Survey on Real-Time Communication Via Ethernet in Industrial Automation Environments

Peter Danielis, Jan Skodzik, Vlado Altmann,
Eike Bjoern Schweissguth, Frank Golatowski, and Dirk Timmermann
University of Rostock
Institute of Applied Microelectronics and Computer Engineering
18051 Rostock, Germany
Tel./Fax: +49 (381) 498-7277 / -1187251
Email: {peter.danielis;jan.skodzik}@uni-rostock.de

Joerg Schacht
Max Planck Institute of Plasma Physics
17489 Greifswald, Germany
Tel.: +49 (3834) 882761
Email: Joerg.Schacht@ipp.mpg.de

Abstract—**For companies in the automation industry, the development of real-time Ethernet to connect devices is of high economic interest to replace conventional fieldbus systems. Therefore, many approaches for adapting Ethernet to real-time requirements come from industrial applications. This is a challenging task as the original Ethernet standard IEEE 802.3 was not designed for real-time data transmission. Likewise, protocols basing on Ethernet like TCP, UDP, and IP do typically not consider real-time requirements. Hence, adaptations on several OSI layers become necessary to make the industrial system meet hard real-time requirements. For this purpose, a multitude of real-time capable Industrial Ethernet systems has been developed, which solve the problems of standard Ethernet- and TCP/IP- or UDP/IP-based communication in a variety of ways. This paper gives a summary of different Industrial Ethernet protocols for the real-time data transmission via Ethernet in automation environments. Advantages and disadvantages of these protocols are analyzed with regard to their sustainability in terms of their real-time capability, reliability, scalability, self-configuration of the network, and hardware requirements. Against the background of connected devices tremendously growing in number and computational power in the prospective "Industrial Internet", consequences for future developments are drawn.**

I. Introduction

After the industry has already undergone three revolutions in the form of mechanization, electrification, and informatization, as fourth industrial revolution, the Internet of Things and Services is predicted to find its way into the factory. For this development, e.g., in Germany the term "Industry 4.0" has been coined [1]. In the most general sense, globally a networked and real-time (RT) capable industrial production is aspired. The US-American company General Electric (GE) recently initiated a comprehensive research initiative called "Industrial Internet" [2]. Thereby, not only the industrial production but also the whole industrial infrastructure shall be intelligently networked. GE forecasts for the future that there will be more intelligent devices, which have to be connected to interact with each other dynamically.

As part of these efforts, companies aim at connecting their devices, storage systems, and supplies as cyber-physical systems (CPS) prospectively. In that way, intelligent devices, storage systems, and supplies shall be created in the industrial production, which exchange data in a self-organizing way, trigger actions, and control each other. CPS are systems with embedded software as part of, e.g., manufacturing facilities but can also comprise buildings and devices, which collect physical data by means of sensors and influence physical processes with actors. CPS are networked among each other with local digital communication systems but also with global networks. By connecting embedded systems with global networks, on the one hand numerous applications for all parts of daily life and novel business opportunities emerge. On the other hand, this poses the challenge of bringing together the features of embedded systems like RT requirements with the openness of the Internet [3].

Finally, the vision is the so-called Smart Factory with a novel production logic: The products are intelligent and can be identified clearly, constantly located, and are aware of their current state. These embedded production systems shall be interconnected with economic processes vertically and combined to a distributed RT capable network horizontally. To reach this vision, the production systems must be flexible and adaptable. Therefore, automation structures are necessary to manage the high complexity arising from the increasing number of devices. Furthermore, the control of a network consisting of thousands of devices is a technical challenge requiring tools and technologies to be able to meet this challenge. As soon as the number of devices increases from 100 to 1,000 or even 10,000, the device networking technology must be ready for this magnitude. Consequently, e.g., the "Industry 4.0" Working Group urges to meet the requirements of guaranteed latencies, i.e., RT capability and high resilience for the desired massive interconnection to ensure the frictionless functionality of the respective applications [1].

To conclude, devices have to be interconnected in industrial facilities to communicate with each other and prospectively, their number will strongly increase in the described Internet of Things and Services. Therefore, this paper first elaborates on the current development from fieldbuses to Ethernet as *device networking technology for devices in the Internet of Things of Services* in Section II. In Section III, established RT capable Ethernet systems are presented and compared. In the following Section IV, against the background of connected devices

2014 IEEE Emerging Technology and Factory Automation (ETFA)
978-1-4799-4845-1/14/$31.00 ©2014 IEEE

tremendously growing in number and computational power, consequences for future developments are drawn and current developments are sketched. Finally, the paper concludes in Section V.

