

Domain-specific modeling of Cooking Recipes

Xiaoxi Dong
Supervised by Prof Hans
Vangheluwe
04 2008

Content

I added some picture
of the results at the
end of the slides

- Some Characteristics of a recipe
- Modeling for cooking recipe
- Use the metamodel to create a recipe
- Transform recipe model to various forms
 - Generating a textual shopping list
 - Generating a textual cooking instruction
 - Generating a real-time cooking instructor
- Conclusions & Further development
- (The outcome of the project is shown with the presentation)

Some Characteristic of a Recipe

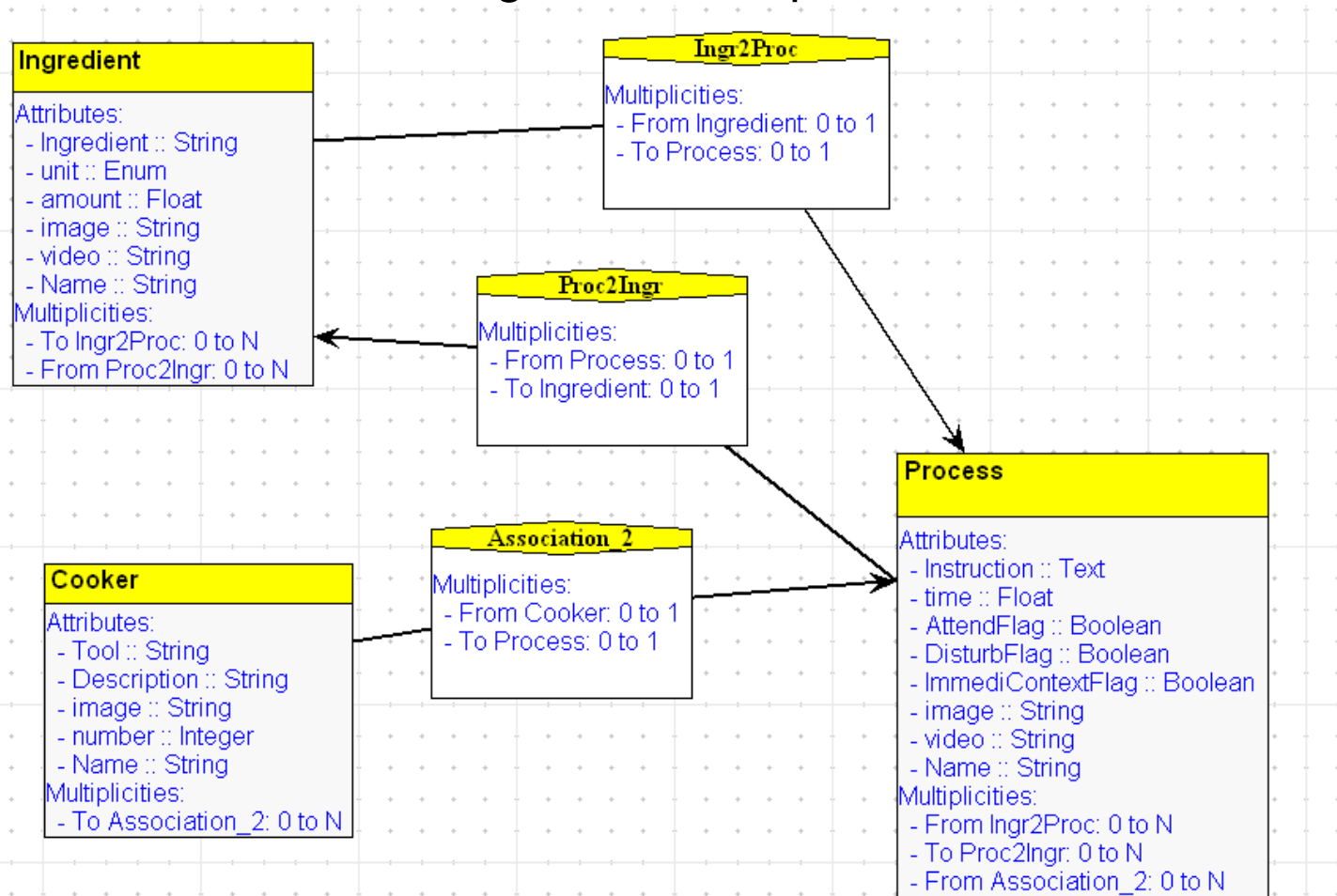
- Cooking conform to many constraints
 - such as the size of the kitchen, the number of cooker, the number of cook and how good the cook is
- What do you expect in a modern recipe?
 - The ingredients and preparation
 - Detailed instructions on every steps
 - These two are what a modern cookbook provides. Any more ideas ?

