

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 7 roles. The ARTCONSUMER (AC) who consumes arts in different forms. The USERHANDLER (UH) which the consumer uses to purchase and browse services related

to art. The TOURGUIDE (TA) which builds and offers virtual tours. The ARTBUYER (AB) who buys art to include in its gallery/museum, the ARTQUOTER (AQ) who quotes the price for arts and sells it to consumers. The ARTSELLER (AS) who sells art-artifacts produced by artists. And finally the ARTIST (A) who produces art.

I.2 Roles Model

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

Role Schema: ARTCONSUMER (AC)			
Description:			
Initiates activity of consuming art, which includes purchasing some service			
Protocols and activities:			
DownloadVirtualTour, BuyArt, <u>ConsumeService</u>			
Permissions:			
	reads	supplied <i>availableServices</i>	// <i>list of services</i>
		<i>money</i>	// <i>money of the consumer</i>
	generates	<i>valuation</i>	// <i>valuation of selected artifact</i>
		<i>artifactTitle</i>	// <i>title of selected artifact</i>
		<i>virtualTourTitle</i>	// <i>title of selected virtual-tour</i>
		<i>virtualTourTitle</i>	// <i>valuation of selected virtualTour</i>
		<i>moneyFor Artifact</i>	// <i>money for selected artifact</i>
Responsibilities			
Liveness:			
ARTCONSUMER = (GetService. <u>ConsumeService</u>) ^ω			
GETSERVICE = (DownloadVirtualTour BuyArt)			
Safety:			
• true			

Figure 1: Schema for role ARTCONSUMER

Role Schema: USERHANDLER (UH)																													
Description: Receives request to buy art-services from consumers and manages the process of the consumer purchasing and obtaining the service.																													
Protocols and activities: ManageArtPayment, GetVirtualTour, GetArtifactsList, GetVirtualTourList, <u>GenerateListOfArtServices</u>																													
Permissions: <table> <tr> <td>generates</td><td><i>availableServices</i></td><td>// list of services</td></tr> <tr> <td></td><td><i>bid</i></td><td>// bid to attempt to buy artifact</td></tr> <tr> <td>reads</td><td>supplied <i>virtualTours</i></td><td>// list of virtual tours</td></tr> <tr> <td></td><td>supplied <i>artifacts</i></td><td>// list of art-artifacts</td></tr> <tr> <td></td><td>supplied <i>money</i></td><td>// money of the consumer</td></tr> <tr> <td></td><td>supplied <i>artifactTitle</i></td><td>// title of artifact-purchase</td></tr> <tr> <td></td><td>supplied <i>virtualTourTitle</i></td><td>// title of virtual-tour selection</td></tr> <tr> <td></td><td>supplied <i>artifact</i></td><td>// bhought artifact or nil</td></tr> <tr> <td></td><td>supplied <i>virtualTour</i></td><td>// virtual-tour downloaded by consumer</td></tr> </table>			generates	<i>availableServices</i>	// list of services		<i>bid</i>	// bid to attempt to buy artifact	reads	supplied <i>virtualTours</i>	// list of virtual tours		supplied <i>artifacts</i>	// list of art-artifacts		supplied <i>money</i>	// money of the consumer		supplied <i>artifactTitle</i>	// title of artifact-purchase		supplied <i>virtualTourTitle</i>	// title of virtual-tour selection		supplied <i>artifact</i>	// bhought artifact or nil		supplied <i>virtualTour</i>	// virtual-tour downloaded by consumer
generates	<i>availableServices</i>	// list of services																											
	<i>bid</i>	// bid to attempt to buy artifact																											
reads	supplied <i>virtualTours</i>	// list of virtual tours																											
	supplied <i>artifacts</i>	// list of art-artifacts																											
	supplied <i>money</i>	// money of the consumer																											
	supplied <i>artifactTitle</i>	// title of artifact-purchase																											
	supplied <i>virtualTourTitle</i>	// title of virtual-tour selection																											
	supplied <i>artifact</i>	// bhought artifact or nil																											
	supplied <i>virtualTour</i>	// virtual-tour downloaded by consumer																											
Responsibilities Liveness: $\text{USERHANDLER} = (\text{ALL})^\omega$ $\text{ALL} = (\text{PresentServices} \parallel \text{HandleConsumerRequest})^\omega$ $\text{PRESENTSERVICES} = \text{GetServices. } \underline{\text{GenerateListOfArtServices}}$ $\text{GETSERVICES} = \text{GetArtifactsList. } \text{GetVirtualToursList}$ $\text{HANDLECONSUMERREQUEST} = \text{ManageArtPaymen} \mid \text{GetVirtualTour}$																													
Safety: <ul style="list-style-type: none"> • <i>availableServices</i> = <i>artifacts</i> + <i>virtualTours</i> 																													