II. Current Development: From Fieldbuses to Ethernet

For the RT capable device interconnection in industrial environments, originally various fieldbus solutions have been used. However, fieldbuses are subject to severe restrictions concerning the number of devices that can be networked thereby limiting the scalability and resilience. Furthermore, interoperability between the various solutions is not provided. Therefore, the networking by means of the wide-spread Ethernet technology currently prevails against fieldbuses [4]. The application of Ethernet to connect field devices offers substantial advantages compared to fieldbuses as Ethernet enables the consistent integration on all levels of a company. Thereby, Ethernet solutions allow for a total vertical and horizontal integration of an automation system from the field level up to company level [5], [6], which is of decisive importance to realize the vision of an Industrial Internet. There are many challenges, e.g., addressing the data representation, which have to be faced, to especially realize a vertical integration of the system as field devices can now communicate with PCs in an office without any gateway between them. This is one future implementation detail, which has to be considered, to properly represent and process data from the lowest to the highest level in an automation scenario [7].

Via Ethernet connections between field devices, time-critical process data, which fulfills control tasks, is to be transmitted meanwhile non-time-critical IT data is sent to different IT services in the company [8]. In the case of this type of communication, status information can be read out or field devices can be controlled remotely. For the transmission of IT data, standard protocols like TCP/IP or UDP/IP can be used whereby time-critical process data may require the application of special protocols. The entire communication via Ethernet takes place on a common hardware base, which has been widely standardized by IEEE 802.3 and offers different plug connections and communication media, which can be adjusted to the specific purpose. Another advantage of Ethernet is the high performance compared to conventional fieldbuses, which are evolving into weak spots between powerful computer systems.

However, standard Ethernet as used today in many areas comprises various mechanisms, which prevent a deterministic data transmission and therefore the application in environments with RT requirements. The first problem of standard Ethernet is its application of the CSMA/CD access method. Due to possible collisions, the data transmission may be interrupted or can only take place at a later undetermined point in time. By using full-duplex switched Ethernet, the problem can be solved but new problems arise. In switches, data is buffered, i.e., frames are put into a queue on the switches, which leads to additional delay or even packet loss under specific traffic load conditions. If, e.g., several network devices send much data to the same destination via one switch buffer overflow and packet loss must be expected, which contradicts a deterministic data transmission. On the IP layer above Ethernet, the problem of non-static routes exists so that the way and transit time of a packet are not precisely predictable. Likewise, the choice of the transport protocol for meeting high RT requirements is challenging to be able to transport larger amounts of data than solely process data in future applications while ensuring deterministic predictable transmission times. This could become more important in the future, which, e.g., the current works of the IEEE Time-Sensitive Networking (TSN) Task Group show aiming at allowing for time-synchronized low latency streaming services through 802 networks [9]. As automotive engineers have shown interest in these works for, e.g., distributing streams in vehicular communication systems [10], the works of the IEEE TSN task group could become prevalent in industrial automation as well. While UDP can be basically considered as RT capable due to sending single independent packets, it does not provide the labeling of related data or data order, which is indispensable for the reliable transmission of larger amounts of data. Contrary, TCP has been designed for the transmission of large amounts of data and therefore labels related data and their order. However, due to the variable transmission speed, which is caused by mechanisms for overload control and error correction, TCP is fundamentally not RT capable.

A multitude of RT capable Industrial Ethernet (IE) systems has already been developed, which remedy the deficiencies of standard Ethernet and TCP/IP or UDP/IP-based communication in a variety of ways [11]. “Industrial” refers to the compatibility of the solutions with rough industrial environments [12]. Therefore, certifications are carried out to prove the compliance with the regulations. As standard Ethernet uses the CSMA/CD mechanism to control the media access, neither the arrival of an Ethernet frame nor its delivery time can be guaranteed. However, to be used in machine halls, IE solutions must provide guarantees in this regard. Control systems have to be implemented as RT systems so that data is transmitted within fixed time limits. Neither switched nor full-duplex Ethernet can ensure fixed time limits and moreover, it cannot restrict the jitter of Ethernet frames. For multimedia applications, this problem has been solved by prioritizing the data flow and by using virtual LAN standards [13], [14]. Thereby, the jitter is reduced and delivery times of 10 ms in case of highly prioritized data traffic is reached. However, these are only statistical assessments. For ensuring lower deterministic delivery times, either the Ethernet protocol itself has to be modified and/or the network components/devices have to take measures to manage the communication. RT for Ethernet can, e.g., be achieved by applying a token-passing procedure as done in [15]. However, the monitoring of the token and the participants is required. If a participant fails, the previous station usually sends the message in the opposite direction as the communication otherwise fails completely. Moreover, a message has to be passed to all participants on the token ring, which can lead to high latency in case of a very high number of participants. Therefore, in most IE systems, a time division multiplex method is applied whereby each device is allowed to communicate in a time slot. This requires a common time base for all devices, i.e., a synchronization among all devices is required.

