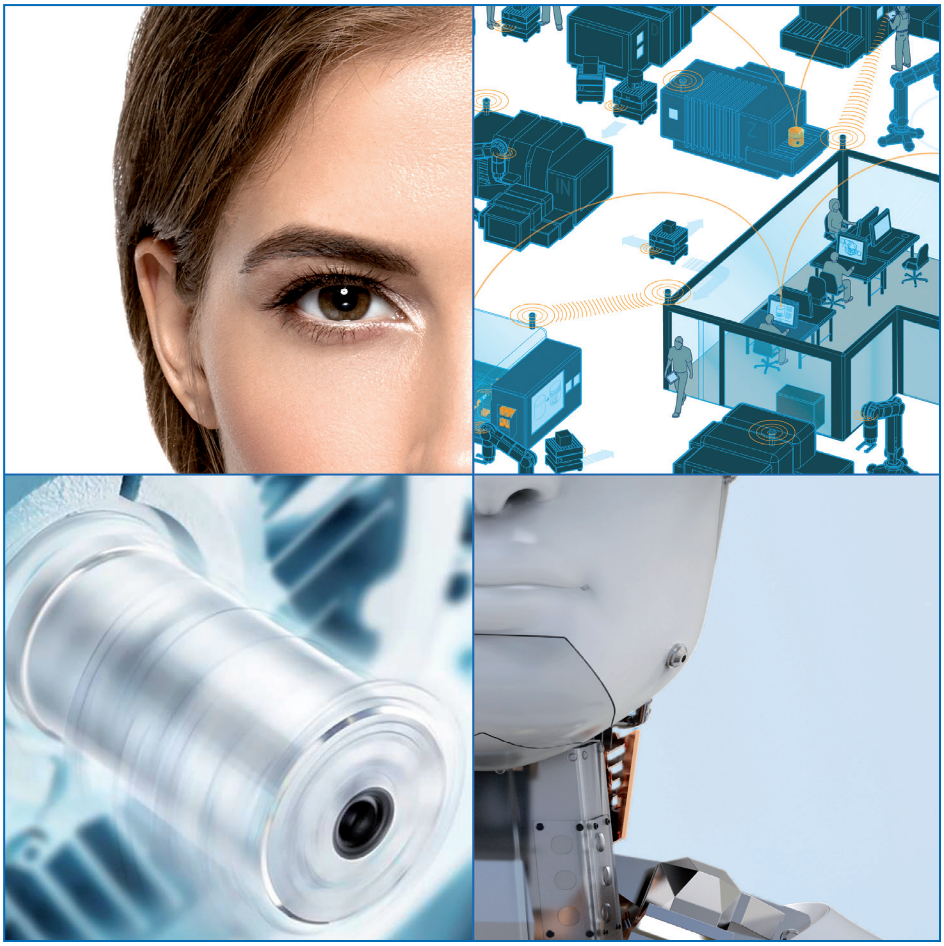
ZVEI: Die Elektroindustrie

Drive 4.0 – Vision Becomes Reality

Properties, data and functions of electrical drive systems in Industrie 4.0 for manufacturers, machine builders and plant operators

Updated and extended edition



German Electrical and Electronic Manufacturers' Association

Die Elektroindustrie

Drive 4.0 – Vision Becomes Reality
(Updated and extended edition)
Published by:
ZVEI - Zentralverband Elektrotechnik und
Elektronikindustrie e. V.
German Electrical and Electronic Manufacturers' Association
Automation Division
Electric Drive Systems Section
Lyoner Strasse 9
60528 Frankfurt am Main, Germany

Responsible: Gunther Koschnick,
Bernhard Sattler, Stefanie Wiesner
Phone: +49 69 6302-377
Fax: +49 69 6302-279
E-mail: antriebe@zvei.org

www.zvei.org

March 2019

Content

1. Management-Summary

In a ZVEI working group, various manufacturers of drive technology met with representatives from mechanical engineering, the process industry and those working in the fields of classification and standards to push forward the implementation of "Drive 4.0" with the aid of university research. An information brochure with a report on the first results was published for the Hannover Messe 2018.

This white paper now presents the results achieved in more detail and explains the implementation of the electrical drive as an Industrie 4.0 component. Standardised, manufacturer-independent data provision plays a particularly important role here. The classification system eCl@ss, which is based on international standards, was chosen to create a uniform data structure with Industrie 4.0 semantics. Numerous properties in various areas of drive technology are already defined in this system. The working group has now defined further key properties for Industrie 4.0 and incorporated them into the standardisation in the eCl@ss specialist group. This group has already adopted the majority of these properties and standardised them for version 11, which is coming in June 2019.

In a further step, initial functions were also described in a manufacturer-independent way. Specifically, these are: oscilloscope, fault memory, maintenance log, energy management and auto-tuning with guided commissioning. To ensure standardised descriptions, a generic function interface was developed in which other functions can also easily be implemented. Now that the first data and functions are available, the working group will work with ten manufacturers on putting the results into practice in a demonstrator based on the motto: Drive 4.0 – Vision Becomes Reality.

For this purpose, the standardised eCl@ss data is to be used in an OPC UA information model and two initial functions (reading the rating plate and the oscilloscope) are to be implemented. The manufacturers are contributing their drive systems and software skills to the implementation of a corresponding hardware and software demonstrator. These efforts are being closely coordinated with other ZVEI working groups, as well as (for example) with the VDMA working group for Industrie 4.0/OPC UA Drive Technology and the Industrie 4.0 Platform working group on reference architectures and standardisation.

In summary, the working group is developing overarching properties, data and functions of electrical drive systems for manufacturers, machine builders and plant operators throughout the product life cycle, thus bringing implementation approaches for the Industrie 4.0 drives of tomorrow closer to reality.

2. Motivation: Benefits for Manufacturers, Machine Builders and Plant Operators

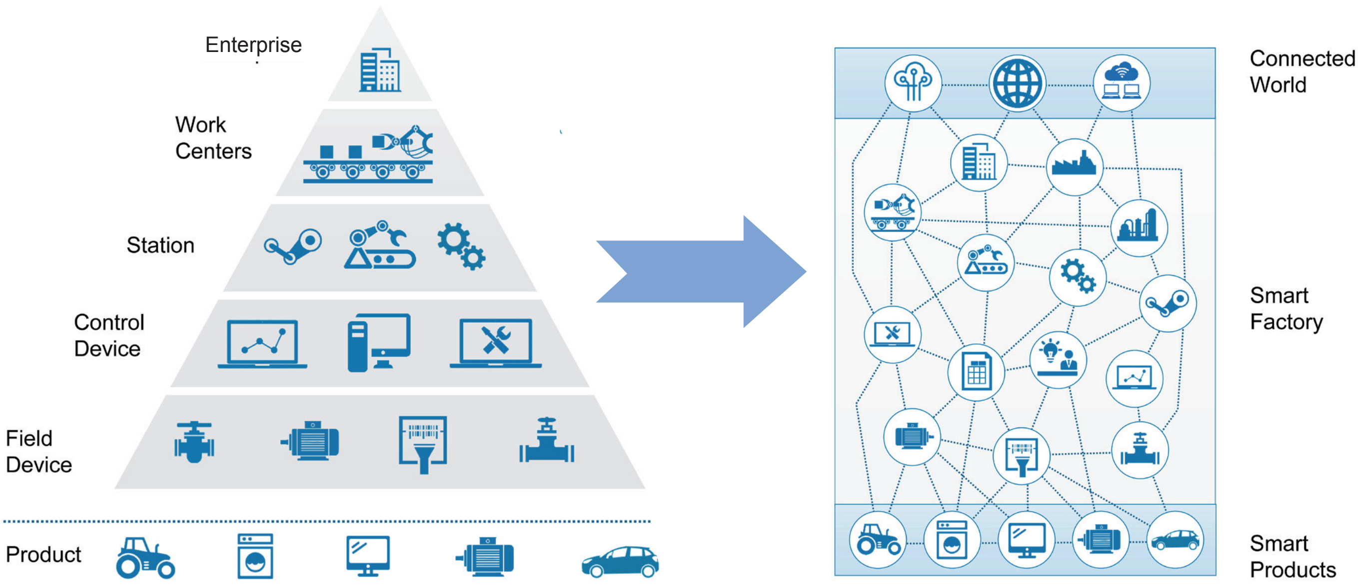
Industrie 4.0 and the corresponding increase of digitalisation and networking provide the opportunity to completely rethink and transform value chains and production processes. The basis for this is digital transformation, which opens the door to smart solutions that afford plant operators seamless knowledg

Digital transformation not only affects the creation of end products, but also production plants themselves, which are becoming modular in a way similar to software products. Each individual module serves as a vehicle of information and knowledge with defined communication interfaces. Suitable horizontal and vertical integration of such modules and their information will result in new, intelligent products and services.

From the perspective of machine builders, Industrie 4.0 – along with related advances in digitisation and modularisation – is presenting tremendous development opportunities based on the potential to combine and interconnect new, intelligent products to create corresponding machines.

The current challenge of integrating aspects such as field bus technologies, drive protocols, secure drive technologies, acyclic data access, firmware downloads, and technologies from the world of IT into machines is becoming easier to manage. The challenge regarding the Internet of Things (IoT) is shaped by a continuity of communication, meaning that everything can interact directly with everything else – from the cloud (the "connected world") and intelligent field devices in a smart factory (drive technology in particular) to smart products – regardless of its place in a given automation pyramid.

Fig. 1: Change in communication from the point of view of the Industrie 4.0 platform



Graphic © Anna Salari, designed by freepik

While machine builders are expecting a great deal of component manufacturers with regard to Industrie 4.0, the expectations that industrial customers are placing on machine builders in terms of establishing standards in this area are much higher still. Past experience has shown that open standards prevail. The opportunity to create standards for data interfaces to field devices should be taken. Possible components for this include:

- Extensions of an established standard such as the drive profile CiA402 (CANopen), PROFINET (PROFIdrive) or the Sercos drive profile with OPC UA.
- Integration of communication options from IT, such as MQTT.
- Data descriptions, for instance from eCl@ss, IEC CDD, AutomationML etc.

The more information the products contained in a machine bring with them, the more intelligent the machine will be. If, along with the provision of the information itself, steps are taken to ensure that it is available across manufacturers in a standardised format, it will be possible to enhance the efficiency and transparency of tasks that machine builders perform themselves, from developing, installing and commissioning machines to monitoring production. This in turn will improve flexibility, help prevent costly production errors, give rise to new solutions, reduce development and commissioning times and make machine builders more productive.

In line with the modularity and aggregation concept, the transparency produced by intelligent machines can be used to increase productivity and reliability. This will usher in new diagnostic functions capable of supplying information about a machine and the components used within it, including on operating condition, device data (parameters and real-time signals), error messages, utilisation (energy efficiency) and required maintenance measures. Such information saves time (including on maintenance), as implementing measures promptly increases machine availability and productivity. Predictive maintenance (condition monitoring) serves as a proactive means of avoiding machine downtime by revealing developments that endanger production. After all, robustness is usually the most important thing in the production industry – even more so than speed.

Drives constitute core components of any machine. Accordingly, these basic components must demonstrate all the aforementioned capabilities and functions to serve as information and knowledge carriers in accordance with digital transformation and bring transparency to the diagnostics of the entire drive system at hand. Complete control of a machine's condition (and of the production process as a result) is impossible without intelligent drives.
The cross-manufacturer data and functions generated in the working group Industrie 4.0 for Electrical Drive Systems are a decisive step towards the desired transparency. The asset administration shell takes care of recording and administering all the necessary data for the entire product life cycle, from development and commissioning to operation and service, and can be expanded if required. The manufacturer-independent standardisation of the data (and the function interfaces) results in the desired flexibility, meaning components are interchangeable without having to adapt systems or system interfaces extensively to manufacturer-specific data structures. The selected, standardised functions will aid machine builders in machine development, commissioning, maintenance and production monitoring throughout each machine's useful life while also contributing to machine optimisation.

Furthermore, ensuring that drive components are interchangeable creates opportunities for machine design in the context of platform production without the need for significant additional effort. In the future, it will thus be possible to design performance levels based on a machine's individual components and their capabilities, with the basic functions always being retained. Differentiation will be based on the quality of the hardware and the software supplied with it. On this basis, machine builders can then use intelligent analysis algorithms to further process the recorded and provided data on their own at a higher level. They will be able to create sophisticated service and maintenance strategies, ensuring the maximum technical availability of machines across the entire product life cycle. Transparency, optimised processes and complete control at all times: the smart factory makes compliant drive components and systems possible thanks to Industrie 4.0.

As the life cycles of machine elements become increasingly short, establishing standardised interfaces will be even more important in the future. Otherwise, machine builders will spend more of their time on integrating new hardware and less on generating innovations for customers. In order to remain competitive, these innovations must be supported and advanced worldwide.

3. Foundations for Industrie 4.0

3.1 The reference architecture model for Industrie 4.0 (RAMI 4.0)

RAMI 4.0 is the first time the key characteristics of objects in Industrie 4.0 solutions have been compiled in a three-dimensional layer model. Based on this structure, the content of Industrie 4.0 technology can be systematically classified and developed further.

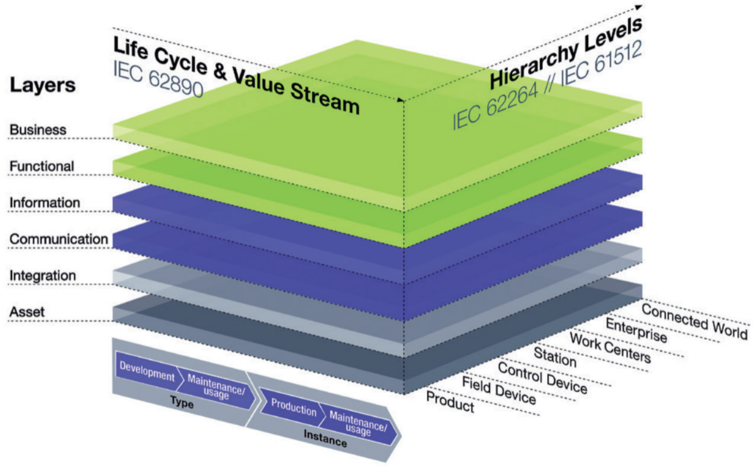
RAMI 4.0 – the model

The reference architecture model for Industrie 4.0, RAMI 4.0 for short, consists of a three-dimensional coordinate system that contains the key aspects of an observed object in Industrie 4.0. This object is termed an object of value, or asset, for Industrie 4.0 purposes. An asset can be an electronic component such as a drive, a system (such as the drive train), a plant, or even an entire factory made up of different elements.

The "Hierarchy Levels" axis

The right-hand horizontal axis represents the hierarchy levels of IEC 62264 from the international standards series on the integration of company IT and control systems. These hierarchy levels assign the various functions within a factory or plant to a hierarchy level. Here, the hierarchy axis has been extended by three levels not included in the standard. At the lower end, these are the "Product" tool and the "Field Device" (due to significantly greater requirements associated with sensors and actuators). At the upper end, the hierarchy level "Connected World" reflects the networked world to other assets.

Fig. 2: RAMI 4.0



Source: Plattform Industrie 4.0

The "Life Cycle & Value Stream" axis

This axis represents the life cycle of Industrie 4.0 assets, including objects such as plants and products. In addition to IEC 62890 (life cycle management), this will be based in the future on DIN 77005-1:2017-11 (life cycle record of technical objects), which is currently being created in several parts. The life cycle record of an asset makes a basic distinction between the statuses "type" and "instance". A "type" becomes an "instance" once development and prototype production are complete, the actual product is being manufactured and can be used.

The "Layers" axis

The six layers on the vertical axis of the model are used to describe the relevant properties of an asset that are particularly significant when presenting this physical-world asset in the information world. The division into layers makes it easier to describe an asset using terms for mapping it in the information world. Terms that provide an informatics description of an asset's characteristics are known as properties, and in standards frequently also as "data elements".

While the asset layer maps the physical world in which the asset is used, the integration layer represents the transition from the physical world to the information world of Industrie 4.0 and vice versa. For example, the conversion of the speed of a motor in the physical world into an electrical binary value for the information world takes place in this layer. For all non-Industrie 4.0 applications, the entire functionality ends in the integration layer. Every Industrie 4.0 asset has either a passive or an active communication interface,[1] which is represented by the communication layer. In order to better distinguish between functions and their data in the future, there are two further layers: "information" and "functional". The joint consideration of functions and their data, which was common until now, will make it easier not only to describe big data and technical functions in the future, but also to localise them "on site", in the cloud, and so on. The business layer contains all the business-relevant rules and framework conditions, including the rules directly related to the business, throughout the life cycle of an asset.

These three dimensions of the reference architecture model are used to describe an object that is relevant for Industrie 4.0, such as a machine, a product or a factory. This concept can be used to describe and implement highly flexible Industrie 4.0 concepts. The reference architecture model facilitates migrations from today's Industrie 3.0 world to the world of Industrie 4.0 by means of the step-by-step transfer of Industrie 3.0 content from the integration layer to the higher levels of the reference architecture model following its conversion to Industrie 4.0-compliant content.

The benefits of RAMI 4.0

The model combines a range of key user perspectives to promote a shared understanding of Industrie 4.0. RAMI 4.0 can be used as a basis for discussing the requirements of the application industry – from manufacturing automation and mechanical engineering to process engineering – in the relevant bodies of the associations, consortia and standardisation bodies. The model is thereby establishing a shared understanding of required standards and practical case studies.

RAMI 4.0 is a kind of 3D map for Industrie 4.0 solutions. It makes it possible to discuss, design and further develop the requirements of the application industry with regard to national and international standards. This in turn reveals overlaps and gaps in standardisation and presents opportunities to address them fully.

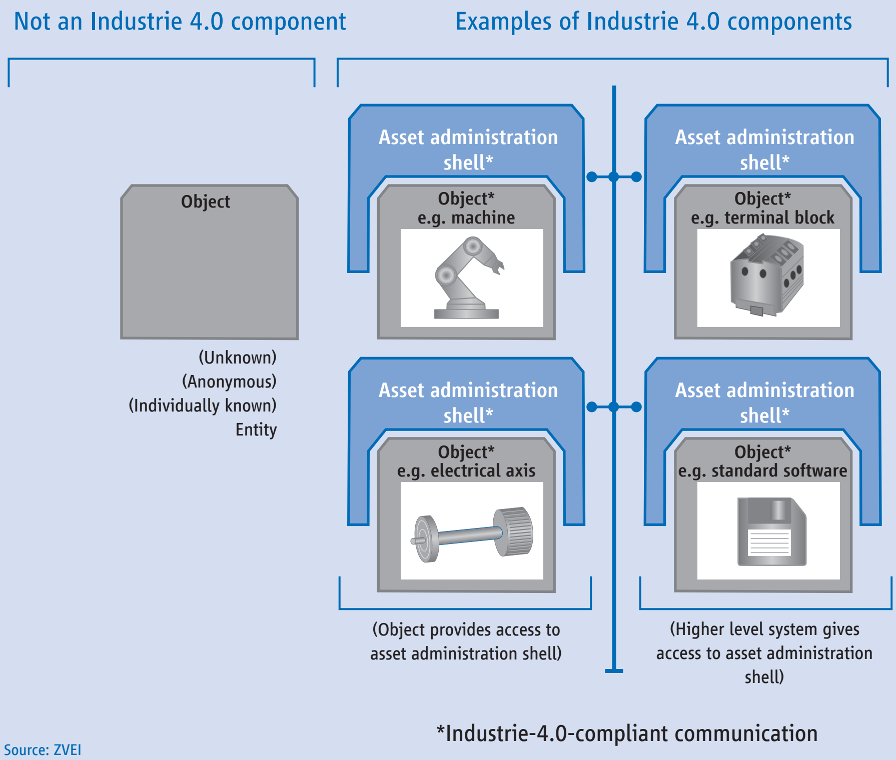
3.2 The Industrie 4.0 component and asset administration shell

The Industrie 4.0 component[2] uses the asset administration shell to map all the relevant properties and information of a physical object in the IT world of RAMI 4.0. The term "asset administration shell" thus covers one or several physical objects ("assets") and the shells that surround them, including all the relevant information and functions of the asset in the information world. This data can be generated, modified or saved, and it facilitates communication and networking capabilities in the scope of Industrie 4.0. A key requirement in this regard is that Industrie 4.0 components must maintain data and functions in the asset administration shell in a secured electronic "container" (be it a storage location created specifically for this purpose or one on the asset itself) throughout their entire life cycle. This results in a high level of transparency for manufacturers, machine builders and plant operators and enables horizontal and vertical integration in accordance with RAMI 4.0. For Industrie 4.0 to succeed, it is vital that not only entire machines, but also information on key machine parts and components be maintained in the asset administration shell. For example, the properties of the drive train significantly affect the quality of the machine function. In the future, it should be possible for this property information also to be recorded directly by the central maintenance system. The same applies in automation technology to production components that do not have their own data interface, for instance cable properties such as cross-section and shielding. In this way, every part becomes a smart component of networked production in Industrie 4.0.

