

LocalFoodEconomyGame.com Project

Draft – General Class Model (Metamodel)

A Sohodojo Project

Lead Model Developer: **Jim Salmons**

Project Team

Timlynn Babitsky, Sohodojo

Jim Salmons, Sohodojo

Jelal Younes, Grinnell College

06 August 2006

Fairfield, Iowa

Table of Contents

Model Detail	2
Logical View	2
Activity	3
Actor	4
Agent.....	5
Business_Process	5
Goal.....	6
Organization.....	7
Person.....	8
Role	9
Task.....	10
Resource.....	11
Role_Collaboration.....	12

Model Documentation

Model Detail

This is an initial draft of the **metamodel** that will guide and constrain the development of the **agent-based simulation** model for the **LocalFoodEconomyGame.com Project**. For more on this project, please visit the project web site at <http://LocalFoodEconomyGame.com>.

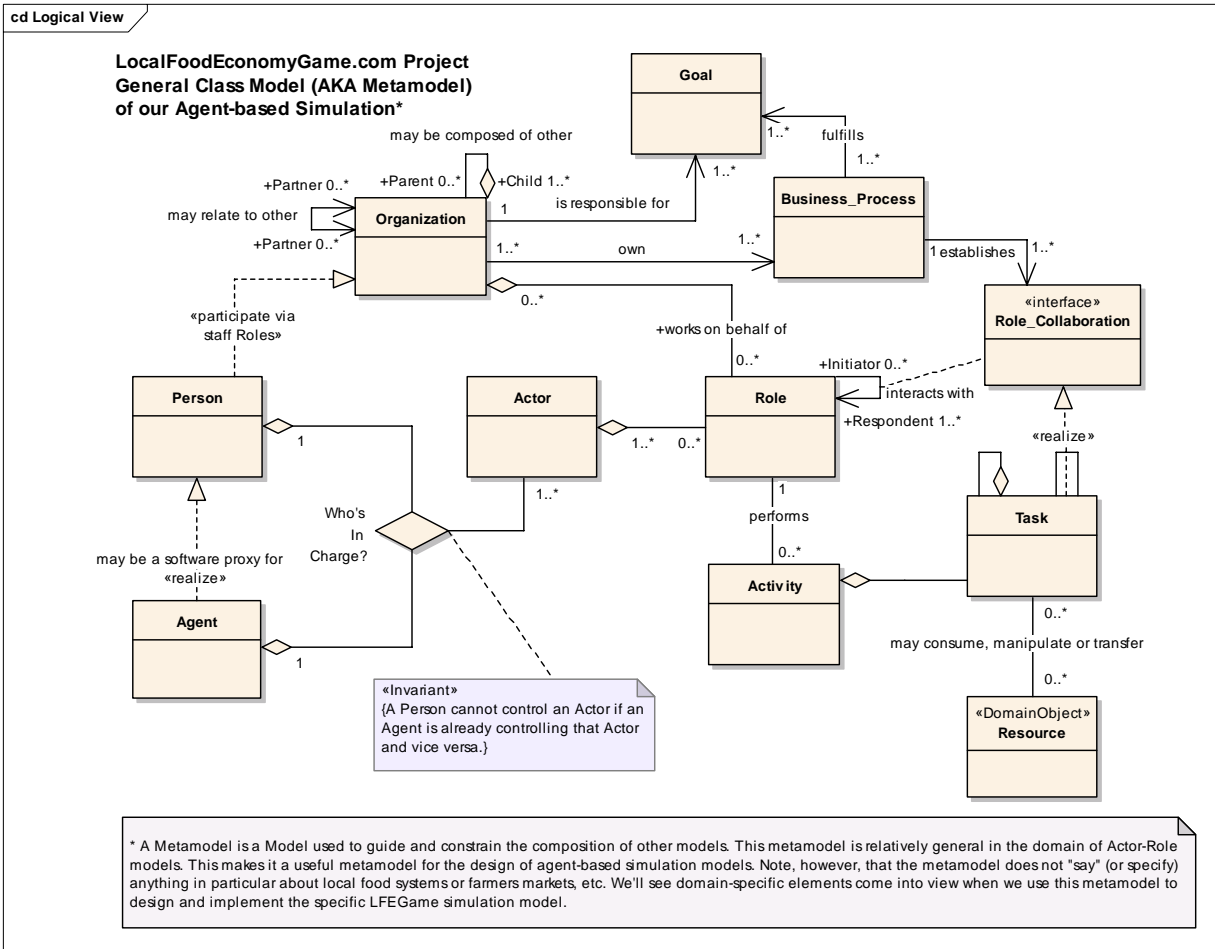
Logical View

Type: **Package**
Status: Proposed. Version . Phase 1.0.
Package: Model
Detail: Created on 8/5/2006 1:11:53 PM. Last modified on 8/6/2006 10:14:09 AM
GUID: {6B64971B-5EBD-4df2-A79D-144C5701B467}

Logical View - (Class diagram)

Created By: Jim Salmons on 12/3/2005
Last Modified: 8/6/2006
Version: 1.0. False
GUID: {CE10C75A-F1B4-4aba-B3D1-AE2D9818EE99}

* A Metamodel is a Model used to guide and constrain the composition of other models. This metamodel is relatively general in the domain of Actor-Role models. This makes it a useful metamodel for the design of agent-based simulation models. Note, however, that the metamodel does not "say" (or specify) anything in particular about local food systems or farmers markets, etc. We'll see domain-specific elements come into view when we use this metamodel to design and implement the specific LFEGame simulation model.



Activity

Type: Class
Status: Proposed. Version 1.0. Phase 1.0.
Package: Logical View *Keywords:*
Detail: Created on 8/5/2006. Last modified on.8/6/2006.
GUID: {311A681F-105B-4f38-94B7-EC55BA0C3049}

A Role is realized through the Activities performed in fulfilling the Role's responsibilities. To fulfill the responsibilities of performing a given Role, an Actor may engage in many different Activities.