This paper analyzes the advantages and disadvantages of RT capable IE systems with regard to their sustainability in terms of

- RT capability: Which performance and thus what kind of RT can be achieved in the best case?
- Reliability: Does the network contain a single point of failure (SPoF)?
- Scalability: How many devices can exchange data in RT? Prospectively, there will probably be several thousands of devices to be connected [1], [2].
- Self-configuration: Does the system show self-configuration features like dynamical adaptation to changes of the network topology or does it have to be statically/manually configured?
- Hardware requirements: Is special Ethernet hardware needed?

III. RT CAPABLE INDUSTRIAL ETHERNET SYSTEMS

A system is called RT capable if the response of a system to a request does not exceed a given time limit [16], i.e., such RT systems have the distinction of keeping time conditions given by applications. That is, the correctness of a RT system not only depends on the correct computation result but also on the time of the result computation [17]. Depending on to what extent the time conditions are mandatory the type of RT is referred to as soft or hard. Soft RT means that the application allows for keeping the time conditions substantially, i.e., time limits may be exceeded slightly without damage incurring. For the definition of hard RT conditions, the following requirement called punctuality is given in Equation 1.

$$A \equiv r + \Delta e \leq d \tag{1}$$

r denotes the point in time, at which a task starts. The task execution takes a time span of Δe and the task must be attended at the point in time d. Hard RT systems are characterized by definitely keeping the time condition A under the boundary condition B so that the condition given by Equation 2 applies:

$$P(A|B) = 1 \tag{2}$$

Depending on the specific task, B indicates that within the time span Δe, elapsing from r to d, neither technical faults occur nor more important tasks have to be attended.

For the precise description of the RT capabilities of IE systems, three classes can be defined depending on the cycle times of the IE systems [18], [19]:

- Class 1: soft RT: scalable cycle time, approx. 100 ms.
- Class 2: hard RT: cycle time 1 to 10 ms.
- Class 3: isochronous RT: cycle time 250 μs to 1 ms; jitter less than 1 μs.

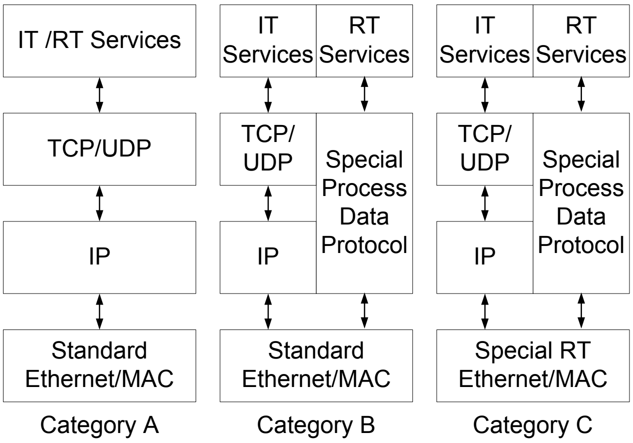

Fig. 1. Categorization of IE systems in terms of software and hardware needs for device implementations [8], [20].

This classification is used in the following and each IE system is classified according to its best RT class achievable. Moreover, an overview of the best delivery time achievable by each system is given.

A. *Established IE Systems*

This section gives an overview of established IE systems. In addition to the classification in terms of RT capability, each of the systems is categorized according to software and hardware needs for device implementations (see Figure 1) [8], [20]:

- Category A: Both IT and RT services are completely TCP/UDP/IP-based and use standard Ethernet controllers and switches.
- Category B: Both IT and RT services use ordinary Ethernet controllers and switches but a special process data protocol is introduced on top of Ethernet.
- Category C: RT services require a dedicated process data protocol like category B and additionally require special RT Ethernet controllers and switches.

1) Modbus-TCP: Modbus-TCP is solely designed for soft RT applications and therefore falls into class 1. It is located on the application layer on top of TCP/IP and completely bases on standard Ethernet components (if used on top of Ethernet) making it one of the few category A approaches [20]. It has a well known TCP port 502 to transmit data and can therefore be controlled remotely [21]. As it is located on the application layer, the protocol can be applied to all device types for communication and is very easy to configure, i.e., shows a high degree of self-configuration. It is implemented as client-server architecture whereby each device may become client or server, which contributes to the high reliability of Modbus-TCP. The server processes requests of clients and confirms them with a success or error message. The number of devices connected by Modbus-TCP is practically only limited by the computational power of the devices themselves.