[1] For information on active and passive communication capabilities, see: Roland Heidel Michael Hoffmeister, Martin Hankel, Udo Döbrich, Basiswissen Industrie 4.0, Referenzarchitekturmodell und Industrie 4.0-Komponente - Industrie 4.0, Beuth-Verlag 2017.

[2] See https://www.zvei.org/presse-medien/publikationen/die-industrie-40-komponente/

Fig. 3: Object as an Industrie 4.0 component



Source: ZVEI

The Industrie 4.0 component

The Industrie 4.0 component is a model for making the relevant properties and information of an object in the IT world accessible by means of its asset administration shell. The term “asset administration shell” comes from the idea that information regarding one or more objects from the physical world surrounds the “asset” like a shell and makes all the relevant information for the physical-world asset available in the information world.

Hardware and software components in production, from the production system to the machine or control level down to the individual components within a machine, become Industrie-4.0-compatible through their addition to asset administration shells. In the process, this information is structured in accordance with RAMI 4.0.

The Industrie 4.0 component model thus describes the requirements of Industrie-4.0-compliant communication between individual hardware and software components (assets). This also includes which data (information) can be generated, modified and saved in connection with which functions.

A key requirement in this regard is that Industrie 4.0 components must maintain data and functions in the asset administration shell in a secured electronic “container”, be it in a (remote) repository or on the asset itself, throughout their life cycle.

The asset administration shell

The asset administration shell contains all the information regarding the relevant properties of an asset consisting of hardware or software components. For example, a machine (asset) consists of many assets that together represent a summary of their overall properties in the information world, whereby the overall functionality is more than simply the sum of the assets’ functions. This results in new possibilities, including in networked production. Ultimately, this creates added value for the companies involved in the value creation process. This added value can be described as follows:

- Data

The asset administration shell of an Industrie 4.0 component contains a great deal of data and information that manufacturers provide in an Industrie-4.0-compatible format, such as CAD data, wiring diagrams or manuals. System integrators and operators of plants and factories can add further key information to the part of the asset administration shell that is made accessible to them. This might include information on maintenance or interfaces to other hardware and software components. The Industrie 4.0 platform defines measures for data security to ensure that the availability, confidentiality and integrity of information is maintained for all users.

- Functions

An asset's technical functions are also provided in the asset administration shell for use in the information world. This includes, for example, planning, project planning, configuration, operation, maintenance and complex business logic functions, as well as application functions such as boring and milling[3].

- Services

Data and functions are available on the Industrie 4.0 component itself, on a company's network or even in the cloud. The added value lies in that information is saved only once in an Industrie-4.0-compatible way and can then be provided transparently to every user and application case via IT services.

- Integration

The combination of Industrie-4.0-compliant communication protocols and the idea of the asset administration shell results in the horizontal and vertical integration of the application at hand.

- Seamless knowledge

The end result is that seamless information is made available for both engineering and operation and maintenance purposes. This can be used to generate knowledge.

- Modularity

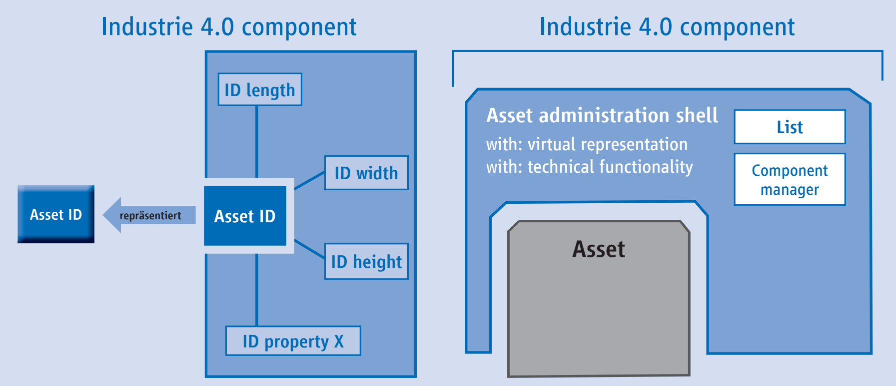
For Industrie 4.0 to succeed, it is vital that not only entire machines, but also information on key machine parts and components be maintained in the asset administration shell. For example, the properties of the drive train significantly affect the quality of the machine function. In the future, it should be possible for this property information also to be recorded directly by the central maintenance system. The same applies in automation technology to production components that do not have their own data interface. For instance, a motor supply cable can transfer information to the asset administration shell regarding what was connected when, for what purpose, and with which cable properties (cross-section, shielding, etc.). In this way, every part becomes a smart component of networked production in Industrie4.0.

3.3 Existing standards

In order to make a drive Industrie-4.0-compatible, certain specifications are required that describe an Industrie 4.0 drive component in accordance with Industrie 4.0 specifications. For this purpose, the drive is described using terms that are stored as properties in the asset administration shell of the respective Industrie 4.0 drive component. The asset and asset administration shell also need to be assigned unique identifiers (IDs). To migrate a modern drive to an Industrie 4.0 environment, the Industrie 4.0 components must contain descriptions in accordance with Industrie 4.0 specifications (e.g. from eCl@ss[4]); in other words, all the properties must be specified based on the rules of IEC 61360[5,6] or ISO 13584 42[7].

[3] Each of these functions can be a component of an asset (information and functional layer) while also representing assets themselves for more precise description.
[4] www.eclass.eu
[5] See IEC 61360-1:2009, Standard data elements types with associated classification scheme for electric items – Part 1: Definitions – Principles and methods
[6] See IEC 61360-2:2012, Standard data element types with associated classification scheme for electric components – Part 2: EXPRESS dictionary schema
[7] See ISO 13584-42:2010, Industrial automation systems and integration -- Parts library -- Part 42: Description methodology: Methodology for structuring parts families

Fig. 4: Industrie 4.0 component



Source: ZVEI

There are already various standards regarding the conversion of a drive to an Industrie 4.0 drive component. We therefore expect that the transition to Industrie 4.0 will not cause too much disruption.

Cooperation capability of Industrie 4.0 components

When standardised communication protocols were being developed, "cooperation capability" ("cooperability") was still referred to as "interchangeability". This term describes important criteria that go beyond simple exchange of data. Here, the perspective of pure data exchange is expanded into an application-oriented viewpoint based on cooperation at the application level. Since this expansion is characterised by standardised "(application) functions" and "compliant dynamic behaviour", standardised semantics play a key role. The structure of Drive 4.0 can be illustrated by the device model from IEC 61804-2.

Fig. 5: Cooperation capability matrix

Compatibility level

Interchangeability = Cooperability

Needed feature	Incompatible	Coexistent	Interconnectable	Interworkable	Interoperable	Interchangeable	
Dynamic performance						•	Cooperation between any objects of value
Application functionality					•	•	
Parameter semantics					•	•	
Data types date access				•	•	•	Communication
Communication interface			•	•	•	•	
Communication protocol		•	•	•	•	•	

Source: Heidelcom/IEC 61804-2

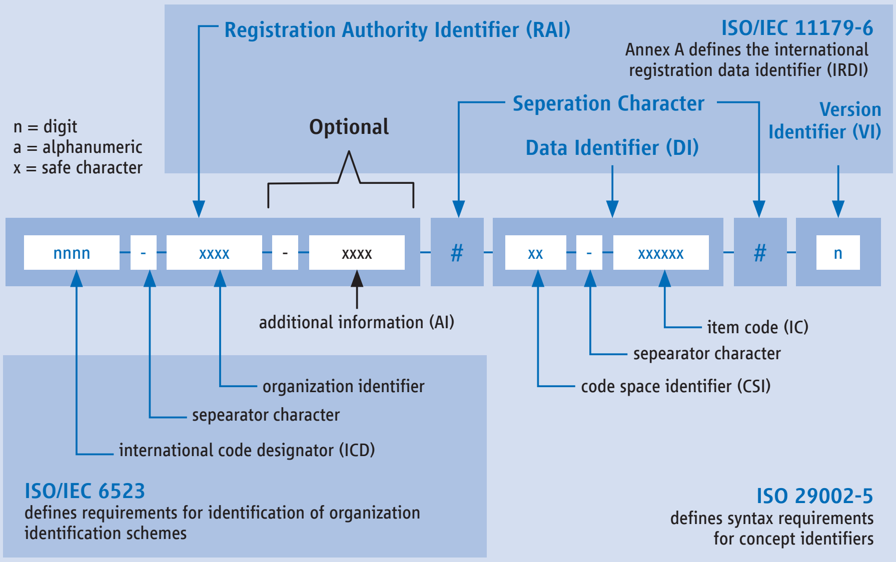
Identification of assets

Achieving the goal of providing standardised information models that can be used across manufacturers will require the use of digital and uniquely generated properties with reusable value ranges by means of an IRDI (International Registration Data Identifier).

Here, the repositories from eCl@ss e. V. (among others) provide the foundation for creating and maintaining properties in a form that is standardised according to Industrie 4.0 in line with ISO/IEC 6523 and ISO/IEC 11179-6 (on the basis of ISO29002-5). A property in eCl@ss essentially consists of the identifier for the issuing organisation and the item code that characterises the property in question. For example, the identifier for the eCl@ss organisation is 0173, and the item code for the property "power data (motor)" is "BAE083". In this way, the item code uniquely identifies every motor with regard to this characteristic. For further information on this, see the section on properties.

In terms of standardisation, an Industrie 4.0 drive is also based on the relevant standards for drives, including the IEC standards series 60034-X (motors); the relevant NEMA, UL and CSA specifications on efficiency classes for three-phase, asynchronous current, low-voltage motors; and the IEC 61800-X series (power drive systems).

Fig. 6: ISO 29002-5:2009



Source: ZVEI

4. User Groups – Value Creation Processes

4.1 Introduction

A drive is a core component that goes through a range of very different value creation processes, from planning, manufacturing and integration into machines to commissioning and eventual use in the production process. In combination with sensors and IT solutions, electrical drives make it possible to rethink actuators. They can therefore be considered as initiators of Industrie 4.0. As a result, drive technology can lay claim to an active role in advancing the design of Drive 4.0 as a future industrial product.

For this purpose, the ZVEI working group Industrie 4.0 for Electrical Drive Systems is currently developing a concept that deals with terminology and classifications relating to these drives, places properties and functions in an Industrie 4.0 context and ultimately lays the foundation for standardised drive digitisation. As a result of digitisation, manufacturers, machine builders and plant operators will obtain extensive knowledge of the production process and increased transparency regarding technical processes. This digital transformation is establishing and driving forward modularisation in the value creation chain of production plants and products similar to that found in software products. Each individual module serves as an information and knowledge carrier with defined properties and communication interfaces.

In particular due to the increasingly shorter life cycles of machine elements, the goal must be to establish standardised and generally applicable standards for the implementation of interfaces and data communication. Suitable horizontal and vertical integration of such modules and their networked information will result in new, intelligent products and thus, in drive technology, also in Drive 4.0.

"Industrie 4.0 stands for full digitisation and integration of the industrial value creation chain. Connecting information and communication technology with automation technology to the Internet of Things and services enables an increasing degree of networking within and between production plants – from the supplier right through to the customer. This also includes digitisation of the product and service offering, which allows new business models. Ultimately, Industrie 4.0 is the realisation of the smart factory within the digital value creation network."

Gunther Koschnick, ZVEI

In order to achieve this ambitious goal, in the working group's view it is necessary to provide data from electrical drives in a standardised, manufacturer-independent form. These standardised, cross-manufacturer interface descriptions and data structures should make it easy to combine components and system parts freely and make simple, efficient use of the relevant data and functions. Communication protocols and mechanisms specified based on Industrie 4.0 should be used to transfer this data. Based on potential application cases, the working group identifies and analyses information that can be allocated to the following roles in the value creation process in the scope of Industrie 4.0:

- Drive manufacturer:

The manufacturer produces the drive from various parts and individual components, tests it and delivers it. The manufacturer brings the drive to market.

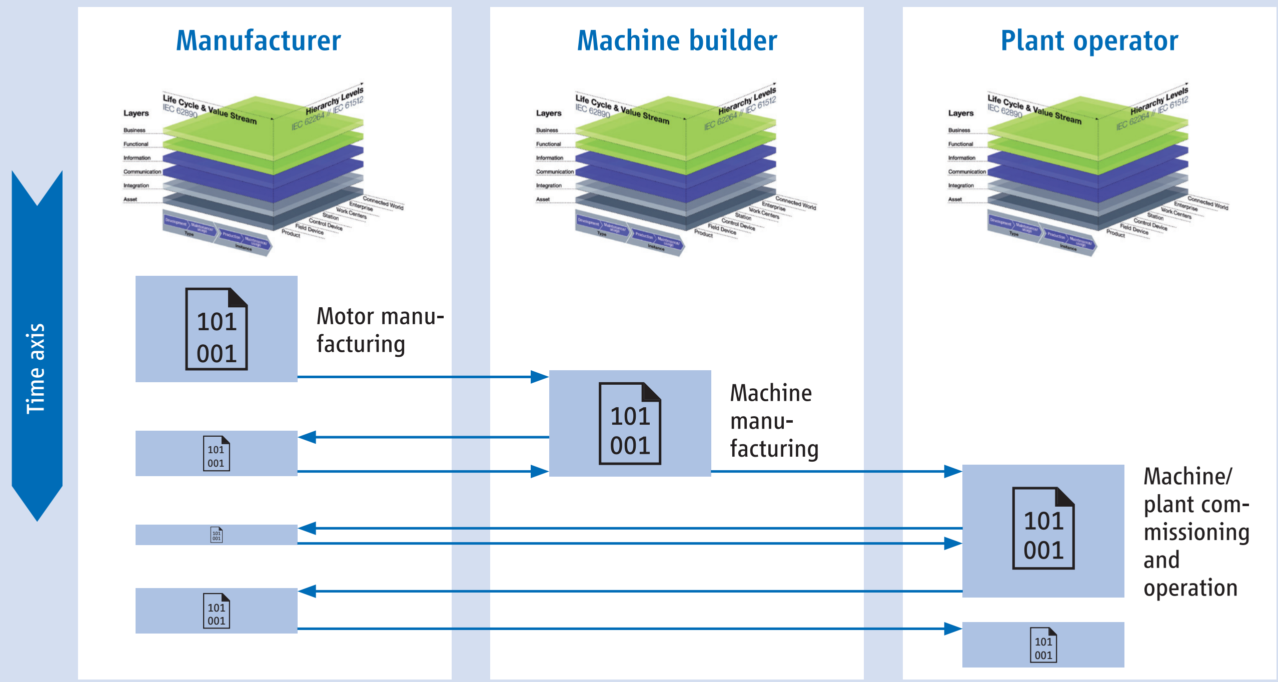
- Machine builders

The machine builder develops a machine by combining and networking drives and other components. He or she then brings this machine to market. Frequently, he or she also takes care of installation and commissioning. This definition can also be used in the same way for plant engineers or similar roles.

- Plant operator

The plant operator integrates the machine produced by the machine builder and operates it following installation and commissioning. Usually several machines and/or machine modules are coupled and networked to establish a production plant or process engineering plant.

Fig. 7: Roles and data in the value creation process



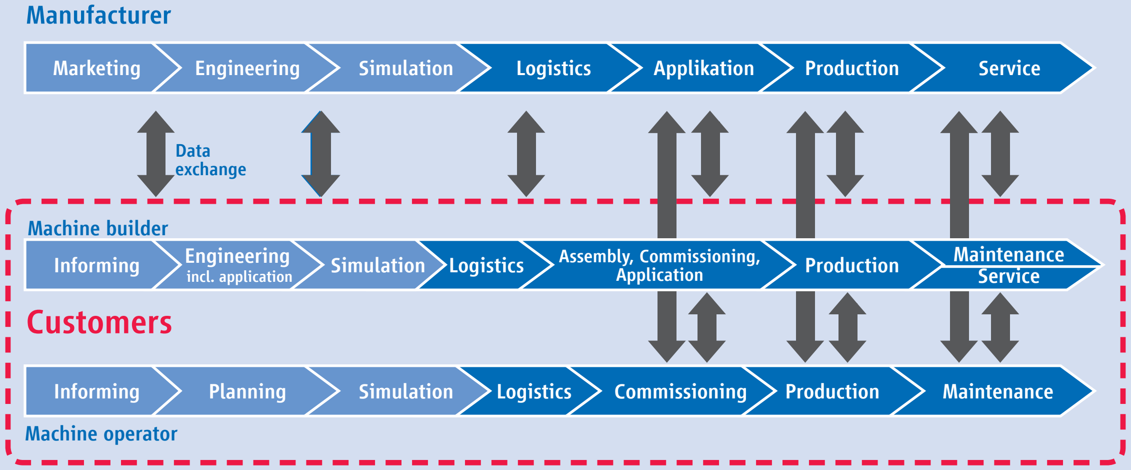
Source: Plattform Industrie 4.0/ZVEI

Based on potential application cases, the ZVEI working group Industrie 4.0 for Electrical Drive Systems has identified data that is relevant for users in various areas of the value creation chain in the context of Industrie 4.0. However, the relevance of the data for individual users strongly depends on the user's role in the value creation process. To allow for clear assignment after the analysis of use cases and data, manufacturers, machine builders and plant operators were specified as user groups.

In reality, the same person can belong to various user groups and take on different roles. For instance, machine builders who produce their own motors are simultaneously manufacturers and machine builders. The same applies to plant operators who, for instance, handle the engineering of their plants themselves and go on to operate them. They are then machine/plant builders and plant operators at the same time. It is also possible that a machine or plant builder no longer sells an actual machine/plant, but production or process services (comparable to a leasing model).

In addition, the networking of individual manufacturer, machine builder and plant operator value creation chains is being increasingly digitised. In this way, the individual value creation chains form a veritable "value creation network" that enables fast, easy exchanges of data and information in all phases of the value creation chain in a form that is machine-readable (and therefore automated). This contact and data exchange takes place not merely through the exchange of an asset (such as a motor or converter); it can also occur earlier in connection with catalogue data, for example, or virtually during the development phase, when the right motor for the machine is being sought and the machine is being simulated in advance. The following figure aims to clarify the digitised exchange of data for all phases within value creation chains.

