Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 56 (2015) 325 – 332

Procedia Computer Science

The 10th International Conference on Future Networks and Communications (FNC 2015)

Drilling Data Management in petroleum industry based on RTPS

Basem Almadani*

KFUPM-Computer Engineering Department, Dhahran 31261- Saudi Arabia

Abstract

The availability of real-time drilling data communication is highly critical to the exploration and production business of any petroleum company. Such critical mission requires oil companies to use the cutting edge technology because doing that can increase the profitability of the company by millions of dollars. Moreover, even short delay in data transfer could results in severe damages, injury, environmental pollution, or even loss of lives. Data Distribution Service (DDS) is the state of the art in middleware technologies, standardized by Object Management Group (OMG) in 2003; it is designed for distributed low latency and high reliability systems. In this paper, we review the current drilling data communication technique in a selected Oil and Gas Company, which will be called (OGC), and propose an optimization solution based on DDS technology. Furthermore, we evaluate the performance of DDS middleware with different data types, data loads, and different networks including limited bandwidth networks. A prototype for our proposed solution is implemented using LABVIEW and RTI-DDS and evaluated in terms of delay and throughput. The results show that DDS is a promising technology that can improve the performance of petroleum industry while maintaining the correct safety margins.

Peer-review under responsibility of the Conference Program Chairs

Keywords: Drilling, Real-time, Data communication, Data Distribution Service, Petroleum, Oil and Gas

1. Introduction

Filling up your car or paying the natural gas bill is the final link in a long chain of businesses that make it possible for us to enjoy these efficient, convenient and economical forms of energy. The entire chain is known as petroleum business or industry. It includes the exploration processes, extraction/drilling, refining, transporting using oil tankers and pipelines, and marketing petroleum products. The largest volume products of the industry are fuel oil and gasoline (petrol). Petroleum is also the raw material for many chemical products, including pharmaceuticals, solvents, fertilizers, pesticides, and plastics[19]. Our concern in this paper is on the drilling process, which is used to verify the existence of crude oil after exploration process is done. In drilling process, a well hole is created using a drilling rig that rotates a drill string (has a sharp bit at the end) with certain speed, Rotate per Minute (RPM), into the earth.

For efficient drilling process, the monitoring and control processes and drilling parameters enhancement should be on real-time basis, i.e. as it occurs. Drilling parameters are enhanced continually during drilling process the drilling real-time data cycle is illustrated in figure 1. This cycle starts at the well-site from the data acquisition of

* Corresponding author. Tel.: +966-13-860-7424; fax:+966-13-860-3059.
E-mail address: mbasem@kfupm.edu.sa

Peer-review under responsibility of the Conference Program Chairs
doi:10.1016/j.procs.2015.07.215

sensors measurements from down-hole and other surface measurements (these measurements will be elaborated more in background section), where some software is used to filter out unwanted readings and extract the information that will be sent to the remote monitoring center using wide area network communication technologies, e.g. satellite communication. At monitoring center, this information is received by database servers and used by the end users experts to monitor process activity on real-time basis; and simulators are used to simulate the future activity based on historical data matching analysis. Thus, if there are corrections then new parameters is extracted and new drilling plan is issued and send back to the remote well-site to take place in the future drilling activity[15].As a result of fast and accurate data flow in drilling activity, a great help is provided to the decision makers to correct and improve the process activity. Drilling activities are very critical, where any aspect of delay in data transmission can cause severe damages in equipment, nearby region environment, and even loss of lives. For example, the drilling process in offshore cost $200,000 per day[4], thus any unwanted delay might increase the cost significantly; therefore, monitoring the drilling data efficiently reduces the operational cost and increases process safety.

