Archives of Transport System Telematics

Volume 7

Issue 4

November 2014

Cloud computing as a tool in smart communication with a motor vehicle

E. GRZEJSZCZYK
WARSAW UNIVERSITY OF TECHNOLOGY, Koszykowa 75, 00 – 662 Warsaw, Poland
EMAIL: egrzejszczyk@zkue.ime.pw.edu.pl

ABSTRACT

The development of on-board electronics and automotive on-board networks, as well as of wireless data transmission systems (GSM, TCP / IP) enabled online monitoring and control of selected processes of a motor vehicle. One of the most important steps was the development of the GPRS protocol, enabling data transmission (including measurement data) over long distances, as well as integration with other network protocols, such as the internet. The first implementations of these solutions such as BMW's telematics services (including iDrive and iAssist) slowly become history. However, these solutions are among the many telematics services, whose development require proper tools. This article aims to explain the issues related to the use of cloud computing for the purposes of these services (SaaS, PaaS, IaaS), as well as to present selected solutions.

KEYWORDS: remote control, intelligent communication systems, cloud computing services, SaaS, PaaS, IaaS

1. Introduction

The introduction of in-car digital buses (in the 70s/90s) to the measurement and control infrastructure was the beginning of intense development of on-board networks. (Fig.1) At the current stage of development, the most important of them are: LIN, CAN, (CAN ++) and, the latest, FlexRay [5] (Fig.2). Communication between the microcontrollers underlying the operation of these networks requires specialised software. The simple act of uploading a control program into the driver's local memory is currently not sufficient. It has to be synchronised with other events and processes in the vehicle's control and measurement infrastructure. A programmable control system requires strictly defined measurement interfaces and data transmission procedures synchronised by time or system events. This behavior is typical of operating systems, and this type of proposed system, named OSEK/VDX (*Vehicle Distributed Executive*)[1] was presented for the first time around 1995 [15]

1.1 Software architecture

On-board software layers are shown in Figure 3. These are typical layers networked microcontroller software. The "Basic Software" layer, whose functionality corresponds to a local operating system, is particularly noteworthy. This layer manages both the *Communication Protocols*, *Utility Functions Services* and *Hardware Input/Output Drivers*, as well as *Task Management - Operating System*. Over the Basic layer, there is a programmable *Runtime Environment*.
This layer is responsible for translating requests and tasks from the *Application Software* layer into tasks executed by the Basic layer. The lowest layer, *ECU Hardware* consists of microcontrollers and peripheral devices (sensors, actuators, etc.), whose software is located in the HAL, or *Hardware Abstraction Layer*.

[1] control and transmission system

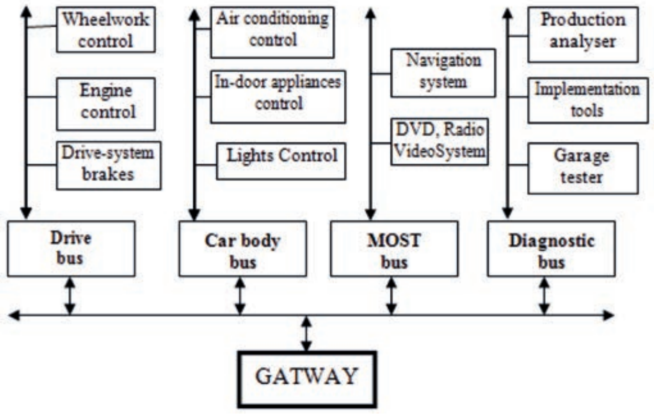


Fig. 1 Digital buses installed in vehicles [9]

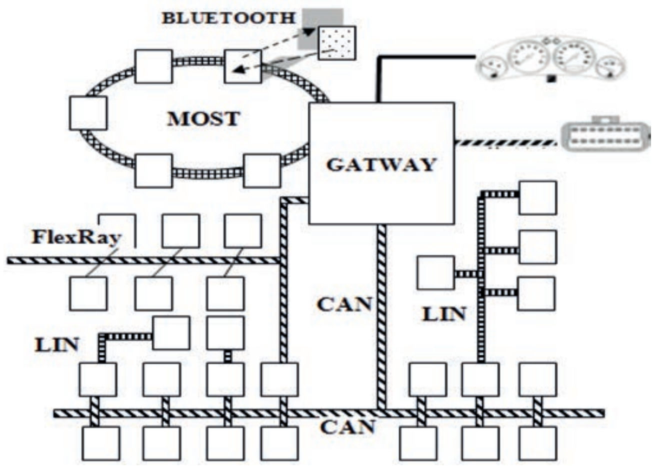


Fig. 2 Typical on-board networks [based on 14]

