

SO2MAS



A new Multi-Agent System platform using AMQP estandar

Índice

1. Introduction of AMQP
2. Diagrams
3. Communication
4. Queues
5. Benchmarks
6. Protocols (Joan)
7. Conclusion

AMQP

(Advanced Message Queuing Protocol)

- Standard Messaging.
- Uses a highly efficient binary protocol, unlike other previous approaches working with XML.
- Enables complete interoperability for messaging middleware
- Offers:
 - Point-to-point
 - Publish/subscribe
 - Many-to-many messaging.

How AMQP works?

AMQP defines three main types of components, connected into processing chains in a server to create the required functionality:

- The “**exchange**” component receives messages from applications and routes these to message queues, based on various criteria, usually message properties or content.
- The “**message queue**” component stores messages until a consumer client application can process them.
- The “**binding**” component defines the relationship between a message queue and an exchange and provides the message routing rules.

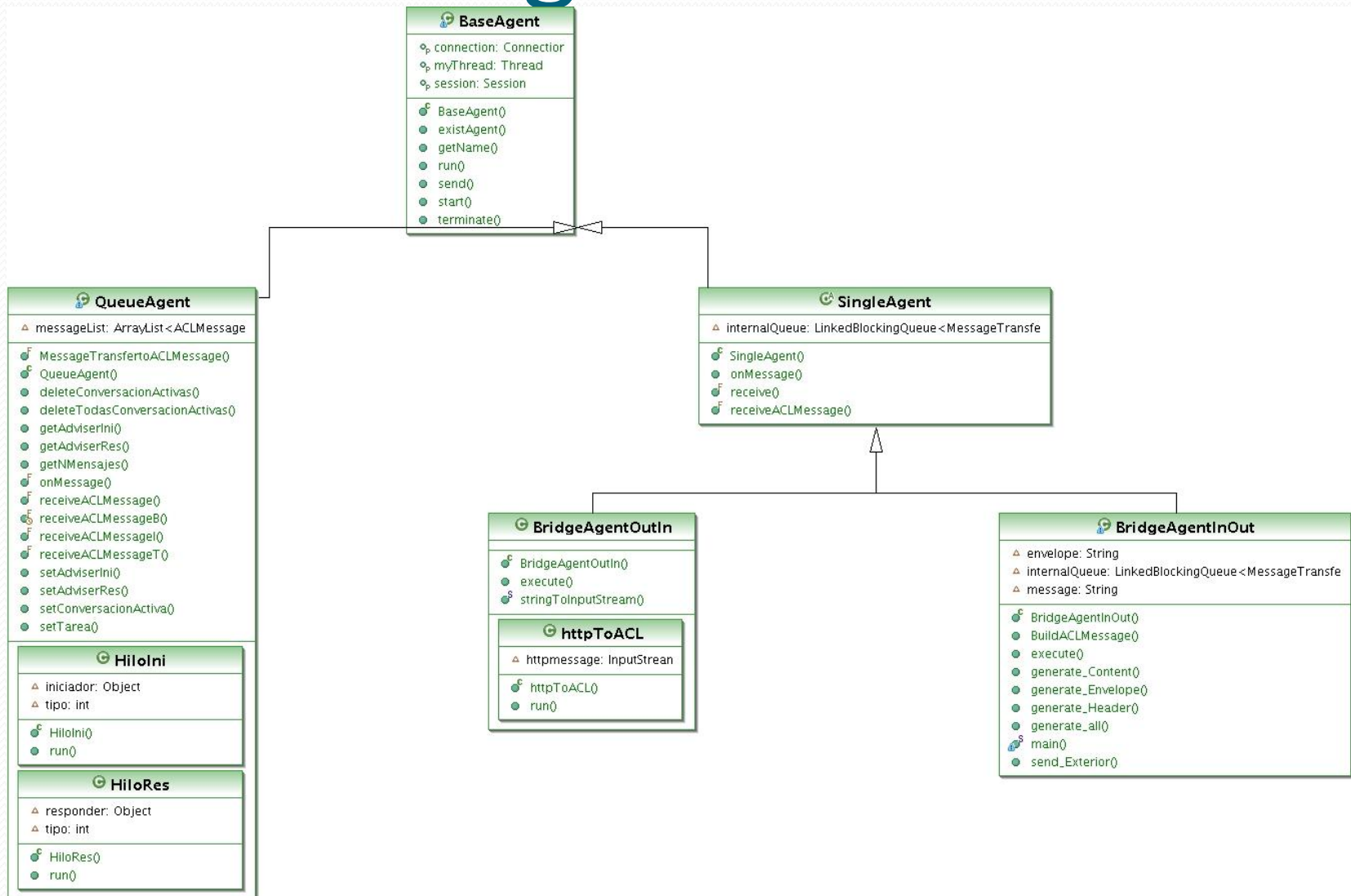
AMQP Features

The AMQP protocol defines a transport layer and a functional layer.

A binary protocol with modern features, it is:

- *Multi-channel*
- *Negotiated*
- *Asynchronous*
- *Quite secure*
- *Portable*
- *Neutral*
- *Quite efficient*

Simple Diagram of Agents Model



Format of Qpid-Messages

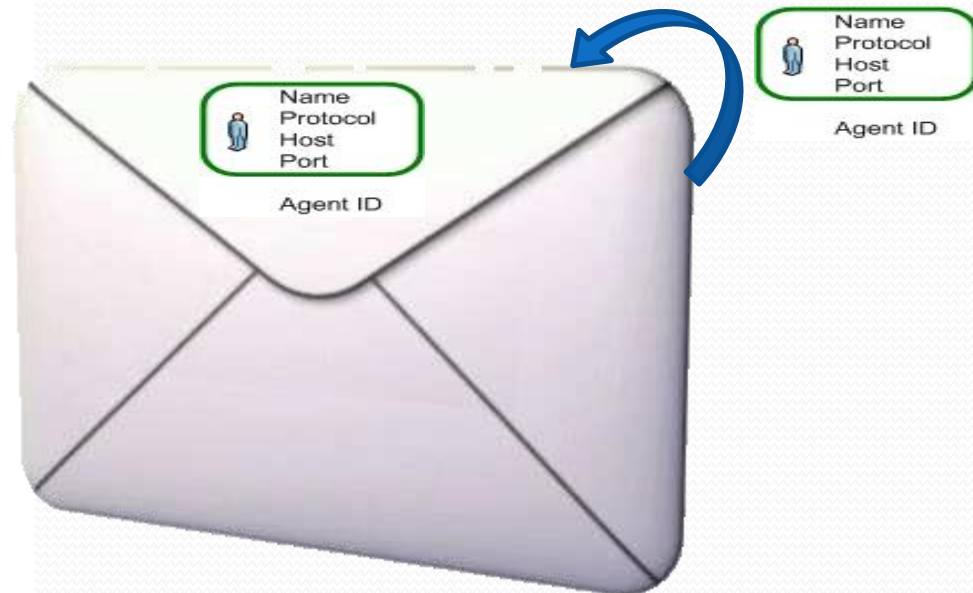
ACLMessage
ACCEPT_PROPOSAL: int
ACREE: int
CANCEL: int
CFP: int
CONFIRM: int
content: String
conversation_id: String
DISCONFIRM: int
encoding: String
FAILURE: int
INFORM: int
INFORM_IF: int
INFORM_REF: int
in_reply_to: String
language: String
NOT_UNDERSTOOD: int
ontology: String
PROPAGATE: int
PROPOSE: int
PROXY: int
performative: int
protocol: String
QUERY_IF: int
QUERY_REF: int
REFUSE: int
REJECT_PROPOSAL: int
REQUEST: int
REQUEST_WHEN: int
REQUEST_WHENEVER: int
receiver: ArrayList<AgentID>
reply_byInMillisec: long
reply_with: String
SUBSCRIBE: int
UNKNOWN: int

