



**HAL**  
open science

## Closing the Lifecycle Loop with Installed Base Products

Martin Imboden, Bernhard Fradl, Felix Nyffenegger

► **To cite this version:**

Martin Imboden, Bernhard Fradl, Felix Nyffenegger. Closing the Lifecycle Loop with Installed Base Products. 15th IFIP International Conference on Product Lifecycle Management (PLM), Jul 2018, Turin, Italy. pp.348-357, 10.1007/978-3-030-01614-2\_32 . hal-02075581

**HAL Id: hal-02075581**

**<https://inria.hal.science/hal-02075581v1>**

Submitted on 21 Mar 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

# Closing the Lifecycle Loop with Installed Base Products

Martin Imboden, Bernhard Fradl, Felix Nyffenegger

HSR, Rapperswil, Switzerland  
imbodenmartin@hotmail.com  
bernhard.fradl@hsr.ch  
felix.nyffenegger@hsr.ch

**Abstract.** Industry trends indicate that in the future, more systems will be rented then sold. The customer rents production capacity and demands a guaranteed high operational readiness, which is hard to achieve with conventional maintenance. Downtimes can never be completely ruled out. In order to solve this problem and guarantee high operational reliability, the predictive maintenance approach is widely discussed: By means of indicators, measured by sensors, a potential problem can be identified before it actually occurs. The application of this concept to new products gets a lot of attention in many areas. However, industrial products such as machines or plants are long living objects. It seems interesting to extend these new technologies and eventually new services business models to the installed base, too.

This paper explores and demonstrates, what it takes to upgrade an operating product in its mid-of-life stage to a smart, connected products with predictive maintenance capabilities. The showcase consists of a jointed-arm industrial robot with six axes. The robot's motions will be retraced in order to determine the state and position of the robot and finally predict, what the robot is about to do. To achieve this, the robot was made IoT-capable by attachment of sensors which communicate directly to a cloud database. Finally, a trained machine learning model allows predication on the robots' behavior. On the way to the final result, many little lessons about sensing, protocols, the right place to process or tag data in the IoT stack had to learnt and will be shared in this publication.

**Keywords.** Closed-Loop PLM, Predictive Maintenance, Machine Learning, IoT, Installed Base

## 1 Introduction

Within the framework of the "Lifecycle Lab", a virtual world is being set up at HSR (university of applied science of Rapperswil) in order to discuss various concepts in the field of digitization (Industry 4.0, Industry 2025) and Closed Loop Product Lifecycle Management [1]. This is illustrated by the example of two virtual companies called "Sortic" and "DropKick". **Sortic** manufactures automation solutions, which are sold in accordance with the "machine as a service" business model. The customer does not buy the machine itself, but only "sorting capacity". This includes

98% operational readiness, defined response times and services for upgrades and retrofitting of the system. The functional model of the machine was developed as a pick and place machine based on Lego and Arduino (electricity and software).

**DropKick** is a user of the Sortic system and uses it to complete its business model. The company is a delivery service startup that is strongly digitized. The interaction between the two companies is the basis for explaining and researching a digital ecosystem and closed-loop PLM. The ideas around the closed loop lifecycle will be put into practice on a real machine, since simulated systems only show the real world to a limited extent. This “installed base” is an example of a system in operation. It consists of a machine with a jointed-arm industrial robot that should be retrofitted or upgraded so that it becomes IoT-capable. This means the machine needs to be outfitted with sensors, which communicate (in this case via Bluetooth) with a microcomputer, who then communicates with a database in the cloud. Using machine learning and cloud computing correlations and trends can be identified. Based on this, predictions on the robot’s motions are made. [1]

Background of this project is Industry 4.0, which means to interlink Industrial production with modern information and communication technology. Intelligent and digitally networked systems are the technical basis for this. With their help, a largely self-organized production is possible. This means that people, machines, plants, logistics and even products communicate directly with each other. Interconnection should make it possible to optimize the entire supply chain in all phases of the product lifecycle and not just one single production step in a single phase. [2]

Industry 4.0 consists of four basic principles that support companies in identifying and implementing industry 4.0 scenarios. [3]

- **Interconnection:** machines, devices, sensors and people can connect and communicate with each other via the internet of things or the normal internet (internet of people).
- **Transparency of information:** In order to create a virtual image of the real world, information systems of digital factory models are extended with sensor data.
- **Technical assistance:** People are physically supported in exhausting, unpleasant or dangerous tasks. Assistance systems also help people with aggregated, visualized and understandable information so that they can make informed decisions and solve problems more quickly.
- **Decentralized decisions:** Industry 4.0 systems are capable of making autonomous decisions and thus carrying out tasks independently. In exceptional cases, tasks are transferred to a higher authority, for example in accidents.