2) Ethernet Powerlink: Ethernet Powerlink basically allows RT communication with standard Ethernet hardware as the protocol can be completely implemented in software (category B) [22]. However, for applications putting high requirements on cycle times of RT communication, i.e., isochronous RT (class 3), an implementation of the protocol in hardware is necessary (category C) [8]. The Powerlink protocol is located between Ethernet and IP. The developed time-slotted access method to Ethernet connections bases on a master-slave concept called Slot Communication Network Management, which provides transmission capacities for cyclic process data as well as acyclic service and control data. The master, called Managing Node by Powerlink, gives permission to the slaves to send data sequentially during a cycle. Hence, the master is decisive for Powerlink's successful operation and must not fail as otherwise, the total system is put out of operation.

The scalability in terms of connected devices is limited as Powerlink uses own 8 bit addresses. The addresses have to be manually configured but Powerlink is hot-plug capable, i.e., devices can be integrated or removed during runtime therefore allowing for some degree of self-configuration. On the application layer, Powerlink uses the CANopen standard for RT capable data transmission, which can solely be used for the transmission of process data rather than large data amounts.

3) EtherCAT: EtherCAT is an IE system, which bases on the master-slave principle and applies a procedure for the processing of cyclic process data in field devices (slaves) [23], [8].

For the communication, a process image is created in the master, which represents the state of various in- and outputs of the overall system comprising several slaves. To change the state of specific outputs of a slave, the respective part of the process image together with a change command has to be dispatched. Slaves themselves can send parts of the process image to the master during the cyclic data exchange to update state information of their inputs. The assignment of these parts of the process image to in- and outputs of the single slaves takes places via logical addresses, which are translated to physical addresses of the particular devices in the EtherCAT Slave Controller (ESC) [23], [8], [24]. Parts of the process image and the respective command to change outputs are dispatched either directly in Ethernet frames (used for achieving isochronous RT within a subnet) or as UDP payload (not RT capable for sending data from another subnet). These EtherCAT frames are cyclically sent from the master and pass the slaves sequentially on a ring structure. Contrary to common procedures of standard Ethernet controllers, which buffer and process an incoming frame and send a new one, the processing takes place completely in hardware while the frame passes the ESC (category C) [23].

EtherCAT provides a high-performant system for RT data transmission via Ethernet. By means of the fast data processing in the ESC and due to low overhead, cycle times far below 1 ms are possible (class 3). EtherCAT is intended to be easily diagnosed and configured and therefore provides configuration tools, which are able to depict the network topology and automatically configure the devices in the network. However, a master must be permanently available, which undertakes the administrative tasks and the storage of all data and addresses available in the network. The scalability is limited due to the logical process image. The advantage of the low overhead by aggregating data to various recipients would reduce substantially as soon as the network would be used for sending larger amounts of data than process data—if possible at all.

4) TCnet: TCnet developed by Toshiba extends the original access method of IEEE 802.3 to Ethernet connections and introduces four communication classes with different priorities.

The four communication classes are (sorted by transmission priority):

- Cyclic data with high timing requirements (class 3)
- Cyclic data with medium timing requirements
- Acyclic data
- Cyclic data with low timing requirements

The cyclic data transmission is used by RT application and can only take place inside one subnetwork as data is directly encapsulated in an Ethernet frame.

To enable the new access method, each TCnet participant needs special TCnet Ethernet controllers (category C). Prior to commissioning, a station number for each participant has to be configured manually, which determines the transmission sequence for the cyclic data transmission [25], [26]. Another principle used by TCnet is a common memory of the TCnet stations participating in the RT communication. Each station contains an own copy of the common memory and can therefore access all process data at any time. Thereby, the RT communication is solely intended for process data being reflected in the memory size of 256 KByte defined by Toshiba. Such a memory size is no longer feasible if larger amounts of data have to be transmitted.

5) TTEthernet: TTEthernet is a RT capable IE system, which can be basically combined with standard Ethernet systems and thereby enables a wide variety of applications.

TTEthernet defines three message types (time triggered, rate constrained, best effort) whereby the time triggered message type is designed for isochronous RT applications (class 3) [27]. To be able to send data of the stated message types, special switches are required supporting the TTEthernet protocol (category C). To dispatch TT packets at the right point in time and to block the transmission channels during this timeframe for other message types, a common time base of the switches and connected devices is indispensable. It is introduced by one or more master devices, which have to be configured manually, by means of the synchronization of distributed clocks.

TTEthernet provides a deterministic transmission method on the two lowest OSI layers. Therefore, for higher layers the transmission by TTEthernet is completely transparent and there are no specifications, which kind of data is transmitted. Basically, a RT transmission mechanism for larger amounts of data could be integrated here.