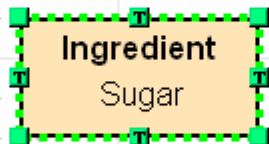
More Characteristic of a Recipe

- What to do first and what second:
 - a reasonable sequence
- Save some effort on calculating, ingredients counting time:
 - automatically scale the amount of ingredients according to the serving
 - provide personalized instructions such as how to make full use of the kitchen
 - a timely reminder through the cooking

Modeling for Cooking Recipes

- Cooking Recipe Meta model
 - Modeled in class diagram
 - Three classes: cooker, ingredient and process





Editing Ingredient [X]

Ingredient

_
 kilogram
 pound
 cup
 tablespoon

unit teaspoon
 ml
 L
 ounce
 gram
 portion

amount

image

video

Name

OK Cancel

Ingredient
egg

Process
Attend True
Disturb True
Contex True

Ingredient

Ingredient
plain flour

Editing Process

Set Spaces Per Tab 4 Set Text Box Height 3

Instruction
fold flour into the mixture

time 5.0

AttendFlag True

DisturbFlag True

ImmediContextFlag False

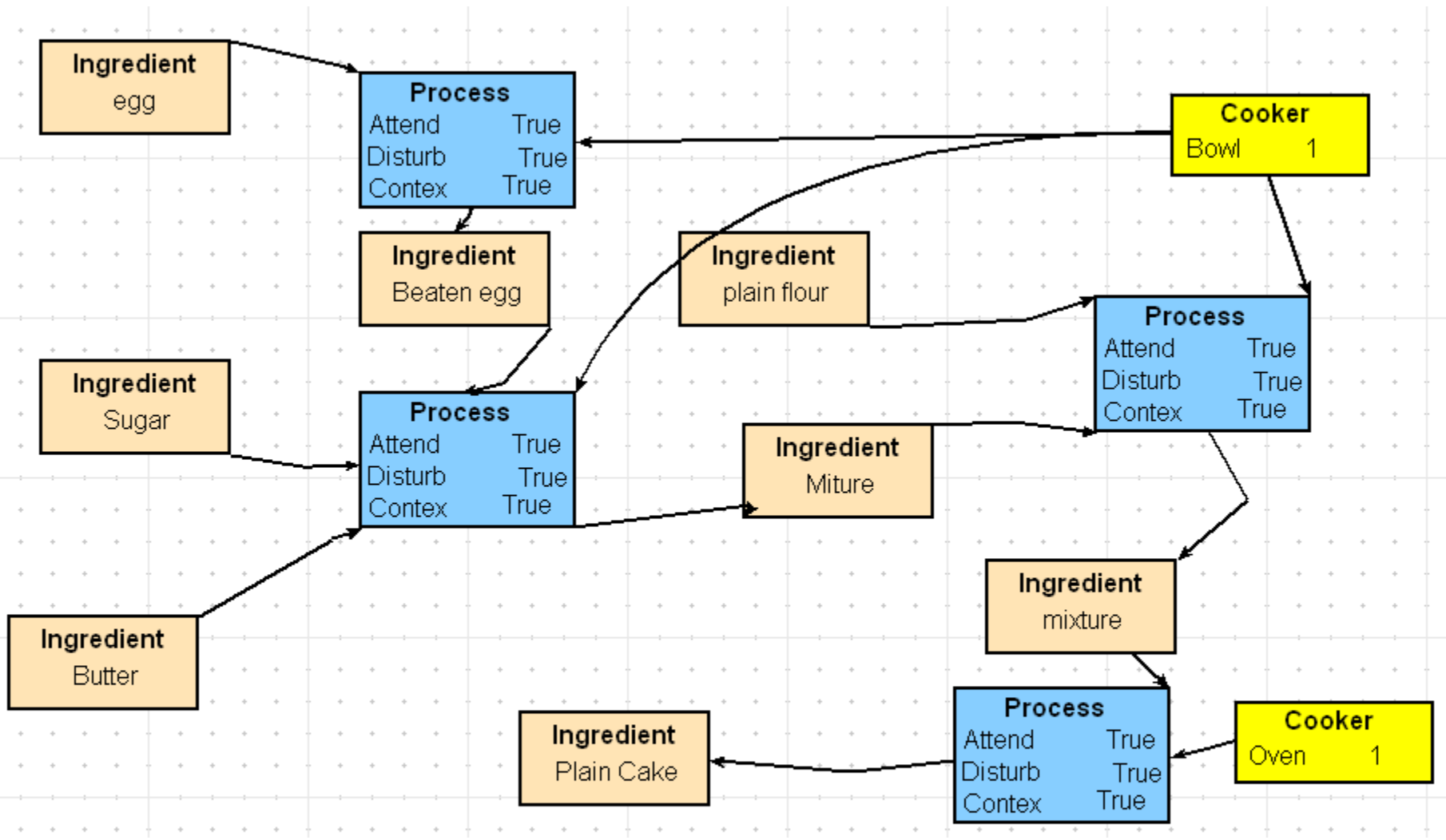
image //

video //

Name Proc_2

OK Cancel

Com



Transform recipe model to textual representation

The following functions can be implemented at this stage

- Generating a textual shopping list
- Generating a textual cooking instruction

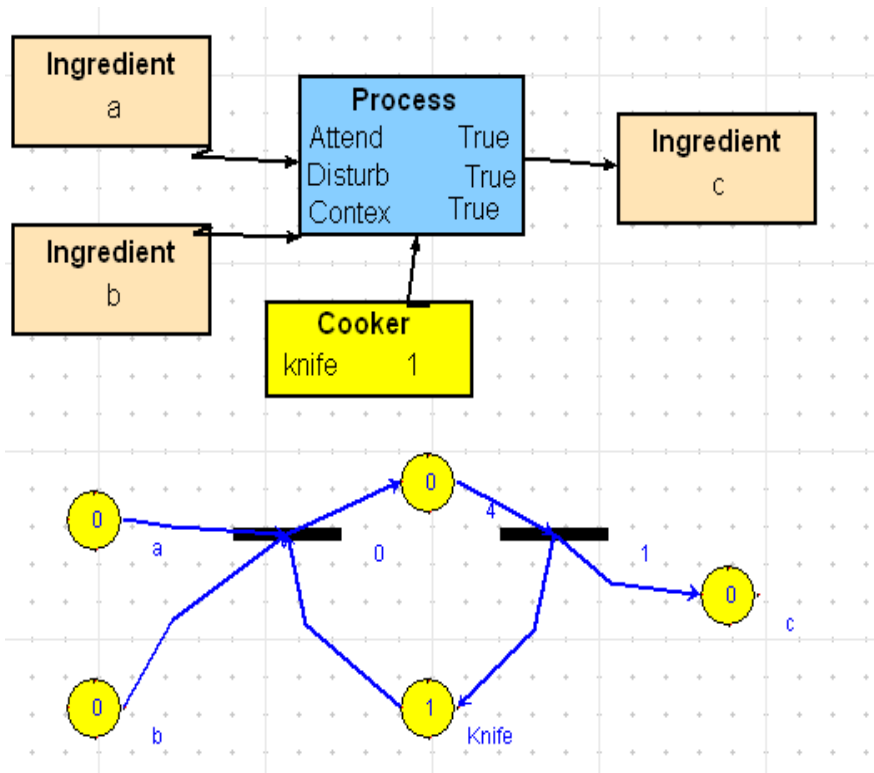
The readers of the recipe can change the number of servings and the amount of ingredients will change.

Here, I make use of the *ASGNode* attributes of in/out connection and *NodeType* in ASG object. By checking the nodes of certain type, those have no input connections are ingredients and the node with no output connections are the dish.

