Performance Evaluation of Operations in RedSib with Substitution Mechanism

Andrey Vasilev, Ilya Paramonov
P. G. Demidov Yaroslavl State University
Yaroslavl, Russia
vamonster@gmail.com, ilya.paramonov@fruct.org

Abstract—**The upcoming emergence of Internet of Things networks and applications require approaches for creation of robust services. We proposed and implemented the substitution mechanism for open source application development platform Smart-M3. In this paper we evaluate the performance of semantic information broker, the core element of the platform. Firstly we make regression testing for plain operations of insertion and removal of triples, then we proceed on to measure mean agent substitution time and conclude this testing by combing previous tests. Results show that the was no performance penalty for implementation of agent substitution mechanism and proves it to be feasible approach for end-user targeted systems.**

Keywords*—*Performance, Smart spaces, Smart-M3, Agent substitution, Reliability.

The next generation of intelligent services will rely on information coming from the variety of sensors embedded into surrounding things [1]. This information will be processed by a series of services communicating with each other and the user in order to fulfill the ones needs in the most efficient way. Services may run on devices embedded in surrounding things, on mobile devices or on high-performance servers.

In order to implement the vision of services composed out of heterogeneous devices interconnected together application designers and middleware architects must solve a series technical challenges that among others include interoperability and fault tolerance [2]. The Smart-M3 platform ”— is an open source information sharing platform. The core element of the platform is the Semantic Information Broker (SIB) that manages a data storage and provides access to it for platform agents called Knowledge Processors (KP). The data is stored according to standards developed by the Sematic Web community making allowing to apply knowledge processing methods to it.

In our previous work [3] we proposed a dataflow network model to describe internet of things systems that perform multi-step data processing and described the agent substitution mechanism that allows to replace the unforeseen disconnected agent with the substitute one till the reconnection of the original agent. The substitute agents are programmable entities that are capable to execute an arbitrary data processing program, therefore they can process data just like the disconnected agent. The substitution mechanism also ensures the subscription notifications are not lost during the substitution procedures.

The substitution mechanism is designed and implemented as the part of the RedSib — the recent version of SIB [4]. This paper presents the setup and results of the performance evaluation of simple SIB operations (insert and remove), the agent substitution operation and the impact of the substitution operation on plain operations of SIB.

The performance evaluation setup for insert and remove operations replicates the setup that was used to test plain RedSib version without the substitution mechanism module [4]. The main goal of the assessment is to measure the impact of the subscription operation on the mean execution time of insert and remove SSAP operations [5]. The evaluation consists of 1000 tests performed in the following way. At the beginning all SIB components and testing KP are started, then KPs performed some actions according to the defined scenario and record mean time of the operation execution. In the end all elements are shut down in order to prevent the effect of previously executed operations.

The setup includes two KPs that execute scenario: worker KP and subscriber KP. UML sequence diagram describing the interaction of processes during scenario is shown on Fig. 1. After connection with the SIB the subscriber KP setups several subscriptions. The subscription count coincides with the test number that is being performed. When all subscriptions are set up the worker KP performs 50 insert and remove operations using randomized triples that does not trigger initiated subscriptions. The worker KP records time before sending operation request and after receiving the notification form SIB. These timestamps then are used to calculate mean operation execution time for this test.

The diagram showing derived results as the correlation between the number of subscriptions and the mean operation time is shows on Fig. 2. The minimum value for the mean operation time is one millisecond, the maximum — three and a half milliseconds. This results coincide with the results received for the previous version of RedSib shown in [4]. It should be noted graph shown non-linear behaviour for subscription count between 1 and 250, which could be interpreted either by timer resolution issues, or by some optimizations in internal data structures for big number of subscriptions.