ACLMessage()
addReceiver()
clearAllReceiver()
clone()
createReply()
getConversationId()
getInReplyTo()
getPerformativeInt()
getReceiver()
getReceiverList()
getReplyBy()
getReplyByDate()
getReplyTo()
getReplyWith()
getTotalReceivers()
setConversationId()
setInReplyTo()
setPerformative()
setReceiver()
setReplyByDate()
setReplyTo()
setReplyWith()

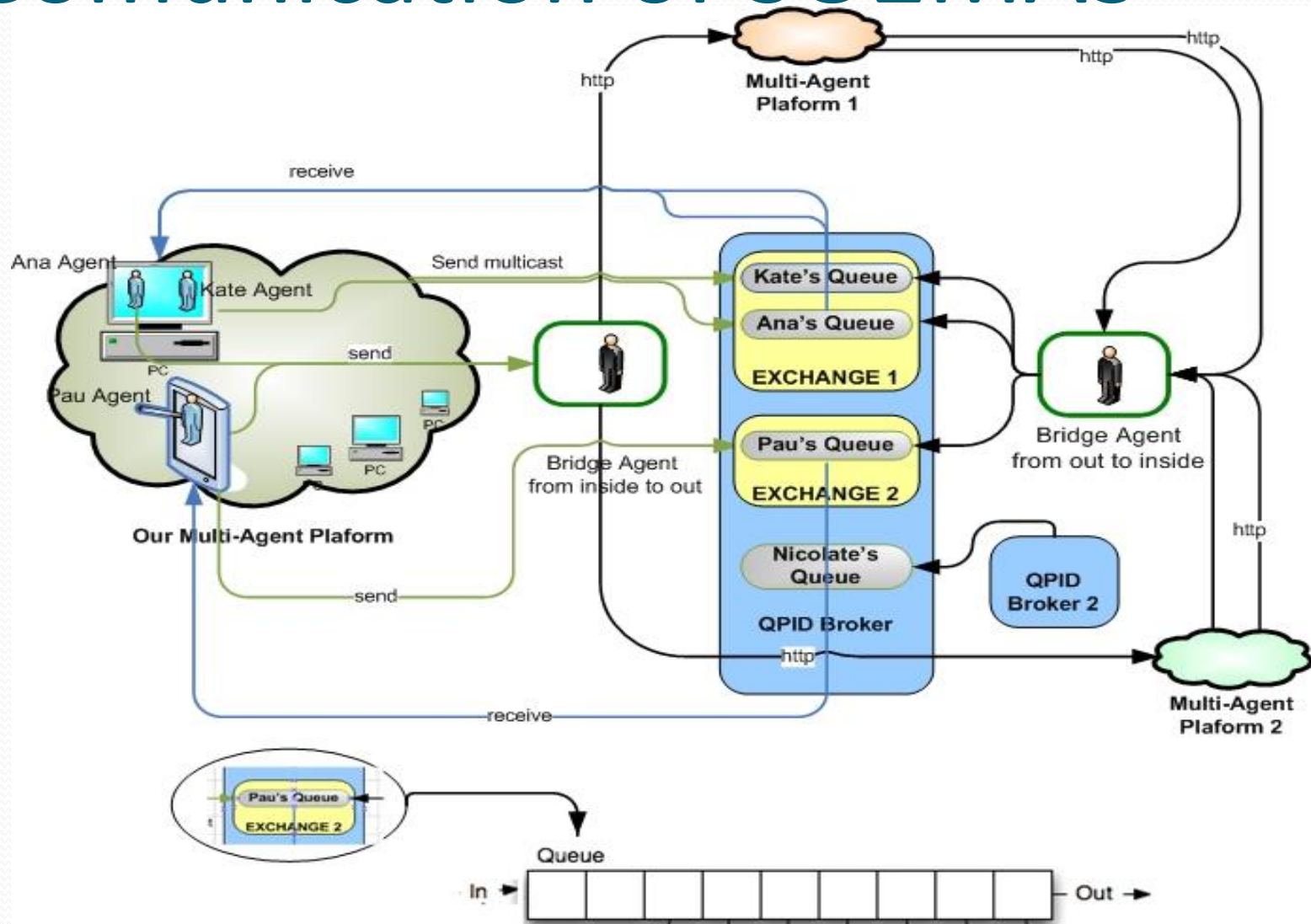
AgentID
host: String
name: String
port: String
protocol: String