Roles relate/interact with other Roles through Activities. Person "Jane" may be a Farmer Actor of a Farmers_Market_Vendor Role. Person "Joe" may be a Consumer Actor of a Farmers_Market_Customer Role. Jane and Joe may interact (that is, their Roles relate to each other) via Joe's Purchase_produce Activity and Jane's Sell_produce Activity within the "Saturday Morning, Aug 8th" instance of the Fairfield Farmers Market Business_Process.

Custom Properties

- isActive =

Connections

Connector	Source	Target	Notes
<u>Association</u> performs Unspecified	Public Activity	Public Role	
<u>Aggregation</u> Source -> Destination	Public Task	Public Activity	

Actor

Type: **Class**
Status: Proposed. Version 1.0. Phase 1.0.
Package: Logical View **Keywords:**
Detail: Created on 8/5/2006. Last modified on.8/6/2006.
GUID: {37341C6C-630E-4c54-B044-2F9134ACD7FF}

An Actor is how we objectify a Person's or Agent's ability/perspective to "act out" a Role. Attributes such as Skill_Level, Certification, Domain_Knowledge_Level, etc. parameterize the Actor's "approach" or profile when performing the Activities of a Role.

By explicitly modeling the Actor of a Role, we provide the flexibility to provide wide variety to a Person's state of "Being in the World." That is, for example, a Person "Joe" who is a Farmer, may be a skilled Actor of the Customer Role in a Feed Store Business_Process context, but a low skilled Actor when acting as a Customer in a New York City Art Gallery Business_Process context. Farmer "Jane," on the other hand, may be skilled in both domains. Or in another example, Person "Fred" may not be able to be considered in the pool of Actors for a House Sale Business_Process if his associated Realtor Actor does not have an up-to-date license.

The main point here is that by having a Person be an Actor of Roles gives us great flexibility in building specific instances of the LFEGame model. One Person can have a collection of Actors that represent the profile of how that Person can be/perform in the world. This keeps us from having to add loads of attribute and behavior complexity to the Person object. In this sense, the LGEGame metamodel reflects the distinction we make between Being and Doing.

Custom Properties

- isActive =

Connections

Connector	Source	Target	Notes
<u>Aggregation</u> Source -> Destination	Public Role	Public Actor	
<u>Association</u> Unspecified	Public Actor	Public Who's In Charge?	