<table>
<tr><td colspan="4">Application Software (Component 1...n)</td><td rowspan="6">Flash Layer</td></tr>
<tr><td colspan="4">Runtime Environment (Software Bus)</td></tr>
<tr><td colspan="4">Basic Software</td></tr>
<tr><td>Hardware Input/ Output Drivers</td><td>Task Management</td><td>Utility Functions (Services)</td><td>Communications Protocols</td></tr>
<tr><td colspan="4">HALL Layer (Hardware Abstraction Layer)</td></tr>
<tr><td colspan="4">ECU Hardware (Microcontroler and Peripherals)</td></tr>
</table>

Fig. 3 Software architecture of future drivers by AUTOSAR [4] and HIS[2]

1.2 The development of off-board communication systems

User communication with on-board diagnostic system is enabled by diagnostic scanners connected to a standard 16-pin diagnostic connector of a vehicle. The average diagnostic scanner is equipped with software to perform the following functions:

- communication between the tester and the vehicle's drivers
- user interface both in hardware (display, keyboard) and logical (interpretation of data, the ability to create queries, database management) sense and
- a set of utilities that are responsible for initialising operation, data backup, as well as software updates

A functional block diagram of a typical scanner is shown in Figure 4 [5]. The connection between a diagnostic scanner and an on-board diagnostic connector may be wired or wireless, and the scanner communicated with the user using standard identification codes.

Remote diagnostic testers

Most of the currently used diagnostic testers cooperate with a computer, which is used to communicate with the on-board diagnostic system. The software controlling the scanned imports data from the reader and saves them in system memory, as well as allows for further data process in other applications (e.g. drawing charts in Excel, printing or archiving data).

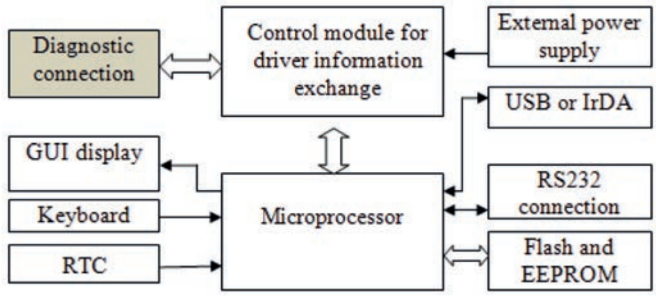


Fig. 4 Block diagram of a diagnostic scanner [5]

Computer diagnostic programs run on Windows XX and support (depending on the manufacturer) most diagnostic protocols on the market.[3] Diagnostic data stored in computer memory can be or are a source of information for further processing, including on-demand **read-outs by remote on-board network monitoring systems.**

2. On-board microcontroller as a GSM network

The dynamic development of wireless networks becomes the foundation for implementing more and more advanced applications. One of them is the communication between the on-board controller with the vehicle's user developed by the Technical University of Warsaw [9]. The on-board microcontroller should continuously monitor chosen measuring sensors[4] connected to its inputs. The microcontroller acts as a wireless transceiver station. In the case of any irregularities (such as a cracked window or an attempted break-in), the microcontroller initialises a connection

[2] AUTOSAR (Automotive Open Systems Architecture) and HIS (German: Hersteller-Initiative Software) are associations of manufacturers working on the standardisation of testing and design methods for on-board electronic systems.

[3] e.g. a VAG diagnostic scanner supports the following protocols: KWP1281, KWP2000, CAN BUS (VW, SKODA, SEAT, AUDI); AMX 550 scanner supports all transmission protocols included in the following standards: ISO 9141, ISO 14230 (Keyword 2000), SAE J1850 (PWM/VPW), ISO 15765 (CAN bus). AMX 550 allows for diagnostic scanning of all types of vehicles equipped with OBDII/EOBD compliant transmission [5]

[4] for example bi-stable sensors (door lock sensor, sunroof sensor, central lock, etc.)

with the owner (with their phone number). The system and software allow for multiple procedures. One of them is remote querying of chosen safety measures (such as asking if the windows are closed or checking the fuel level).

