A CLASSIFICATION OF REAL TIME ANALYTICS METHODS. AN OUTLOOK FOR THE USE WITHIN THE SMART FACTORY

Sebastian TRINKS
TU Bergakademie Freiberg, Chair of Management Information Systems,
Silbermannstraße 2, 09599 Freiberg, Germany; Sebastian.Trinks@bwl.tu-freiberg.de

Abstract. The creation of value in a factory is transforming. The spread of sensors, embedded systems, and the development of the Internet of Things (IoT) creates a multitude of possibilities relating to upcoming Real Time Analytics (RTA) application. However, already the topic of big data had rendered the use of analytical solutions related to a processing in real time. Now, the introduced methods and concepts can be transferred into the industrial area. This paper deals with the topic of the current state of RTA having the objective to identify applied methods. In addition, the paper also includes a classification of these methods and contains an outlook for the use of them within the area of the smart factory.

Keywords: Real Time Analytics, Smart Factory, Industry 4.0, Smart Manufacturing, Internet of Things, Machine Learning.

1. Introduction

The fast technological progress of the last decade leads to fundamental changes in nearly all areas of our life and affects also the industry. Nowadays, production facilities include artificial intelligence, internetworked physical devices, cyber-physical systems, nanotechnology, and biotechnology. Such technological progress makes it possible to gain a decision support in (near) real time, because data can be gathered and analyzed at the moment of their occurrence. Besides the development of the Internet of Things (IoT), the powerful hardware creates the possibility of analyzing data with minimal latency posing the basis for decision-making in real time. The employment of analytical applications in real time enables a process automation and leads to fundamental changes in enterprises. This underlines why Real Time Analytics (RTA) can be deemed as an important concept to acquire the ability to realize Industry 4.0 scenarios. The aim of this publication is to bring together the methods of RTA by applying a literature review and classifying them according to different types of latency. Whereby, this builds the basis for further research into the possibilities of RTA methods within the area of Industry 4.0.
Currently, there are a lot of publications available which considers RTA in the context of big data. However, publications about RTA in context of Industry 4.0 or smart factory are much less existing. Hence, the knowledge of which methods of RTA from the area of big data are interesting for the Industry 4.0 is not available. In addition, most of these papers consider a single application or a single method of RTA without considering alternatives. An overview or even a classification of these methods is currently not obtainable. Accordingly, it cannot be stated, which methods from the discussion about big data have made their entry into the area of Industry 4.0. Thus, this paper contributes to the scientific discussion with an overview and a classification of RTA methods.

In our research, we collect the current state of RTA in the scientific discussion and derive their applied methods from it. Our objective is to answer the question of which methods exist and how they are used. In order to acquire an overview about these methods we will classify them by means of existing models in the area of latency.

2. Real Time Analytics within the Smart Factory

For a deeper consideration of RTA within the smart factory, it is first necessary to consider fundamentals of them. In addition to the explanation of Industry 4.0 and smart factory, this Section contains an explanation of different available expressions of Analytics and the integration of RTA within the analytical environment.

2.1. Industry 4.0 and Smart Factory

The constant networking of numerous devices and also machines via the IoT forms the basis of the fourth industrial revolution, which is titled by the term Industry 4.0. The increasing amount of sensors in devices and machines in a factory produce a huge volume of data. Processing it in real time using analytical methods leads to automation of the entire production process (Lee, Bagheri, and Kao, 2015, p. 18-23). Hence, there is a fusion of the physical and the virtual world to cyber-physical systems (CPS), which are transformative technologies for administrating the connection of physical assets and computational capabilities (Lee, Bagheri, and Kao, 2015, p. 18-23). Intelligent products can actively control the production process and lead to a new quality of automation (Kagermann, 2017, p. 235-246). In the production environment of the smart factory, material, machines, storage, and logistics systems communicate directly with each other (Dais, 2017, p. 261-277). Thereby, the objective of this continuous communication is to maximize efficiency and profitability (Wang, Wan, Li, and Zhang, 2016, p. 3159805). To achieve this purpose, there is a variety of possibilities to apply analytical methods. For example, it is possible to manage the performance of a production process continuously by using RTA (Kumaraguru, and Morris, 2014, p. 175-182). The amount of data, which was generated within the smart factory is already being discussed under the
umbrella term big data. Algorithms are used to perform statistical evaluations and to identify patterns within the data. From the gained information, new knowledge can be deduced (Kagermann, 2017, p. 235-246). In order to contribute to the automation of the processes and thus to increase the added value, it is necessary that the data processing can be done as soon as possible. RTA offers the possibility to increase added value and leads to fundamental transformations within production processes.