Agent

Type: Class
Status: Proposed. Version 1.0. Phase 1.0.
Package: Logical View *Keywords:*
Detail: Created on 8/5/2006. Last modified on.8/6/2006.
GUID: {42AC85A9-3C3E-41b0-B20C-6FE513A538F5}

In an agent-based simulation, there may be no actual Person's who are Actors of Roles.

All Actors may be "animated" (become active) via Agent (Sims) which are software objects that can function as proxies for Persons. In such proxy roles, the Agent must be capable of recognizing Task interaction triggers, must be able to assess the state of the World in which the Agent is to act, and then perform a CognitiveAct (subject of another supplemental model) to determine the Task response.

Custom Properties

- isActive =

Connections

Connector	Source	Target	Notes
<u>Aggregation</u> Source -> Destination	Public Who's In Charge?	Public Agent	
<u>Realisation</u> may be a software proxy for Source -> Destination	Public Agent	Public Person	

Business_Process

Type: Class
Status: Proposed. Version 1.0. Phase 1.0.
Package: Logical View *Keywords:*
Detail: Created on 8/5/2006. Last modified on.8/6/2006.
GUID: {EF18B42D-6A06-4970-85DE-43D6201A12CC}

In this metamodel, Business_Processes are relatively lightweight. That is, their role is to know which Actor Roles and initial Activities to instantiate in order to achieve their Goals. This lightweight nature is what gives this metamodel its characterization an Actor-Role oriented model.

In most Workflow or Business Process Management systems, Process is "King." The Business Process model element is the "prime mover" and "referee" of the model's activity. Such Process-oriented systems can be easily understood and implemented as useful systems. However, they often lack some of the "faithfulness" to how we think about and participate in Real World systems.

For example, we may know that a Sales_Process is at work in a given context. But we more often think of this as a Buyer and Seller (Roles) engaged in a series of Activities (Negotiation, Purchase Activities) that result in a Sale (Goal). The Sales_Process may conceptually encapsulate this scenario, but it is the Actors' playing Roles engaged in

interactive Activities that realize the process.

So Business_Processes in an Actor-Role system are more like a Matchmaker than a Traffic Cop or Orchestra Conductor. Yes, Business_Processes play a vital role. But this role is not preeminent.

Custom Properties

- isActive =

Connections

Connector	Source	Target	Notes
<u>Association</u> establishes Source -> Destination	Public Business_Process	Public Role_Collaboration	
<u>Association</u> own Source -> Destination	Public Organization	Public Business_Process	
<u>Association</u> fulfills Destination -> Source	Public Goal	Public Business_Process	

Goal

Type: **Class**
Status: Proposed. Version 1.0. Phase 1.0.
Package: Logical View *Keywords:*
Detail: Created on 8/5/2006. Last modified on.8/6/2006.
GUID: {9EC47DB7-FFFC-463f-8FF5-3B4F334295D2}

A Goal is a specification of a desired acquisition or action state that motivates current action. A Goal is often tangible or at least measurable to the extent necessary to know if the Goal Thing/State has been acquired or achieved.

Though not strictly needed to implement an Actor/Role-based system, measurable Goals do provide a valuable means to cease or adjust Role Activity.

Custom Properties

- isActive =

Connections

Connector	Source	Target	Notes
<u>Association</u> fulfills Destination -> Source	Public Goal	Public Business_Process	

Connector	Source	Target	Notes
Association is responsible for Source -> Destination	Public Organization	Public Goal	

Organization

Type: Class
Status: Proposed. Version 1.0. Phase 1.0.
Package: Logical View *Keywords:*
Detail: Created on 8/5/2006. Last modified on.8/6/2006.
GUID: {C4AC0742-31F4-4f5d-A827-C174C37D6BA9}

An Organization is a "branded wrapper" of model elements that collectively perform some activity in order to achieve some goal(s). An Organization cannot do anything itself. It is not corporal in this sense. It's ability to "act" is based on its ability to be perceived and related to in a persistent manner. In the private sector, an Organization may be a corporation. More generally, it may be an association, school, or club. The Family of a family farm may be an Organization in the context of our LFEGame model.

