On a method to improve your service BOMs within spare parts management

On a method to improve your service BOMs within spare parts management

(Accepted for publication in the International Journal of Production Economics, 1 August 2019)

J. Stip[a,b], G.J. Van Houtum[a,*]

[a] *Department of Industrial Engineering & Innovation Sciences, Eindhoven University of Technology, The Netherlands*
[b] *ASML, Veldhoven, The Netherlands*

Abstract

For advanced capital goods with high system availability requirements, it is common that all customers have service contracts with the Original Equipment Manufacturer (OEM). These service contracts include service level agreements on spare parts supply. The OEM operates a service network to support these logistic contracts. To determine spare parts stock levels the OEM needs to forecast spare parts demand. An important input for this forecast is the service Bill Of Material (BOM) per installed machine in the field, which specifies the applicable spare parts for a machine, and is usually derived from the machine configuration. Because of a growing installed base, increasing machine complexity, and an increasing number of machine variants, companies face a challenge in defining and maintaining machine configurations, which is why the service BOM is not always in line with the actual installed machine. An incorrect service BOM results in either a too low or a too high forecast for spare parts demand, and will result in under- or overstock.

In this paper we study the service BOMs at ASML, a large OEM in the semiconductor industry. We develop a method to generate alerts for possible errors. This method builds on multiple sources of machine information. Our method was tested in a pilot study, and found to be very effective. 95% of the generated alerts were correctly triggered and did result in actions that improved the service BOM. As a result, the method has been implemented by ASML. By this method, ASML reduced spare part non-availabilities by approximately 4-5 percent per year.

Keywords: Configuration Management, Spare Parts, Inventory Management, Forecasting, Data Science

---

*Corresponding author, Tel.: +31 40 247 5163, E-mail: g.j.v.houtum@tue.nl

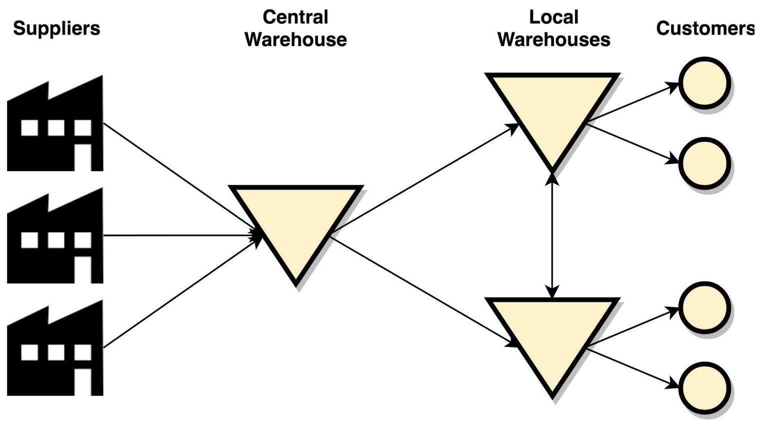

Figure 1: Graphical representation of the spare parts network

1. Introduction

In industries where equipment is complex and the uptime of equipment is highly important, customers often have service contracts with the OEM because they are better able to maintain the equipment. These contracts specify the aggregate machine downtime. In order to cost-efficiently meet these service contracts, the OEM needs to determine the optimal spare parts stock levels. One of the key enabling factors is the spare parts demand forecast, that is heavily dependent on the quality of the machine configuration data. In this paper we study the configuration data at ASML, an OEM of lithography equipment for the semiconductor industry. We come up with a method to identify possible errors in the service BOM in order to improve the demand forecast.

ASML is concerned with the planning of spare parts (and tools) in order to meet service level agreements with its customers. ASML operates a service network of central and local warehouses, located close to its customers. The central warehouses supply the local warehouses, but lateral transshipments between the local warehouses are also possible. The service network is depicted in Figure 1.

ASML is confronted with high system availability targets since system downtime can have a vast financial impact on the operations of the customer. These high targets, combined with the slow moving and intermittent characteristics of spare parts demand makes the positioning of inventory throughout the service network a challenging task (Huiskonen, 2001). Because of the low demand rates, the resulting local basestock levels are, for a large part of the assortment, either zero or one. One of the key inputs to determine spare parts stock levels is the forecast of spare parts demand.

Stochastic inventory management approaches for spare parts often make use of distributional assumptions for demands. The parameters of these distributions, typically the mean and variance, need to be estimated. This is referred to as parametric forecasting. Various methods to forecast spare parts demand are discussed in the literature. An overview is given by Boylan and Syntetos (2010), but many of these methods are focused on demand data only.

Since spare parts demand is generated by machines installed in the field that need to be repaired, the *installed base* is an important causal variable in forecasting spare parts demand. Taking this variable into account is discussed by various authors, e.g. Jalil et al. (2011), Minner (2011), Dekker et al. (2013), and Kim et al. (2017), and is referred to as installed base forecasting. A recent paper by van der Auweraer et al. (2019) gives an overview of the literature related to this topic. An important requirement for this approach is that companies keep track of their installed base (i.e. the location, number of machines, and machine types in use). Wagner and Lindemann (2008) state that many companies have only a 'cloudy view' of their installed base, which is why they have to resort to forecasting based on parts demand only. In the case of ASML, the company does have an accurate view of their installed base, and uses the installed base information to create demand rate forecasts on a part level per local warehouse.