Figure 2: Schema for role USERHANDLER

Role Schema: TOURGUIDE (TG)			
Description: Responsible for constructing virtual tours of art-artifacts. Looks up available artifacts at curators and then builds different types of tours. Sends tours to user-handlers.			
Protocols and activities: SendVirtualTours, SendVirtualTour, GetArtifactList, <u>BuildVirtualTour</u>			
Permissions:			
	generates	<i>virtualTour</i>	// <i>virtual tour of art-artifacts</i>
		<i>virtualTours</i>	// <i>list of virtual-tours</i>
	reads	<i>supplied artifacts</i>	// <i>list of artifacts</i>
		<i>supplied virtualTourTitle</i>	// <i>specific virtual-tour title</i>
Responsibilities			
Liveness:			
	TOURGUIDEBUILDER = (ConstructTour [Send]) ^ω		
	CONSTRUCTTOUR = (GetArtifactList. <u>BuildVirtualTour</u>) ^ω		
	SEND = SendVirtualTours SendVirtualTour		
Safety:			
	• $\forall virtualTour.artifact \quad virtualTour.artifact \in artifacts$		

Figure 3: Schema for role TOURGUIDE

Role Schema: ARTBUYER (AB)			
Description: Buys art-artifacts from art-sellers.			
Protocols and activities: BuyArt, SendArtifacts			
Permissions:			
	generates	<i>artifacts</i>	// <i>list of purchased artifacts</i>
		<i>bid</i>	// <i>bid to attempt to buy artifact</i>
	reads	<i>money</i>	// <i>the buyer's money</i>
		<i>artifactTitle</i>	// <i>title for a specific artifact</i>
		<i>supplied artifact</i>	// <i>bhought artifact or nil</i>
Responsibilities			
Liveness:			
	ARTBUYER = ([BuyArt] [SendArtifacts]) ^ω		
Safety:			
	• true		

Figure 4: Schema for role ARTBUYER

Role Schema: ARTQUOTER (AQ)			
Description: Quotes art and resells it to consumers			
Protocols and activities: <u>QuoteArt</u> , SellArt, GetArtifacts, SendArtifacts			
Permissions:			
	reads	supplied <i>artifacts</i>	// <i>list of artifacts</i>
		supplied <i>bid</i>	// <i>bid for artifact</i>
	generates	<i>quote</i>	// <i>quote of artifact</i>
Responsibilities			
Liveness:			
	ARTQUOTER = ((GetArtifacts. <u>QuoteArt</u> . SellArt) SendArtifacts) ^ω		
Safety:			
	• true		

Figure 5: Schema for role ARTQUOTER

Role Schema: ARTSELLER (AS)
Description: Sells art to art-traders/curators.
Protocols and activities: SellArt, GetArtDetails
Permissions: reads <i>supplied artifact</i> // <i>details of the artifact to be sold</i>
Responsibilities Liveness: Safety: • true
ARTSELLER = (GetArtifact. SellArt) ^ω

Figure 6: Schema for role ARTSELLER

Role Schema: ARTIST (A)
Description: Sells art to art-traders/curators.
Protocols and activities: <u>ProduceArt</u> , SendArtDetails
Permissions: generates <i>artifact</i> // <i>details of the artifact to be sold</i>
Responsibilities Liveness: Safety: • true
ARTIST = (<u>ProduceArt</u> . SendArtifact) ^ω