2.2. Concepts of Analytics

Initially, it seems to be necessary to differentiate between the terms analysis and analytics. Analysis deals with the systematic investigation of an object of observation. It is a process of obtaining and evaluating data with the target to attain information. This is a targeted evaluation, because information has a purpose and such a purpose is determined by the user. Analytics describes the theory of analysis and is used as a reservoir of all existing analytical methods. Thereby, analytics is frequently used in combination with other terms to clarify the application scenario, which leads to a broader focus on related concepts (Lanquillon, and Mallow, 2015, p. 55-89). Table 1 depicts some common expressions of Analytics.

<table>
<thead>
<tr>
<th>Concepts</th>
<th>Description</th>
<th>IEEE</th>
<th>ACM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advanced Analytics</td>
<td>Combines methods of machine learning and statistics with the purpose to derive prediction models. In addition, the term includes the areas of Predictive Analytics and Prescriptive Analytics (Chamoni, and Gluchowski, 2017, p. 8-17).</td>
<td>124</td>
<td>47</td>
</tr>
<tr>
<td>Big Data Analytics</td>
<td>Identifies Advanced Analytics techniques used in the area of Big Data (Russom, 2011).</td>
<td>689</td>
<td>290</td>
</tr>
<tr>
<td>Business Analytics</td>
<td>Describes the continuous investigation of past-oriented business data with the aim of obtaining knowledge about past and future business activities (Felden, 2017).</td>
<td>37</td>
<td>53</td>
</tr>
<tr>
<td>Descriptive Analytics</td>
<td>This is a descriptive analysis that aims to describe and summarize an event. For this purpose, tools from the reporting or OLAP (Online Analytical Processing) can be used (Lanquillon, and Mallow, 2015, p. 55-89).</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>In-Memory Analytics</td>
<td>This term implies that the data for processing and analyzing are read directly from the central memory. Access to the hard disk is not necessary in the area of In-Memory Analytics (Dinsmore, 2016).</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Predictive Analytics</td>
<td>Characterizes an approach that learns from the data of the past to predict the future to support the decision-making process (Siegel, 2013).</td>
<td>111</td>
<td>83</td>
</tr>
<tr>
<td>Prescriptive Analytics</td>
<td>Describes a prescriptive analysis that generates a recommendation for the achievement of certain business targets. It is the highest form of decision support (Lanquillon, and Mallow, 2015, p. 55-89).</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>Real Time Analytics</td>
<td>When the time between the triggering event and the required response is very low, it is spoken about RTA. If the total execution time is zero, this is referred to as a real time application (Lanquillon, and Mallow, 2015, p. 55-89).</td>
<td>42</td>
<td>62</td>
</tr>
</tbody>
</table>

1 Amount of publications in the IEEE Xplore Digital Library between 2016 and now. Title and abstract were searched for using the concept name as search term (day of the execution: 19/12/2018).
2 Amount of publications in the ACM Digital Library between 2016 and now. Title and abstract were searched for using the concept name as search term (day of the execution: 19/12/2018).
Analytics’ terms always represent a mixture and emphasize different aspects. It is not possible to make a clear distinction between the individual terms, because there exist a lot of overlaps between them. However, it can be noted that all the associated concepts can also be seen as part of RTA, because the only characterizing aspect of RTA is the application in real time and not the business domain or a defined business scenario.