In order to perform Activity to meet its Goals, an Organization aggregates Roles and Resources to implement its Business_Processes. (More accurately, we should say its Organization_Processes. But we have elected to go with the more widely recognized term Business Process. But we're thinking more in the loose sense of "getting down to business" rather than the more specific sense of business as in private sector business processes.)

Persons relate to Organizations through Staff or Employee Roles. Organizations may be aggregated through whole-part relationships (such as a Company Organization being made up of many Department Organizations).

Organizations may relate to other Organizations. These relationships are operationalized through Role-Activity interactions that are initiated by Business_Processes that have a Goal of fulfilling such Organizational relationships (partnerships or service-responsibilities).

Like Persons and Business_Processes, the Organization is relatively lightweight. Its implementation is modeled as little more than the attributes essential to identity. An Organization's ability to "Be in the World" is, rather, a function of the aggregation of the Roles, Business_Processes, and Resources that it can bring to bare on pursuit of its Goals.

Custom Properties

- isActive =

Connections

Connector	Source	Target	Notes
Aggregation may be composed of other Source -> Destination	Public Parent An organization may be composed of elements that are organizations in their own right. These	Public Child A sub-organization is often known as a Department, division, subsidiary, etc. Organization	

Connector	Source	Target	Notes
	sub-organizations may be loosely or tightly coupled to the parent Organization. Organization		
Association may relate to other Bi-Directional	Public Partner Organization	Public Partner Organization	Organizations may relate to other Organizations. These relations are objectified through Business Process collaborations. These collaborations are objectified through interactions of Person as Actors of Roles whose Activities require Task interaction with another Organization's Actor of Role Activity.
Aggregation Source -> Destination	Public works on behalf of Role	Public Organization	
Realisation Source -> Destination	Public Person	Public Organization	
Association own Source -> Destination	Public Organization	Public Business_Process	
Association is responsible for Source -> Destination	Public Organization	Public Goal	

Person

Type: Class
Status: Proposed. Version 1.0. Phase 1.0.
Package: Logical View *Keywords:*
Detail: Created on 8/5/2006. Last modified on.8/6/2006.
GUID: {4E1954A5-478D-4e24-B915-ADA8BA5283B1}

A Person is both the Real World, flesh and blood, classic software system "User" of the LFEGame exploratory learning environment. But he/she is also represented in the "Silicon World" realm as a proxy object that represents that User on "the other side of the glass."

A Real World Person/User will interact with the LFEGame model by using an Internet browser to view the a running instance of the model, think about his/her interactions, then use the keyboard and mouse to interact/participate with the simulation.

As a software proxy object, the Person is a rather minimal "placeholder" object. Its attributes are limited to such personally identifiable things as Name, Age, Address at a minimum.

Constraints

- Person objects are minimalist : (*Invariant.Status is Approved*)
Do not model the Person with attributes beyond the minimal ones that provide identity. A Person "is/does" in the world (Real and simulated) as a Actor of Roles.

Custom Properties

- isActive =

Connections

Connector	Source	Target	Notes
<u>Aggregation</u> Source -> Destination	Public Who's In Charge?	Public Person	
<u>Realisation</u> Source -> Destination	Public Person	Public Organization	
<u>Realisation</u> may be a software proxy for Source -> Destination	Public Agent	Public Person	

Role

Type: Class
Status: Proposed. Version 1.0. Phase 1.0.
Package: Logical View *Keywords:*
Detail: Created on 8/5/2006. Last modified on.8/6/2006.
GUID: {416279E9-E470-43d3-A3A2-681C698830B0}

Roles are a central "organizing principle" of an agent-based simulation system.

Through aggregation (Roles being made up of sub-Roles, whole-part relations), we get "compositional power." For example, a Manager Role may be composed of a number of sub-Roles which include being an Approver (Role) of some decision or action. By modeling the fine-grained Exchange Activity of Submitter-Approver (Roles), we can more easily and effectively build higher-order Roles where a Role requires Approver behavior.

Note that Role-(sub)Role aggregation is optional. The metamodel provides for this level of complexity, but this does not mean that every instance of a LFEGame model will have to use it. There are plenty of situations where Roles with be "one level deep."