In order to apply installed base forecasting, ASML uses the service BOM to determine on which machines a spare part is applicable. This service BOM is a list that specifies the applicable spare parts for a machine and is derived from the machine configuration. In the case of ASML, each installed machine has its own service BOM, since almost every machine has a unique configuration. Machine configurations can change over time due to maintenance or upgrades and the OEM needs to keep track of these changes. Dekker et al. (2013) find that in practice, the level of configuration control varies.

If machine configurations are not properly defined or maintained, the service BOM is not always in line with the actual machine, which affects the demand rate forecast. Puurunen et al. (2014) show that an inaccurate demand rate forecast leads to inaccurate stocking decisions within spare parts planning. A missing part in the service BOM of a machine results in a forecast which is too low and can lead to too little stock to meet the service level agreements from the service contracts. This is called understock. The opposite, overstocking, might occur when a part occurs in the service BOM, but does not occur in the machine. These effects are amplified when such an error holds for a whole group of machines. Unavailable stock and, more specifically, machine downtime, will result in lost revenues and customer dissatisfaction (Driessen et al., 2015). Excess stock will lead to a high operational cost for the OEM because of unnecessary holding costs and risk of obsolescence.

To the best of our knowledge, the literature on configuration management and spare parts planning is scarce. Some articles on installed base forecasting address the issue of inaccurate installed base information, but do not go into the level of individual machine configurations. Pintelon et al. (1999) already indicate that: "configuration management is of course a basic need for sound spare parts management" in the context of complex equipment. A recent article by Boone et al. (2017) highlights planning and forecasting, as well as configuration management, as two of the most critical downstream challenges faced by spare parts decision makers that show potential for big data applications. According to Li et al. (2015), "big data" is generated during a product's lifecycle, such as maintenance and failure information, maintenance support information, and maintenance history information. Andersson and Jonsson (2018) also indicate the value of using product-in-use data for causal-based forecasting in spare parts supply chains.

The main contributions of this paper are as follows:

- In the growing body of literature on installed base forecasting, we are the first to study the problem of configuration data errors and their effect on forecasts and spare parts stocks.
- We develop a method to identify these configuration errors, using multiple sources of configuration information such as the engineering BOM, maintenance procedures, and historical spare parts usage. This method is new and generic. It is relevant for all cases where installed base forecasting is applied.
- The method proved to be very effective in a pilot study at ASML, where we showed that 95% of the generated alerts were found to be correct.
- The method was successfully implemented at ASML, and resulted in a structural reduction of 4-5% of all spare parts non-availabilities (and thereby long machine downtimes).

The remainder of this paper is organized as follows. Section 2 describes the setting, the possible service BOM errors, and their effect. Section 3 describes the principles of configuration management and how it is applied within ASML. The method we developed is described in Section 4. Our method was tested in a pilot study that is described in Section 5. Finally, Section 6 presents the conclusions.

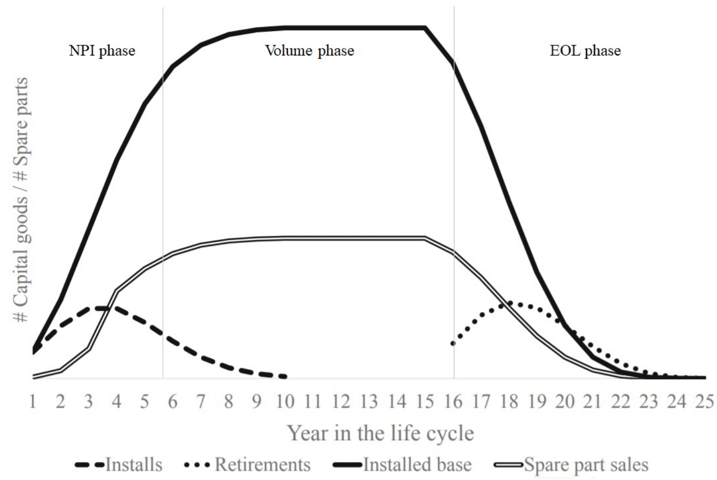

Figure 2: Life cycle of the installed base (Arts et al., 2019)

2. Setting

2.1. Installed base

For the purpose of this paper, we introduce the following notation. Let B denote the total installed base (e.g. the set of machines installed at customers), where $b \in B$ refers to a single machine *instance*. The total set of machines can be divided into subsets for different machine types. We introduce the subset $B_A \subset B$ to denote the subset of all machines of machine type A.

It is important to mention here that the setting for planning and forecasting is applicable for all machine types which are in the volume phase. This is illustrated by Figure 2, which shows the life cycle for the installed base of a machine type. After the ramp-up of installs, it can be seen that the spare parts demand intensity is quite stable. Planning and forecasting spare parts demand for the installed base in the new product introduction (NPI) or end of life (EOL) phase requires a different approach, which will not be discussed in this paper.

2.2. Spare parts planning model

In a bi-annual planning cycle, the worldwide basestock levels are (re)calculated. ASML uses the spare parts inventory model of van Houtum and Kranenburg (2015) in which demand processes are assumed to follow a Poisson process. This requires only one parameter to be estimated. Each machine $b \in B$ consists of multiple parts (also called SKUs). Set I denotes the set of parts, numbered $i = 1, ..., |I|$. Demands for each part $i \in I$ and machine $b \in B$ are assumed to follow a Poisson process with a constant rate $m_{i,b}$. Each machine $b \in B$ is assigned to exactly one local warehouse $j \in J$, where J denotes the set of all local warehouses. A local warehouse serves multiple machines.

