{J}_e} S_j(\xi_j \tilde{\mu}_j) \leq \sum_{j \in \mathrm{J}_e} S_j(\mu^*_j), \forall e \in \mathrm{E}$.
2. *Capacity costs will always be reduced,* $c^{\mathrm{T}} \tilde{\mu} < c^{\mathrm{T}} \mu^*$.

3. *Total costs will always be reduced,* $D(\tilde{\mu}) < \mathrm{C}(\mu^*)$.

Proof. The proof is carried out sequentially for the four statements.

1. On all paths without penalty we instantly know that

$$\sum_{j\in \mathrm{J}_e} S_j(\xi_j \tilde{\mu}_j) = \sum_{j\in \mathrm{J}_e} S_j(\mu_j^*) = S_e^T \quad \forall e \in \mathrm{E}_=.$$

Therefore, we need to show that on all paths where a penalty occurs

$$\sum_{j\in \mathrm{J}_e} S_j(\xi_j \tilde{\mu}_j) < \sum_{j\in \mathrm{J}_e} S_j(\mu_j^*) \quad \forall e \in \mathrm{E}_>$$

holds. Since $S_j(\cdot)$ is strictly monotonically decreasing we can prove the statement by showing that $\xi_j \tilde{\mu}_j > \mu_j^*, \forall j \in \mathrm{J}$ where $\xi_j > 1$.

Following the proof of Proposition 2 for the updated cost function $D(\mu)$ as defined in (21) we get

$$c^{\mathrm{T}} = -\sum_{e\in \mathrm{E}_>} \gamma_e \nabla \sum_{j\in \mathrm{J}_e} S_j(\xi_j \tilde{\mu}_j) \qquad (22)$$

for the solution of the differentiable optimization problem. On the first sight, when comparing this result to Equation (7), one could think that the updated *Case 1*-optimal solution is given by $\tilde{\mu} = A^{-1}\mu^{*1}$. Although we can simply replace μ_j by $\xi_j \mu_j$ to compute $S_j(\xi_j \mu_j)$, we have to explicitly evaluate the first-order derivative of $S_j(\xi_j \mu_j)$ in order to compute the gradient of the updated cost function. The original first-order derivative of $S_j(\mu_j)$ is given by

$$\frac{\partial S_j(\mu_j)}{\partial \mu_j} = -\left\{ \frac{1}{\mu_j^2} + \frac{(ca_j + cs_j)\lambda_j}{2\mu_j(\mu_j - \lambda_j)} \exp\left\{ -\frac{2(1-ca_j)(\mu_j - \lambda_j)}{3\lambda_j(ca_j + cs_j)} \right\} \cdot \left[\frac{2\mu_j - \lambda_j}{\mu_j(\mu_j - \lambda_j)} + \frac{2(1-ca_j)}{3\lambda_j(ca_j + cs_j)} \right] \right\}. \qquad (23)$$

The updated first-order derivative of $S_j(\xi_j \mu_j) = \tilde{S}_j(\mu_j)$ is given by

$$\frac{\partial \tilde{S}_j(\mu_j)}{\partial \mu_j} = -\left\{ \frac{1}{\xi_j \mu_j^2} + \frac{(ca_j + cs_j)\lambda_j}{2\xi_j \mu_j(\xi_j \mu_j - \lambda_j)} \exp\left\{ -\frac{2(1-ca_j)(\xi_j \mu_j - \lambda_j)}{3\lambda_j(ca_j + cs_j)} \right\} \cdot \left[\frac{2\xi_j^2 \mu_j - \xi_j \lambda_j}{\xi_j \mu_j(\xi_j \mu_j - \lambda_j)} + \frac{2\xi_j(1-ca_j)}{3\lambda_j(ca_j + cs_j)} \right] \right\}. \qquad (24)$$

Since $\partial_{\mu_j} S_j(\mu_j)$ and $\partial_{\mu_j} \tilde{S}_j(\mu_j)$ are differentiable and strictly monotonically increasing in **dom** D we can show that $\xi_j \tilde{\mu}_j > \mu_j^*, \forall j \in \mathrm{J}_e, e \in \mathrm{E}_>$ where $\xi_j > 1$ by inserting $\mu_j = \mu_j^* / \xi_j$ in equation (24) and comparing it to equation (23) with $\mu_j = \mu_j^*$. I.e., if

$$-\frac{\partial \tilde{S}_j\left(\frac{\mu_j^*}{\xi_j}\right)}{\partial \mu_j} > -\frac{\partial S_j(\mu_j^*)}{\partial \mu_j} \tag{25}$$

holds, we can conclude that $\tilde{\mu}_j > \mu_j^*/\xi_j$ or $\xi_j \tilde{\mu}_j > \mu_j^*$. The substitutions yield

$$-\frac{\partial S_j(\mu_j^*)}{\partial \mu_j} = \underbrace{\frac{1}{(\mu_j^*)^2}}_{(1a)} + \underbrace{\frac{(ca_j + cs_j)\lambda_j}{2\mu_j^*(\mu_j^* - \lambda_j)}}_{(1b)} \underbrace{\exp\left\{-\frac{2(1-ca_j)(\mu_j^* - \lambda_j)}{3\lambda_j(ca_j + cs_j)}\right\}}_{(1c)}$$

$$\cdot[\underbrace{\frac{2\mu_j^* - \lambda_j}{\mu_j^*(\mu_j^* - \lambda_j)}}_{(1d)} + \underbrace{\frac{2(1-ca_j)}{3\lambda_j(ca_j + cs_j)}}_{(1e)}]$$

and

$$-\frac{\partial \tilde{S}_j\left(\frac{\mu_j^*}{\xi_j}\right)}{\partial \mu_j} = \underbrace{\frac{\xi_j}{(\mu_j^*)^2}}_{(2a)} + \underbrace{\frac{(ca_j + cs_j)\lambda_j}{2\mu_j^*(\mu_j^* - \lambda_j)}}_{(2b)} \underbrace{\exp\left\{-\frac{2(1-ca_j)(\mu_j^* - \lambda_j)}{3\lambda_j(ca_j + cs_j)}\right\}}_{(2c)}$$

$$\cdot[\underbrace{\frac{2\xi_j\mu_j^* - \xi_j\lambda_j}{\mu_j^*(\mu_j^* - \lambda_j)}}_{(2d)} + \underbrace{\frac{2\xi_j(1-ca_j)}{3\lambda_j(ca_j + cs_j)}}_{(2e)}].$$

When comparing the terms in the two equations we see that $(2a) > (1a)$, $(2b) = (1b)$, $(2c) = (1c)$, $(2d) > (1d)$ and $(2e) > (1e)$ if we assume $\xi_j > 1$. Therefore, inequality (25) holds and the proof of the first statement is concluded.

2. The proof of the second statement is similar to the preceding proof. In order to show that $c^{\mathrm{T}}\tilde{\mu} < c^{\mathrm{T}}\mu^*$ we show that $\tilde{\mu}_j < \mu_j^*, \forall j \in \mathrm{J}$ where $\xi > 1$. This means, following the arguments of the proof of the first statement, we need to show that

$$-\frac{\partial \tilde{S}_j(\mu_j^*)}{\partial \mu_j} < -\frac{\partial S_j(\mu_j^*)}{\partial \mu_j}. \tag{26}$$

The substitution of $\mu_j = \mu_j^*$ in $\tilde{S}_j(\mu_j)$ yields

$$-\frac{\partial \tilde{S}_j(\mu_j^*)}{\partial \mu_j} = \underbrace{\frac{1}{\xi_j(\mu_j^*)^2}}_{(3a)} + \underbrace{\frac{(ca_j + cs_j)\lambda_j}{2\xi_j\mu_j^*(\xi_j\mu_j^* - \lambda_j)}}_{(3b)} \underbrace{\exp\left\{-\frac{2(1-ca_j)(\xi_j\mu_j^* - \lambda_j)}{3\lambda_j(ca_j + cs_j)}\right\}}_{(3c)}$$

$$\cdot[\underbrace{\frac{2\xi_j^2\mu_j^* - \xi_j\lambda_j}{\xi_j\mu_j^*(\xi_j\mu_j^* - \lambda_j)}}_{(3d)} + \underbrace{\frac{2\xi_j(1-ca_j)}{3\lambda_j(ca_j + cs_j)}}_{(3e)}].$$

When comparing the terms in the two equations we see that $(3a)<(1a)$, $(3b)<(1b)$, $(3c)<(1c)$, $(3d)<(1d)$ and $(3e)>(1e)$ if we assume $\xi_j > 1$. While the first three inequalities are easy to see the fourth inequality needs some basic calculus which we don't show here for conciseness.

In order for inequality (26) to hold we finally have to show that the influence of the last term $(3e)>(1e)$ is insignificant in comparison to the four other terms. Therefore, we multiply it with $(3b)$ and $(1b)$, respectively, and compare the products. With some rearranging we end up with

$$\underbrace{\frac{1-ca_j}{3\mu_j^*(\xi_j\mu_j^*-\lambda_j)}}_{(3b)\cdot(3e)} < \underbrace{\frac{1-ca_j}{3\mu_j^*(\mu_j^*-\lambda_j)}}_{(1b)\cdot(1e)}$$

which concludes the proof of the second statement.

This means, with the two proofs we know that $\tilde{\mu}_j = \mu_j^*, \forall j \in \mathsf{J}_e, e \in \mathsf{E}_>$ where $\xi_j = 1$ and $\tilde{\mu}_j \in (\mu_j^*/\xi_j, \mu_j^*), \forall j \in \mathsf{J}_e, e \in \mathsf{E}_>$ where $\xi_j > 1$ for product families incurring a penalty. For penalty free paths we can deduce that $\tilde{\mu}_j \le \mu_j^*, \forall j \in \mathsf{J}_e, e \in \mathsf{E}_=$ where $\xi_j = 1$ and $\tilde{\mu}_j \in (\mu_j^*/\xi_j, \mu_j^*), \forall j \in \mathsf{J}_e, e \in \mathsf{E}_=$ where $\xi_j > 1$ in order for the actual mean approximate total sojourn time to be equal to the contractually defined maximum mean sojourn time.

3. The proof for the third statement follows instantly from the two preceding proofs. Thus, the proof is concluded. □

Due to the equality constraint (10) of the optimization problem solving (CAP) for *Case 2*, increases and reductions in service rate at the work stations will balance each other out. Mean service rates will increase at work stations with improvement and decrease or remain constant at work stations without improvement (although it may seem counterintuitive to consciously decrease service rate, i.e., increase waiting time, at nodes without improvement).

Optimized spare parts management (see Subsection 3.1.3.1) and preparation of service does not only lead to an increased service rate, but also to a reduction of service time variability. Additionally, collaborative scheduling of engine arrival slots as introduced in Subsection 3.1.3.2 reduces the interarrvial time variability.

In order to obtain some structural insights regarding the effects of variability reduction, we introduce $\hat{cs}$ and $\breve{ca}$ such that $\hat{cs}_j \le cs_j$ and $\breve{ca}_j \le ca_j, \forall j \in \mathsf{J}$. If we define the optimal solutions of the original and the updated optimization problems as $\mu^*, \hat{\mu}$ and $\breve{\mu} \in \mathbf{dom}\ \mathsf{C}$, respectively, we can derive the following insights.

Proposition 5 *For a single node and assuming constant service rate the following statements hold:*

1. *A reduction of service time variability* $\hat{cs} < cs$ *leads to a reduced sojourn time,* $\hat{S}(\hat{\mu}) < S(\mu^*)$.

2. *A reduction of interarrival time variability* $\breve{ca} < ca$ *leads to a reduced sojourn time,* $\breve{S}(\breve{\mu}) < S(\mu^*)$.

Proof.

1. We need to show that $S(\cdot)$ is strictly monotonically increasing in cs, i.e., $\partial_{cs} S(\mu) > 0, \forall \mu \in \mathbf{dom}\ \mathrm{C}$.

$$\frac{\partial}{\partial cs}[\frac{1}{\mu}+\frac{(ca+cs)\lambda}{2}\frac{1}{\mu(\mu-\lambda)}\exp\left\{\frac{-2(1-ca)(\mu-\lambda)}{3\lambda(ca+cs)}\right\}]$$

$$=[\underbrace{\frac{\lambda}{2\mu(\mu-\lambda)}}_{>0}+\underbrace{\frac{1-ca}{3\mu(ca+cs)}}_{>0}]\underbrace{\exp\left\{\frac{-2(1-ca)(\mu-\lambda)}{3\lambda(ca+cs)}\right\}}_{>0}>0$$

This concludes the proof of the first statement.

2. The proof of the second statement is omitted as it is similar to the preceding proof. Thus, the proof is concluded □

In fact, although it follows intuitively from Proposition 5, it is cumbersome to prove that total costs will be reduced if service and interarrival time variabilities are reduced. This is due to the interdependence of the nodes described in Section 3.1.3.3.3. More precisely, a reduction of service or external interarrival time variabilities will always have an impact on the interarrival time variabilities at the nodes in the network due to interdeparture time variability equation (32). On the other hand, a reduction of the external interarrival time variabilities has no influence on the service time variabilities at the nodes. When comparing the effects of reduced service and interarrival time variability for a single node, we can derive the following proposition.

Proposition 6 *Given the same parameters, the cost reduction through reduced interarrival time variability $\breve{c}a$ exceeds the reduction through reduced service time variability $\hat{c}s$ for a single node.*

Proof. As capacity costs depend on the negative sum of slopes $\partial_{\mu_j} S_j(\mu_j)$ of the mean sojourn times at the nodes, we need to show that

$$\frac{\partial}{\partial ca}\left[\frac{\partial S}{\partial \mu}\right]<\frac{\partial}{\partial cs}\left[\frac{\partial S}{\partial \mu}\right] \tag{27}$$

for the same parameters ca, cs, μ and λ. Therefore, we first find the derivatives.

$$\frac{\partial}{\partial cs}\left\{-\frac{1}{\mu^2}-\frac{(ca+cs)\lambda}{2\mu(\mu-\lambda)}\exp\left\{\frac{-2(1-ca)(\mu-\lambda)}{3\lambda(ca+cs)}\right\}\left[\frac{2\mu-\lambda}{\mu(\mu-\lambda)}+\frac{2(1-ca)}{3\lambda(ca+cs)}\right]\right\}$$

$$=-\exp\left\{\frac{-2(1-ca)(\mu-\lambda)}{3\lambda(ca+cs)}\right\}\left[\frac{\lambda(2\mu-\lambda)}{2\mu^2(\mu-\lambda)^2}+\frac{(2\mu-\lambda)(1-ca)}{3\mu^2(\mu-\lambda)(ca+cs)}\right.$$

$$\left.+\frac{2(1-ca)^2}{9\lambda\mu(ca+cs)^2}\right]$$

Since all individual terms are positive we see that $\partial_{cs}[\partial_\mu S]<0$ is strictly monotonically decreasing.

The other derivative is

$$\frac{\partial}{\partial ca}\left\{-\frac{1}{\mu^2}-\frac{(ca+cs)\lambda}{2\mu(\mu-\lambda)}\exp\left\{\frac{-2(1-ca)(\mu-\lambda)}{3\lambda(ca+cs)}\right\}\left[\frac{2\mu-\lambda}{\mu(\mu-\lambda)}+\frac{2(1-ca)}{3\lambda(ca+cs)}\right]\right\}$$

$$=-\exp\left\{\frac{-2(1-ca)(\mu-\lambda)}{3\lambda(ca+cs)}\right\}\left[\frac{\lambda(2\mu-\lambda)}{2\mu^2(\mu-\lambda)^2}+\frac{(2\mu-\lambda)(1-ca)}{3\mu^2(\mu-\lambda)(ca+cs)}\right.$$

$$\left.+\frac{2(1-ca)^2}{9\lambda\mu(ca+cs)^2}+\frac{2\mu-\lambda}{3\mu^2(\mu-\lambda)}+\frac{2(1-ca)}{9\lambda\mu(ca+cs)}-\frac{1}{3\mu(\mu-\lambda)}\right].$$

In order for Equation (27) to hold we therefore need to show that

$$\frac{2\mu-\lambda}{3\mu^2(\mu-\lambda)}+\frac{2(1-ca)}{9\lambda\mu(ca+cs)}>\frac{1}{3\mu(\mu-\lambda)}.$$

If we multiply both sides with $3\mu(\mu-\lambda)$ we get

$$\underbrace{\frac{2\mu-\lambda}{\mu}}_{>0}+\underbrace{\frac{2(1-ca)(\mu-\lambda)}{3\lambda(ca+cs)}}_{>0}>1.$$

Since

$$\frac{2\mu-\lambda}{\mu}=2-\frac{\lambda}{\mu}>1$$

and $\mu>\lambda$ the proof is concluded. □

On the other hand, if $\lambda/\mu>\sqrt{0.5}$, service time variability has the higher impact on interdeparture time variability, see Equation (32). We can also see that the effects of scheduling increase for lower service time variability, meaning that maintenance demand forecasting and optimized planning foster the benefits of scheduling.

Due to the complexity of the queueing network performance parameter algorithm, we numerically illustrate how an improvement of the external interarrival time variabilities $\breve{ca}_e$ and service time variabilities $\hat{cs}_j$ affect sojourn times, capacity costs and total costs in the subsequent section.

3.1.3.3.6 Numerical example

In order to illustrate the benefits of collaborative maintenance management, we consider the DTA aircraft engine overhaul case introduced in the previous sections.

We use a queueing network with $J=11$, i.e., 11 work stations and two paths through the network, $\mathrm{J}_1=\{1,3,4,5,7,8,10,11\}$ and $\mathrm{J}_2=\{2,3,4,6,7,9,10,11\}$ for engine types $e\in\{1,2\}$, respectively. The external arrival rates for the two engine types are given by $\lambda_1=5$ and $\lambda_2=8$. The squared coefficients of variation of the external arrivals are given by $ca_1=ca_2=1$ as we assume Poisson processes without arrival scheduling. As cost parameters we choose $c_j=1, \forall j\in\mathrm{J}$, and $\gamma_1=20$, $\gamma_2=22$.

The queueing network parameters of the individual nodes resulting from the computations described in Section 3.1.3.3.3 are summarized in Table 32. The values for ca_j were obtained using an initial guess of $\mu_j^* = \lambda_j + 2$.

Node j	λ_j	cs_j	ca_j
1	5	0.0443	1.0000
2	8	0.0443	1.0000
3	13	0.0443	0.4699
4	13	0.1173	0.1502
5	5	0.0644	0.2618
6	8	0.0644	0.2107
7	13	0.7532	0.1869
8	5	0.2834	0.3592
9	8	0.2834	0.4540
10	13	0.0281	0.2539
11	13	0.0281	0.0843

Table 32: Numerical example queueing network parameters.

If we assume that service provider and customers are considering a collaboration to reduce costs and mean turnaround times, we are now able to estimate the benefits. We illustrate the benefits of supply chain collaboration for Scenario a) where $S_1^T = S_2^T = 0.7$. Total cost for Poissonian demand $ca_e = 1, \forall e \in \mathrm{E}$, $c_j = 1, \forall j \in \mathrm{J}$ and cs as defined in Table 32 is given by 150.33.

The collaboration is assumed to impact the production network in three dimensions. Firstly, the mean service rates are increased through forecasting of maintenance demand. If we consider the network shown in Figure 32, the NDT (Non-Destructive Test) and inspection process steps are accelerated since more information is known in advance through continuous sharing of real time sensor data. Optimized spare parts management and service preparation lead to faster lead time for the repair and assembly process steps. Therefore, as a numerical example, we assume that the optimization of the mean service rates denoted as $A\mu$ where $A \in \mathbf{R}_+^{J \times J}$ is a matrix where the elements on the main diagnoal are given by by $\xi_j = 1.5, \forall j \in \{4,5,6,7,8,9\}$ and $\xi_j = 1, \forall j \in \{1,2,3,10,11\}$ and all other elements are zero.

Secondly, the service time variability is also reduced due to optimized spare parts management and service preparation. As a numerical example, we choose at the corresponding repair and assembly nodes $\hat{c}s_j = cs_j/2, \forall j \in \{7,8,9\}$.

Finally, we assume collaborative scheduling of engine arrivals such that the interarrival time variabilities are minimized. As numerical example, we choose $\check{c}a_e = 0.1, \forall e \in \mathrm{E}$, i.e, close to constant interarrival times to illustrate the effects. The numerical results for different combinations of the three effects are summarized in Table 33.

$A\mu$	$\hat{cs}$	$\check{c}a_e$	$\sum_{j\in\mathcal{J}_1} S_j$	$\sum_{j\in\mathcal{J}_2} S_j$	$\mathcal{C}(\mu^\star)$	Cost reduction
-	-	-	1.08	1.02	150.3	0%
•	-	-	1.00	0.94	126.9	16%
-	•	-	1.05	0.98	145.9	3%
-	-	•	1.01	0.92	139.0	8%
•	•	-	0.98	0.91	123.2	18%
•	-	•	0.94	0.85	116.3	23%
-	•	•	0.98	0.88	134.4	11%
•	•	•	0.92	0.82	112.5	25%

Table 33: Numerical test of benefits through collaboration.

We see that in this numerical example the improvement of mean service rates has the highest impact on total cost. Collaborative scheduling of engine arrivals exhibits the second highest cost reduction. The reduction of service time variabilities has the smallest effect on cost. On the other hand, in many cases it will come along with the improvement of mean service rates and therefore with little additional implementation costs. In the columns displaying the sums of the approximate mean sojourn times on paths J_1 and J_2 we see that not only total costs but also mean turnaround times are reduced as stated in Proposition 4.

If we consider the case where only the mean service rates are increased and compare the resulting optimal service rates μ^* and $\tilde{\mu}$, we obtain

$$\mu^* ./ \tilde{\mu} = [1.00 \;\; 1.00 \;\; 1.00 \;\; 1.44 \;\; 1.39 \;\; 1.43 \;\; 1.42 \;\; 1.38 \;\; 1.41 \;\; 1.00 \;\; 1.00]^{\mathrm{T}}$$

where $\mu^* ./ \tilde{\mu}$ is the element-wise division, which verifies the statement in the proof of Proposition 4 that $\tilde{\mu}_j = \mu_j^*, \forall j \in \mathrm{J}_e, e \in \mathrm{E}_>$ where $\xi_j = 1$ and $\tilde{\mu}_j \in (\mu_j^*/\xi_j, \mu_j^*), \forall j \in \mathrm{J}_e, e \in \mathrm{E}_>$ where $\xi_j > 1$ for product families incurring a penalty.

With the information displayed in Table 33 and knowledge regarding implementation costs of the collaborative maintenance management system, service provider and customers can decide whether to consider the investment or discard the opportunity. Of course, to do so, the benefits for the individual players in the system need to be evaluated. If we, as an example, assume that there exists only one customer who pays a price $p = 14$ per overhauled engine, total expected revenues R per time period for the MRO service provider are given by $R = (\lambda_1 + \lambda_2)p = (5+8)14 = 182$. Since costs per time period are given by $\mathrm{C}(\mu^*) = 150.3$, the service provider's net profit P is given by $P = R - \mathrm{C}(\mu^*) = 31.7$. As total costs of the service provider are composed of capacity costs and penalty costs, $\mathrm{C}(\mu^*) = C_{\mathrm{cap}} + C_{\mathrm{pen}}$, with $C_{\mathrm{cap}} = c^{\mathrm{T}}\mu^* = 135.7$ and $C_{\mathrm{pen}} = 14.6$ and the penalties are payed back to the customer, the net price per engine changes to $p_{\mathrm{net}} = p - C_{\mathrm{pen}}/(\lambda_1 + \lambda_2) = 12.9$.