2.3. Real Time Analytics

Adding the term real time, Analytics gets a time-oriented perspective. This means to allow as little time as possible to pick up occurring relevant data to deliver it to its processing. In view of this timely perspective, the term real time has to be further elaborated. The Oxford dictionary defines real time regarding data processing as follows: *Relating to a system in which input data is processed within milliseconds so that it is available virtually immediately as feedback to the process from which it is coming...* (Oxford Dictionaries). Due to this definition, the time between the triggering event and the required response has to be nearly zero to be labeled as real time application. The underlying supporting concepts are summarized by the term RTA (Lanquillon, and Mallow, 2015, p. 55-89). The objective of RTA applications is to analyze the data in the moment of its occurrence (Oxford Dictionaries). Cundius (Cundius, and Alt, 2013) divided the processing of real time applications into the following three levels (1) real time processing, (2) near real time processing, and (3) non real time processing. Thereby, the time horizon until analyses results are relevant are seconds to minutes for (1), real time processing. Within (2), near real time processing, this horizon is several minutes, and for (3), non-real time processing applications, the task to be done is less urgent. It can take hours, days, or even more (Cundius, and Alt, 2013). The time between the triggering event and the required response is the overall latency of an application in the analytics environment. Already in BI applications, the execution time is critical. According to Hackathorn (Hackathorn, 2004, p. 24) and Cundius (Cundius, and Alt, 2013), the following latencies can be separated: Data latency (required time for data provision), Analysis latency (required time for data analysis), and Decision latency (required time for decision making) (Cundius, and Alt, 2013). The following figure depicts the process of an analytic application within a flow chart.

![Figure 1. Latency of RTA according to Hackathorn (Hackathorn, 2004, p. 24)](image)

The greater the overall latency of the application, the lower the value of the information generated by the application. Accordingly, the given time frame in which the result has to be calculated and provided depends upon the definition of a project. An important requirement of
real time applications is that the results of data processing are available in due time (Lanquillon, and Mallow, 2015, p. 55-89). Determining the time frame in which a response has to be generated is contingent on the individual application. If the covered period of the task is exceeded, the result will become unusable (Furtner, Wildhölzl, Schlager-Weidinger, and Promberger, 2016, p. 163-171). In the manufacturing sector this could lead to inefficient production or in the worst case to a loss of production. Thus, RTA is a required concept for the development of a smart factory.

3. Literature Review to Identify Relevant Papers

The aim of our research is to issue an outline of the current state of RTA and specially to identify and to classify the methods of them. To reach this objective, we have chosen the literature review as research method. According to Cooper, a literature review is a research method which considers a number of primary studies with similar subjects and research objectives. Cooper's approach consists of the following five steps: (1) problem development, (2) data collection, (3) data evaluation, (4) data analysis and data interpretation, and (5) presentation of results (Cooper, 1998). In the first step of the literature review, we searched Google scholar by using the search term “Real Time Analytics” for development of the research problem. Thereby, we found 8,800 results. Thus, we limited the search to the title of the publication and as a result we found 316 publications. Table 2 depicts the sources situation, which were used in our literature review. When looking at the identified sources, we found that the sources we found using the search term "Real Time Analytics" can be classified into the following areas: tool-oriented, application-oriented, database-oriented, processing-oriented and evaluation-oriented.

Table 2.
Entries in the selected databases using the search term "Real Time Analytics"

<table>
<thead>
<tr>
<th>Database</th>
<th>Without limitation</th>
<th>Limited to title &amp; abstract</th>
<th>Limited to title</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEEE</td>
<td>122</td>
<td>83</td>
<td>0</td>
</tr>
<tr>
<td>ACM</td>
<td>73</td>
<td>62</td>
<td>19</td>
</tr>
<tr>
<td>Springer Link³</td>
<td>584</td>
<td>-</td>
<td>12</td>
</tr>
<tr>
<td>Google Scholar⁴</td>
<td>8800</td>
<td>-</td>
<td>316</td>
</tr>
</tbody>
</table>