2.3. Forecasting process per part i

At the tactical planning level, estimates for the demand rates are generated by a forecasting process. Each month t', the realized demand rate per machine is calculated by dividing the observed usage over the total applicable installed base $u_i(t')/B_i(t')$, where $u_i(t')$ is the world-wide demand for part i in month t', and $B_i(t')$ represents the number of machines on which part i is installed in month t'. We refer to this ratio as $m_i(t')$ and use this ratio as the basis for the demand rate forecasts. Two assumptions need to hold:

1. All machines with part i have the same number of units of part i built in.
2. The demand rate per machine is the same for all machines that contain part i.

These assumptions are realistic for the case of ASML, and reasonable in the case of installed base forecasting in the volume phase. Relaxing these assumptions will not have any consequences for the method that is developed later on.

The demand rate calculation is further differentiated towards regions or product groups if strong deviations are observed. A (weighted) moving average is applied to calculate the forecasted demand rate $f_i(t'+1, ..., t'+n)$ from the past demand rate observations $m_i(t')$.

As a next step, the demand rate is aggregated to the local warehouse level by summing over all machines assigned to local warehouse j at time t. This way, a future increase or decrease in installed base can also be taken into account.

It is important to notice that the applicable installed base is an important factor in the calculation of the total usage rate $m_i(t')$, and in the calculation of the forecasted usage rate at the local warehouse level. To determine the applicable installed base, ASML uses the service BOM. If the service BOM is not in line with the actual machine, various errors can be observed. These errors are discussed in the following section.

2.4. Service BOM errors

The problem of an inaccurate service BOM can be split into underspecification and overspecification. Underspecification means that the service BOM for a machine misses parts that actually need to be planned. Overspecification means that the service BOM contains parts that do not need to be planned. Under- and overspecification can occur at the same time. Furthermore, we

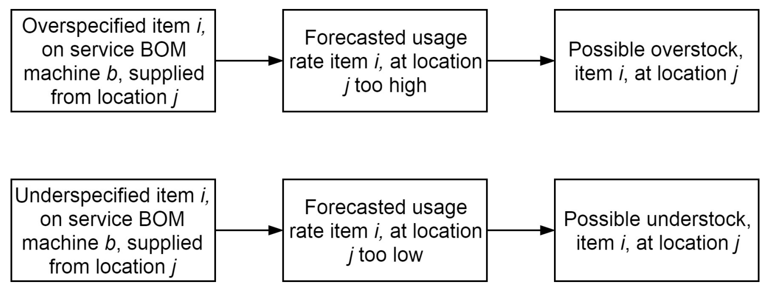

Figure 3: Instance errors

can distinguish between instance errors and type errors. Instance errors are related to only one machine $b \in B$, where type errors are related to a machine type A and the error is applicable to a group of machines $B_A \subset B$.

For instance errors, the effect is quite linear. Underspecification results in possible understock, and overspecification results in possible overstock. The causal relationships are depicted in Figure 3. An additional effect is that overspecification also leads to a slightly lower forecast for part i at other locations, just as underspecification leads to a slightly higher forecast. However, for instance errors this additional effect is very small and therefore negligible. The effect becomes stronger for type errors.

For type errors, the same relationships hold, but the additional effect needs to be taken into account. This effect originates from a structural forecast error since we assume that each part i has the same failure rate. Therefore, *under*specification for machine type A can lead to worldwide *over*stocking for other machine types, and vice versa. The effects are depicted in Figure 4.

3. Configuration Management

In this section we will describe the basic principles of configuration management as applied within ASML. Configuration management is used by many companies to establish and maintain all product information needed to support the control of the *design*, *manufacture*, and *support* phases of a product. A simple definition is given by Daniels (1985): "Configuration Management is a management tool that defines the product, then controls the changes to that definition". Configuration management is a complete life cycle approach that is applicable to much more than just hardware. According to ISO:10007 (1997), configuration management applies to all kinds of product information such as: hardware, software, processed materials, services, and related technical documentation. Kidd and Burgess (2004) discuss the four basic configuration management

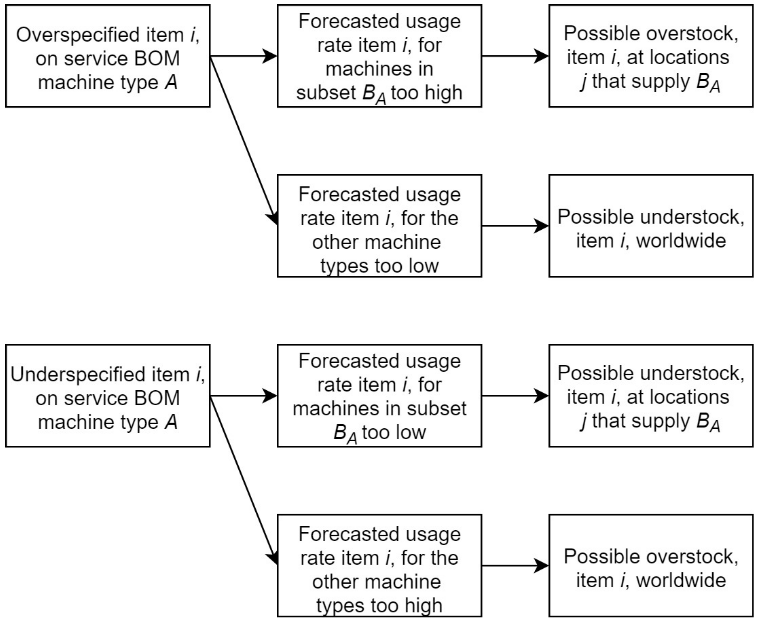

Figure 4: Type errors, additional effect

functions:

1. Configuration identification
2. Configuration control
3. Configuration status accounting
4. Configuration audit

3.1. Configuration definitions

In practice, different “baselines” of the product are defined which are covered by the basic configuration management functions. A baseline defines the product and its parts. Three major states can at least be distinguished, which correspond to the design, manufacture, and support phase of a product. Similar states are also found in the steam turbine business (Müller et al., 2012). Figure 5 provides an overview of these three configuration stages.