## 2 Related Work

In this paper, machine learning is used to monitor how a system behaves and when and how something changes, i.e. applying predictive maintenance.

Orhan, Aktürk and Celik found out that vibration monitoring and analysis in rotating machines produces information about the internal structure of the machines. With this knowledge, maintenance activities can be planned and scheduled. [4]

Pedregal, García and Schmid developed with RCM2 a predictive maintenance system for railway turnouts. By combining the maintenance techniques Reliability Centered Maintenance (RCM1) and Remote Condition Monitoring (RCM2), they have created a system that detects errors at an early stage and thus massively improves system safety and reliability of the turnouts [5].

Gilabert and Arnaiz dealt with Predictive Maintenance for non-critical machines such as elevators and machine tools. The elevators were modeled with neural networks and the machine tools were monitored with vibration systems. The final system consists of a sensor-processing unit and a remote maintenance module system [6].

Based on this previous work, this study aims to develop a simple applicable and vendor neutral solution to upgrade installed base products.

### 3 Methods and Theories

#### 3.1 Machine Learning

**Machine Learning** can be described as a computer system that improves itself with experience and as a method that converts data into software. Machine Learning is a sub-area of predictive analytics, the art of estimating or calculating the future from historical data. It consists of a collection of methods used to create software models that are trained by large amounts of data and are then able to recognize certain patterns or trends or even make predictions.

**What drives** the development of Machine Learning? An infinite amount of data is available through digitized historical data or real-time data from embedded systems. Data storage space is globally available for a low cost and there is a number of mechanisms to capture the endless stream of data from portable devices and public clouds. Computing power is virtually endless through cloud computing services which are available everywhere through a large number of cloud and hosting partners. The successful use of the economic power of predictive analytics in many business use cases clearly indicates the rise of predictive analytics.

**Examples** of predictive analytics and Machine Learning can be seen in the behavior of spam filters, which is triggered by the content, title and origin of messages. Another example is pattern recognition, which is used in speech or face recognition. Also, there are ads in web search engines which are tailored for users based on their search history. [5]

#### 3.2 Data Science

Data Science addresses the following five fundamental questions. Each of these questions is answered with different algorithms i.e. Machine Learning methods. One can imagine the algorithm as a recipe and the input data as ingredients. The algorithm specifies how the data is to be combined and mixed in order to get a response.

**Is that A or B?** Simple classification algorithms are used for questions with two possible answers, multiclass classifications for questions with more than two possible answers.

**Is this strange?** Detection of patterns and anomalies are used to identify unexpected or unusual events or behaviors. This indicates where to look for problems.

**How much? How many?** Regression algorithms make numerical predictions and therefore help answer questions regarding numbers.

**How is this organized?** The structure of a dataset needs to be understood. Clustering separates data into natural clusters, simplifying interpretation. Understanding how data is organized makes it easier to understand and predict behavior and events.

**What should I do next?** A group of algorithms called „in-depth learning” answers this question. These algorithms learn from results and choose the next action (reward driven).

In-depth learning is generally well suited for automated systems that require many small decisions without human guidance. Algorithms for in-depth learning collect data during operation and learn through systematic try and error. [6]

### 3.3 Predictive Maintenance

By comparing a sufficient amount of real-time data with previous data, errors can be detected in their origin. Cross comparisons can be used to find correlations and thus not directly related disturbances. Predictive maintenance can eliminate faults long before damage occurs. [9]

To put it bluntly, one wants to find out when something breaks before it breaks, so that a reaction can be planned and implemented at an early stage. This principle applies to large machines, hand-held devices, software and even to humans.

The general approach is to gather suitable data. This data must be prepared, and incorrect or flawed data must be removed to enable Machine Learning. The prepared data then needs to be tagged for recognition while training a machine learning model. After tests and possible alteration, the model is able to do predictions when it is fed with new and unknown data.