AgentID()
AgentID()
AgentID()
addresses_all()
addresses_single()
getLocalName()
name_all()
toString()

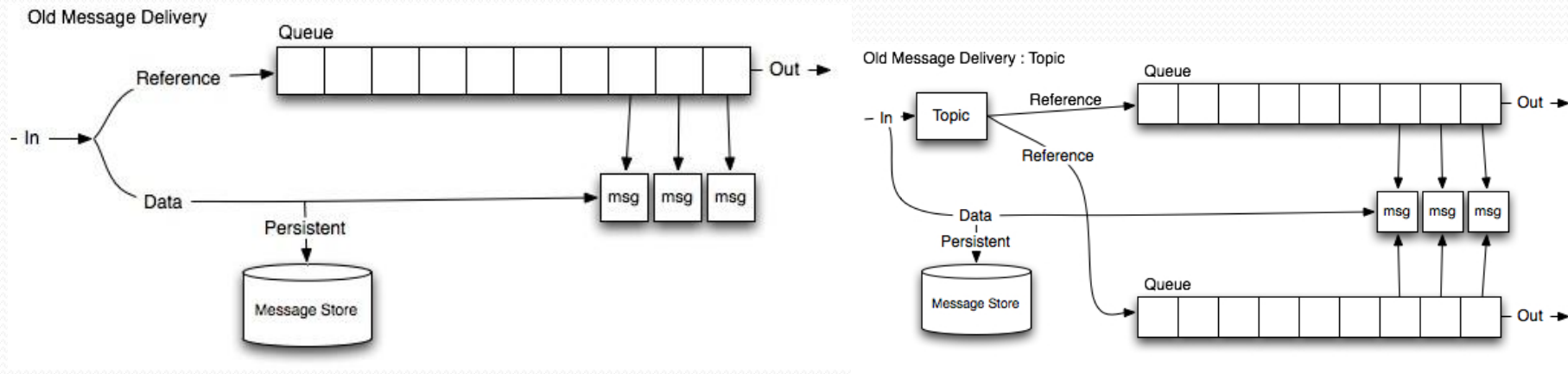
ISO8601
ISO8601()
main()
toDate()
toRelativeTimeString()
toString()
toString()