(a example of generated shopping list)

Generate real-time personalised cooking instructor

- The main difficulties towards the realtime and personalization are



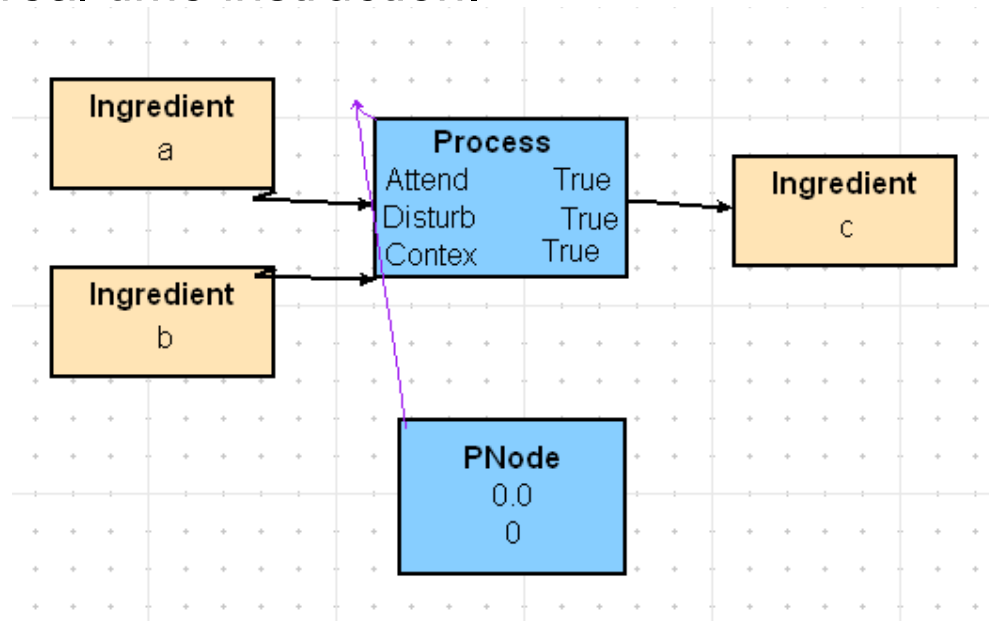
Constraints:

- Constraints are caused by limited resources. The formalism which is good at representing limitations is PetriNets. The resources, from cookers to people, can be represented by tokens. However, it does not have time attributes and as a result, it is not used at last.

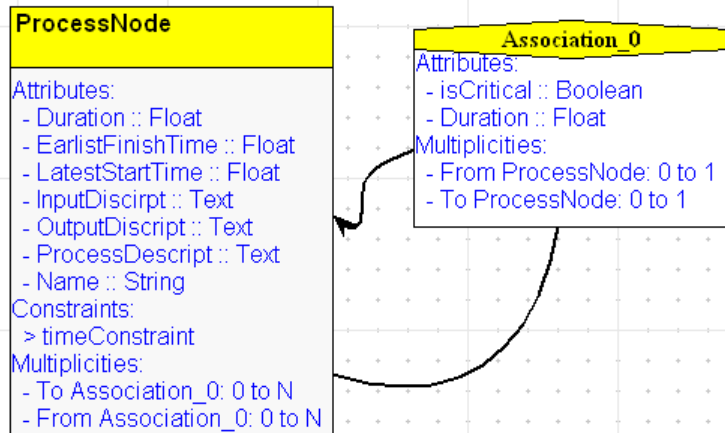
Generating a real-time cooking instructor

– Scheduling:

- it is not easy to design rules to suggesting reasonable cooking process. The cooking is a very flexible process. Many steps can be executed in paralelle, and depending on how fluent the cook is, they are able to be interrupted without causing damages.
- A good choice for this purpose is to use DEVS or StateCharts. They have time attributes which is important for give real-time instruction.



ProcessNode Metamodel



The ProcessNode Metamodel represents the recipe model only use process, which comprises of input, output and other description. By using this metamodel, I redraw the recipe model as a graph with only one class and one association, which makes the future work easier.