If we now assume a cost reduction through collaborative maintenance management to $\tilde{C}(\tilde{\mu}) = 112.5$, net profit for the service provider would more than double. But since implementation costs occur for the supplier as well as the customer, the benefit must be shared through a price reduction $\tilde{p} < p$, e.g., $\tilde{p} = 11.5$. With this reduction, the updated net profit $\tilde{P}$ for the supplier is given as $\tilde{P} = (\lambda_1 + \lambda_2)\tilde{p} - \tilde{C}(\tilde{\mu}) = (5+8)11.5 - 112.5 = 37.0$, i.e., an increase of net profit by 17% or 5.4 in absolute numbers. On the other hand, as penalty costs of $\tilde{C}_{\text{pen}} = 7.0$ occur, the new net price per engine for the customers is given by $\tilde{p}_{\text{net}} = \tilde{p} - \tilde{C}_{\text{pen}}/(\lambda_1 + \lambda_2) = 11.0$. This corresponds to a cost reduction of 15% for the customer, or 25.0 in absolute terms per time period. Therefore, depending on the implementation costs per party, a fair benefit sharing rule can be established easily and cost-effectiveness can be assessed quickly. Additionally, customers also benefit from reduced mean turnaround times which can be translated in a reduction of the total number of engines needed for flight operations.

3.1.3.3.7 Implementation

The core algorithm of the capacity planning mechanism is based on the subgradient method in order to have one algorithm to solve the problem independent of the problem characteristics.

Algorithm 3: Subgradient method for capacity planning

$\mu^{(1)} \leftarrow \lambda + 1$ *// $\mu^{(k)}$and λ are J-dimensional vectors*

$C \leftarrow zeros(1, K)$ *// K is the maximum number of iterations*

$C^{(1)} \leftarrow c^T\mu^{(1)} + \sum_{e=1}^{E} \gamma_e \max\{0, \sum_{j\in J_e} S_j\left(\xi_j \mu_j^{(1)}\right) - S_e^T\}$ *// Starting point*

For $k = 1{:}K$

    $g \leftarrow zeros(1, J)$

    For $e = 1{:}2$ *// Computation of subgradient*

        If $\sum_{j\in J_e} S_j\left(\xi_j \mu_j^{(k)}\right) > S_e^T$

            $g \leftarrow g + c^T + \gamma_e \nabla \sum_{j\in J_e} S_j\left(\xi_j \mu_j^{(k)}\right)$

        else if $\sum_{j\in J_e} S_j\left(\xi_j \mu_j^{(k)}\right) = S_e^T$

            $r \leftarrow rand(1)$

            $g \leftarrow g + c^T + \gamma_e \nabla r \sum_{j\in J_e} S_j\left(\xi_j \mu_j^{(k)}\right)$

        end

    end

    $\mu^{(k+1)} \leftarrow \mu^{(k)} - \alpha g$

    $C^{(k+1)} \leftarrow c^T\mu^{(k+1)} + \sum_{e=1}^{E} \gamma_e \max\{0, \sum_{j\in J_e} S_j\left(\xi_j \mu_j^{(k+1)}\right) - S_e^T\}$

    $C_{best}^{(k+1)} \leftarrow \min\{C_{best}^{(k)}, C^{(k+1)}\}$ *// Remember optimal cost*

    $k_{best} \leftarrow \arg\min_k C^{(k)}$

18: $\mu^* \leftarrow \mu^{(k_{best})}$ *// Remember capacity associated to optimal cost*

19: **end**

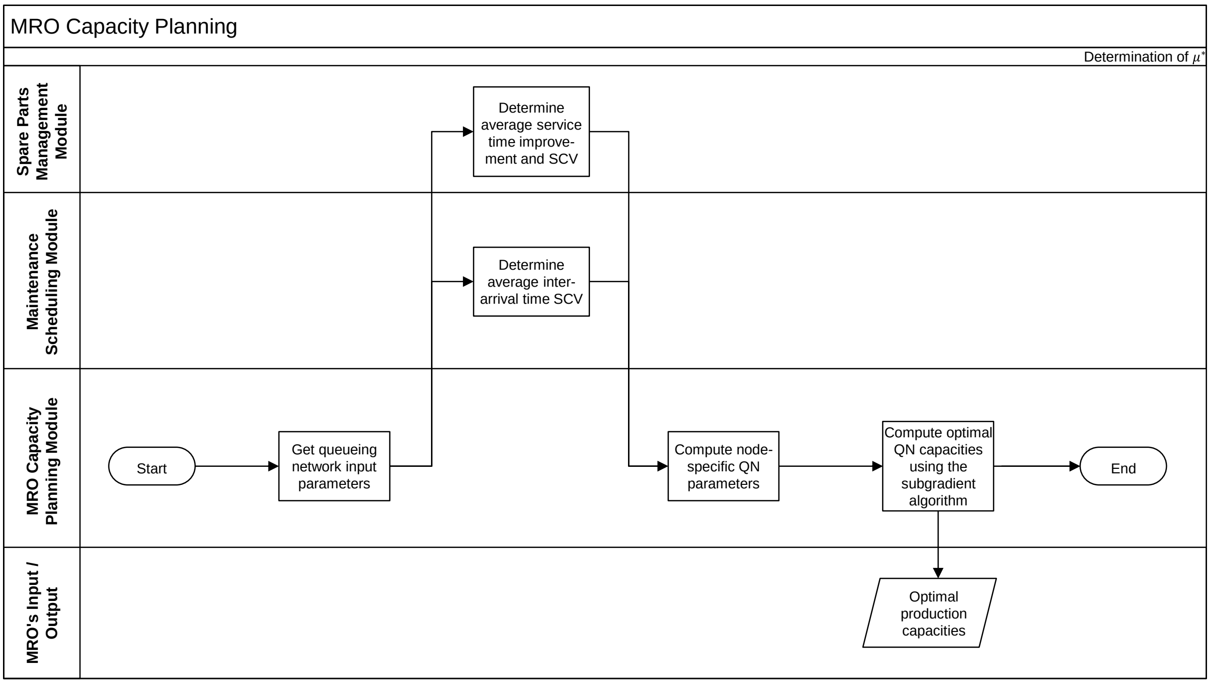


Figure 35: Capacity planning module.

3.1.3.3.8 Data sensitivity assessment

Variable	Description	Party concerned	Security requirements
C	Total overhaul cost for MRO (output of algorithm)	MRO	High security requirements – total cost should not be visible for the MRO's customer as it could influence contract decisions and negotiations
μ^*	Optimal production capacities in the MRO network (output of algorithm)	MRO	High security requirements – capacities should not be visible for the MRO's customers as they could estimate total costs from production capacities
λ	Mean arrival rate vector (input for algorithm)	MRO	Medium security requirements – the MRO may want to protect the total mean demand served in order to keep his bargaining position
ca	Vector of SCVs of interarrival times (input for algorithm)	MRO	Medium security requirements – the MRO may want to protect the aggregated actual interarrival time variabilities of all customers in order to keep his bargaining position
cs	Vector of SCVs of service times (input for algorithm)	MRO	Medium security requirements – the MRO may want to protect the service

Variable	Description	Party concerned	Security requirements
			rate variabilities induced by his work stations in order to keep his bargaining position
γ	Penalty costs vector (input for algorithm)	MRO, customer	Not sensitive as known to all parties involved
S_e^T	Maximum mean contractually defined turnaround times (input for algorithm)	MRO, customer	Not sensitive as known to all parties involved

Table 34: Data sensitivity assessment for capacity planning.

3.1.3.3.9 Conclusion and further references

In the preceding subsections the problem of determining the cost-minimizing capacity in an acyclic production network for a service provider setting with contractually defined lead time requirements and associated penalty costs was solved. The production network was described as a network of GI/G/1 queues and through classification of different cases distinguished by the differentiability of the objective function we were able to find analytical and numerical models to solve the optimization problem.

Additionally, we considered a collaborative maintenance management scenario where service provider and customers jointly employ advanced information available through sensors on aircraft engines in order to further minimize maintenance costs. As the associated collaborative planning system is an investment for all parties involved, we show how to compute the expected benefits under improvement assumptions for the queueing network's input parameters and numerically illustrate one example scenario.

Although the capacity planning algorithm must not necessarily be implemented in the cloud, it can be used to compute the benefits of the the cloud-based collaborative planning modules: collaborative maintenance demand forecasting, spare parts management and maintenance scheduling. It is therefore the first step regarding benefit analysis subject to PRACTICE deliverable D24.5.

Further references regarding queueing network analysis can be found in Whitt (1994) or Bitran and Morabito (1996). Hopp et al. (2002) use queueing networks to determine optimal production capacities in a semiconductor fab. Da Silva and Morabito (2009) determine how to optimally allocate capacity in a job shop-like queueing network of a metallurgical plant. They use approximate parametric decomposition models to compute performance metrics such as WIP and lead time and apply optimization models to minimize capacity costs.

3.1.4 Summary

In Sections 3.1.2 and 3.1.3 we have developed formal models for collaborative maintenance management in the aerospace industry. We defined multiple collaborative forecasting mechanisms depending on data availability and developed three models for collaborative planning of maintenance activities: collaborative spare parts management, scheduling of engine arrival slots and capacity planning for the job-shop like production network of the MRO.

In all subsections we not only described the mathematical models but also the according algorithms that need to be implemented. We also provided process flow charts for the different modules, indicating their interdependencies. Finally, we also considered data sensitivity issues as this is needed in order to securely implement the entire collaborative planning system in a cloud environment.

In conclusion, all information to start implementation of the system is available. In order to provide a holistic view of the entire collaborative planning system architecture we provide an overview of the modules, their triggers and the corresponding time scale in Figure 36.

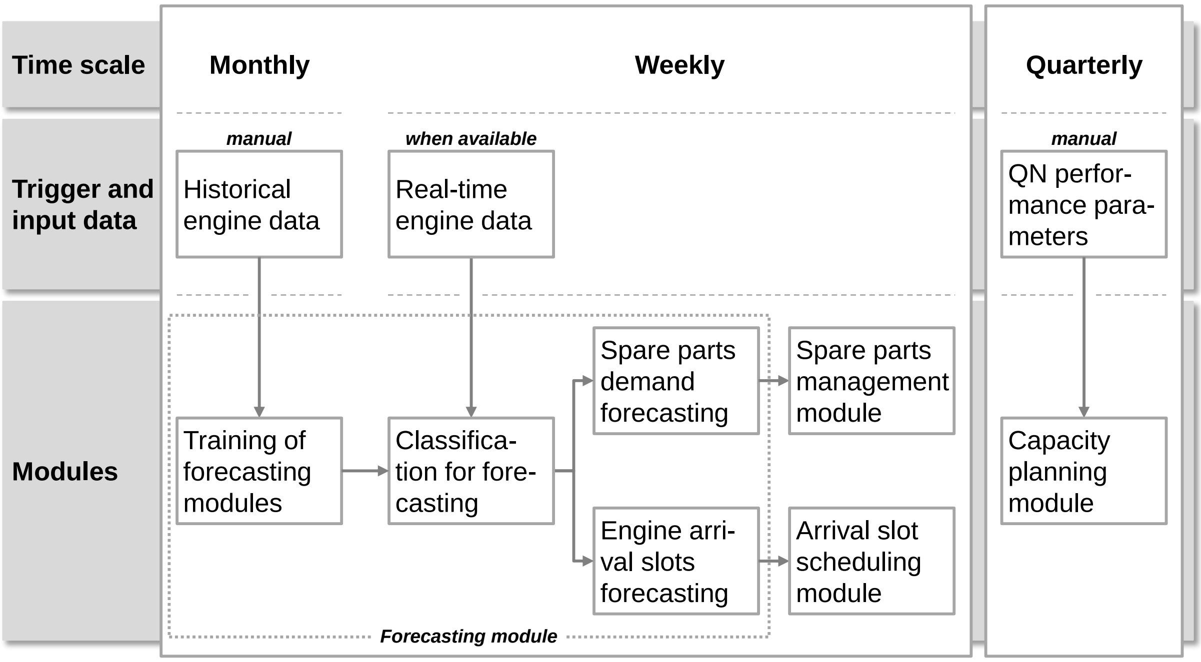


Figure 36: Overview collaborative maintenance management implementation.

We see that every month the forecasting module must be trained based on the data collected until the date of the training. The training of the module is triggered manually every month. As soon as new real-time data is available from the engines (which is approximately weekly or fortnightly), the classification and actual forecasting modules are triggered. The entire forecasting module is described in Section 3.1.2.

New forecasts then trigger the spare parts management and arrival slot scheduling modules described in Subsections 3.1.3.1 and 3.1.3.2, respectively, automatically.

The capacity planning module is triggered manually every three months as capacity planning needs stability in order to avoid high overtime and flexibility costs. As described in Subsection 3.1.3.3, the capacity planning modules takes into account the performance of the spare parts management and arrival slot scheduling modules.

3.2 Secure cloud-based Vendor Managed Inventory in the consumer goods industry

3.2.1 Overview and introduction

Arçelik A.S (in the following referred to as ARC) is a household appliance manufacturer in Turkey. They supply to direct customers and Arcelik's subsidiaries. Direct customers are for example large retail stores in many countries. They work independently, so ARC has no access to information besides the incoming (pre-)orders. Subsidiaries are part of the ARC consortium so inventory and point of sales data are available for the ARC order management department. The supply chain is illustrated in Figure 37

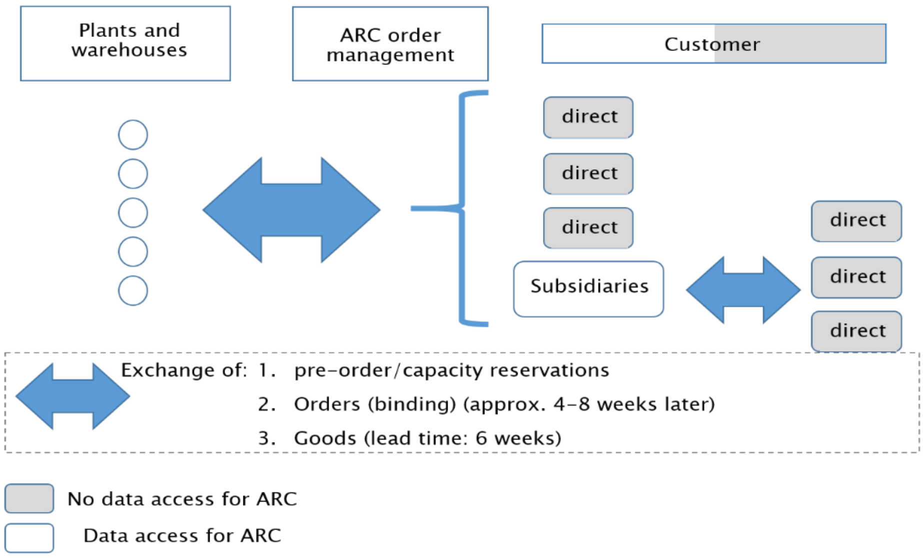


Figure 37: ARC supply chain

Due to long lead-times of approximately 6 weeks all customers have to place pre-orders. Based on these pre-orders the ARC order management department makes capacity reservations in their various plants and warehouses. About 4-8 weeks after the pre-ordering the customers place their binding orders.

The problem ARC is facing are significant deviations between pre-orders and actual orders. This causes severe misallocation of resources since they are planned based on the pre-orders and the flexibility to deal with short-time deviations is limited.

Possible explanations for this problem are:

- Customers have incentives to inflate pre-orders to make sure that their actual orders will be available on time
- Volatile markets indeed cause fluctuation in demand
- Customers' abilities to forecast uncertain demand are limited

In this chapter we propose two concepts to overcome these issues. On the operational level (medium-term planning horizon) we propose a collaborative forecasting mechanism that allows aggregating the customers' forecasts without the need to reveal each individual's input. Customer forecasts can be obtained from their forecasting systems or derived from secure linear regression analysis of POS data.[20] Both should limit the chances to manually inflate their individual forecasts. This concept is described in subsection 3.2.2.

On the tactical level (short-term planning horizon) we propose a concept for secure vendor managed inventory. In this scenario the vendor (i.e. ARC) is responsible for the inventory of its customers (i.e. the retailers of ARC products). This would tackle all three dimensions of the the problem. The concept is described in Section 3.2.3.

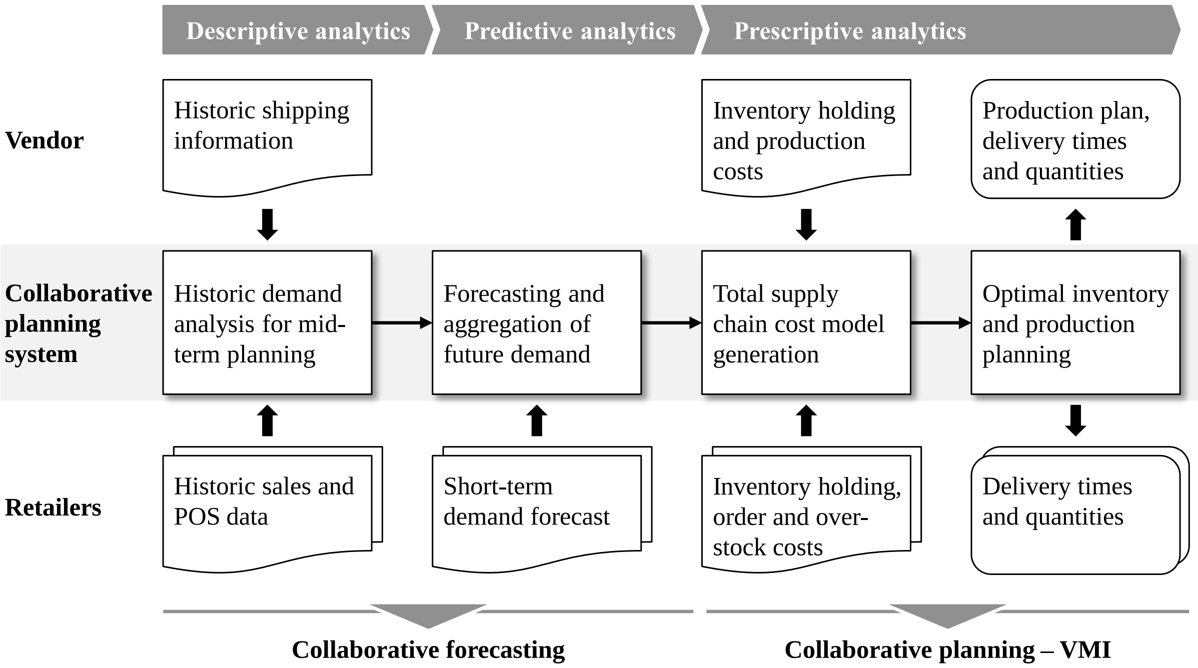


Figure 38: Forecasting and planning processes in the consumer goods case

The rationale behind the framework depicted in Figure 38 is to first improve the sales forecasting of the supply chain as a whole and then optimize the distribution of the inventory in the supply chain through Vendor Managed Inventory. Since, according to Sari (2008 p. 577) VMI alone tends to make ineffective use of retail-level information. The implementation of this framework should ideally result in

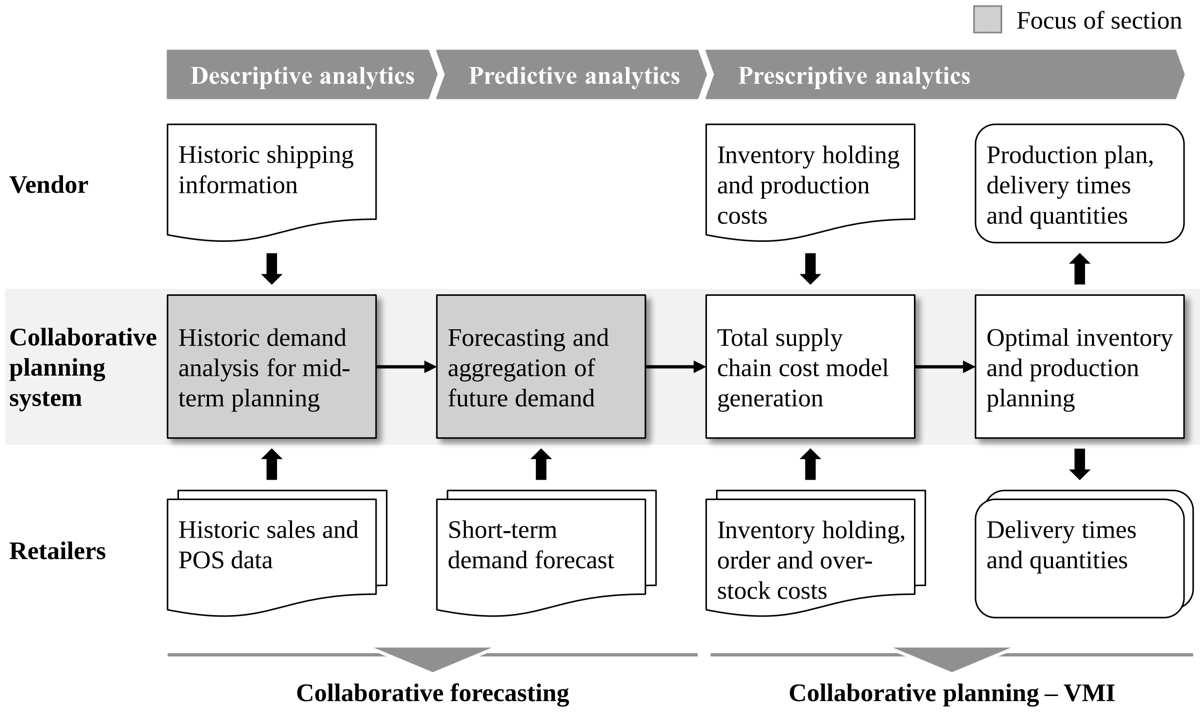
- a leaner supply chain (measured by inventory turnover),
- a more responsive supply chain with improved product availability (measured in customer service level) and
- a more cost effective inventory system (measured in total inventory costs of the supply chain)

Collaborative forecasting will improve the demand visibility throughout the supply chain. By generating a collaborative forecast both parties can combine their knowledge to generate a common forecast that correctly depicts the end customer's demand. The collaborative forecast can then be used as a starting point for further VMI activities.

[20] The assumption of linear demand (i.e. fixed demand rates) is quite common in literature. Typically, this rate is constant for each time period. Estimating a new demand rate in each period based on current POS data could improve the accuracy.

3.2.2 *Collaborative forecasting for mid-term production planning*[21]

In this section we focus on the operational planning activities with a mid-term planning horizon. The main goal is to find an alternative approach to solve the problems regarding inaccurate or inflated pre-orders. We predict the aggregated demand for a product that is sold to multiple customers 6-8 weeks ahead. The results can then be used by the ARC order management to reserve capacities in their plants and warehouses.



3.2.2.1 Secure aggregation protocol

The basic idea is that ARC does not need the individual data of each customer but the aggregated numbers as depicted in Figure 39. These can be computed without the need to reveal the individual input data, e.g. using the concept of (additive) homomorphic encryption. We note that the rationale here is quite similar to the one in subchapter 3.1.2.2. There we wanted to aggregate expected demands for overhaul services from different customers. Here we need to aggregate expected demand for a product sold by different customers (which are retail companies in the ARC scenario). Hence, the basic infrastructure for this problem can be used in various industrial scenarios once it is available in a cloud-based platform.

As an example for the consumer goods scenario consider a product x and three customers reselling this product called A, B and C. Their internal forecasts for product x are named x_A, x_B and x_C. For a solid capacity planning, ARC needs to know the overall expected demand for product x.They want to know the sum over all individual forecasts: $S_x := x_A + x_B + x_C$.

As in subchapter 3.1.2.2.3 we just note here that there exist concepts that provide this form of secure aggregation, e.g. the additive homomorphic encryption scheme by Paillier (1999). It will be part of the next deliverable to discuss in detail how such an encryption scheme could be applied in the given scenario and how it maps with the Practice architecture.

[21] Main authors: Fabian Taigel (UWUERZ), Florian Hahn (SAP)

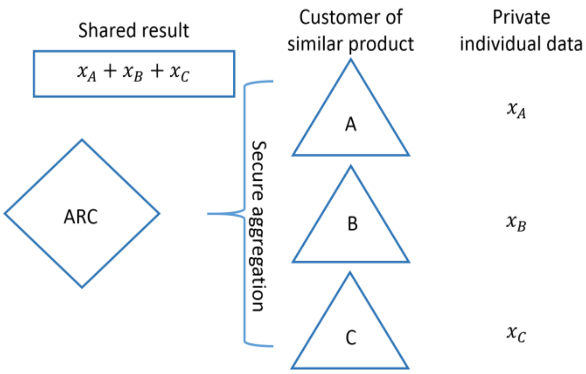


Figure 39 Basic concept of secure forecast aggregation

The important point will be that none of the parties can learn the input data of another individual party neither can the computing party. Furthermore, ARC does get the aggregation and only the aggregation and is not able to reconstruct values of individual parties.