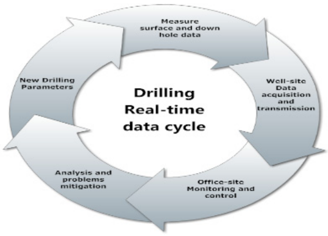
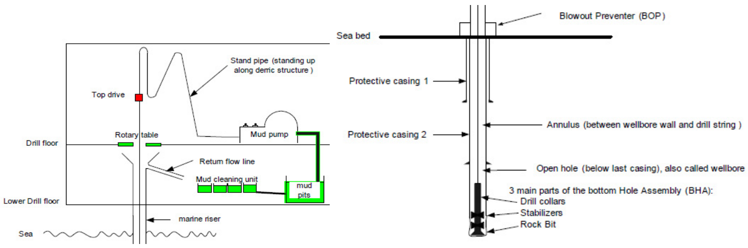


Figure 1. Drilling real-time data communication cycle



(a) Upper part (b) Down Part

Figure 2. Offshore drilling process[6]

The selected OGC is one of the biggest Oil and gas companies around the world[17], and it is one of the top three largest in the world in crude oil production where it produces around 11 million barrels per day. In this paper we review the current state of drilling process data communication in OGC and discuss the issues that can be enhanced using Data Distribution Service (DDS) technology. DDS is standardized in 2003 by Object Management Group[28] (OMG) as the first Real-Time Publish/Subscribe (RTPS) Middleware standard. It is a promising technology that can significantly improve the real-time distributed systems; it has been used in many real world fields such as military and industrial applications. A few proprietary DDS solutions had been available for several years. Starting in 2001, two major DDS vendors, the American group Real-Time Innovations and the French Thales Group teamed up to create the DDS specification which was subsequently approved by the OMG resulting in Version 1.0 in 2003[16]. The rest of this paper is organized as follows: the next section gives a background about the existing architecture used in drilling process in OGC, and in the same section we give an overview about DDS technology and their QoS services that can be utilized by drilling process. The literature review is discussed in section 3. In section 4, our proposed solution is described in details. An intensive performance evaluation is presented in section 5, including experimental work and results & analysis. Then, we conclude our paper with some recommendations, conclusions, and open research issues are also discussed in the same section.

2. Background

2.1. Drilling process in petroleum industry

Drilling process comes after the exploration process to ensure the existence of crude oil. Several rig contractors and different service companies implement and manage the drilling process at well sites. The tool used to drill an oil well called rig and illustrated in figure 2, which is subdivided into two main parts, the upper part (On surface, figure 2a) and the down part (below the surface, figure 2b)[22]. The drilling string is protected from the well walls and underground water layer by using multiple casing layers sealed off the drilling string all the way until the drilling bit. Measurement While Drilling (MWD) tool, is associated with the BHA to efficiently taking the measurements while drilling, such that monitoring the down-hole conditions at real-time. The most measurements taken by MWD tool are: rock properties, drilling direction, torque, Weight-On-Bit (WOB) (used to speed up the penetration rate), and wellbore pressure. The wellbore pressure is a very important measurement, where it helps do balancing in the wellbore and mud pressure to avoid the blowout occurrences, which is a very risky event in drilling[2,22,23]. Further, Mechanical Specific Energy (MSE) is a very important measurement that calculates the work that is being performed to penetrate a given volume of rock. It helps to identify the best drilling parameters and justify any design changes, such as bit selection, BHA design, markup torque, directional target sizing and motor differential ratings. The MSE depends on the fact that the input energy from the rig (Rotate Per Minute (RPM), WOB, torque and pump pressure) is equivalent to the output energy (vibration and Rate Of Penetration (ROP)), the vibration must be minimized to optimize the ROP. The following equation has been defined to calculate MSE [8]:

$$\mathrm{MSE} = \frac{480 * \mathrm{Torque} * \mathrm{RPM}}{\mathrm{Dia}^2 * \mathrm{ROP}} + \frac{4 * \mathrm{WOB}}{\mathrm{Dia}^2 * \pi}$$

The blowouts are very dangerous, it would result in injure, rig distortion, harm the environments, and even lives loss. Thus, monitoring the down-hole conditions on real-time basis is very important for drilling safety, efficiency, and also it significantly reduces the drilling time and cost [22].

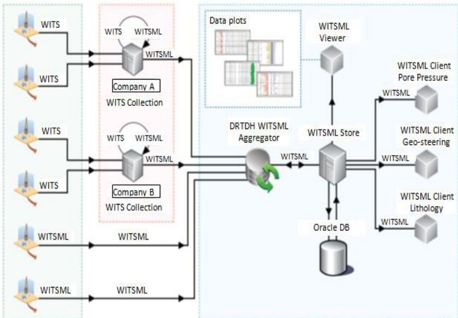


Figure 3.Current real-time well site data flow at OGC[8]

Thus, for more economic and efficient drilling process the monitoring and control cycle should be carried out on real-time basis. As described earlier, the cycle starts from applying the drilling plan with certain parameters, and then sending the well measurements (taken by MWD tools and mud system loggings) to remote monitoring and control center. Herein, the real-time data is analyzed and new drilling plan parameters is generated and sent back to the remote well site to execute the new corrected plan.