³ Springer Link was only searched for publications that have the search term "Real Time Analytics" in the title, because the filter settings did not allow a search via abstract and title.
⁴ Google Scholar was only searched for publications that have the search term "Real Time Analytics" in the title, because the filter settings did not allow a search via abstract and title. Moreover, the quantity of sources would otherwise have been unworkable.
In addition, we could not find any publication containing the state of the art of RTA presents the applied methods. That is why we have declared our research aim to reply to the question which methods exist in the area of RTA and how they are applied. In our previous research we were able to identify that time is the characterizing element for RTA applications (Trinks, and Felden, 2017). Therefore, we take the division of the latency of Hackathorn (Hackathorn, 2004, p. 24) as a starting point for our research. After specifying our research aim, the sources for the literature search were determined. The IEEE Xplore database, AIS digital Library, and Springer Link databases were taken as sources for the literature review, because there provide numerous publications dealing with the topic of RTA. Within these three databases, the term \textit{Real Time Analytics} was searched by title and abstract. Figure 2 depicts the thematic classification of the identified publications. Since we do not want to focus on applications in our research, we removed all application-oriented publications. As a result, the number of relevant papers were reduced to 76. Fig. 3 depicts the whole process of literature review. After all, the remaining relevant papers were used for identifying the methods of RTA. In the final step, a classification model was derived from the latency model of Hackathorn (Hackathorn, 2004, p. 24). The found methods were assigned to their class.

\begin{figure}
\centering
\includegraphics[width=\textwidth]{figure2.png}
\caption{Thematic classification of the publications.}
\end{figure}

\begin{figure}
\centering
\includegraphics[width=\textwidth]{figure3.png}
\caption{Process of literature review.}
\end{figure}
4. Classification of Real Time Analytics Methods

The processing of large amounts of data in real time contains a lot of technical challenges. When the time of processing is too high, a beneficial use of the results of the completed analyses and predictions is not possible. Consequently, it is necessary to apply other technologies and methods supporting the minimization of latency. In this chapter, we introduce the methods of RTA identified by the literature review and describe their classification.

In a first step, we subdivided the methods that are supporting a latency minimization into data access methods, data analysis methods, decision methods, and data processing methods. These classes are derived from the classification of latency of Hackathorn (Hackathorn, 2004, p. 24), which were depicted on Fig. 1 in Chapter 2. We have chosen this classification, because the latency is the critical issue for RTA applications (Trinks, and Felden, 2017). Hence, we have classified the methods into the different types of latency. Added to this are the methods that support the entire processing and thus shorten the overall latency. We classify these methods using the name data processing methods. During the progress of the literature research, we found that the data analysis and the decision methods are difficult to separate as the analysis and the decision are closely connected. Thus, we put these classes together under the designation data analysis and decision methods.

![Figure 4. RTA Classification of RTA methods derived from the latencies of Hackathorn (Hackathorn, 2004, p. 24).](image)

Fig. 4 depicts the methods of RTA, which were found by literature review and their classification. Besides the found methods of RTA, also further methods of them could be classified within the existing classes. The following sections contains the descriptions of the different methods which were contained within Fig. 2.

4.1. Data Processing Methods

Processing methods serve to accelerate the data processing, especially the computational operations of an RTA application. Thereby, the purpose of the data processing methods is to minimize overall latency.

**Streaming.** The processing and analyzing of data without storage is called streaming. Thereby, data streaming is the opposite of batch processing, where the date is loaded from disk.
The main benefit of this approach is the higher velocity of the processing, because the memory reading and writing operations need the most time of the overall latency (Lanquillon, and Mallow, 2015, p. 55-89). In addition to the higher velocity of the on-the-fly processing of the data, streaming allows also to reduce storage costs (Ari, Olmezogullari, and Çelebi, 2012). Thereby, the direct processing of data streams is also called Data-In-Motion or Complex Event Processing (Lanquillon, and Mallow, 2015, p. 55-89). The avoidance of access to the hard disk generates an economy of time in Real Time Streaming applications. A significant disadvantage of the streaming approach is that the data cannot be used for further processing steps (Lanquillon, and Mallow, 2015, p. 55-89). However, in order to achieve a short as possible latency, data streaming has disadvantages in the area of flexibility. This is due to the use of static queries, which is the only possible opinion (Perera, and Suhothayan, 2015). Nevertheless, Real Time Streaming Analytics is really suitable for special usage application scenarios and can make a decisive contribution to the minimizing of latency. As a result, there are a large number of tools available now that support the streaming approach, such as Apache Storm, Apache Spark, Heron or Samza (Floratou, Agrawal, Graham, Rao, and Ramasamy, 2017, p. 1825-1836). Furthermore, streaming is described as the dominant processing method in the area of RTA (Katsipoulakis, Labrinidis, and Chrysanthis, 2017, p. 1286-1297).

