

# JADE/MAPS Gateway



Domenico Cimadoro, Giancarlo Fortino, Stefano  
Galzarano, Maria Ganzha, Mariusz Mesjasz,  
Marcin Paprzycki



# Motivation (1)

- System supporting a glider pilot
  - Health monitoring sensors (context – altitude)
    - Blood pressure
    - Temperature
    - Oxygen level
  - Battery level sensors (context – temperature)
- Care for elderly people
  - Calling emergency services (context – health monitoring)
- Intellignet home
  - Reacting on sensor input to start Roomba (context – party)

# Motivation (2)



- What is needed?
  - Sensors are small / focused / have minimal resources
  - Context awareness / “intelligent” operation requires “global view” / large(r) computational resources
- Needed → combination of sensors and “more powerful” resources
- Proposal
  - MAPS + JADE
- Here, we solve a specific technical problem for two selected agent systems



# MAPS

- MAPS – agent oriented, JAVA based framework for WSNs, which are based on the SunSpot technology
  - Component-based lightweight agent server architecture
  - Due to limited amount of resources on SunSpot devices, MAPS agents cannot be used to complete complex tasks
  - MAPS agents can read data from built-in and attached sensors
  - MAPS agents communicate via *MAPS Events*

# JADE

- Java Agent DEvelopment framework
  - One of the most popular Java-based agent platforms
  - Comply with the Foundation for Intelligent Physical Agents specification (*FIPA*)
- JADE agents communicate via ***ACLMessages***
  - comply with FIPA ACLMessage Structure Specification

# MAPS $\Leftrightarrow$ JADE



- For considered systems “MAPS” and “JADE” agents have to communicate
  - Both platforms Java-based
  - Both use different communication mechanisms
- Proposed solution
  - JADE/MAPS Gateway (the *gateway*) → to send (bidirectionally) messages between JADE agents and MAPS agents



# JADE/MAPS Gateway



- *Gateway* has to be on the JADE side of the bridge
- The *gateway* is composed of two parts
  - *JADE part* (a JADE agent) – JADE agents communicate with the *Gateway agent* as with a normal JADE agent
  - *MAPS part* (a *semi-functional MAPS Execution Engine*) - MAPS agents are **not aware** that the gateway (and the JADE agents) belong to a different platform.



# JADE/MAPS Gateway (translation mechanism)

- Since ACLMessages are more complex than MAPS Events, the communication from the JADE side to the MAPS side has to be simplified
- The gateway provides a special message structure (within the *GatewayOntology*) similar to the structure of MAPS Events
- The *GatewayAgent* fills out a MAPS Event based on the special content of an ACLMessage
- The translation from a MAPS Event to an ACLMessage is the reverse process.