The system works in two basic modes: initiation (Mode 1), when the microcontroller connects to the owner of the vehicle (to report irregularities, such as an attempt to open the door) and paging (Mode 2), in which the owner initiates a connection to remotely check the state of devices supervised by the microcontroller.

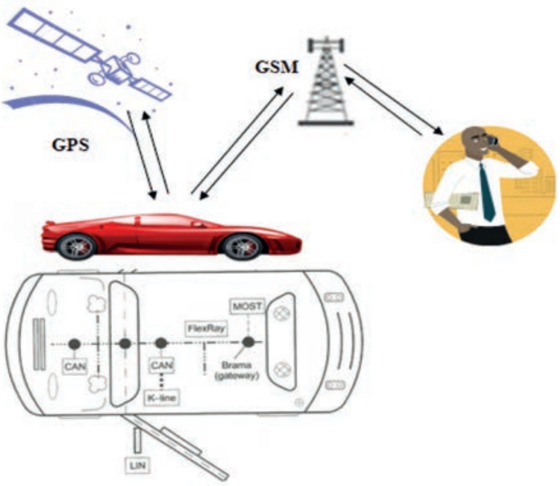


Fig. 5 Remote communication between the vehicle's owner and the speaking module of on-board microcontroller [9]

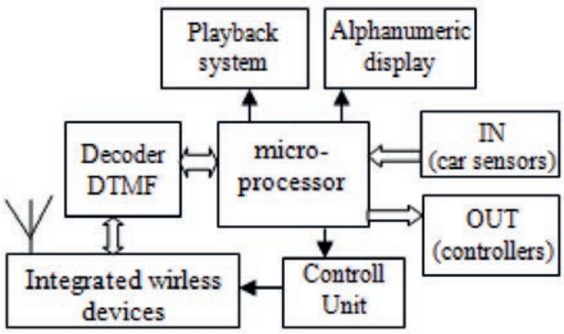


Fig. 6 A diagram of the system [9]

Principle of operation. DTMF (Dual Tone Multi Frequency) messages received by the GSM device are decoded by a DTMF decoder and transferred to microprocessor's bus. The microcontroller interprets the received data using an application saved in the system memory. On the other hand, a signal from the systems coupled with the processor to any of its inputs triggers reverse software procedures, that send a message to the vehicle's owner's phone concerning the event.

The on-board microcontroller's transceiver function mentioned above [9] is cited in this paper as an example of many emerging telematic services combining digital technologies, telecommunication and information technologies provide new concepts and functionalities located in cloud computing, as described below.[5]

2.1 Logging the on-board controller into the GSM network

The microcontroller board should be integrated/included in the GSM network. Such integration is possible by set up a connection with GSM controllers (as well as GPRS/UMTS) - Fig. 7 [9]. Such connection means that the network login process will check the caller's (microprocessor's) authenticity, its current location (the so-called LAI - Local Area Identifier), assign a Temporary Mobile Subscriber Identity (TMSI), and the parameters such as LAI and TMSI will be stored in the VLR register. The described activation procedure involves the following controllers: **MS** - *Mobile Station*, **BTS** - *Base Transceiver Station*, **BSC** - *Base Station Controller*, **MSC** - *Mobile Switching Centre*, **GMSC** - *Gateway Mobile Switching*, **VLR** - *Visitor Location Register*, **HLR** - *Home Location Register*.

The order of information exchange between the controllers is shown in the sequence diagram for UML(Fig. 7).