6) CC-Link IE Field: CC-Link IE Field from Mitsubishi Electric bases on full-duplex Gigabit Ethernet so that the topology is almost arbitrary and achieves hard RT (class 2

[20]). The addressing of field devices takes place by station numbers ranging from 1 to 254 so that the number of devices is limited [8]. Similar to other IE systems, CC-Link IE Field uses a master-slave principle. Thereby, the master is responsible for the initialization of the total network and controls the data transmission. It stores the state of all in- and outputs of the devices in the network as well. The information memory of the master is limited to 32,768 bits and 16,384 words and represents a summary of all device memories in the network. The data transmission takes place cyclically by means of a token passing mechanism. Starting with the master, the token is passed in the network, which allows the current token owner to send data. For realizing CC-Field IE Field either special hardware in the slave (Mitsubishi CP220 Chip) or an implementation of the Seamless Message Protocols is required (category C) [28], [29]. As stated in [20], for third parties the implementation of CC-Link IE Field is challenging leading to no visible introduction outside Mitsubishi.

7) Profinet: The communication in Profinet takes place cyclically and is divided into several phases [30]. Each cycle starts with the isochronous phase, in which Isochronous RT (IRT) frames are transmitted (class 3). The transmission of IRT frames is already configured during the installation of the network. By means of synchronized clocks, in each device the point in time is precisely scheduled when a IRT frame may be sent. The synchronization is carried out by a master. In spite of data transmission in Ethernet frames, the addressing does not take place by MAC addresses but frames are forwarded through switches on a fixed route depending on the transmission time. Therefore, special Profinet switches are required (category C). After the isochronous phase, another RT phase follows and finally, a phase for non-time critical data transmitted via UDP or TCP is provided [26]. Due to [20], the crucial issue of Profinet IRT is the complex system planning but still Profinet gained a noticeably high market share (14.5 % estimated for 2015) due to Siemens encouraging and supporting its development.

8) EtherNet/IP: Ethernet/IP bases on the Common Industrial Protocol (CIP), which is located on top of TCP/IP and UDP/IP (category A) [31]. For the RT transmission of process data, UDP/IP is used whereby a direct communication between all devices is possible. Ethernet/IP can be operated with all protocols on the application layer and a limitation of the number of devices is basically not given. However, in practice there is an obvious limitation if isochronous RT is to be achieved. Moreover, if isochronous RT has to be reached (class 3), the time synchronization has to be implemented in hardware by means of special switches with built-in IEEE 1588 timestamp support (category C) as otherwise the stack performance is not sufficient. To complicate matters, routers must have multicast/broadcast control features available and there is no standard to implement or configure these features [20].

9) SERCOS III: SERCOS III organizes devices as double ring structure with hardware redundancy or as line structure without hardware redundancy [32]. Per ring/line, a maximum of 511 devices are permitted thereby limiting the scalability. SERCOS III preserves its communication ability even in the case of an error like cable break or node failures and new devices can be integrated at runtime so SERCOS III provides a high degree of self-configuration and flexibility. The communication in SERCOS III is grounded on a time-slot method with cyclic telegram transmission on the basis of a master-slave principle. A central master sends so-called Master Data Telegrams to slaves, which can communicate with each other directly as well (Cross Communication and Controller-to-Controller communication profile). The telegrams base on standard Ethernet frames and transport only small amounts of process data. The SERCOS III master can be realized with special hardware to achieve isochronous RT (category C) or alternatively be completely implemented in software (category B, SERCOS III SoftMaster).

10) Comparison of Established IE Systems: All investigated IE systems show similar basic principles, which are solely implemented in different ways. Actually, several solutions apply a shared memory and most systems require a master (information not specified for TCnet) or a comparable management system, which controls the communication or have to be configured manually. The manual configuration effort, which has to be expended if devices have to be integrated or changed, differs according to the IE system whereby basically the effort is intended to be kept minimal for achieving high flexibility at runtime.

Most isochronous RT capable solutions have in common that new devices have to be recognized by the master to adapt the communication mechanism and assign time slots or a polling or token procedure to new devices. The master as central instance represents a SPoF and bottleneck thereby limiting the system reliability and scalability. A completely self-organizing network, in which devices act autonomously, does not yet exist. Moreover, all realizations require dedicated and expensive hardware to realize isochronous RT behavior.

Another similarity of the systems is the optimization to process data, which is especially demonstrated by the small amounts of data exchanged in RT. None of the IE systems provides for a transport protocol to be able to transport larger amounts of data, e.g., streams in vehicular communication systems [10] while ensuring deterministic predictable transmission times. TTEthernet is one exception as it aims at an RT Ethernet solution, which does not restrict the way of the data transmission on upper layers, but does not specify a protocol either.