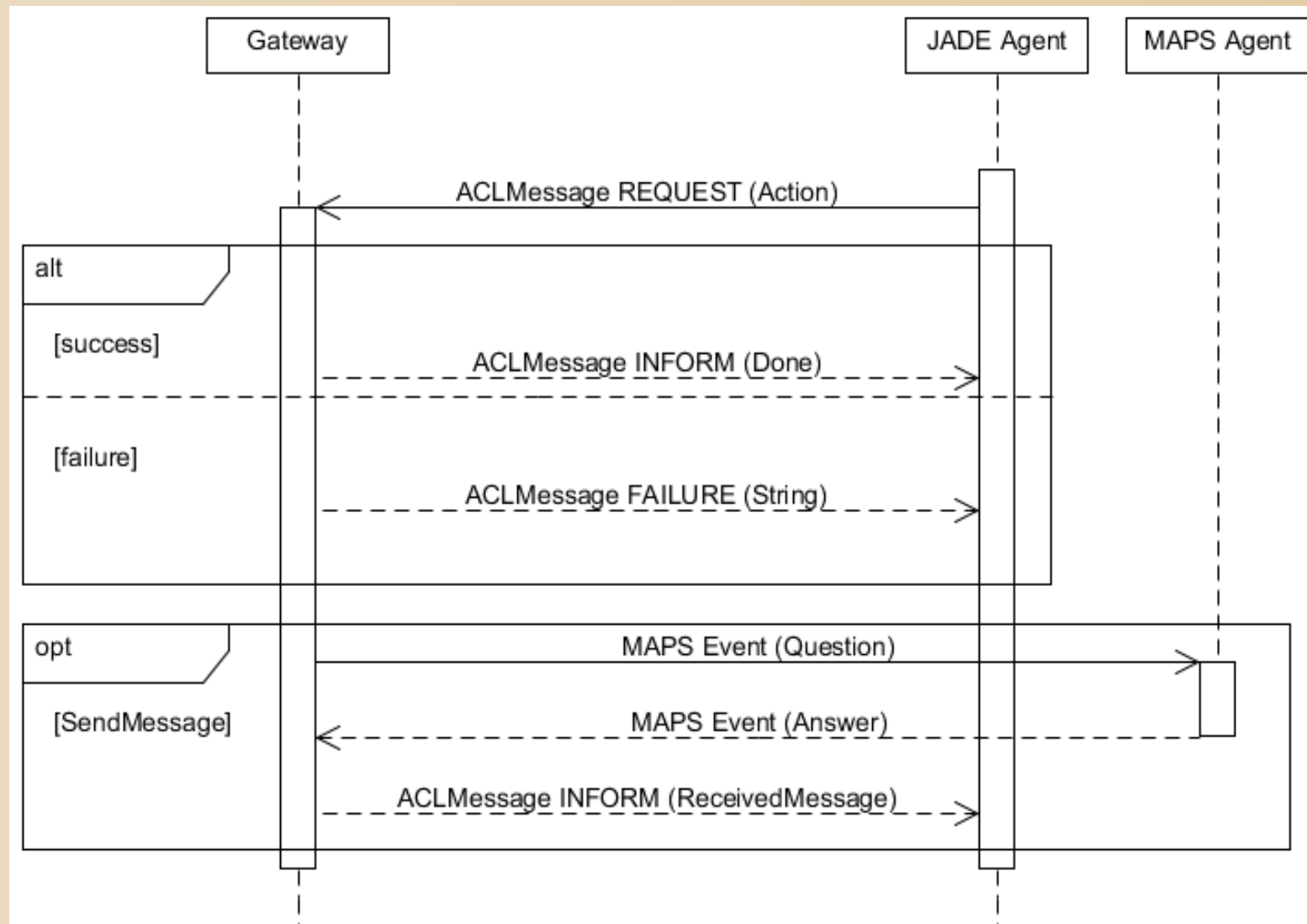


# JADE/MAPS Gateway

## (interaction with the *gateway*)


- For the JADE side, the *gateway* provides a special ontology (*GatewayOntology*), which describes actions that can be performed
  - *Register* – assign a MAPS ID to the specified AID and store a pair (AID, MAPS ID)
  - *Unregister*
  - *GetRemoteAgent* – receive list of known MAPS agents
  - *SendMessage* – translate an *ACLMessage* to a *MAPS Event*
- MAPS agents communicate with the *gateway* as with an *ExecutionEngine* (**nothing has changed**)