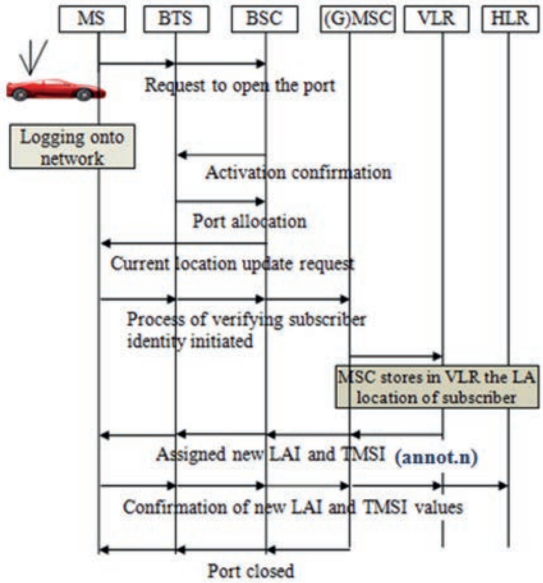


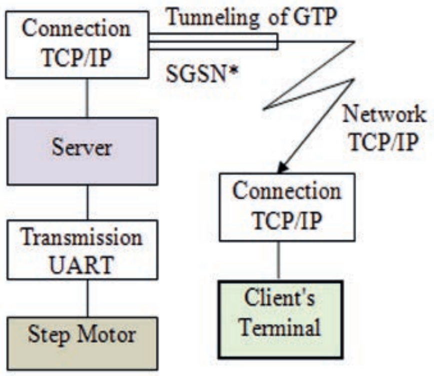
Fig. 7 User network login, annot. n) The VLR register assigns to the subscriber the TMSI identification number and LAI area identifier, which are encoded and sent to the mobile station and saved to the SIM (Subscriber Identity Module) card [9]

3. On-board controller as a TCP/IP network user

The development of the GPRS networks concept allowing packet wireless data transmission (including measurement data) enabled and started the process of remote measurement and control of multiple systems based on this technology[6]. One example of this is the remote control of a low-power executive system using TCP/IP (tunnelled through GTP), which is described below. (Fig. 8 and Fig. 9) [11]

[5] a description of the telematic services in communicating with a motor vehicle (BMW services), is provided in [7], [8]

[6] a broader description of GPRS technology is included in [5]



* SGSN - controller Serving GPRS Support Node, GTP - GPRS Tunnelling Protocol

Fig. 8 Model of remote control over the TCP/IP protocol [11]

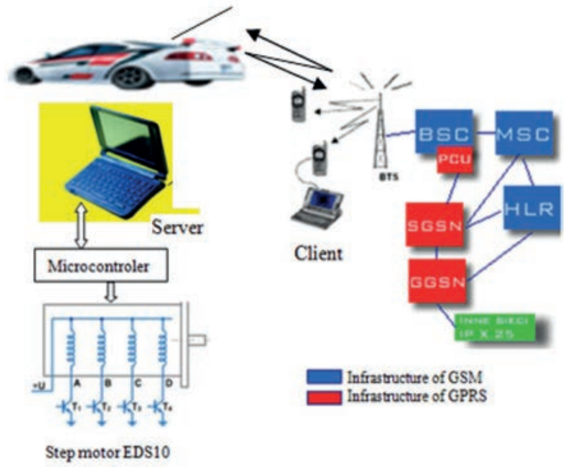


Fig. 9 Pictorial diagram of remotely controlled step motor [11]

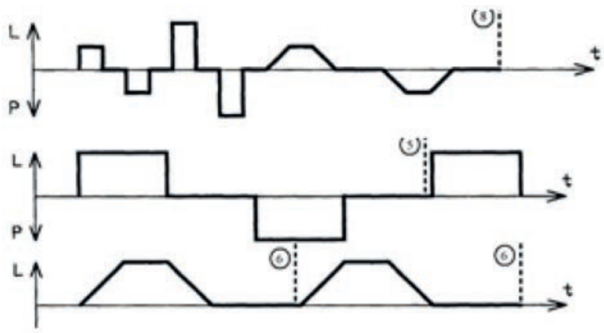


Fig. 10 Characteristics of remotely controlled step motor [11]

Figure 11 shows a user interface where request and control parameters are input. This screen is also used for feedback messages from the controlled system. The exact operation of the system and a description of the control program is included in [11]. For the sake of this paper, it has to be noted, that the communication between the user and the terminal device allows for full integration.

This means that the user can both remotely control the system and read the state of the system. In special circumstances, they can read the memory of the terminal device (on-board microcontroller) **via TCP/IP,** which stores, among others, the measurement/diagnostic data from the on-board microcontroller.