Figure 7: Schema for role ARTIST

I.3 Interaction Model

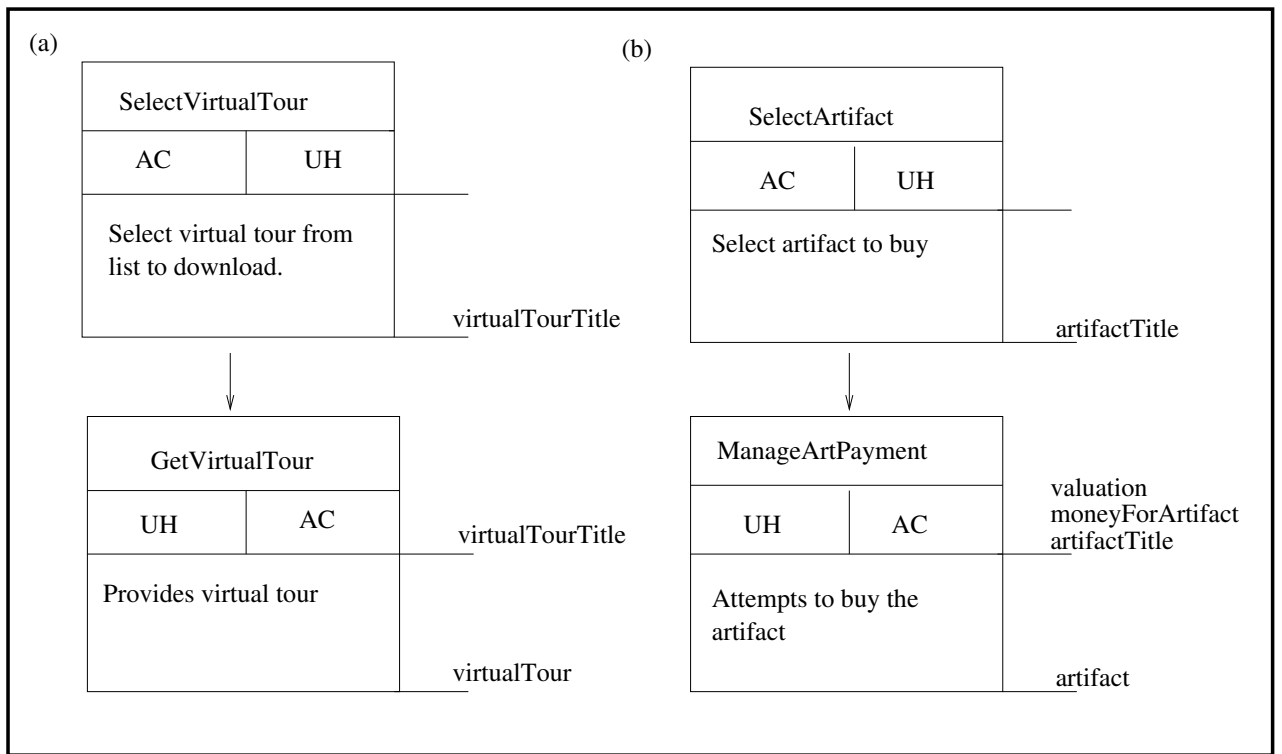


Figure 8: Definition of protocols associated with the ARTCONSUMER role: (a) DownloadVirtualTour, (b) BuyArt