Fig. 8: Exchange of data within the life cycle in the value creation network



Source: ZVEI

Data can be extremely diverse: simple properties or parameters such as "maximum rated current", files with 2D information for CAD drawings or manuals, or even complex descriptions such as a 3D simulation model.

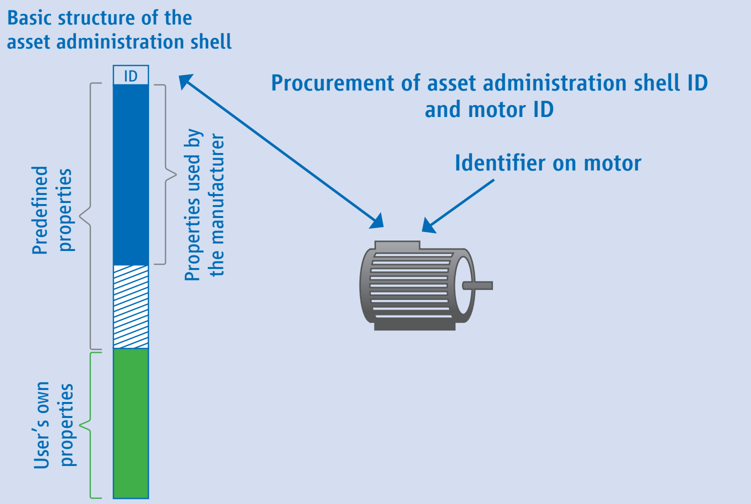
The data that the various users generate is collected and stored in the asset administration shell.

The asset administration shell receives a unique identifier (ID) and also contains a clear reference to the asset in question – for instance, a motor.

First, the predefined applicable data must be filled out (predefined properties). Only through the use of agreed, generally valid data are automatic exchanges possible later on. Of the many predefined properties, only the data that is relevant for the use case in question needs to be filled out.

Manufacturer-specific data that is intended for internal use (the user's own properties) can also be stored. This makes it possible to realise additional functions, special features of the manufacturer in question or new customer requirements, such as sensor data from a test run or internal design drawings.

Fig. 9: Template of the data in the asset administration shell



Source: ZVEI

An asset administration shell is thus generated for the same "motor" asset for all three value creation partners. Each partner fills its asset administration shell with the data they require for their purposes; some is exchanged among the partners, while other data remains with the individual value creation partners. The entirety of all the data that is manufacturer-independent from the working group's perspective was compiled and sorted based on the respective generation phases within the value creation chain and the roles such data plays within the value creation network. This resulted in an initial template for classifying the data.

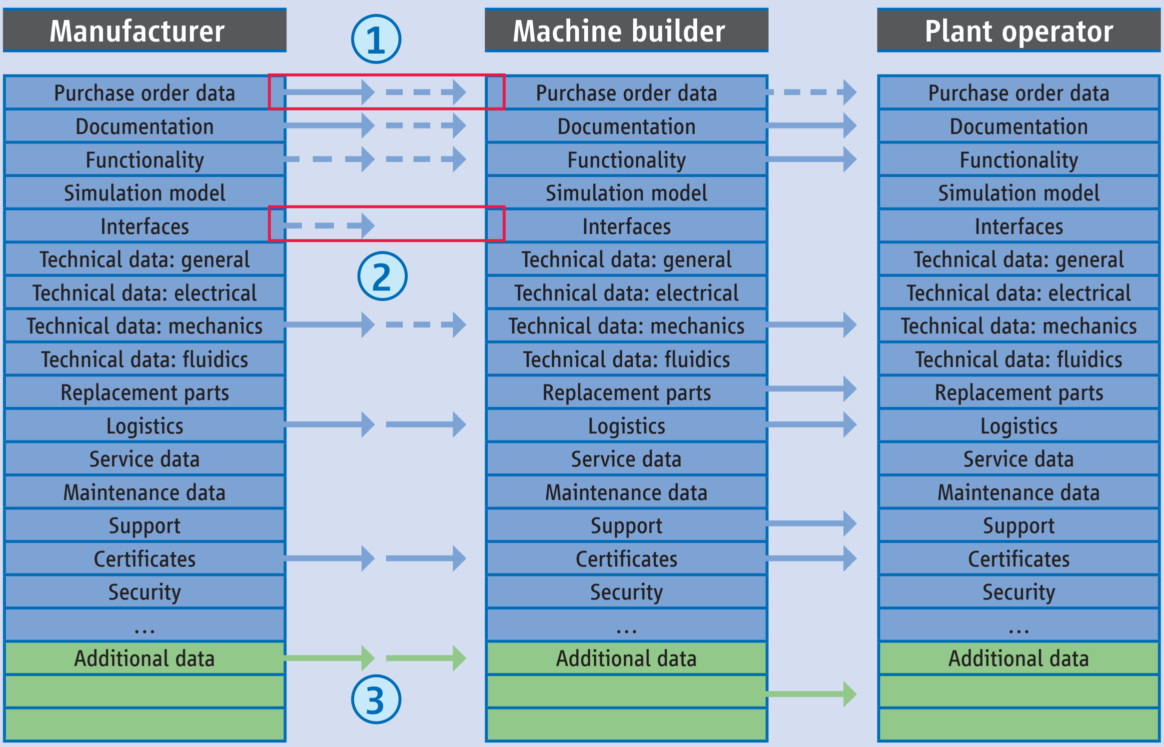
Manufacturers add data to the template that they require both internally and for their customers. Figure 10 shows how a manufacturer offers some data to a machine builder (blue arrow). The solid arrows indicate that all the data is offered, while the dashed arrows denote that only some of the available data is offered.

As the recipient of the data, the machine builder now decides whether to add all the data to his or her asset administration shell for the same motor or takes a more differentiated approach. 1 indicates that the machine builder only uses some of the order data, as he or she may already have some. For 2, the interfaces, the machine builder does not take on any of the manufacturer's data in this example. Perhaps this information is not needed for his or her purposes. And for 3, the machine builder adds all the non-standardised data from the manufacturer to the asset administration shell. The machine builder then also decides which data to pass on to the plant operator.

Like the machine builder, the plant operator can choose which data to add to his or her own asset administration shell for the same motor.

The transfer of data also generally works in reverse in the value creation chain; the plant operator can also pass data on to the machine builder or directly to the manufacturer in the same manner. Since all the involved parties use the same motor template, they can all understand and classify the data.

Fig. 10: Template of the data and exchange of data between value creation partners



Source: ZVEI

During operations, other data is generated that may be of interest to the various participants in the value creation chain. This data is generated by both the machine builder and the plant operator, and is also stored in the asset administration shell.

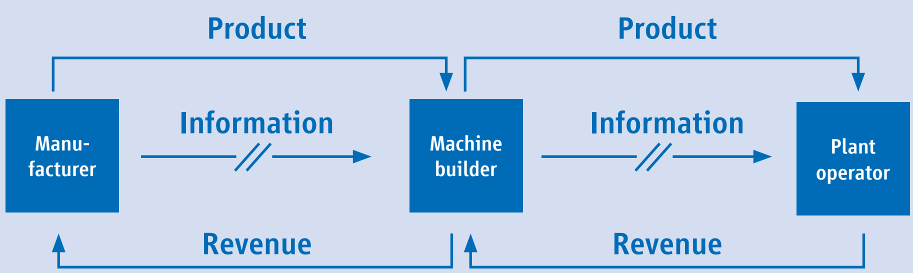
The fact that so many users change the asset administration shells underlines how dynamic they are and that asset administration shells can be filled with different data for different users of a certain product (for example, the motor). A unique product ID is the only way to ensure that data can be traced among various users.

4.2 Changes in the value creation process of drive technology due to Industrie 4.0

Drive 4.0 is successively expanding the available options for implementing new value creation potential and architectures. Although drive manufacturers, machine builders and plant operators continue to form the core of the value creation chain, thanks to closer networking and additional information flows, further value creation constellations with new actors will form.

Manufacturers, machine builders and plant operators have aligned today's value creation efforts to achieve high efficiency and an optimal supply of products across the various stages involved. The objective here is to increase productivity, quality, delivery capability and innovation speed. In this context, shorter product life cycles – which intensify the time pressure and coordination effort involved in developing new product and machine generations – are becoming increasingly challenging for machine and plant builders. In today's value creation process this development is often slowed down in practice as information is painstakingly collected along the value creation chain and converted to new target formats and systems. Although interface technologies enable this conversion and transfer, the latter requires manual assignment of properties and property values in mapping tables. Drive 4.0 can reduce these interruptions in the flow of information thanks to its standardised product properties, which in turn will increase the quality, speed and efficiency of cross-company information processing in today's value creation process.

Fig. 11: Today's value creation process

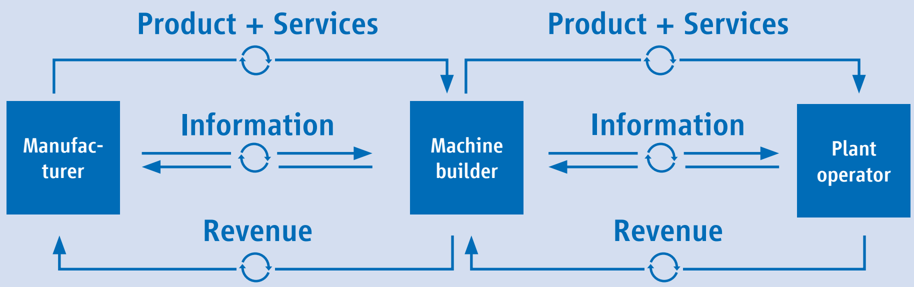


Source: ZVEI

However, Drive 4.0 also makes it possible to tap into new value creation potential within existing structures. For example, in the future it will be possible to provide digital services along the classic value creation chain for the entire product life cycle of a drive system, which will result in new business models. For this purpose, the standard functions of Drive 4.0 (which all products must include as the minimum standard) can be supplemented with manufacturer-specific digital services, thus becoming distinguishing features. New digital elements such as enhanced sensors and IT communication interfaces allow additional data to be recorded, enriched and processed as a drive system is being used. This forms the basis for data-driven services, which can increasingly be added to new service offerings and accessed as required. Here, the increasing role of software is changing how value is created.

Manufacturers are investing many hours in the early stages of software development. Only open standards that apply to (and are supported by) a large number of manufacturers can secure these investments and ensure the necessary technical interoperability. The standardised basic data and functions of Drive 4.0 enable companies to act more flexibly, as drive systems can also be integrated into heterogeneous environments without any specific configuration or programming. They can thus be made available in a cost-effective manner. The rising significance of digital services and data is also reflected in a changed revenue model. In contrast to one-off product sales revenue, flexible usage and time-based revenue models will become increasingly important in the future. These changes have already become established in the IT industry. Instead of selling software products by means of one-off payment transactions, software companies have increasingly switched to offering their products via a licence model. Now, as Industrie 4.0 makes its way into drive technology, such revenue models are also becoming an option for both machine builders and drive manufacturers. This gives drive manufacturers and machine builders a source of recurring income, while plant operators benefit, for example, from the latest services or optimised usage fees.

Fig. 12: Changes in the existing value creation structure

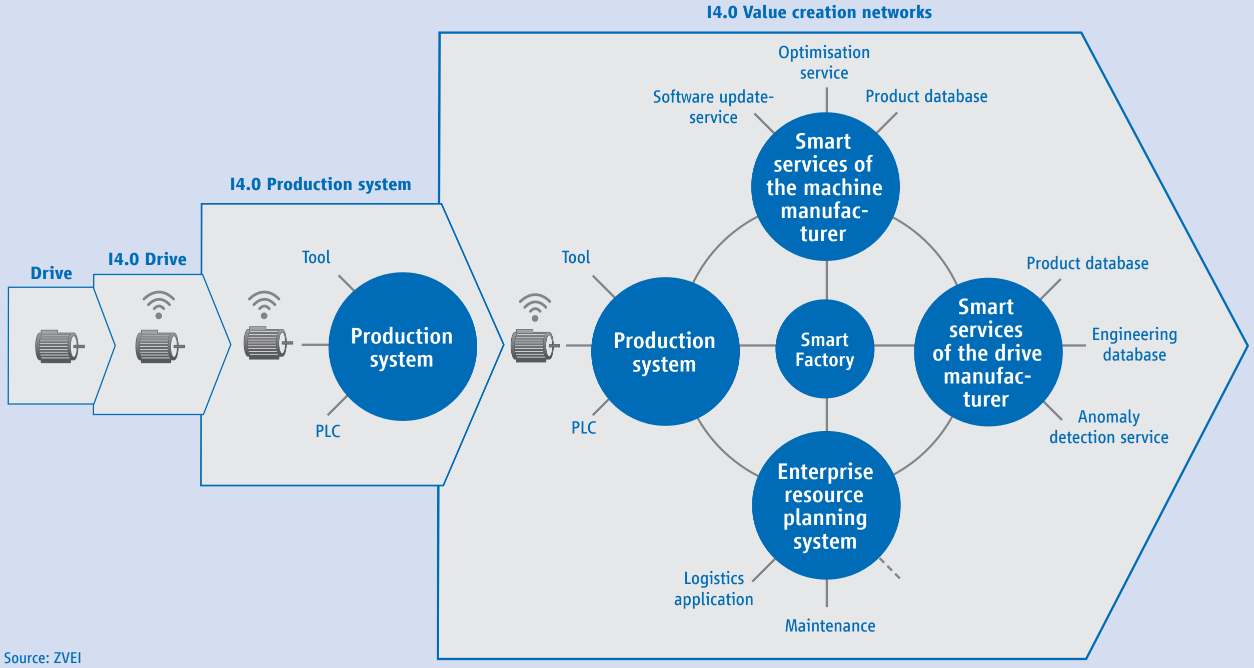


Source: ZVEI

The bridges built between automation and the IT world are also generating new value creation structures. While plant engineers and automation providers have previously only had access to local PLC programs and interfaces, software providers can now use high-level IT languages to access plants and drive systems that are globally connected via the Internet. Cloud services allow the use of almost endless memory and computing capacity and make it possible to launch solutions on the market at an unprecedented speed. Thanks to networked plants and IT systems, in the future it will no longer be possible for a single manufacturer to develop solutions in-house and offer them to consumers as a finished product or service. Instead, future solutions will build on shared basic services and access distributed data. For example, it would be uneconomical if plants were connected to the Internet several times – or worse, if individual components were connected directly and based on different "connectivity services". The associated configuration effort and security risks also surely cannot be in the interests of plant operators.

Here, the classic architecture of the value creation chain will prove to be a limiting framework in the future. Conversely, platform architectures and the associated new understanding of roles will provide a more flexible framework for these future data-driven and digital structures of value creation. Here, Drive 4.0 plays an enabling role for the new platform architectures in machine and plant construction, as it will be possible to procure functions and basic services via IoT platforms in the future. Plant operators can take on the role of data providers and make the information generated by the drive systems in their smart factories available to various actors through a number of platforms. Service providers use the data supplied via a given platform, develop solutions that they can draw on in addressing the specific problems of individual actors and expand them into scalable services. The corresponding platform provider defines the governance framework within which the actors can organise and do business efficiently. By concentrating on their own expertise, consciously deciding on one or more platform roles and using additional services from their partners, automation manufacturers and the machine construction and IT industries can quickly achieve viable, future-proof and innovative solutions and thus form entirely new value creation networks.

Fig. 13: Drive 4.0 as an enabler of new value creation networks[8]



Source: ZVEI

Open value creation networks result in completely new revenue strategies. Up to now, almost all revenue strategies in machine and plant construction have concentrated on the basic principle of providing or developing goods or services in exchange for money. Now, however, revenue strategies are possible in which a product or service is paid for indirectly. This could, for example, involve using data that is generated while a machine is operating, the value of which is difficult to quantify in a universally applicable way. For a machine builder or drive manufacturer, this data could be so valuable that it would allow maintenance or repairs to be performed at no cost, as they would be indirectly funded throughout the life cycle of the machine.

How today's value creation chains in the machine and plant construction industry will change as a result of this cannot be foreseen in detail from our current perspective. However, the exchange value of data and the significance of digital value creation chains can be highlighted based on the case of smartphones.

Taking the methods of paying for apps on smartphones as an example and a pioneer of digitisation, three basic approaches can be determined:

1. Monetary payment, either on acquisition or through in-app purchases
2. Advert-financed apps
3. Payment by means of user data

In the first strategy, the classic approach to value creation – either through one-off payment or billing procedures based on time or usage – is readily apparent. These procedures have already been explained in the previous sections.

With the second method, the products themselves are free of charge, but the user is shown adverts while using them, and the manufacturer is compensated accordingly by the advertisers. Though common among smartphone apps, this approach cannot easily be transferred to machine construction and will therefore not be explored further at this point.

Finally, the third case in the context of smartphones refers to apps that are free of charge for the user and do not contain adverts, but require the user to grant the manufacturer access to certain personal data. Usage is therefore paid for through the surrender of data,

[8] See M. Porter, "How smart, connected products are transforming competition" https://hbr.org/2014/11/how-smart-connected-products-are-transforming-competition

so to speak. However, it is difficult to quantify the value of data in monetary terms, which is why this is referred to as an indirect revenue strategy. In the scope of Industrie 4.0, it can be expected that the emergence of networked systems will result in such revenue strategies also becoming significant in industrial machine construction. During machine operations, plant operators receive data that would certainly be valuable to machine builders or drive manufacturers. Based on this data, the drive manufacturer or machine builder can in turn provide services to the plant operator. From today's perspective, it is not possible to predict in detail the changes these new indirect revenue strategies will bring to the value creation chain. However, they will certainly gain significance in the future.

In summary, the following changes to the value creation process and the associated implementation strategies can be derived from Drive 4.0:

- Improvements within existing structures and business models:
  - Shorter development times thanks to "virtual" machines
  - More efficient exchanges of information in everything from engineering to after-sales services
  - Faster commissioning thanks to open standards
  - Optimised service and maintenance strategies
- Development of new business models:
  - Recurring revenue thanks to additional data-driven services
  - New services based on data acquired
  - Increased availability of plants and machines
  - Indirect revenue from data or similar sources
- Changed framework conditions due to new value creation architectures:
  - Platform architectures ushering in a new understanding of roles
  - Jointly used basic services enable profitable value creation
  - Value creation networks provide potential starting points for new data usage and additional actors

5. Drive Technology in Industrie 4.0

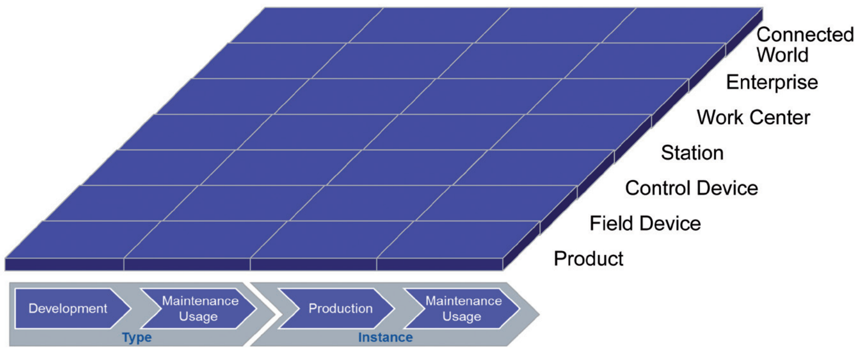
This chapter describes the specifications and standards that have already been defined for Industrie 4.0 and should be used in drive technology.

5.1 Specifications in the communication layer of RAMI 4.0

5.1.1 Introduction

The communication layer in RAMI 4.0 is intended solely for Industrie 4.0 communication. Here, communication is treated as a type and an instance throughout the life cycle (from development to production to service) and divided into logical hierarchical levels (from the product to the connected world). In this way, the communication layer is divided into small individual blocks that can be discussed on an individual basis.

