Collaborative Discussion: Agent Communication Languages

by Maria Ingold

Initial Post

Knowledge Query and Manipulation Language (KQML) is used to enable the sharing of knowledge between intelligent agents through a common language and set of protocols (Finin et al., 1994). KQML enables the exchange of synchronous as well as asynchronous communication. Communication is done via performatives, like action verbs, that can be used one-to-one, such as *ask* or *tell*, or broadcast one-to-many, such as *advertise* or *subscribe*. The message is a form of *speech act*, described by Searle (1969) as the “basic unit of communication”. Interpreting the message exchanged requires a shared ontology, or shared understanding, of the applicable domain (Fensel et al., 2000). While ontologies ensure understanding and standardisation, they add complexity and maintenance.

Technology has evolved since Finin et al. (1994). Other, more recent, agent communication languages (ACL), like Foundation for Intelligent Physical Agents’ (FIPA) FIPA-ACL, also enable distributed systems, known as Multi-Agent Systems (MAS) (Poslad, 2007). In contrast to the Semantic Web, FIPA enables association of application-specific ontologies. Java Agent Development Framework (JADE) and Smart Python multi-Agent Development Environment (SPADE) are FIPA-compliant distributed middleware (Donancio et al., 2019). JADE can use Java Remote Method Invocation (RMI) to communicate between containers.

Method invocation is the way object-oriented programming languages activate a procedure, whereas an agent has an internal autonomy that allows it to not only be reactive to method invocations but proactive (Briot, 2024). This means while object-oriented programming languages use external messages to invoke behaviour, agents are interactive and can use messages to support internal method invocation (Odell, 2002). Method invocations tend to have a fixed number of parameters, while ACL, albeit formal, allows format and content variations. Furthermore, agents may require longer-term interaction than just method invocation.

References:

Briot, J. P. (2024) From Procedures, Objects, Actors, Components, Services, to Agents 125–146. DOI: https://doi.org/10.1007/978-3-031-34518-0.

Donancio, H., Casals, A. & Brandão, A. (2019) Exposing agents as web services: a case study using JADE and SPADE. Available from: https://www.semanticscholar.org/paper/Exposing-agents-as-web-services%3A-a-case-study-using-Donancio-Casals/ffeff5b9c2e60866ac1d111101e8497fddbecc49 [Accessed 7 January 2024].

Fensel, D., van Harmelen, F., Klein, M. & Akkermans, H. (2000) On-To-Knowledge: Ontology-based Tools for Knowledge Management. Available from: http://www.ontoknowledge.org;contact:dieter@cs.vu.nl [Accessed 4 December 2023].

Finin, T., Fritzson, R., McEntire, R. (1994) KQML as an agent communication language 456–463. DOI: https://doi.org/10.1145/191246.191322.

Odell, J. (2002) Objects and Agents Compared, *Journal of Object Technology* 1(1): 41–53. Available from: http://www.jot.fm/issues/issue_2002_05/column4 [Accessed 7 January 2024].

Poslad, S. (2007) Specifying protocols for multi-agent systems interaction, *ACM Transactions on Autonomous and Adaptive Systems (TAAS)* 2(4). DOI: https://doi.org/10.1145/1293731.1293735.

Searle, J.R. (1969) Speech Acts: An Essay in the Philosophy of Language. DOI: https://doi.org/10.1017/CBO9781139173438.