# Eve: a Novel Open-source Web-based Agent Platform[1]

Jos de Jong [a] Ludo Stellingwerff [b] Giovanni E. Pazienza [a]

[a] *Westerstraat 50, 3016 DJ, Rotterdam, The Netherlands*

**Abstract**

The existing approaches to build multi-agent systems fail at addressing the challenges posed by the current technology, where ubiquitous interconnected electronic devices are no more passive machines operated *by* humans but rather active computational components cooperating *with* humans. In this paper, we present a novel open-source web-based agent platform called 'Eve' that features some specific characteristics (eg, platform and language independence, openness) that make it particularly suitable to be applied to real-life applications.

## 1 Introduction

The proliferation of smart devices endowed with an extensive variety of sensors and a significant computing power has opened up countless new visionary applications. Being proactive is no longer an exclusive prerogative of human beings: both humans and software applications can be considered as entities with some degree of autonomy that interact with each other and with the environment though without need of centralized coordination. In this framework, devices are modeled as *agents* that mimic some of the characteristics of human beings: they are autonomous (ie, capable of taking decisions), intelligent (ie, capable of adapt their behaviour on the basis of available data), and social (ie, capable of communicating with humans as well as other agents). Therefore, the fundamental tool to handle (and profit from) such a complex yet promising scenario is an effective multi-agent system (MAS) [2] tailored for this novel technological context. Even though there are numerous implementations of MAS, the great majority – usually Java based – consist of a closed and controlled environment (operating system or simulation environment) where agents can live and interact with each other. Here, we propose 'Eve' that is a multi-agent system specifically thought to be deployed on a diverse distributed environment. Eve is a web-based agent platform featuring a number of characteristics that differentiate it from other widely-employed agent platforms. A thorough comparison is therefore hard to make, since the core concepts of Eve are different from those of traditional agent platforms.

## 2 A glimpse on the architecture of Eve

Figure 1 presents an overview of Eve architecture from which it is possible to catch a glimpse of its main features. First of all, Eve is *platform independent* since agents can live on any device: smartphones, robots, servers or, more generically, in the cloud. Second, Eve is *language independent*[2] because it dictates only the communication protocol (JSON-RPC) which works over existing transport layers (HTTP, XMPP). Third, Eve is an *open agent platform*: each agent has its own public urls and hence existing Eve-systems can be easily connected to the others. Furthermore, non-Eve systems can be connected to the Eve platform by making the API of the system available via an Eve agent acting as a wrapper.
Eve is a fully decentralized system: there is neither central coordination nor centrally-stored list of all

[1] The full paper has been published in *Proceedings of 2013 IEEE International Conference on Systems, Man, and Cybernetics (IEEE SMC 2013)*.

[2] Currently, there are two mature implementations of Eve (one in Java and one in Javascript) whereas a third one (in Python) is in a embryonic stage.

available agents. Interactions among agents are asynchronous and request-driven: agents get to know each other via the so-called *shared services* (eg, acting in the same calendar or registering at the same locations service). The architecture of Eve leads to further advantages among which is worth to emphasize the following ones:

*Scalability*: Eve is fully web-based and hence, from a practical point of view, there is no upper bound to the number of new agents that can be added to the system without degrading its performances.

*Robustness*: the state persistency in Eve is offloaded from the agents to the environment (ie, the states of the agents are distributed), which makes Eve insensitive to server/device failures. Although the network transport layer can contain single points of failure, Eve itself is a distributed agent platform, with no single point of failure by design.

*Massive parallelization of the workload of an agent*: agents can be multiplexed (ie, multiple instances of the same agent sharing the same state can exist at once). Traditional agent platforms have 1 thread per agent; in contrast, Eve allocates up to $n$ threads (where $n$ has virtually no upper bound) when the agent is heavily loaded and no thread at all when the agent is idle. This approach has the additional advantage of reducing resource consumption for idle agents.

*Seamless migration*: In Eve, there is no difference in accessing local or remote agents, as agents are fully location agnostic. This allows seamless migration of agents between run-time environments.

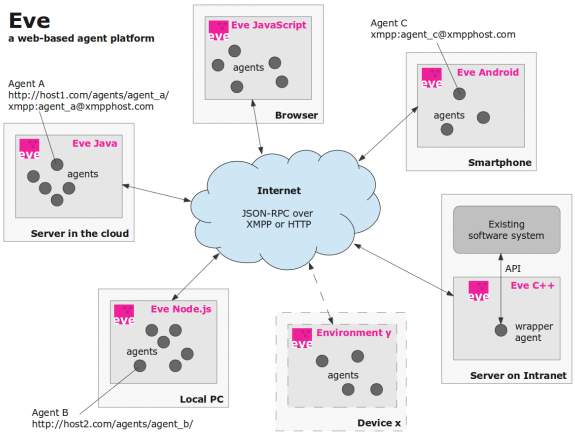


Figure 1: An architectural overview of the web-based agent platform Eve.

As already mentioned above, each Eve agent has its own URLs. Eve agents communicate via regular HTTP POST requests through the JSON-RPC protocol, which is a simple protocol using JSON (JavaScript Object Notation) to format requests and responses.

# References


[1] J. de Jong, L. Stellingwerff, and G. Pazienza, "Eve: a novel open-source web-based agent platform," *IEEE International Conference on Systems, Man, and Cybernetics (IEEE SMC 2013)*, 2013.

[2] N. Jennings, K. Sycara, and M. Wooldridge, "A roadmap of agent research and development," *Autonomous agents and multi-agent systems*, vol. 1, no. 1, pp. 7–38, 1998.