3.2.2.2 Implementation

The process flow for the implementation of the secure aggregation protocol described in the previous subchapters is described in Figure 40.

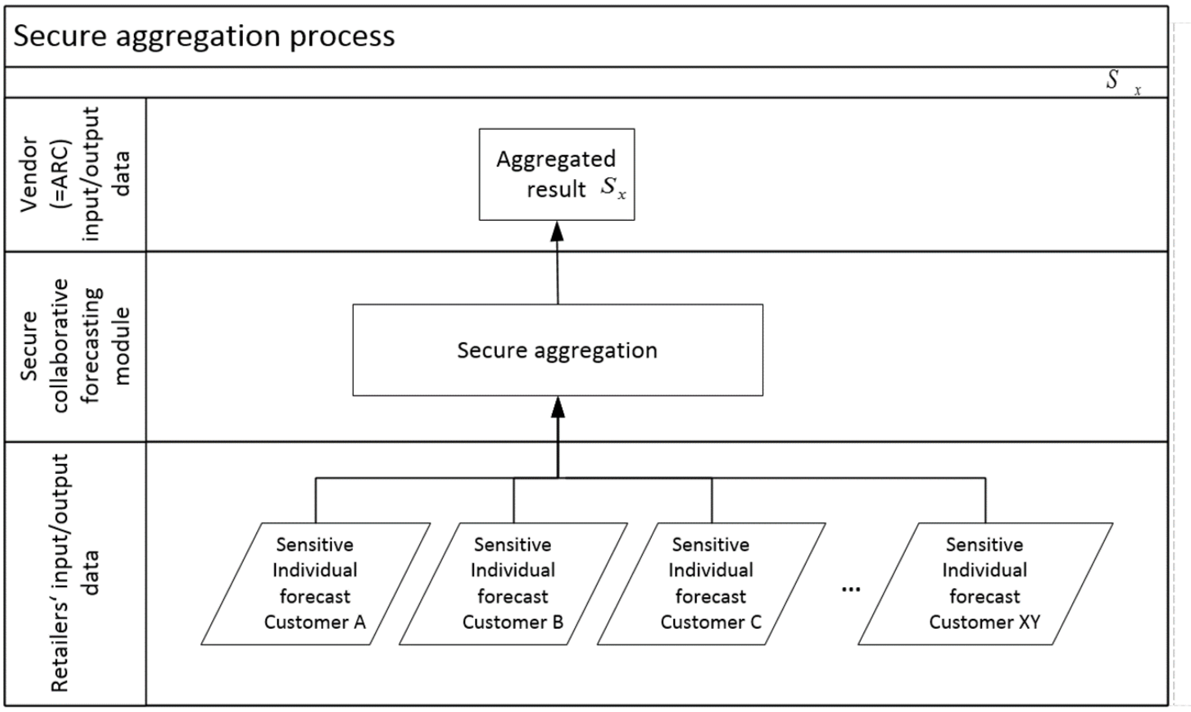


Figure 40: Secure aggregation protocol

3.2.2.3 Data sensitivity assessment

Variable	Description	Party concerned	Security requirements
$x_A, x_B, \ldots$	Individual forecasts for product x from customers A, B, …	customers	High security requirements – individual forecasts must not be visible to competitors and should not be visible to ARC as this might inhibit participation
S_x	Aggregated forecast for product x	MRO customers	Low security requirements – allows no inference to individual inputs and could even be shared amongst supply chain partners as additional market information

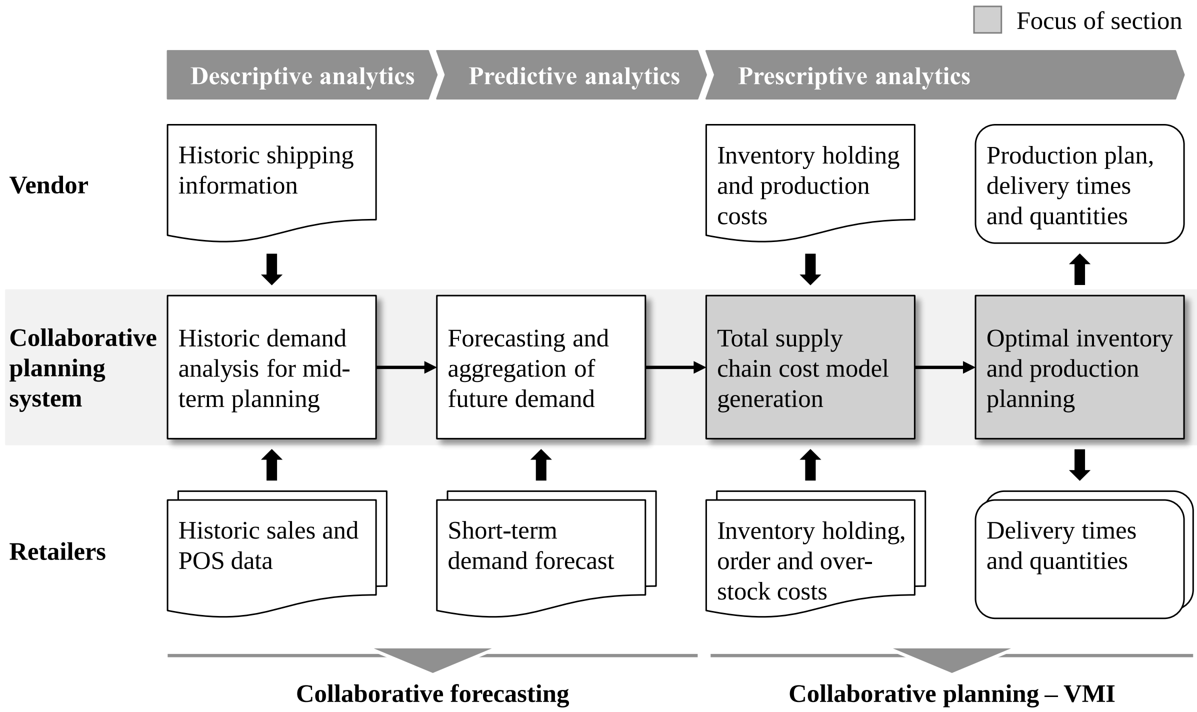
3.2.2.4 Conclusion and possible extensions

In the previous subchapters we outlined the need for a protocol that allows us to compute the sum over multiple inputs without revealing these individual inputs to any other party but the original owner. Such a secure aggregation protocol can be used in the consumer goods scenario described above as well as in the aerospace industry scenario. It therefore is a versatile feature of a cloud based planning system.

A possible extension for future work to such a basic protocol in the ARC case would be to include customers' POS[22] data for advanced collaborative forecasting. This would reduce the chance for manipulation of input data through the customers even further, since it could be directly drawn from their ERP[23] systems. The idea is to obtain the forecasts that we need as inputs for the secure aggregation protocol not from the retailers' forecasting systems but from linear regression models we run on their private POS data. The output for each customer is then again aggregated via the secure aggregation protocol described above. The linear regression for one product of a specific customer is depicted in Figure 40. The output of each linear regression is the time interval when this customer will reorder Q units of a product. Aggregation over all customers gives the overall demands for each time interval.

[22] POS=point of sales
[23] ERP = Enterprise-Ressource Planning

3.2.3 Collaborative planning – secure Vendor Managed Inventory[24]



Vendor Managed Inventory is a collaborative Supply Chain Management concept between a vendor and one or more buyers (in the following called retailers) with the focus on inventory management. Even though the inventory stock remains under the ownership of the retailers, the vendor assumes the decision making authority regarding the inventory replenishment process of its retailers. Under VMI, the vendor acts as a central controller who takes decisions that benefits the supply chain as a whole and whose decisions are being obliged to by all supply chain actors. However, this can lead to adverse cost situations for specific actors of the supply chain (compared to local optimization). To counter this situation, there are several methods possible under VMI collaboration to incentivize all actors to keep up the collaboration like benefit sharing mechanisms or renegotiations of over stock penalty costs.

In the following, our focus is on an actual Vendor Managed Inventory (VMI) model that is presented for a single vendor multi-retailer supply chain operating under a Vendor Managed Inventory mode of operation that can be used to optimize the replenishment policy of, for example, ARC's supply chain. Furthermore, the model allows the opportunity to manage the VMI operation in a secure cloud-based setting as it should be able to be implemented into a cloud environment in a straightforward fashion.

Included in this mode of operation is a VMI contractual arrangement between the vendor (e.g. ARC) and its retailers that specifies certain inventory upper bounds for the retailers' stock. Each retailer negotiates its own retailer-specific upper limit of inventory (which can be influenced by e.g. storage space) as well as the penalty cost if this limit is exceeded. The contractual agreement gives the retailers a sense of control over how much inventory they have to account for and simultaneously disincentives ARC from pushing too much of the inventory downwards onto the retailers. If the stock nevertheless exceeds these bounds (which still can be beneficial for the total supply chain system cost), the retailer gets compensated for this via overstock penalty payments negotiated into the VMI contract. The aim of our VMI model is to minimize the total supply chain costs of the system as whole and distribute the inventory accordingly. The model we build on was developed by M.A. Darwish

[24] Main author: Fabian Diehm (UWUERZ)

and O.M. Odah in their 2009 paper "Vendor managed inventory model for single-vendor multi retailer supply chains" and is adjusted to better reflect the ARC use case.

3.2.3.1 Notation

For the remainder of this chapter, the following notation is used:

m:	Number of retailers
D_j:	Demand rate for retailer *j*
D:	Demand rate for the vendor ($D = \sum_{j=1}^{m} D_j$)
U_j:	Upper limit on the inventory level of retailer *j*
π_j:	Overstock penalty cost for retailer *j*
A_v:	Vendor production cost for one production lot
A_j:	Order cost for retailer *j* (incl. transportation cost)
h_v:	Vendor holding cost
h_j:	Holding cost for retailer *j*
Q:	Production lot
q_j:	Quantity dispatched to retailer *j* per shipment
q:	Total quantity dispatched from vendor to all retailers ($q = \sum_{j=1}^{m} q_j$) in one common cycle T_R
T_j:	Cycle time for retailer *j.* Time between shipments
T_R:	Common cycle for all retailers ($T_R = T_j$). Time between shipments
T:	Vendor cycle time. Time between Production lots
n:	Number of shipments (which can also be defined as the number of common cycles for all retailers T_R per vendor cycle time T: $n = \frac{T}{T_R}$)
$\lfloor n \rfloor$:	Largest integer which is less than *n*
$\lceil n \rceil$:	Smallest integer which is greater than *n*
S:	Set of all retailers whose upper limit U_j is exceeded
r:	Number of elements in the set *S*, $r = 0, 1, 2, \ldots, m$
Φ:	The empty set *S* ($r = 0$)
Ω:	The set of all retailers ($r = m$)
TC_V:	Total cost incurred by vendor
TC_j:	Total cost incurred by retailer *j*
TC_R:	Total cost incurred by all retailers ($TC_R = \sum_{j=1}^{m} TC_j$)
TC:	Total cost incurred by system ($TC = TC_V + TC_R$)

To better understand the context of our notation and see the link to the Arcelik use case we can show the main parameters within the Arcelik supply chain as seen in Figure 41.

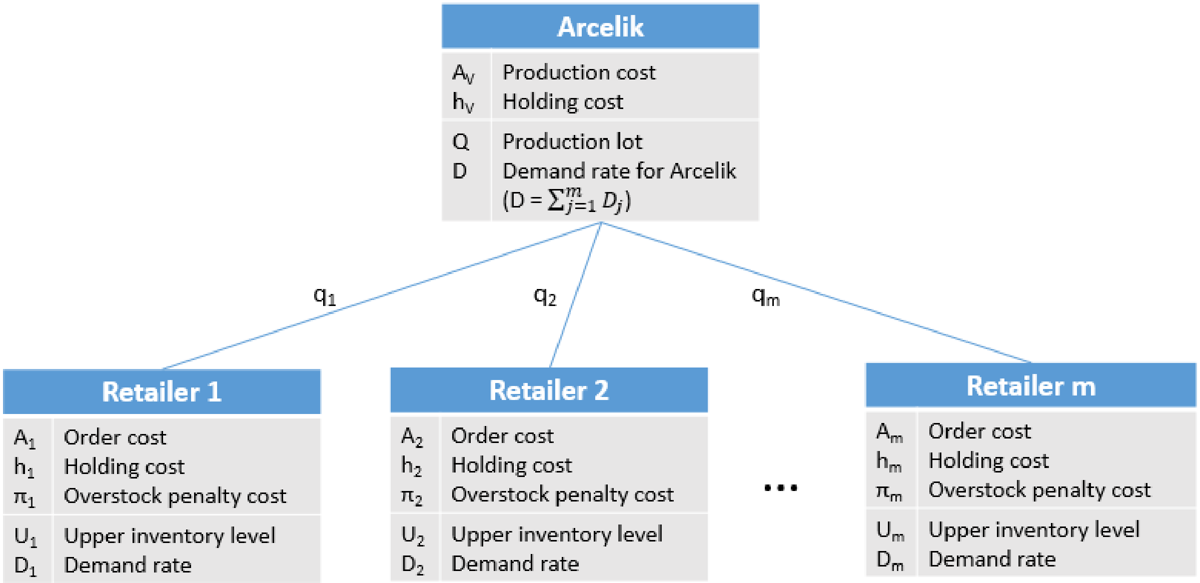


Figure 41: Model notation for VMI in the ARC supply chain.

As specified in the VMI contract, the vendor (ARC) pays its own production cost A_V that takes into account all costs that the manufacturing of one production lot Q comprises as well as the inventory holding cost h_V, which includes storage cost for storing the inventory (warehousing cost etc.) and the cost of capital tied up in stock. Characteristic for a VMI mode of operation, the vendor (ARC) also pays the order cost A_j for the retailers, which includes order-receiving costs like warehouse inbound handling. That's because retailers are exempted from ordering costs in the VMI setting. Furthermore, ARC pays the penalty cost π_j per unit when overstocking a retailer j i.e. when the inventory distributed to the retailer j exceeds the contractually agreed upon upper limit U_j.

The retailers, on the other hand, only have to pay their respective inventory holding cost h_j that, again, is comprised of storage cost for storing the inventory (warehousing costs etc.) and the cost of capital tied up in stock.

3.2.3.2 Assumptions

It is important to note some of the assumptions that are incorporated into the model. We assume that the supply chain actors (one vendor i.e. ARC and *m* retailers) share (forecasted) demand information, cost information like holding and order cost as well as storage capacity information. Some of this information however is sensitive and therefore will only be used in the cloud for model optimization without the other supply chain actors having insight into this. We will address the cloud aspect of our model in more detail in 3.2.3.6 and 3.2.3.7.

It is also important to note that it is assumed that the vendor incurs less cost of holding inventory (h_V) than the retailers (h_j). This is realistic as the vendor's physical storage costs tend to be less than that of its retailers.

Additionally we assume *deterministic demand, i.e. fixed given demand rates* . Even though this may not be an ideal reflection of the real-life environment, it enables us to show the underlying logic of a reasonable VMI replenishment policy without the model getting too convoluted.

We also assume that the *vendor replenishes all retailers at the same time* ($T_i = T_j = T_R$) which, hence, also means an equal shipment frequency (n) for all retailers. This is reasonable because under VMI, the vendor has the decision authority about the replenishment timing and by bundling the replenishment of all retailers, the vendor can save on transportation cost.

3.2.3.3 Model development

Figure 33 illustrates the inventory levels for ARC and three customers. At the time the vendor produces its production lot Q, a first delivery shipment, sized $q = q_1 + q_2 + \ldots + q_m$, is immediately delivered to m retailers. This commences a new vendor cycle time T until a new lot Q is produced, which in turn initiates a new vendor cycle time T. Each retailer j receives n shipments of equal size q_j within one vendor cycle time T, which in sum equals the jth retailer's demand rate of D_j times the vendor cycle time T.

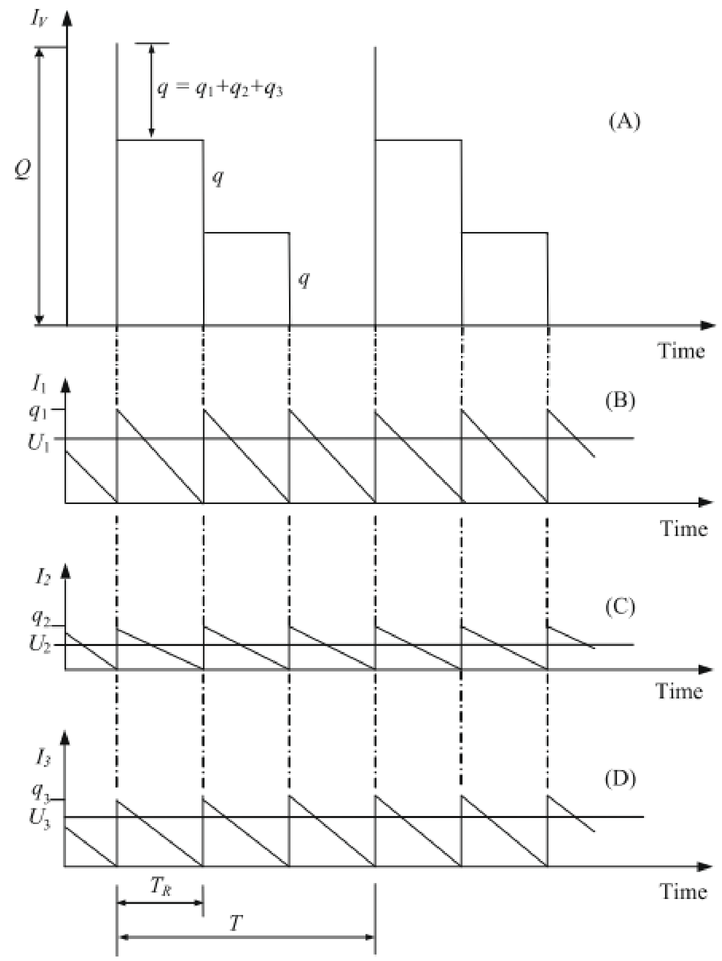


Figure 42: Inventory level for vendor (A) and m = 3 retailers (B, C and D).

[illustration according to Darwish and Odah (2009)]

The relationship between the two different shipment sizes of retailer i (q_i) and retailer j (q_j) is directly influenced by the relationship of their respective demand rates (D_i and D_j) for this specific vendor cycle time ($\frac{q_j}{q_i} = \frac{D_j}{D_i}$). This leads to

$$q_j = \frac{D_j}{D_i} q_i \qquad (1)$$

With (1) the delivery quantity for each retailer (q_j) can be calculated as soon as the delivery quantity for one retailer (q_i) is known (assuming of course that the demand rates are known) which will prove to be a very advantageous characteristic in order to define the VMI model.

The quantity $q = \sum_{j=1}^{m} q_j$, which the vendor dispatches each common cycle length (T_R) to all *m* retailers, can be simplified using (1) to

$$q = \frac{\mathrm{D}}{D_1} q_1 \,, \tag{2}$$

where *D* is the sum of the demands for all *m* retailers $D = \sum_{j=1}^{m} D_j$ per vendor cycle time *T.*

Additionally, the vendor's *production* quantity *Q* that the vendor produces for each *T* time units can be broken down to the sum of the quantities dispatched to all retailers per shipment (*q*) multiplied by the number of shipments (*n*). Using (2) this can further be simplified to

$$Q = \mathrm{nq} = \mathrm{n}\frac{\mathrm{D}}{D_1} q_1 \,, \tag{3}$$

After the first couple of simple algebra operations to get (1), (2) and (3), we now develop the VMI model that seeks to minimize the average total supply chain cost per unit time, or **total cost incurred by system (*TC*)**.

To get the total cost incurred by the system, we can break down the total cost into

1) **cost incurred by the vendor ($\mathbf{TC_V}$)** and
2) **cost incurred by all retailers ($\mathbf{TC_R}$)**, which is the sum of the cost for one retailer ($\mathrm{TC_j}$) over all retailers ($TC_R = \sum_{j=1}^{m} TC_j$)

$$Total\ System\ Cost = Total\ Vendor\ Cost + Total\ Retailers\ Cost$$

$$TC = \mathrm{TC}_V + \mathrm{TC}_R \tag{3}$$

3.2.3.3.1 Total Vendor Cost ($\mathrm{TC_V}$)

The total cost incurred by the vendor consists of

- the **vendor production cost**,
- the sum of **order cost for all retailers**, which the vendor has to pay according to the VMI contract,
- the **inventory cost per cycle** at the vendor's site and
- the **overstock penalty cost** the vendor has to pay for overstocking at the retailers' sites.

The **inventory cost per cycle** can be determined best by looking at the inventory level over time as seen in Figure 43.

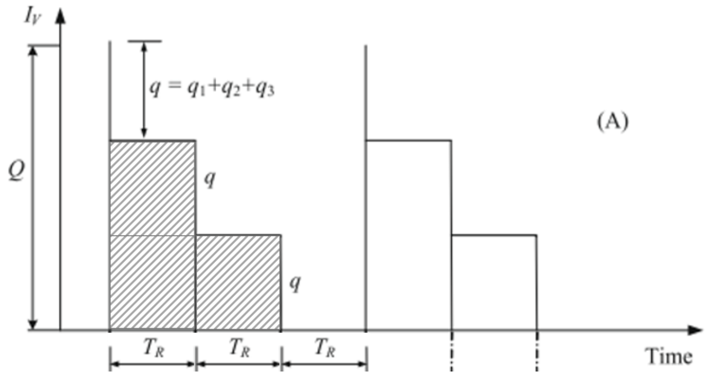


Figure 43: Inventory level over time for vendor

[illustration according to Darwish and Odah (2009)]

As seen in Figure 43, the stock level of the vendor is reduced every T_R period by the quantity *q*. For e.g. *n = 3* shipments, the vendor inventory level consists of 3 rectangles with the length of T_R and the height of *q*. The time period of T_R can further be written as $T_R = \frac{q}{D}$, as it represents the time needed for the quantity *q* to be consumed by the demand rate for the vendor *D*.

The area for one rectangle is hence $\frac{q}{D} * q = \frac{q^2}{D}$ and using (2) this leads to

$$h_V\left[(n-1)\ \frac{q^2}{D} + \ (n-2)\frac{q^2}{D} + \cdots + \ \frac{q^2}{D}\right] = h_V\frac{n(n-1)}{2}\ \frac{q^2}{D} = \ h_V\frac{n(n-1)}{2}\ \frac{1}{D}\ \frac{D^2}{D_1^2}q_1^2$$

$$= h_V\frac{n(n-1)}{2}\frac{Dq_1^2}{D_1^2} \tag{3}$$

as the general term for the **vendor inventory cost per cycle**.

As written before, the vendor additionally incurs **overstock penalty cost** when the stock level of a retailer exceeds its contractually agreed upon upper limit U_j. The overstock penalty cost π_j has to be paid for every item that exceeds the upper limit.

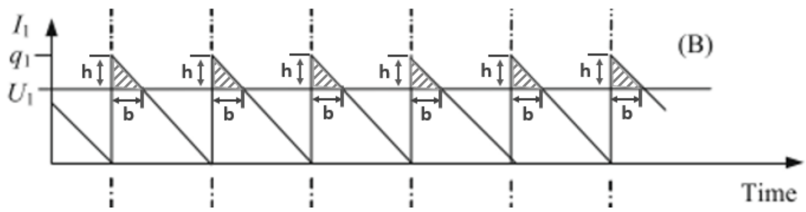


Figure 44: Overstock for retailer j

[illustration according to Darwish and Odah (2009)]

The amount of items that exceed the upper limit is visualized in Figure 44 by triangles with diagonal lines. The easiest way to calculate the area of one of these triangles is by using the formula $A = \frac{1}{2}bh$ with *b* as the length of the base of the triangle, and *h* as the height of the triangle. In our case *h* is simply $(q_j - U_j)$, the amount that is exceeding the upper limit, and *b* is the time it takes to consume the amount $(q_j - U_j)$, therefore *b* can be written as $\frac{(q_j - U_j)}{D_j}$. It follows that the area of one rectangle is $A = \frac{1}{2}\left(q_j \ - \ U_j\right)\frac{(q_j - U_j)}{D_j} = \frac{1}{2}\frac{\left(q_j - U_j\right)^2}{D_j}$.