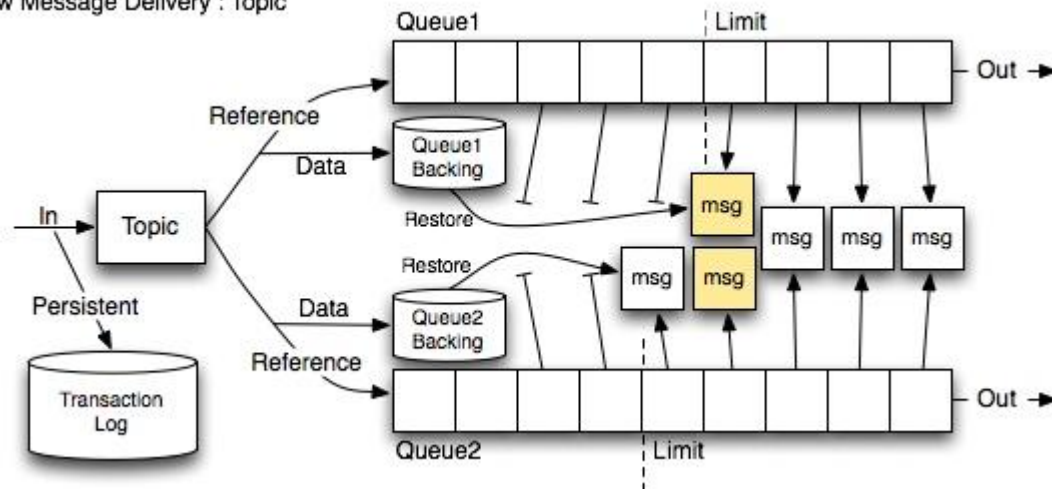
Comunication of SO2MAS



Queues

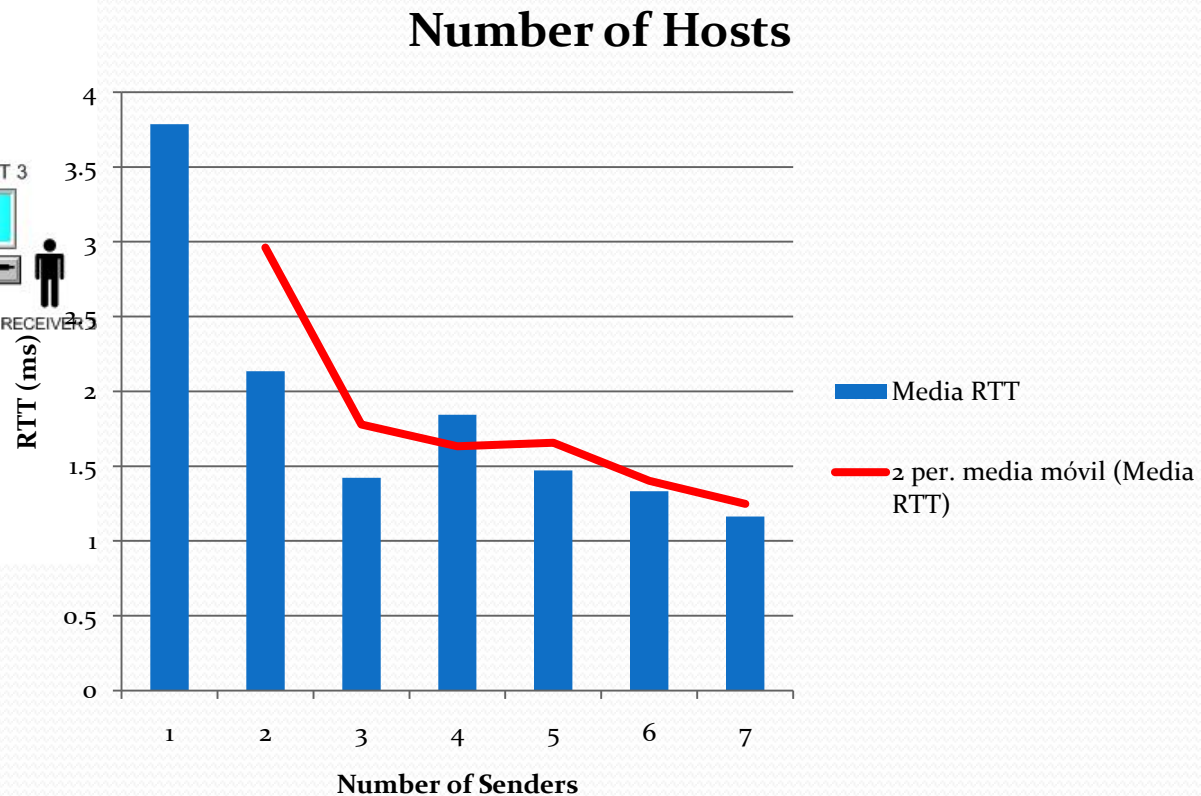
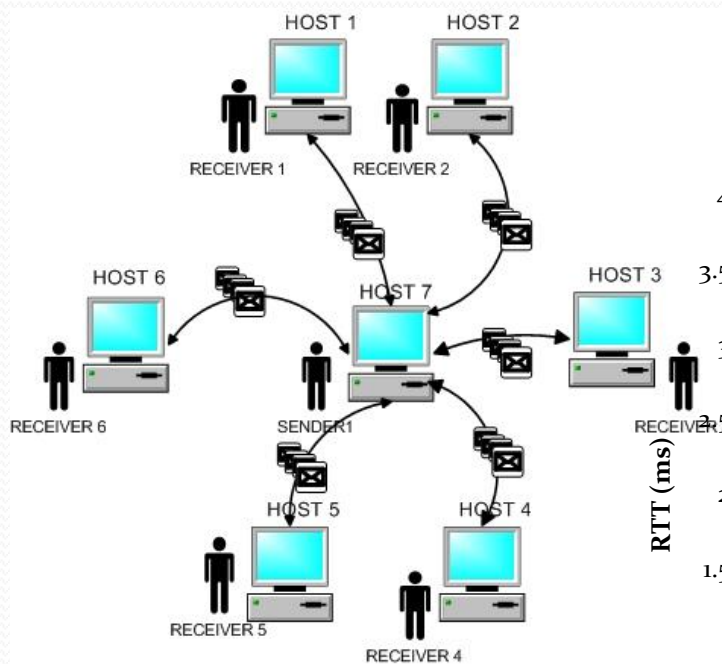