# JADE/MAPS Gateway (interaction with the *gateway*)

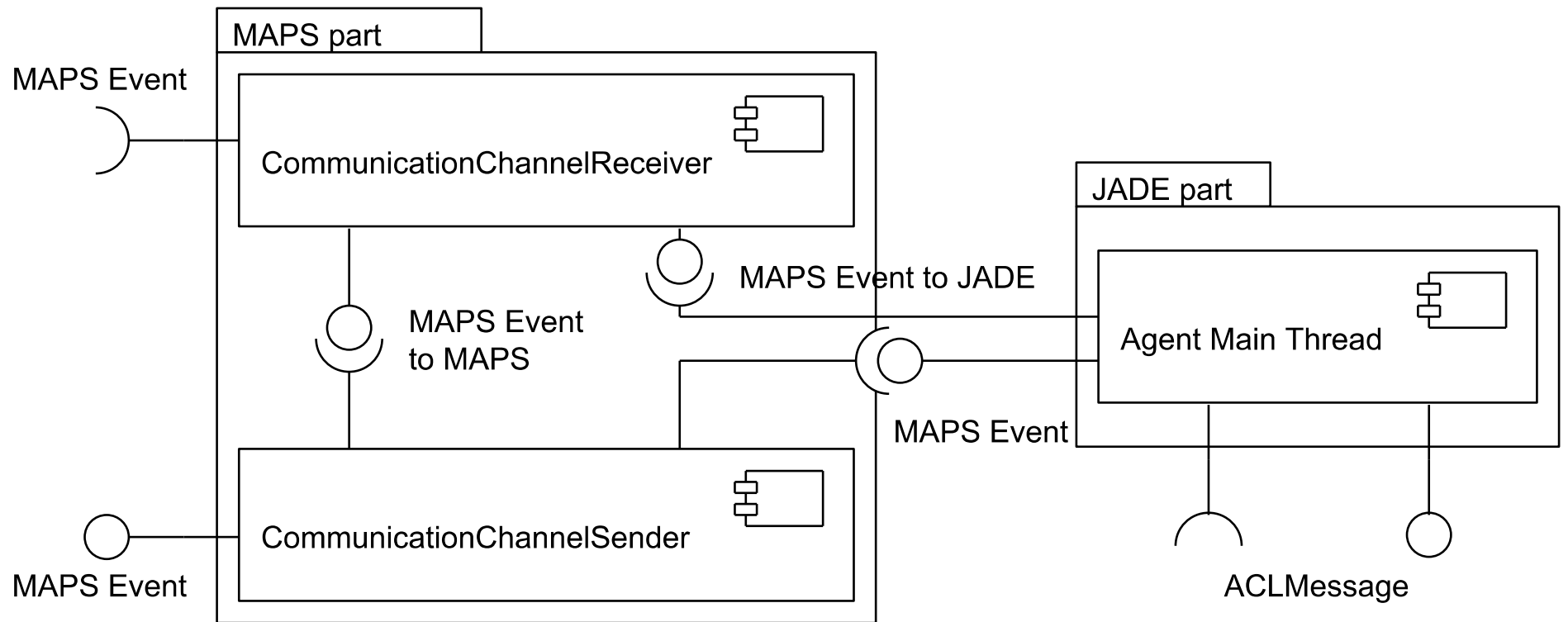


# JADE/MAPS Gateway



- Benefits : the gateway has all advantages of being a JADE agent (autonomously performs complex tasks) and this solution does not interfere with the simplicity of MAPS agents
  - Restrictions
    - JADE agents have to register themselves with the *gateway* (to be known to the MAPS agents)
    - The *gateway* has no capability of running actual MAPS agents
- 