1. Configuration As Designed/Engineered (CAE)
2. Configuration As Built (CAB)
3. Configuration As Maintained (CAM)

The product life cycle starts with the development and engineering of a certain machine type, which results in the CAE, and is also referred to as the engineering BOM. As a next step, the factory

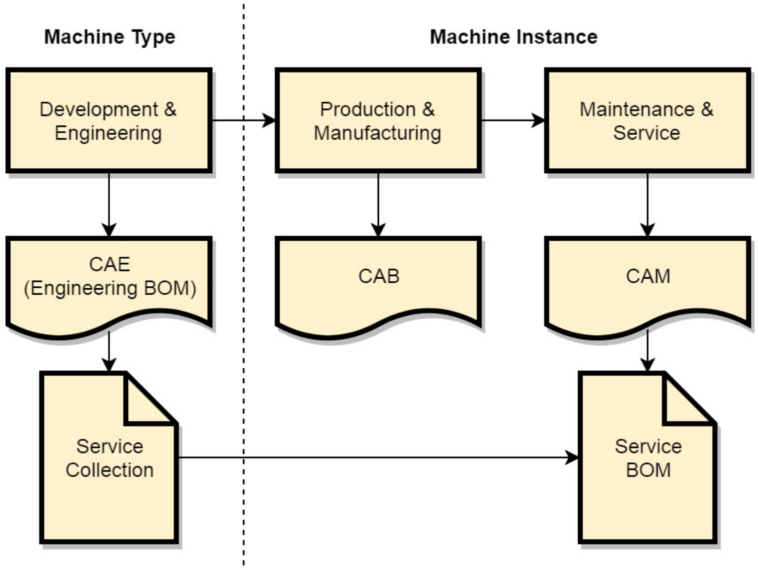

Figure 5: Configuration definitions

starts to build the machines that may include some customer specific options or requirements. This results in a CAB which is unique per new machine instance, and is the starting point for the CAM. From the moment the machine is installed at the customer it can be subject to change due to modificative maintenance or machine upgrades, resulting in a CAM per machine instance. The CAM must be updated with every relevant change to the machine made.

3.2. Configuration management at ASML

At ASML, during the development and engineering stage it is decided, together with the maintenance department, which spare parts are needed to maintain the machine, resulting in a *service collection* for the machine type. Spare parts in the service collection are parts that have a potential to fail or degrade and need to be replaced in the field. The service collection only contains the parts or assemblies at the level they will be replaced, also referred to as Line Replaceable Units (LRUs). Together with the definition of the spare parts collection, maintenance procedures are created that provide service engineers the information needed for maintaining the machines.

The CAM is updated with every change to the machine made, using the built-in and built-out information provided by service engineers on a service order. Changes can be very simple, when a defect part is built-out and a new part is built-in, but can also be very complex, when hardware upgrades are executed to enhance the functionality or performance of a machine. For these complex upgrades hundreds of parts can be built-in and built-out, which makes the registration a challenging

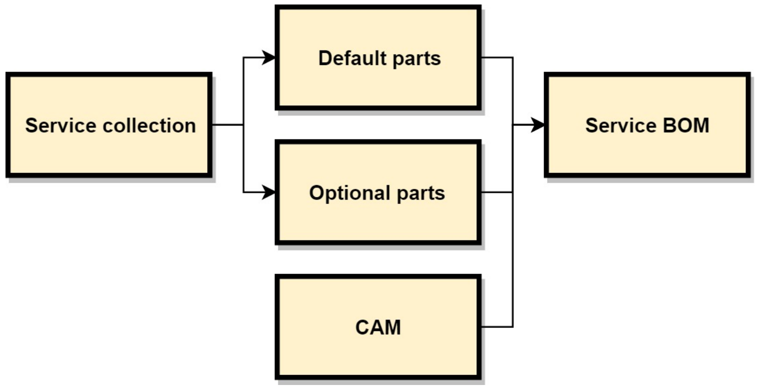

Figure 6: Process to create the service BOM

task.

The service collection and the CAM now form the basis for the service BOM of a machine instance. The service collection consists of two types of parts, default and optional parts. The majority of the spare parts is applicable to all machines within the machine type, referred to as the *default parts*. Due to several reasons (e.g. engineering changes, software updates, commercial upgrades) the configuration of each machine is unique. Therefore, some spare parts are not applicable to all machines within the machine type, which leads to the concept of so-called *optional parts*. These parts only apply to a limited group of machines within the machine type.

To determine which optional spare parts are applicable to the machine instance, the equipment structure of the machine (CAM) is matched with the list of optional parts. The matching parts will, together with the default parts, make up the service BOM of the machine. This is displayed in Figure 6.