Fig. 14: Communication layer of RAMI 4.0



Source: ZVEI

Here, we define communication as the pure transmission of information from a sender to a recipient. The data transferred is part of the information layer and is specified there.

An initial collection of possible candidates for the communication layer indicated that it would make sense to divide this layer further. Transparency and simpler separation of the available candidates is not that easy in a two-dimensional perspective.

For this reason, the communication layer itself was again divided based on the existing ISO/OSI layer model, and the possible candidates for preferential communication were discussed within this new model (consisting of the communication layer and the ISO/OSI layer model). Overlaps and additions can now be seen more easily.

Fig. 15: ISO/OSI layer model as a basis for the communication layer in RAMI 4.0

ISO/OSI layer	
7	Applications (application)
6	Presentation (presentation)
5	Communication control (session)
4	Transport (transport)
3	Transmission/package (network)
2	Backup (data link)
1	Bit transfer (physical)

5.1.2 Requirements of Industrie 4.0 communication

Industrie 4.0 communication takes place based on specific services and protocols that must take into account the requirements of the applications in question. The application relationships that link the Industrie 4.0 components involved from a business perspective (and the resulting functional viewpoint) also form part of the foundation for this communication. The requirements themselves are determined by the functions provided and used, and by the positioning of the Industrie 4.0 components involved both on the hierarchy axis and throughout the life cycle.

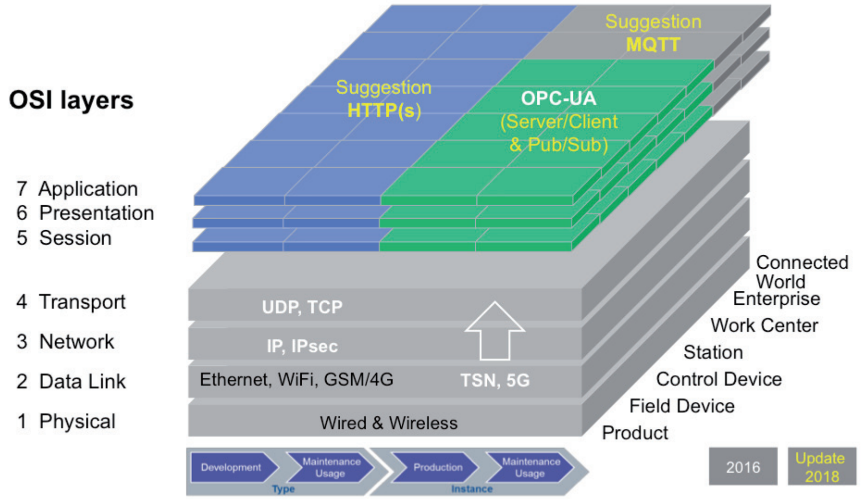
These requirements also determine which application-related services and protocols are used. If necessary, these will be negotiated by partners as they establish their interaction. For this purpose, individual communication solutions provide specifications regarding their properties (capabilities), which are compared against the requirements at hand. These properties are described based on the information models used in Industrie 4.0 and represent a sub-model of the asset administration shell of Industrie 4.0 components.

In order to select a preferred form of communication, the following requirements must be met, which arise in the scope of Industrie 4.0 communication:
- Transmission should be possible both with and without wires.
- The communication should support the service-oriented architecture.
- Services should be easy to implement. This results in a stipulation towards TCP/IP or UDP/IP transmission.
- Open and standardised.
- Existing standards should be used.
- The communication must not be tied to the hardware or software of a particular manufacturer that is advancing its own standard.
- In addition to technology, consideration of the market is important. Established standards support faster implementation of Industrie 4.0 communication.
- Specification of only one preferred communication method for each cube of the communication layer (whenever feasible). This makes implementation easier for manufacturers and customers.
- Industrie 4.0 communication should already be possible (that is, based on a selection of standards already established on the market).
- Industrie 4.0 communication should also be aligned to future requirements. This can be ensured by checking upcoming standards.
- In Industrie 4.0 communication, automation technology is also linked to IT technology. IT standards in particular should therefore also be considered.
- Transparent access should be made possible. It should become possible for everything to communicate with everything else regardless of the hierarchy levels at hand. For instance, a sensor should be able to deliver data directly to a line controller, and an ERP system should be able to send data to a field device.
- Industrie 4.0 communication must be capable of supporting the necessary security mechanisms.

5.1.3 Proposal for Industrie 4.0 preferred communication

These preliminary considerations result in the following representation of the communication layer and its individual ISO/OSI layers.

Fig. 16: Communication layer, additionally divided based on the ISO/OSI layer model



Source: ZVEI

The technologies available today with TCP/IP and UDP/IP can be used via Ethernet cables and are common on the market. For wireless communication, WLAN or existing mobile communication technologies (GSM to 4G) can be used for TCP/ICP and UDP/IP transmissions.

In the future, the upcoming standards TSN (Time-Sensitive Networking) and 5G should be kept in mind. It will be necessary to determine the effects they may have on ISO/OSI layers 3 and 4.

The bottom three layers of the ISO/OSI layer model do not need to be individually subdivided within the life cycle record and the hierarchy, as these standards are preferred throughout the field.

Differentiation then takes place in the top three layers of the ISO/OSI model. For the formation of preferred standards, these three layers were considered together.

In the area of production, OPC UA was selected as the overarching preferred standard. It does not yet meet all the related requirements, but is classified as the future standard in this area. Its high prevalence and open options for further Industrie 4.0 adjustments present further advantages. From today's perspective, it provides all the options for Industrie 4.0 communication, from products and field devices to an entire line at the work-centre level.

In addition, standards from the IT industry should be the first choice in production, but at the factory level and in the connected world. Several protocols are already prevalent on the market that are characterised by manufacturer-driven standards.

As a sub-group of WG1 for the Industrie 4.0 platform, a bitkom group is currently working on a description of a preferred standard for this area of the communication layer. The paper has not yet been published, but it appears that MQTT is establishing itself as the preferred standard.

If we look at the life cycle at the start of the development phase, there are other requirements that need to be considered. Here, the exchange of data (e.g. CAD, CAX, etc.) is the focus. Further topics include the exchange of simulation models and the exchange of data regarding the design of components, systems and machines. Here, communication is required that is more asynchronous and requires fewer low-latency (among other concerns). This communication must also be possible with optimal transparency via the Internet. For this reason, simple HTTP is suggested as the preferred transmission standard here. The specification of data and file formats is then an item for the information layer.

With this combination of three transmission methods, all the key communication topics in Industrie 4.0 can be implemented.

The communication standards that are not listed here as preferred standards have not been eliminated in RAMI 4.0; they are all located in the integration layer and can continue being used there. Such standards are important in operating today's machines and plants and will also be useful in the migration to Industrie 4.0. For example, not all the real-time requirements in production can currently be met with OPC UA. This means that a type of communication that is possible using Ethernet-based field buses (for example) is required at present. Only when these requirements can also be met using the new communication solutions listed under Industrie 4.0 can replacement be considered.
Derived consequences:

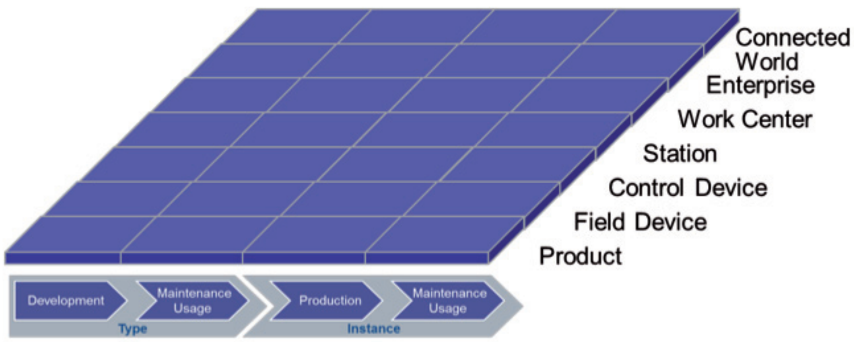
- A logical field device such as a servo inverter needs to provide its data in production via OPC UA, but in the development area its data – and the files in particular – need to be transparently accessible from within a catalogue via HTTP (for instance, as CAD, PDF or XML data files).
- A factory system is to provide MQTT for the factory-wide networking of production data. However, it should also have an OPC UA interface in order to establish communication with things that are lower in the hierarchy. For developmental planning, it should also provide its data and files in a transparent format via HTTP.

5.2 Specifications in the information layer

5.2.1 Introduction

All Industrie-4.0-compliant data is located in the information layer. The information layer is also arranged based on the life cycle phases and the hierarchy levels.

Fig. 17: Information layer



Source: ZVEI

Various specifications are necessary for the description of the data, so the information layer is divided again into individual additional layers to allow the individual specifications to be shown and explained better.

5.2.2 Requirements of Industrie 4.0 data

What are the preconditions for a plant in the form of Industrie 4.0 data in the information layer vis-à-vis proprietary data in the integration layer?

- The use of open standards
- Data includes properties and parameters, but also data records or files, etc.
- Globally unique identifiers for data in accordance with ISO 29005-2 or URI
- Description of each property in its characteristics, including its attributes (for instance, in accordance with IEC 61360 or ISO 13584-42)
- Reference to an asset or immaterial asset (to its identifier)
- The specification, description and use must be published and freely accessible to all

5.2.3 Proposal for Industrie 4.0 preference for the information layer

Data can be very diverse. It includes simple properties such as "maximum rated current", but also parameters. Corresponding attributes must be applied to the data in accordance with the IEC 61360 or ISO 13584-42 standard; these attributes can include a definition, value ranges, a data type, or a description, for example. These standardised properties can be found in IEC CDD and eCl@ss, among other places. There, the properties are also assigned to classes, so it is possible to see, for instance, which properties for a "servo motor" have already been defined and summarised. This data is defined once and reused at various points in the life cycle. For this reason, the definition of properties serves as the basis for describing data.

However, data can also be complex files such as CAD drawings or manuals in PDF format. These also have an identifier and can be described using properties; however, in addition to the properties, the file itself is transferred and/or saved. This type of data is also required throughout the entire life cycle.

At present, data must be converted at every level of the automation pyramid so that it can be transmitted using the respective communication method; ideally, however, it should always be kept the same throughout the entire hierarchy. This ensures that everything can exchange the same information with everything else regardless of where it is in the hierarchy.

If individual properties or even entire sets of properties are to be exchanged between different systems, exchange formats for data are required. Depending on the technology and application at hand, formats like XML or JSON may be preferred for this.

A few standardised structures are required to transfer data between the asset administration shells of Industrie 4.0 components – for instance, so that the Industrie 4.0 components can find one another or even automatically exchange data directly. For this purpose, ZVEI and the Industrie 4.0 platform are currently collaborating on the creation of a document on "the asset administration shell in detail".

In it, a meta model is used to describe the minimum specifications and structure an asset administration shell must have.

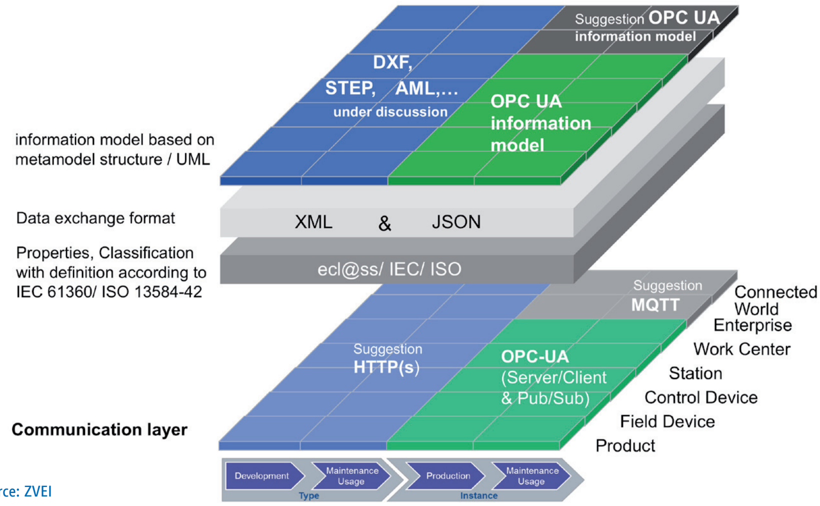
At the top level of the information layer, the technology is specified in which the entire information model must be available. OPC UA makes sense for the area of production, from products to entire lines. Based on the same communication, this technology is already widespread on the market and can cover the necessary requirements.

To ensure transparency in the area of factory communication, as well, an information model based on OPC UA is also recommended here. This will allow the same data (such as properties) to be used across the entire hierarchy, from the sensor into the cloud; the only difference is that the communication is via OPC UA in one instance and MQTT in another.
The same data is used in the development area, meaning the data always remains identical and available throughout the life cycle. Only the way it is prepared is different. In

production, the OPC UA information model is transferred via OPC UA communication. In development, however, this data should be available via HTTP so that it can be read into CAD programs (for instance). For this reason, a definition in accordance with AutomationML would be a preferred solution here. Most programs can now use this format. Information such as STEP or DXF is then embedded and transmitted. Adoption of this preferred solution in the development area has been discussed through the Industrie 4.0 platform, but it has not yet happened.

The connections among the preferred solutions discussed and the division of the information layer are shown in the following figure.

Fig. 18: Preferred solutions in the information layer



Source: ZVEI

Conclusion:
As a result, the working group defined the required data for the Industrie 4.0 Drive as properties and standardised them in eCl@ss. For the first stage of the desired demonstrator, this data was then created in an information model in accordance with OPC UA and exchanged using OPC UA communication.

5.3 Meeting the Industrie 4.0 product criteria

The two working groups ZVEI Industrie 4.0 Standards & Models and WG1 Reference Architectures and Standards (Industrie 4.0 platform) have specified the criteria products must meet to be Industrie-4.0-compliant today. At the same time, they provided an indication of the criteria that could be added in the next five and ten years.[9]

This chapter presents the general product criteria for 2018 – and how they relate to this white paper for electrical drive technology. Agreed eCl@ss properties can be found in the classification “27-02 Electrical Drive” of eCl@ss.

In total, seven criteria must be met, which are described with their requirements and the necessary product properties for 2018.

[9] See https://www.zvei.org/presse-medien/publikationen/welche-kriterien-muessen-industrie-40-produkte-erfuellen-2-ueberarbeitete-version/

I. Identification

	Criterion	Requirements	L	E	Product properties 2018
1.	Identification	Cross-manufacturer identification with unique identifier (ID) attached to the product, electronically readable. Identification in: 1) Development 2) Goods traffic (logistics), production 3) Sales, service, marketing 4) Network	T	M	For 1) material number (electronic) in accordance with ISO 29002-5 or URI
			I	M	for 2) Serial number or unique ID for 3) Manufacturer + serial number or unique ID with 2) and 3) electronically readable, for physical products via 2D code or RFID for 4) Identification of participants via IP network

In the white paper, specific eCl@ss properties were defined that can be used for identification.

Properties:	eCl@ss identifier
URI of the product:	AAY811
Manufacturer name:	AAO677
Serial number:	AAM556
Manufacturer article number:	AAO676
Manufacturer product type:	AAO057

II. Industrie 4.0 communication

	Criterion	Requirements	L	E	Product properties 2018
2.	Industrie 4.0 communication	Transfer of product data and data files for interpretation or simulation, for example; product data in standardised form	T	M	Manufacturer makes data available/accessible online in digital form. The data should be relevant for the customer and available/accessible with the aid of the identification, e.g. PDF via HTTP(S) and URI
		The product can be addressed via the network; supplies and accepts data; plug-and-produce via Industrie-4.0-compliant services	I	M	The asset administration shell of the product can be addressed online (at any time) via TCP/UDP&IP with the aid of the identification with at least the information model from OPC UA

Many properties are already available as of eCl@ss version 10.1 that can be exchanged regardless of the manufacturer in question. As of version 11 (from 2019), further features will be made available that should be used here.

For example, the “equivalent circuit diagram” block (0173-1#01-AGG318#001) with its technical properties: number of poles, motor type, wiring of the motor winding, insulation resistance, torque ratio, power factors, etc.

These eCl@ss data structures are also included in the OPC UA Companion Specification of the Joint Working Group Industrie 4.0/OPC UA Drive Technology, which involves VDMA and the OPC Foundation; they are being tested using the demonstrator described in Chapter 7.

III. Industrie 4.0 semantics

	Criterion	Requirements	L	E	Product properties 2018
3.	Industrie 4.0 semantics	Standardised data in the form of properties with a unique, cross-manufacturer identifier and syntax for the following (for example): 1) Commercial data 2) Catalogue data 3) Technical data: mechanics, electronics, functionality, location, performance 4) Dynamic data 5) Data regarding the life cycle of the product instance	T	M	for 2) catalogue data can be accessed online in an open standard
			I	M	for 2) catalogue data can be accessed online
				O	for 5) data regarding the life cycle of the product instance can be accessed online

With eCl@ss version 10.1, there is already a range of properties for catalogue data that should be used here. Further properties will be added with the new version 11 (from 2019).

IV. Virtual description

	Criterion	Requirements	L	E	Product properties 2018
4.	Virtual description	Virtual representation in Industrie-4.0-compliant semantics Virtual representation throughout the entire life cycle. Characteristic attributes of the actual component, information regarding relationships among the attributes, relationships relevant to production and related processes among Industrie 4.0 components, formal description of relevant functions of the actual component and its processes	T	M	Relevant information for customers can be accessed digitally based on the type identification (product description, catalogue, image, technical features, data sheet, security properties, etc.)
			I	M	Digital contact to service and product support information (incl. on spare parts) possible from the field

The relevant information for customers has not yet been standardised, meaning every manufacturer can provide the currently available information in the form of files, such as PDFs. These should be accessible via the Internet, along with a means of contacting service and support.

However, initial properties for the provision of potential replacement parts have already been discussed and specified. These have not yet been submitted to eCl@ss, but will be included in one of the next versions.

In the future (read: the next five years), standardised catalogue properties can also be expected for the product criteria. We have already defined these in eCl@ss for electrical drive technology, meaning they can be used.

V. Industrie 4.0 services and statuses

	Criterion	Requirements	L	E	Product properties 2018
5.	Industrie 4.0 services and statuses	Definition still open (service system) General interface for loadable services and status reports. Essential basic services that an Industrie 4.0 product must support.	T	O	Digital description of the device interface available
			I	O	Information such as statuses, error messages, and warnings available via OPC UA information model in accordance with an industry standard

In the context of the white paper, initial cross-manufacturer functions were also defined, including a function for "fault memory and warnings". This function collects the error messages and warnings from all devices in a standard form and presents them in a clear, manufacturer-neutral fashion. The properties are defined in eCl@ss and an initial version of the function will be implemented on the demonstrator using OPC UA.

VI. Standard functions

	Criterion	Requirements	L	E	Product properties 2018
6.	Standard functions	Basic standardised functions that run on various products regardless of manufacturer and provide the same data in the same functions. These serve as the foundation for the functionality on which all manufacturers can build their own enhancements.	T	N	Not defined
				N	Not defined

Today's product criteria do not yet require standardised functions. With the white paper, however, we are beginning to introduce initial functions that are to be kept general enough that they can quite easily be applied to other automation products despite their origins in drive technology. Standardisation of the interface for these functions is also being planned via eCl@ass, and preliminary discussions have taken place.

VII. Security

	Criterion	Requirements	L	E	Product properties 2018
7.	Security	Minimum requirements to guarantee the security functionality.	T	M	A threat analysis has been performed. Appropriate security capabilities have been considered and publicly documented
				M	The existing security capabilities are documented. Appropriately secure identities are available.