Parallelization. The use of parallelization of operations is named parallel computing and is an opportunity for a fast and efficient execution of calculations. This happens with different processors or processor cores, which speeds up the execution. This architecture supports the current trend of using more than one processor core for the execution of one single system (Prassol, 2015, p. 358-372). For example, SAP HANA uses this architecture for generating performance benefits. However, also open source systems like Apache Hadoop or Apache Spark use the parallel computing for analyzing large amount of data with a lower latency (Shoro, and Soomro, 2015). The increasing use of parallel processing has been initiated with the decreasing cost of the processors that began in 2000 years (Szpisják, and Rádai, 2016).

4.2. Data Access Methods

Data access methods accelerating the writing and reading operations as well as shorten the access times are summarized in the data access methods. These serve to reduce the data latency of an RTA application.

In-Memory Databases. An important technological basis for the data processing in real time are In-Memory Database Systems. In contrast to traditional database systems, the main components of the database are kept in central memory. As a result, the access times and the data queries are faster (Mertens, Bodendorf, König, Picot, Schumann, and Hess, 2017). The inexpensive storage of large data sets is a big advantage of traditional database systems, despite increased access times. The technological progress of the central memory and the cost reduction of them are the reasons for the distribution of In-Memory Database Systems (Luo, 2014). Regarding technology, In-Memory Computing is possible since the 1980s, but only since the
cost reduction of the central memory has increased the spread enormously (Furtner, Wildhölzl, Schlager-Weidinger, and Promberger, 2016, p. 163-171). Thereby, examples of in-memory databases are SAP HANA, Oracle 12C (Szpisják, and Rádaí, 2016) or MemSQL (Chen, Jindel, Walzer, Sen, Jimshileishvili, and Andrews, 2016, p. 1401-1412).

**Column-oriented Database.** The architecture of In-Memory database systems differs from the traditional disk-based database systems. Here, column-oriented databases are applied, storing the attributes in columns instead of rows, like they are usually saved in traditional relational database systems (Larson, Birka, Hanson, Huang, Nowakiewicz, and Papadimos, 2015. The advantage of the column-oriented data storage is the calculation, which can be executed within one or a few columns. In addition, a table can be searched for attributes of just a small set of columns (Prassol, 2015, p. 358-372). Especially in the area of Analytics, this generates speed advantages. Furthermore, this architecture makes it possible that only the required columns have to be kept available in the central memory, thereby ensuring an efficient use of the other ones. Access to specific values is possible by a query that is stored under a defined row number (Mertens, Bodendorf, König, Picot, Schumann, and Hess, 2017). Hence, it is possible to execute analytical applications in a smaller timeframe compared to traditional row-oriented database systems (Herden, 2013). An example of a column store engine is the Apollo engine included in the SQL Server (Larson, Birka, Hanson, Huang, Nowakiewicz, and Papadimos, 2015).

**Database Compression.** Compression methods for databases have the purpose to reduce the amount of data, which are necessary to read. Values, which exist more than once, will be replaced by abbreviations and will be translated into original value by a dictionary, if necessary (Bößwetter, 2010, p. 61-65). However, only lossless compression methods can be used in the database area. Examples of these methods are dictionary compression, run-length encoding and differential storage (Herden, 2013). In order for the compressed data to be usable for the analyses, an additional step of decompression is necessary. If the compressed data is kept in the central memory and available powerful processors, the access to this data will be faster than accessing uncompressed data on the hard drive (Mertens, Bodendorf, König, Picot, Schumann, and Hess, 2017). As an example, the database system TRISTAN should be mentioned here, because it supports real-time analysis directly on the compressed data (Marascu, Pompey, Bouillet, Wurst, Verscheure, Grund, and Cudre-Mauroux, 2014).