Moreover, such reading can be done (1) periodically or (2) at the request of the user or (3) the microcontroller can use the interrupt system to send a service/read request. (e.g. when exceeding the max/min measurement range or other irregular event)

An on-board real-time operating system continuously monitors the operation of hundreds of the vehicle's on-board drivers, making cyclical measurements of the equipment's state and saving the readouts in system memory. The SIM/USIM card containing information such as a user identification code, their authentication key and algorithm, encryption algorithms and phonebook **can be coupled to and supported** from the on-board diagnostic system. (see item 2 Fig. 6)

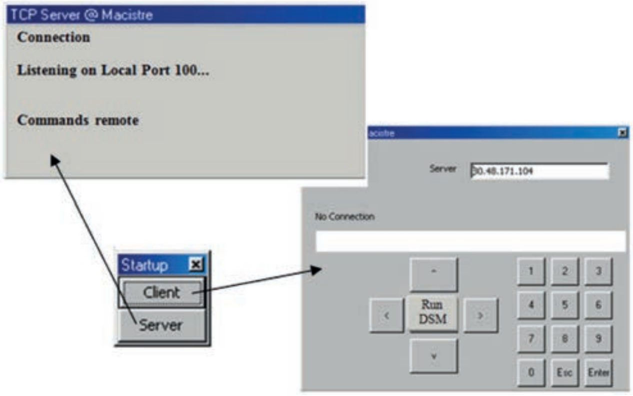


Fig. 11 User interface in the remote control system - Server/Client mode [11]

From the telematics service provider's side (GSM or other network), this card looks like an ordinary phone number subject to standard procedures. The range of services provided in this way depends on both the software and the model of the monitored vehicle, as well as the scope of purchased services. [7],[8].

As the number of users/companies is growing, there is a rapidly increasing need for memory to support applications, as well as for their efficient management. The spatial location of subscribers, as well as the dynamic changes in their location that necessitate constant system updates are an important element of the system. Monitoring and programming a distributed user environment means managing a distributed database environment that requires specialised management strategies. (Fig.13). Such solutions can be provided by cloud computing. Application, event and function management concepts provided by cloud computing are advanced technologies, which perform real-time analyses of different strategies for querying huge datasets, optimise different variants of communication technology, as well as base their actions on decision support systems in order to develop the best strategies. Tuning the system to the needs of the user/application is means **system intelligence**, which enables the development of numerous new applications, which would be impossible without the support of cloud computing. An example would be the use of telematics applications for intelligent transport systems (including automotive).

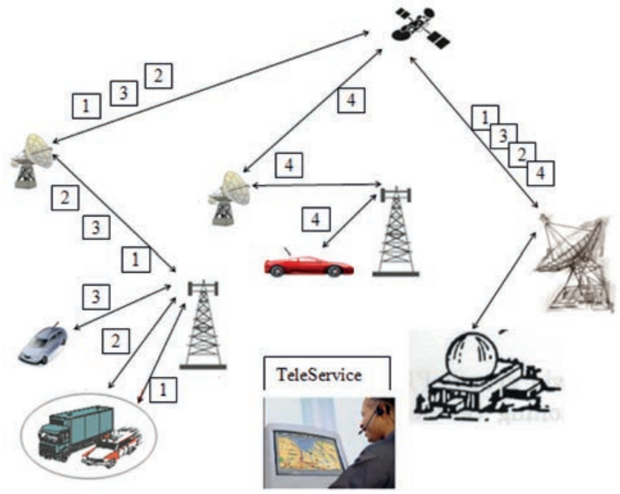
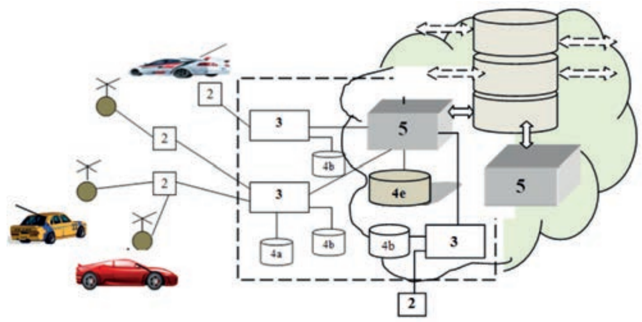


Fig. 12 Telematic transmission of information (for example diagnostics) [5]



2 - **LMT** Local Manager of Transaction; 3 - **GMT** - Global Manager of Transaction (for Local Area Identifiers) ; 5 - **MC** - Module of Communication (for Local Area Identifiers); **4a** - Log out users items; **4b** - Register of Users (temporary); **4e**- Register for LAI Users

Fig. 13 Data Base distributed management system[7]. Database sharding[8] [own study]

4. The concept of work with cloud computing. Basic elements

Cyt.: " *By 2014, cellphones and other mobile devices will send and receive more data each month than they did in all of 2008. Three-fourths of the total will come from Internet access and nearly all the rest from audio and video streaming .A big part of the increase in mobile data will come from cloud computing applications. Utility software (such as maps), will lead the way, followed closely by productivity tools (especially for sales, data sharing, and collaboration), then social networking and search. So predicts ABI Research, a telecom analysis firm in Oyster Bay, N.Y.* "[9]

The exponential growth of mobile devices and applications that support them forces the rapid development of IT tools and computing power necessary to meet their requirements. In order to support the applications, web servers are clustered and equipped with access mechanisms for enormous data centres.