The overstock penalty cost for one retailer *j* per vendor cycle time *T* is therefore given by $\pi_j n \frac{(q_j - U_j)^2}{2D_j}$, $if\ q_j > U_j$.

Therefore the **total penalty cost** the vendor has to pay for all retailers with overstocked inventory can be written by using (1) as

$$\frac{n}{2}\sum_{j\in S}\pi_j \frac{(q_j - U_j)^2}{D_j} = \frac{n}{2}\sum_{j\in S}\pi_j \frac{(\frac{D_j}{D_1}q_1 - U_j)^2}{D_j} = \frac{n}{2}\sum_{j\in S}\frac{\pi_j}{D_j}(\frac{D_j}{D_1}q_1 - U_j)^2,$$

where *S* is the set of all retailers whose upper inventory limit is exceeded, i.e. who are overstocked.

As mentioned above, the **total cost incurred by the vendor** consists of

- the **vendor production cost** ($\mathbf{A_V}$),
- the sum of **order cost for all retailers**, which the vendor has to pay according to the VMI contract ($\boldsymbol{n\sum_{j=1}^{m} A_j}$),
- the **inventory cost per cycle** at the vendor's site ($\boldsymbol{h_V \frac{n(n-1)}{2}\frac{Dq_1^2}{D_1^2}}$) and
- the **overstock penalty cost** the vendor has to pay for overstocking at the retailers' sites ($\boldsymbol{\frac{n}{2}\sum_{j\in S}\frac{\pi_j}{D_j}(\frac{D_j}{D_1}q_1 - U_j)^2}$).

The **total vendor cost per cycle** therefore can be calculated as

$$\mathrm{TC}_V\,(q_1, n, S) = A_V + n\sum_{j=1}^{m} A_j + h_V\frac{n(n-1)}{2}\frac{Dq_1^2}{D_1^2} + \frac{n}{2}\sum_{j\in S}\frac{\pi_j}{D_j}(\frac{D_j}{D_1}q_1 - U_j)^2.$$

The function TC_V depends on the quantity dispatched to retailer 1 (q_1), the number of shipments (*n*) and the set (*S*) of all retailers whose upper limit is exceeded.

3.2.3.3.2 Total Retailers Cost (TC_R)

The retailers, on the other hand, only incur their own inventory cost. For retailer *j* this simply means holding cost per unit (h_j) multiplied with the average inventory ($\frac{1}{2}q_j$), hence the inventory cost for retailer *j* is $h_j\frac{1}{2}q_j$. Using (1) to get q_1 as the only decision variable, we get $h_j\frac{1}{2}q_j = h_j\frac{1}{2}\frac{D_j}{D_1}q_1 = \frac{q_1}{2D_1}h_jD_j$. Added up over all retailers, we get

$$TC_R = \frac{q_1}{2D_1}\sum_{j=1}^{m} h_jD_j \qquad (4)$$

3.2.3.3.3 Total System Cost (TC)

Looking at the system-wide cost, the total system average cost per time unit t (*TC*) is the sum of the total vendor cost (TC_V) and the total retailers cost (TC_R).

$$TC = \mathrm{TC}_V + \mathrm{TC}_R$$

$$= A_V + n\sum_{j=1}^{m} A_j + h_V \frac{n(n-1)}{2}\frac{Dq_1^2}{D_1^2} + \frac{n}{2}\sum_{j\in S}\frac{\pi_j}{D_j}\left(\frac{D_j}{D_1}q_1 - U_j\right)^2 + \frac{q_1}{2D_1}\sum_{j=1}^{m} h_j D_j$$

This equation can be further transformed to:

$$A_V + n\sum_{j=1}^{m} A_j = \frac{(A_V + n\sum_{j=1}^{m} A_j)D_1}{D_1} = \frac{(A_V + n\sum_{j=1}^{m} A_j)D_1}{nq_1},$$

$$h_V \frac{n(n-1)}{2}\frac{Dq_1^2}{D_1^2} = h_V \frac{(n-1)}{2}\frac{Dq_1 nq_1}{D_1^2} = h_V \frac{(n-1)}{2}\frac{Dq_1 D_1}{D_1^2} = h_V \frac{(n-1)}{2}\frac{Dq_1}{D_1}$$

and $\frac{n}{2}\sum_{j\in S}\frac{\pi_j}{D_j}\left(\frac{D_j}{D_1}q_1 - U_j\right)^2 = \frac{1}{2}\frac{D_1}{q_1}\sum_{j\in S}\frac{\pi_j}{D_j}\left(\frac{D_j}{D_1}q_1 - U_j\right)^2$ in order to get our problem formulation as presented in 3.2.3.4.

3.2.3.4 Model

1. Model:

$$\min TC = \frac{(A_V + n\sum_{j=1}^{m} A_j)D_1}{nq_1} + h_V \frac{(n-1)}{2}\frac{Dq_1}{D_1} + \frac{1}{2}\frac{D_1}{q_1}\sum_{j\in S}\frac{\pi_j}{D_j}\left(\frac{D_j}{D_1}q_1 - U_j\right)^2 + \frac{q_1}{2D_1}\sum_{j=1}^{m} h_j D_j$$

s.t.,

$q_j \geq U_j,\ j \in S$

$q_i \leq U_i,\ i \in \bar{S}$

$\bar{S}$ is the complement of S, which means it's the set of all retailers whose upper limit is not exceeded. As noted above, our cost function *TC* depends on the quantity dispatched to retailer 1 (q_1), the number of shipments (n) and the set (S) of all retailers whose upper limit is exceeded. Because of this, the model as constructed above contains *m* constraints because all retailers (m = number of retailers) have to be checked whether their respective upper limit is exceeded or not.

However, Darwish and Odah (2009) show that there is a way to decrease the number of constraints to two for a given set *S*:

Formula (1) gives $q_j = \frac{D_j}{D_1}q_1$, which means the first set of constraints in our model can be rewritten as $\frac{D_j}{D_1}q_1 \geq U_j,\ j \in S$ or $q_1 \geq \frac{D_1}{D_j}U_j,\ j \in S$. Remembering that *r* is the number of retailers in set *S*, it is obvious that $q_1 \geq \frac{D_1}{D_j}U_j,\ j \in S$ gives *r* constraints but only the constraint with the biggest ratio of $\frac{U_j}{D_j}$ in fact constraints q_1. Hence the constraint $q_j \geq U_j,\ j \in S$ can be rewritten as $q_1 \geq D_1 \max_{j\in S}\frac{U_j}{D_j}$. Analogously the second set of constraints $q_i \leq U_i,\ i \in \bar{S}$ can be rewritten as $q_1 \leq D_1 \min_{i\in \bar{S}}\frac{U_i}{D_i}$. This brings our model to the following.

2. Model:

$$\min TC = \frac{(A_V + n\sum_{j=1}^{m} A_j)D_1}{nq_1} + h_V \frac{(n-1)}{2}\frac{Dq_1}{D_1} + \frac{1}{2}\frac{D_1}{q_1}\sum_{j\in S}\frac{\pi_j}{D_j}\left(\frac{D_j}{D_1}q_1 - U_j\right)^2 + \frac{q_1}{2D_1}\sum_{j=1}^{m} h_j D_j$$

s.t., $q_1 \geq D_1 \max_{j\in S}\frac{U_j}{D_j}$ and $q_1 \leq D_1 \min_{i\in \bar{S}}\frac{U_i}{D_i}$

Because beforehand the model doesn't know which retailers have overstock, the members of the set *S* is unknown at the start. One possible way to solve the model is hence by finding optimal values for the decision variables q_1 and *n* for all possible combinations of the set *S* and then selecting the set *S* with minimum total cost TC. However, the complete enumeration requires solving 2^m problems.

The following reduces the number of sets that have to be explored from 2^m to (*m+1*):

From our current model we have $D_1 \max_{j\in S}\frac{U_j}{D_j} \leq q_1 \leq D_1 \min_{i\in \bar{S}}\frac{U_i}{D_i}$. The likelihood of overstocking is indicated by the ratio of the retailers' respective upper limits (U_j) to their respective demands (D_j). We can sort the retailers in ascending order according to their likelihood of overstocking i.e. putting the retailer with the highest likelihood of overstocking (lowest ratio of $\frac{U_j}{D_j}$) to position 1 and the retailer with the lowest likelihood of overstocking (highest ratio of $\frac{U_j}{D_j}$) at position m and so on. This leads to the ratios of the newly sorted retailers being: $\frac{U_1}{D_1} \leq \frac{U_2}{D_2} \leq \cdots \leq \frac{U_{i-1}}{D_{i-1}} \leq \frac{U_i}{D_i} \leq \cdots \leq \frac{U_r}{D_r} \leq \frac{U_{r+1}}{D_{r+1}} \leq \cdots \leq \frac{U_m}{D_m}$, which means that if retailer *i* is in the Set *S* (i.e. is overstocked), retailer *i-1* has to be in *S* too. If retailers are arranged in ascending order, the feasible set for *S* can only be: $\{0\},\{1\},\{1,2\},\{1,2,3\},\ldots,\{1,2,3,\ldots,m\}$, where *S* = $\{0\}$ represents the situation where no retailer gets overstocked and *S* = $\{1,2,3,\ldots,m\}$ represents the situation where all retailers gets overstocked.

If we look at the sorting $(\frac{U_1}{D_1} \leq \frac{U_2}{D_2} \leq \cdots \leq \frac{U_{i-1}}{D_{i-1}} \leq \frac{U_i}{D_i} \leq \cdots \leq \frac{U_r}{D_r} \leq \frac{U_{r+1}}{D_{r+1}} \leq \cdots \leq \frac{U_m}{D_m})$ again and remember that *r* is the number of elements in the set *S*, we can see that $\max_{j\in S}\frac{U_j}{D_j} = \frac{U_r}{D_r}$ and $\min_{i\in \bar{S}}\frac{U_i}{D_i} = \frac{U_{r+1}}{D_{r+1}}$. Therefore the constraints of our second model become $D_1\frac{U_r}{D_r} \leq q_1 \leq D_1\frac{U_{r+1}}{D_{r+1}}$ and our model can be simplified to:

Final Model:

$$\min TC = \frac{(A_V + n\sum_{j=1}^{m} A_j)D_1}{nq_1} + h_V \frac{(n-1)}{2}\frac{Dq_1}{D_1} + \frac{1}{2}\frac{D_1}{q_1}\sum_{j\in S}\frac{\pi_j}{D_j}\left(\frac{D_j}{D_1}q_1 - U_j\right)^2 + \frac{q_1}{2D_1}\sum_{j=1}^{m} h_j D_j$$

s.t., $D_1\frac{U_r}{D_r} \leq q_1 \leq D_1\frac{U_{r+1}}{D_{r+1}}$

where *r* is the number of retailers in Set S.

Looking at our cost function we can see that the costs involved in our model include the ordering and holding cost for both the vendor and the retailers as well as the overstocking penalty cost incurred by the vendor. The decision variables are q_1, n and S and our model is restricted by two constraints.

3.2.3.5 Solution algorithm

To solve our model, we have to find all Karush-Kuhn-Tucker (KKT) points associated with the cost function and then calculate the respective costs for each KKT point. The KKT point with minimum cost is our optimal solution. This approach works for our model because if it is possible to find all KKT points (which it is in our case) for a given nonlinear problem, the global optimal solution is one of these KKT points. The KKT points for our model can be found using the following theorem, which M.A. Darwish and O.M. Odah introduced in their paper “Vendor managed inventory model for single-vendor multi retailer supply chains”.

Theorem 1[25]. The KKT points associated with our model are as follows

(A) $n = \sqrt{\frac{2A_V\left[\sum_{j=1}^{m} h_j D_j + \sum_{j\in S}\pi_j D_j - D h_V\right]}{h_V D\left(2\sum_{j=1}^{m} A_j + \sum_{j\in S}\frac{\pi_j}{D_j}U_j^2\right)}}$ and $q_1(n) = D_1\alpha(n)$, if $\frac{U_r^2}{D_r^2} \leq \alpha(n) \leq \frac{U_{r+1}^2}{D_{r+1}^2}$, where

$\alpha(n) = \frac{\frac{2}{n}\left(A_V + n\sum_{j=1}^{m} A_j\right) + \sum_{j\in S}\frac{\pi_j}{D_j}U_j^2}{\sum_{j=1}^{m} h_j D_j + \sum_{j\in S}\pi_j D_j + D h_V(n-1)}$.

(B) $n = \frac{D_r}{U_r}\sqrt{\frac{2A_V}{D h_V}}$, and $q_1 = D_1\frac{U_r}{D_r}$, if $\alpha(n) \leq \frac{U_r^2}{D_r^2}$.

(C) $n = \frac{D_{r+1}}{U_{r+1}}\sqrt{\frac{2A_V}{D h_V}}$, and $q_1 = D_1\frac{U_{r+1}}{D_{r+1}}$, if $\alpha(n) \geq \frac{U_{r+1}^2}{D_{r+1}^2}$.

The following algorithm finds the global optimal solution based on Theorem 1.

Algorithm 2 VMI

Procedure

Sort retailers in an ascending order such that $\frac{U_1}{D_1} \leq \frac{U_2}{D_2} \leq \cdots \leq \frac{U_m}{D_m}$.

Set $k = 0$, $S = \Phi$, $U_0 = 0$, and $U_{m+1} \rightarrow \infty$.

Let $TC_k^A, TC_k^B, TC_k^C \rightarrow \infty$.

(i) Use part A of Theorem 1 to determine n, let $n_1^A = \lfloor n \rfloor$ and $n_2^A = \lceil n \rceil$ and compute $\alpha(n_1^A)$ and $\alpha(n_2^A)$.

(ii) If $\frac{U_k^2}{D_k^2} \leq \alpha(n_1^A) \leq \frac{U_{k+1}^2}{D_{k+1}^2}$, compute $TC(q_1(n_1^A), n_1^A, S)$.

(iii) If $\frac{U_k^2}{D_k^2} \leq \alpha(n_2^A) \leq \frac{U_{k+1}^2}{D_{k+1}^2}$, compute $TC(q_1(n_2^A), n_2^A, S)$.

(iv) If $TC(q_1(n_1^A), n_1^A, S) \leq TC(q_1(n_2^A), n_2^A, S)$, then $TC_k^A = TC(q_1(n_1^A), n_1^A, S)$.

(v) If $TC(q_1(n_1^A), n_1^A, S) > TC(q_1(n_2^A), n_2^A, S)$, then $TC_k^A = TC(q_1(n_1^A), n_1^A, S)$.

(i) Use part B of Theorem 1 to determine n, let $n_1^B = \lfloor n \rfloor$ and $n_2^B = \lceil n \rceil$ and compute $\alpha(n_1^B)$ and $\alpha(n_2^B)$.

(ii) If $\alpha(n_1^B) \leq \frac{U_k^2}{D_k^2}$, compute $TC(q_1(n_1^B), n_1^B, S)$.

[25] See Darwish and Odah, 2009, p. 482 for derivation.

(iii) If $\alpha(n_2^B) \le \frac{U_k^2}{D_k^2}$, compute *TC*($q_1(n_2^B)$, n_2^B, *S*).

(iv) If *TC*($q_1(n_1^B)$, n_1^B, *S*) ≤ *TC*($q_1(n_2^B)$, n_2^B, *S*), then TC_k^B = *TC*($q_1(n_1^B)$, n_1^B, *S*).

(v) If *TC*($q_1(n_1^A)$, n_1^A, *S*) > *TC*($q_1(n_2^B)$, n_2^B, *S*), then TC_k^B = *TC*($q_1(n_1^B)$, n_1^B, *S*).

7:

(i) Use part C of Theorem 1 to determine n, let $n_1^C = \lfloor n \rfloor$ and $n_2^C = \lceil n \rceil$ and compute $\alpha(n_1^C)$ and $\alpha(n_2^C)$.

(ii) If $\frac{U_{k+1}^2}{D_{k+1}^2} \le \alpha(n_1^C)$, compute *TC*($q_1(n_1^C)$, n_1^C, *S*).

(iii) If $\frac{U_{k+1}^2}{D_{k+1}^2} \le \alpha(n_2^C)$, compute *TC*($q_1(n_2^C)$, n_2^C, *S*).

(iv) If *TC*($q_1(n_1^C)$, n_1^C, *S*) ≤ *TC*($q_1(n_2^C)$, n_2^C, *S*), then TC_k^C = *TC*($q_1(n_1^C)$, n_1^C, *S*).

(v) If *TC*($q_1(n_1^C)$, n_1^C, *S*) > *TC*($q_1(n_2^C)$, n_2^C, *S*), then TC_k^C = *TC*($q_1(n_1^C)$, n_1^C, *S*).

8: Set *k* = *k+1*.

9: If *k* ≤ *m*, add retailer *k* to *S* and go to Step 4.

10: The optimal solution corresponds to min (TC_k^A, TC_k^B, TC_k^C), *k = 0, 1, …, m*.

The above algorithm gives us the optimal solution for our model. We start with sorting the retailers in an ascending order regarding their ratio of $\frac{U_i}{D_i}$, which represents their likelihood of overstocking (the smaller the ratio, the greater the likelihood of overstocking). In Step 3 & 4 the parameters get initialized. Steps 5, 6 and 7 use parts A, B and C of theorem 1 respectively to calculate the values for *n* (frequency of shipments), $q_1(n)$ (quantity replenished to retailer 1) and *TC* (total system cost). More specifically (i) finds the values for *n* and the rounded down ($\lfloor n \rfloor$) and rounded up ($\lceil n \rceil$) integer values that surround n. Steps (ii) & (iii) check the corresponding feasibility condition and find the total system cost *TC* for both integers surrounding *n*. In (iv) & (v) *TC* for both integers are compared and the solution with less total cost is selected. Step 8 & 9 add the next retailer to set *S* and loops the algorithm back to calculate *n* and *TC(n)* for the new set. This happens until all retailers are part of set *S* when in step (10) the minimal total cost is determined, which corresponds to the optimal solution.

The optimal solution thus determines *n* (frequency of shipments), $q_1(n)$ (quantity replenished to retailer 1) and *S* (set of overstocked retailers).

Using (1), $q_1(n)$ can then be used to determine the respective quantities dispatched to all the other retailers (q_j). Other relevant values can also be determined using previously found, simple formulas. For example, the needed production lot *Q* for the vendor can be calculated using Q=*nq* with $q = \sum_{j=1}^{m} q_j$, the vendor cycle time *T* can be calculated using $T = \frac{Q}{D}$ and the common cycle time for retailers T_R can be calculated by $T_R = \frac{T}{n}$ oder $T_R = \frac{q_j}{D_j}$.

3.2.3.6 Implementation

Some of the values used in our model are obviously very sensitive and companies are hesitant to share them openly with other companies. To accommodate the sensitive nature of some of the shared parameters, we suggest an implementation within a secure cloud environment which serves as a way to protect the privacy of the sensitive parameters from other actors of the supply chain. Implementing the presented model into a cloud environment can be done following the implementation process shown in Figure 45.

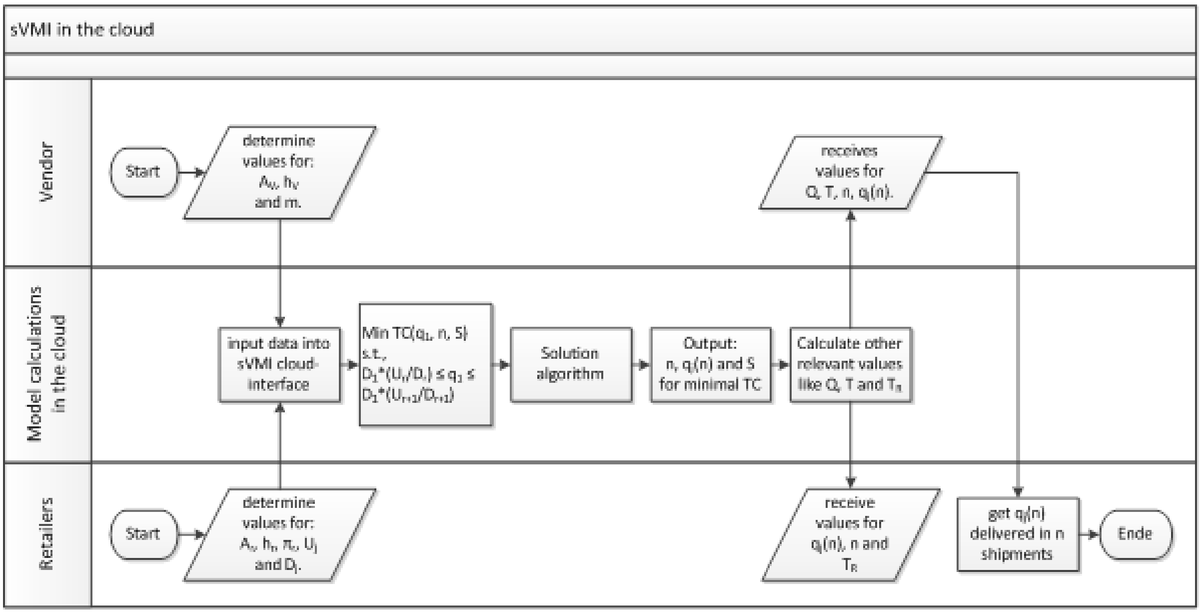


Figure 45: Implementation process.

The sensitive input parameters can be directly used in the cloud to solve the suggested model without disclosing them to the other actors. Instead, the supply chain actors only receive the calculated output values that are relevant to them.

3.2.3.7 Data sensitivity assessment

Before such an implementation can take place however, it is important to analyse which variables need to be protected and why.

Vendor	**Input data**	
Variable	**Description**	**Security requirements**
m	Number of retailers	Low security requirements – number of retailers is most likely known (at least approximately) throughout the supply chain and doesn't need to be protected
A_V	Vendor production cost for one production lot	High security requirements – production cost is private data of the vendor and needs to be protected
h_V	Inventory holding cost for vendor	High security requirements – holding cost is private data of the vendor and needs to be protected

Table 35: Sensitivity analysis of vendor inputs.

Retailers	**Input data**	
Variable	**Description**	**Security requirements**
h_j	Inventory holding cost for retailer *j*	High security requirements – holding cost is private data of the retailers and needs to be protected against vendor and other retailers
D_j	Demand rate for retailer *j*	Medium security requirements – demand rate may be openly shared with the vendor but needs to be protected against other retailers
U_j	Upper limit on the inventory level for retailer *j*	Medium security requirements – upper inventory limit may be openly shared with the vendor but needs to be protected against other retailers
A_j	Order cost for retailer *j*	Medium security requirements – order cost may be openly shared with the vendor (as the vendor has to pay it anyway) but needs to be protected against other retailers
π_j	Overstock penalty cost for retailer *j*	Medium security requirements – overstock penalty cost may be openly shared with the vendor (as the vendor has to pay it anyway) but needs to be protected against other retailers

Table 36: Sensitivity analysis of retailers input.

Variables that are calculated within the model like *D, Q* or *q* are not included in the input tables as they are not "real" input data for the model but rather calculated aggregates within the model calculation processes.

	Output data	
Variable	**Description**	**Security requirements**
q_j	Quantity dispatched to retailer *j*	Medium security requirements – quantity dispatched to retailer *j* may be openly shared between the vendor and the respective *j*th retailer but needs to be protected against other retailers
n	Number of shipments	Low security requirements – number of shipments is known throughout the supply chain (as it is the same for every retailer) and doesn't need to be protected
S	Set of retailers whose upper limit is exceeded	High security requirements – set of retailers whose upper limit is exceeded is sensitive information which could lead to conclusions about upper inventory limits and/or holding costs of other retailers and hence needs to be protected

Table 37: Sensitivity analysis of variables calculated through inputs.