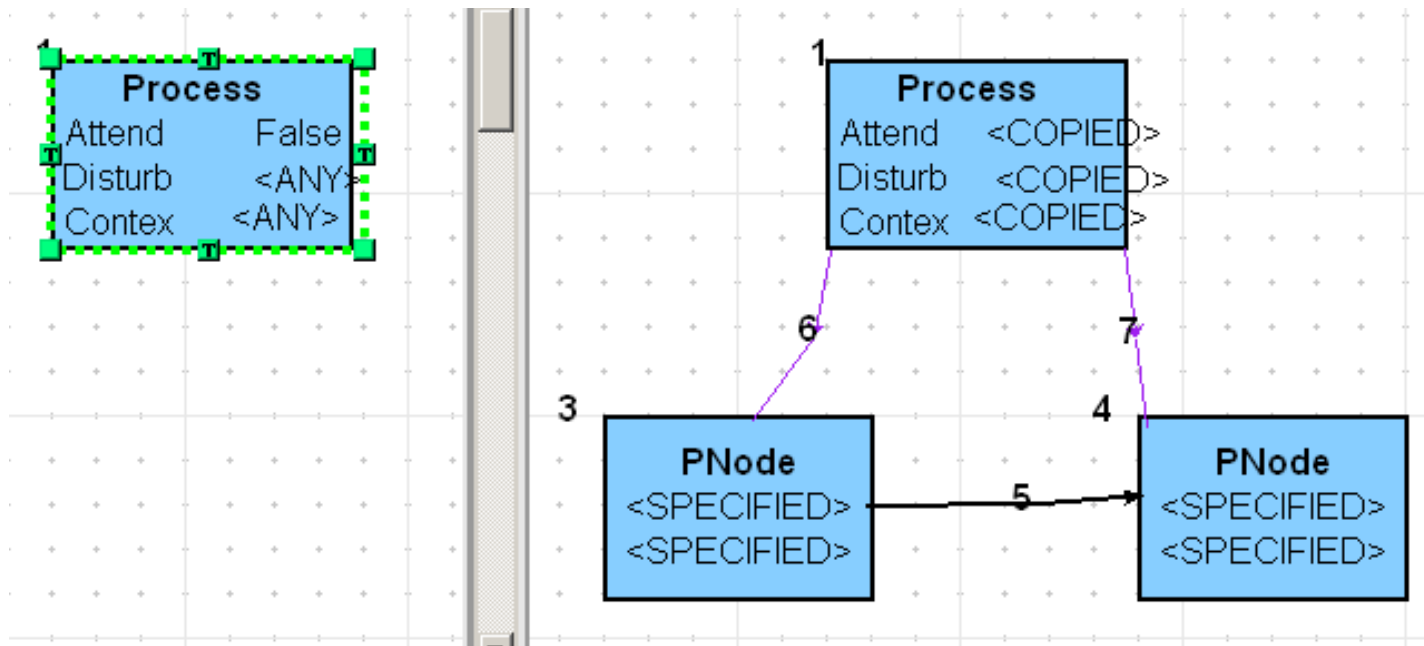
First, transform recipe model into a processNode model.

Second, merge different threads of node into one thread, which is supposed to be easier for users to follow.

Third, transform the thread of nodes into StateCharts to generate code.

