

Distributed Artificial Intelligence and Intelligent Agents (ID2209): Project assignment

KIM HAMMAR, STOCKHOLM 16446

kimham@kth.se

I. INTRODUCTION

The requirements statement is essentially just a set of articulated requirements for the system/organization to be designed, for structural reasons the requirements are divided into various related models that use different levels of detail. The system in this context is a SmartMuseum Agent Framework, as of following the GAIA methodology [1] I will from here on frequently use the *organization* metaphor when referring to the system.

II. TASK 1 - MODELING WITH GAIA METHDOLOGY

I. Analysis

I.1 Requirements Statement

I.1.1 Mission Statement

The SmartMuseum organization has the purpose of connecting different people and entities that are in some sense involved in consuming or providing services related to art. The goal of the organization is to improve the overall experience for everyone involved. The organization should make it easier for consumers to view and find interesting art, for art-curators to provide art and reach out to consumers, for tourguides to find interested consumers as well as building relevant tours and finally for artists to sell their work.

I.1.2 Organization Description

The activity of a consumer viewing an art-artifact involves atleast three, sometimes four, or five main divisions: *tour-guide division*, *art-curator division*, *artist-management division*, *user-service division* and *artist-division*. The activity is initiated by the consumer who contacts the user-service division and selects some type of art-service, the user-service division support the consumer in requesting/retrieving the service from either the art-curator division or tour-guide-division. In parellel to managing consumer requests the tour-guide division browses art-artifacts that is curated by the art-curator division. Further more, the art-curator division participates in auctions for obtaining art-artifacts from the artist-management division, in parallel to managing requests from consumers and tourguides. Finally, the artist-management division initiates auctions for art-artifacts on request from artists.

The activities described above can the be modelled as an organization in the following way. The organization consists of 8 roles. The ARTCONSUMER who consumes arts in different forms. The USERHANDLER which the consumer uses to purchase and browse services for art-artifacts. The

TOURGUIDEADVERTISER which presents and attempts to sell virtual tours. The TOURGUIDEBUILDER which constructs a virtual tour of arts by retrieving arts matching some given preferences. The ARTBUYER who buys art to include in its gallery/museum. The ARTSELLER who sells art-artifacts. The ARTADVERTISER who promotes the curated artifacts to be included in virtual tours, visited directly by consumers or sold. And finally the ARTIST who produces art.

I.2 Roles Model

Assumption 1-A. Roles can find each other in some way in order to communicate

Role Schema:	<i>ArtConsumer</i>
Description:	Initiates activity of consuming art, which includes purchasing some service
Protocols and activities:	SelectService, PayForService, <u>ConsumeService</u>
Permissions:	reads supplied <i>availableServices</i> // list of services
Responsibilities	
Liveness:	$\text{ARTCONSUMER} = (\text{SelectService. PayForService. } \underline{\text{ConsumeService}})^{\omega}$
Safety:	<ul style="list-style-type: none"> • true

Figure 1: Schema for role ARTCONSUMER

Role Schema:	<i>UserHandler</i>		
Description:	Receives request to buy art-services from consumers and manages the process of the consumer purchasing and obtaining the service.		
Protocols and activities:	SelectService, PayForService, BuyArt, BuyVirtualTour, GetArtifacts, GetPublishedVirtualTours, <u>GenerateListOfArtServices</u>		
Permissions:	generates	<i>availableServices</i>	// <i>list of services</i>
	reads	supplied <i>publishedVirtualTours</i>	// <i>list of published virtual tours</i>
		supplied <i>artifacts</i>	// <i>list of art-artifacts</i>
Responsibilities			
Liveness:	$\text{USERHANDLER} = (All)^\omega$ $ALL = (\text{GetServices. } \underline{\text{GenerateListOfArtServices. UserInteraction. Service}})^\omega$ $\text{USERINTERACTION} = \text{SelectService. PayForService}$ $\text{GETSERVICES} = \text{GetArtifacts. GetPublishedVirtualTours}$ $\text{SERVICE} = \text{BuyArt} \mid \text{BuyVirtualTour}$		
Safety:	<ul style="list-style-type: none">• $availableServices = artifacts + publishedVirtualTours$		

Figure 2: Schema for role USERHANDLER

Role Schema:	<i>TourGuideAdvertiser</i>			
Description:	Presents virtual tours to potential buyers and sells tours.			
Protocols and activities:	<u>PublishVirtualTour</u> , RetrieveVirtualTour			
Permissions:	generates	<i>publishedVirtualTour</i>	//	<i>published description of tour</i>
	reads	supplied <i>virtualTours</i>	//	<i>list of virtual tours</i>
Responsibilities				
Liveness:	$\text{TOURGUIDEADVERTISER} = ([\text{PublishVirtualTour}] \parallel \text{RetrieveVirtualTour})^\omega$			
Safety:	<ul style="list-style-type: none">• $\text{publishedVirtualTour} \implies \exists\{x \in \text{virtualTours} \mid x \equiv \text{publishedVirtualTour}\}$			

Figure 3: Schema for role TOURGUIDEADVERTISER

Role Schema:	<i>TourGuideBuilder</i>			
Description:	Responsible for constructing virtual tours of art-artifacts. Looks up available artifacts at curators and then builds different types of tours.			
Protocols and activities:	GetArtifacts, <u>BuildVirtualTour</u>			
Permissions:	generates	<i>virtualTour</i>	//	<i>virtual tour of art-artifacts</i>
	reads	supplied <i>artifacts</i>	//	<i>list of artifacts</i>
Responsibilities				
Liveness:	$\text{TOURGUIDEBUILDER} = (\text{GetArtifacts. } \underline{\text{BuildVirtualTour}})^\omega$			
Safety:	<ul style="list-style-type: none">• $\forall \text{virtualTour.artifact} \quad \text{virtualTour.artifact} \in \text{artifacts}$			

Figure 4: Schema for role TOURGUIDEBUILDER

Role Schema:	<i>ArtBuyer</i>
Description:	Buys art-artifacts.
Protocols and activities:	BuyArt
Permissions:	generates <i>artifacts</i> // <i>list of purchased artifacts</i>
Responsibilities	
Liveness:	$\text{ARTBUYER} = (\text{BuyArt})^\omega$
Safety:	<ul style="list-style-type: none"> • true

Figure 5: Schema for role ARTBUYER

Role Schema:	<i>ArtSeller</i>
Description:	Sells art-artifacts.
Protocols and activities:	SellArt
Permissions:	read <i>artifacts</i> // <i>list of artifacts</i>
Responsibilities	
Liveness:	$\text{ARTSELLER} = (\text{SellArt})^\omega$
Safety:	<ul style="list-style-type: none"> • true

Figure 6: Schema for role ARTSELLER

Role Schema:	<i>ArtAdvertiser</i>
Description:	Advertises art, responds to queries of arts
Protocols and activities:	GetArtifacts
Permissions:	read <i>artifacts</i> // <i>list of artifacts</i>
Responsibilities	
Liveness:	$\text{GETARTIFACTS} = (\text{GetArtifacts})^\omega$
Safety:	<ul style="list-style-type: none"> • true

Figure 7: Schema for role ARTADVERTISER

II. Design

REFERENCES

- [1] Michael Wooldridge, Nicholas R. Jennings, and David Kinny. The gaia methodology for agent-oriented analysis and design. *Autonomous Agents and Multi-Agent Systems*, 3(3):285–312, September 2000.