3.3. Root causes for errors

Two main causes can lead to a service BOM that is not in line with the actual machine:

1. The CAM might show under- or overspecification. This can be due to the fact that the starting point (CAB) was already inaccurate because in the past, configuration management processes might not have been executed as good as they should have been. The quality of the CAB is the starting point for the quality of the CAM. Subsequently, inaccurate change documentation by engineers can also lead to errors in the CAM. This can be due to missing service orders, or service orders that contain wrong part numbers or machine numbers, an issue that is also discussed by Hodkiewicz and Ho (2016). Finally, inaccurate change registration by the configuration management department may also lead to an inaccurate CAM.

2. The service collections might show under- or overspecification. At the moment of defining the service collection, the starting point is often the service collection for a similar machine type. By adding new parts, and removing parts that are not needed, the service collection for the new machine type is created. This can be one of the sources of error. Also, during the lifetime of a machine type, the service collections are subject to change when new or alternative parts are introduced (called: engineering changes), and old parts become obsolete. This can be another source of error.

The use of the machine type as a starting point for the service collections, in combination with an increasing number and complexity of options and upgrades, leads to a growing number of optional parts in the service collection, which increases the risk of errors, especially in combination with an inaccurate CAM. Because the configuration information is dynamic (i.e. changing over time) and is not necessarily maintained centrally (because different organizational functions use different views), this increases the risk for the OEM that the configuration information becomes incorrect or incomplete. The result is an incorrect service BOM that does not reflect the actual machine.

4. Method

To overcome the challenges described in the previous sections, a method was developed that is able to generate alerts on possible errors in the configuration data. The method builds on multiple sources of configuration information. During the support phase of a product, various sources of information are available that contain information about the machine configuration and the applicable spare parts. Besides the service collection, the CAM, and the service BOM, company databases often contain other relevant data such as the CAE, maintenance procedures, and historical spare parts usage. The CAE contains product information from the development and engineering phase, where the maintenance procedures contain information about which part can be used where. Historical spare parts usage provides information on what was actually used on a machine. In an ideal world, configuration management functions are executed such that all these information sources are in accordance with each other. In reality however, differences between these sources of information exist. We will refer to these differences as “mismatches”. Some of these mismatches are particularly interesting to investigate, because they indicate under- or overspecification in the service BOM inputs (service collections or CAM) and are therefore likely to result in service BOM errors.

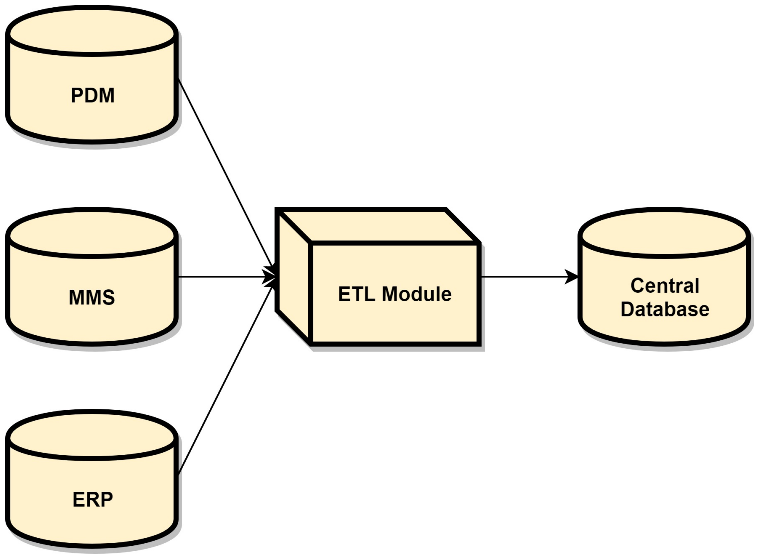

Figure 7: Data integration scheme

4.1. Data integration

To automate the identification of the mismatches, we first need to integrate the data from the different data sources in order to create a unified view. Engineering related data, such as the CAE, is usually stored in a product data management (PDM) system, where the maintenance procedures might be stored in a different maintenance management system (MMS), and the data related to production and planning (e.g. service collections, CAM, service BOM, and historical spare part usage) is stored in an enterprise resource planning (ERP) system. The data needs to be extracted, transformed, and loaded (ETL) into a central database, which is shown in Figure 7. From this central database, alerts can be generated on the improvement candidates.

The central elements in this database are the set of machines, B, and the set of parts, I. Based on the characteristics of each machine $b \in B$, we can obtain from the various data sources the applicable spare parts for that machine. As output we can generate per machine, a list of the applicable spare parts and their presence in one or more data sources. An example of the output is displayed in Table 1. The table shows only two parts for one machine, in reality the list is much longer and can contain thousands of parts per machine, for thousands of machines. We can see that part ABC is in the service BOM of machine 1234, because it is an optional part in the service collection of the machine type, and the part is found in the CAM of machine 1234. Part ABC is also found in the CAE and the maintenance procedures. The part has never been used on machine 1234. For this part and this machine, we see no mismatch in the data. On the contrary, part