3.2.3.8 Conclusion and further references

In the aforementioned chapters, we presented a Vendor Managed Inventory model for a single-vendor multi-retailer supply chain that can be securely implemented within the cloud. The model takes input data with varying degrees of confidentiality requirements from both vendor and multiple retailers and determines a replenishment policy that minimizes the total system cost throughout the whole supply chain. This specific model setting provides the vendor with relevant data as model output (like replenishment quantities to all retailers (q_j) and shipment frequency (n)) without giving the vendor insight into sensitive input data of its retailers (like holding cost (h_j)). Our model specifically considers a mutual VMI agreement regarding upper stock levels to accommodate for the retailers' loss of control in VMI settings and counters this by establishing specific inventory bounds as well as penalties when exceeded.

VMI is a relatively new concept that began reaching prominence in the late 1990s. For a long time, VMI literature tended to focus on the possible benefits of VMI and the differences between VMI and traditional replenishment setups (e.g. Lee et al. (1997), Disney and Towill (2003). Even when quantitative models were developed, they were oftentimes used to show the benefits of VMI over the traditional replenishing process (Yao (2005), Rad et al. (2014), Mateen and Chatterjee (2015) etc.) rather than actual quantitative models that can be used in VMI settings to operate the replenishment process. By and large, those kind of quantitative models are discussed a lot less in VMI research.

One important step in this stream of research is the work by Banerjee and Banerjee (1994) who developed an analytical model based on a common cycle time approach for a one vendor multiple retailers VMI-like situation without using the actual term Vendor Managed Inventory. The model, however, incorporates not only stochastic demands but also stochastic lead times. For our case, stochastic lead times are not necessary to incorporate into the model as the variance of lead times should be negligible.

Cetinkaya and Lee (2000) developed an analytical model based on renewal theory for coordinating inventory and transportation decisions in VMI systems. The model considers a vendor realizing a sequence of random demands from a group of retailers located in a given geographical region. However, inventory bounds are not reflected in the model and they considered only one inventory warehouse for the whole system.

Zavanella and Zanoni (2008) present a model for a particular VMI policy, known as Consignment Stock, but only consider the case where the holding cost of retailers are lower than the holding cost of the vendor, which we deem unrealistic for ARC.

Hariga et al. (2014) develop a more complex model related to the model we use. The main difference to our model is that Hariga et al. allow for unequal shipment frequencies for the different retailers and their greater focus on storage constraint which we deem unnecessary.

As noted above, our model is based on the work by Darwish and Odah (2009) as their quantitative model is easy to comprehend and the assumptions taken seem to be reasonable for our case. Even though unequal shipment frequencies might minimize the cost even further, it also leads to higher transportation cost (which is only implicitly included in our model and the model developed by Hariga et al. via retailer's ordering costs). For ARC however, equal shipment frequencies mean lower transportation cost as shipments to more retailers can be combined and in general lower planning complexities as e.g. outbound logistic activities can be combined for equal cycle times. Additionally the assumption of deterministic demand may not be an ideal reflection of reality but it enables us to model the basic interactions regarding the replenishment policy without the model getting too complex.

3.2.4 Summary

In the preceding subchapters of Section 0 we describe two approaches to improve the performance of the ARC consumer goods supply chain. First, we propose a rather simple secure aggregation protocol to obtain aggregated forecasts for medium-term demand for each product as depicted in the flowchart of Figure 46. This provides ARC order management with more accurate information for their capacity reservations. Second, we develop a vendor managed inventory and propose a secure implementation in a cloud based planning system to encourage retailer participation. Both methods aim to reduce excessive inventories which currently are a main cost driver for ARC. The main features of each concept are listed in Table 38.

Approach	Time horizon	Computational complexity	Security requirements
Secure aggregation protocol	Medium-term; New run every week	Low – only simple arithmetic operations. The focus is on the secureness of the protocol	High – sensitive customer data is used.
sVMI	Short-term; Frequent updates required (e.g. daily)	High – each run requires numerous operations including non-linear functions	High – sensitive input data from all parties -retailers and ARC – required.

Table 38: Main features of secure aggregation and sVMI

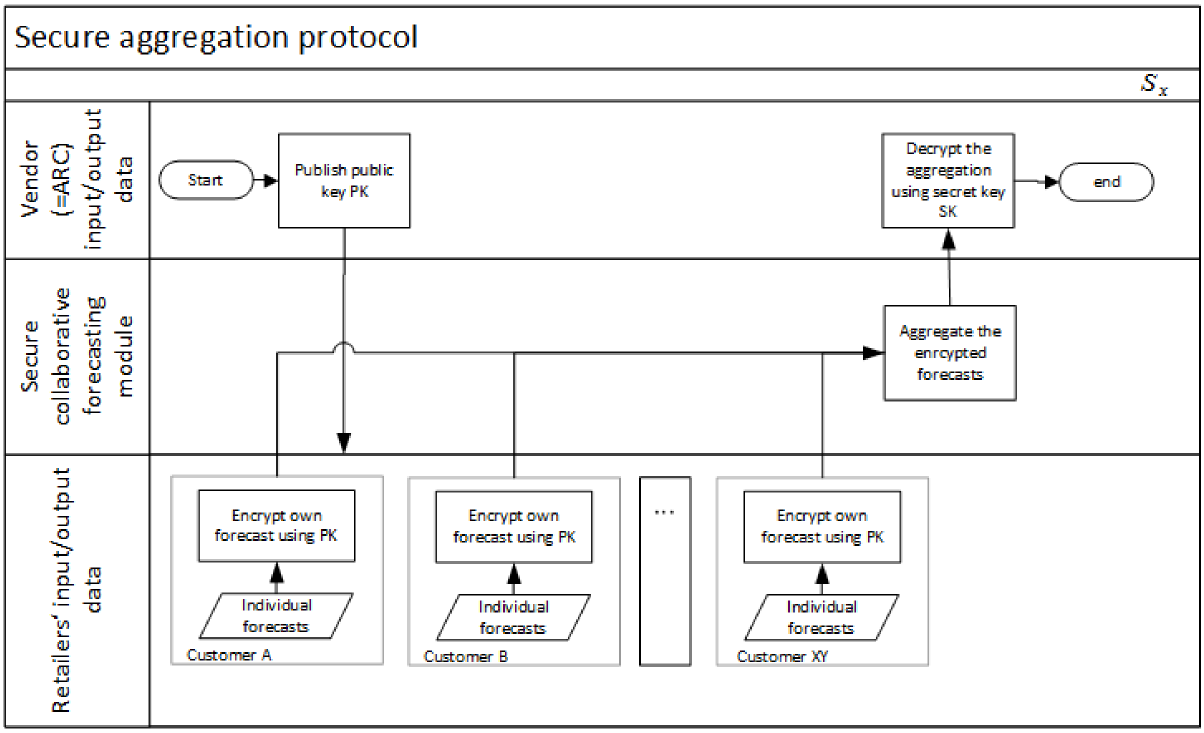


Figure 46: Secure aggregation protocol.

We propose a two-step approach for the implementation: Since the collaborative forecasting via a secure aggregation protocol requires little adaption in the business processes and can be easily understood and therefore accepted due to low computational complexity this is the first step of implementation. The concept of vendor managed inventory requires extensive changes in various processes which makes it more adequate offer this concept in a second step as intensified collaboration. Figure 47 gives an overview on the steps in the VMI setting to obtain the quantities and number of shipments to each retailer for only one product.

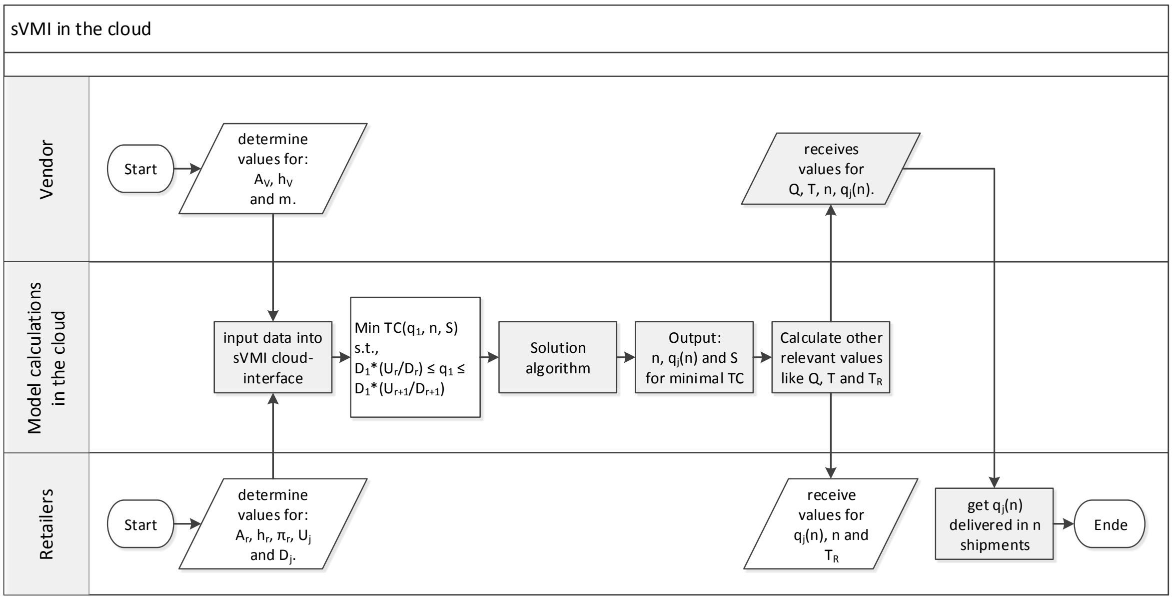


Figure 47: Computational process for one product in VMI setting

The details are given in the algorithm in subchapter 3.2.2.

3.3 Auctions and supply chain coordination[26]

In this section we briefly describe how auctions can be used to coordinate activities in supply chains. Potential application of auctions in the consumer good case is briefly discussed in this work package. There are, however, no plans and allocation of resources for implementing SMC based auctions in PRACTICE at this stage. We suggest, however, how emerging auction designs may improve more advanced supply chain coordination e.g. in combinations with analytics like the other tools presented in this chapter.

Unlike a traditional market, an auction is a set of trading rules (or a protocol) that dictate how to select the winners (one or more) of an auction and what the winners buy or sell - essentially a pure market place at work in its finest form. In fact, this high level of control makes it possible to solve certain market failures by designing auction rules that improve the coordination in a competitive environment. The improved coordination may come from allowing new players to enter the market or by ensuring a better coordination across related goods and between price and other attributes.

The auctioneer that manages an auction ensures that everyone follows the protocol and shares information as prescribed. In theory and practice it has been shown that SMC is an ideal replacement of the auctioneer (see e.g. Malkhi et al 2004 and Bogetoft et al 2009). SMC may be a cost efficient way to improve integrity and confidentiality by not granting any single person or institution the possibility to manipulate the protocol or to tap confidential

[26] Main author: Kurt Nielsen (PAR)

information from the protocol. In fact, auctions have been the first commercial application of SMC (see partisia.com).

Advanced auctions involve more and more confidential information from the participants in order to facilitate more coordination. One of the main challenges in these auctions is to make it as simple as possible to participate without compromising the intended goal i.e. to maximize profit or social welfare. On one side, simplifying the bidding is all about reducing the effort required to bid optimally (address cognitive overload) and on the other side, to reduce information required to bid optimally (address informational overload). As discussed later, an interesting class of auctions is one where the optimal bidding strategy is to tell the truth. Such auctions have a great potential interplay with other information systems – systems that basically try to picture the true state of nature - and therefore may be used directly to further simplify the bidding. The vision is that truth-telling mechanism address cognitive overload and other information systems address informational overload. Ideally, such mechanisms may be used in automated processes and potentially harvesting the many small gains in a “big data” reality. Equally important, those type of integrated auction systems emphasize the importance of integrity and confidentiality i.e. the properties delivered by SMC.

The section is organized as follows: Section 3.3.1 introduces briefly auctions in theory and practice and Section 3.3.2 elaborates on the class of truth-telling auctions and the role of SMC. Selected advanced auctions are discussed in Sections 3.3.33.3.4 and 3.3.4 with potential applications in the consumer goods industry.

3.3.1 Auctions in theory and practice

Auctions are an ancient institution and history provides evidence as far back as around 500 BC, where auctions were used for selling marriageable women in Babylon and slaves by the Romans. In China auctions were recorded in the 7'th century for selling the belongings of deceased monks. Today auctions are used for selling anything from low value items like fish and flowers to highly valued items like fine art, corporations, drilling rights, spectrum rights, treasury bills, other financial products and much more.

Unlike the employment of auctions, the theory of auctions is new and has developed along with the discipline of information economics. Especially the interpretation of auctions as a game of imperfect information has been at the core of auction theory. There are numerous types of auctions addressing numerous coordination problems. Since the seminal paper Vickrey (1961) and especially since the early 1980's, the literature on auction theory has grown considerably. The greater part of the literature considers the simple one-sided auctions, where a seller (monopolist) chooses the auction rules to maximize expected revenue. The seller basically uses an auction to solve the problem of private information. The seller is uncertain about the values that the bidders attach to the item being sold - the maximum amount each bidder is willing to pay. This uncertainty is an inherent part of auctions. Otherwise, if no uncertainty existed, the seller could just set the price and sell to the bidder with the highest valuation.

This fairly simple case has become the benchmark case in the theory on auctions. In Vickrey's famous paper from 1961, he compared among others the so-called first and second price auctions and finds that they yield the seller the same revenue. This so-called revenue equivalence theorem remained a puzzle until 1981 where Riley and Samuelson (1981) and Myerson (1981) simultaneously proofed the theorem.

In the first price sealed bid auction, the bidders submit a sealed bid and the bidder with the highest bid gets the item at the highest bid.

In the Second price sealed bid auctions, the bidders submit a sealed bid and the bidder with the highest bid gets the item at the second highest bid. This auction was designed by Vickrey himself and is also known as the Vickrey auction.

Although the revenue equivalence theorem may suggest the opposite, auction design does matter a great deal. The revenue equivalence theorem relies on assumptions that are seldom fulfilled in real markets. Things like common value, signals about the value, risk attitude, number of participants, number of units for sale, a.o. do all have an influence on the design of auctions. Also the large number of different auctions and market mechanisms indicate that auction design matters. The different auctions can be divided into several groups by their institutional setting and market characteristics e.g. private value, common value, single unit, multiple units, single product, multi product, discriminatory, uniform, one-sided or two-sided and single and multi-attribute auctions. For a textbook introduction to auction theory see e.g. Krishna (2002) or Klemperer (2004).

3.3.2 Truth telling and Secure Multiparty Computation

The generalization of the second price auction is known as the Vickrey, Clarke Growes mechanism (the VCG mechanism) and uses a similar mechanism to motivate truth-telling. In a second price auction, the winner pays the second highest price-bid, which would have been the value generated if the winner did not participate. Likewise, in the generalized VCG mechanism, the winner pays the valuation that would have been generated if the winner did not participate in the auction. The basic VCG mechanism runs in two steps: In step one the auctioneer computes the first best allocation (given truth-telling) and in step two the winner (one or more) is compensated to tell the truth in the first place - according to the second price principle. In practice, the VCG mechanism is mostly known from high stake auctions such as auctions for selling spectrum rights or high valued energy contracts, see e.g. Milgrom (2004). However, it is also used for low valued trades, most prominent in the sponsored link auctions used by google and yahoo, which is approximately a VCG mechanism, Varian (2007).

As mentioned, SMC may replace the auctioneer as trustee or trusted third party (TTP) to confidentially coordinate private information according to the auction rules. The notion of a TTP has been a common construct in information economics since the original work on The Revelation Principle: see, e.g., Gibbard (1973) or Myerson (1979). Based on the revelation principle, any equilibrium outcome that is based on any mechanism can also be arranged as the outcome of a direct revelation game in which the participants have an incentive to honestly reveal their private information – as in a VCG mechanism.

The SMC approach is analogous to paying a consultancy house to act as a single TTP; however, there are some fundamental differences between the two ways of coordinating confidential information:

- With the SMC approach, no single person or institution gains access to confidential information. This is very different from a consultancy house, wherein confidential information is revealed to some trusted person, which opens up room for human errors and bribes.
- With SMC, trust is based on the intentions of the TTPs. Because collusion requires coordination between more TTPs, selecting TTPs with opposing interests strengthens trust. In contrast, when a consultancy house is used, opposing interests may lead to higher stakes and therefore higher payments due to possible bribes.
- With SMC, coordination is represented by a piece of software. Therefore, the marginal cost of repeating trusted coordination is lower than that achieved in a traditional scenario.

Because SMC limits the dissemination of private information to an absolute minimum, political questions regarding whom to trust are greatly simplified. This issue was acknowledged by the initial commercial application of SMC auctions used for reallocating sugar beet contracts in: “The secure approach contributed positively to our decision to deploy the contract exchange. Afterwards, it was of a greater importance for the growers than I initially thought it would be.” (Klaus Sørensen, Danish Sugar Beet Growers’ Organization)

3.3.3 Coordination of price and quantity

Consider the situation with N bidders, where each bidder wishes to buy only a single unit, and M units are offered for sale, M<N. In this set-up a multiple price (discriminatory) auction is analogous to a first price auction, where the M highest bidders are awarded a unit at their individual bids. The analogy to a second price auction is a single price or uniform auction, where all winning bidders gets to pay the highest bid of the rejected N-M bids, Chari and Weber (1992). Harris and Raviv (1981) found that many of the results from the analysis of the single unit auction could be generalized quite easily to this situation where the bidders are limited to demanding only a single unit.

However, in an auction where more units of a good is for sale, the bidders typically demand/supply more units, in such cases bidders' demand/supply is expressed as schemes showing the number of units they want to buy/sell at various prices.

Multi-unit auctions have been extensively discussed in the literature about treasury auctions, where the governments sell treasury bonds. There has been a wide theoretical discussion of whether the auction format should be a discriminating auction or a uniform auction. The literature tends to favour the uniform price auction. However, when market power is introduced, multi-unit uniform price auctions do not have the same truth revealing properties as single-unit second price auctions. In a uniform price auction, the bidders can use their market power to reduce the price on the auction by reducing their demand, for more see Binmore (1992). On the other hand, Chari and Weber (1992) argue that the problem with the discriminatory auction is that the buyers acquire more information than is socially optimal. The bidders may gain by shading the bid below the reservation value. This shading requires considerable information in order to gain on the trade-off between lowering a winning bid against the risk of not winning. Therefore they argue that the uniform auction yields more revenue to the Treasury.

A related type of auction is the multi-unit Double Auctions (or two-sided auctions), where sellers and buyers reallocate multiple units of a single product or service. These auctions are sometimes called exchanges. Some of the most important real world markets are double auctions, e.g. power or stock exchanges.

The literature on double auctions focuses in particular on three problems: 1) incentive compatibility (i.e. truth-telling must be an optimal bidding strategy), 2) ex post efficiency (i.e. the realization of all trades that improve social welfare) and 3) budget balancing (i.e. aggregated value sold must equal aggregated value bought). The two first problems follow from the so-called Myerson-Satterthwaite theorem, Myerson (1983). It says that delays and failures are inevitable in private bargaining if the goods start out in the wrong hands. This follows from the central observation that in any two-persons bargaining game the seller has incentives to exaggerate the value and the buyer has incentives to pretend the value is low.

There have been a few attempts to design truth-telling double auctions, see McAfee (1992) and Yoon (2001). Attempts to solve the first two problems are typically at the cost of the third problem of balancing the budget. Fortunately, the magnitude of the three problems diminishes rapidly as the number of participants grows. Test auctions with as few as 2-3 buyers and 2-3 sellers have generated almost efficient outcomes see e.g. Friedman et al

(1995) and Satterthwaite et al (1989) show analytically that a double auction modelled as a Bayesian game, converges rapidly towards ex post efficiency as the market grows.

Multiple Goods

So far, we have only considered a single divisible good. However, typically coordination across multiple goods is required and challenging for both buyers and sellers that may end up with undesired bundles of goods or be exposed to extreme prices. Advance market design that allows buyers and/or sellers to bid on their desired bundles of commodities may solve these expositor problems.

The core problem is interrelated valuations of the different goods, which add a new layer of complexity to the design of auctions. These issues have been widely discussed in the literature on high stake auctions used for trading e.g. spectrum rights.

The problem with spectrum auction is the general problem of trading heterogeneous goods, that the goods may be either substitutes or complements. That is, the value of a set of goods does not necessarily equal the aggregated value, it could be more (complements) or less (substitutes). The first spectrum auction was held in New Zealand in 1990. This auction is also one of the big failures in auction design. A set of licenses was auctioned simultaneously in second price sealed bid auctions. Such an auction would only be optimal if the different auctions were independent of each other. In case of any complementary or substitutional effects among the different licenses (which there clearly were) the optimal bid will inevitably involve guessing. The catastrophic result of this auction illustrated the problems with this auction. As an example, for one of the licenses the highest bid was NZ $ 7 million and the second highest was NZ $ 5000. Also, the actual revenue of the auction was NZ $ 36 million, which is much smaller than the expected revenue of NZ $ 225 million, see Milgrom (2004).

The first auction designed to handle this situation was the so-called simultaneous ascending auction, which has been used for selling radio frequencies and telecom licenses for more than $100 billion in total. In a simultaneous ascending auction, the bidding occurs in rounds. At each round, the bidders simultaneously submit sealed bids for any items in which they are interested. After each round the result is posted - new bids, standing bids and the corresponding bidders. Also, in each round a minimum bid is calculated for each license. Different versions of this auction have different additional rules. The simultaneous ascending auction is one attempt to handle heterogeneous goods at an auction. This auction deals effectively with substitutes but works significantly worse in the general case where the goods are both substitutes and complements, see Milgrom (2004). The so-called combinatory auction is another attempt at solve the trade of heterogeneous goods. In a combinatory auction the bidder can submit bids on combinations of the items for sale. Most of the theoretical combinatory auctions are good at handling complements, but as with the simultaneous ascending auction the problem arises when we have both complements and substitutes in the same. A recent textbook introduction to combinatory is Cramton et al (2006).

Case 1: Auction market for reallocation of stocks.

One applications of auction that has been discussed in WP24, is the idea of introducing a new auction market for reallocation of stocks among the retailers. In the case of Arcelic or similar firm, such a market may gain insight into the value of the different commodities and stock levels. Reallocation of stocks may also reduce waste in the retail sector and correct for misjudgements of sale at the individual retailer.

The design of the auction market will depend on the expected quantity of different types of commodities. With larger quantities of some commodity (e.g. at standard refrigerator) a separate double auction might be optimal. With larger numbers of different but related commodities, an interrelated auction market is relevant. With smaller quantities of multiple products, a market solution with a large number of parallel one-sided single- and multi-unit auctions might be the most optimal choice.

In either case, developing auctions that promote truth telling and thereby make it optimal for the retailers to express their true demand and supply of different commodities, will add additional value by providing information that can improve the upstream producers' market predictions.

In conclusion, this use of auctions aim at correcting misallocations and by doing that, it may produce information that can be used to better plan production and sale in the first place. As such, it may be seen as a complementary service to the prediction systems presented previously in this Chapter.

Case 2: Auctioning discount

Another application of auction that has been discussed in WP24, is the idea of using the discount to retailers to improve the quality of preorders and to use auctions to find the competitive discount.

The discount is inversely proportional to the distance between preorders and actual orders. If a retailer buys almost the same quantity as stated in the preorders, the highest discount is granted. On the other side, if the retailer's actual orders deviate significantly from the preorders, a lower discount is granted.

The elicitation of preorders by discount may be facilitated in various ways. One approach may be to offer a menu of contracts to the retailers that deviate with respect to cost of deviation. Auctions may be used directly to settle the price in such contracts or to allocate such contracts.

3.3.4 Coordination of prices and other attributes

It is essential for private and public organizations alike to find and select the right commodity or service at the most competitive price. If the desired commodity is well specified, a simple procurement auction (facilitating price and/or quantity competition) is probably the most efficient approach. Often, however, it is difficult to describe all the desired commodity or service characteristics before the auctioning, and a more flexible approach is therefore required.