2.2. Current drilling infrastructure in OGC

The real-time data flow from well site to the monitoring and control center in OGC is illustrated in figure 3. The figure shows the real-time drilling and completion data architecture used by OGC; it is leveraging the Well Information Transfer Standard Markup Language (WITSML), which is a continually developing industry standard for the transmission of real-time, historical and contextual drilling and completions information. The measurements are collected using different tools from different companies each of which has its own data format and it converts it to WITSML standard as shown in the figure. Then this data is collected by a central hub in the remote monitoring and controlling site called Drilling Real-time Data Hub (DRTDH) and then distributed to the other parts of the monitoring and control system through a central WITSML store server.

2.3. DDS Overview

Middleware is a software layer between an application and the operating system. Network middleware isolates the application from the details of the underlying computer architecture, operating system and network stack. Network middleware simplifies the development of distributed systems by allowing applications to send and receive information without having to program using lower-level protocols such as sockets and TCP or UDP/IP. Data Distribution Service (DDS) is network middleware for real-time distributed applications. It provides the communications service programmers need to distribute time-critical data between embedded and/or enterprise devices or nodes[20].With DDS, systems designers and programmers start with a fault-tolerant and flexible communications infrastructure that will work over a wide variety of computer hardware, operating systems, languages, and networking transport protocols.

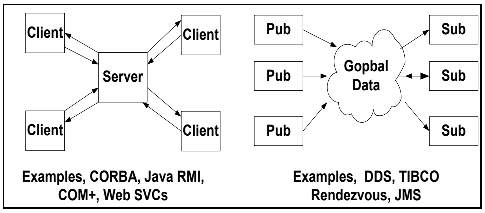


Figure 4. Publish-subscribe vs. client-server architecture

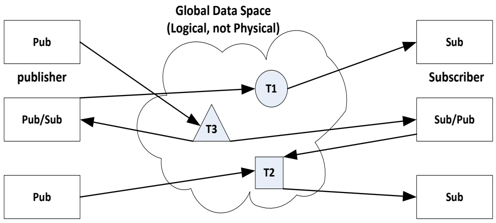


Figure 5. DDS architecture

3. Literature review

So far, many middleware solutions for optimizing and integrating control systems have been employed including Web Services, OPC and COBRA[12,14]. Interestingly, in recent years DDS approaches have been implemented in integrating control systems. For example, Tarek[26] proposed an architectural design that interacts with the main components described by the DDS specification. The aim of this interaction is to integrate the DDS QoS parameters to improve data delivery, to optimize network behavior and to integrate the network resources control to a middleware in order to enable applications to have more precise control over their end to end resources. Ageriano[1] suggested using DDS as the backbone for manufacturing communications at the control level. Particularly, DDS is used to combine sending critical information such as periodic information with less critical information related to configuration, diagnostic or maintenance operations. The focus of this work is mapping of different kinds of traffic involved among the controllers at the control layer, such as PLCs, robots or Manufacturing PCs, over DDS. This technique utilizes the QoS mechanisms offered by DDS. Other research has proposed using DDS automation environments such as in Ryll and Ratchey[21] where DDS QoS mechanisms are analyzed from the point of view of automation applications or in Clavo[7], where the implementation of publisher/subscriber communications over the IEC61499 standard are proposed.

The authors in Tijero and Gutirrez[27] analyze how DDS guarantees the real-time behavior through the mechanisms included in the standard, and proposes some extensions to this standard. Furthermore, they analyze other approaches to build distributed systems based on object distribution and remote procedures calls which can guarantee predictability. It also presents how to use DDS to enhance real-time applications. The work in Poza[18] proposes middleware architecture, called Frame Sensor Adapter Control (FSA-Ctrl), to control distributed systems. The aim of this work is to provide a QoS level between the communications layer and the control layer. This architecture depends on the use of a hierarchical communications structure called logical namespace tree and a structured set of control processes interconnected, called logical sensors graph. In this architecture the communication layer is based on DDS and the component QoS requirements are managed by use of message queues. Control components can make important decisions about distributed questions, like components mobility or information redundancy detection, by means of QoS policies.