A comparison (together with current developments) is apparent in Table I. If available the estimated market share for 2015 of the respective IE system is given as indicator for the system's market penetration. The maximum number of devices is taken from [11] if not specified otherwise and represents the case of minimum delivery time (only RT). Some facts were not available (n/a) or are not specified in the respective descriptions (n/s).

TABLE I
Comparison of established IE systems and current developments

IE system	Delivery Time [ms]	Class/ Category	Reliability: Contains SPoF?	Scalability: Max. nr. of devices	Self-configuration features?	Special hardware requirements?	Market share [%] (est. 2015 [33])
Modbus-TCP	1-15	1/A	no	Unlimited [8]	yes	no	6.4
Ethernet Powerlink	0.4	3/C	yes	4	yes	optional	4.2
EtherCAT	0.15	3/C	yes	180	yes	yes (EtherCAT Slave Controller)	3.1
TCnet	2	2/C	n/s	24	no	yes (TCnet controller)	n/a
TTEthernet	n/a	2/C	yes	n/s	no	yes (TTEthernet switches)	n/a
CC-Link IE Field	1.6	2/C	yes	254 [8]	no	optional	0 outside Mitsubishi
Profinet	1	3/C	yes	60	no	yes (Profinet switches)	14.5
EtherNet/IP	0.13	3/C	no	90	no	yes (IEEE 1588 switches)	13.9
SERCOS III	0.0398	3/C	yes	9	yes	optional	2.1
DRTP	5-10	3/B	yes	n/s	yes	no	0 (under development)
DARIEP	0.1 - 0.3	3/C	yes	n/s	no	yes (FPGA synchronization slave)	0 (under development)
HaRTKad	0.7	2-3/A	no	n/s	yes	no	0 (under development)

IV. Towards the Industrial Internet of Things: Going beyond master-slave patterns and special hardware

To summarize, among the currently established RT Ethernet technologies there is no solution without master component and special hardware. Moreover, none of the existing systems allows for a reliable RT transmission of increasing amounts of data. These issues will become more relevant in the future. As mentioned in [1], [2], the future for the industry will be more intelligent devices to be connected, which can act more dynamically. Facilities as one main area of application may consist of several thousands of devices still requiring RT behavior. For instance, the high flexibility as a rising challenge cannot be ensured in hierarchical and centralized systems due to their highly static behavior. So we think, the existing solutions will not fulfill the future challenges in terms of reliability, scalability, and flexibility as they are right now. They either need revision or new solutions will have to be developed. For example, decentralized distributed systems could help to solve these issues [34].

A. Current Developments

Consequently, there are already some developments targeting the weaknesses of established IE solutions.

(1) Distributed Real-Time Protocols for Industrial Control Systems (DRTP): Schmidt et al. [35] propose a RT Ethernet solution (class 3), which can act in a distributed manner and dynamically changes the bandwidth allocation to the shared Ethernet medium. The concept bases on two additional proprietary layers on top of the Ethernet layer to manage the media access and a master, which synchronizes the slaves by means of the IEEE 1588 synchronization protocol (category B). The result is an TDMA-approach using time slots. There is no statement about a high number of attendees and its applicability for large scale networks. Furthermore, as TCP/UDP and IP are not supported the total vertical and horizontal is not possible without further effort. The authors' work sounds promising; especially, the dynamic bandwidth allocation is an interesting self-organization feature. However, many implementation issues are subject to future work and still a central synchronization instance is required.

(2) Design and application of a RT industrial Ethernet protocol (DARIEP): In [36], a RT industrial Ethernet protocol is developed adopting a master-slave pattern (class 3). The master is developed under Linux using Real Time Application Interface and coordinates the RT data exchange with the nodes by precise cyclic timing. Moreover, the slaves base on universal FPGA and ARM chips so special hardware is necessary (category C). The authors' approach seems to be a high-performant alternative for current IE system at the expense of the need for dedicated hardware.

(3) HaRTKad: A Hard Real-Time Kademlia Approach (RTKad): Skodzik et al. [37] sketches a completely decentralized approach to realize a fully decentralized Kad network meeting at least hard RT constraints for connecting devices in automation scenarios (class 2-3). They renounce using a master but let the peers synchronize themselves by a decentralized algorithm, which makes the network comparable to Modbus-TCP in terms of high reliability [38], [39]. Each peer is assigned a time slot depending on its hash value and can communicate during this time slot in RT. A RT Kad prototype has been developed, which runs on standard Ethernet

hardware on top of UDP (category A) [37]. By applying the P2P paradigm as device connection technology, basically the SPoF in terms of synchronization and communication can be avoided. Together with the high capability of self-organization and the restriction to standard hardware, the approach is appealing.