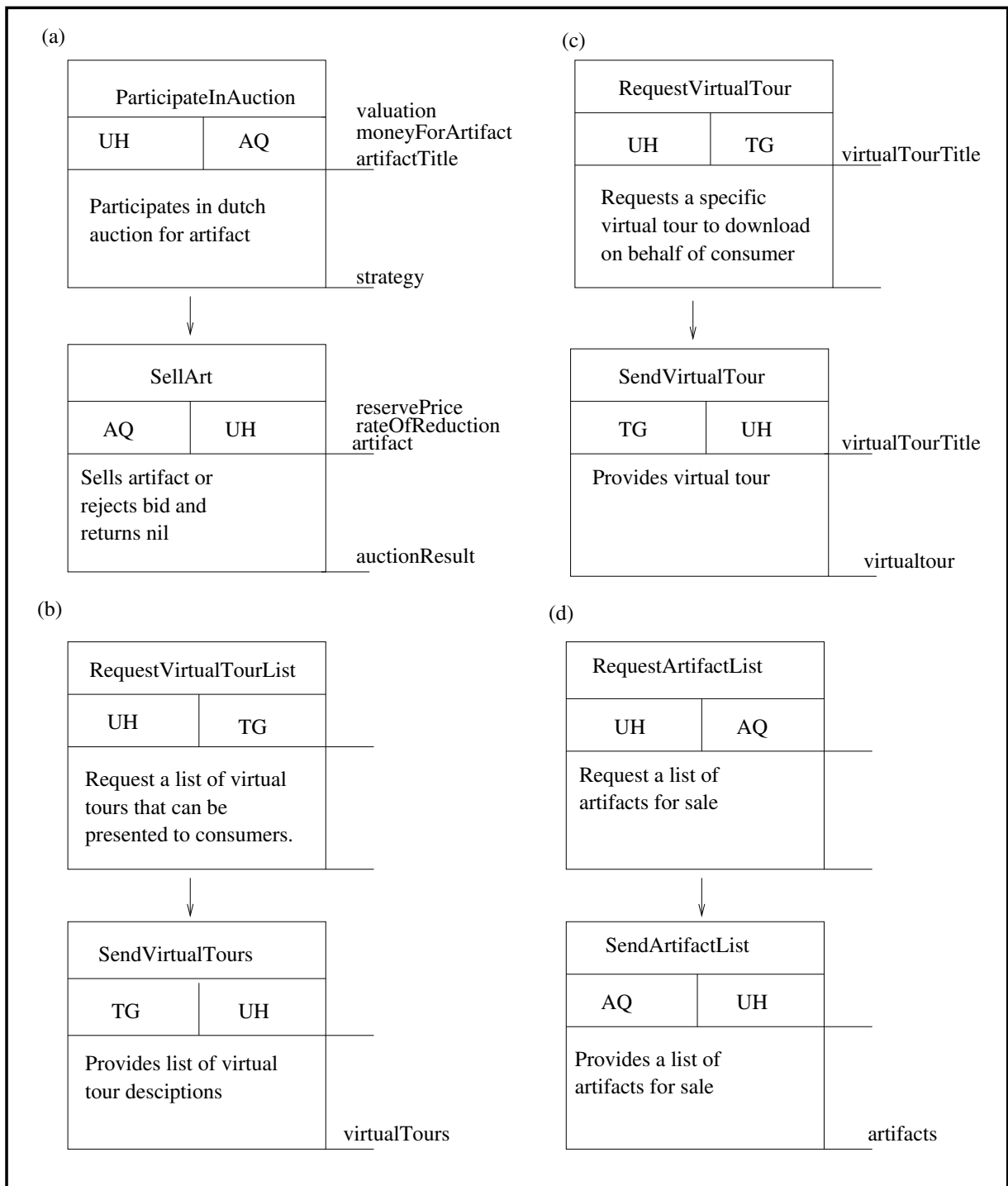


Figure 9: Definition of protocols associated with the USERHANDLER role: (a) ManageArtPayment, (b) GetVirtualTourList, (c) GetVirtualTour, (d) GetArtifactsList