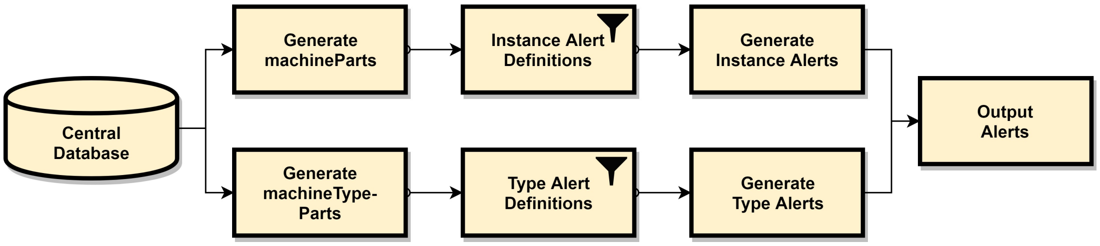

Figure 8: Alert functionality

XYZ is not in the service BOM of machine 1234 because it is not in the CAM of the machine. However, part XYZ is found in the CAE, the maintenance procedures, and has also been used on the machine. This gives a strong indication that part XYZ is a spare part that is underspecified in the CAM of this machine. The next subsection describes how alerts can be generated for such a mismatch.

Table 1: Example of output

Machine	Spare Part	Service BOM	Service collection	Optional	CAM	CAE	Maintenance Procedures	Usage
1234	ABC	1	1	1	1	1	1	0
...	...	..	...	...	...	...	...	...
1234	XYZ	0	1	1	0	1	1	1

4.2. Generating the alerts

Figure 8 provides a general overview of the alert functionality.

The starting point is the central database, and a list of alert definitions. These alert definitions can best be seen as filters for the dataset. From the central database, we can generate from each data source a set of applicable spare parts for a machine. These parts are merged in a new set, defined as “machineParts”. This set now contains all spare parts that are found for the machine in at least one data source. Each machinePart in this set refers to the unique combination of a machine and a spare part. Furthermore, the machinePart stores for each of the data sources an integer value which is 1 if the material is present, and 0 if the material is not present. For usage, the integer value stores the total amount of usage on the machine.

Next to the machineParts, so-called “machineTypeParts” will be created, which store the same properties as a machinePart, but now the values are summed over all the machines within a machine type. The machineParts and machineTypeParts are needed to generate the alerts. The split between machineParts and machineTypeParts is made because instance alerts are triggered on machine instance level, to identify under- or overspecification in the CAM where type alerts are triggered on the machine type level to identify under- and overspecification in the service collection of the machine type.

As a next step, the machineParts and machineTypeParts are subsequently matched with the prespecified alert definitions, and an alert will be triggered on the machineParts and machineTypeParts that meet the definitions. These definitions are discussed in the following subsection.

4.3. Instance alert definitions

One example of an instance alert definition is specified in Table 2.

1. The spare part is not found in the service BOM or CAM, but is an optional part in the service collection, and has been used on the machine in the past. This alert will be triggered on machinePart level.

Using this alert definition, we are able to find parts that are possibly underspecified in the CAM of the machine instance.

Table 2: Instance alert - underspecification

Id:	**Machine**	**Spare Part**	**Service BOM**	**Service collection**	**Optional**	**CAM**	**CAE**	**Maintenance Procedures**	**Usage**
Instance 1	....	....	0	1	1	0	*	*	>0

In general, overspecification in the CAM of the machine is harder to detect, because there is no usage trigger. A part which is never used on a machine, could still belong to the machine and its service BOM. In Table 3 we give an example of an instance alert to detect overspecification.

2. The spare part is in the service BOM, because it is an optional part and is found in the CAM. The fact that the optional part is not in the service collection, indicates that the part has been removed from the collection and should no longer be used.

Table 3: Instance alert - overspecification

Id:	**Machine**	**Spare Part**	**Service BOM**	**Service collection**	**Optional**	**CAM**	**CAE**	**Maintenance Procedures**	**Usage**
Instance 2	....	....	1	0	1	1	*	*	0

4.4. Type alert definitions

A number of type alert definitions for underspecification are specified in Table 4.

1. The spare part is found in the CAM, the CAE, and in the maintenance procedures, but is not defined as a spare part in the service collection and will therefore not be in the service BOM.
2. The spare part is found in the CAM and the CAE, but not in the service collection.
3. The spare part is found in the CAM and the maintenance procedures, but not in the service collection.
4. The spare part is found in the CAE and in the maintenance procedures, but not in the service collection.
5. The spare part does not meet definition 1,2,3, or 4, but is used three or more times on the machine type.

These alerts describe all possible scenarios where the spare part is at least found in two data sources, but is missing in the service collection. This can occur for example when service parts are defined at a later stage than the service collection, as a consequence of new functionality or new part versions being introduced. It can also happen that during the lifetime of a machine, new maintenance procedures are created that require spare parts which are not in the service collection.

These alerts will be triggered on machine type level. Using these alert definitions, we are able to find parts that are underspecified in the service collection.

Table 4: Type alerts - underspecification

Id:	**Machine Type**	**Spare Part**	**Service BOM**	**Service collection**	**Optional**	**CAM**	**CAE**	**Maintenance Procedures**	**Usage**
Type 1	....	....	0	0	0	1	1	1	*
Type 2	....	....	0	0	0	1	1	0	*
Type 3	....	....	0	0	0	1	0	1	*
Type 4	....	....	0	0	0	0	1	1	*
Type 5	....	....	0	0	0	*	*	*	≥ 3

Another example of a type alert definition is specified in Table 5. This concerns an example of *over*specification.

6. The spare part is *not* found in the CAM, the CAE, or in the maintenance procedures, and

has never been used on a machine of this type, but is defined as a spare part in the service collection and will therefore be in the service BOM.

This alert is also triggered on machine type level and enables to find parts that are overspecified in the service collection.