A traditional negotiation approach allows full flexibility in this two-sided matching of buyers and sellers, but it is typically ill structured and opaque. Multi-attribute auctions, on the other hand, specify a priori transparent rules for the procurement "game" but obviously allow for less flexibility. In either case the transaction costs are usually high. In a traditional negotiation it is time consuming to ensure competition across many sellers and in the traditional multi-attribute auction with a scoring function, it is time consuming to decide on a weighting of price and other attributes a priori.

If you have a scoring function any multi-dimensional bids can be mapped into a one-dimensional score and the traditional first and second price (but now score) auction mechanisms can be used, see Che (1993). However, it is well recognized that preference elicitation in e.g. multi-attribute auctions is costly (or impossible) to establish and requires substantial decision support or mechanism design that mitigate these costs, see e.g. Yang et al (2014), Parkes (2005) and Parkes and Kalagnanam (2005). For instance, when the buyer is a single person and the scoring represents the buyer's intra-personal trade-offs between price and quality; these complications are a central topic of a large literature on Multiple Criteria Decision Making (MCDM), see e.g. Tzeng and Huang (2011). Furthermore, when the buyer represents a group of persons (e.g., an organization), the construction of the scoring function may further involve interpersonal conflicts. The complication of this is reflected on the large literature on Social Choice, see e.g. Arrow (1963) and Moulin (1991).

In the literature on multi-attribute auctions there are only a few papers relaxing the assumption of an a priori given value function for the principal. Cripps and Ireland (1994) investigate the issues of setting quality thresholds that are unknown to the bidders. Beil and Wein (2003) study the sequential learning of the value function and bidders' cost functions by a sequence of scoring auctions with different scoring functions. However, Beil and Wein (2003) do not directly address the risk of strategic bidding and in essence presume truthful revelation in the sequence of trial auctions.

The procurement system should enhance competition on both price and quality while being as simple as a posted price market. An attempt to do exactly this is multi-attribute yardstick auctions, that uses the so-called yardstick competition in order to both facilitate price competition across multi-attribute offers and to postpone the buyer's procurement decision to simply select the most preferred commodity or service as in a traditional market with posted prices. However, as Bogetoft and Nielsen (2008) point out, straightforward use of yardstick competition introduces sophisticated strategic manipulation. Hougaard et al (2013) analyze this in more depth and show that a simple straightforward use of yardstick competition promotes truth telling and makes complex procurement look like posted prices. These papers are in line with the ongoing challenge in designing operational flexible trading systems that facilitate transparent competition on both price and other attributes - while keeping the transaction costs low.

A central part of the yardstick auctions is to never reveal the submitted bid, i.e., only the resulting yardstick bids are revealed. Therefore, SMC has a central role in such auctions.

3.3.5 On the use of auctions in PRACTICE

Auctions are widely used to coordinated demand and supply both in terms of price and quantity but also with respect to other attributes such as quality. Since, the set of rules that make up an auction differs a lot from market to market, a large variaty of auction design exists and more is coming. However, most auctions rely on a trusted third party to make sure that information is sealed or disclosed at the right time to the right parties, which makes it an ideal use case for SMC.

Different use cases of auctions have been discussed in PRACTICE. Two of the most promissing use cases are briefly discussed in this Section: An auction market for reallocation of stocks among retailers and an auction to settling discount rates to motivate truthful preorders. Though, there are no plans or allocated resources for implementing SMC based auctions in PRACTICE at this stage. However, the various uses of auctions will be further discussed in WP24.

Auctions that promote truth-telling is an interesting class of auction with a potential interplay with other information systems. On one side such auctions may feed back information to decision support systems (e.g. for predictions) and on the other side decision support systems may be used directly to construct optimal bidding (ultimately automate bidding). This type of auctions emphasize the importance of integrity and confidentiality i.e. the properties delivered by SMC.

3.4 Conclusions and outlook

In the previous sections we have described formal models and algorithms for collaborative supply chain scenarios in the aerospace and consumer goods industry. We did not only develop and analyze scenario-specific models to actually match DTA's and Arcelik's business requirements and processes, we also provided insights regarding implementation details such as process flows, interdependencies between different software modules, pseudocode and input and output data sensitivity analyses.

Specifically, for the aerospace scenario we proposed a secure cloud-based collaborative planning approach comprising collaborative maintenance demand forecasting, spare parts management, engine arrival slot scheduling and capacity planning modules. Some first exemplary numerical experiments indicate potential benefits in a two digit range. For the consumer goods scenario we again developed a collaborative demand forecasting mechanism. This forecast is then used in a second step for a collaborative inventory planning model, more precisely single-vendor multi-retailer vendor managed inventory.

Additionally, some first ideas regarding the use of auctions for supply chain coordination where explained.

In the next phase of the project we will further detail implementation requirements in order to make secure implementation in a cloud environment feasible. We will also continue to analyze the models presented in this deliverable. In order to estimate potential benefits of the collaborative planning scenarios we need to perform numerical analyses based on actual (though normalized and historical) company data. The benefit analysis is needed in order to convince the supply chain members to partner, invest in and use the collaborative planning systems. Naturally, this means we also have to develop appropriate benefit sharing mechanisms in order to incentivize the respective parties in the supply chain.

Chapter 4 Risk assessment in supply chains

4.1 Introduction

In the highly competitive scenarios of today's economy, supply chains face relentless pressure to perform better, faster, and cheaper. Supply chains supporting aeronautic maintenance processes like the aero engine overhaul supply chain (see Section 2.2) are no exception. Within these supply chains, collaboration is about coordinating multiple flows of information among supply chain partners, linking their business processes, and providing solutions to ensure the eventual generation of consistent outcomes. It is well known (Kaipia & Hartiala, 2006) that supply chain collaboration can become unreliable due to partners failure or unwillingness to share the right information at the right time, especially when they perceive that timely information sharing will not be *per se* a source of revenue or benefits for themselves (Damiani, 2009).

As said before, information sharing in aeronautic supply chains includes on one side optimization and cost reduction for all the involved actors, and risks of information leakage and break of the confidentiality for the shared data on the other side. In particular we consider risks related to the unappropriated use of the accessed information to gain advantages per se or for one of the competitors. Some of the risks for the actors of the aero engine overhaul supply chain are listed in Table 2. In this **chapter** we focus on a specific but important category of data disclosure events, the ones that bring one or more parties taking part to a cloud-based business process to know more information than the process execution would entail. These unwanted disclosures may be due to intentional publishing of supposedly protected information items, or to carelessness in the communication protocol implementation and deployment, e.g. when one party is using the same mobile terminal previously used by another and can reconstruct the information items held. We call these events *process-related data disclosures*, in order to distinguish them from disclosures due to conventional eavesdropping attacks.

4.2 Risk in supply chains: general notions

Traditionally, the term supply chain risk factor has been used to designate various types of unfavourable events (see (Gaonkar & Viswanadham) (Rao & Goldsby, 2009) (Sodhi, 2012) for a classification) affecting one or more supply chain actors, and endangering the achievement of the supply chain's business goals. While early efforts toward supply chain modelling did not explicitly address risk today it is widely recognised (Jüttner, Peck, & Christopher, 2003) (Jüttner U. , 2005), (Manuj & Mentzer, 2008) that supply chain models need to be enhanced to include means by which risks can be represented and addressed, increasing the supply chain's resilience.

To date, much attention has been given to exogenous risk factors, i.e. factors external to the chain ((Jüttner, Peck, & Christopher, 2003) (Rao & Goldsby, 2009) (Peck, 2005)); among them volatility of demand (Jüttner, Peck, & Christopher, 2003), technological or market dependencies (Hallikas, Karvonen, Pulkkinen, Virolainen, & Tuominen, 2004), supplier concentration (Tang, 2006), scarce sources (Park & Ungson, 2001), and issues related to the social and natural environment (Peck, 2005). Less attention has been paid to endogenous risk factors: among the least studied risk factors are *dysfunctional behaviours*. Some of these risks relate to (hostile) actions that actors can undertake using the information shared in the supply chain.

The structure of supply chains consists of potential suppliers, producers, distributors, retailers or customers. These units are interconnected by material, financial, information, and decisional flows. Optimization of these flows allows a supply chain to successfully sustain competition with other supply chains. As any other collaborative alliance, supply chains rely on trust among partners, and even the perception of a risk due to partners' misbehaviour can have repercussions on the chain's effectiveness. The literature shows an emerging trend toward the study of advanced supply chain management (SCM) techniques, which can be effective, however, only after overcoming some barriers originally defined by (Mentzer, 2001) as “SCM antecedents”, including confidence, diligence, coupling, organisation structure compatibility, vision, key processes’ specification and top management support. To all those factors an appropriate degree of information sharing represents a key element.

The level of information sharing among partners plays a key role in the effectiveness of the supply chain and has two sides. Indeed, substantial amounts of information are interchanged daily between manufacturers and retailers, retailers and consumers, companies and investors. Usually the “vertical” transmission of information (i.e. between a retailer and a manufacturer) has both “direct” effects on the profit sharing between the engaged parties and “indirect” effects on other competing companies (Li & Zhang, 2003). The positive side of information sharing consist in the fact that it can enhance the performance of the chain, the negative side in that it can increase the probability of dysfunctional behaviour.

Some researchers argue that a high level of information sharing is a good indicator of a supply chain's effectiveness, because it increases the partners' confidence in each other: knowledge sharing has a number of advantages in a supply chain since important decisions can be made by evaluating the interest of all the actors of a chain, attaining overall cost optimization. The well-known “bullwhip effect” (Lee Hau & Whang, 2000) is an example of the inefficiencies occurring within a distributed control supply chain when insufficient information sharing makes the chain unable to respond to demand variability: in that case small changes in consumer demand can result in large variations in orders placed upstream. The bullwhip effect can produce very large swings as each company inside the supply chain tries to solve the problem from its own perspective, resulting in increased cost and poorer service. Information sharing is recognized as one way of countering this effect (Liu & Kumar, 2003).

On the other hand information sharing opens the door to dysfunctional behaviour from insiders that can interfere, typically, with the chain optimization process. A key process of supply chain management is the optimization of value and product flows on the basis of the local parameters (such as capacity, etc.) provided by each partner. In order to compute the chain-wide optimum, each actor needs to share information with the other partners in the supply chain (or with a trusted external decision maker). This process may pose some stress on the relations among the actors taking part in the chain, since the chain-wide optimal point do not usually coincide with the maximum profit for individual participants. Hostile actions can be undertaken during this process, including deliberate misrepresentation of reality based on the information obtained about the other partners (e.g. with the intention of altering the distribution of orders). Typically misrepresentation can refer to of supply, market and warehouse capacity: if a major product line is supplied by a sole supplier, the entire chain operation may be at risk if the supplier capacity results to be much lower than the capacity the supplier announced; even when a supply chain has multiple suppliers, supply issues may arise when high percentages of a crucial product are sourced from few suppliers: similarly, when a supply chain has multiple Points of Sale (POS), market issues may arise when the demand at one or more POS changes.

Due to the opposing effects of information sharing a given pre-agreed level of sharing can be enforced among supply chain partners. In principle any deviation from this pre-agreed level can influence the chain performance.

In this chapter we develop the case in which the deviation from the planned level of information sharing is due to information leakage, defined as the disclosure of given information items to parties, who are not supposed to have any visibility over that item, according to the communication protocol of the business process.

4.3 Rationale of the presented approach

In the deliverable D31.1 we presented a novel, *process-oriented* risk assessment methodology aimed at assessing process-related data disclosure risks on cloud computing platforms we defined the follow concepts:

Risk. From the economic standpoint, the *risk* of an "adverse event" (a.k.a. "feared event") E for a given actor A is often represented as the product of the damage $I(E)$ (expressed in currency units) in which A incurs when E really happens, times the likelihood that E might happen, traditionally represented in terms of probabilities. In symbols:

$$R(A,E)=I(E)*Pr(E)$$

In the computer security context, one needs to identify all the adverse events as manifestations of security threats and for each estimate $I(E)$ and $Pr(E)$; then the overall risk is computed by a suitable aggregator.

Probability assessment based on Shapley value. In the above scenarios, a process-based model can provide insight on probabilities of disclosure events, i.e. participants putting together the information they know. First, we derive the probability distribution associated to a continuous variable, namely the degree of perceived unfairness in the maintenance process resource allocation. Then, we use the probability values of this distribution to compute discrete probabilities of disclosure attacks associated to each subset of actors. Our technique is firmly grounded in the economics underlying the process: insider attacks are more frequent when there is a perceived unfairness of the redistribution of process payoffs in the process.

The problem of how profits of a coalition should be distributed is a well-known one in economics (it is an instance of a general problem called distributive justice). In the case of the partners in a collaborative processes like supply chains, a quick solution can be obtained by computing the so-called Shapley Value: "given a coalition, the contributions of the actors to the process, and the surplus value produced by the process, the Shapley value yields a ideal allocation of that value, often called the process fairness point".

Our methodology uses the Shapley approach to compare the cloud-based process' fairness point to its current allocation of value. The idea is that the distance between these two points provides a measure of disgruntlement that can be used to estimate the probability of an insider attack by each actor: the closer the two points for a party, the less more likely there will be no attack on its part. Technically, given the Shapley Value for each actor in each subset, we define a probability distribution based on the difference between the Shapley Value and the actual resource allocation for that actor. If this difference is positive, it means that the actor is under-rewarded for his contribution, corresponding to a non-zero probability that it will take part to an insider attack.

Impact. As far as the impact (a.k.a. severity) is concerned, its precise quantification is often a challenge. In our methodology, we provide an evaluation of costs taking into account the value of the disclosed information by means of a set of techniques known as Value of Information (*VoI*) analysis.

4.3.1 *The overall methodology*

In our approach, managing risks related to the execution of a business process *P* in presence of threats constitutes itself a process (usually called *risk management process*, in symbols $M_{R(P)}$) where alternative techniques for dealing with threats are compared according to a procedure. The output of $M_{R(P)}$ is a *risk mitigation strategy*, which consists of modifications to *P* that have some effect on the risk of executing it, including the introduction or removal of security controls.

In this subsection, we summarize our methodology for comparing alternative risk mitigation strategies. This methodology does not provide specific guidance on the choice of mechanisms that will actually counter the threats; rather, it allows comparing the residual risk of competing risk strategies. Although qualitative comparison is supported, the methodology aims to quantitative cost-benefit calculations, assessments of risk tolerance, and quantification of preferences involved in $M_{R(P)}$. Let's provide a step-by-step description:

- The first step is the stakeholder identification, where we identify the actor set A of our business process P and compute its power set 2^A. In our approach, process stakeholders include all participants to *P*. Namely, our actor set includes all actors who, according to the risk assessor, may in any way get the capability of reading (or writing) information shared during *P*'s execution.
- The second step consists in the *formalization of the business process model*, using the syntax proposed in D32, which represents two types of actions: (i) *message exchanges* and (ii) l*ocal computations*. It is important to remark that while execution-oriented process models usually contain control structures like conditions and loops, our process model syntax expresses all possible execution paths independently, i.e. as separate models. The next step takes care of this.
- The third step consists of *process streamlining*, which includes *loop unrolling* and *re-encoding of conditions as parallel paths*. Here we do not enter into the details of business process streamlining, as process improvement techniques have been deeply studied since the Eighties and are discussed in detail in the technical literature (see for instance the rich bibliography of (Scheer & Nuttgens, 2000). However, software toolkits supporting our methodology will have to provide guidance w.r.t. process streamlining.
- The fourth step, *identifying re-constructible knowledge*, consists in computing the knowledge set K_S for each subset *S* in 2^A. Each knowledge set includes all the knowledge that members of *A* can achieve by putting together the information they hold.
- The fifth step consists in *estimating the disclosure impact* of K_S for each subset *S* in 2^A at each step of the business process *P*,
- The sixth step consists in *estimating the possibility distribution for the defection* for each individual actor, based on process related information, on elicitation of expert opinion and on the techniques shown in D32.
- The seventh step consists in *estimating the collusion possibility distribution* for each subset *S* in 2^A at each step of the process P. Once again it is important to remark that this estimate needs to be process-specific (as it will take into account the micro-economics and social relations underlying *P*) and take into account multiple causes of collusion, including dysfunctional behavior, intervention of regulatory authority and others.
- The last step consists in *aggregating the products* between *i*) the possibility distributions of collusion for each subset S in 2^A and ii) the possibility distributions of the impact of K_S at each step of the process *P*, obtaining the *total risk* related to the process, in the form of a possibility distribution, based on which one can take the risk management decisions.

4.4 Risk Assessment case study

In this subsection we apply our risk assessment methodology to one of the processes described in this deliverable, namely the scheduling maintenance process presented in section 3.1.3.2.

Organizations are now aware of the importance of explicitly defining their business process models at the level of detail suitable for the purposes of the models themselves, easing the communication and understanding of business needs. Here we adopt a Model Driven Development (MDD) approach to describe and enact transformations from design to risk assessment-oriented process models (Delgado, García-Rodríguez de Guzman, Ruiz, & Piattini, 2010).

The maintenance scheduling module deals with determining optimal delivery points for upcoming engine to be overhauled by the MRO service provider. As shown in Figure 25(Section 3.1.3.2), the input data of the maintenance scheduling module is provided from one its predecessors that is not directly connected to it.

To carry out our risk assessment, we compute a new view over overhaul management process consisting of modules with two states, being either visible or hidden. The former refers to the sub-processes that determine the input data of the maintenance scheduling module, while the latter relates to the sub-processes that do not play any role in providing data inputs to the scheduling module.

Figure 48 shows a representation of the process view for two airlines and one engine producer. White and gray shaded rectangles represent visible and hidden sub-processes, respectively.

With the assumption that each sub-process has a starting and ending time, the difference between two such times gives the duration of sub-process, which is considered equal to zero for the hidden sub-processes. In Figure 48, the input data of each airline and engine supplier is a vector *V* containing both sensitive and non-sensitive data.

The engine arrival time *T** is a function of *F* that is, in turn, a function of airlines and supplier inputs (denoted here as *V1* and *V2*). Formally, we can write *T*= S(F)=S(G(V1,V2))*.

In the following, we will schedule the next engine arrival time slot for overhaul by computing the functions *G* and *S*.

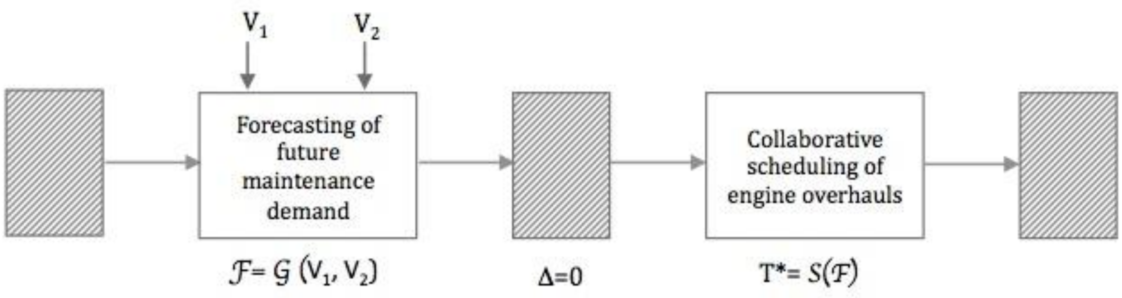


Figure 48 View of the overhaul management process to compute the timeline scheduling

4.5 Computing the timeline scheduling

In this subsection, we compute the timeline scheduling for engine arrival considering the process view shown in Figure 48.

We claim that there is an alternative way for reducing arbitrariness when doing quantitative risk estimates: making use of knowledge of the business processes that are being moved to the cloud and of the economics underlying them. Let us focus again on confidentiality breaches on cloud users' data. In order to "go quantitative" on this risk, we can model a process taking place on the cloud whose participants include service and cloud providers.

Figure 49 shows a simplified representation of the process actors' set, including two airlines and one supplier. This set of actors underlies the process for scheduling engine arrival slots. Essentially, the airlines and the supplier (IN nodes) provides vector Vi, which contains sensitive data (usage parameters) and non-sensitive data (threshold parameters and standard information). This vector is sent to a cloud-based service (COMP node) provided by a global scheduling maintenance company. This service, using algorithm 1 presented in Section 3.1.3.2.2, elaborates the timeline scheduling.

Each airline receives the portion of the scheduling that concerns its engines (RES node).

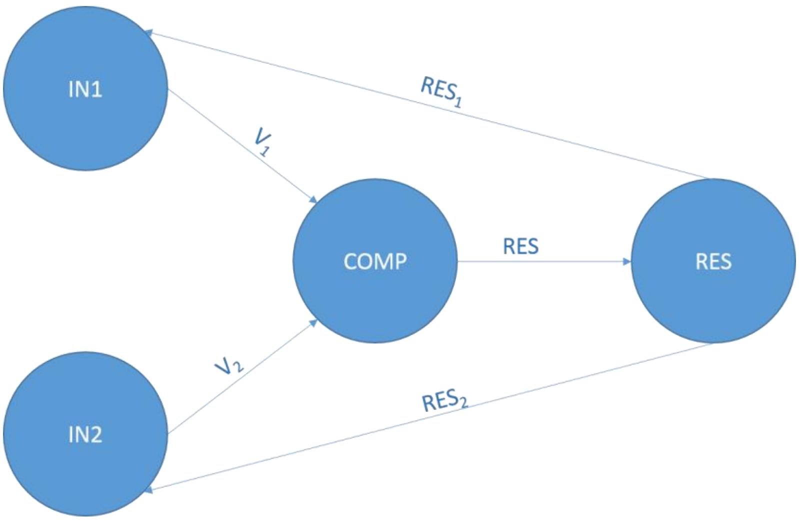


Figure 49 The scheduling maintenance process (two airlines) represented by the Bogdanov model

Of course, no airline involved in the overhaul maintenance process wants to disclose the sensitive portion of its *Vi* and *RESi* data that is sensitive to competitors, especially to the ones working on the same routes who could use the competitors information to improve their own decision-making. Such information would be even more sensitive for military aircrafts. Our simple view over the overhaul maintenance process highlights very well the three dangers we mentioned before.

Firstly, moving the process to the cloud may alter the probability of someone laying hands on the V_i, e.g. by reducing the effort necessary to carry on an attack with respect to an on-premises implementation of the same process.

Secondly, the opacity of the cloud stack may bring airlines to assume that their V_i – being transmitted to the cloud-based service using a secure channel - will also be stored in encrypted form on the cloud, while the cloud provider knows – but does not say – that this may not be the case. Finally, participating airlines may not fully realize that potential inside attackers include the cloud provider itself and its employees, who could simply leak the V_i to competing airlines.

Impact Assessment

A quantitative estimate of the impact of information disclosure can be obtained in a similar way. Namely, we quantify the Value of Information (VoI) for each data item shared in the overhaul maintenance process. The idea here is that, for each participant to the process getting to know additional information (e.g., messages exchanged among other participants, or the content of another actor's local memory) beyond what is strictly necessary to carry out the process may or may not bring a benefit. For each information item potentially disclosed by a (non-zero probability) attack, we quantify the benefit of knowing it for the attackers and check whether this benefit would match the cost of collecting it and the potential penalties.

Risk-based comparisons of security mechanisms.

Applying our assessment technique to alternative versions of the overhaul maintenance process obtained by inserting different security controls, the risk analyst can easily determine how much risk alleviation can be bought by each dollar invested in each type of security control. Let us focus on what we have at our disposal to guarantee confidentiality on the cloud. While simply encrypting data may be enough to protect users' privacy when storage is outsourced, the going gets tougher when computational tasks are also moved to the cloud. Not only cloud-based computations must be carried out without decrypting the data; the structure of the outsourced process itself (e.g. the order of tasks) must also be protected.

In the deliverable D31.1 we described a set of techniques for securing cloud-based processes using Secure Multiparty Computation (SMC). In Secure Multiparty Computation (SMC) protocols, parties hold private inputs that are used to jointly compute a function, still hiding the input to the other parties. Indeed, the security requirement of such kind of computation is that nothing is learned from participating to the protocol other than the final result.