Security must not be omitted from the key criteria for Industrie 4.0 products. We do not yet have any key figures in the form of properties that we can provide across manufacturers, so each individual manufacturer is responsible for examination and documentation.

Summary for product criteria

Many parts of the white paper aid manufacturers, machine builders and plant operators in improving the digitisation of electrical drive technology products. This standardises and simplifies processes in development/design and commissioning, for instance, but also in operations and service.

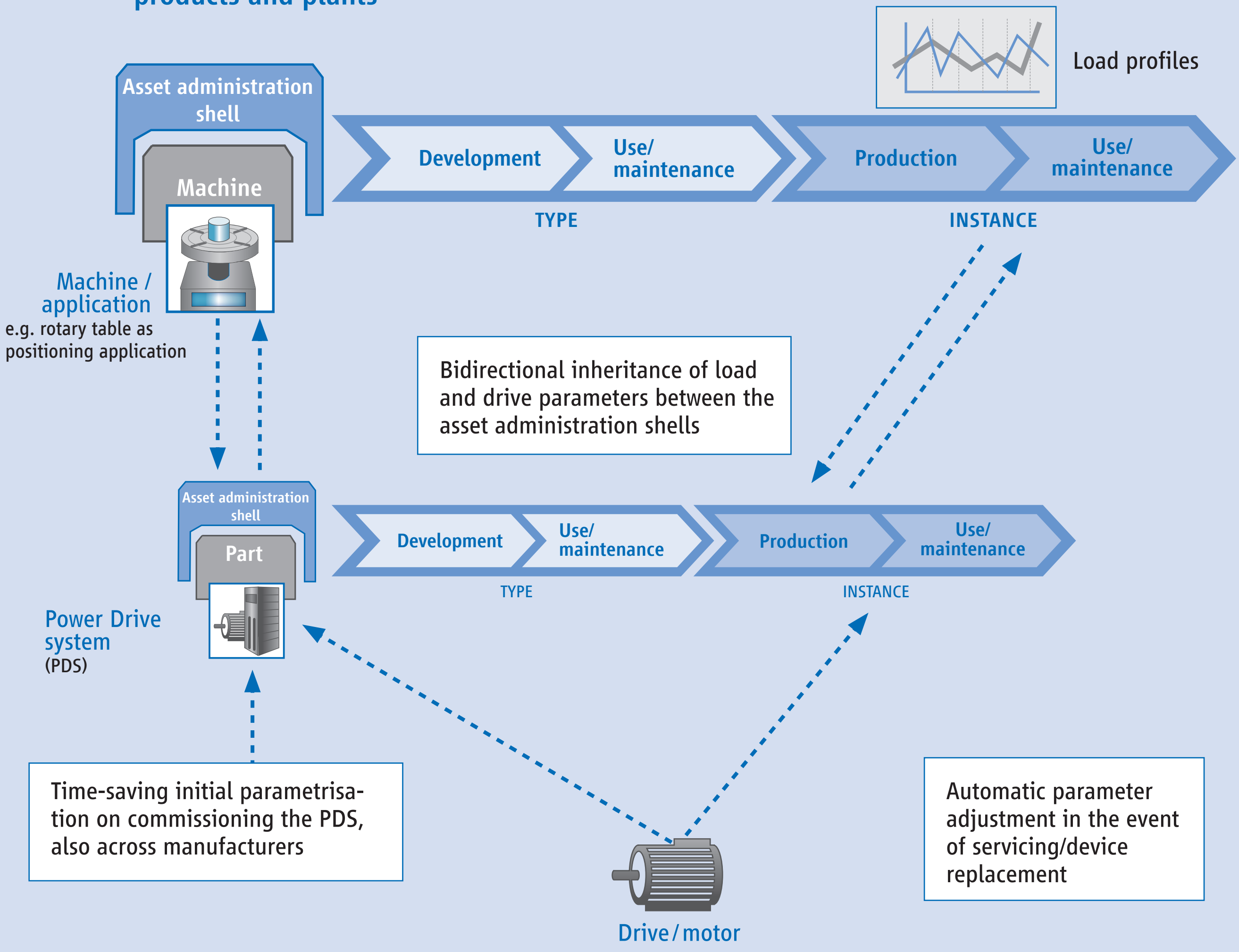
6. Drive 4.0

6.1 Properties

The attributes of devices and components (e.g. a motor and converter) are described in the asset administration shell using properties that are based as far as possible on definitions from global or regional standards (ISO, IEC, EN, DIN, VDE, etc.).

Throughout the value creation process, the properties will develop and sometimes change, including in the design, manufacturing and development phases; see Chapter 4, "User groups in the value creation process". As RAMI 4.0 intended, this corresponds to consideration of the "type" (with typical values for certain properties) and "instance" (with values that can be clearly assigned to the individual asset for certain properties that are the same). The assignment of properties to technical groups such as suppliers and users of data makes it possible to quickly identify the data that may be relevant in each phase.

Fig. 19: Provision of information for electrical drives throughout the life cycles of products and plants



Source: ZVEI

However, despite the efforts of various bodies, the details of the properties differ. Along with potential deviations in physical units (e.g. [H] for Henry or [mH] for millihenry), above all there are differences in the data formats used, which are generally not standardised (integer, floating point figure, number of decimal places, etc.), and sometimes terms are used without a reference to the public standards. The working group chose the eCl@ss classification system, which already defines properties across various areas of drive technology, as a basis for initial standardisation. For motors, classes are used that follow IEC definitions.

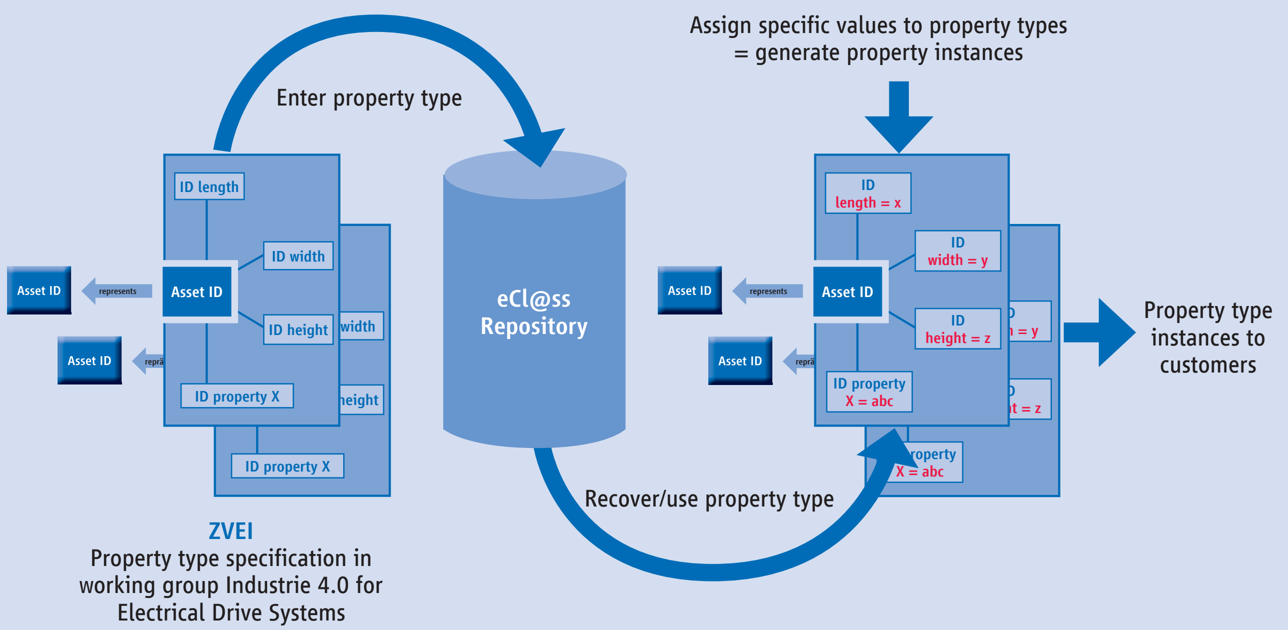
These eCl@ss classes can be understood as dictionaries that define possible properties and offer value lists and translations of both the properties and the value constellations into numerous languages. The extent to which these properties are used and therefore have to be filled depends on the application at hand and is agreed between the requester (e.g. a machine builder) and the supplier (e.g. a motor or converter manufacturer).

A catalogue of properties is currently being developed and will later be available to drive manufacturers, machine builders and operators for use and maintenance.

"The intelligent manufacturing networks of the digital factory can only become a reality with automatically standardised information exchange formats. They must enable a secure, reliable and error-free flow of data across various systems (ERP, PLM, MES, logistics, production automation, etc.); ideally, it should also be possible to deploy them across companies and (even) industries."[10]

To ensure that the properties of electrical drives are clearly characterised in the information world, they also require unique identifiers to clearly classify their significance. As these indicators are independent of language, they also simultaneously allow the properties to be translated into multiple languages. For this purpose, eCl@ss provides the basis for retaining property types in repositories in accordance with ISO/IEC 6523 and ISO/IEC 11179-6 (based on ISO29002-5). The ZVEI working group has added the properties that are missing and need to be revised for the key drive classes and is now adding them to eCl@ss, where they are to be published with version 11 in the first half of 2019. The objective is to make these properties available to every user so that a concrete instance description can be generated through the assignment of KPIs and characteristics of the user's drive product. Furthermore, users ultimately gain access to information on such products in a standardised data structure.[11]

Fig. 20: Generation of a property



Source: eCl@ss/ZVEI

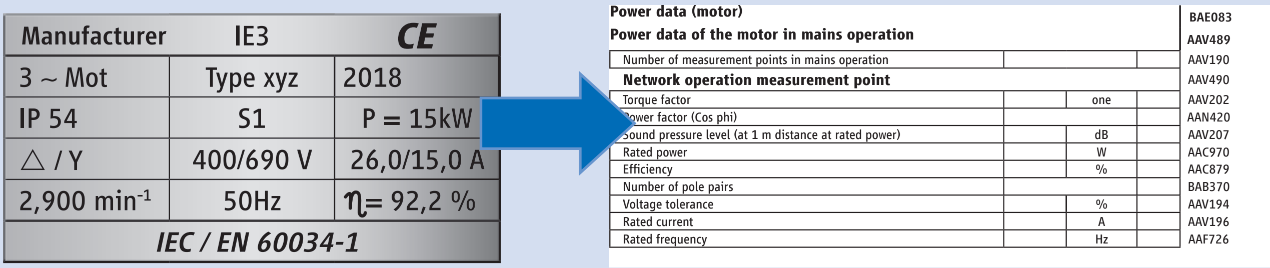
[10] See https://www.eclass.eu/fileadmin/downloads/ecl-Whitepaper-Industrie40_DE_klein.pdf
[11] See http://wiki.eclass.eu/

Example of an electronic rating plate

One of the first implementation examples of Drive 4.0 is the digital representation of rating plates (which are mandatory for all drives) via eCl@ss.

The realisation of the “electronic rating plate with Industrie 4.0 functionality” is to be trialled on a virtual drive train with (among other things) unique identification of the motor, including instance information such as the serial number and wiring of the motor. Here, the eCl@ss data model also serves as a template for other motor types – for instance, for subject group 27-02-26 servo motors, for which the properties and property blocks are currently being expanded to include specific attributes.[12]

Fig. 21: Transferring a rating plate to eCl@ss



Source: ZVEI

The rating plate data is described in properties and mapped using structural elements, which can, for example, be summarised in a property block with corresponding cardinality to show the power data (switching mode, voltage, frequency, power factor, etc.). The value ranges are assigned individually.

Here, “cardinality” means that one property block (voltage, frequency, power, torque, etc.) is created for each different measurement (e.g. the different torque speed operating points of an inverter-fed three-phase motor).

In the future of Drive 4.0, properties in the described form will no longer be used only for static information that is already specified through the design and manufacture of the drive, but for all data throughout the life cycle of the drive, right up to real-time processes. Identifiers are also to be defined for these properties in a way that will make it possible to transmit and understand values (e.g. the time progression of a current) regardless of the devices involved.

The structure of the classes and basic properties in the current version (currently 10.1; from mid-2019, version 11) can be accessed online.[13]

The complete list of all properties is referred to as “eCl@ss ADVANCED” and can be opened using the relevant link on the specified page. This link leads to the “Content Development Portal” (“CDP”). This can also be accessed without an eCl@ss membership; however, you have to register in order to log in.

6.2. Industrie 4.0 functions

6.2.1 General information

Building on the properties and data handling, the ZVEI working group has developed initial proposals for Industrie 4.0 functions. For this, functions were selected that are particularly significant for machine builders and plant operators and are often already integrated into

[12] See http://www.eclasscontent.com/index.php?id=27022101&action=det&searchtxt=&options=&version=10.1&language=en

[13] See also: http://www.eclasscontent.com/index.php?id=27020000&version=10_1&language=en#node_27020000

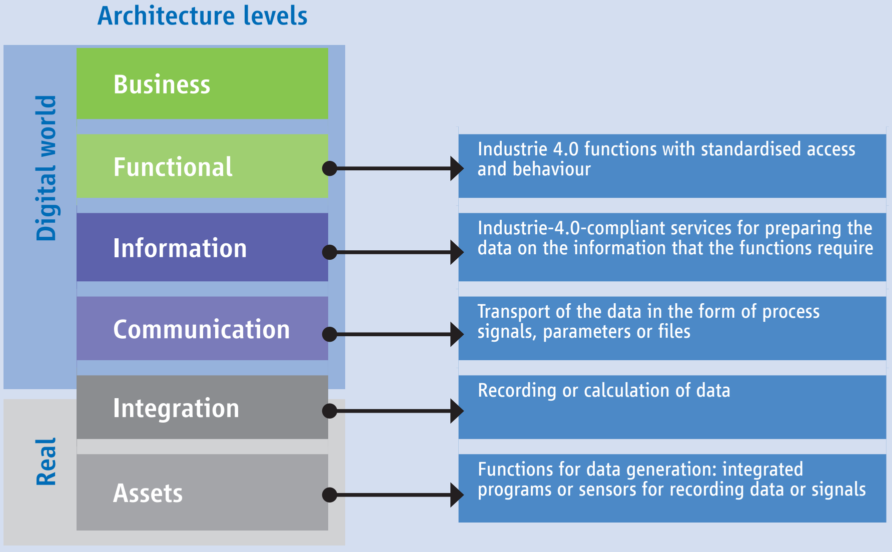
drives (in manufacturer-specific versions). A key advantage of a manufacturer-independent description and access layer is that machine builders and plant operators do not require a separate tool for every drive from every manufacturer. The standardised interfaces allow third-party providers of software to bring specialised tools to the market. Thanks to its standardised presentation, the data acquired from different sources (drives and other machine functions) can be complied and evaluated. This opens up new possibilities to analyse and optimise the machine at hand in its entirety.

It also presents a further benefit: while the selection of drives was previously often limited to the products offered by a machine builder's preferred manufacturer, standardised Industrie 4.0 functions will increasingly allow the most suitable drive to be selected regardless of the manufacturer in the future. It is also easier to switch drive manufacturers, which stimulates competition to the benefit of machine builders.

The Industrie 4.0 functions are located in the function layer of RAMI 4.0. They are independent from the asset and manufacturer in question, and from the communication infrastructure used. The function layer is tasked with making the functions it contains available in a standardised manner and methodology. This methodology includes access to the information, the semantics (meaning) of the information and the internal logic of the function, i.e. the manner in which the information is formed from the defined data.

The location of the Industrie 4.0 functions in RAMI 4.0 is shown in graphical form in Figure 22.

Fig. 22: Location of Industrie 4.0 functions in RAMI 4.0



Source: ZVEI

In principle, there are two different types of functions:

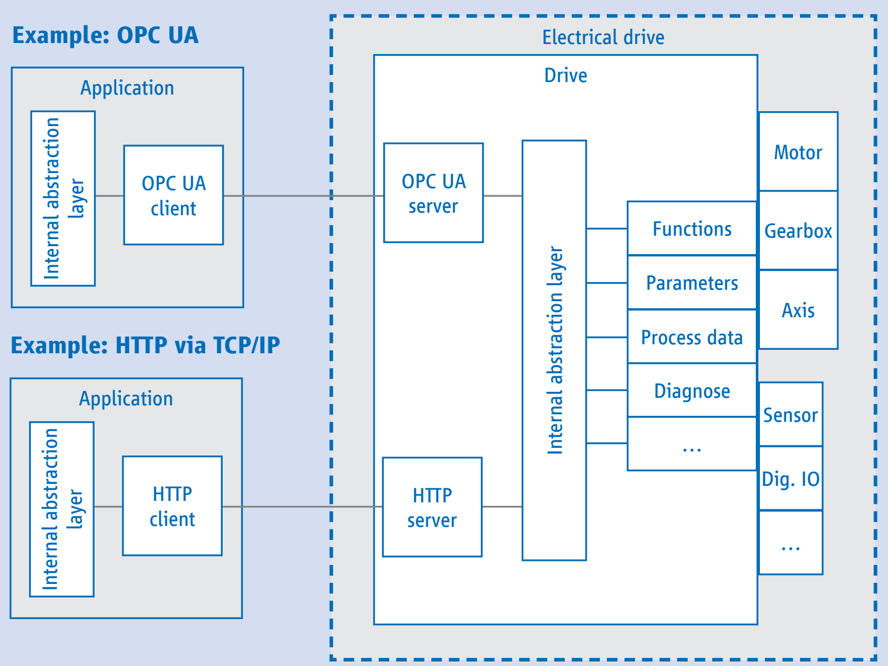
1. Functions that take place in assets because certain real-time data is available there or the functionality itself is required (independent from the Industrie 4.0 context). For these functions, a representation is formed in the function layer, which enables standardised access. One key task is the conversion of manufacturer-specific data to the standardised Industrie 4.0 format.

2. Functions that run in the function layer itself as programs. Here, assets primarily deliver data that is compiled in the integration layer, transported via the communication layer and pooled in the information layer with any other data that is required to execute the function (if applicable). The actual processing of the data for the required output in Industrie 4.0 format is then performed in the function layer.

The standardised Industrie 4.0 format for an output signal includes at least an identifier (name), data format, unit and system time. The system time serves to create a temporal relationship between different signals from different actuators and sensors. End-to-end use of a real-time clock (RTC) would be a suitable candidate for the system time. Alternatively, an exact time can be obtained using Time-Sensitive Networking (TSN). In this case, the general Precision Time Protocol in accordance with IEEE 1588 is used for the synchronisation of networks.

Figure 23 uses an "electrical drive" asset to show how an abstraction layer integrated into the asset can be used to provide the data available in the device regardless of its manufacturer. Here, OPC UA and HTTP are named as possible communication methods for providing the data. The functions are executed or the functionality of the drive is mapped in a manufacturer-independent manner as part of the application. Standardised access to the functions also takes place via an abstraction layer, which is executed in the client.

