Decentralized Field Service Automation Using Adeptik Platform

Dmitry Spanderashvili, Alexander Morozov, Adelya Mamleeva
Astrakhan State Technical University
Astrakhan, Russia
spanderashvili@gmail.com, {morozov, mamleeva}@ilabsltd.com

Sergey Gudyrin
Adeptik Company
Astrakhan, Russia
gudyrin@adeptik.com

Denis Tarasov
OOO "Gazprom dobycha Astrakhan"
Astrakhan, Russia
tarasovdear@gmail.com

Abstract—**Most systems of field service automation are built on the client-server technology. In such systems, the majority of computing tasks is solved on a server side. However, there are challenges that need to be solved quickly, even in the absence of connection with the server. In this article we describe a method for organizing a decentralized field service automation using the platform being developed by Adeptik Company.**

I. Introduction

Field Service Management is an important tool for every service provider conducting on-site installations. An efficient solution is required to support various areas including service order placement, scheduling, technical assignment depending on skills and availability, route optimization, equipment allocation, as well as service level agreements and reports management. The platform must be seamlessly integrated with a CRM or Order Management system. Only comprehensive tools which support end to end field service delivery processes can reduce operating costs and improve customer experience of every successful service provider.

At the moment, a lot of companies offer their complex solutions for the automation of field service, e.g. ClickSoftware, SAP Mobile Platform, IBM WorkLight, Pega AMP, Motorola RhoMobile, Appcelerator Titanium, Kony Mobility Platform and others.

As all of mentioned solutions are built on client-server architecture, they require a constant connection to the server, as the server needs to communicate with all mobile devices.

This feature gives rise to the following problems:

- High requirements for computing performance in your office or cloud. Computing performance of mobile devices is used inefficiently.
- Reliability determines the availability of the office part of the solution. To ensure a high degree of reliability backup data channels, as well as backup server infrastructure are needed.
- High costs of scaling solutions. In case of increase of mobile devices, it is necessary to purchase a new server hardware and its settings.
- With a large number of mobile employees, the speed of information delivery decreases in proportion to the number of mobile devices used by employees. The limiting factor is the bandwidth of the channel, as well as changeable speed communication channels.

We propose a new approach, – decentralized field service automation, which is implemented through a platform called Adeptik.

II. Adeptik Platform Structure

Among the platforms for the development of multi-agent systems can be distinguished JASA platform, JADE, AnyLogic, JaCoMo and others. However, most existing platforms of construction of multi-agent computing systems are research-oriented and poorly applied under enterprise systems.

To avoid above mentioned limitation of existing approaches, we have identified the following requirements for the platform:

- The platform is to be multilayer to provide more flexibility and operability for different application, which is to be used in field service automation;
- Layers of platform is to be independent from each other to provide extensibility of functions realized at each layer;
- In the platform the universality of interlayer interconnection is to be realized. It is to provide encapsulation of layer logic.
- With a large number of mobile employees, the speed of information delivery decreases. The limiting factor is the bandwidth of the channel, as well as changeable speed communication channels.

Observing these requirements, we suggested the structure of the platform, which you can see at Fig. 1.

Adeptik platform structure includes three layers:

- Application layer.

- Distribution layer.
- Network layer.

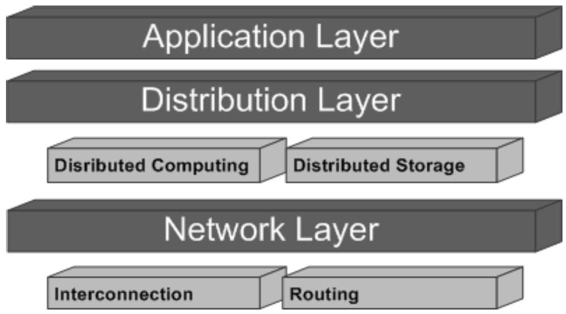

Fig. 1. Adeptik platform structure

Each layer has its tasks, but functioning together they form a unique functionality of decentralized field service automation with specific functions:

- Ability of functioning without permanent connection with the server.
- Performance increases with the usage of distributed computation.
- Reliability increases with the usage of distributed storage with replication and synchronization mechanisms.

A. Application layer

At this layer different types of field services automation tasks adapt for distributed computing.