Table 5: Type alert - overspecification

Id:	**Machine Type**	**Spare Part**	**Service BOM**	**Service collection**	**Optional**	**CAM**	**CAE**	**Maintenance Procedures**	**Usage**
Type 6	....	....	1	1	*	0	0	0	0

In this section we provided multiple alert definitions, and showed how they can be used to identify under- or overspecification in the data sources. Based on these principles, even more definitions can be specified or customized to the data sources that are available.

5. Pilot study

Our method was tested in a pilot study at ASML. In the pilot study we tested, with a subset of the alerts, whether the generated alerts were correct. The focus of the pilot study was on underspecification alerts, since these will most likely result in spare part non-availabilities. The pilot was executed for a machine type that constitutes a significant proportion of the ASML's installed base.

Table 6 summarizes the results of the pilot study. We give the results for "Instance 1" and the aggregated results for "Types 1-5". For Instance 1, 31 alerts were triggered and validated. For Types 1-5, in total 10 alerts were triggered and validated (2, 2, 1, 3, and 2 alerts, respectively). These numbers are listed in the second column of Table 6. Validation of an alert means that the alert is studied in further detail by employees who are responsible for the different types of configuration data, that they assess whether there is a real mismatch, and, if so, they fix the mismatch. This can be a maintenance engineer who is familiar with the procedures, or a development engineer who is familiar with the machine functionality and can make this judgment. Physical inspection would in some cases provide the final answer, but this is often not allowed by customers as long as there is no maintenance or upgrade need to physically open up the machine. The third column of the table shows how many alerts were correct and did lead to an adjustment in the data. On average, the success rate of the alerts was 95%, which means that the generated alerts were to a very large extent correctly triggered, and did result in actions that improved the service BOM.

The unsuccessful alerts provided interesting insights as well. In one case, it was learned that the spare part was not in the service collection because it was no longer needed after all machines were upgraded. In another case the part was not in the service collection because an improved version of the part was already in the collection that should have been used instead. Additionally, it was learned that the unsuccessful alerts largely related to registration errors (i.e. part codes booked on wrong machines) or to exceptions that were logged in textual fields which could not be interpreted digitally. An exception list and handling mechanism was created to prevent the tool from triggering these alerts again. The exception list contains the machinePart or machineTypePart combinations that should be excluded, the handling mechanism prevents the alerts on these combinations to be triggered. Please note that the results in Table 6 are reported before the introduction of this exception list and handling mechanism.

Table 6: Pilot study results

Alert id	**Alerts validated**	**Alerts correct**	**Success rate**
Instance 1	31	30	0.97
Type 1,2,3,4,5	10	8	0.80
Total:	41	38	0.95

5.1. Implementation

After a successful pilot study, the method was developed in a JAVA-based application and implemented by the organization in June 2018. The application was embedded in the monthly forecasting cycle, in order to pro-actively identify issues and take action before they can lead to problems. The implementation process was executed similar to the steps taken in the pilot study. We now generated alerts for the full installed base, after which each alert was validated (investigation plus a possible follow-up action to fix a mismatch) by a configuration expert or service engineer. This validation is done based on information systems and expert judgment, because opening up a machine for a physical inspection would consume too much valuable machine up-time.

Still not all alerts could be validated at once, since it takes time and effort to validate each alert. This could be a few minutes for an 'easy' alert, up to a couple of hours if multiple experts need to be involved. Therefore, a trade-off between the time invested and occurrence of the alerts was made and actions were taken on the alerts that were most promising regarding their impact.

After validation, the service BOMs were updated and the failure rates were recalculated. Before and after implementation of the tool, the failure rate per part i was monitored to quantify the impact on the forecast. The implementation results are shown in Table 7. The total failure rate, calculated base stock levels, total corresponding inventory value, and expected number of backorders before and after implementation are reported. For confidentiality reasons, the numbers before implementation are normalized to 1.

The total failure rate is calculated before and after the service BOM changes. Because the majority of the alerts concerned underspecified items on the service BOM of the machine type, parts were predominantly added to the service BOM. The total failure rate after implementation decreased as a result of the additional effect for type errors that was described in Section 2.4 and depicted in Figure 4. The optimal base stock levels are calculated as a consequence of the old and new failure rates. It can be observed that, although the total failure rate decreased, the base stock levels increased in order to prevent understock at locations that supply the machine types for which the items were underspecified. Looking at the total corresponding inventory value, we can see an increase of 4.6 percent, which is larger than the 1.2 percent base stock level increase. This indicates that the increase in base stock levels occurred relatively more often for the expensive than cheap parts. The expected number of local non-availabilities has been calculated via ASML's planning software by taking the optimal base stock levels for both scenarios and calculating the performance using the new failure rates in both scenarios.

The calculated decrease in spare part non-availabilities (and thereby long machine downtimes) is 4-5 percent per year. Preventing long downtimes is very valuable, since one hour of machine downtime can cost a customer up to 100,000 EUR. As another advantage, ASML reported an increased trust in forecasting and planning.