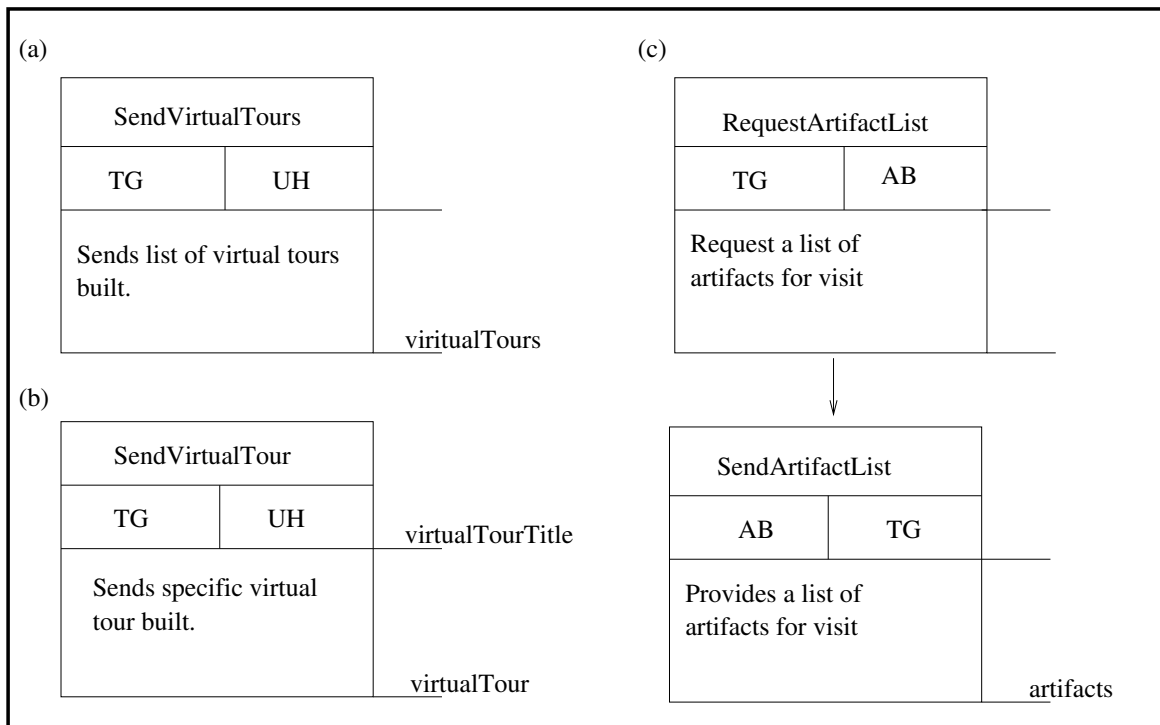


Figure 10: Definition of protocols associated with the TOURGUIDE role: (a) SendVirtualTours, (b) SendVirtualTour, (c) GetArtifactList