Fig. 23: Abstraction layers for providing information and functions

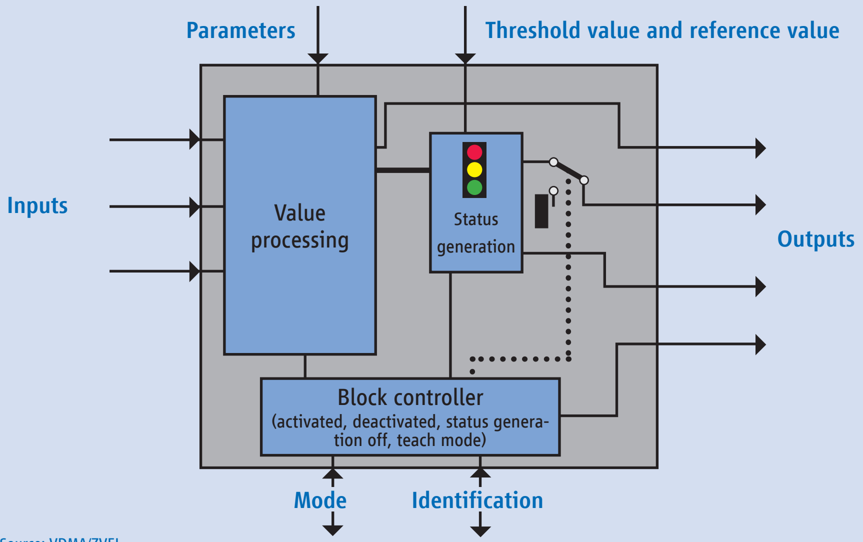


Source: ZVEI

The descriptions of the cross-manufacturer functions in the following chapters are aligned to the function blocks following the example of the VDMA specification[14] for condition monitoring (see Fig. 24). This standardised presentation ensures that the interfaces and data relevant to the function in question (regarding input, output or reference values, for example) are clearly assigned regardless of the function's purpose.

[14] See VDMA 24582 "Fieldbus Neutral Reference Architecture for Condition Monitoring in Factory Automation"

Fig. 24: General block diagram for Industrie 4.0 functions



Source: VDMA/ZVEI

Each function can be viewed as value processing that is fed the data to be processed at its inputs and then produces the result at its outputs. The function is called up via an enable input. The user can influence the processing via parameters and threshold values; the functionality can also depend on the selected mode and the system state. The output of the function can be compared with the values of the accessing page with the aid of reference values. Along with the result of the value processing, each function delivers a status that provides information on the validity of the output; it is also output in the form of a traffic light status. Each function has its own set of inputs, parameters and other aspects that are described in the following chapters. However, there are some interfaces that are present in most or all of the functions; these are listed here:

Enable (input)	Starts the function.

System state (input)	The system state can influence the output of the function or block the execution of the function. Possible values in accordance with the VDMA specification:
	• Fault
	• Ready for operation
	• Operating mode
	  • Automatic operation
	  • Manual operation
	  • Set-up operation
	  • Start-up operation
	  • Stationary operation
	  • Parameter-setting operation

Mode	The mode can be used to influence the behaviour of the function. Possible values in accordance with the VDMA specification: • “Activated” (function is executed) • “Deactivated” (function is not executed) • “Manual” (data fed in from external sources is processed, or the initial values, status word and traffic light status are written from an external location for test purposes)

	• "Simulation" (input values are ignored; the actual value planning is performed based on simulation values specified externally or generated elsewhere) • "Status generation off" (output values are calculated, but no status is formed; traffic light status is "no condition statement") • "Learning mode" (input values are fed in to form reference and/or threshold values in a learning process; traffic light status is "no condition statement") • "Reset" (all internal data such as interim values and learned reference and threshold values are reset; traffic light status is "no condition statement")
Language (parameter)	Selection of the language for plain text messages in the form of a priority list (e.g. "DE, EN"). If none of the desired languages is available, the function provides a default language (generally defined on a manufacturer-specific basis).
Time stamp (reference value)	The accessing page transmits the system time to the function as a reference value, allowing the function to adjust time-related data and events and return them in the system time.
Status word (output)	Statement regarding the validity of the output; possible values are: • "Valid" (function delivers valid output data or has delivered valid output data) • "Invalid" (function works successfully but does not (yet) deliver output data) • "Faulty" (the function failed; for details, see the "error status" output) • "Ready" (function can be parametrised or started) • "Waiting" (function was activated, but is waiting for a trigger condition to occur) • "Alarm" (function is active and a threshold value was reached)
Traffic light status (output)	Summarising statement regarding the condition; possible values are: • Off/no condition statement (e.g. in "Status generation off" mode) • Green/good, OK (status is "Valid", "Ready" or "Waiting") • Amber/warning (status is "invalid" or "alarm") • Red/error, defect, critical condition (status is "faulty")
Error status (output)	Additional information for the "Faulty" status: • Parameter error (the assigned parameters are incorrect, invalid or impermissible) • Runtime error (an error occurred while executing the function) • Return error (an error occurred while outputting the result) • Function or mode not available
Time stamp (output)	The function returns the time for which the condition statement applies. This is based on the shared system time.
Language (output)	Language of the returned plain text messages.

According to the VDMA specification, maintenance strategies can also be specified for functions if this is appropriate and sensible.

The typical user of a function generally depends on the task that is realised with the function. However, at present there are no further details in the function description regarding the extent to which there should be restrictions that only allow access to certain groups of people. Corresponding authorisation mechanisms are currently being explored in the ZVEI project group on "the asset administration shell in detail" and may be included in the definition of the functions presented here in the future.

6.2.2 Oscilloscope

The term "oscilloscope" refers to the characteristic of recording real-time signals of a drive system and making them available to "higher" architecture levels outside of the "drive controller" asset. The recording of a signal takes place as a sequence of amplitude values, with each amplitude value being assigned a time stamp. Generally speaking, the amplitude values are scanned at equidistant time intervals (as with a digital storage oscilloscope).

Here, real-time signals are understood to be quasi-analogue (for example, motor phase current or intermediate circuit voltage) and digital signals (for example, quick stop or pulse inhibition), which are recorded in the temporal resolution on which the drive system is based. Typical cycle times in the drive system could, for example, be 1 ms for motion control, 250 µs for the speed control circuit and 62.5 µs for the current control circuit.

Every oscilloscope signal consists of a signal type and a number of scan values. The number of scan values is unlimited.

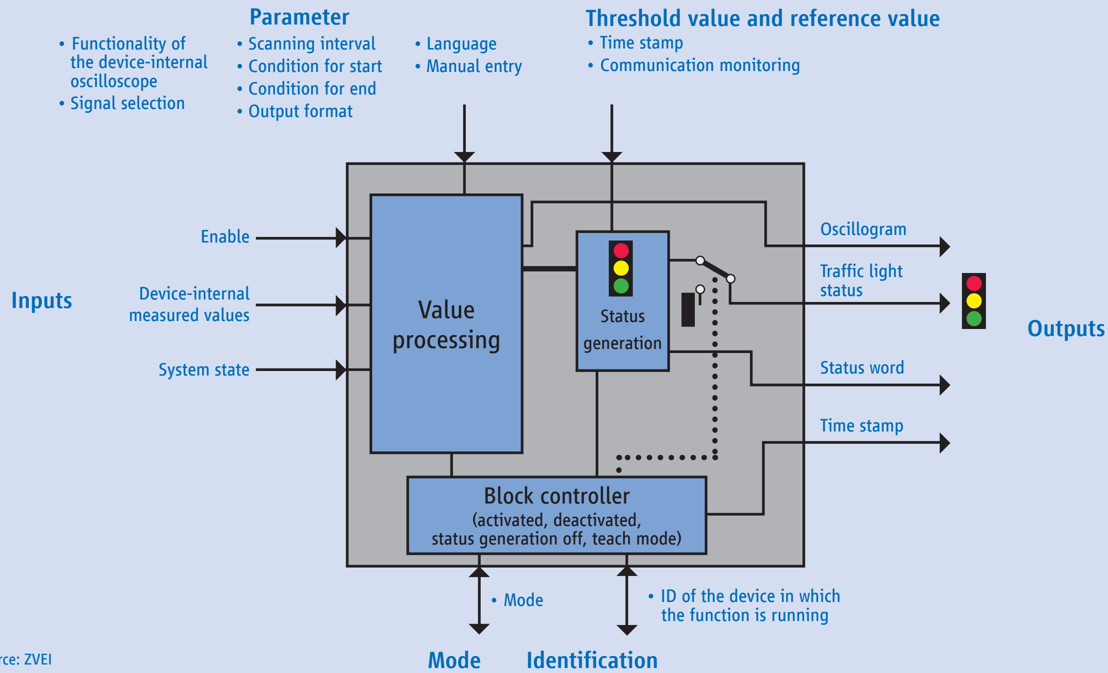
The signal type of an oscilloscope signal has the following attributes:
- Identifier (e.g. Motor_Current_Phase_U)
- Data type (e.g. INT32)
- Quasi-analogue signal or binary signal
- Unit (e.g. A, V, W)
- Measurement tolerance (e.g. in LSB)

Each scan value of an oscilloscope signal consists of:
- Algebraic sign (with passive sign convention, not with binary signals)
- Numeric value (pre-decimal and post-decimal positions according to the data type)
- Scan time based on the specified system time (real time or plant time)

A standardised oscilloscope allows faster commissioning of drives, machines and plants. Here, the benefits lie in standardised handling, the ability to select relevant/required information, the definition of specific trigger conditions, and the superimposition of information from various subsystems with the correct time.

During operation, the oscilloscope can be used to create the process signature of a drive, machine or plant and evaluate the signature online. In the event of deviations, the operator can intervene before an error occurs to maintain secure operation, avoid downtime and prevent major damage.

Fig. 25: Block diagram of oscilloscope Industrie 4.0 function



Source: ZVEI

Description of the function:

- Output of device-internal signals that the user previously selected
- The device-internal signals are prepared so that they are available at the output in a manufacturer-independent form
- The signals consist of a scan value, time stamp (system time) and unit
- The signals can be provided as a complete oscillogram (once recording is complete), continuously (as a cyclical data stream) or on request (using a handshake procedure).

The function can be accessed in every system state of the machine/plant.

Interface description:

In addition to the standard interfaces (see Chapter 6.2.1), the oscilloscope function uses the following interfaces:

Enable (input)	➡ See Chapter 6.2.1
System state (input)	➡ See Chapter 6.2.1
Quantity (parameter)	The signals (real-time signals and non-real-time signals, each with a value, unit and scanning time) specified via the parameter interfaces are read in at this function input
Scope (parameter)	Using the parameters channel, the device can inform the Industrie 4.0 oscilloscope about the measured values available and their quality • Number of channels that can be recorded simultaneously • Size of the memory area (total) • Minimum scan rate • Time reference and internal resolution • List of process signals (real-time signals) with data type, standardisation and unit • List of parameters (as NRT signals) with data type, standardisation and unit • List of trigger signals (if different from process signals)

Language (parameter)	Determines the input data that is processed in the function (as a basis for the recording of measured values in the drive controller)
Manual entry (parameter)	Determines the cycle time for the input data (as a basis for the recording of measured values in the drive controller)
Number (threshold value)	Specifies when recording will begin within the oscilloscope function: • Via the enable input • Time (based on system time) • Trigger signal(s) with link, reference value, edge and lead time
Time stamp (threshold value)	Determines when recording will end for the oscilloscope function: • Automatic (no enabling) • Time (based on system time) • Trigger signal(s) with link, reference value, edge and follow-up time • Upon reaching data volume X • Upon reaching a maximum number of measured values Y
Time stamp (reference value)	The oscillogram can be output as: • A file (data matrix) • A cyclical data stream • Acyclically via a handshake procedure
Fault memory/logbook (input)	➡ See Chapter 6.2.1
Mode	Add a manual entry to the oscillogram as a signal plot if "Manual" mode is implemented (see below).
Fault codes/messages (output)	Monitoring of input signals, parametrising channel and output data. This includes the specification of return formats, response times, etc.
Scope (output)	➡ See Chapter 6.2.1
Status word (output)	➡ See Chapter 6.2.1
Traffic light status (output)	ID of the device in which the function is running
Error status (output)	Provision of the oscillogram that was generated from the signals read in: • As a file with a signal identifier, length and data sequence (each as value + unit + scanning time). The file can container several signal identifiers (channels) • As a sequence of individual data, with each piece of data having a signal identifier and a value + unit + scanning time being queued for defined, synchronised scanning intervals • As a sequence of individual data, with each piece of data having a signal identifier and value + unit + scanning time being queued until it is called up
Time stamp (output)	➡ See Chapter 6.2.1
Language (output)	➡ See Chapter 6.2.1
Error status (output)	➡ See Chapter 6.2.1
Time stamp (output)	➡ See Chapter 6.2.1
Language (output)	➡ See Chapter 6.2.1

For the oscilloscope function, the maintenance strategies (application scenarios) listed below arise during the life cycle of a plant:

During commissioning:
- Triggers for certain events or error patterns during commissioning
- Recording of "good samples" as a reference for later operation

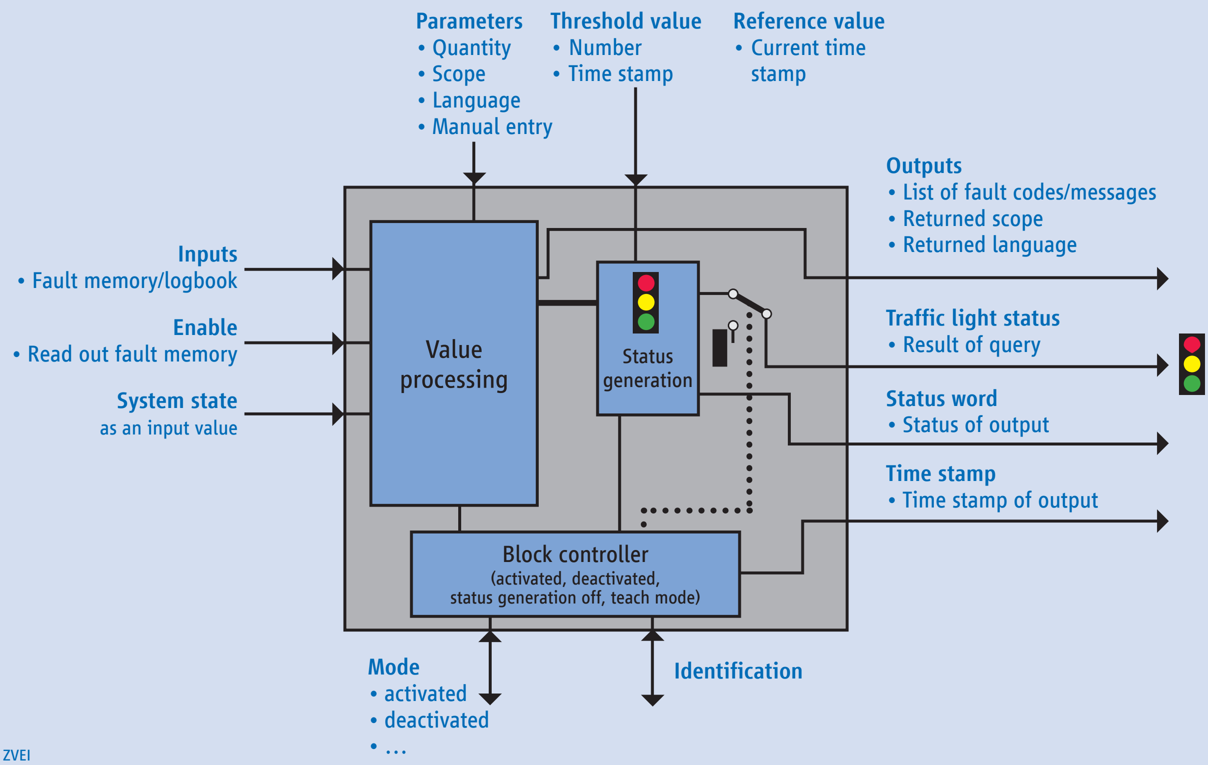
During operation:
- Continuous recording of measured values during operation
- Detection of deviations from specified references
- Automated detection of changes in repeated/cyclical processes
- Detection of unexpected events or their accumulation

In the event of servicing:
- Determination of fault causes
- Triggers for certain events or error patterns
- Access to operational state prior to fault

6.2.3 Fault memory/warnings function

During commissioning or operation, messages regarding events or operating conditions may occur. These range from information (e.g. reference run events, switching of operating mode) to warnings (e.g. overload/current limit or position error) and faults (e.g. limit switch reached or motor blocked). A standardised interface can help users access these messages in the higher-level control or when using remote access.

Fig. 26: Presentation of the manufacturer-independent function "fault memory/warnings"



Source: ZVEI

Description of the function:

- Reads out the device-internal fault memory or logbook
- Selects the entries to be output based on the user's specifications (quantity, scope, language)
- Adjusts the time stamp to the system time, if necessary
- Converts the list into the standardised cross-manufacturer form and outputs it
- Provides further output information at the outputs
- Forms and outputs the status and traffic light status

A manual entry is added to the fault memory/logbook or the list is deleted depending on the mode at hand.

The function can be accessed in every system state.

Interface description:

In addition to the standard interfaces (see Chapter 6.2.1), the fault memory/warnings function uses the following interfaces:

Enable (input)	➡ See Chapter 6.2.1
System state (input)	➡ See Chapter 6.2.1
Quantity (parameter)	Affects the length of the returned list; possible values: • "Read out last X entries": returns a specified number of entries. • "Read out all entries since [time stamp]": returns only entries that are no older than up to a specified time.
Scope (parameter)	Determines which types of entries are returned: • "Faults only": this is the default setting. • "Faults and warnings": warnings that do not result in faults are also output (if implemented). • "Faults, warnings and notes": also output notes if implemented).
Language (parameter)	➡ See Chapter 6.2.1
Manual entry (parameter)	Add a manual entry to the list of messages if "Manual" mode is implemented (see below).
Number (threshold value)	If quantity = "Read out last X entries".
Time stamp (threshold value)	If quantity = "Read out all entries since [time stamp]".
Time stamp (reference value)	➡ See Chapter 6.2.1
Fault memory/logbook (input)	The source for the messages to be output; a list that is not defined in more detail with time stamps, fault codes and further data (if applicable). This data is currently implemented on a manufacturer-specific basis, sometimes only partially or even not at all.
Mode	Depending on the mode, the following methods are available: ➡ "Activated": delivers real data ➡ "Deactivated": no recording (e.g. to ensure that an expected accumulation of faults is not saved unnecessarily) ➡ "Manual": add a manual entry to the fault memory/logbook (see "manual entry" parameter above) ➡ "Simulation": o utput a list with artificially generated entries for test purposes ➡ "Reset": delete list (this can also be realised without actually deleting the entries, e.g. through selections or filters)

Fault codes/messages (output)	List with the required entries. Every entry consists of a time stamp, an ID for the type (e.g. fault, warning, note), plain text, and an optional fault code that can be specified based on a standard or a particular manufacturer.
Scope (output)	If the output does not meet the user's specification or was different – for example, because the device does not store notes or warnings
Status word (output)	➡ See Chapter 6.2.1
Traffic light status (output)	➡ See Chapter 6.2.1
Error status (output)	➡ See Chapter 6.2.1
Time stamp (output)	➡ See Chapter 6.2.1
Language (output)	➡ See Chapter 6.2.1

Maintenance strategies:
The following usage options are available for the fault memory/warnings function:
- Condition monitoring during operation; for example, detection of unexpected events or their accumulation
- In the event of servicing: Access to operational state prior to fault or maintenance

6.2.4 Auto-tuning function

Modern drive tools guide users quickly and easily through the individual steps of commissioning or optimising the PDS (Power Drive System). The information required for this can be generated from a motor database, an electronic rating plate or, in the future, from the contents of the asset administration shell. It is no longer necessary to enter individual motor parameters.

The auto-tuning function ensures that all the information relevant for the PDS is recorded, analysed and optimised. The manufacturer-independent standardisation of the data and also the auto-tuning function result in the desired flexibility, meaning in terms of optimising components from various manufacturers. The goal must be to avoid time-consuming adjustments of systems or system interfaces to manufacturer-specific data structures. The auto-tuning function will help both machine builders and users to optimise machines and plants during commissioning, maintenance and production.

Control and pre-control parameters are automatically determined using the auto-tuning function.