4. Proposed solution

The main proposed architecture is illustrated in figure 6. As we can see, the well-sites have different service company each one has its own WITS data format, but based on Khudiri[13], all the different service companies has to adhere to WITSML format to shorten the delay of data conversion from WITS to WTISML, where actually in current situation there are some of the service companies working in OGC changed to WITSML standard. Thus, our solution assumes that all the companies have started transmit their data in WITSML standard. Consequently, our proposed architecture has a DDS WITSML-Enabled, which means that the DDS standard has to be modified such that it can read WITSML format easily and quickly; and because they are both based on XML, it will be an easy mission to build a data model that enable DDS to read WITSML data. Many QoS services can be utilized at well-site location such as, TIME-BASED FILTER, CONTENT-BASED FILTER, and RESOURCE MANAGEMENT.

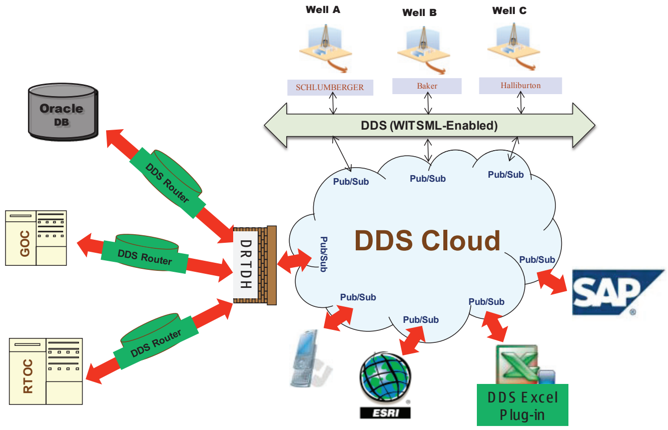


Figure 6: DDS-based proposed solution for real-time drilling data communication

The most reason that slow the deployment of DDS in the industry field is that it is still not mature enough and a lot of academic studies still evaluating it. However, DDS community is now growing up and as we mentioned early many companies competing to sell DDS products, even there are now some sensors and communication devices which are DDS-enabled. Thereby, as in figure 6, it is most likely that in the near future DDS will be enabled in other communication systems like PDA (Personal Digital Assistance), mobile phones, and other management systems such as Excel, Data base, and SAP (Systems Application Programming); such that these applications can be smoothly integrated with the whole systems to form one single system. Further, the cost of operating separate systems, tools, and data formats will be significantly reduced by operating single system from rig to end user reviewers. Also, the inefficiencies introduced by not being able to treat data from all rigs uniformly will be disappeared.

5. Performance evaluation

In this section, we describe our experimental work in evaluating the DDS middleware technology with data and video under different network situations, i.e. wired, wireless, and limited band width. In addition, we describe the implementation of our proposed solution prototype and evaluate its performance.

5.1. DDS technology evaluation

As shown in Figure 7, the latency results in case of using different number of subscribers for different packet sizes it is clear that it increases exponentially with packet size. In most cases, the measurements data that needs to be in real time not to go beyond the 1024 packet size which takes few milliseconds, which is acceptable in real time systems[10]. Also, in case of LAN communication the delay is in microsecond and the difference between LAN and Bluetooth is about 15 ms in case of packet size less than 1024 bytes, see figure 7. Some results of the delay with statistics are shown in table 1. These results give an idea about the behavior of DDS over limited bandwidth networks, especially for embedded systems. Also, still there are more places for improvements in DDS to be more suitable for limited bandwidth networks like in Sunnyvale[25].DDS is also subjected to further examination over different conditions in network and traffic types as illustrated in figures 10. In figure 8, we test the ability of DDS to transmit video traffic, it shows that also DDS is a promising middleware that can be used in video transmission; this is also has been proven in several researches [9,11,3]. In figure 9, two of the most important QoS parameters have been tested, the reliable and best-effort. Because DDS is working over UDP network protocols it can provide both services to the end user; as we can see in the figure the reliable consumed BW is higher than in best-effort and that is intuitive because of the acknowledgments' packets. Also, the throughput increases linearly as number of subscribers increase. The last result is shown on figure 10, which measures the effect of the interference in wireless communication; here the difference between the transmission with and without interference increases as number of subscribers increases, which means that the interference has significant impact on the DDS scalability metric. The interference here is introduced by using general purpose network.