New Message Delivery : Topic

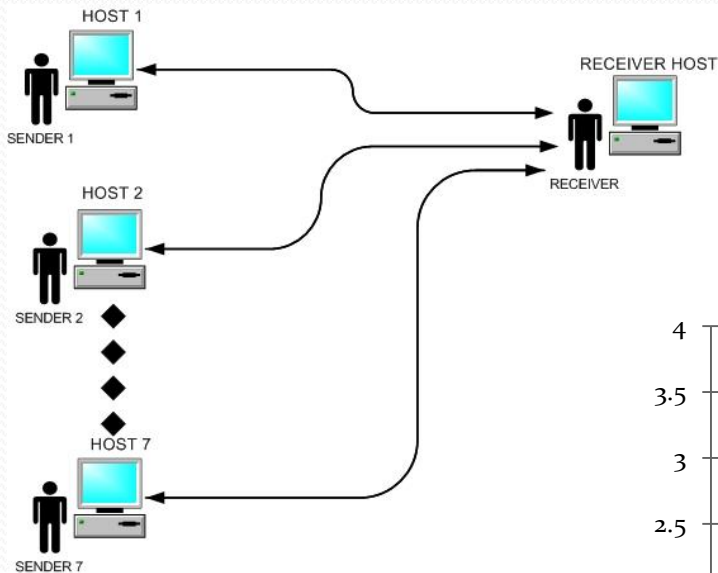


Bencharks:

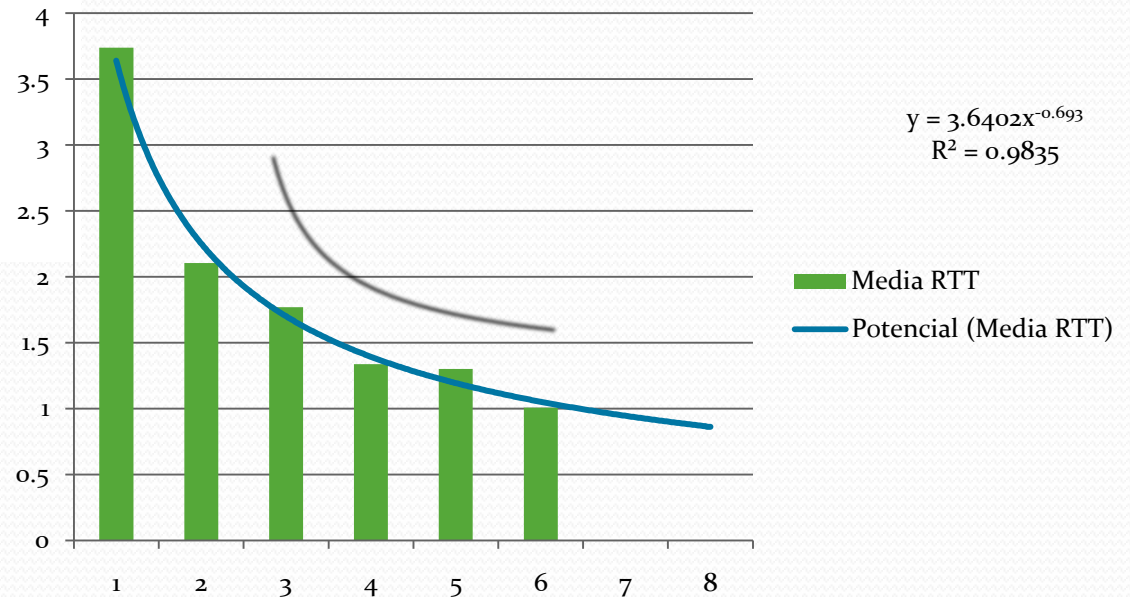
1.Number of Hosts



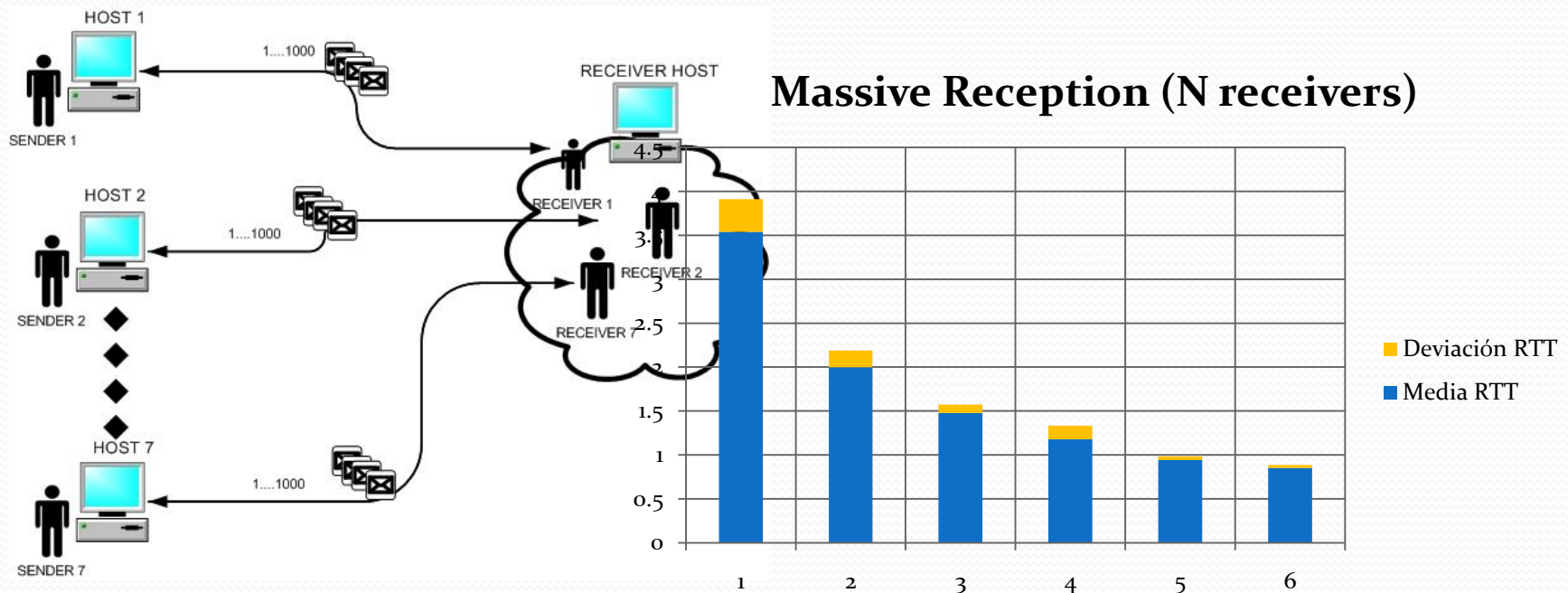
2. Massive Reception (1 receiver)