Fig. 14 Photo of Google data center in Dalles, Oregon, built near a dam in order to gain access to cheap energy. [17]

Basic cloud services offered to the users are: virtualisation[10] of computing power and Data Centre resources, dynamic resource scaling, reliable data security mechanisms and guaranteed privacy of information access. The outsourcing services gained "customary"[11] names according to the resources being provided. The most popular are: **IaaS** - Infrastructure as a Servise (equipment provided) **PaaS** – Platform as a Service – (licensed software as a service) or **SaaS** – Software as a Servise – software (proprietary software from the provider) as a service.

The most well-known cloud providers are Internet corporations such as: Google, IBM, Microsoft and Amazon.[12] Due to the available programming tools (Visual Studio 2012 with C# language and the Azure SDK), the study of selected issues (concerning databases) will focus on Windows Azure from Microsoft.

5. The Windows Azure Platform

The components of the Windows platform are shown in Fig. 15. The basic branches of the Platform are: **Windows Azure** (the platform's operating system), **Windows Azure AppFabric** (Supervising data access (*Access Control*) through the data bus (*Service Bus*), and **SQL Azure** – a relational database based on Microsoft SQL Server operating using transactional query language.

One aspect of the platform's environment important to this study is the organisation of non-relational data in the, which uses a special storage environment called Azure Storage, with some characteristics of a virtual drive (.vhd). Azure Storage ("Data") has its own concepts and techniques of organising data access. These are: **Azure Blobs, Azure Tables and Azure Queues**.

Azure Blobs (*Binary Large Object)* store binary data blobs (binary large objects) in containers identified by name:

[7] Elements of two-phase commit protocol - 2PC (distributed algotithm)
[8] Sharding means distributing a database into many smaller units (called shards), that support certain requests independently.
[9] http://spectrum.ieee.org/telecom/wireless/cloud-computing-drives-mobile-data-growth

[10] the virtualisation of data centres means on-demand provision of a fragment of computing power of the Centre, along with resources for data storage, and/or selected software services.
[11] there is no cloud computing standard yet
[12] Internet corporations such as Amazon, Yahoo, Google, Intuit and Apple, have accumulated mega-centres with thousands of servers, that serve as a starting point for cloud computing infrastructure and make these companies leaders in proposed solutions

http://<account_name>.blob.core.windows.net/<container_name>/<blob_name>

Azure Tables contain EAV-type (Entity/Attribute/Value) structural table data as entities[13], which may be combined into partitions located in different cloud nodes. Each table in Azure Tables is identified by a unique identifier: http://<account_name>/table.core.windows.net/<table_name>.

Azure Queues are different cloud process and service requests stored as messages[14]. Queues can store any number of messages and the unique identifier of the queue is: http://<account_name>.queue.core.windows.net/<queue_name>

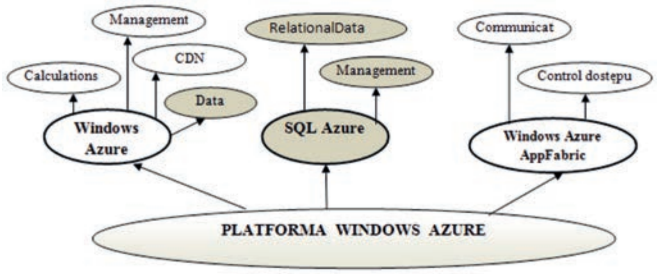


Fig. 15 Components of the Windows Azure Platform [13]

6. Case study. Cloud SQL as IaaS and PaaS in intelligent communication systems with motor vehicles

The system is an example of a cloud computing application using Azure Storage to support binary objects/files stored in the system's memory. Data are saved in databases at any time by the end users. The system database is stored at a pre-set http address. (e.g. http://www.zkue.republika.pl)

The data (structured/unstructured) are then queried by the operator/administrator according to the criteria entered by them. Processes supported by the Service require 3 defined roles (types of virtual servers), which communicate using Azure Queues.