**4.3. Data Analysis and Decision Methods**

In order for a beneficial use of the analysis results, the overall latency must be as low as possible in RTA applications. However, especially in socio technical systems where humans are part of the decision-making process, the decision latency can be deemed the critical issue, due to the fact that one requires more time for a decision (Lanquillon, and Mallow, 2015, p. 55-89). Therefore, it is important to automate and not just support the decision-making process. Through the use of analytical methods from the area of data mining, knowledge can be obtained
from large amounts of data that can be used as the basis for decision-making. Methods from the field of machine learning are used to make a decision based on the recognized data patterns (Witten, Frank, Hall, and Pal, 2016). Thereby, an important part of an analytics environment are the applied algorithms. In the area of RTA, an approximate analysis with a lower latency is preferred, compared to an exact analysis with a higher latency, because it is necessary that the result is calculated in real time. Therefore, techniques and methods for reducing and summarizing data are applied (Kejariwal, Kulkarni, and Ramasamy, 2015, p. 2040-2041). The selection of the analytical method depends strongly on the application. Since analytical methods are used to analyze data and to make decision based on it, we summarized these methods under the class data analysis and decision methods.

**Data mining.** Data mining is a collection of techniques, methods, and algorithms for analyzing data (Cleve, and Lämmel, 2016). Thereby, it described the extraction of knowledge from data by pattern recognition and contains the opportunity to solve problems by analyzing data which is already present in databases (Witten, Frank, Hall, and Pal, 2016). Data mining needs a set of data which is divided into a training set and a test set. The training set will be used for learning and the test set will be used for the validation of the developed model (Cleve, and Lämmel, 2016). Data mining algorithms can be found in the areas of clustering, classification or association rules, for example (Witten, Frank, Hall, and Pal, 2016). In addition, concepts like Real Time Data Mining (Djorgovski, Graham, Donalek, Mahabal, Drake, Turmon, and Fuchs, 2016, p. 95-104; Chen, Man, Li, Sun, Wong, and Yu, 2014, p. 414-429) or Data Stream Mining (Parikh, and Tirkha, 2013, p. 5234-5239; Rahnama, 2014, p. 789-794) make it possible to analyze data with Data Mining techniques in near or real time. For example, StreamDM is a library for Apache Spark, which contains components for Data Stream Mining (Bifet, Maniu, Qian, Tian, He, and Fan, 2015).

**Machine learning.** Machine learning enables systems in the factory to understand their environment, plan actions, respond to obstacles, and communicate with people (Gewiehs, 2017, p. 30-31). Thereby, the term machine learning means the implementation of learning algorithms within a computing-based resource. It describes the automatic computation process of pattern recognition and decision making based on a training data set (Dua, and Du, 2016). In addition, machine learning is divided in supervised and unsupervised learning. If the discovered patterns are completely unknown, it will be called unsupervised learning. No examples with a predefined result are available in this case. An example is the cluster analysis. If there are examples available which set the result, it is spoken about supervised learning. An example technique of it is the classification. This approach assumes that future data will behave similarly to the given example. Conclusions are drawn from the past and transferred into the future. Thereby, it is necessary that the given sample quantity is representative (Cleve, and Lämmel, 2016). One important requirement to machine learning applications is the low latency. The emerging tasks for machine learning applications make it necessary to process the data within milliseconds (Nishihara, Moritz, Wang, Tumanov, Paul, Schleier-Smith, and Stoica, 2017).
There are packages and libraries for the various tools in the area of RTA available. For example, Apache Spark contains the MLlib and spark.ml packages, which include machine learning algorithms (Rahnama, 2014, p. 789-794).