# JADE/MAPS Gateway (implementation)

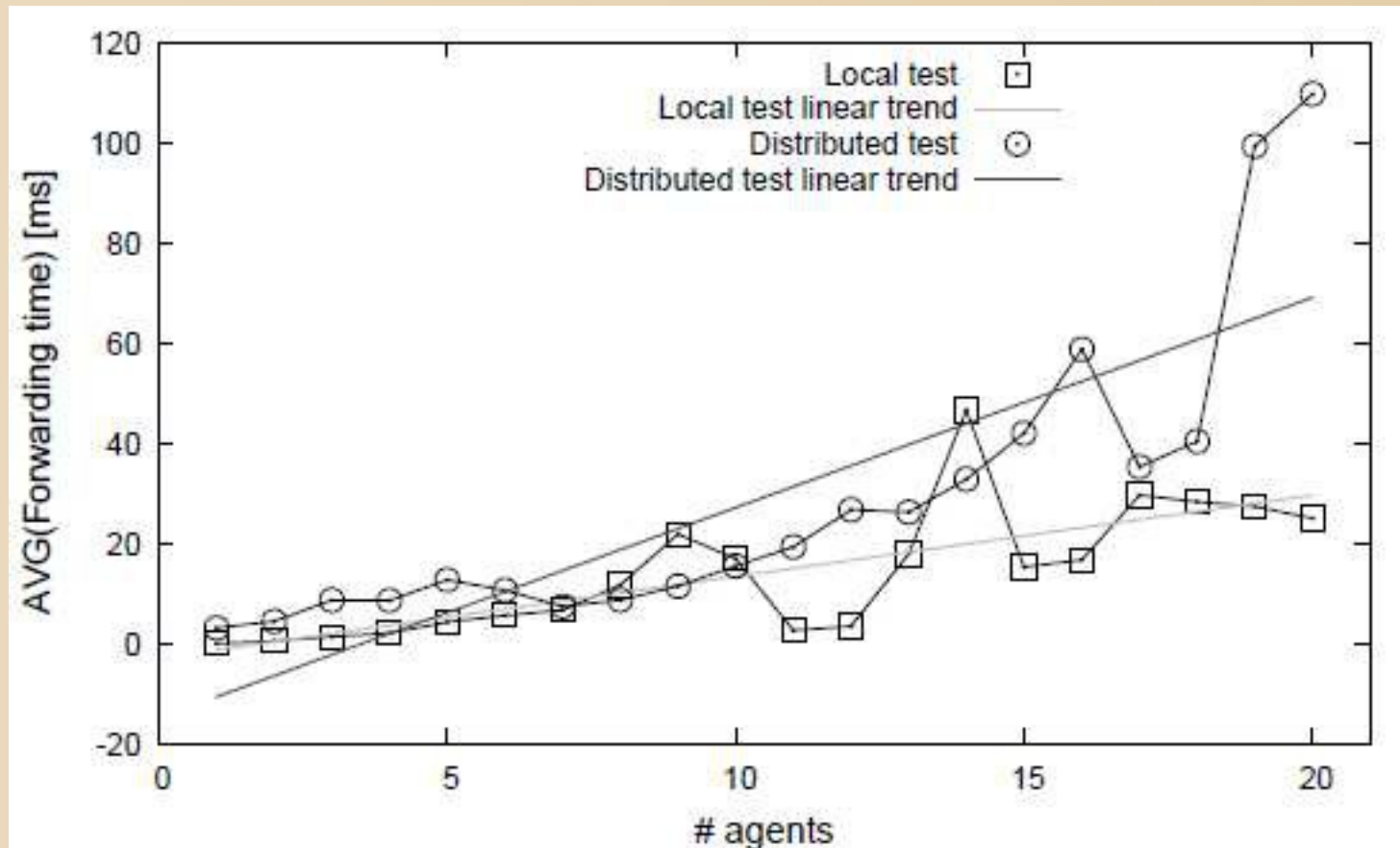


The component diagram of the *gateway*

# Test Scenario

- Performance test
  - Measures the average time needed by the *gateway* to forward a message from the JADE agent and to process a reply from the MAPS agent (“round-trip” communication)
  - Variable
    - Number of agent pairs composed of one JADE and one MAPS agent (from 1 to 20 pairs)
  - Each test took 5 minutes to complete and was repeated 20 times to obtain a good confidence measure

# Test Results



Forwarding time for multiple agents.

# Concluding remarks



- *Gateway* will be added to the JADE add-on library
  - If MAPS will be ported to the different sensor hardware, *gateway* will be naturally usable there
  - *Gateway* mechanism can be used also for connecting JADE agents to other agent systems
    - Translation is localized
    - Encapsulation of details for both sides
- 