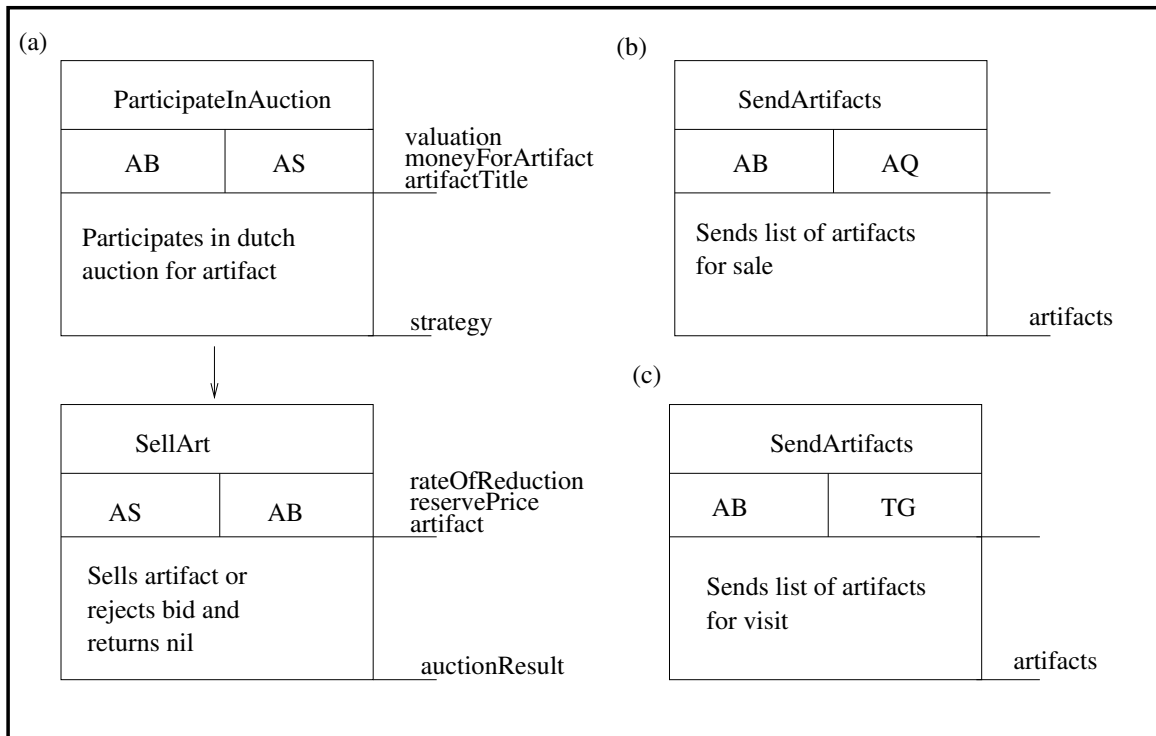


Figure 11: Definition of protocols associated with the ARTBUYER role: (a) BuyArt, (b) SendArtifacts

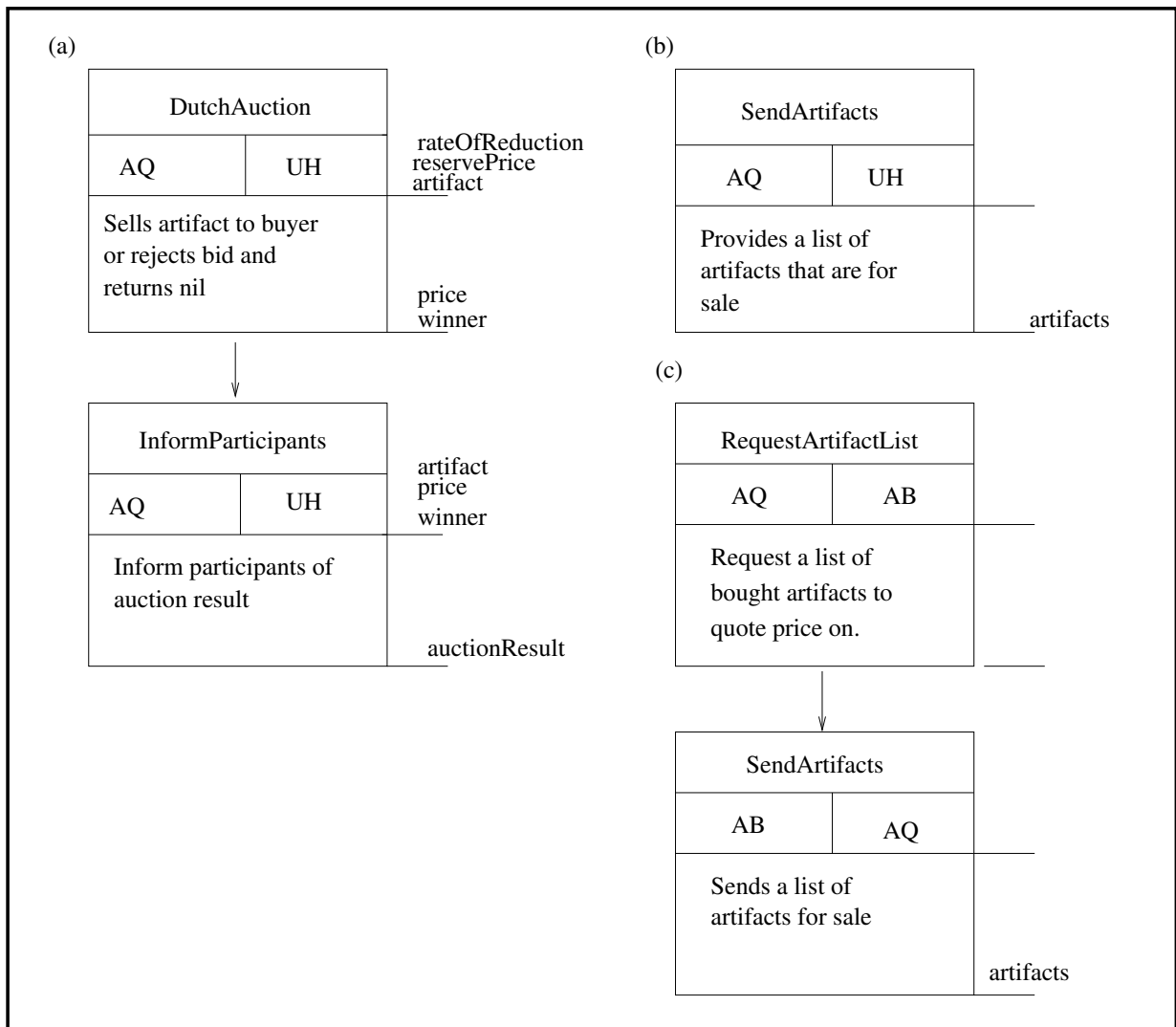


Figure 12: Definition of protocols associated with the ARTQUOTER role: (a) SellArt, (b) GetArtifacts, (c) SendArtifacts

