AGLOBE Messaging
AGLOBE platform architecture
Message Transport
Message sending method

Messages are sent as TCP/UDP packets, which makes this facility a dedicated solution for fast (and reliable) way of communication between AGLOBE components.

TCP (default) packets are slower as protocol is connection oriented providing flow control, which ensures a delivery correctness and completion.

UDP packets are faster than TCP but there is no guarantee for message delivery, however this might be added at sender/receiver level.

UDP frames are built manually and serialized chunked data is than sent to the recipient.

Recipient after receiving all frames, decapsulates them and deserialize a proper message, which is next passed for further processing.
public class SendAgent extends Agent implements MessageConstants {
    public SendAgentGUI gui = null;

    protected void send(Message m) {
        try {
            sendMessage(m);
        } catch (InvisibleContainerException ex) {
            logSevere("Cannot send message: "+ex);
            if (gui != null) {
                gui.logString("Cannot send message: " + ex);
            }
        }
    }

    public void handleIncomingMessage(Message m) {
        if (gui != null) {
            gui.logString(m.getContent().getClass().getName());
            gui.logMessage(m);
        }
        m.release();
    }
}
Running the container GUI

Run the following command to start the container:

```
java -jar AGlobe.jar -name containernname -[server|client] -gui
```
Start SendAgents

![Image of SendAgents interface]

**Sender**
- Container: 10.0.0.2 : 50624 /
cserver
Name: SendAgentA-cserver001.1323231597327 A

**Receiver**
- Container: 10.0.0.2 : 50624 /
cserver
Name: nTA-cserver001.1323231597327 A

**Performative:** REQUEST
Hello World!

**Content:**

**Protocol:**

**Ontology:** java.lang.String

**Conv. ID:**

**In Reply To:**

**Reply With:**

**Reason:**

Send

Clear incoming log
Senders and receivers

Identification of a message recipient is solved through Address object which holds a unique String field fulfilling the following format:

```
aglobe://platformIP:port/container_name/[agent|service]/name
```

Agents have globally unique names. Services, on the other hand, have unique names within the container. Agent names are generated by the platform during creation process. Agents must implement following methods:

```
public void sendMessage(Message m);
public void handleIncomingMessage(Message m);
```
Messaging characteristics

Single facility responsible for dealing with all kinds of communication (local and remote, between agents and services).

Shared (connection) socket for handling incoming and outgoing remote messages.

Local messages are delivered instantly by passing them to an appropriate recipient.

**Unicast** and **multicast** message sending
Messaging efficiency

Using the lower level (Transport layer) from the TCP/IP model for data transfer makes it a specialized solution for this type of communication.

It does not employ any higher level protocol, which results in speed increase, and in case of TCP packets provides a sufficient level of reliability.
Conclusions

AGLOBE Messaging is suited for a large number of small messages.

Suggestions for further improvements:

Adding message queue to the MessageTransport could reduce the blocking time for sendMessage function call.

Separating Messaging solution as a standalone component could make it more reusable.
Thank you