Once a process has been converted into a SMC computation it can be moved to the cloud without fear of disclosing to the provider the data upon which execution will take place. However, many different techniques exist, requiring different security controls with different deployment, execution and management costs. The risk analyst needs to be able to compare the risk reduction attainable using a given security solution with these costs.

Let us now go back to the maintenance scheduling process (Section 3.1.3.2). In our view the actor set is $A= (IN_1, IN_2, COMP,RES)$ where actors *IN1, IN2* are the airlines and they hold the information items in the vector containing the data related to the engine V_n. Actor COMP1 is the cloud-based service that computes the overall engine maintenance schedule, while actor RES securely sends the timelines to the airlines.

The process execution is represented by the simple diagram shown in Figure 50, where the input actors *IN1* and *IN2* – modelling the airlines - send data V_1 and V_2 to the cloud-based node COMP, who (i) computes a local function *f() =schedule(V1,V2)* that gives the overall maintenance schedule and (ii) sends the result to node RES who, in turn, sends the result vectors to *IN1* and *IN2*. Of course, these are two different vectors that do not show the maintenance events of the competitor.

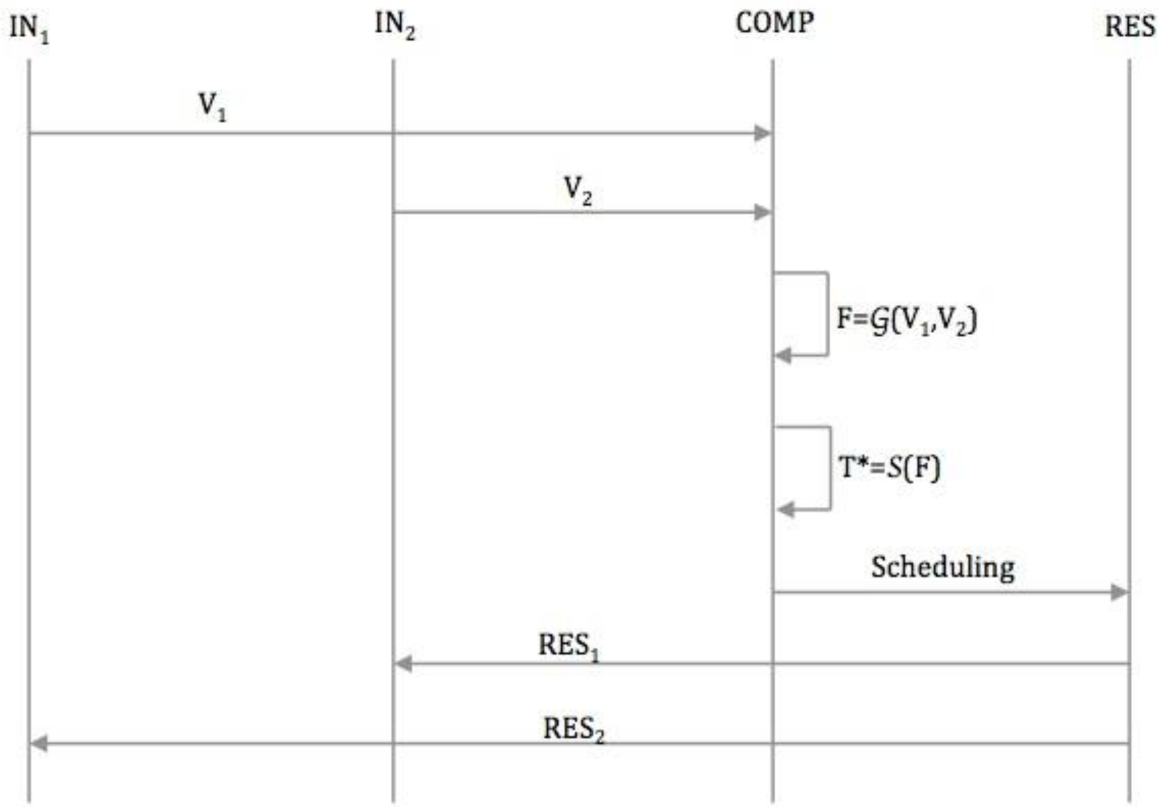


Figure 50 The scheduling maintenance process (two airlines)

At each time t, the (possibly empty) common knowledge of any subset of actors is composed of the V items that have been received in their entirety by all members of S. For conciseness, here we will take into consideration only the input and computational nodes; therefore the cardinality of the actor set *A* is 3 and its power set includes all possible 2^3 subsets. We focus our attention on the subset {IN2 COMP}: in fact, at a certain time in the process execution, this coalition shares both vectors V1 and V2 and it permits to the node IN2 to know the V_1 of the competitor. Starting from this assumption, we can evaluate the risk as seen by IN1:

$$Risk(IN1)=P(COMP \cup IN2)* LOSS(IN1)$$

Here, *P()* represents the probability (assessed by IN1) that the set (COMP1 $\cup$ IN2) will disclose the information it holds and LOSS() expresses the Loss Expectancy of the airline. Assuming that the events of COMP be willing to peek V1 and IN2 to be interested in them are disjoint, $P(COMP \cup IN2)= P(COMP)*P(IN2)$ and

$$Risk(IN1)= P(COMP)*P(IN2)*LOSS(IN1)$$

Assuming the Shapley analysis of the collaboration gives us P(COMP)=0.3 and P(IN2)=0.6 we obtain Risk(IN1)=0.18*LOSS(IN1). Of course, the loss of IN_1 when IN_2 gets to know the V_i time of its engines cannot be quantified once for all; this Value of Information (VoI) depends on the specific instance of process execution. However, an estimate can be provided based on the better decision making that IN_2 will be able to achieve thanks to the leak. Computing the VoI is out of scope of our methodology; however, we remark that in the air transport industry the business value of information regarding the maintenance status of a competitor's aircraft an be (and has been(empirically quantified. This issue will be discussed in detail in the next sub-sections

Alleviating Disclosure Risk

Let us now see what happens when we introduce secret sharing for computing the maintenance schedule. Figure 51 shows an example where the cloud-based service is performed by two nodes, possibly located on different clouds.

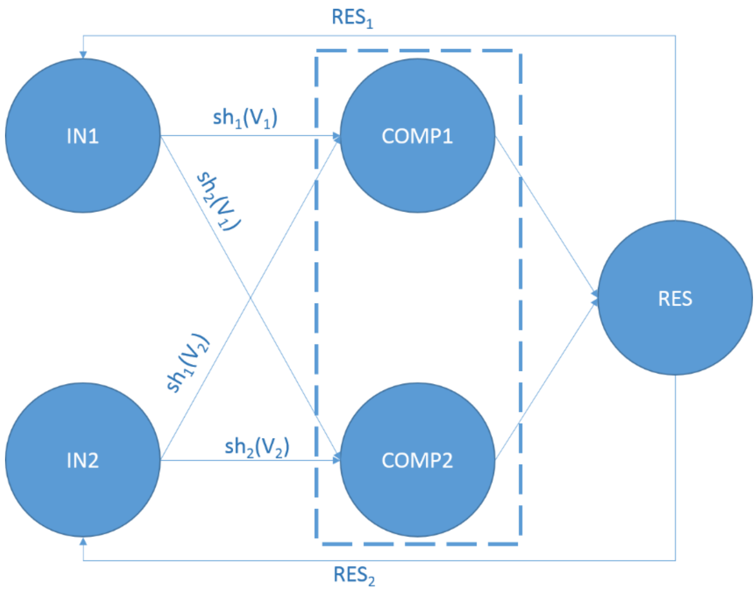


Figure 51 The secured scheduling maintenance process

In this way, the original process *P* is modified introducing a new node (COMP2) and the function that permits to schedule the engine maintenance is computed by the cooperation of two cloud services.

Again, the airlines locally compute the Public Variable for the engines of all their aircrafts obtaining a vector V_i, and using a secret sharing technique creates two shares ($sh(V_i)$) which are passed to a cloud-based service published by the maintenance company. Using this technique, each computational node knows only partial data and cannot disclose the entire vector provided by the airlines. Figure 52 shows our model of the new maintenance scheduling process.

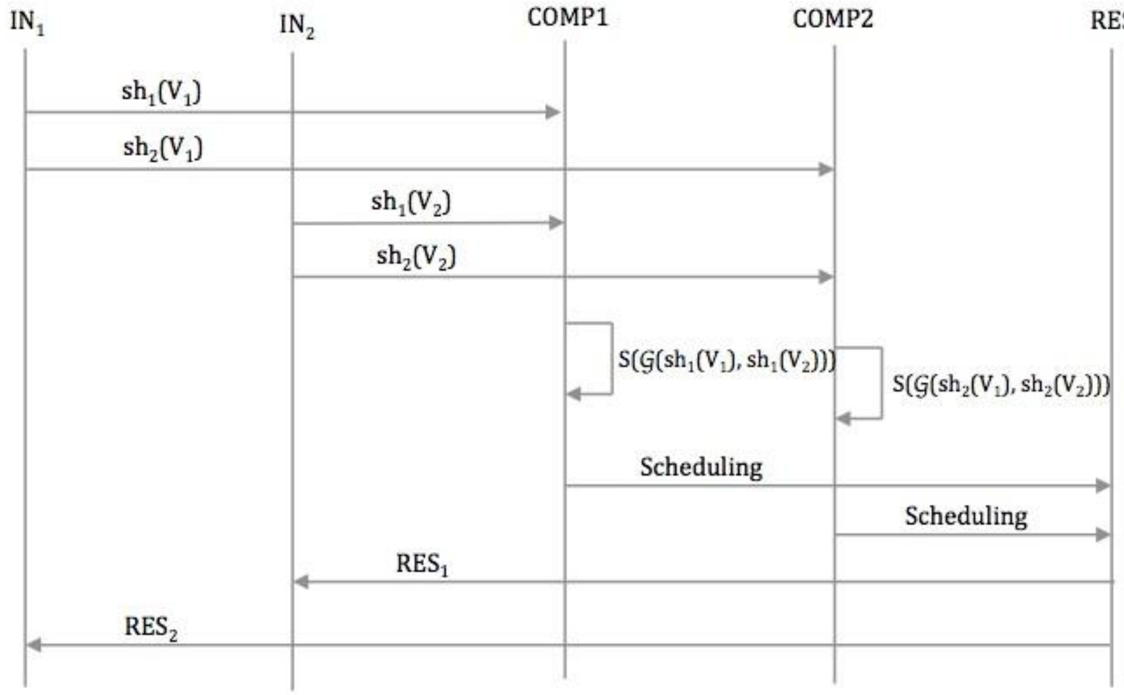


Figure 52 The secured scheduling maintenance process (two airlines)

Adopting this business process only the coalition that includes all computational nodes can disclose the initial vector Vi, and for this motivation we focus our risk analysis on the set composed by {IN2, COMP1, COMP2}:

$$Risk(IN1)=P(COMP1 \cup COMP2 \cup IN2)*\ LOSS(IN1)$$

$$Risk(IN1)= P(COMP1)*\ P(COMP2)*P(IN2)*LOSS(IN1)$$

Assuming P(IN2)=0.6, P(COMP1)=0.3 and P(COMP2)=0.3 the final estimation of the risk decreases to

Risk(IN1)= 0.054*LOSS(IN1)

This gives us a risk reduction of 01.26*LOSS(INI), which must be compared with the cost of deploying the process on two services rather than one.

This analysis can be repeated periodically following the evolution for IN2 of the value of the information held by IN1 along market cycles, or triggered by changes in the economic context where the two airlines operate.

As mentioned before, the loss of IN1 when IN2 gets to know the V_i of IN1's engines cannot be quantified once for all; indeed, the corresponding Value of Information (VoI) depends on the specific instance of maintenance process we consider. However, a worst-case estimate can be done based on the improved decision-making that IN2 will be able to achieve thanks to the leak. By knowing the moment the competitor has high RTFs, IN2 can hope to lure valued customers away. Airlines sell a variable number of seats' packages in order of price (less expensive package first) (Mumbower, Garrow, & Higgins, 2014)**.** If a competitor airline, operating flights on the same route of IN1, starts a special offer to lure new customers away from IN1 and deploys an aircraft with a higher number of seats, IN1 will lose part of its customers, those who would have purchased the most expensive seats and, therefore, the most profitable ones. This way, IN1 experiences the following direct and indirect losses: 1) direct economic loss due to the lost customers, 2) reduction in its brand recognition, 3) future losses due to the fidelity the lost customers will devote to the competitor in future flights.

Let us now discuss how IN1 can get a quantitative estimate for LOSS(IN1). While this computation requires some knowledge of the operating context, it is far from impossible.

Knowing when IN1’s high capacity aircrafts need to undergo maintenance, IN2 can take the opportunity to deploy its own higher capacity aircrafts and offer more business class seats. Assuming for simplicity that IN1 and IN2 operate on the same route, the loss of IN1 will depend on the demand for business class seats on this route and on other variable costs (for example, the cost of fuel) at the time the attack takes place.

Considering that the profit margin (EBIT) of the European airline industry in the period 2007-2013 lies in the interval from +4% (2007) to -2% (2009) (Khan, CEO Association of European Airlines, 2014)**.** it is clear that loosing even a fraction of the highly profitable business class traffic can seriously affect IN1. Assuming that (i) IN1’s yearly revenue on the route shared with IN2 to be 200 MEuros (a figure in line with revenues generated by several EU routes27), and (ii) the schedule to be computed monthly, so that he attack duration is at most one month, IN1’s loss can be bounded by (200/12)(0.04-(-)0.02)=200(0.06)=1 MEuros. The risk can be then quantified at 0.18 MEuros.

Adopting the modified business process only the coalition that includes all computational nodes can disclose the initial vector Vi, and for this motivation we focus our risk analysis on the set composed by {IN2, COMP1, COMP2}.

Assuming P(IN2)=0.6, P(COMP1)=0.3 and P(COMP2)=0.3 the final estimation of the risk decreases to

*Risk(IN1)= 0.054*LOSS(IN1) = 0.054 Meuro = 54 KEuro*

This gives us a total risk reduction of 126 KEuros, which IN1 can compare with the costs of (i) developing or buying the share generator (ii) deploying the maintenance process on two cloud services rather than one.

[27] For example, in 2011 the Rome-Milan route (around 1,5 million passengers in 2012) generated around 6% of Alitalia’s total revenue, which amounted to 3,5 billions Euros in the same year (see http://espresso.repubblica.it/attualita/cronaca/2012/04/20/news/roma-milano-che-cosa-conviene-1.42394)

Chapter 5 PRACTICE architectures and tools

In this section we give a brief overview of possible techniques that are introduced in other work packages of the PRACTICE project and can be useful for implementing the models presented in this work in Chapter 3. Different approaches are possible and selecting one comes along with a trade-off between functionality, security and efficiency.

In Deliverable D11.1 a theoretical evaluation of the existing secure computation solutions is presented. One important aspect for choosing a concrete technique for implementing a system for secure cloud collaboration is the adversarial power. For this, different attacker models are defined. Furthermore, it is highlighted that using specialized protocols for computing specific functions -- as we do -- can be preferable to using generic protocols in aspects of performance. However, designing such specialized protocols requires cryptographic expertise.

Examples for existing real-world applications are given in Deliverable D21.1 - Deployment Models and Trust Analysis for Secure Computation Services and Applications. One comparable project is described there in Section 3.8 that has been implemented in EU project secureSCM[28]. While the physical deployment model may differ, adopting the trust relations and security assumptions can be considered.

A state of the art analysis is given in Deliverable D22.1. Introducing existing secure application frameworks and secure programming languages to specify different parts of a secure computing system ranging from easy to consume SDKs to secure databases; not all are fitting approaches for our use case. For example, TASTY and CMBC-GC are not applicable because they are designed for addressing secure two-party computation. While the same argument holds for Fairplay, its extension FairplayMP is a system for multi-party computation using a high-level language called Secure Function Definition Language (SFDL). However, FairplayMP has not been used in real-life application, and so it is questionable if it is an appropriate technique for implementing such comprehensive models as described in Chapter 3 While performing secure multi-party computation using generic constructions like the Beaver-Micali-Rogaway protocol is one possible method encrypted databases are another promising technique for building reliable privacy mechanisms for cloud applications. Here, an encrypted database can ensure data protection against a cloud service provider. In more detail, the solution to this outsourced security problem is to encrypt data before sending it to the cloud. This is easy to implement for simple storage, but the clients must remain able to query the outsourced database. “Sharemind 2 Secure Database” for Sharemind applications and “Encrypted Query Processing for Business Applications” for SAP HANA are two possible solutions for encrypted databases. Since SAP software is a widely spread solution for managing business operations and customer relations it is quite reasonable to apply techniques that are easy to integrate into the SAP environment.

Finally, designs for tools which make secure computation possible are given in Deliverable D22.2. Specifications for upcoming secure computation frameworks for different programming languages like Java and C are presented. Furthermore, a large subset of SQL queries can be implemented by combining different encryption types and using the appropriate one. Supporting these common used languages will enable programmers to write secure computation programs and even reuse already existing source code.

[28] secureSCM: project no. FP7-213531 in the EU FP7-ICT FET Open scheme

Figure 48 illustrates the links of our use case to other work packages (depicted as underlined text). As an example we show one possible solution for implementing a few models described in Chapter 3 in a secure way. While the used technology – here SAP HANA processing encrypted SQL queries -- can be replaced with other techniques the links to the other work packages remain the same.

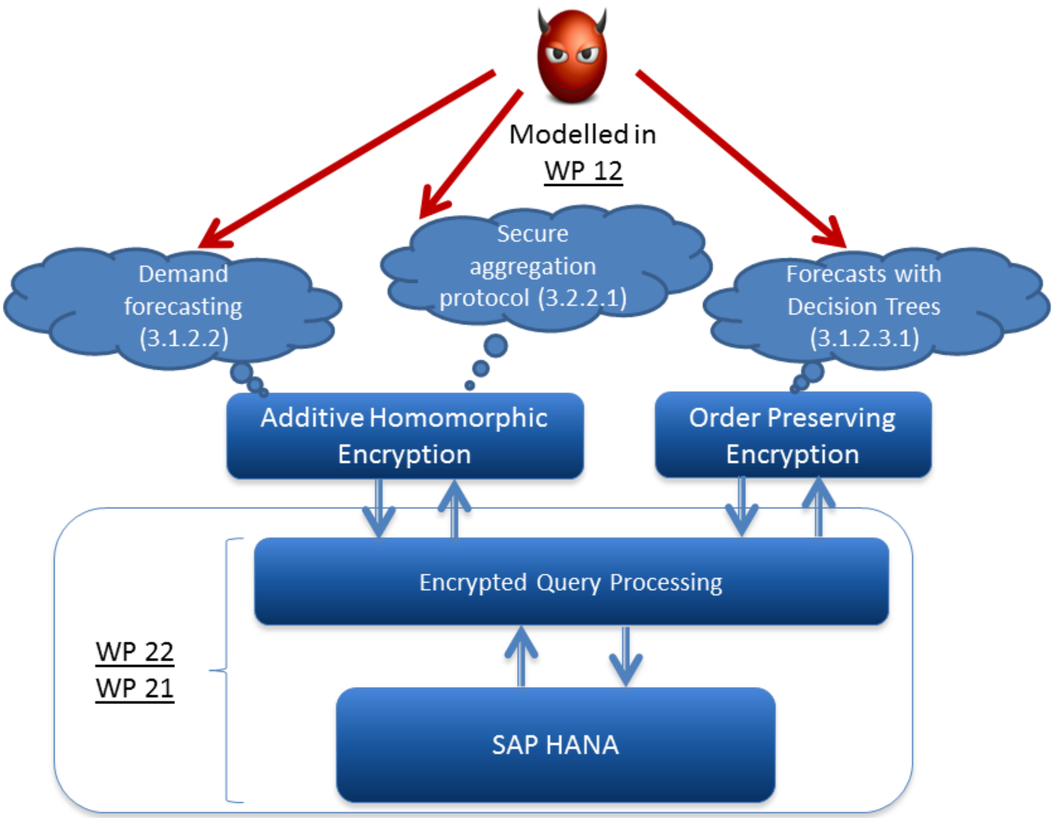


Figure 53: One possible implementation for some models presented in Chapter 3.

All these aspects will help to develop a protocol for implementing a prototype and computing the models described in Chapter 3 in a secure and efficient way for the upcoming deliverable D24.4.

Chapter 6 Conclusions

The competitiveness of companies involved in complex products production is strongly related to their capability to be connected with customers and suppliers. This is the case in the aeronautic and the consumer goods industry, the first one delivering high capital and long life products, the second one characterized by an intertwined supply chain and high volatility.

In order to effectively focus the management effort and the process modelling strategy on the supply chain relationships and IT network infrastructure, it is required to reshape individual business models, in particular by defining a global supply chain business objective.

In the aeronautic after sales service scenario (specifically, the aircraft engine MRO service), **the fleet management is a global supply chain business process** requiring the committed participation of many actors, from end users to part suppliers. In particular, it requires **service provider and end users** (but also other parties) **to agree on the same business objective: the availability, at the end user bases, of the engines in the ready-to-use status.** Only in such a condition, engines can be efficiently used to provide profits to their owner.

In order to make this business model actually running, **activities carried out in each node of the supply chain were designed**. Also, **a common database, populated with data from supply chain participants was designed. Forecasting and planning algorithms** compute demand forecasts, service plans and local inventory levels by processing data available in this common database. It emerged that the common database will contain **data considered confidential by the owners** with respect to other supply chain participants (process insiders) or with respect to external players, i.e., one having a business relationship with a supply chain participant. Thus, storing confidential data in such a system exposes owners to data leakage risks. The risks run by the engine owners are related to the possibility that other competitors can access information on its fleet status and on engine usage pattern, even by exploiting the relationship with a common MRO service provider. These kinds of events are enabled by MRO service provider having access to the collaborative planning system or the competitor being an user of the same collaborative cloud system (offered to another supply chain). Other parties run similar risks.

In the consumer goods industry, providing the right quantity of products to the right reseller in the right moment, when the demand actually appears, implies that all parties involved in the supply chain need to align their production and delivery activities. Aligning individual business objectives to a common one means developing a supply chain business model. In particular, a model based on **the Vendor Managed Inventory is proposed and redesigned** in order to meet the specificities of the target industry. Also in this case, **the model is implemented through sharing individual data, many of them considered confidential,** by owners with respect to other supply chain participants (for example other resellers operating in the same local area).

In order to satisfy the data protection requirements and to overcome concerns of the supply chain parties involved in the innovative business models, **novel computing models and algorithms** are developed by **taking into consideration the specifics of the secure computation technology**.

It appears that also **auction models can be designed to satisfy specific sub-goals of the supply chain business model**. In the consumer goods scenario, auctions can be modelled to reallocate stocks in the retailers' network. On the other hand, auctions can also be used to push (by rewarding) resellers to reduce the gap between 'pre-orders' and 'actual purchasing orders', in other words, they will be rewarded if they improve their forecasting capabilities or if they take on the risk of purchasing a product quantity closer to the pre-ordered one.

Many architectural solutions, data encryption strategies and protocol implementations can lead to similar collaborative forecasting and planning systems, differentiated only with respect to their security features. For example, specific implementations can concentrate the most confidential parameters on the most trusted cloud node, or can distribute the computation algorithm on many nodes so that no one of them has access to data to infer more information than what is expected. Different implementations will be associated with different service costs at the users' side, as well as to a higher cloud system complexity at the service provider side. For this reason, **a methodology to evaluate the actual business risks associated to a specific secure cloud implementation is proposed**.

In the next period two tasks will be carried out: the first one will be focused on the identification of specific requirements or conditions raised by the selected industrial scenario; the second task will be directly dedicated to the preparation of the industrial prototypes by leveraging the architectural frameworks and the tools developed in other WPs of the PRACTICE project.

Chapter 7 Bibliography

Aggarwal, Charu C.; Elmagarmid, Ahmed K.; Sheth, Amit P.; Yu, Philip S. (2008): Privacy-Preserving Data Mining. Boston, MA: Springer US (34).