This function can be mapped in the drive or in a higher-level intelligence. Auto-tuning means optimising the drive's control behaviour through automatic comparison of the physical characteristics of the motor and converter. With all relevant motor parameters available to it, the function is able to not only make the motor rotate, but also provide options for analysis, diagnosis and optimisation of the PDS.

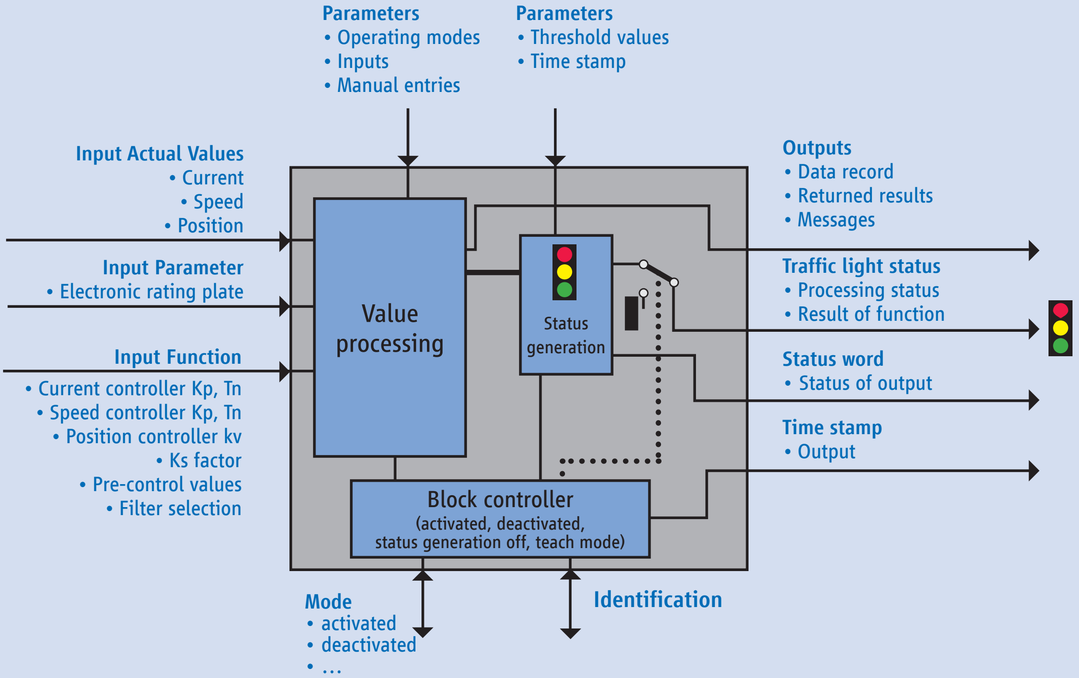
Any optimisation results, parameter information, notes, messages or warnings are transmitted via a standardised interface of the higher-level intelligence.

With the aid of the auto-tuning function, it is possible for the PDS to set and/or optimise itself by reading in and analysing relevant target and actual values. The basis for this is a function block similar to that of the VDMA specification on condition monitoring.

The value processing, block controller and status generation blocks form the basic structure. There is also the option to expand the function with application-specific settings, such as "run clockwise/anticlockwise" or "drive must not rotate".

The results are available as status information and data set information and can be used via the Industrie 4.0 communication methods.

Fig. 27: Presentation of the manufacturer-independent function "auto-tuning"



Source: ZVEI

Description of the function:

- Reads in the specific motor parameters from an electronic rating plate, a motor database from the manufacturer, or from the contents of the asset administration shell
- The function (e.g. optimisation of position, speed or current controller) is to be defined based on the user's specifications (operating types, modes, manual entries, language)
- If applicable, the time stamp must be adjusted to the system time (e.g. for multi-axis systems)
- The results must be converted into a standardised, cross-manufacturer format and output
- Further information on the results of the auto-tuning is available at the outputs for further use
- The status can be output via a traffic light function in the same way as in the specification

As an alternative to the auto-tuning function, manual entries can also be added or commented depending on the mode at hand.

The function can only be called in defined system states.

Use of the function:

Some possible applications of the auto-tuning function are:

- Automatic optimisation of a PDS or PDS groups
- Simulation of application conditions
- Automatic commissioning
- Service
  - Diagnostic function in the event of a fault
  - Diagnosis of sub-systems (PDS coupled with mechanics)

Interface description:

In addition to the standard interfaces (see Chapter 6.2.1), the auto-tuning function mainly uses the following interfaces:

<table>
<tr><td>Enable (input)</td><td>➡ See Chapter 6.2.1</td></tr>
<tr><td>System state (input)</td><td>➡ See Chapter 6.2.1, Operating mode</td></tr>
<tr><td>Operating mode (parameter)</td><td>Depending on what is required of the application, the following options should be possible as operating modes:
• Fully automatic function
• Partially automatic function
• Simulation
The selection can be made by setting a flag in the block controller. Setting the flag defines how the function is executed.
"Read out all entries since [time stamp] or upon function start" returns only entries that are no older than a specified time or that were recorded following a start signal.</td></tr>
<tr><td>Inputs (parameter)</td><td>The types of entries that are required to implement the function are specified:
• Data from an electronic rating plate, the asset administration shell or another database
• "Current parameter list": should be the default setting
• Comparison of parameter lists before and after executing the function
• "Notes, faults and warnings": also output warnings that do not result in faults (if implemented)
• Threshold value information
• Oscilloscope recordings
• Progress display: status display during the current optimisation run, with the option to create a log
Status:
Green Optimisation complete
Amber Optimisation running
Red Optimisation faulty</td></tr>
<tr><td>Inputs (actual values)</td><td>• Current
• Voltage
• Frequency
• Speed
• Temperature
• Position
• Brake control
These values are also monitored for threshold values.</td></tr>
<tr><td>Inputs (function)</td><td>The auto-tuning function records all control parameters that are relevant for the drive. Extracts from these parameters are listed below.
• Current controller Kp, Tn
• Speed controller Kp, Tn
• Position controller kv
• Pre-control values
• Filter selection
Result
The result of the auto-tuning function is a new, optimised data record. It can be displayed by means of a traffic light or in detail via the optimised data record.
It should be possible to select output of the entire data record or the changed parameters.</td></tr>
<tr><td>Manual entry (parameter)</td><td>• Selection of options to specify the procedure for the auto-tuning function
• Assignment of access rights</td></tr>
</table>

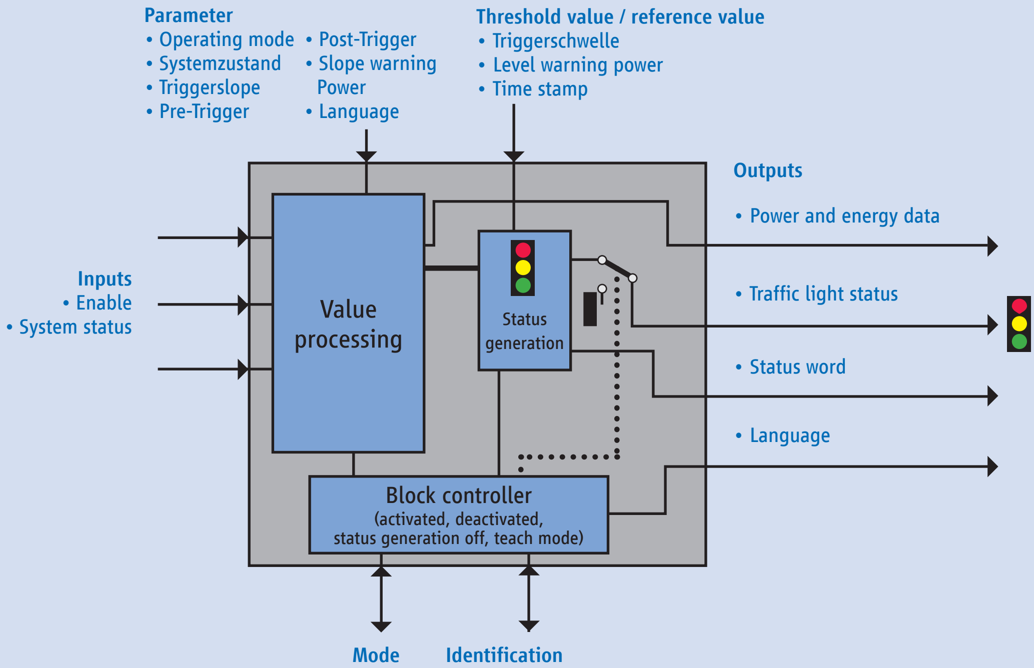
Logbook/fault memory (input)	The source for the messages to be output; a list that is not defined in more detail with time stamps, data records, error codes and further data (threshold values), if applicable, that are currently implemented on a manufacturer-specific basis, sometimes only partially or even not at all..

Mode (parameter)	➡ See Chapter 6.2.1
	Of the modes specified there, the following are relevant for the function:
	• "Activated": starts the function and delivers real data
	• "Deactivated": no execution or ends the function
	  • "Manual": specification of the procedure for the auto-tuning function (manual entries parameter)
	  • Selection of the optimisation processes
	  • (e.g. speed controller, position controller, current controller)
	  • Selection of multi-axis system; for multi-axis systems, axis groups also need to be optimised (e.g. for path-related production)
	• "Simulation": real measured values are ignored; the actual value planning is performed based on simulation values specified externally or generated elsewhere
	• "Reset": restart of the function with standard parameter sets (this can also be realised without actually deleting the entries, e.g. by calling saved predecessor parameter sets)
	• "Learning mode" (input values are fed in to form reference and/or threshold values in a learning process; traffic light status is "no condition statement")
	• Status display of the current status of optimisation, create log
	Status:
	Green Optimisation completed
	Amber Optimisation running
	Red Optimisation faulty

Fault codes/messages (output)	List with the required entries. Every entry consists of a time stamp, an ID for the type (e.g. fault, warning, note), plain text, and an optional fault code that can be specified based on a standard or a particular manufacturer.
Output (data record, parameter)	Delivers the results as lists of parameters and data records. It should also be possible to provide comparison information.
Status word (output)	➡ See Chapter 6.2.1
Traffic light status (output)	➡ See Chapter 6.2.1
Error status (output)	➡ See Chapter 6.2.1
Time stamp (output)	➡ See Chapter 6.2.1
Language (output)	➡ See Chapter 6.2.1

6.2.5 Energy management

Fig. 28: Presentation of the manufacturer-independent function "energy management"



Source: ZVEI

Alongside throughputs and reliable operation, the consumption of electrical power (e.g. 15-minute peak values in power consumption) is playing an increasingly important role in the process and manufacturing industry.

In addition, current power values can be used to draw conclusions regarding the manufacturing process in the sense of quality control: for work processes that constantly recur when manufacturing the same parts, a sudden or gradual deviation in power or energy consumption indicates discrepancies (e.g. an incorrect work piece, different material quality, wear of cutting tools).

Every converter or inverter assigned to a PDS has current sensors for current/speed control on the machine side. The intermediate circuit current is also monitored, as this is significant for the generation of the machine voltage to be supplied in the scope of (space vector) modulation. This sensor data and the switching state of the inverter on the machine side can be used to determine the active power delivered to the machine, and thus indirectly the power extracted from the DC intermediate circuit, as well.

If the converter has a diode rectifier, the fundamental frequency reactive power and the distortion power absorbed on the mains side can (along with the mains-side AC active power) be estimated from the DC power without the need for additional current sensors on the mains side. When using an active-infeed converter (rectifier/inverter with active semiconductor switches, usually IGBT), the mains-side current sensors and the mains voltage measurement can also be used to determine the absorbed reactive and active power.

In this way, the electrical power quantities taken from the mains are available in the inverter and can be transferred to the higher-level control together with the reference and actual values for the torque and speed. This data can then be used to determine or estimate losses in the inverter and the motor, and thus the efficiency and heat output.

Example of how the function can be used in the production industry: The higher-level control can use the combination of values from several drives installed on a tooling machine, for example, as well as the data from other systems (heaters, cleaning systems, etc.) to create a load profile for the entire tooling machine in relation to the processing steps performed. This enables the following functions at the tooling machine control level:

1. Recording of the entire active and reactive energy absorbed by the tooling machine in relation to the processing steps performed (metering functionality)
2. Assessment and (cost) optimisation of the absorbed energy/peak power through variation of the reference values for the individual drives within the thresholds determined by the manufacturing process – for example, through Monte Carlo analysis (finding the optimum based on randomly distributed values) or other "intelligent" optimisation procedures such as particle swarms or similar methods
3. Monitoring the driven load with regard to unexpectedly high levels of torque/force or mechanical performance across the movement trajectory, which indicates faults
4. Coordination of the power consumption between tooling machines, which reduces the peak load within a machining unit consisting of several tooling machines
5. Offline optimisation: minimisation of the peak power consumed (see item 2), which reduces the connection cross-sections through prior simulation and optimisation of the entire system in a digital twin – the "simulation" mode is used for this purpose

The function can be parametrised and used as follows:

The function generally determines mean values over a certain time period (to be specified in each case) on the mains side and on the motor side; the duration for forming the mean values can be set differently for each side. For AC values, the RMS (root mean square) is determined, while the arithmetical mean value is determined for DC values.

In addition to the standard interfaces (see Chapter 6.2.1), the energy management function uses the following interfaces:

Enable (input)	➡ See Chapter 6.2.1
System state (input)	➡ See Chapter 6.2.1
Mode (parameter)	➡ See Chapter 6.2.1 • Of the values specified there, the following are relevant for the function: • "Activated": function is executed • "Deactivated": function is not executed • "Simulation": real measured values are ignored; the actual value planning is performed based on simulation values specified externally or generated elsewhere • "Reset": all internal data is reset, including interim values and learned reference and threshold values
Operating mode (parameter)	• The operating mode can be used to influence how energy and performance data is made available: • "Deactivated": no determination/recording of performance and energy data • "Streaming": performance and energy data is delivered continuously without any separate request (according to the available bandwidth for data transmission) • "Polling": performance and energy data are transferred on request • "Start": start data collection or recording • "Stop": stop data collection or recording • "Reset" all energy quantity values are set to zero • "Trigger": wait for predefined trigger event and then begin recording in accordance with the parameters set for the trigger function; the data is automatically transferred once the post-trigger expires
Duration of averaging on mains side (parameter)	• Duration (in seconds) of average value formation on the mains side; default: 10 mains periods

Duration of averaging on machine side (parameter)	• Duration (in seconds) of average value formation on the machine side; default: 10 mains periods
Trigger threshold (threshold value)	• "Threshold value in W": value of the active power that has to be undercut or exceeded to initiate the trigger
Trigger slope (parameter)	• "Rising" or "falling": trigger is initiated if the power is above or below the trigger threshold
Pre-trigger (parameter)	• "Pre-trigger": time (in seconds) for which recording should occur before the trigger event
Post-trigger (parameter)	• Post-trigger": time (in seconds) for which recording should occur after a trigger event
Power and energy values (outputs)	• Power and energy values output the determined power and energy values according to the selected operating mode (each labelled with a time stamp): • "P_{mains}": active power absorbed from the mains in W • "Q_{mains}": reactive power absorbed from the mains in W • "S_{mains}": apparent power absorbed from the mains in VA • "$P_{mains,peak}$": peak active power absorbed in W • "Power factor": ratio of absorbed active to apparent power • "P_{motor}": active power delivered by the motor in W • "E_{grid}": real energy absorbed from the mains in Ws • "Q_{mains}": reactive power absorbed from the mains in Var • "S_{mains}": apparent power absorbed from the mains in VA • "$W_{W,mains}$": real energy absorbed from the mains in Ws • "$W_{Q,mains}$": reactive power absorbed from the mains in VAsr • "$W_{S,mains}$": apparent energy absorbed from the mains in VAs • "U_{LN}": line-to-neutral voltage (for single-phase devices only) • "U_{L1L2}": line-to-line voltage L1-L2 in V • "U_{L2L3}": line-to-line voltage L2-L3 in V • "U_{L3L1}": line-to-line voltage L3-L1 in V • "k_U": unbalance factor of the mains voltage (line-to-line voltage) as a ratio of the negative sequence/positive sequence of the basic oscillation • "THDu": total harmonic distortion of the mains-side voltage (mean value of the three line-to-line voltages of three-phase devices or line-to-neutral voltage of single-phase devices) up to 2 kHz • "THDi": total harmonic distortion of the current absorbed from the grid up to 2 kHz for three-phase devices: mean value of all three phase currents)
Level warning: power (threshold value)	This warning makes it possible to specify thresholds for power values; a warning will be triggered whenever the value falls below or exceeds them. The function works independently from the other transfer of power values: • "$P_{warning}$": threshold value for triggering the warning, in W
Slope warning: power (parameter)	• "Rising" or "falling": warning is triggered if the warning value is undercut or exceeded
Status word (output)	➡ See Chapter 6.2.1
Traffic light status (output)	➡ See Chapter 6.2.1
Time stamp (reference value)	➡ See Chapter 6.2.1
Time stamp (output)	➡ See Chapter 6.2.1
Language (parameter)	➡ See Chapter 6.2.1

6.2.6 Maintenance log function

Availability and reliability are crucial properties for the cost-effective operation of production plants with electrical drive systems throughout their life cycle (TCO). The "maintenance log" function promotes economical operation and supports condition monitoring and predictive maintenance concepts. It provides meaningful and complete information regarding the maintenance status, as well as on the time and money spent on planned and unplanned maintenance activities. The prepared data creates transparency for optimisations and increased efficiency in operational plants. At the same time, it forms the basis for optimisations in the scope of product observations and future investment decisions.

During the usage phase of a power drive system (PDS), the operator receives various data on maintenance, support and service that is of interest to all participants in the value creation process. This data is recorded in a fully (or partially) automatic manner and stored chronologically in the asset administration shells of the Industrie 4.0 components at hand. The "maintenance log" function is typically implemented in the function layer of a machine. The electrical drives (assets) deliver data that is combined in the information layer with other data from the information layer and then transported to the communication layer. Executing the function generates chronological information regarding maintenance, support and service activities, which users can process or filter in different ways.

Attributes and application of the function:

The "maintenance log" function is created based on a given instance and updated throughout the entire period of use of the instance in the form of an intelligent logbook. This ensures continuous and seamless documentation of the information throughout the product life cycle.

Manufacturer and operator data, relevant inputs and outputs, and prepared values and status information are processed with time and usage data. They then provide a basis for deriving maintenance recommendations or instructions.

The maintenance activities are recorded automatically or confirmed manually and documented in a log with time stamps. The data records are supplemented through manual maintenance of the entries – for example, in the event of an unplanned part replacement.

The log can be displayed in full or filtered, and can be output via various interfaces for further processing.