Even though Roles may be course-grained (one-level deep), this does not mean that a model need be too simplistic to be useful. Since a Actor can "act out" many Roles, there is plenty of opportunity to model variety and comprehensive action. For example, Person "Joe" may be a Farmer Actor who, at various times in various contexts, plays the Roles of Crop_Planter, Tractor_Mechanic, Farmers_Market_Vendor, etc. In each of these Roles, Joe may engage in a wide variety of contextually relevant Activities.

Custom Properties

- isActive =

Connections

Connector	Source	Target	Notes
<u>Aggregation</u> Source -> Destination	Public Role	Public Actor	
<u>Aggregation</u> Source -> Destination	Public works on behalf of Role	Public Organization	
<u>Association</u> performs Unspecified	Public Activity	Public Role	
<u>Association</u> interacts with Source -> Destination	Public Initiator A Role's Actor may be engaged in Activities that relate the Role to zero or more other Roles. These relationships are implemented through Task interactions within Activities. Role	Public Respondent A Respondent is a Role engaged in Activities that cause interaction with other Roles. Role	

Task

Type: Class
Status: Proposed. Version 1.0. Phase 1.0.
Package: Logical View *Keywords:*
Detail: Created on 8/5/2006. Last modified on.8/6/2006.
GUID: {94DF2575-25F9-4061-8DE1-ABA3CC30A4AD}

Tasks are the fine-grained actions or exchanges that collectively implement an Activity.

Like Roles, Tasks may be aggregated in Whole-:Part compositions. Tasks within a Role relate to each other primarily through time-dependent workflow. Tasks may also be "wired up" to Tasks in Activities of other Actor's Roles. In this way, inter-Activity Task connection realize the Role_Collaborations of Role relationships within Business_Processes.

Custom Properties

- isActive =

Connections

Connector	Source	Target	Notes
<u>Realisation</u> Source -> Destination	Public Task	Public Role_Collaboration	
<u>Aggregation</u> Source -> Destination	Public Task	Public Activity	
<u>Association</u> may consume, manipulate or transfer Unspecified	Public Task	Public Resource	
<u>Aggregation</u> Source -> Destination	Public Task	Public Task	
<u>Association</u> Unspecified	Public Task	Public Task	

Resource

Type: **Class**
Status: Proposed. Version 1.0. Phase 1.0.
Package: Logical View *Keywords:*
Detail: Created on 8/5/2006. Last modified on.8/6/2006.
GUID: {8C4871E2-BB30-4630-9602-2236F68ADE62}

Resources are anything tangible that may be consumed, manipulated, or transferred as a result of Person's Role Activity. An obvious and primary example in the LFEGame's model of a Farmers Market is the Food_Stuff bought and sold at the Market. Craft objects are another likely candidate for transfer through exchanges of Buyers and Sellers at the Market.

An example of something consumed within this model would be ice to keep some Food_Stuff fresh and salable. Environmental factors of a specific instance of a Farmers Market Business_Process might determine how long the ice would last which, in turn, could affect how long a Vendor is able to stay open for business that given market session.

Custom Properties

- isActive =

Connections

Connector	Source	Target	Notes
<u>Association</u> may consume, manipulate or transfer Unspecified	Public Task	Public Resource	

Role_Collaboration

Type: **AssociationClass**
Status: Proposed. Version 1.0. Phase 1.0.
Package: Logical View *Keywords:*
Detail: Created on 8/5/2006. Last modified on.8/6/2006.
GUID: {245C96DE-98DE-44f8-AF7D-1E1B1B31EA1A}

We objectify Role_Collaborations as these specify the details of interaction responsibilities such that a Business Process can know how to instantiate the set of Roles and initial Activities in pursuit of its Goal.

Think of a Role_Collaboration as an Interface specification such that a Business_Process can know enough but not too-much of the inner structure and behavior of the Roles needed to achieve its Goals.

Custom Properties

- isActive =

Connections

Connector	Source	Target	Notes
Association establishes Source -> Destination	Public Business_Process	Public Role_Collaboration	
Realisation Source -> Destination	Public Task	Public Role_Collaboration	