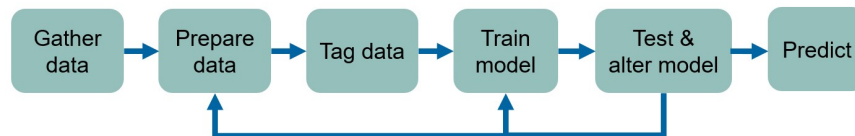


Figure 1: General approach in predictive maintenance

### 3.4 The IoT Stack

“IoT Stack” refers to the overall structure of an IoT application with all layers. In simple terms, an “IoT Stack” consists of three layers. At the bottom is the first layer, consisting of a device that generates data, e.g. a sensor. The second layer includes the communication between the device and the cloud, like a gateway. The third and last layer is located in the cloud and consists of a cloud storage and software, which turns the stream of data into insights and uses.

## 4 Use Case to Showcase IoT on an Installed Base Product

### 4.1 The sample product

The existing system, also referred to as “installed base”, consists of a handling robot of the type M20iA from Fanuc. The System has various interfaces such as ProfiBus, TCP/IP and digital as well as analog inputs and outputs. In this work, the Fanuc robot is considered a black box, which does not communicate. The measurement setup, which is attached to the robot, must work independent.

### 4.2 Use Cases

**Detect motion patterns.** If the robot performs a motion program, the attached sensors send acceleration data via Bluetooth to a microcomputer, which in turn sends combined data packets to a cloud storage. In the cloud, the data will be prepared and built into datasets. Those datasets will then be fed to a machine learning model to identify the active motion.

**Predict load-dependent drift.** Like before, acceleration data will be gathered and sent to a cloud storage, built into datasets and then fed to a machine learning model. Additionally, the information on which motion is active and with which load weight is given. Based on that, the machine learning model then predicts a drift in motion or acceleration of the loaded robot.

### 4.3 Measurable parameters for both use cases

**Acceleration data** in three axis directions will be measured. Based on that, conclusions on the values and directions of acceleration during a motion can be drawn. Also, the alignment of the sensors and therefore the alignment of the robot can be calculated on the “direction” of the earth’s gravitation.

**Gyrometer data** describes the rotational acceleration which will be measured around three axes.

**Environmental data** like temperature, humidity, air pressure or light intensity will most likely have little impact on the given use cases. However, it could be possible to draw conclusions about the season, the daytime and whether someone was near the machine or not.

## 5 Communication Concept

### 5.1 Key considerations

In the case of the used jointed-arm industrial robot, several questions needed to be answered:

**How many sensors?** Nearly any number of sensors can be linked to a single microcomputer. The question is how many should actually be used. A single sensor is not enough because not all of the joints can be observed. With three sensors, a motion can

theoretically be recorded precisely enough depending on various factors. Furthermore, it would probably be possible to reconstruct the position and alignment of the robot axes with three correctly placed sensors.

**Where to put the sensors?** The positioning of the sensors on the robot arm has an influence on the quality of the measurement. Possible positions for sensors could be directly on the wrist of the robot or at the end of axis 5, a sensor on the elbow or directly after axis 3, and also on axis 2 or on the upper arm of the robot.

**How to communicate?** There are several ways to transfer measurement data from the sensors to a cloud database. Data can be sent directly from the sensors to the database via a smartphone or microcomputer. It would also be imaginable to collect the measurement data with a microcomputer and then send it to the cloud via the robot controller, or to send it from the sensors directly to the robot controller and then on from there on.

**Sampling frequency?** In order to be able to retrace a motion, a sufficiently high sampling frequency is required, which can be determined experimentally. However, a higher sampling frequency also means higher data throughput.

## 5.2 Components

**Sensors:** The TI SimpleLink SensorTags CC2650STK are used as sensors. In addition to environmental sensors for temperature and humidity, these include acceleration sensors and gyroscopes.

It would be possible to use other sensors, but the TI sensor tags include the most relevant sensors to cover the desired values for this work.

Apart from that and a reasonable price, the TI sensor tags have a preconfigured Bluetooth interface, which makes further work easier.

**Gateway:** For grouping and forwarding the sensor data a Raspberry Pi 3 microcomputer is used which runs with the operating system Raspbian Jesse. Again, other microcomputers would be possible. However, the operation of the Raspberry is generally known and very simple, which is why it is preferred. If only a single sensor tag is used, it can also be connected to a smartphone.

**Software:** For data handling, Node-Red is to be used, which with its simple, graphical programming is very easy to use. Self-written functions can also be implemented with little effort.

**Database:** The data is stored on the Microsoft Azure environment. Several different database types are available, for the use case SQLite was chosen, again for simplicity reasons.