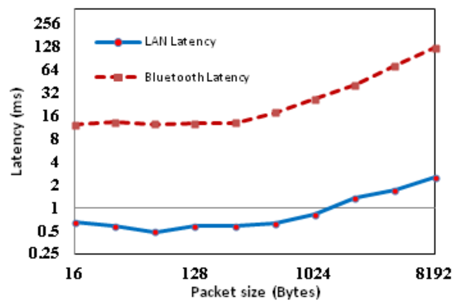


Figure 7: DDS latency in LAN and Bluetooth networks

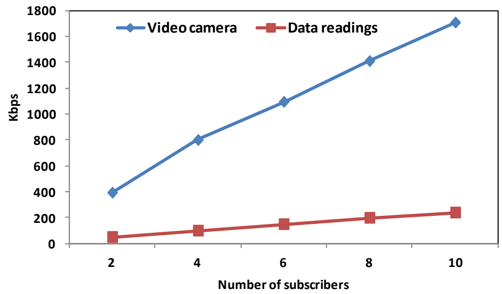


Figure 8: Different data types over DDS

5.2. Proposed solution evaluation

In this section, we describe our prototype implementation that has been used in our evaluation. Lab-VIEW and RTI-DDS were used as the main tools for implementing the prototype. And the measurement tools were as the previous work in DDS evaluation. The Heat Exchanger scenario in lab-VIEW is used in our prototype to represents the measurements coming from the drilling process; where hot air passes through several cooling units and then exits the system as cold air. In this system, a set of coolers are considered as publishers where each publisher is responsible to publish the temperature using real-time DDS to an external excel sheet which is considered as subscriber and as a monitoring tool for the measurements coming from the well-site, as an example. The experiment was carried out using hardware and software tools; the measurement and monitoring tools and hardware platform specifications that were used are described in Table 2.

The experiment test-bed is shown in Figure14, the backbone communication is based on publish-subscribe DDS middleware over WLAN network. The workload used in these experiments to evaluate the performance was periodic traffic which is represented by sensors' data readings of temperature. As is depicted in the figure, five machines were used to push the traffic towards the sixth machine where the measurements were taken. For each experiment, we computed the average throughput of 10 runs; each run lasted for 1 minute, in order to examine accurately the changes in the bandwidth. We started by testing the throughput of the background of our WLAN to precisely calculate the capacity left available for data streams. Five different scenarios were built with different data rates, i.e. 1, 20, 40, 60, 80 and 100ms as well as with different number of publishers, i.e. 1,2,3,4 and 5. For 1/ms update rate, the throughput increases as the number of publishers increase. The total bandwidth consumed in case of 5 publishers was almost 3.54 Mbps.It is worth to mention here that one update per one millisecond is very high rate in real time systems. In this scenario the throughput is about 0.11 for 5 publishers with rate of 40ms while it doubled with rate of 20ms, it shows that as the publishing rate increases the throughput increases. Also, 100 ms still considered as a real time system.

6. Conclusion and future work

In this paper we review the current state in drilling process and its real-time data communication in petroleum industry. As a case study our research was focused on the OGC Company infrastructure. As Data Distribution Service middleware has emerged as a promising technology for transfer and manage real-time data in distributed systems, also its performance has been proven in commercial products and academia researches. Thus the main research question in this paper was; is this technology suitable for petroleum industry, and if yes how petroleum industry could utilize its capabilities? This paper is a one step for answering this question. Our results show that it is a promising technology for improving drilling process in terms of operation cost reduction while maintaining the safety conditions. In future work, we intend to do intensive experimental work on real data extracted from one well of OGC and apply the DDS QoS parameters to see their effect on optimizing drilling activities in real-time scenarios.