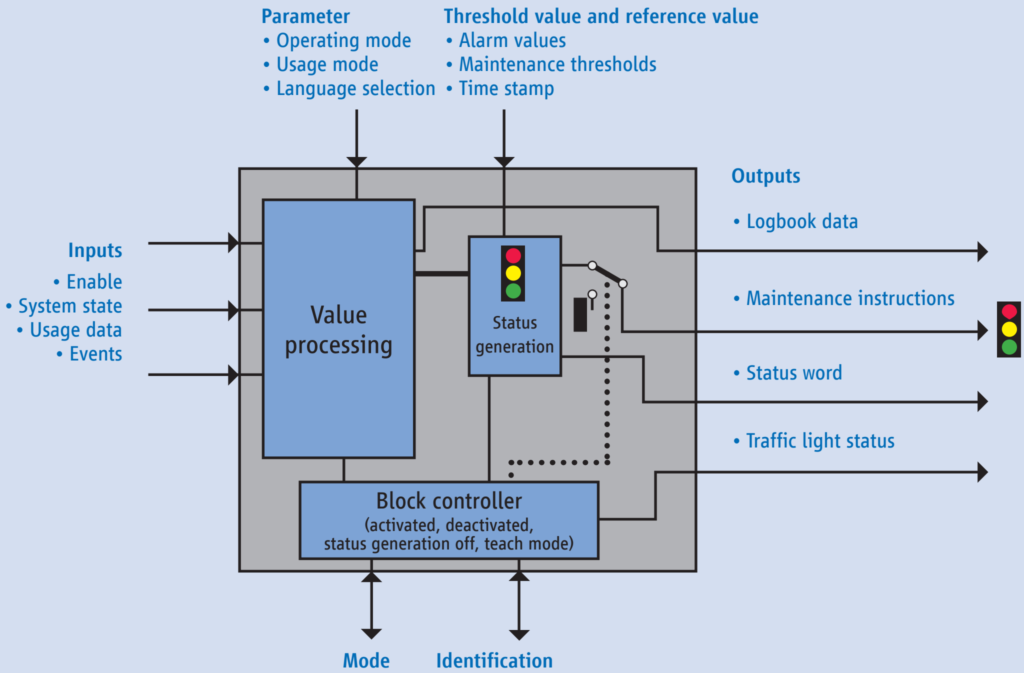
All information is archived in a database and retained for further use. The function includes an authorisation concept with configurable parameters for the individual instances. Authorisations for data access (read-only), data processing and manual data maintenance can be specifically authorised here.

User benefits of the function:

The interaction between the asset layer, information layer and functional layer (see RAMI 4.0 for term definitions) results in decisive synergies and increases in efficiency.
The automated combination of cross-divisional and instance-based information enables considerable synergies and quality increases when preparing and updating maintenance activities and workloads.

- Data provision for cost assessments of plants and production units
- Automated allocation of maintenance activities and the corresponding costs
- Up-to-date and complete contact information is available in all instances and layers
- Responsibilities are transparent and the corresponding users can be contacted automatically
- Time savings when maintaining and identifying responsibilities

Fig. 29: Presentation of the manufacturer-independent function “maintenance log”



Source: ZVEI

In addition to the standard interfaces (see Chapter 6.2.1), the maintenance log function mainly uses the following interfaces:

Enable (input)	➡ See Chapter 6.2.1
System state (input)	➡ See Chapter 6.2.1

Operating mode (parameter)	Individual operation of the “maintenance log” function is enabled through specific user settings.
	• Signal selection defines the relevant values and quantities for the database
	• Output format
	  • File/data matrix with header and data, or;
	  • Cyclically automated output, or;
	  • Results-based, or;
	  • Manual (acyclical)
	• Maintenance parameters
	  • Manufacturer specifications – for instance, cyclically after twelve months
	  • Operator specifications – for instance, after 1,000 operating hours or cycles
	  • Laws/guidelines – for instance, 20 years of service life
	  • Recommendations – for instance, when measured-value-based thresholds are reached (anomalies)
	• Instance-based authorisations
	  • for viewing and processing data for manual data maintenance

Threshold values (parameter)	The function can be parametrised on various threshold values of the instance
	• Alarm values
	  • Time-based interval reached
	  • Use-based interval reached
	  • Measured-value-based interval reached
	• Communication monitoring
	  • Parameters channel
	  • Input signals/communication
	  • Output signals/communication

Usage data (input)	The input data for operation of the instance is used and processed in various functional algorithms. • Manufacturer's specification The function's algorithm processes type-based parameters of the component manufacturer with current operating and usage data. • Operator's specification The function's algorithm processes instance-based specifications of the operator with current operating and usage data.

Events (input)	The "maintenance log" function automatically records events and uses them to create a logbook throughout the life cycle. Maintenance activities are added in a partially automated or manual fashion.
	• Logbook (automated)
	  • Entries regarding errors
	  • Entries regarding warnings and alarms
	  • Entries regarding events, messages, information and notes
	• Maintenance activities (manual/partially automated)
	  • Entries regarding controls/checks
	  • Entries regarding cleaning/inspection work
	  • Entries regarding troubleshooting
	  • Entries regarding the exchange of components
	  • Entries regarding changes in parameter settings (software)
	  • Entries regarding changes made to the configuration/settings (hardware)

Time stamp – reference time (input)	➡ See Chapter 6.2.1 A standardised, end-to-end time reference is crucial for all functions that exchange data among one another.
Language selection (parameter)	Language switching can be added to the function. This allows different users to read in or process data records in their own language. The language display can be configured based on site or user.
Mode (parameter)	➡ See Chapter 6.2.1 The following modes are relevant for the "maintenance log" function: • Activated • Deactivated • Manual • Simulation • Status generation ON/OFF • Learning mode

Current time stamp (output)	• Automatically assigned time stamp ➡ See Chapter 6.2.1 For entries that are generated by the system, the time stamp is automatically generated from the reference time and assigned. The time stamp always contains the year, month, day and hour. In specific cases, the resolution can be supplemented with minutes and seconds. • Manually assigned time stamp Maintaining data records manually by assigning a date and time makes it possible to complete the entries with events that are not automatically recorded, such as an unplanned part replacement that takes place when a device is switched off.
Status word (output)	• Status display ➡ See Chapter 6.2.1 • Valid • Invalid • Faulty • Ready • Parameter transfer active • Data transfer active • Error display • Parameter error • Runtime error • Return error
Traffic light status (output)	➡ See Chapter 6.2.1 The traffic light status provides information on the quality and completeness of the log data. It signals the result of the query or the function call in three colours: GREEN, AMBER and RED.
Operating data (output)	The database of functions is output in various structures and formats. • File Standardised, cross-manufacturer output of the file/data matrix generated specifically for the function in the specified output format, including a header and data record • Data stream Output of the data stream/data packages generated specifically for the function – for example, as input for other functions or simulation models • Generic data Output/call up of data generated specifically for the function in a handshake procedure

7. Application

7.1 Demonstrator

7.1.1 Objective

To check how drives from different manufacturers interact and to accelerate and test the further development of data and information models, a demonstrator is being developed. The objectives of the demonstrator are as follows:

- Testing the data structures and models developed
- Demonstrating the cooperability of electrical drives as Industrie 4.0 components
- Demonstrating additional functions that Industrie 4.0 drives make possible (starting with the function of an oscilloscope)

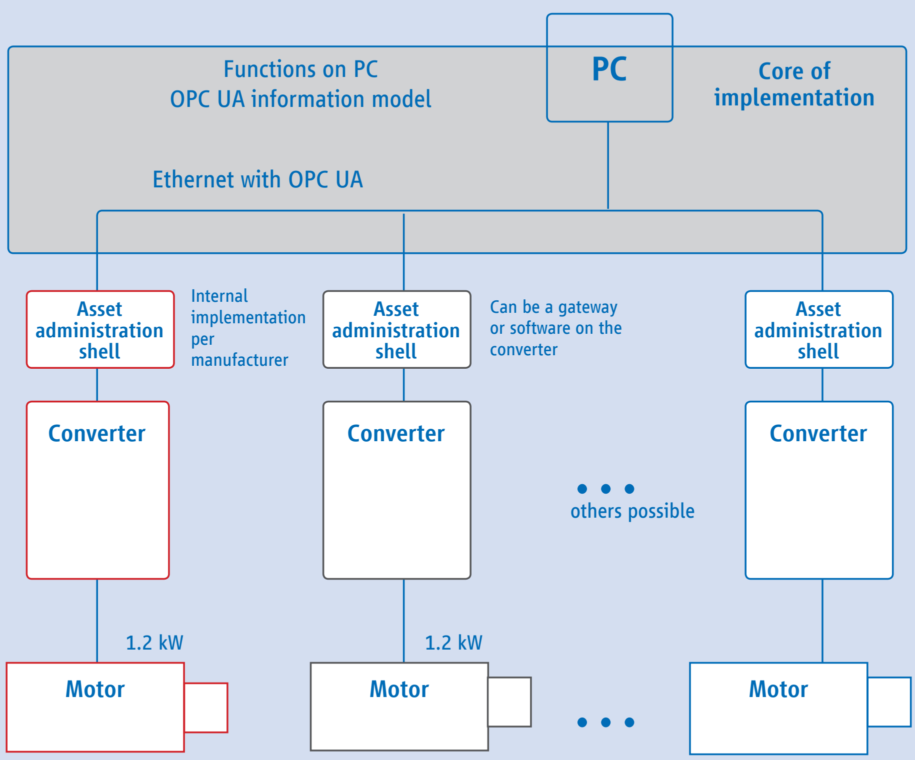
The demonstrator is also to be exhibited at trade fairs (SPS, Hannover Messe) to help attendees grasp how Industrie 4.0 drives work.

At a later stage, the demonstrator can be expanded into a test system that can then be used to test other Industrie 4.0 functionalities.

7.1.2 Description

The demonstrator contains drive systems/servo drives from ten manufacturers, each with a rated power of approx. 1 kW. The drives are connected via Ethernet to an OPC/UA client that displays the information transferred from the drives on a monitor. The Ethernet connections can be disconnected and connected at any time; the communication starts on its own each time.

Fig. 30: Overview of demonstrator structure



Source: ZVEI

Electrical structure:

The drives are wired and mutually connected to the 400 V grid (16 A CEE plug) via two EMERGENCY STOP switches, protective devices (LS) and an isolating transformer or EMC filter to avoid RCD triggers. Drives with a single-phase input (230 V) are connected between an external conductor (alternating) and a neutral conductor. A main switch is provided for the drives.

The power supply for the IT is switched on in advance via a separate switch. The power supply for the drives can only be switched on once the power supply for the IT has been activated.

For safety reasons, the drives only start operating after a potential-free input has been activated externally by means of an external release switch. Following the external release, the drives move independently based on a specified and non-adjustable speed-time profile.

Mechanical structure:
The drives are installed on a test vehicle (lightweight construction profiles); each is equipped with a flywheel and a manually operated brake to be able to simulate load situations. Every drive from the involved manufacturers is installed on a 250 mm x 1,000 mm plate. Adapters are used to install all the drives so that their axes are at the same height and the flywheels are aligned in an axial direction (radially and axially aligned flywheels).

To avoid accidents (involving clothing or body parts being drawn in), the rotating parts of the drives are located under a transparent cover (e.g. polycarbonate).

Fig. 31: Sketch of the demonstrator structure

Source: ZVEI

7.2. OPC UA model

7.2.1 Objective

From a software perspective, the core of the demonstrator is asset administration shells, which build on the existing hardware and software of the drive systems provided and typically provide Industrie 4.0 functions such as an electronic rating plate and the Industrie 4.0 oscilloscope previously described in more detail. The corresponding data structures and software interfaces are agreed among the companies participating in the working group Industrie 4.0 for Electrical Drive Systems at various development stages. To ensure a largely platform-independent implementation of the asset administration shells, the working group selected the OPC UA standard. This supports the exchange of data from various devices, as well as their preparation and aggregation for higher-level layers of production control (MES, ERP, etc.) regardless of the available communication protocols (TCP, HTTP, SOAP). Both event-driven and procedural programming concepts are used here.

7.2.2 Description of the hardware/software structure

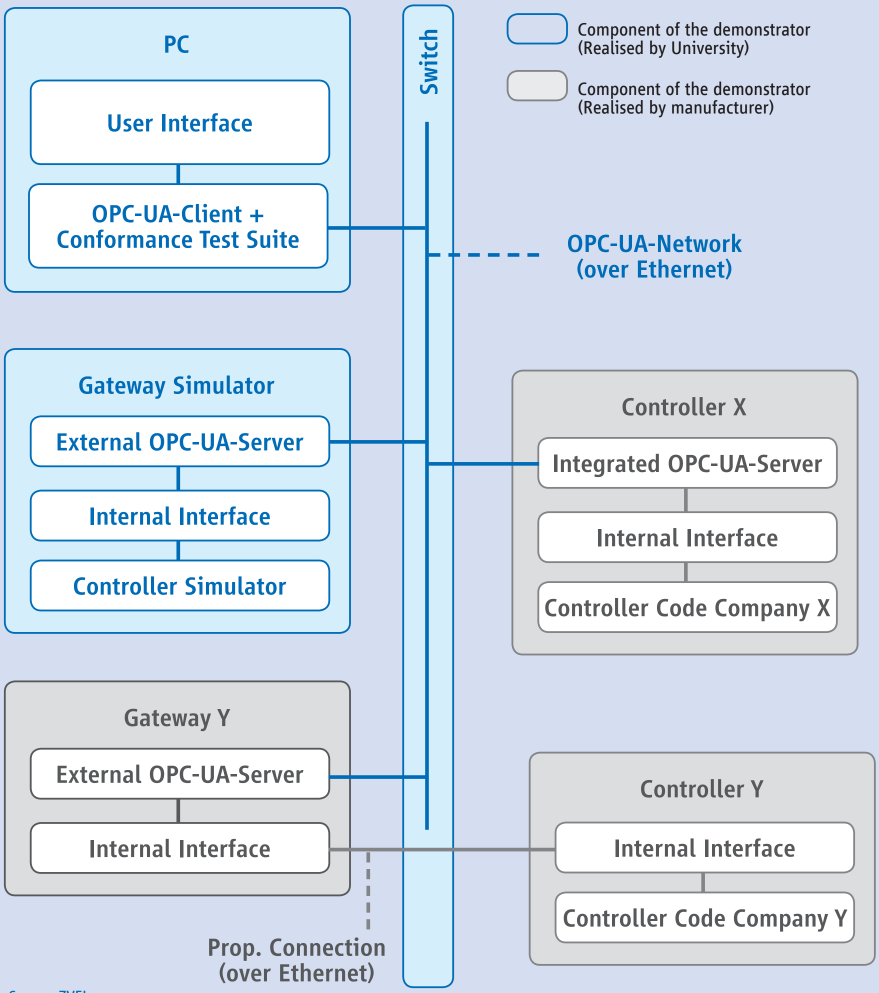
All the drives installed on the demonstrator are connected to a PC via Ethernet and a switch. On this PC, the data transmitted by the drives can be displayed and the querying of operating data (e.g. oscilloscope) initiated. An OPC UA client is implemented on the PC, allowing the exchange of data with the OPC UA servers (asset administration shells) on the various drive systems.

There are two variants for implementing the predefined OPC UA information model on the drives in the form of an OPC UA server: depending on the available memory and computing capacity, it can either be installed directly on the existing controller of the servo controller or on an additional computer platform (gateway). If manufacturers provide for an external computer platform, the communication between the external platform and the servo controller takes place via a manufacturer-specific protocol. The following specifications apply to the implementation of an OPC UA server for an Industrie 4.0 drive:

1. Specification of the supporting information model in various development stages
2. A reference implementation of an OPC UA server (on a gateway simulator) and an OPC UA client (on the demonstrator PC)
3. A conformity test suite for validation of the various manufacturer-specific implementations of OPC UA servers

Here, only basic functionalities of OPC UA are used in order to ensure compatibility with the maximum number of freely available or commercial OPC UA server implementations (e.g. we are initially avoiding "Historical Data Access" services).

Fig. 32: IT architecture of the OPC UA information model for implementation of asset administration shells



Source: ZVEI

The Ethernet interfaces provided are used solely to transfer data from the OPC UA information model – not real-time data for specifying the actual values of the drives (torque, speed, etc.) that need to meet real-time requirements in the sense of the drive's task.

The reference implementation provided for an OPC UA server simulates a converter or its corresponding gateway. It is made available as a framework or product line in a way that allows the corresponding C code to be used as a basis for implementing the asset administration shells on the converter of the various manufacturers. The reference implementation consists of the actual server (with call-back interfaces for the proprietary code of the drive in question) and a simple simulation of a motor with a converter and the corresponding control. The implementation of an OPC UA client provided on a PC aggregates the data from the various motor controls and displays it, partially as text and partially in graphical form. It also includes a test suite for checking the conformity of the various OPC UA server implementations.

Various OPC UA stacks are currently available, both as free software (e.g. the "open62541" OPC UA implementation from the Fraunhofer Institute of Optronics, System Technologies, and Image Exploitation (IOSB) or the C implementation from the OPC Foundation) and commercial versions (e.g. the products from Ascolab and Softing). We plan to use various stacks in parallel in the demonstrator. In this way, manufacturers can gain experience with using different stacks and test their interaction.

8. Outlook

Overall, this white paper merely covers the initial work that has begun on Industrie 4.0 drives. The first part of this outlook therefore looks at the consistent continuation of the work that has been started on electrical drive technology:

Functions:
A host of useful functions were identified in a workshop with manufacturers and operators of machines and plants. This white paper describes the first five, and further functions can be developed based on the schema presented.

Specific products:
The demonstrator is giving the member companies an initial impression of how intelligent field devices can cooperate and work with one another in the scope of Industrie 4.0. Some companies are also using the demonstrator to validate their own product development. Initial product releases from the individual manufacturers can therefore be expected as the demonstrator is developed further.

OPC UA information model for drives:
The information is also being developed further in collaboration with VDMA and the OPC Foundation. Results from the working group that are important for the information model are being incorporated into the work. All in all, the collaboration of the working groups from VDMA, the OPC Foundation and ZVEI will lead to the creation of shared OPC UA information models. Various working groups and their members have committed to this effort.
Standardised semantics and data structures:

Properties for Industrie-4.0-compliant electrical drive technology are standardised using eCl@ss. Further functions, as well as feedback from practical implementations on the demonstrator and from the working group for the OPC UA information model, will result in further requirements for standardised properties. The working group would like to continue supporting the eCl@ss specialist group in this regard to ensure that the set of properties for electrical drives is continuously expanded.

New tasks are also planned in addition to continuation of the previous work.

Life cycle phase – development:
The working group looks at data throughout the product life cycle: from the manufacturer to the machine builder to the machine operator, and from the initial idea to development, production and logistics right up to service, including the end of service life.

Until now, the functions concentrated on the life cycle phase of production, including commissioning and service. Now, in a further step, the focus is to be more on functions in the development and application phase, with the aim of increasing the potential benefits for users in this context, as well.

Standardised data:
Standardised properties are also required for other products and components from the electrical drive train. Following prioritisation, other components will follow, such as converters, cables and gearboxes.

It is also necessary to integrate the standardised properties into international standards beyond eCl@ss (for instance, IEC).

There is also a need for definition when it comes to the groups of people that should be permitted to access properties and functions. Corresponding authorisation mechanisms are currently being investigated in the ZVEI project group on "the asset administration shell in detail" and may be included in the working group Industrie 4.0 for Electrical Drive Systems in the future.

Continuation of the demonstrator:

The project described in this white paper on implementing an Industrie 4.0 drive technology demonstrator has a specified functional scope that is meant to facilitate a presentation of results in one year. Many of the members' ideas and requirements had to be deferred to allow this first step. Discussions are thus already under way regarding continuation, particularly in the area of software functions. This means that on completion of the first project, further research can immediately be performed and practically implemented on the demonstrator.

Conclusion:

In summary, a great deal of work is needed to continue developing a product group as involved as electrical drive technology towards Industrie 4.0 implementation. Along with active associations and universities that provide their infrastructure, specialist expertise and contacts, this will require dedicated employees in the working groups from the member companies. In the field of electric drive technology, the right mix has been achieved to enable everyone involved to realise a common vision in the spirit of constructive cooperation. Let yourself be surprised by the results this working group achieves in the future.

Die Elektroindustrie

ZVEI - German Electrical and
Electronic Manufacturers' Association
Lyoner Strasse 9
60528 Frankfurt am Main, Germany
Phone: +49 69 6302-0
Fax: +49 69 6302-317
E-mail: zvei@zvei.org
www.zvei.org