## 5.3 Communication along the IoT Stack

Three sensor tags are used to trace the motions of the robot. With three sensor tags, acceleration and rotation can be monitored at three different positions. Those positions are at the wrist, elbow and shoulder of the robot (axes 2, 3 and 5, as seen on the picture below).



Figure 2: Installed base with mounted SensorTags

The datasets travel from the sensor tags via Bluetooth to a Raspberry Pi3, representing the local factory system. There, the datasets get accumulated and slightly reshaped before they are sent to a virtual computer in the IoT Gateway in the cloud via a web socket. In this setup a NodeRed instance serves as IoT Gateway. On the virtual computer, the datasets get their final form, which is compatible to the logic of the database. Also, identifying information is added here to make machine learning possible. From there on the datasets get forwarded to a cosmosDB database on Microsoft Azure cloud services. A prescript is necessary to ensure an automated connection to the database and a graphical user interface shows the user the current sensor values.

xpos1p	ypos1p	zpos1p	xpos1n	ypos1n	zpos1n	xpos2p	ypos2p	zpos2p	xpos2n	ypos2n	zpos2n	xpos3p	ypos3p	zpos3p	xpos3n	ypos3n	zpos3n
108	1	1	0	1	0	44	6	0	0	0	0	0	0	1	0	0	0

Figure 3: Example dataset used for training the Machine Learning Model

From the database the prepared datasets get imported into Microsoft Azure Machine Learning Studio using simple sql-query language. The datasets are used to train a Machine Learning Model, which is then able to make (inaccurate) predictions. Tests of that model show if it works or if further improvements are necessary. With a tested and approved Machine Learning Model, predictions can be made. For that a web service must first be set up, which then forwards the prediction of the Machine Learning Model back to the user. Again, node-red is used to show the user graphically what is happening and what the results of the prediction are.



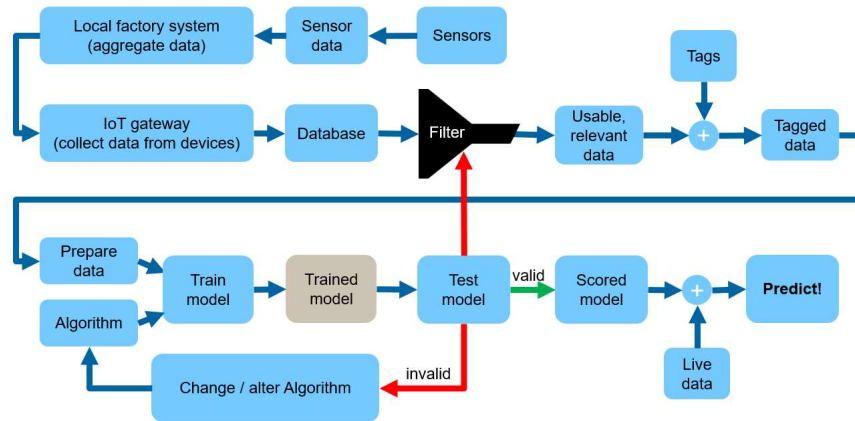


Figure 4: Communication along the IoT-Stack

## 6 Implementing Machine Learning

### 6.1 Providing learning Data

Machine Learning requires that prepared data is available, so-called learning data. Based on this data, a model is trained. The trained model then contains rules and instructions with which it can assign incoming data to the corresponding patterns and thus make predictions.

Depending on the kind of prediction that is to be made, different approaches on recording learning data are possible.

In this case for example, there was no use to feed acceleration curves to the machine learning model as it “learns” row by row and thus will completely lose the time-information of the data. That’s why the time-information needs to be extracted beforehand. A simple one-line dataset is needed that contains all the important data points. The data had to be prepared so that it was simple enough, yet accurate enough for the learning. Eventually, we ended up using a rather simple approach: From the sample shown in Figure 5, the values which were greater than one g or smaller than minus one g per sensor and axis were counted. Such that a complete motion can be represented a one single sample.

### 6.2 Choosing a learning strategy

In the given work a question is asked which has several possible answers. A robust learning strategy is required. Although there are selection guidelines and cheat sheets, it is not trivial to choose a learning strategy and requires extensive tests.