The performance evaluation setup for the main agent substitution operation is based on the same idea of performing several operations simultaneously and measuring the corresponding mean time. The evaluation consists of 35 tests during which the number of primary agents perform the substitution operation at the same time. These tests also run in isolation from each other preventing effect of one test on another. The

ISSN 2305-7254

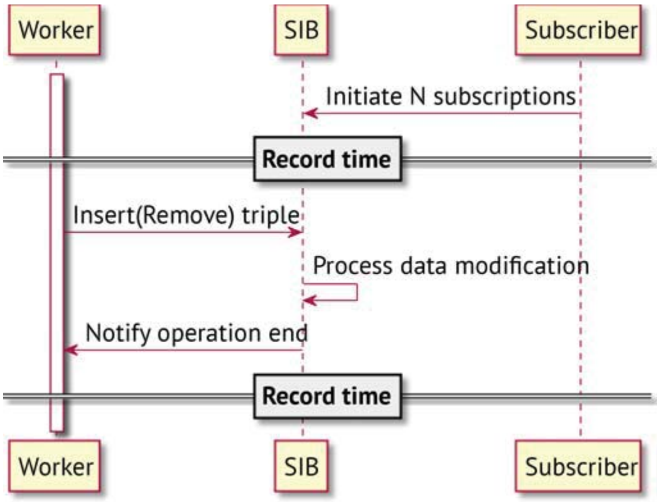

Fig. 1. Insert and remove operation time measurement setup

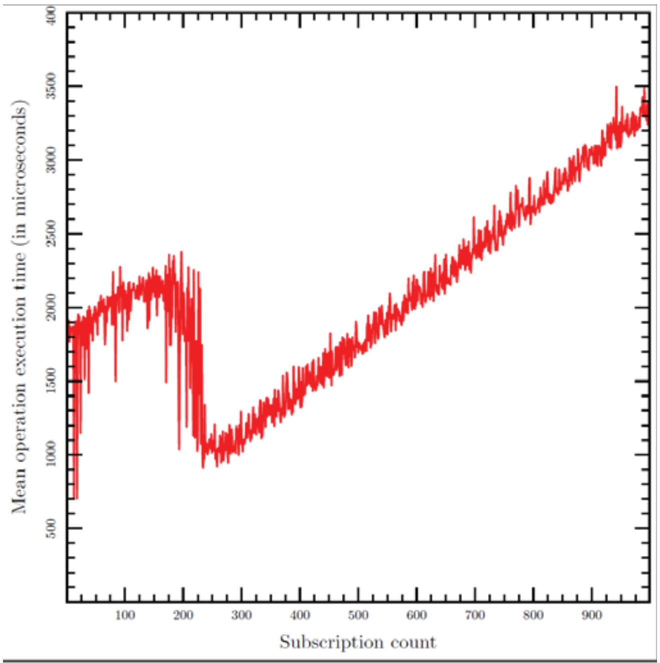

Fig. 2. Mean time of insert and remove operations

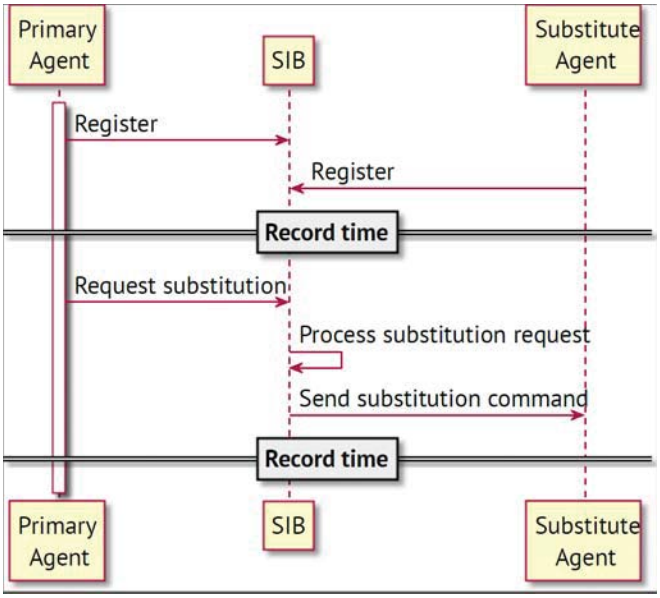

Fig. 3. Substitution operation time measurement setup

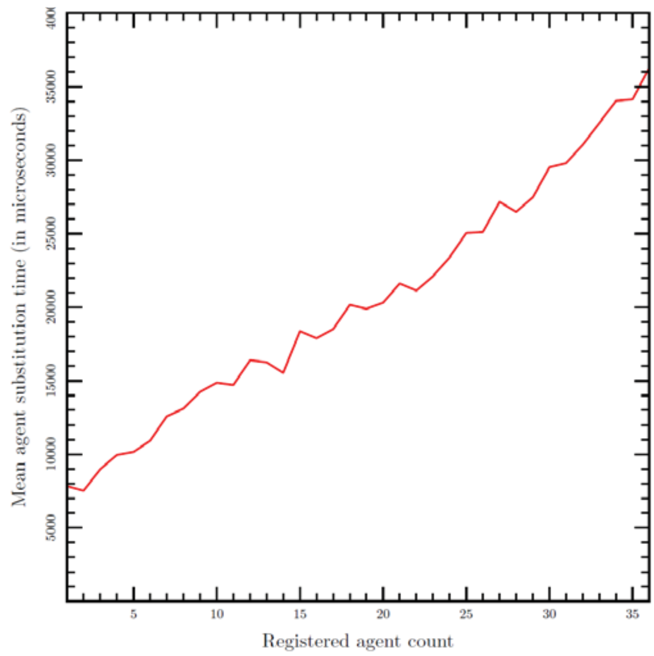

Fig. 4. Mean time of the agent substitution operation

UML sequence diagram showing the interaction of components during performance evaluation is shown on Fig. 3.

At the beginning primary agents and substitute agents of the dataflow network connect to the SIB and register in the substitution mechanism. The number of primary-substitute agent pairs is equal to the number of the test. When all agents are registered, primary agents request the voluntary substitution from the mechanism. They record the time before sending a request to the mechanism. Then the substitution mechanism processes request, selects substitution agent from the pool of free substitution agents and sends command to the selected agent. Upon receiving the command, substitution agent registers time. The captured interval is considered to be the substitution time.