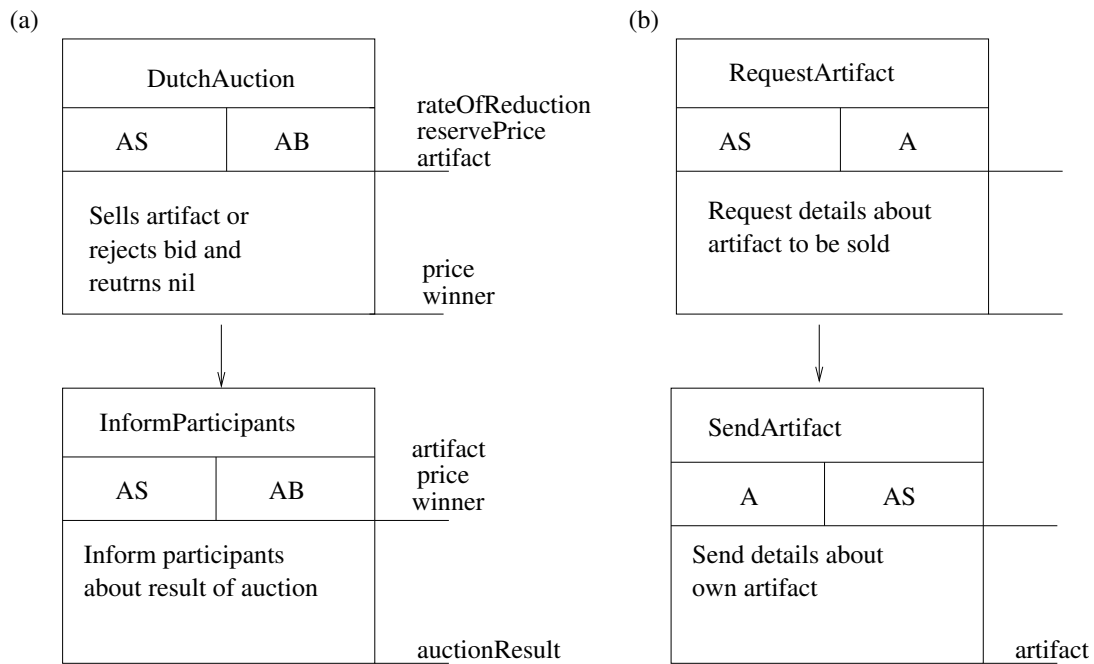


Figure 13: Definition of protocols associated with the ARTSELLER role: (a) SellArt, (b) GetArtDetails

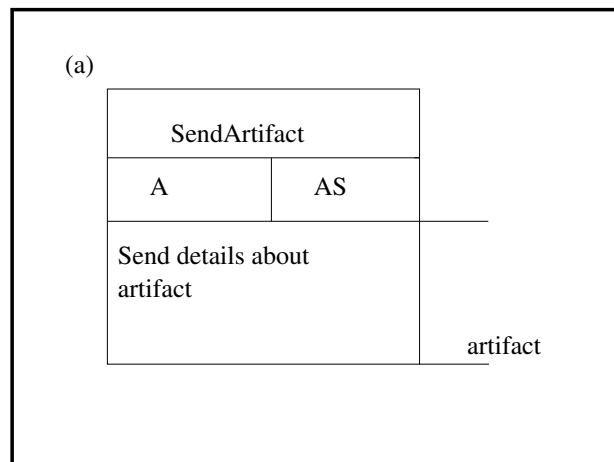


Figure 14: Definition of protocols associated with the ARTIST role: (a) SendArtDetails

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.