Business vision of the system.

A cloud system allows registered users to store certain information (such as measurement data) in the system database. User registration means that they are assigned a given memory address in the database. All information generated by the user are stored in the assigned memory space, for which they have write permissions.

The system operator/administrator may at any time recall the image of data entered by the end users using their interface (http://UI). The administrator is able to select and display any record entered into the database on their terminal. This is done using a UI form and selecting users (e.g. from a drop-down list)

One the operator's request is identified, the system begins to process it (including Azure Storage objects), which generates a response in a feedback form. The response may be text or processed graphic or media[15]. This intelligent interaction is possible due to the combination of codified files, as an effect of cloud processes.

Project vision of the system

The basis of a project is a definition of data model. In this case, it is a relatively simple relational model used to build the Operator's interface.

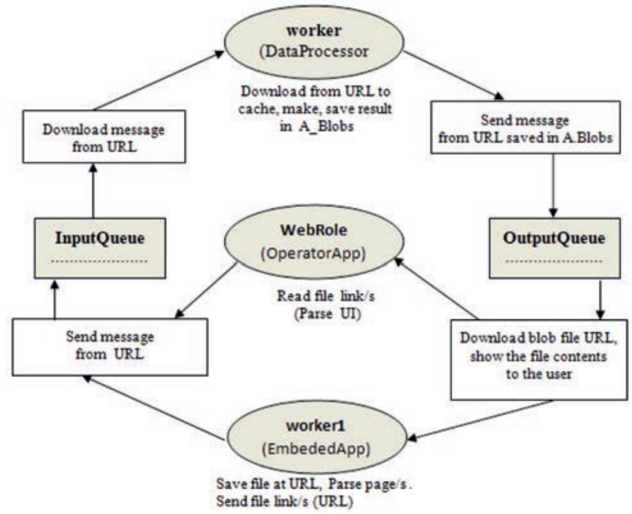


Fig.16 Schematic diagram of role interaction between WebRole (OperatorApp), workerRole (DataProcessor) and worker1Role (EmbededApp)

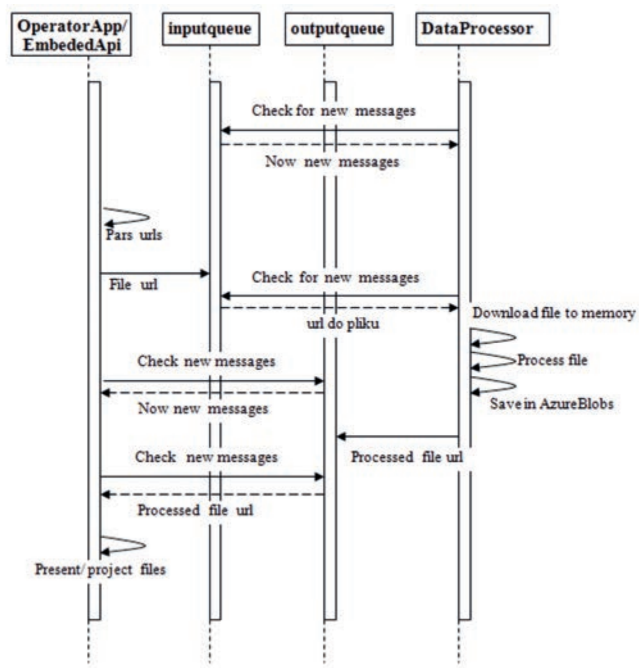


Fig.17 UML sequence diagram showing the communication between service components/roles

The log on and log off processes use the authentication and authorisation mechanisms available in the ASP.NET framework of

[13] Permitted/maximum size of a single entity is 1 Mb

[14] A single message can not occupy more than 8 kB

[15] The form of presentation depends on an application

VS 2012 environment, which have their own business logic.[16] The service includes three roles (virtual servers): Web (OperatorApp), worker (DataProcessor) and worker1 (EmbededApp). The Web (OperatorApp) role is responsible for downloading (from the operator UI) a link or links to http:// stored data, which are then downloaded by the worker (DataProcessor) to the local cache and stored in AzureBlobs. In the final phase, the OperatorApp (web) presents the associated files in the UI interface. The worker1 (EmbededApp) role is responsible for placing the relevant records/data at a given identifier http://. The interaction between individual roles and their message workflow is shown in Fig. 16 while the algorithm of cooperation between system components (resources) is shown in Fig. 17 using a UML sequence diagram.