In addition, machine learning could be seen as a subdomain of artificial intelligence, which aims to enable machines to take over human activities. To achieve this objective, the learning behavior and the development of the human memory will be imitated (Felden, 28.11.2016). Thus, the target should be reached through the use of collective knowledge, which has to be saved and expanded. Furthermore, it forms the basis for the recognition of knowledge that can be discovered by pattern (Laudon, Laudon, and Schoder, 2010).

4.4. Software tools for Real Time Analytics

As data processing and analyzing of data in real time becomes increasingly significant, the market contains a lot of different tools that support this. A large part of the considered sources, which deals with the topic of RTA, consider software tools that enable RTA. Most of these tools use a combination of the methods described in the previous section to speed up data processing (Perera, and Suhothayan, 2015). Thereby, the tools could be subdivided into batch and streaming processing (A.A.E.S.Y., &. Baldominos, 2014). In addition, there are also tools available supporting both approaches. Table 3 shows some of the most discussed open source tools in the area of RTA in the literature from 2016 to now.

Table 3. Overview of open source Real Time Analytic tools

<table>
<thead>
<tr>
<th>Name</th>
<th>Processing</th>
<th>Website</th>
<th>IEEE 5</th>
<th>ACM 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Akka</td>
<td>Stream</td>
<td><a href="http://akka.io/">http://akka.io/</a> (last visit: 23.01.2018)</td>
<td>10</td>
<td>9</td>
</tr>
<tr>
<td>Apache Flink</td>
<td>Batch and stream</td>
<td><a href="https://flink.apache.org/">https://flink.apache.org/</a> (last visit: 24.01.2018)</td>
<td>21</td>
<td>22</td>
</tr>
<tr>
<td>Apache Kafka</td>
<td>Stream</td>
<td><a href="https://kafka.apache.org">https://kafka.apache.org</a> (last visit: 23.01.2018)</td>
<td>17</td>
<td>3</td>
</tr>
<tr>
<td>Apache Samza</td>
<td>Stream</td>
<td><a href="http://samza.apache.org/">http://samza.apache.org/</a> (last visit: 23.01.2018)</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>Apache Spark</td>
<td>Batch and stream</td>
<td><a href="https://spark.apache.org/">https://spark.apache.org/</a> (last visit: 24.01.2018)</td>
<td>281</td>
<td>104</td>
</tr>
<tr>
<td>Apache Storm</td>
<td>Stream</td>
<td><a href="http://storm.apache.org/">http://storm.apache.org/</a> (last visit: 24.01.2018)</td>
<td>51</td>
<td>20</td>
</tr>
<tr>
<td>Heron</td>
<td>Stream</td>
<td><a href="https://twitter.github.io/heron/">https://twitter.github.io/heron/</a> (last visit: 23.01.2018)</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Pulsar</td>
<td>Stream</td>
<td><a href="http://gopulsar.io/">http://gopulsar.io/</a> (last visit: 23.01.2018)</td>
<td>58</td>
<td>3</td>
</tr>
</tbody>
</table>

5 Amount of publications in the IEEE Xplore Digital Library between 2016 and now. Title and abstract were searched for using the name of the software as search term (day of the execution: 26/01/2017).

6 Amount of publications in the ACM Digital Library between 2016 and now Title and abstract were searched for using the name of the software as search term (day of the execution: 26/01/2017).
The tool Apache Spark was mostly considered in scientific publications within this timeframe. The framework of Spark contains tools in the area of machine learning or graphical analysis (Dinsmore, 2016). Thereby, Spark is a hybrid solution that supports data streaming, such as batch processing, to reach the objective of accelerating data analysis. The In-Memory approach, which reduces the time of reading and writing, is also applied by this solution. Spark is based on parallelization, which means that the analysis of large data sets can also be accelerated (Shoro, and Soomro, 2015).

There are also many publications about Apache Storm which uses the approach of parallelization and is called Hadoop for real time (van Rotterdam, 2016, p. 17). Apache Hadoop is a batch processing tool (Venner, Wadkar, and Siddalingaiah, 2014, ) while Apache Storm is a representation of Stream Analytics. Thus, data are not stored in Apache Storm, but processed directly (Iqbal, and Soomro, 2015. p. 9-14). Parallelization, as the used streaming approach generates a high velocity in which data can be processed (Yadav, 2017).