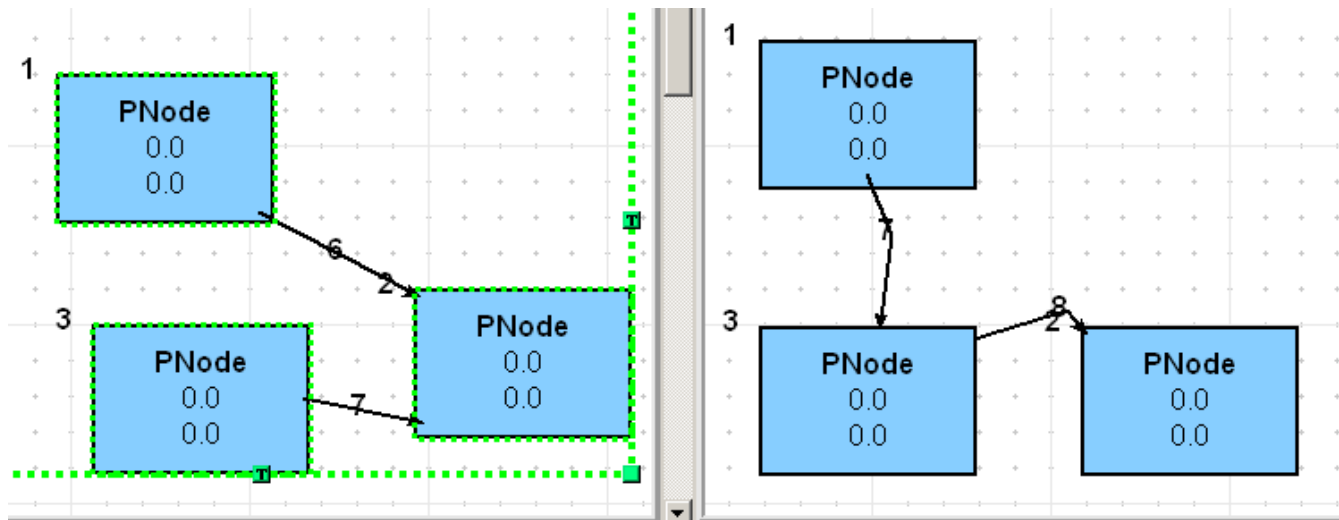
Model Transformation

- As mentioned before, the cooking process is flexible, so I created some detailed rules to cover these situation:
 - The cooking process does not need tendency, and some resources (people) become available.



Merging the Thread

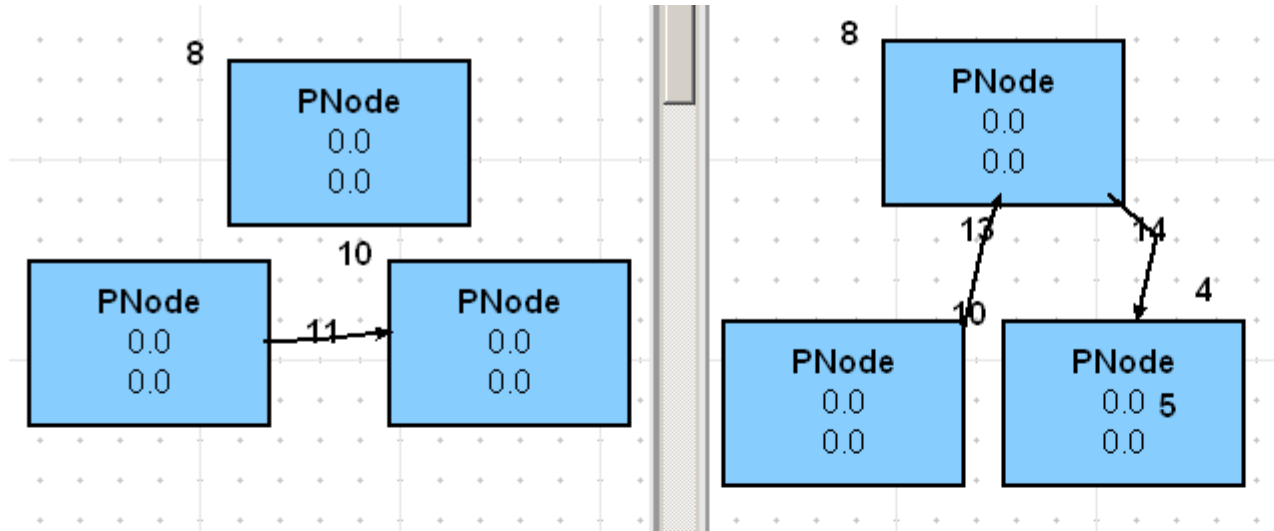
- One cooking process should happen immediately after the other, so no injection is permitted



— .

Merging the Thread

- Some actions are too short and they are able to be inserted in the middle of any process



- .
- .

Conclusions & Further development

- My project is to model for cooking recipe. In this project, one part is to build a modeling environment for cookbook writers; the other part is to design transformations to implement real-time cooking instructor.
- The current formalisms are not totally suitable for this modeling environment, which need both timing and constrains features. It is possible to merge the two kinds of formalism together to get better result.
- The model works well for many cake recipe, but not satisfactory for main courses.

.

Summary

- Some Characteristic of a recipe
- Modeling for cooking recipe
- Use the metamodel to create a recipe
- Transform recipe model to various forms
 - Generating a textual shopping list
 - Generating a textual cooking instruction
 - Generating a real-time cooking instructor
- Conclusions & Further development

TBC>>>>>

- Thanks!

Using the example provided before.