To test the correctness of our assumptions we have considered the most general task for all types of field services: The Field Service Scheduling (FSS) Problem. The Field Service Scheduling problem has been the prime focus of many studies. Its main attributes are described quite comprehensively in [1] that discuss the problem as a combinatorial NP-hard optimization one. The problem schedules technicians (resources) to serve a set of tasks (demands), scattered at different locations.

Much of the data required to solve the problem centrally is usually unavailable and a centralized solution to the problem can't be obtained, because the necessary data is located only on the mobile devices. Furthermore, a solution of this sort has little value, since different business entities involved in the decision making do not have to follow it. Moreover, a decentralized solution method, which considers different interests of the business entities and provide ability of usage advantages of distribution layer, must be applied.

There are two approaches of solving decentralized problem of FSS: solving centralized problems using distributed algorithms and solving decentralized problems.

1) Solving centralized problems using distributed algorithms

R. Kohout and K. Erol [2] present an agent-based method developed to cope with a dynamic Pickup and Delivery problem with Time Windows (PDPTW). They developed a computationally efficient algorithm that identifies a solution to the problem and then improves it by a stochastic mechanism. The use of agents allows decomposing the problem into several sub-problems. Each sub-problem can then be solved independently and simultaneously using parallel computers.

2) Solving decentralized problems

S. J. Rassenti, V.L. Smith and L. Bulfin [3] present a version of a combinatorial auction (more on this – to follow) applied in order to find the allocation of various airports' time slots to competing airlines so as to maximize the sum of values of slots allocated among all airlines. A time slot can be a slot for departure or a slot for arrival. Since values for these slots are generally correlated, grouping of slots into packages allows bidding on them by a combinatorial auction and allocating them to the airlines.

3) Decetralization in Adeptik platform

The modular structure of the platform allows the use of any of the described methods of decentralization, depending on the context of the problem being solved.

B. Distribution layer

1) Distributed computing

Distributed computing layer includes:

- Implementation of the platform for the organization of multi-agent computing [4].
- Usage of mobile cloud computing features [5].
- Usage of mobile grid computing features [6].
- Tasks optimization for distributed processing.

2) Distributed storage

This layer includes combination of mechanisms allows a user to make the most of what a non-uniform network has to offer in terms of gaining fast access to fresh data, without incurring the foreground penalty of keeping distributed elements on a weak network consistent. Distributed storage layer includes:

- Replication features, – sharing information so as to ensure consistency between redundant resources, such as software or hardware components, to improve reliability, fault-tolerance, or accessibility [7].
- Sharding, or horizontal scaling, divides the data set and distributes the data over multiple servers, or shards. Each shard is an independent database, and collectively, the shards make up a single logical database.
- File-sharing networks functionality on the basis of P2P interactions [8];
- multicast-like solution for locating data
- The functions of distributed databases with support of pre-aggregation mechanisms and distributed OLAP.

C. *Network layer*

This layer provides the ability of reliable connection even if cell network is not available. Nodes may connect to each other and form mobile ad-hoc network (MANET). MANET was described in many researches [9,10,11], but the uniqueness of approach that is used in Adeptik platform is that MANET is organized in heterogeneous network. For example, some nodes may be connected to a base station and other may be connected to each other via D2D technologies, but if one of nodes from the second group has cell connection, all nodes from this group will also have connection with all devices from the first group.

This advantage allows to use Adeptik platform in such cases as Functioning in incomplete coverage of cell network, presented at Fig. 2 and Functioning in Disasters, presented at Fig. 3.

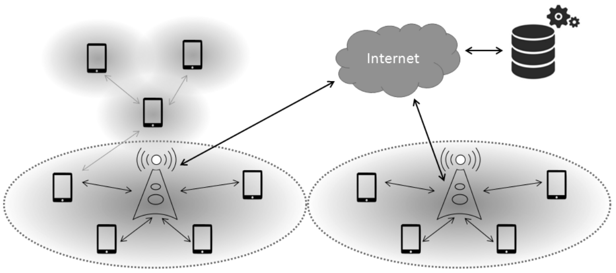

Fig. 2. Functioning in incomplete coverage of cell network

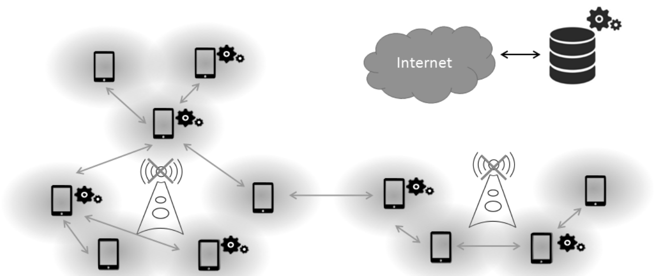

Fig. 3. Functioning in Disasters

1) *Interconnection*

At this layer the choice of optimal connection type with base station or with the nearest device via all possible technologies is carried out:

