**Collaborative Discussion 2 Summary:**

**Agent Communication Languages**

**by Vasilisa Lukashevich**

**(Intelligent Agents May 2023)**

We have had a rather extensive discussion about the ways of communication in agent systems, contrasting various possible languages and approaches. Knowledge Query and Manipulation Language (KQML) was a highly popular agent communication language about 20-30 years ago, which is probably excellent for familiarising oneself with the topic. Nowadays, another option called FIPA-ACL seems to be more widely adopted (Gan et al., 2019).

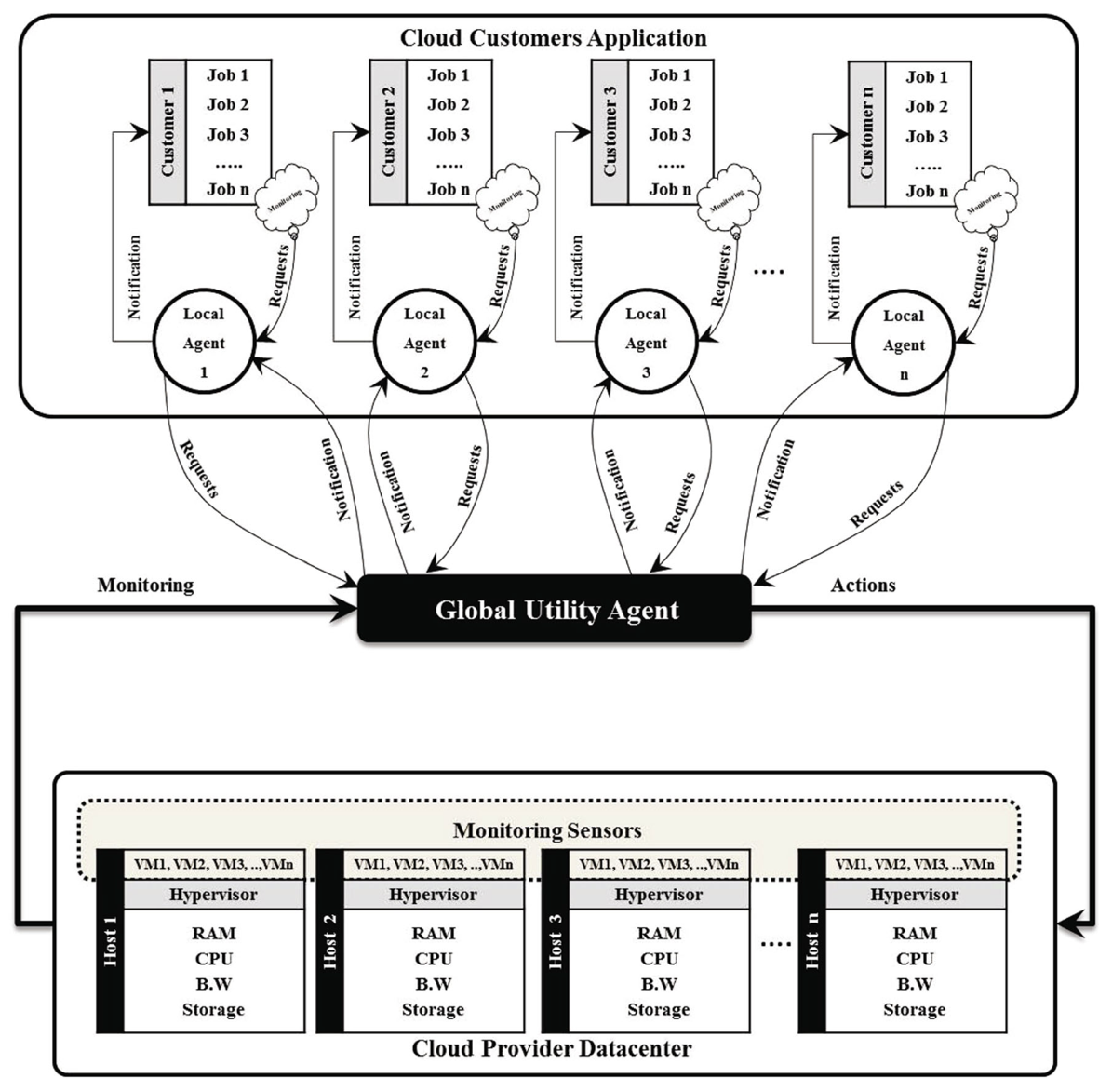
Toor (2023) proposed object-oriented programming (OOP) methods such as Java or Python to bridge the gap between ACLs and other programming languages. I believe this is an obvious solution because OOP is a programming paradigm that utilises the concepts of Classes and Objects, which are also employed in application programming interfaces (APIs). APIs, in turn, have already filled this communication gap in complex systems. For instance, thanks to APIs, we are able to engage in online shopping on Amazon.