In our case the task will be managed using multi-class classification, since they produced the most promising results. The ultimate goal is to train one or more Machine Learning Models so that they can identify robot motions and positions when fed with

life data. To train a model one feeds it with prepared and tagged data and an algorithm. This algorithm contains the logic. Which algorithm to use can be determined via the type of the problem or experimentally

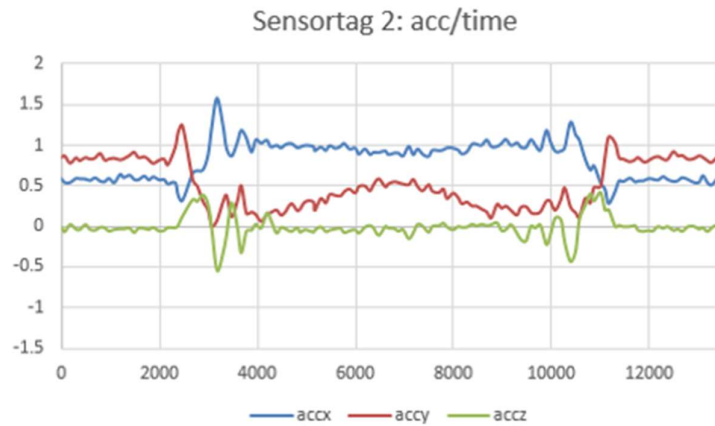


Figure 5: Data sample

## 7 Conclusion and Outlook

It is possible to upgrade existing installed base products to make them IoT capable with simple technologies and tools. However, on the way to the presented solution a lot of unexpected barriers had to be conquered. E.g. unexpected updates of cloud providers can change the complete communication pathway.

Machine learning needs a lot of data. In the frame of this work only a limited number of samples was recorded, which led to unsatisfying predictions. Much more samples are needed to get reliable predictions.

The aim of this study was to use standard tools and algorithms. Machine Learning means an infinitely large collection of tools. It is extremely difficult to understand how these tools interpret data.

Furthermore, the preprocessing of data has an immense influence on the quality of the predictions. It would be thinkable to try out the mathematical programming language "R" to prepare datasets or even to write learning-algorithms. This would probably increase transparency of data processing a lot.

In this work acceleration sensors were the main source of data, it would be conceivable to use other sensors or even other sensor principles.

## References

- [1] B. Fradl, A. Sohrweide and F. Nyffenegger, «PLM in Education - The Escape from Boredom,» in *Product Lifecycle Management and the Industry of the Future*, Turin, Italy, Springer, 2017, pp. 297-307.
- [2] F. Nyffenegger, *Aufgabenstellung\_BA\_2017\_imboden*, Rapperswil: HSR, 2017.
- [3] «Plattform Industrie 4.0,» [Online]. Available: <http://www.plattform-i40.de/I40/Navigation/DE/Industrie40/WasIndustrie40/was-ist-industrie-40.html;jsessionid=D7ED4118DED525F176F382A8F939369D>. [Zugriff am 18 05 2017].
- [4] M. Hermann, T. Pentek and B. Otto, «Design Principles for industrie 4.0 Scenarios,» in *2016 49th Hawaii International Conference on System Sciences (HICSS)*, Hawaii, 2016.
- [5] S. Orhan, N. Aktürk and V. Çelik, «Vibration monitoring for defect diagnosis of rolling element bearings as a predictive maintenance tool: Comprehensive case studies,» in *NDT & E International, Volume 39, Issue 4*, Kırıkkale, Turkey, Elsevier, NDT & E, 2005, pp. 293-298.
- [6] D. J. Pedregal, F. P. García and F. Schmid, «RCM2 predictive maintenance of railway systems based on unobserved components models,» in *Reliability Engineering & System Safety, Volume 88, Issue 1*, Ciudad Real, Spain, Elsevier, 2004, pp. 103-110.
- [7] E. Gilabert and A. Arnaiz, «Intelligent automation systems for predictive maintenance: A case study,» in *Robotics and Computer-Integrated Manufacturing*, Gipuzkoa, Spain, Elsevier, 2005, pp. 543-549.
- [8] Jeff Barnes, *Azure Machine Learning - Microsoft Azure Essentials*, Redmond, Washington 98052-6399: Microsoft Press, 2015.
- [9] Microsoft, «Data science for beginners,» Microsoft, [Online]. Available: <https://docs.microsoft.com/de-de/azure/machine-learning/machine-learning-data-science-for-beginners-the-5-questions-data-science-answers>. [Zugriff am 15 03 2017].
- [10] S. Diamond and A. Marfatia, *Predictive Maintenance for DUMMIES*, Hoboken, New Jersey: John Wiley & Sons, Inc., 2013.