Agrawal, Rakesh; Kiernan, Jerry; Srikant, Ramakrishnan; Xu, Yirong (2004): Order Preserving Encryption for Numeric Data. In : Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data. New York, NY, USA: ACM (SIGMOD '04), pp. 563–574.

Alfalla-Luque Rafaela, Medina-Lopez Carmen and Schrage Heribert, (2013), "A study of supply chain integration in the aeronautics sector", Production Planning & Control: The Management of Operations, 24:8-9, 769-784;

Anderson J. C. and Narus J. A., (1990), "A Model of Distributor Firm and Manufacturer Firm Working Partnerships", Journal of Marketing, Vol. 54, No. 1, pp. 42-58;

Arrow K.J. (1963), Social Choice and Individual Values. Yale University Press.

Atallah, Mikhail; Elmongui, Hicham; Deshpande, Vinayak; Schwarz, Leroy B. (2003): Secure supply-chain protocols. In CERIAS Tech Report 2003-13.

Aumann Y. and Lindell Y., (2009), "Security Against Covert Adversaries: Efficient Protocols for Realistic Adversaries", Journal of Cryptology, Vol. 23 Issue 2, pp.y 281-343;

Babaï, M. Z.; Syntetos, A. A.; Dallery, Y. and Nikolopoulos, K., (2009), "Dynamic re-order point inventory control with lead-time uncertainty: analysis and empirical investigation", International Journal of Production Research 47 (9), 2461–2483.

Banerjee A., Banerjee S., (1994), "A coordinated order-up-to inventory control policy for a single supplier and multiple buyers using electronic data interchange", International Journal of Production Economics, 85-91.

Barratt, Mark (2004): Understanding the meaning of collaboration in the supply chain. In Supply Chain Management: An International Journal 9 (1), pp. 30–42.

Beil DR, Wein L. (2003), An inverse-optimization-based auction mechanism to support a multiattribute RFQ process. Management Science 49(11), 1529–1545.

Bharadwaj A., El Sawy O. A., Pavlou P. A., Venkatraman N., (2013), "Digital business strategy: toward a next generation of insights", MIS Quartely Vol. 37, No 2, pp. 471-482;

Binmore, K. and Swierzbinski, J. (1999), Treasury auctions: Uniform or discrim- inatory?, Review of Economic Design 5, 387–410.

Bitran, Gabrial and Devanath Tirupati. Tradeo_ curves, targeting and balancing in manufacturing queueing networks. Operations Research, 37(4):547-564, 1989.

Bitran, Gabriel and Reinaldo Morabito. State-of-the-art survey: Open queueing networks: Optimization and performance evaluation models for discrete manufacturing systems. Production and Operations Management, 5(2):163-193, 1996.

Bogetoft P, Nielsen K. (2008), DEA based auctions. European Journal of Operational Research 184, 685–700.

Bogetoft, P., D.L. Christensen, I.B. Damgaard, M. Geisler, T.Jacobsen, M. Krøigaard, J.D. Nielsen, J.B. Nielsen, K. Nielsen, J. Pagter, M Schwartzbach and T.Toft (2008), Multiparty Computation Goes Live, Cryptology ePrint Archive, Report 2008/068.

Boyd, Stephen, Lin Xiao, and Almir Mutapcic. Subgradient methods. Lecture notes of EE392, Stanford University, Autumn Quarter, 2003.

Capgemini, (2009), “Maintenance, Repair and Overhaul (MRO)”;

Cetinkaya S., Lee C.Y., (2000), “Stock replenishment and shipment scheduling for vendor-managed inventory systems”, Management Science, 217-232.

Chari, V. V. and Weber, R. J. (1992), How the u.s. treasury should auction its debt, Federal Reserve Bank of Minneapolis quarterly review 16(4), 3–12.

Che, Y. (1993), Design Competition Through Multidimensional Auctions, RAND Journal of Economics 24(4), p 668-680.

Chen, F.; Drezner, Z.; Ryan, J. K. and Simchi-Levi, D. (2000): "Quantifying the Bullwhip Effect in a Simple Supply Chain: The Impact of Forecasting, Lead Times, and Information", Management Science 46 (3), 436–443.

Cramton, P., Shoham, Y. and Steinberg, R. (eds) (2006), Combinatorial Auctions, MIT Press.

Cripps, M. and Ireland, N. J. (1994), The design of auctions and tenders with quality thresholds: The symmetric case, The Economic Journal 104 (423), 316–326.

Cripps, M.W. and J. M. Swinkels (2005), Efficiency of Large Double Auctions. Econometrica, 74(11), pp. 47-92.

Damiani, E. (2009). Risk-aware Collaborative Processes. ICEIS.

Delgado, A., García-Rodríguez de Guzman, I., Ruiz, F., & Piattini, M. (2010). From BPMN business process models to SoaML service models: A transformation-driven approach. 2nd International Conference on Software Technology and Engineering (ICSTE)

Darwish M.A., Odah O.M., (2009), “Vendor managed inventory model for single-vendor multi-retailer supply chains”, European Journal of Operational Research, 473-484.

Deloitte, (2012), “Smarter MRO. 5 strategies for increasing speed, improving reliability, and reducing costs – all at the same time”;

Demirkan H., Delen D, (2013), “Leveraging the capabilities of service-oriented decision support systems: Putting analytics and big data in cloud”, Journal Decision Support System, Vol. 55 Issue 1, pp. 412-421;

Disney S.M, Towill D.R., (2003), “The effect of vendor managed inventory (VMI) dynamics on the bullwhip effect in supply chains, International Journal of Production Economics, 199-215.

Du T. C., Lai V. S., Cheung W., Cui X., (2011), “Willingness to share information in a supply chain: A partnership-data-process perspective”, Information & Management, Vol. 49 Issue 2, pp. 89-98;

Eppen, Gary D.; Martin, R. Kipp (1988): Determining Safety Stock in the Presence of Stochastic Lead Time and Demand. In Management Science 34 (11), pp. 1380–1390.

Eurich M., Oertel N., Boutellier R., (2010), “The impact of perceived privacy risks on organizations’ willingness to share item-level event data across the supply chain”, Electronic Commerce Research, Vol. 10 Issue 3-4, pp. 423-440;

Fang, Weiwei; Zhou, Changsheng; Yang, Bingru (2013): Privacy preserving linear regression modeling of distributed databases. In Optim Lett 7 (4), pp. 807–818

Flynn, B.B., Huo, B., Zhao, X., (2010), The impact of supply chain integration on performance: A contingency and configuration approach, Journal of Operations Management, Volume 28, Issue 1, January 2010, Pages 58-71

Frank, Eibe; Hall, Mark (2001): A Simple Approach to Ordinal Classification. In Luc de Raedt, Peter Flach (Eds.): Machine Learning: ECML 2001, vol. 2167: Springer Berlin Heidelberg

(Lecture notes in computer science), pp. 145–156. Available online at http://dx.doi.org/10.1007/3-540-44795-4_13.

Friedman, D. (1984), On the Efficiency of Experimental Double Auctions Markets. The American Economic Review, 74 (1), pp. 60–72.

Friedman, D. and J. Ostroy (1995), Competitivity in Auction Markets: An Experimental and Theoretical Investigation. Economic Journal, 105, pp. 22 –53.

Gaonkar, R., & Viswanadham, N. (s.d.). A conceptual and analytical framework for the management of risk in supply chains. In Proceedings of 2004 IEEE International Conference on Robotics and Automation, (p. 2004).

Gibbard, A. (1973), Manipulation of Voting Schemes: A General Result, Econometrica, 41, pp. 587–601.

Graves, Stephen C. (1999): A Single-Item Inventory Model for a Nonstationary Demand Process. In Manufacturing & Service Operations Management 1 (1), pp. 50–61.

Groves T., Loeb M., (1979), “Incentives in a divisionalized firm”, Management Science, Vol. 25 Issue 3, pp. 221-230;

Guajardo J. A., Cohen M. A., Kim S.H., Netessine S., (2012), “Impact of Performance-Based Contracting on Product Reliability: An Empirical Analysis, Management Science, Vol. 58 Issue 5, pp. 961-979;

Hall, Rob; Fienberg, Stephen; Nardi, Yuval (2011): Secure Multiple Linear Regression Based on Homomorphic Encryption. In Journal of Official Statistics (4), pp. 669–691.

Hallikas, J., Karvonen, I., Pulkkinen, U., Virolainen, V., & Tuominen, M. (2004). Risk management processes in supplier networks. International Journal of Production Economics(90), 47-58.

Hamlen K. W., Thuraisingham B., (2012), “Data security services, solutions and standards for outsourcing”, Computer Standard & Interfaces, Vol. 35 Issue 1, pp. 1-5;

Hariga M., Gumus M., Daghfous A., (2014), “Storage constrained vendor managed inventory models with unequal shipment frequencies”, Omega, 94-106.

Harris M. and Raviv, A. (1981), Allocation mechanisms and the design of auctions, Econometrica 1(49), 1477–1499.

Hoogh, Sebastiaan de; Schoenmakers, Berry; Chen, Ping; op den Akker, Harm (2014): Practical Secure Decision Tree Learning in a Teletreatment Application. In Nicolas Christin, Reihaneh Safavi-Naini (Eds.): Financial Cryptography and Data Security, vol. 8437. Berlin, Heidelberg: Springer Berlin Heidelberg (Lecture notes in computer science), pp. 179–194.

Hopp, Wallace, Mark L. Spearman, Sergio Chayet, Karen L. Donohue, and Esma S. Gel. Using an optimized queueing network model to support wafer fab design. IIE Transactions, 34(2):119-130, 2002.

Hougaard, J.L., Nielsen, K, Papakonstantinou A. (2013), A multi-attribute yardstick auction without prior scoring, MSAP Working Paper Series; Nr. 02/2013, University of Copenhagen

Hoyt J., Huq F., (2000), “From arms-length to collaborative relationships in the supply chain: An evolutionary process”, International Journal of Physical Distribution & Logistics Management, Vol. 30 Issue 9, pp. 750-764;

IBM, (2008), “Keep them flying. Find your winning position in the MRO game”;

Jammernegg, W., Reiner, G., (2007), Performance improvement of supply chain processes by coordinated inventory and capacity management, International Journal of Production Economics, Volume 108, Issues 1–2, July 2007, Pages 183-190

Jason Shiu Kong, (2007), “Information sharing in supply chain. Improving the performance of collaboration”;

John DC Little. A proof for the queuing formula: L=

Jouini, Oualid, et al. "Online scheduling policies for multiclass call centers with impatient customers." European Journal of Operational Research 207.1 (2010): 258-268.

Jüttner, U. (2005). Supply chain risk management: Understanding the business requirements from a practitioner perspective. The International Journal of Logistics Management, 16(1), 120-141.

Jüttner, U., Peck, H., & Christopher, M. (2003). Supply chain risk management: outlining an agenda for future research. International Journal of Logistics Research and Applications, 6(4), 197-210.

Kaipia, R., & Hartiala, H. (2006). Supply-Chain Integration through Information Sharing: Channel Partnership between Wal-Mart and Procter & Gamble. International Journal of Logistics Management, 3(17).

Karlin, Samuel (1960): Dynamic Inventory Policy with Varying Stochastic Demands. In Management Science 6 (3), pp. 231–258.

Khan, CEO Association of European Airlines. (2014). ECAC/EU Dialogue.

Klemperer, P. (2004), Auctions: Theory and Practice, Princeton University Press.

Koçoğlu, İ., İmamoğlu, S. Z., İnce, H., Keskin, H., (2011), The effect of supply chain integration on information sharing:Enhancing the supply chain performance, Procedia - Social and Behavioral Sciences, Volume 24, 2011, Pages 1630-1649

Krämer, Wolfgang and M. Langenbach-Belz. Approximate formulae for the delay in the queueing system GI/G/1. In Proceedings ITC, volume 8, pages 235-1, 1976.

Krishna, V. (2002), Auction Theory, Academic Press.

Lee H. L., Whang S., (2000), “Information sharing in a supply chain”, International Journal of Manufacturing Technology and Management, Vol.1 n°1, pp. 79-93;

Lee H.L., Padmanabhan V., Whang S., (1997), “Information distortion in a supply chain: the bullwhip effect”, Management Science, 546-558.

Letourneau, S.; Famili, F.; Matwin, S. (1999): Data mining to predict aircraft component replacement. In IEEE Intell. Syst. 14 (6), pp. 59–66.

Li, L., & Zhang, H. (2003). Confidentiality and Information Sharing in Supply Chain Coordination.

Li, S., Ragu-Nathan, B., Ragu-Nathan, T.S., Rao, S. S., (2006), The impact of supply chain management practices on competitive advantage and organizational performance, Omega, Volume 34, Issue 2, April 2006, Pages 107-124

Lindell, Yehuda; Pinkas, Benny (2009): Secure Multiparty Computation for Privacy-Preserving Data Mining. In The Journal of Privacy and Confidentiality 2009 (1), pp. 59–98.

Liu, E., & Kumar, A. (2003). Leveraging Information Sharing to Increase Supply chain configurability. . Kumar. “Leveraging Information Sharing to Increase Supply chain configurability.” In Proc. of the Twenty-Fourth International Conference on Information Systems, (p. 523-537). Seattle.

Mak, Ho-Yin, Ying Rong, and Jiawei Zhang. "Sequencing appointments for service systems using inventory approximations." Manufacturing & Service Operations Management 16.2 (2014): 251-262.

Malkhi, D., Nisan, N., Pinkas, B., and Sella, Y. (2004), Fairplay - A Secure Two-Party Computation System, in Proceedings of the 13th USENIX Security Symposium, pp. 287–302.

Manatsa P. R., McLaren T. S., (2008), “Information Sharing in a Supply Chain: Using Agency Theory to Guide the Design of Incentives”, An International Journal, Vol.9 n°1;

Manuj, I., & Mentzer, J. T. (2008). Global supply chain risk management strategies. International Journal of Physical Distribution & Logistics Management, 3(38), 192-223.

Mateen A., Chatterjee A.K., (2015), “Vendor managed inventory for single-vendor multi-retailer supply chains”, Decision Support Systems, 31-41.

McAfee, R., 1992. A Dominant Strategy Double Auction. Journal of Economic Theory, 56(2), pp.434–450.

Mentzer, J. (2001, January). Defining supply chain management. Journal of Business Logistics.

Milgrom, P. (2004), Putting Auction Theory to Work, Cambridge University Press.

Milgrom, P. R. and Weber, R. J. (1982), A theory of auctions and competitive bidding, Econometrica 50(5), 1089–1122.

Milgrom, P., and J. Robert (1992), Economics, Organization and Management, Prentice Hall, New Jersey.

Miyaji Atsuko and Rahman M. S., (2010), “ Privacy-Preserving Data Mining in Presence of Covert Adversaries”, Advanced Data Mining and Applications Lecture Notes in Computer Science, Vol. 6440, pp- 29-440;

Moulin H (1991), Axioms of Cooperative Decision Making, Cambridge University Press.

Mumbower, S., Garrow, L., & Higgins, M. (2014, August). Estimating flight-level price elasticities using online airline data: A first step toward integrating pricing, demand, and revenue optimization. Transportation Research Part A: Policy and Practice, 66, 196-212.

Murphy, Kevin P. (2012): Machine learning. A probabilistic perspective. Cambridge, Mass.: MIT Press (Adaptive computation and machine learning series).

Myerson, R.B. (1979), Incentives Compatibility and the Bargaining Problem, Econometrica 47, pp. 61–73.

Myerson, R.B. (1981), Optimal auction design, Mathematics of Operations Research 6, 58–73.

Myerson, R.B. and Satterthwaite, M.A. (1983), Efficient mechanisms for bilat- eral trading, Journal of Economic Theory 1(29), 265–281.

Naddor, Eliezer (1975): Optimal and Heuristic Decisions in Single- and Multi-Item Inventory Systems. In Management Science 21 (11), pp. 1234–1249.

Neely, Andy (2008): Exploring the financial consequences of the servitization of manufacturing. In Oper Manag Res 1 (2), pp. 103–118.

Nyaga, G. N., Whipple, J. M., Lynch, D. F., (2010), Examining supply chain relationships: Do buyer and supplier perspectives on collaborative relationships differ?, Journal of Operations Management, Volume 28, Issue 2, March 2010, Pages 101-114

Paillier, Pascal (1999): Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. In Jacques Stern (Ed.): Advances in Cryptology — EUROCRYPT ’99, vol. 1592. Berlin, Heidelberg: Springer Berlin Heidelberg (Lecture notes in computer science), pp. 223–238.

Park, S., & Ungson, G. (2001). Inter-firm rivalry and managerial complexity: a conceptual framework of alliance failure. Organization Science, 12, 37-53.

Parkes D (2005), Auction design with costly preference elicitation. Annals of Mathematics and Artificial Intelligence 44(3), 269–302.

Parkes D. and Kalagnanam J. (2005), Models for iterative multiattribute procure- ment auctions. Management Science 51, 435–451.

Peck, H. (2005). Drivers of supply chain vulnerability: an integrated framework. International Journal of Physical Distribution & Logistics Management, 35, 210-232.

Petersen, K. J., Handfield, R. B., Ragatz, G. L., (2005), Supplier integration into new product development: coordinating product, process and supply chain design, Journal of Operations Management, Volume 23, Issues 3–4, April 2005, Pages 371-388

Prajogo, D., Olhager, J., (2012), Supply chain integration and performance: The effects of long-term relationships, information technology and sharing, and logistics integration, International Journal of Production Economics, Volume 135, Issue 1, January 2012, Pages 514-522

Provost, Foster (2003): Tree Induction for Probability-Based Ranking. In Machine Learning 52 (3), pp. 199–215.

Rad R.H., Razmi J.,Sangari M.S., Ebrahimi Z.F., (2014), "Optimizing an integrated vendor-managed inventory system for a single-vendor two-buyer supply chain with determining weighting factor for vendor's ordering cost", International Journal of Production Economics, 295-308.

Rai, A., Patnayakuni, R., Seth, N., (2006), Firm Performance Impacts of Digitally Enabled Supply Chain Integration Capabilities, MIS Quarterly, Vol. 30, No. 2 (Jun., 2006), pp. 225-246

Ramanathan, U., Gunasekaran, A., (2014), Supply chain collaboration: Impact of success in long-term partnerships, International Journal of Production Economics, Volume 147, Part B, January 2014, Pages 252-259

Rao, S., & Goldsby, T. J. (2009). Supply chain risks: a review and typology. The International Journal of Logistics Management, 1(20), 97-123.

Rebolledo C., Nollet J., (2010), "Learning from suppliers in the aerospace industry", International Journal of Production Economics, Vol. 129 Issue 2, pp. 328-337;

Reliability Analysis Center, 2004, "Operational Availability Handbook";

Riley, J. G. and Samuelson, W. F. (1981), Optimal auctions, American Economic Review 71(3), 381–392.

Sari, K., (2008), "On the benefits of CPFR and VMI: A comparative simulation study", International Journal of Production Economics, 575-586.

Satterthwaite, M.A. and S.R. Williams (1989). The Rate of Convergence to Efficiency in the Buyer's Bid Double Auction as the Market Becomes Large. Review of Economic Studies 56: 477–498.

Satterthwaite, M.A. and S.R. Williams (2002), The Optimality of a Simple Market Mechanism, Econometrica, 70(5): 1841-1863.

Scarf, Herbert (1959): The optimality of (s,S) policies in the dynamic inventory problem. In Kenneth J. Arrow, Samuel Karlin, Patrick Suppes (Eds.): Mathematical methods in the social sciences. Proceedings of the first stanford symposium. Stanford, California: Stanford university press.

Schauenburg A. and Sinha A., (2008), “Fleet management system for advances helicopter platform-design & development requirements”, 26th Congress of International Council of the Aeronautical Sciences including the 8th AIAA Aviation Technology, Integration, and Operations (ATIO);

Schubert P. and Legner C., (2011), “B2B integration in global supply chains: An identification of technical integration scenarios”, Journal of Strategic Information Systems, Vol. 20 n° 3, pp. 250-267;

Shamir, A. (1979), How to share a secret, in Communications of the ACM 22, 11, pp. 612–613.

Shih S. C., Hsu S. H. Y., Zhu Z., Balasubramanian S. K., (2012), “Knowledge sharing-A key role in the downstream supply chain”, Information & Management, Vol. 49 Issue 2, pp. 70-80;

Siavash H. K., Partanen J., Holmstrom J., (2013), “Additive manufacturing in the spare parts supply chain”, Computer in Industry, Vol. 65 Issue 1, pp. 50-63;

Silva, Claudio Rogerio Negri and Reinaldo Morabito. Performance evaluation and capacity planning in a metallurgical job-shop system using open queueing network models. International Journal of Production Research, 47(23):6589{6609, 2009.

Simatupang T.M., and Sridharan R., (2002), “The collaborative supply chain”, International Journal of Logistics Management. Vol. 13 n°. 1, pp. 15-30;

Simatupang T.M., and Sridharan R., (2004), “A benchmarking scheme for supply chain collaboration”, Benchmarking: An International Journal, Vol. 11 n° 1, pp. 9-30;

Simatupang T.M., and Sridharan R., (2005) “The collaboration index: a measure for supply chain collaboration”, International Journal of Physical Distribution and Logistics Management, Vol. 35 n°. 1, pp. 46-62;

Smith D.J., (2013), “Power-by-the-hour: the role of technology in reshaping business strategy at Rolls-Royce”, Technology Analysis & Strategic Management, Vol. 25 Issue 8, pp. 987-1007;

Sodhi, M. S. (2012). Researchers' perspectives on supply chain risk management. Production and Operations Management, 1(21), 1-13.

Tang, C. (2006). Robust strategies for mitigating supply chain disruptions. International Journal of Logistics: Research and Applications, 9, 33-45.

Tang,O., Musa, S. N., (2011), Identifying risk issues and research advancements in supply chain risk management, International Journal of Production Economics, Volume 133, Issue 1, September 2011, Pages 25-34

Tzeng GH, Huang JJ (2011), Multiple Attribute Decision Making. Methods and Applications, Chapman and Hall.

Varian, H., (2007). Position auctions. International Journal of Industrial Organization, 25, 1163-1178

Vickrey, W. (1961), Counterspeculation, auctions, and competitive sealed tenders, The Journal of Finance 16, 8–37.

Vickrey, W. (1962), Auctions and bidding games, Princeton University Press pp. 15–27.

W. Operations Research, 9(3):383-387, 1961.

Whitt, Ward. The queueing network analyzer. Bell System Technical Journal, 62(9):2779{2815, 1983.

Whitt, Ward. Towards better multi-class parametric-decomposition approximations for open queueing networks. Annals of Operations Research, 48(3):221-248, 1994.

Wu, F., Yeniyurt, S., Kim, D., Cavusgil, S. T., (2006), The impact of information technology on supply chain capabilities and firm performance: A resource-based view, Industrial Marketing Management, Volume 35, Issue 4, May 2006, Pages 493-504

Xu K., and Dong Y., (2004), “Information gaming in demand collaboration and supply chain performance”, journal of business logistics, Vol. 25 Issue 1, pp. 121-144;

Yang N, Liao X, Huang WW (2014), Decision support for preference elicitation in multi-attribute electronic procurement auctions through an agent-based intermediary. Decision Support Systems 57,127–138.

Yao Y., Evers P.T., Dresner M.E., (2005), “Supply chain integration in vendor-managed inventory”, Decision Support Systems, 663-674.

Yi, Xun; Paulet, Russell; Bertino, Elisa (2014): Homomorphic Encryption and Applications. Cham: Springer International Publishing.

Yoon, K., 2001. The Modified Vickrey Double Auction. Journal of Economic Theory, 101, pp.572–584.

Zacharias, C. and Pinedo, M. (2014), Appointment Scheduling with No-Shows and Over-booking. Production and Operations Management, 23: 788–801. doi: 10.1111/poms.12065.

Zavanella L., Zanoni S., (2008), “A one-vendor multi-buyer integrated production-inventory model: The ‘Consignment Stock’ case”, International Journal of Production Economics, 225-232.

Zhou, H., Benton Jr., W.C., (2007), Supply chain practice and information sharing, Journal of Operations Management, Volume 25, Issue 6, November 2007, Pages 1348-1365, ISSN 0272-6963