5. Summary of the Obtained Results

Scientific publications about RTA have been widely spread in recent years. In the three selected sources for our literature review, we identified 157 relevant contributions since 2013. However, most publications focus on the subdomains of software tools or applications, as shown in Figure 2 on page five. Our publication contributes to the clarification of the methods and the homogeneous of the understanding of RTA. The outcome of our research contains an outline about a state of the art of RTA methods, which were mostly spread in the scientific discussion. We have already found out in previous research (Trinks, and Felden, 2017) that the decisive characterizing element of RTA is the time and specially the latency. Therefore, we have used the division of the latency by Hackathorn (Hackathorn, 2004, p. 24) for the classification of RTA methods. Previously, such a classification was not available in scientific literature and can serve now as a basis for further research. The developed classification is not evaluated now and in a next research step, it should be reviewed. Our research is a contribution to the development of a uniform definition of RTA, which are currently not available. Thus, this forms a starting point for future research. Thereby, the outcome of our research can also be helpful for the design of new RTA applications and specially the design of application in the area of smart factory. Implementing the idea of smart factory or the vision of Industry 4.0 is not possible without RTA. Thereby, the huge amount of generated data has to be processed with a least possible latency. However, not every identified method is required for such an application. By the use of a combination of some of these methods, the benefits of them could be partly reversed. Thereby, it depends on the application which selection of methods makes sense. In part, the benefits cancel out when certain methods are used together. Further research
may be concerned with the aspect regarding the useful combinations of methods to reach the highest possible benefit. Thereby, this paper provides the necessary overview and declaration of the existing methods of RTA and the assignment of which latency them could minimize.

Mostly, the identified and described methods do not come from publications related to the area of smart factory. It is therefore necessary to evaluate whether a transfer of the individual methods into the application area of the smart factory is possible without restrictions. Also, possible barriers have to be considered. Furthermore, our literature review was limited to the sources ACM digital library, IEEE Xplore, and Springer Link. Thus, methods published in other sources could not be identified within our research approach. Additional methods could be found by the search in further data sources. Again, there is a starting point for further research.

6. Conclusion

The technical progress of hardware and the continuous networking of all devices and machines in a factory through the IoT shows the possibility to use RTA applications to support decision making and a processes automation within the smart factory. Based on this, the objective of this paper is the increase of the understanding of RTA and the identification of applied RTA methods. Thereby, we considered, how these methods are used and which latency minimization of the application are supported by them. Thus, we have done a literature review to consider this issue. Once we have identified the methods, we made a classification of them. We developed a classification model, which was derived from the classification of latency by Hackathorn (Hackathorn, 2004, p. 24) and put all identified methods into it. The outcome of this paper is interesting for researchers, because it contains suggestions for further research in the area of RTA within the context of smart factory. However, also practitioners can benefit from the results, because this contribution could be used to increase the understanding of possible RTA application methods. It can be used as a basis for decision makers within the introduction of RTA in the industry. Nevertheless, our research is limited to the methods discussed in the scientific area. Methods, which may already be used in practice, but have not been scientifically investigated, yet, are not included. However, additional methods can be classified into the existing classes of the proposed model. Further methods can be found, for example, by surveys in the industry.

The purpose of our future research is the consideration of the opportunities of RTA applications in a smart factory. In this context, the following research question should be taken into account: Which smart factory applications use the detected methods of RTA? Besides the identification of the applications, we would like to investigate the potentials and limitations of the methods of RTA found within these applications. Furthermore, we will consider, what
contribut (Lee, Bagheri, and Kao, 2015, p. 18-23) ion can be made to the creation of value by the use of RTA applications.

Bibliography


43. Szpisják, P., and Rádai, L. (2016). Performance issues of In-Memory Databases in OLTP systems. Applied Computational Intelligence and Informatics (SACI), 11th International Symposium on IEEE.