

8. Communicatie

In sommige multiagent-based-simulations is het nodig dat de agents met elkaar kunnen communiceren. Wanneer een agent een simpele taak heeft zal dit niet noodzakelijk zijn. Maar als de taak van de agent te complex, uitgebreid of ingewikkeld is, dan zullen de agents met elkaar moeten communiceren om hun taak zo goed mogelijk uit te voeren. Ik heb met behulp van een externe bron uit gezocht hoe de agent communicatietaal "KQML" werkt.

KQML

KQML staat voor Knowledge Query and Manipulation Language, en dit is een standaard die agents kunnen gebruiken om met elkaar te communiceren. En dit kan bijvoorbeeld gebruikt worden voor om een database autonoom te onderhouden en up-to-date te houden. Een KQML-bericht bestaat uit drie verschillende lagen een content-, bericht- en communicatie-laag. KQML is een standaard voor de communicatie tussen agents. KQML is dus geen taal die de agents spreken.

Content-laag

De content-laag zorgt er daarom voor dat de agents met elkaar communiceren, overeenkomen met welke taal gecommuniceerd gaat worden, zoals KIF, KRSL of LOOM. Dit wordt gedaan doormiddel van een parameter **content-language** mee te geven in het bericht. Maar dit zou je ook van tevoren vast kunnen stellen, zodat de agents dit niet zelf hoeven te doen.

Bericht-laag

De bericht-laag is de belangrijkste laag uit het protocol, deze laag zorgt er namelijk voor dat de berichten verzonden worden. De volgende parameters zijn onderdeel van de bericht-laag.

Type: Type bericht bijvoorbeeld een vraag, een antwoord of een foutmelding.

Qualifiers: Wat er met dit bericht moet gebeuren, bijvoorbeeld dat er minimaal 1 antwoord terug gestuurd moet worden.

Content: De inhoud van het bericht.

Communicatie-laag

In de communicatie laag wordt aangegeven wie de verzender en ontvanger is van het bericht. Ook kunnen er multicast of broadcast berichten verzonden worden. En deze laag zorgt er ook voor dat elk bericht een uniek ID heeft. De volgende parameters zijn onderdeel van de communicatie-laag.

From: De verzender van het bericht.

To: De ontvanger van het bericht.

ID: Een uniek ID van het bericht.

Response-to: Het ID van het bericht waar op gereageerd wordt.

Voorbeeld

Een voorbeeld van hoe zo een bericht er uit zou kunnen zien is hier onder weergegeven.

```
{
  "From": 10
  "To": 11
  "ID": 0002
  "Response-ID": 0001
  "Type": "Response"
  "Qualifiers": None
  "Content": "agreed"
}
```

Gebruik in de simulatie

De agenten in onze simulatie communiceren op het moment niet met elkaar. Dus de agenten weten ook niet van elkaar of ze ziek zijn en kunnen hiermee dus geen rekening mee houden. Als een agent tegen de agenten in zijn directe omgeving kan zeggen "Pas op ik ben ziek!". Dan kunnen die agenten die niet aangestoken willen worden de directe omgeving van de zieke agent verlaten. Met deze vorm van communicatie kan kunnen de agenten rekening met elkaar houden. Of dit ook echt leidt tot minder zieke agenten is nog de vraag. Hier zo dan een experiment mee uitgevoerd moeten worden om dit verder te onderzoeken.

Bronnen

[https://www.researchgate.net/publication/324941890_KQML - From Scenario to Technology](https://www.researchgate.net/publication/324941890_KQML_-_From_Scenario_to_Technology)

https://www.researchgate.net/publication/2810566_An_Overview_of_KQML_A_Knowledge_Query_and_Manipulation_Language