The diagram showing derived results as the correlation between the number of registered primary agents and the mean substitution operation execution time is shown of Fig. 4. The mean time linearly increases from 6 milliseconds for one registered pair to 37 milliseconds for 35 registered pairs. On one hand this time can be considered appropriate for end user smart environments that include about 1000 of intelligent devices, on the other hand it should be improved in order to support large-scale installations.

Having received such results we considered the following problem: can the substitution operation impair the performance

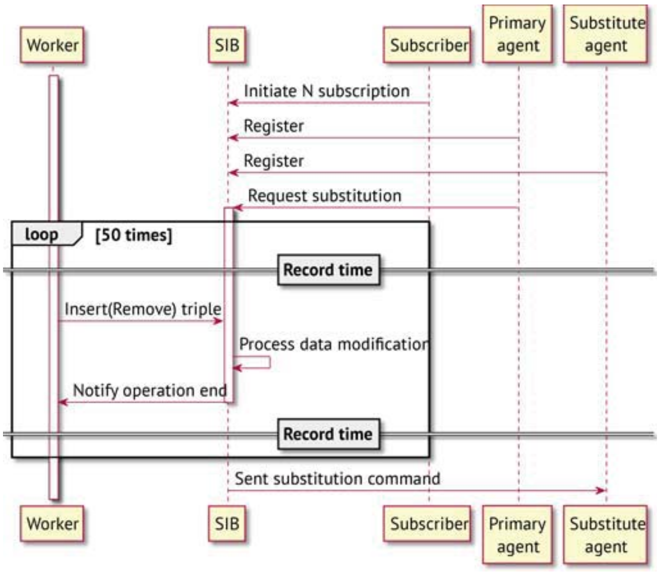

Fig. 5. Insert and remove operations time measurement setup

of simple SIB operations? Therefore another performance evaluation was designed and performed. The main idea behind this assessment is to check mean operation time for insert and remove operations during the substitution process, i.e. combine first two evaluations and measure mean time of simple operations. The evaluation consists out of 900 tests run, because the maximum number of subscriptions is limited to 1024 and some amount of subscriptions is used by dataflow agents participating in the test.