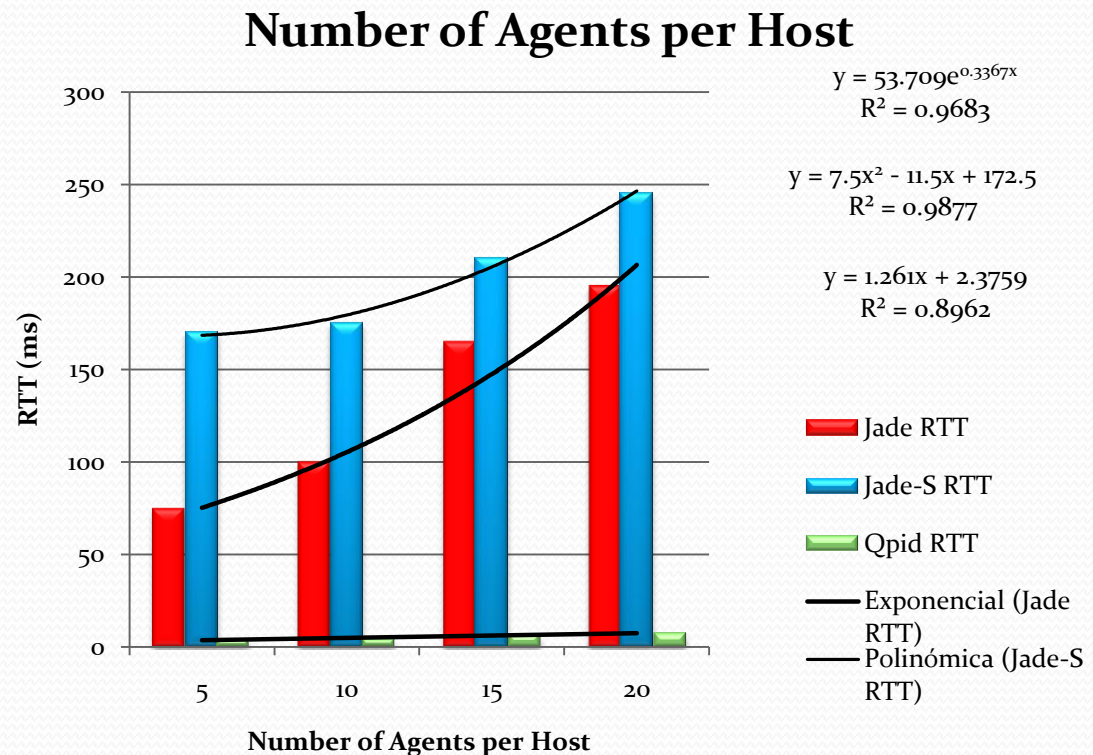
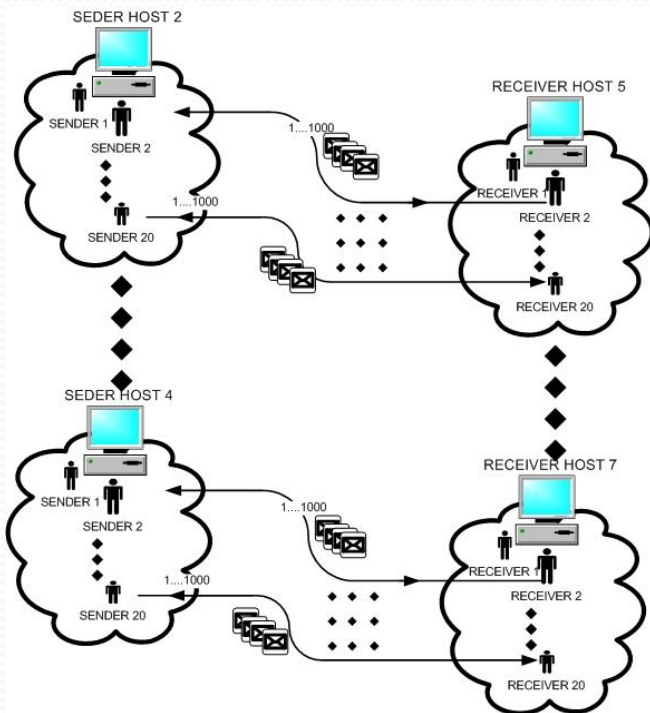
Massive Reception (1 receiver)



3. Massive Reception (N receivers)



4. Number of Agents per Host





Protocols (Joan....)



Joan 2

Conclusions

- Bla bla bla..

- ..

- .

- .