Using abstraction mechanisms such as classification in *e*^{3}*value*

Background

The *e*^{3}*value* methodology is an approach for understanding networks of enterprises creating, distributing and consuming things of economic value. The methodology includes an ontology for representing *e*^{3}*value* models, as well as software tool support for analyzing such models. For more information, see [1,2] and the master-level course e-Business Innovation.

Problem

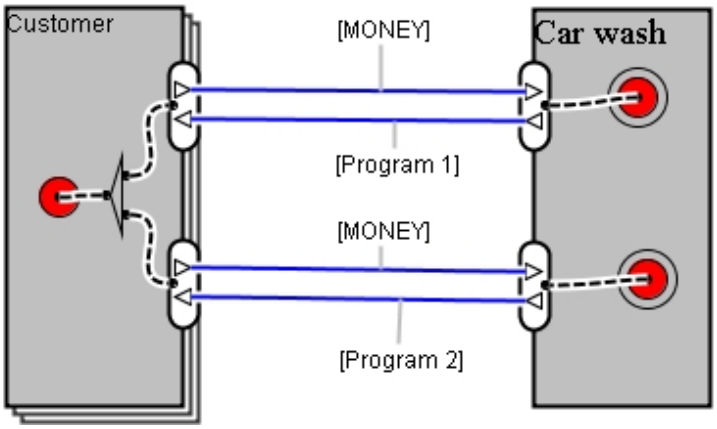

Figure 1: Each program as a separate value object

In figure 1, we show an example in which customers want to use a car wash. Typically, a car wash has several *programs* a customer can choose from, with differences prices. In *e*^{3}*value*, this can be modeled as in figure 1: a customer selects the program and depending on his choice either program 1 or program 2 are transfers. For each corresponding money transfer we can set a different valuation function, so we can calculate the money flow associated with *each* car wash program (as the table below shows: assuming 2 customers, each needing 5 program 1 car washes a Euro 10 and 5 program 2 car washes a Euro 15).

Value Interface	Value Port	Value Transfer	Occurrences	Valuation	Economic Value	Total
{Program 1,MONEY}			10		100	
	out: Program 1	(all transfers)	10	0	0	
	in: MONEY	MONEY	10	10	100	
{Program 2,MONEY}			10		150	
	out: Program 2	(all transfers)	10	0	0	
	in: MONEY	MONEY	10	15	150	
INVESTMENT					0	
EXPENSES					0	
total for actor						250

However (for a full scale case), this requires to split up all programs in separate value objects (here program 1 and program 2). This result is complex value models. As an alternative, consider figure 2.

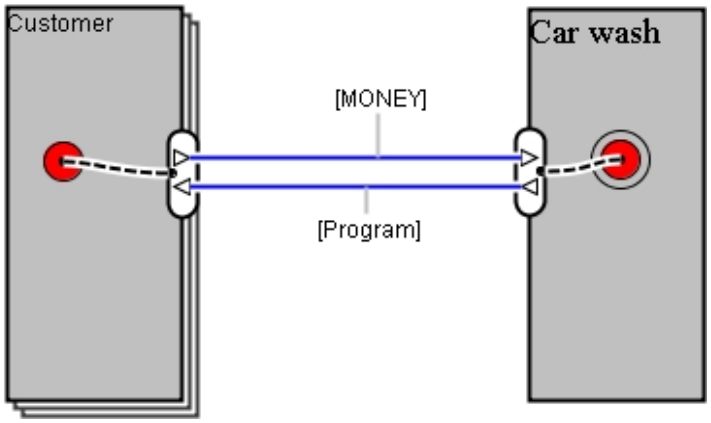

Figure 2: A class of programs (Program) representing program 1, program 2, etc.

Here we have introduced 'Program' as a class of value objects / programs representing program 1, program 2, etc. Note that currently, *e*^{3}*value* contains no such thing as a class. Although this is much easier to represent than figure 1, we can not see any difference anymore on program 1, program 2, etc. (see table below).

Value Interface	Value Port	Value Transfer	Occurrences	Valuation	Economic Value	Total
{MONEY,Program}			20		250	
	in: MONEY	MONEY	20	12,5	250	
	out: Program	(all transfers)	20	0	0	
INVESTMENT					0	
EXPENSES					0	
total for actor						250

The goal of this assignment is to explore a solution direction that allows us to:

- use mechanisms like classification to simplify *e*^{3}*value* model,
- while still be capable of showing value flow sheets in which the various instances (here: program 1, program 2) are shown.
-

Requirements

- good understanding of *e*^{3}*value*, preferably you have at least followed the course e-Business Innovation;
- good conceptual modeling knowledge;

Organization

This is a VU/UTwente-internal assignment. You will be supervised by Jaap Gordijn (VU) and Lianne Bodenstaff (UTwente). Participation in the Greeting research meetings is compulsory.

References

[1] J. Gordijn and J.M. Akkermans, "Value based requirements engineering: Exploring innovative e-commerce idea", Requirements Engineering Journal, Springer Verlag, Vol. 8, Nr. 2, pp 114-134, 2003

[2] J. Gordijn and J.M. Akkermans, "e3-value: Design and Evaluation of e-Business Models", IEEE Intelligent Systems, special issue on e-business, Vol. 16, Nr. 4, pp 11-17, 2001.