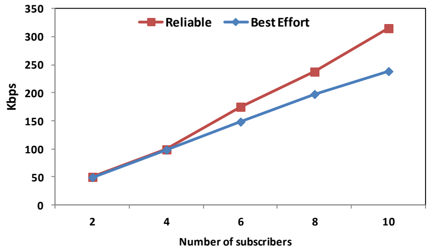


Figure 9. DDS QoS test (reliable and best-effort)

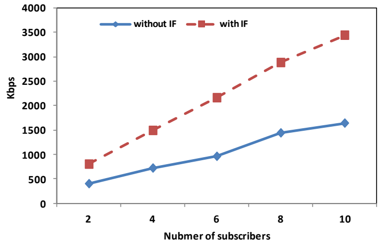


Figure 10. DDS over WLAN with and without interference

Table 1: Latency result 1-Publisher, 8-Subscriber

Packet Size (Byte)	ST-DEV	AVG (m Sec)	MIN (m Sec)	MAX (m Sec)
16	267.7	34.302	24.859	177.158
32	263.2	33.897	24.794	142.513
64	268.5	34.126	24.88	162.346
128	264.2	35.528	25.467	157.276
256	263	36.554	26.968	217.634
512	309.7	52.064	38.28	959.974
1024	295.2	81.645	60.886	283.519
2048	253.7	128.084	97.066	459.703
4096	200.3	226.154	199.678	712.288
8192	188.6	418.849	352.366	1278.287

Table 2:Hardware and software used in the experiment

	Tools / Platform	Purpose / Specifications
	LabVIEW	Publisher / ver. 2011
	Wireshark	Get network throughput / ver. 1.2
	MS Excell	Subscriber / ver. 2010
	Network	802.11g WLAN 54 Mbps
PCs' Specif.	CPU	Intel(R) Core (TM) i5 2.40 GHZ
	Memory	1.8 GiB
	OS	Windows7

Acknowledgments

We would like to thank the Real-Time Distributed Systems Lab in Computer Engineering Department at King Fahd University of Petroleum and Minerals, Dhahran- Saudi Arabia. And also we would like to acknowledge Real Time Innovation Company for their support during this research.

References

1. (Ageriano, 2011) I. Etxeberria-Agiriano, I. Calvo, F. Perez, and G. de Albeniz, “Mapping different communication traffic over DDS in Manufacturing environments,” in Information Systems and Technologies (CISTI), 2011 6th Iberian Conference on, june 2011, pp. 1 –6.
2. (Al-Khudiri, 2010) Al-Khudiri, M.; Naser, A.; Al-Shehry, M; Al-Nassir, A.; Mokhtar, H.; “Saudi Aramco RTOC, Collaboration, Safe and Effective Delivery of Wells from Start to Finish,” Saudi Aramco Journal of Technology, Spring 2010.
3. (Al-madani, 2012) Al-madani, B.; Al-Roubaiey, A.; Al-shehari, T.; "Wireless video streaming over Data Distribution Service middleware," Software Engineering and Service Science (ICSESS), 2012 IEEE 3rd International Conference on , vol., no., pp.263-266, 22-24 June 2012.
4. (Angar,2013) AgnarAamodt, Pal Skalle, Erik Gundersen, FrodeSormo, Jorgen Solstad; “Method and System for Monitoring a Drilling Operation,” Patent Application Publication, United States, Pub. No. US2013/0080062 A1, Mar. 28, 2013.
5. (Angelo, 2008) Angelo Corsro,” The OMG Data Distribution Service for Real-Time Systems” opensplice DDS, PrismTech, 2008.
6. (Ben-Duheash, 2009)Ben-Duheash, A.; Batouq, M.; “Saudi Aramco Abqaiq Plants Partners with Integration Objects to Optimize Plant Operations,” Integration Objects, July 28, 2009.
7. (Calvo, 2010) I. Calvo, F. Pe andrez, I. Etxeberria, and G. Mora andn, “Control communications with DDS using iec61499 service interface function blocks,” in 2010 IEEE Conference on Emerging Technologies and Factory Automation (ETFA), sept. 2010, pp. 1 –4.

