Updated Architecture and agent definition of Baking and Packaging stage
Multiagent and Agent System

Arun Prabhu
Md Zahiduzzaman
Dharmin Bakaraniya

December 3, 2018

1 Updated Architecture

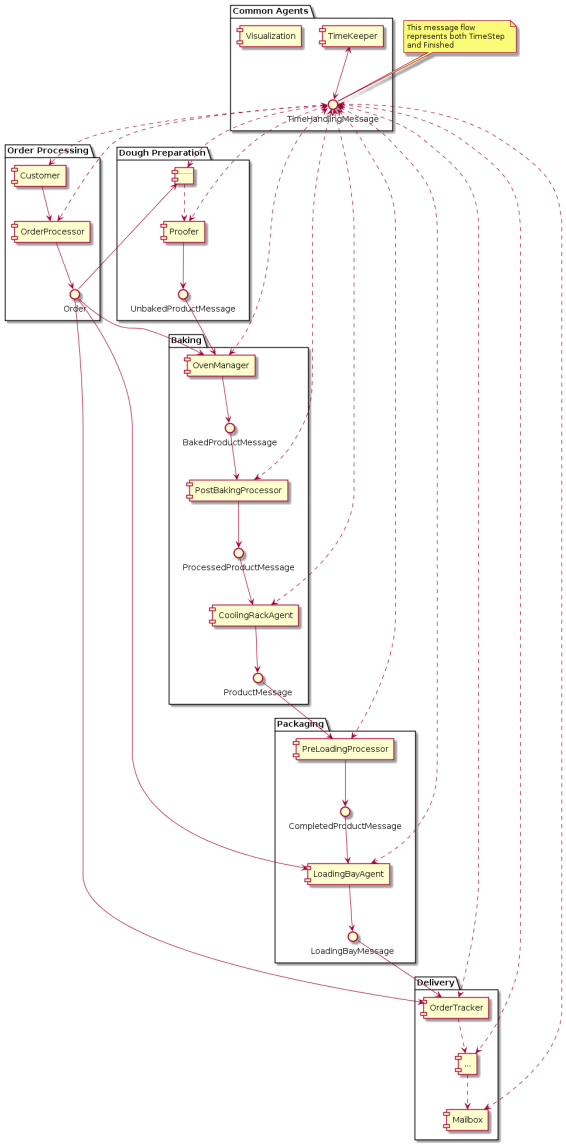

Figure 1: Architecture diagram explaining the interaction between customer and bakery agents

2 Architectural Description

We (right-brothers) are responsible for the Baking and Packaging stages, common agents like BaseAgent and TimeKeeper agent and Order board by stage visualization agent.

- **Common Agents**
  - **TimeKeeper**: It is responsible for the movement of time in the entire bakery eco-system. This is responsible for providing all the other agents with a common time reference so that everyone are in sync except the visualization agents. This also gets feedback from all the agents about the status of their tasks. If all the agents are done with whatever task they were supposed to finish in the time step, the TimeKeeper increments the time step. It is also responsible for shutting down the platform when the simulation time ends.
  - **BaseAgent**: BaseAgent is a parent to all agents in bakery simulation. It is mainly responsible for registering the agent to yellow pages and talking to TimeKeeper. It is also responsible for sending the message to visualization agents. Every agent that inherits BaseAgent only have to call `finished` when their task is finished for that time step.
  - **Visualization**: Visualization agent receives messages from baseAgent of all agents and parses this message to extract whetever information it needs. It then uses this information to create graphical representation using `JavaFX`.
- **Baking Stage Agents**
  - **OvenManager**: We assign a single Agent to manage all the ovens in the bakery. It receives the Order from OrderProcessor agent. It receives `UnbakedProductMessage` from `Proofer`. It bakes the products it received from `Proofer` and sends `BakedProductMessage` to `PostBakingProcessor`. It agregates products of same type if the product is not being baked.
  - **PostBakingProcessor**: It receives `BakedProductMessage` from `OvenManager`. It processes all the steps in the recipe of that product which occur between `Baking` and `Cooling` steps. It sends the `ProcessedProductMessage` to `CoolingRackAgent`.
  - **CoolingRackAgent**: It receives `ProcessedProductMessage` from `PostBakingProcessor`. It performs cooling step of the recipe corresponding to that product. It sends `ProductMessage` to `PreLoadingProcessor` in the packaging stage.