7. Conclusion

Citation: Arthur Mateos, Jothy Rosenberg: "the most significant transformation IT has ever undergone: In the case of the cloud, we're confident that the future will involve a pervasive, game-changing reconstruction of IT from the ground up that'll take a full generation to occur." [17]

To prove the above sentence, we should quote several ongoing processes that served/serve as a foundation for the reconstruction cited above. These are

- massive increase of hardware power, and the resulting equal increase in processing power. The price of 1 megaFLOPS[17] has fallen from about USD 33 thousand in 1976 to USD 0.02 now. [17]
- the cost of hard-drive space worth over USD 200 (*in the nineteen-eighties*) is now about USD 0.01 [17]
- huge advancement in networking technologies with a bandwidth of 100 to 1000 Gb/s and the availability of fast wireless networks with a bandwidth of up to 150 Mb/s (LTE)
- the development of computer technologies as a result of implementing advance information processing algorithms and models
- the resulting implementation of data warehouse methods, the co-called Business Intelligence that provide the analytical inference tools such as DataMining, Decision Support Systems (DSS) and Management Information Systems (MIS)

All of the above give us unprecedented opportunities to build intelligent management systems located in the cloud. The solutions presented in this article were aimed at presenting the subjecy of programming such systems.

Bibliography

[1] Cloud Application, Computer World, Nr 14/1029, www.computerworld.pl (accesed: 25.06.2014)
[2] A CIO's Guide to Hybrid Cloud Performance, How Riverbed Accelerates Public and Private Clouds: A Framework for Hybrid Cloud Adoption, Riverbed White Paper, www.riverbed.com (accessed: 25.06.2014)
[3] Building a private cloud, Computerworld Guide, www.microsoft.pl/cloud (accessed: 25.06.2014)
[4] AUTOSAR(AUTomotive Open System ARchitecture) http://www.autosar.org/find02_07.php (accessed: 25.06.2014)
[5] FRYSKOWSKI B., GRZEJSZCZYK E.: Data transmission systems, Transport and Communication Publishers, 5a (2010)
[6] FRYŹLEWICZ Z., NIKOŃCZUK D.: Windows Azure. Introduction to cloud programming. Helion (2012)
[7] GRZEJSZCZYK E.: Teleservice communication with motor vehicle, Electrical Review, (2011)
[8] GRZEJSZCZYK E.: Analysis of selected telematic services of the BMW service, Electrical Review (2012)
[9] GRZEJSZCZYK E.: Selected issues of wireless communication over GSM with cars network computer system, GSTF Journal of Engineering Technology, vol.2, No. 2 (2013)
[10] GRZEJSZCZYK E.: Selected issues of the Bluetooth technology in control on-line and telemetry, International Conference on Advanced ICT and Education, Advances in Intelligent Systems Research, vol. 33 Hainan, China (2013)
[11] GRZEJSZCZYK E.: The control on-line over TCP/IP exemplified by communication with automotive network, Federated Conference on Computer Science and Information Systems (FedCSIS) , IEEE Xplore, vol.1, Cracow, (2013)
[12] GRZEJSZCZYK E.: Vehicle Stability Control (VSC) and Supervision based on CAN network. Part II, Journalof Engineering Technology, vol.2 No 4 (2014)
[13] GRZEJSZCZYK E.: Distributed database systems as a source of scientific/utility information in Interactive Intelligence systems, International Scientific Conference: Electronic publications in the development of science. Białystok (2014).
[14] MERKISZ J. MAZUREK S.: On-board diagnostic systems in motor vehicles OBD, Transport and Communication Publishers, Warsaw (2006)
[15] OSE/VDX http://www.osek-vdx.org (accessed: 25.06.2014)
[16] Microsoft Azure Platform, Computerworld Guide, www.microsoft.com/poland
[17] MATEOS A., ROSENBERG J.: The Cloud at Your Service, Helion (2011)
[18] ZIMMERMANN W. SCHMIDGALL R.: Data bus in vehicles, Transport and Communication Publishers, Warsaw (2008)

[16] This logic stores "its" data in a separate database not connected with the service data model

[17] FLOPS - Floating Point Operations Per Second - the number of floating point operations executed per second