For a complete comparison of current developments with established IE systems, see Table I.

V. Conclusion

It becomes apparent that none of the established IE systems meets all requirements concerning high reliable, scalability, flexibility in terms of self-configuration and cost-effective standard Ethernet hardware. No currently established solution achieves isochronous RT data transmission without special hardware and no SPoF at the same time. The protocols on application layer introduced or applied by the established IE systems are optimized to the transmission of process data and do not allow for the reliable RT transmission of increasing amounts of data. Only TTEthernet offers degrees of freedom in terms of a protocol for the deterministic transmission of any data but does not specify a protocol either. Current developments achieve isochronous RT but all except one require special hardware and contain a SPoF limiting their reliability and scalability.

Finally, the authors conclude that there is a need for the advancement of existing and for the development of new IE approaches to keep in step with a rising numbers of devices to be connected and increasing amounts of data to be exchanged in RT in the future Industrial Internet and Internet of Things and Services, respectively. The existence of related work in this field speaks for the validity of this hypothesis and the authors hope to have motivated the research community regarding this.

Acknowledgment

The authors would like to thank Kerstin Thurow, professor for Automation Technology / Life Science Automation at the University of Rostock, Germany, and Thilo Sauter, director of the Institute for Integrated Sensor Systems, Austrian Academy of Sciences, for their advice and assistance in the preparation of this contribution.

This work is partly granted by the Research Fund Mecklenburg-West Pomerania, Germany, as well as the European Social Fund.

References

[1] Communication Promoters Group of the Industry-Science Research Alliance, acatech - National Academy of Science and Engineering, “Recommendations for implementing the strategic initiative INDUSTRIE 4.0,” Industrie 4.0 Working Group, Tech. Rep., April 2013. [Online]. Available: http://www.plattform-i40.de/finalreport2013

[2] P. C. Evans and M. Annunziata, “Industrial Internet: Pushing the Boundaries of Minds and Machines,” General Electric, Tech. Rep., November 2012.

[3] acatech - National Academy of Science and Engineering, “Cyber-physical systems. driving force for innovation in mobility, health, energy and production.” Tech. Rep., December 2011. [Online]. Available: http://www.acatech.de/de/publikationen/stellungnahmen/kooperationen/detail/artikel/cyber-physical-systems-innovationsmotor-fuer-mobilitaet-gesundheit-energie-und-produktion.html

[4] Panel Building & System Integration, “Ethernet adoption in process automation to double by 2016,” 2013. [Online]. Available: http://www.pbsionthenet.net/article/58823/Ethernet-adoption-in-process-automation-to-double-by-2016.aspx

[5] T. Sauter, “Integration aspects in automation - a technology survey,” in *10th IEEE Conference on Emerging Technologies and Factory Automation*, 2005, pp. 255–263.

[6] ——, “The three generations of field-level networks 2014—evolution and compatibility issues,” *Industrial Electronics, IEEE Transactions on*, vol. 57, no. 11, pp. 3585–3595, Nov 2010.

[7] T. Sauter and M. Lobashov, “How to access factory floor information using internet technologies and gateways,” *Industrial Informatics, IEEE Transactions on*, vol. 7, no. 4, pp. 699–712, Nov 2011.

[8] F. Klasen, V. Oestreich, and M. Volz, *Industrial Communication with Fieldbus and Ethernet*. VDE Verlag GmbH, 2011.

[9] IEEE 802 LAN/MAN Standards Committee, “Time-sensitive networking task group,” July 2014. [Online]. Available: http://www.ieee802.org/1/pages/tsn.html

[10] T. Steinbach, H.-T. Lim, F. Korf, T. Schmidt, D. Herrscher, and A. Wolisz, “Tomorrow’s in-car interconnect? a competitive evaluation of ieee 802.1 avb and time-triggered ethernet (as6802),” in *2012 IEEE Vehicular Technology Conference (VTC Fall)*, 2012, pp. 1–5.

[11] M. Felser, “Real time ethernet: Standardization and implementations,” in *IEEE International Symposium on Industrial Electronics*, 2010.

[12] B. Wilamowski, *Industrial Communication Systems (The Industrial Electronics Handbook)*. CRC Press, 2011.

[13] *IEEE 802.1D Edition 2004 IEEE Standard for Local and metropolitan area networks - Media Access Control (MAC) Bridges*, Institute of Electrical and Electronics Engineers Std., 2004.

[14] *IEEE 802.1Q Edition 2005 IEEE Standard for Local and metropolitan area networks - Virtual Bridged Local Area Networks*, Institute of Electrical and Electronics Engineers Std., 2005.