- Connection may be performed by cell network
- Connection may be performed by organizing pear to pear interconnection using any of D2D techniques.

2) *Routing*

At this layer the choice of the most suitable routing protocol is carried out.

III. Conclusion

As a result of modelling the process of problem setting and optimal routes search the following features have been revealed:

- The approach based on distributed computing showed better decisions at a fixed time.
- The time of algorithm execution for solving a fixed quality grew slower at the problem dimension increase.

The distributed approach can use the computing performance of mobile devices and effectively solve the problem of field staff scheduling.

According to the results of the simulation the proposed approach allows the following:

- higher-speed delivery of information to devices (multicast),
- provides remedy peak loads on the server channel,
- increases the reliability of the system (estimated using Markov models).

Acknowledgment

This research was supported by Foundation for Development of the Center for Elaboration and Commercialization of New Technologies.

We thank our colleagues from Astrakhan State Technical University who provided insight and expertise that greatly assisted the research.

We would also thank Arash Asadi from IMDEA Networks Institute, Kaigui Bian from Peking University, Li Liu from Shandong Jiaotong University, Thouraya Toukabri Gunes from Orange labs for their scientific collaboration.

References

[1] M. Zerdin, A. Gibrekhterman, U. Zahavi and D. Yellin. "Optimization strategies for Restricted Candidate Lists in Field Service Scheduling", *Intelligent Computational Optimization in Engineering, SCI 366, Springer: Heidelberg*, 2011, pp.55-83.

[2] R. Kohout and K. Erol. "In-time agent-based vehicle routing with a stochastic improvement heuristic", *Eleventh Conference on Innovative Applications of Artificial Intelligence: Orlando*, 1999.

[3] S. J. Rassenti, V.L. Smith and L. Bulfin. "A combinatorial auction mechanism for airport time slot allocation" *Bell Journal of Economics*, 13, 1982, pp. 402-407.

[4] J. Pitt, P. Venkataram, A. Mamdani, "QoS management in MANETs using norm-governed agent societies", *Proc. 6th International Workshop Engineering societies in the agents' world, Kusadasi, Turkey*, 2005.

[5] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for vm-based cloudlets in mobile computing," *IEEE Pervasive Computing*, vol. 8, no. 4, 2009, pp. 14–23.

[6] I. Foster, C. Kesselman, J. M. Nick and S. Tuecke, "The physiology of the Grid, an open Grid services architecture for distributed systems integration" *Open Grid Service Infrastructure WG, Global Grid Forum*, 2002.

[7] J.A. Holliday, R. Steinke, D. Agrawal, and A.E. Abbadi. "Epidemic algorithms for replicated databases", *IEEE Transactions on Knowledge and Data Engineering*, 15(5), 2003, pp. 1218–1238.

[8] A. Klemm, C. Lindemann, and O. Waldhorst, "A special-purpose peer-to-peer file sharing system for mobile ad hoc networks", *IEEE vehicular technology conference*, vol. 4, Citeseer, 2003, pp. 2758–2763.

[9] M. Meisel, V. Pappas, and L. Zhang, "Ad hoc networking via named data", *fifth ACM international workshop on Mobility in the evolving internet architecture, New York, NY, USA: ACM*, 2010, pp. 3-8.

[10] A.K. Gupta, H. Sadawarti, A.K. Verma, "Review of various Routing Protocols for MANETs", *International Journal of Information & Electronics Engineering (IJIEE)*, No. 40, 1(3), Nov. 2011, pp. 251-259.

[11] A.K. Gupta, H. Sadawarti, A.K. Verma, "Performance Analysis of MANET Routing Protocols in Different Mobility Models", *International Journal of Information Technology and Computer Science (IJITCS)*, 5(6), May 2013, pp. 73-82.