Table 7: Implementation results

Measure	**Before implementation**	**After implementation**
Failure rate	1	0.991
Base stock levels	1	1.012
Total corresponding inventory value	1	1.046
Expected local non-availabilities	1	0.954

6. Conclusions

In this paper we have investigated the service BOM at ASML, a large OEM in the semiconductor industry. This service BOM is an important input for the forecast of spare parts demand when installed base forecasting is applied. We discussed the root causes for mismatches between the service BOM and the actual machine, and described the effects on spare parts planning. We developed a method to identify these mismatches, using multiple sources of configuration information, such as: the CAE, maintenance procedures, and historical spare part usage. By integrating all information from these data sources, and comparing them on a part basis for the individual machine, or machine types, the differences enable us to identify under- or overspecification in the data sources that are used to create the service BOM (service collections and CAM). Our method automates this identification by generating alerts on parts that meet a set of predefined alert definitions. The alerts were tested in a pilot study, and proved to be very effective. By implementing the method and resolving the alerts, ASML was able to improve the quality of the service BOM and the forecast. This will not only prevent under- and overstocking, but will also lead to an increased trust in forecast and planning.

One of the limitations of the method is that overspecification is much harder to detect than underspecification. This is caused by the fact that there are no demand observations available in the case of overspecification, and physical inspection of the machine is the only way to know for sure. Another limitation of the method is that validation of the alerts by subject matter experts can be a cumbersome task, especially when a large number of alerts is generated. This is also reflected in the pilot study, where the number of validated type alerts is much smaller than the number of validated instance alerts. This was due to the fact that there were time constraints on the people that needed to validate these type alerts. We therefore advise to introduce a prioritization mechanism in order to investigate the most promising alerts first.

Companies can learn from this study that installed base forecasting depends heavily on the quality of configuration data. Furthermore it can be learned that configuration errors can be identified using the method as proposed by the authors. It can be generalized to any setting where a company applies installed base forecasting and has multiple data sources available that contain configuration information.

References

Andersson, J., Jonsson, P., 2018. Big data in spare parts supply chains: The potential of using product-in-use data in aftermarket demand planning. International Journal of Physical Distribution & Logistics Management 48, 524–544.

Arts, J., Basten, R., Van Houtum, G.J., 2019. Maintenance service logistics. In: Zijm, H., Klumpp, M., Regattieri, A., Heragu, S. (eds.), Operations, Logistics and Supply Chain Management, Springer , 493–517.

van der Auweraer, S., Boute, R.N., Syntetos, A.A., 2019. Forecasting spare part demand with installed base information: A review. International Journal of Forecasting 35, 181 – 196.

Boone, C.A., Skipper, J.B., Hazen, B.T., 2017. A framework for investigating the role of big data in service parts management. Journal of Cleaner Production 153, 687 – 691.

Boylan, J.E., Syntetos, A.A., 2010. Spare parts management: a review of forecasting research and extensions. IMA Journal of Management Mathematics 21, 227–237.

Daniels, M., 1985. Principles of configuration management. Rockville, Md.: Advanced Applications Consultants.

Dekker, R., Çerağ Pinçe, Zuidwijk, R., Jalil, M.N., 2013. On the use of installed base information for spare parts logistics: A review of ideas and industry practice. International Journal of Production Economics 143, 536 – 545.

Driessen, M., Arts, J., van Houtum, G.J., Rustenburg, J.W., Huisman, B., 2015. Maintenance spare parts planning and control: a framework for control and agenda for future research. Production Planning & Control 26, 407–426.

Hodkiewicz, M., Ho, M.T.W., 2016. Cleaning historical maintenance work order data for reliability analysis. Journal of Quality in Maintenance Engineering 22, 146–163.

van Houtum, G.J., Kranenburg, B., 2015. Spare parts inventory control under system availability constraints. New York: Springer.

Huiskonen, J., 2001. Maintenance spare parts logistics: Special characteristics and strategic choices. International Journal of Production Economics 71, 125 – 133.

ISO:10007, 1997. Quality management - Guidelines for configuration management. ISO 10007.

Jalil, M.N., Zuidwijk, R.A., Fleischmann, M., van Nunen, J.A.E.E., 2011. Spare parts logistics and installed base information. Journal of the Operational Research Society 62, 442–457.

Kidd, C., Burgess, T.F., 2004. Managing configurations and data for effective project management. In: Morris, P.W.G., Pinto, J.K. (eds.), The Wiley Guide to Managing Projects, John Wiley and Sons, Inc. , 498–513.

Kim, T.Y., Dekker, R., Heij, C., 2017. Spare part demand forecasting for consumer goods using installed base information. Computers & Industrial Engineering 103, 201 – 215.

Li, J., Tao, F., Cheng, Y., Zhao, L., 2015. Big data in product lifecycle management. The International Journal of Advanced Manufacturing Technology 81, 667–684.

Minner, S., 2011. Forecasting and inventory management for spare parts: An installed base approach. In: Nezih, A., Litteral, L.A. (eds.), Service Parts Management: Demand Forecasting and Inventory Control, Springer, London , 157–169.

Müller, P., Muschiol, M., Stark, R., 2012. Plm-based service data management in steam turbine business, in: Rivest, L., Bouras, A., Louhichi, B. (Eds.), Product Lifecycle Management. Towards Knowledge-Rich Enterprises, Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 170–181.

Pintelon, L., Preez, N.D., Puyvelde, F.V., 1999. Information technology: opportunities for maintenance management. Journal of Quality in Maintenance Engineering 5, 9–24.

Puurunen, A., Majava, J., Kess, P., 2014. Exploring incomplete information in maintenance materials inventory optimization. Industrial Management & Data Systems 114, 144–158.

Wagner, S., Lindemann, E., 2008. A case study-based analysis of spare parts management in the engineering industry. Production Planning & Control 19, 397–407.