8. (Carey, 2005) R. Carey, C. Estes, J. Fisher, L. Lagin, D. Mathisen, C. Reynolds, and R. Sanchez, “Status of the use of large-scale corba-distributed software framework for NIF controls,” ICALEPCS2005, Geneva, Switzerland, 2005.
9. (Detti, 2010) A. Detti, P. Loreti, and N. Blefari-Melazzi, “Streaming H.264 Scalable Video over Data Distribution Service in a Wireless Environment”WOWMOM '10 Proceedings of the 2010 IEEE International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM).
10. (Hermann, 2011) Hermann Kopetz, “Real-Time Systems, Design Principles for Distributed Embedded Applications,” Second Edition, Springer, 2011.
11. (Jose, 2008)Jose M. Lopez-Vega, Javier Sanchez-Monedero, Javier Povedano-Molina, Juan M. Lopez-Soler."QoS Policies for Audio/Video Distribution Over DDS Middleware," Real-time and Embedded Systems Workshop 2008.
12. (Kalogeras, 2006) A. Kalogeras, J. Gialelis, C. Alexakos, M. Georgoudakis, and S. Koubias, “Vertical integration of enterprise Manufacturing systems utilizing web services,” Manufacturing Informatics, IEEE Transactions on, vol. 2, no. 2, pp. 120 – 128, may 2006.
13. (Khudiri, 2008) Khudiri, M. M., Shehry, M.A. and Curtis, J.D.: “Data Architecture of Real-Time Drilling and Completions Information at Saudi Aramco, “ SPE paper 116848, Abu Dhabi International Petroleum Exhibition and Conference (ADIPEC), Nov. 3, 2008.
14. (Li, 2012) J. Li, J. Zhang, N. Zheng, and X. Li, “The process control system based on LabVIEW in a hardening die steel production line,” in Information and Automation (ICIA), 2012 International Conference on, june 2012, pp. 206 –209.
15. (Millheim, 1988) Millheim, K.; Tulsa, Okla; “Method of Drilling a Well Utilizing Predictive Simulation with Real-time Data,” Patent Number. 4,792,534; Dec. 27, 1988.
16. (OMG, 2007) Object Management Group. “Data Distribution Service for Real-time systems,” 1.2 formal/07-01-01 edition, January 2007.
17. (Petro, 2012) Petro Strategies, Inc. - Consulting, Research and Training, cited in 10 Dec. 2012, available from, http://www.petrostrategies.org/Links/Worlds_Largest_Oil_and_Gas_Companies_Sites.htm
18. (Poza, 2009) J. Poza, J. Posadas, and J. Sim, “QoS-based middleware architecture for distributed control systems,” vol. 50, pp. 587–595, 2009.
19. (PSAC, 2012) Industry Overview. website of Petroleum Services Association of Canada (PSAC), 2012 (http://www.psac.ca)
20. (RTI, 2011) Real-Time Innovations, “RTI Data Distribution Service,” RTI, October, 2011.
21. (Ryll, 2008) M. Ryll and S. Ratchev, “Application of the data distribution service for flexible manufacturing automation,” in Proc. of World Academy of Science, Engineering and Technology, jul 2008, pp. 178 –185.
22. (Skalle, 2011) Skalle, P. "Pressure Control During Oil Well Drilling", Pal Skalle&Ventus Publishing APS, 2011
23. (Solids, 2010) Solids Control, "Drilling fluids (mud) Function and Type," Solids Control, 2010.
24. (Stout, 2008) E. Stout, R. Carey, C. Estes, J. Fisher, L. Lagin, D. Mathisen, C. Reynolds, and R. Sanchez, “Corba-based distributed software framework for the nif integrated computer control system,” Fusion Engineering and Design, vol. 83, no. 23, pp. 535 – 538, 2008.
25. (Sunnyvale, 2012) Sunnyvale, Calif. “RTI Inroduces Limited-Bandwidth Plug-Ins to Address the Challenginges of Constrained Network Communication,” Real-Time Innovations (RTI), August, 2011.
26. (Tarek, 2007) Tarek Guesmi, Rojdi Rekik, Salem Hasnaoui, and Houria Rezig,“Design and performance of DDS-based middleware for real- time control systems,” IJCSNS International Journal of Computer Science and Network Security, vol. 7, no. 12, December 2007.
27. (Tijero, 2012) H. P. Tijero and J. J. Gutirrez, “On the schedulability of a data-centric real-time distribution middleware,” Computer Standards & Interfaces, vol. 34, no. 1, pp. 203– 211, 2012.
28. Object Management Group (OMG), www.omg.org