Feaviour (2023) noted that Amazon is more of a distributed system than a multi-agent system, which may not necessarily require the features that warrant the use of an ACL. However, I believe that if the service conducts real-time data analysis and provides personalised product offerings based on the consumer's previous preferences, or even if the web-shopping is connected to Alexa, the entire process can be considered as a multi-agent system.

Actually, Amazon uses a multi-agent system to manage the cloud provider's resources while taking into account the customers' quality of service requirements. The architecture is depicted in the picture below (Al-Ayyoub et al., 2015).

Castro (2023) highlighted that although communication between intelligent agents typically involves transactional requests and responses through APIs, both the client and server must have prior knowledge of the payload used for communication. However, in certain scenarios, intelligent agents may be unaware of the existence of other agents. Nevertheless, they should possess the ability to communicate effectively.

Adeniyi (2023) added that comparing method invocation within the same program, ACLs offer numerous advantages. However, it is important to acknowledge that distributed systems utilising the object-oriented paradigm can overcome these limitations by utilising frameworks, implementing APIs for cross-language communication (such as JSON), or employing asynchronous communication through queues.



Dudkin (2023) noted that although multi-agent systems face similar challenges as any distributed system, they can also leverage solutions commonly employed in modern commercial systems. One such solution, proposed by researchers, involves incorporating a Broker that manages the transport layer of communication between agents, utilising a message queue manager like RabbitMQ (Amrani et al., 2020). Additionally, ontology alignment can be facilitated by employing a neural network to assist agents in interpreting communication messages.

In conclusion, on the one hand, ACL communication offers a declarative language to describe a specific state, defining the types and semantics of messages that agents are able to exchange (Padovitz et al., 2004).

On the other hand, the main disadvantage of ACLs like KQML is their complexity. They can be difficult to learn, especially for developers who are not familiar with the underlying concepts of agent-based systems.

I agree that it's difficult to differentiate between multi-agent systems and distributed systems because sometimes they can be the same (Feaviour, 2023). I feel there is a gap in the literature for a clear three-column comparison of distributed systems, single-agent, and multi-agent systems.

After the discussion I feel that I need a special table with two columns to compare distributed systems and multi-agent systems. Or maybe compare them in another way. For instance, according to Darrow (2023), multi-agent systems generally combine Artificial Intelligence, Intelligent Agents, and Game Theory (which is utilised to enable agents to negotiate with each other).

**References**

Al-Ayyoub, M., Jararweh, Y., Daraghmeh, M. & Althebyan, Q. (2015) Multi-agent based dynamic resource provisioning and monitoring for cloud computing systems infrastructure. *Cluster Computing, 18*, pp.919-932.

Amrani, N.E.A., Youssfi, M., Bouattane, O. & Abra, O.E.K. (2020) Interoperability between heterogeneous multi-agent systems recommended by fipa: Towards a weakly coupled approach based on a network of recurrent neurons of the lstm type. In *2020 3rd International Conference on Advanced Communication Technologies and Networking (CommNet) (pp. 1-6). IEEE.*

Castro, A., Feaviour, L., Dudkin, M., Rizzo, A., Lukashevich V. & Adeniyi, S. (2023) Initial post [Collaborative Learning Discussion 2]. Intelligent Agents May 2023. University of Essex Online. Available from: https://www.my-course.co.uk/mod/forum/discuss.php?d=163651 [Accessed 9 Jul 2023].

Darrow, R.M. (2023) The future of AI is the market. In Artificial Intelligence and Machine Learning in the Travel Industry: Simplifying Complex Decision Making (pp. 177-182). Cham: Springer Nature Switzerland.

Finin, T. & Labrou, Y. (2002) Theory and Practice of Agent Communication Languages (Presentation). Available from: https://redirect.cs.umbc.edu/~finin/talks/691m.pdf [Accessed 16 Jun 2023]

Gan, K. S., Anthony, P., Chin, K. O. & Hamdan, A. R. (2019) 'Enforcing Social Semantic in FIPA-ACL Using SPIN', Agents and Multi-agent Systems: *Technologies and Applications*. St. Julians, Malta. Manhattan: Springerlink.

Lukashevich, V., Feaviour, L. & Toor, A. (2023) Initial post: KQML for dummies. [Collaborative Learning Discussion 2]. Intelligent Agents May 2023. University of Essex Online. Available from: https://www.my-course.co.uk/mod/forum/discuss.php?d=163792 [Accessed 9 Jul 2023].

Padovitz, A., Zaslavsky, A., Loke, S.W. & Tosic, M. (2004) Agent communication using publish-subscribe genre: architecture, mobility, scalability and applications. *Annals of Mathematics, Computing and Teleinformatics, 1(*2), pp.35-50.