- **Packaging Stage Agents**
  - **PreLoadingProcessor**: It recieves `ProductMessage` from `CoolingRackAgent` from Baking stage. It performs all steps in recipe of that product which lie between `Cooling` and `Packaging`. It sends `CompletedProductMessage` to `LoadingBayAgent`.
  - **LoadingBayAgent**: It recieves `CompletedProductMessage` from `PreLoadingProcessor`. It prioritizes the pending orders in the todo list according to the earliest order delivery dates. Whenever any product requirement of any order in the todo list becomes available, a `LoadingBayMessage` is created with that particular product, and is sent to the `OrderTracking` agent of the `Delivery Stage`.

3 Class descriptions

3.1 TimeKeeper

- **Stage**: Common Agent
- **Agent/Object**: Agent
- **Static/Dynamic**: Static
- **Behaviour**: SendTimeStep (OneShot), TimeStepConfirmationBehaviour (Cyclic)
- **Messages in**:
  - finished (Sender: all agents)
- **Messages out**:
  - TimeStep (Receiver: all agents)

3.2 BaseAgent

- **Stage**: NA
- **Agent/Object**: Agent
- **Static/Dynamic**: Static
- **Behaviour**: PermitAction (Cyclic)
- **Messages in**:
  - TimeStep (Sender: TimeKeeper)
- **Messages out**:
  - finished (Receiver: TimeKeeper)
  - all messages(Receiver: Visualization Agents)

3.3 Visualization agent

- **Stage**: Common Agents
- **Agent/Object**: Agent
- **Static/Dynamic**: Static
- **Behaviour**: MessageServer (Cyclic)
- **Messages in**:
  - all messages (Sender: all agents)

3.4 OvenManager

- **Stage**: Baking
- **Agent/Object**: Agent
- **Static/Dynamic**: Static
- **Behaviour**: Bake (Cyclic), OrderServer (Cyclic), UnbakedProductsServer (Cyclic)
- **Messages in**:
  - Order (Sender: OrderProcessor)
  - UnbakedProductMessage (Sender: Proofer)
- **Messages out**:
  - BakedProductMessage (Receiver: PostBakingProcessor)

3.5 PostBakingProcessor

- **Stage**: Baking
- **Agent/Object**: Agent
- **Static/Dynamic**: Static
- **Behaviour**: BakedProductsServer (Cyclic), Process (Cyclic)
- **Messages in**:
  - BakedProductMessage (Sender: OvenManager)
- **Messages out**:
  - ProcessedProductMessage (Receiver: CoolingRackAgent)

3.6 CoolingRackAgent

- **Stage**: Baking
- **Agent/Object**: Agent
- **Static/Dynamic**: Static
- **Behaviour**: ProcessedProductServer (Cyclic), CoolProducts (Cyclic)
- **Messages in**:
  - ProcessedProductMessage (Sender: PostBakingProcessor)
- **Messages out**:
  - ProductMessage (Receiver: PreLoadingProcessor)

3.7 PreLoadingProcessor

- **Stage**: Packaging
- **Agent/Object**: Agent
- **Static/Dynamic**: Static
- **Behaviour**: Process (Cyclic), CooledProductServer (Cyclic), OrderServer (Cyclic)
- **Messages in**:
  - ProductMessage (Sender: CoolingRackAgent)
  - Order (Sender: OrderProcessor)
- **Messages out**:
  - CompletedProductMessage (Receiver: LoadingBayAgent)

3.8 LoadingBayAgent

- **Stage**: Packaging
- **Agent/Object**: Agent
- **Static/Dynamic**: Static
- **Behaviour**: OrderReceiver (Cyclic), CompletedProductReceiver (Cyclic)
- **Messages in**:
  - CompletedProductMessage (Sender: PreLoadingProcessor)
- **Messages out**:
  - LoadingBayMessage (Receiver: OrderTracker)