The UML sequence diagram describing the testing procedure is shown on Fig. 5. At the beginning 10 primary and 10 substitute agents register in the substitution mechanism, the subscriber KP setups subscriptions to the SIB by the number of the running test. Selected number of primary and substitute agents is sufficient enough to see the impact of operation on each other. When all agents performed initialization, then all agent are notified to start the testing scenario. Primary agents request the substitution form the substitution mechanism and worker KP begins to insert and remove triples from the smart space. Timestamps are being recorded before sending request to the SIB for data modification and after receiving the notification from the SIB.

The diagram showing derived results as the correlation between the number of subscription and the mean time of simple operation execution is shown on Fig. 6. Minimum and maximum values of the mean operation execution time did not change, that shows minimum impact of the substitution operation on insert and remove operations. The root for such behaviour is that the substitution mechanism is running as in the separate execution thread apart from the main operation of the SIB. Also, it should be noted that shown graph is shifted to the left compared to the graph shown on Fig. 2, because a set of subscriptions is used by the participating primary and substitute dataflow agents.

The implementation of the substitution mechanism will continue to evolve in direction of supporting all corner cases and improvement of algorithms efficiency.

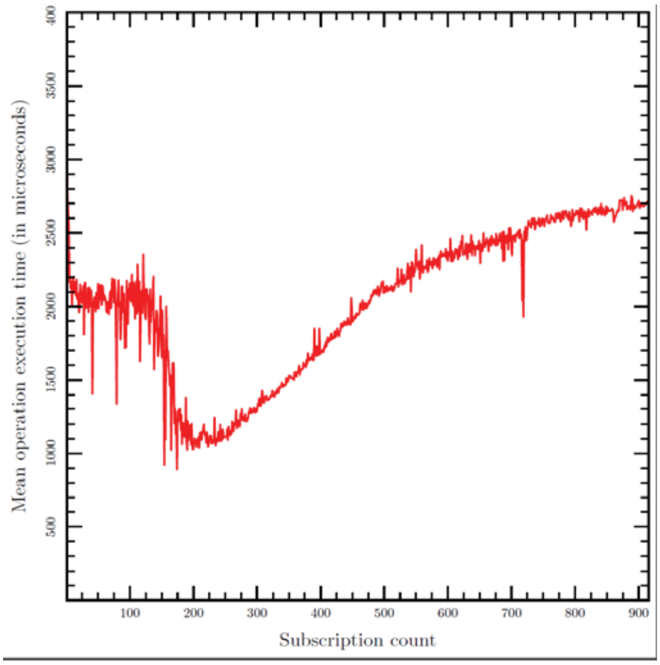

Fig. 6. Mean time of insert and remove operations

After conducting this research it could be proposed the research and development for the common solution, that could enforce the robustness of the substitution operation in the Smart-M3 platform. It could be beneficial to use advanced messaging queue protocols providing robust data delivery in case of unforeseen communication interruption that support heterogeneous networks.

REFERENCES

[1] Y.-K. Chen, "Challenges and opportunities of internet of things," in *17th Asia and South Pacific Design Automation Conference (ASP-DAC)*, February 2012, pp. 383–388.

[2] F. Mattern and C. Floerkemeier, "From the internet of computers to the internet of things," in *From active data management to event-based systems and more*. Springer, 2010, pp. 242–259.

[3] I. Paramonov, A. Vasilev, D. Laure, and I. Timofeev, "Agent substitution mechanism for dataflow networks: Case study and implementation in smart-m3," in *Internet of Things, Smart Spaces, and Next Generation Networking*, ser. Lecture Notes in Computer Science, S. Balandin, S. Andreev, and Y. Koucheryavy, Eds. Springer Berlin Heidelberg, 2013, vol. 8121, pp. 60–71.

[4] F. Morandi, L. Roffia, A. DElia, F. Vergari, and T. S. Cinotti, "RedSib: a Smart-M3 semantic information broker implementation," in *Proceedings of the 12th Conference of Open Innovations Association FRUCT and Seminar on e-Tourism. Oulu, Finland*. St.-Petersburg: SUAI, November 2012, pp. 86–98.

[5] J. Honkola, H. Laine, R. Brown, and O. Tyrkkö, "Smart-M3 information sharing platform," in *IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 2010, pp. 1041–1046.