Results, Modeling Environment Appearance

AToM3 v0.3 using: RecipeClassDiagram_META

RecipeClassDiagram_META

EDIT ? Ingredient Process Tool GEN GEN GEN

Ingredient

Shopping List

Plain Cake

- beat the egg for 5 mins
- Fold Sugar and Butter and beaten egg for 5 mins
- fold flour into the mixture for 5 min3
- Bake at 350F for 30 mins

Ingredient

Su

Using the example provided before.

Results1, Textual Representation of the Shopping List

Shopping List

Plain Cake

- This recipe is for serving of 12



6.0 _ egg



6.0 cup plain flour

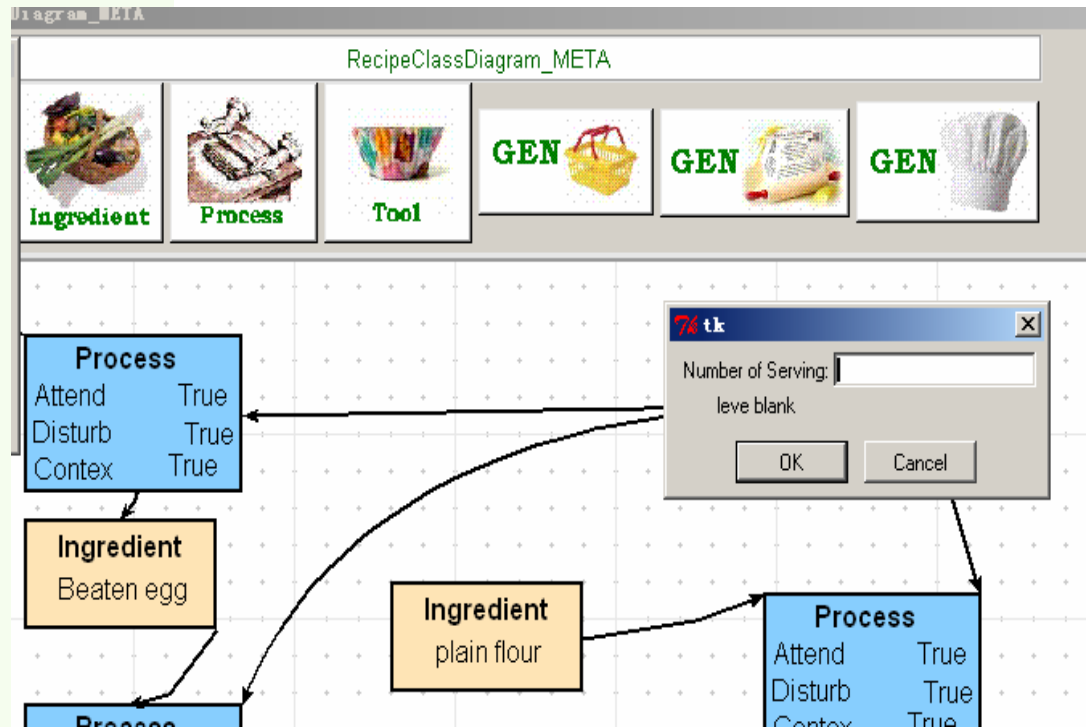


3.0 cup Butter



3.0 cup Sugar

- You also need the following tools,
- 1.0 big bowl
- 1.0 oven



Shopping list generated from the modeling environment

Using the example provided before.

Results2, Textual Representation of the Cooking Instruction

Shopping List

Plain Cake

-

beat the egg for 5 mins



Fold Sugar and Butter and beaten egg for 5 mins



fold flour into the mixture for 5 min3



Bake at 350F for 30 mins



Cooking Instruction

Using the example provided before.

Results3, Real-time Cooking Reminder

Hi, this is what you gonna do!

Sugar, butter, Beaten Egg

Fold Sugar and Butter and the beaten egg



Tips

- You also need the following tools,
Big bowl

This .html file is refreshed when they content need to change. The system generate a sequence of .html files. The pages will navigate from the live_instruction_0.html until the end of all the process(the last html file).

Some further nodes

- Changes before hand in:
- I choose to not using state charts to generate code because the in-efficiency of the transformation. Instead I used the function call at the end of the processNode merging transformation. This function updates the ,html source file.