[15] R. Moraes, F. B. Carreiro, P. Bartolomeu, V. Silva, J. A. Fonseca, and F. Vasques, “Enforcing the timing behavior of real-time stations in legacy bus-based industrial ethernet networks,” *Computer Standards & Interfaces*, vol. 33, no. 3, pp. 249–261, Mar. 2011.

[16] P. A. Laplante and S. J. Ovaska, *Real-time systems design and analysis : tools for the practitioner*. John Wiley & Sons, Inc., Hoboken, New Jersey, 2012.

[17] J. Stankovic, “Misconceptions about real-time computing: a serious problem for next-generation systems,” *Computer*, vol. 21, no. 10, pp. 10–19, Oct 1988.

[18] S. Y. Nof, *Springer Handbook of Automation*. Springer-Verlag, 2009.

[19] P. Neumann, “Communication in industrial automation—what is going on?” *Control Engineering Practice*, vol. 15, no. 11, pp. 1332–1347, 2007.

[20] M. Rostan, “Industrial Ethernet Technologies: Overview,” EtherCAT Technology Group, Tech. Rep., 2011.

[21] Modbus-IDA.ORG, “Modbus messaging on tcp/ip implementation guide v1.0b,” 2006.

[22] Ethernet Powerlink Standardization Group, “Ethernet powerlink,” January 2014. [Online]. Available: www.ethernet-powerlink.org

[23] EtherCAT Technology Group, “Ethercat technical introduction and overview,” 2013. [Online]. Available: http://www.ethercat.org/en/technology.html

[24] ——, “Ethercat - the ethernet fieldbus,” 2013. [Online]. Available: http://www.ethercat.org/pdf/ethercat_e.pdf

[25] Toshiba, “Tcnet technology,” 2013. [Online]. Available: http://www.toshiba.co.jp/sis/en/seigyo/tcnet/technology.htm

[26] M. Felser, “Real-time ethernet - industry prospective,” 2013. [Online]. Available: http://www.control.aau.dk/∼ppm/P7/distsys/01435742.pdf

[27] TTA-Group, “Ttethernet - a powerful network solution for all purposes,” 2013. [Online]. Available: http://www.ttagroup.org/ttethernet/doc/TTEthernet_Article.pdf

[28] CC-Link Partner Association, “Cc-link ie field brochure,” 2013. [Online]. Available: http://www.cclinkamerica.org/functions/dms/getfile.asp?ID=045000000000000001000009146100000

[29] ——, "Cc-link ie field white paper (clpa-2327)," 2013. [Online]. Available: http://www.cclinkamerica.org/functions/dms/getfile.asp?ID=045000000000000001000007488300000

[30] R. Pigan and M. Metter, *Automating with PROFINET: Industrial Communication Based on Industrial Ethernet*. Wiley-VCH, 2008.

[31] ODVA, "The organization that supports network technologies built on the common industrial protocol (cip) - device net, ethernet/ip, componet, and controlnet," February 2014. [Online]. Available: http://www.odva.org

[32] SERCOS, "International e.v. and sercos north america, group of associations dedicated to developing, promoting and expanding the use of the sercos digital interface," February 2014. [Online]. Available: http://www.sercos.com

[33] industrial ethernet book, "The world market for industrial ethernet components," 2013. [Online]. Available: http://www.iebmedia.com/index.php?id=8595&parentid=74&themeid=255&showdetail=true&bb=true

[34] A. Bratukhin and T. Sauter, "Functional analysis of manufacturing execution system distribution," *Industrial Informatics, IEEE Transactions on*, vol. 7, no. 4, pp. 740–749, Nov 2011.

[35] K. Schmidt and E. Schmidt, "Distributed real-time protocols for industrial control systems: Framework and examples," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 23, no. 10, pp. 1856–1866, 2012.

[36] T. Hu, P. Li, C. Zhang, and R. Liu, "Design and application of a real-time industrial ethernet protocol under linux using rtai," *International Journal of Computer Integrated Manufacturing*, vol. 26, no. 5, pp. 429–439, 2013.

[37] J. Skodzik, P. Danielis, V. Altmann, and D. Timmermann, "Hartkad: A hard real-time kademlia approach," in *11th IEEE Consumer Communications & Networking Conference (CCNC)*, 2014, pp. 566–571.

[38] ——, "Time Synchronization in the DHT-based P2P Network Kad for Real-Time Automation Scenarios," in *2nd IEEE WoWMoM Workshop on the Internet of Things: Smart Objects and Services (IoT-SoS)*, 2013, pp. 1–6.

[39] J. Skodzik, V. Altmann, P. Danielis, and D. Timmermann, "A kad prototype for time synchronization in real-time automation scenarios," in *World Telecommunication Congress*, 2014.