Πανεπιστήμιο Κρήτης
University of Crete

ΣΧΟΛΗ ΘΕΤΙΚΩΝ & ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

SCHOOL OF SCIENCE & ENGINEERING
COMPUTER SCIENCE DEPARTMENT

Tranformation and Modeling of a Value Network into Business Processes.

Goggolidis Vaggelis, March 2008
(goggol@tsl.gr)

Supervisor: Christos Nikolaou
(nikolau@tsl.gr)

Keywords: Value Network, Business Process Management, WBI Modeller
Technical Areas: E-commerce Technologies

Abstract

The concept of value creation due to business relationships among organizations has changed the way economic structure is developed. Organizations form networks and interact with each other in order to capture and share value. This report presents the work done to develop the Value Network Tool that implements a first attempt to develop and validate a methodology for transforming a value network into a business process model.

1. Introduction

Value Networks (VNs) comprise business entities working together to co-produce and share value. Business entities might be individuals, firms, government agencies, or any organization of people and technologies. Several studies focus on creating and reconfiguring service value networks (see Ref. [1-4]).

In this report, we address the currently existing gap between service value networks from one side and service system implementations on the other side. VNT (Ref. [5]) has been created in order to visualize, represent and analyze service value networks. We focus on the connection of VNT with the WBI Modeler in order to improve the performance of the value network using the detailed in terms of knowledge business process model. WBI modeler (Ref. [6]) is a tool that represents and simulates business process models. We use the representation of the VN as given by the VNT tool (Ref. [7,8]) in order to create an xml file according to the definitions of the data defined in the xsd schema of the WBI modeler. This xml file contains information about the business process model. We then import the xml file to the WBI modeler in order to extract the business process model. (Note that the xml file is available for import only in Modeler 6.0 or greater.)

2. Making a Business Process out of a Value Network

A value network consists of a number of participants that interact with each other in order to create value. It is important for a business analyst to understand how value is created, how value is shared among the participants and how value is increased or optimized. The main problem encountered is that the value network realizes value exchanges but does not show how they are created. Thus, a mapping of the VN is needed in order to assign participant roles and describe the processes that result in the creation of value. In this report, we propose a framework that transforms the abstract VN into a detailed web of business processes and is implemented by the VNT.

In our approach a participant of the network is an organization in the business process model. In the VN level the participants interact with each other and each interaction is mapped into business activities in the business process level.

In order to perform the transformation the user has to define paths and the sequence of each participant within these paths. A path is a collection of interconnected participants with successive numbers. Each participant may be part of more than one paths. Each path is represented by a new process and each participant is represented by a new sub-process in the WBI Modeler. A process includes the exchange of money or offerings of the whole path and is refined to a number of a sub-processes that include activities in order to give or acquire the selected offering. In particular, if the connection is a two-way connection, we create the sub processes in order to represent the ordering, the acquiring, the pay-back and the acceptance of the pay-back procedure of a value network. We assign sub processes to each organization in the business process level. A sub-process may be assigned a business item that represents an offering or money. Thus, we have taken a path of interconnected participants and transformed it into a business process.

In the special case where a participant has employees and equipment then we can add these characteristics to increase the functionality. We create employees and add costs to them. The employees are created for each participant from the extra resources table in VNT that each participant has. The equivalent element of an employee in the WBI Modeler is a definition of a person resource. Furthermore, we can add one time costs and costs per time the participant has to pay to the employee. The description of how to define employees and costs will be described in the example in section 4.

In the same way, we can define equipment and add costs to it. The equipment is created for each participant from the extra resources table in VNT that each participant has. The equivalent element of equipment in the WBI Modeler is a definition of an equipment resource. Furthermore, we can add the one time cost, the cost per quantity and the cost per quantity and time that the participant has to pay in order to buy or use the equipment. The description of how to define equipment and costs will also be described in section 4.

We have also developed a process named Template that includes all processes in order to have the overall transformation of a value network into a business process model.

3. Class Diagram

The package named wbiModellerPackage has been developed in order to implement the transformation described above. It contains 4 classes that are used to create the required elements. Each class contains methods that perform the required functionality. The classes and methods are presented below.

1. public **CreateAll**(java.lang.String filename);

   This class is initially called with given Object a String that contains the desired name of the xml file that will be imported into Modeler. The methods included in CreateAll are the following:

   - public java.io.File **getFile**();
     This method returns a File pointer to the file that is being processed for import to the Modeler.
   - public java.lang.String **getFileName**();
     This method returns the name of the xml file.
   - public void **CreateFile**();
     This method contains the xml code that is required to be firstly written to the file in order to be a valid Business Process import.
   - public void **CreateBussItem**(int argNum, java.lang.String... args);
     This method is called in order to create a Business Item. ArgNum is the number of arguments of the class and args can be all the requested arguments. With this method we can create any type of business item, with all the given types, with as much properties as we want.

- private void **BussItemCode** (int argNum,
  java.io.BufferedWriter writer,
  java.lang.String... args);
  This method is called by the **CreateBussItem** class and writes to the file all the required code to be a valid Business Item import. The given arguments are a writer to the xml file while the other two are the same as the previous method.

- public void **CreateOrganizationUnit**(java.lang.String name);
  This method creates a new Organization in Modeler. The given argument is the desired name of the organization.

- public void **OrganizationUnitCode**(java.lang.String name,
  java.io.BufferedWriter writer);
  This method is called by the class **CreateOrganizationUnit** and writes to the xml file all the required code to be a valid Organization Unit import. Name is the name of the organization and writer is a writer to the xml file.

2. public **WbiProcess**(java.io.File nm, java.lang.String Nm);
   This is the second main class of the package. In this class we define a new process with name Nm, in the xml file nm. The methods included in WbiProcess are the following:

- public void **CreateProcess**();
  This method finds the correct positioning of the process declaration in the xml file. Both xml name and process name are stored from the call of the constructor.

- private void **ProcessCode**(java.lang.String ProcName,
  java.io.BufferedWriter writer);
  This method writes all the required code in the xml file in order to be a valid process import. ProcName is the name of the process while writer is the writer for the file.

- public void **CreateTask**(java.lang.String ProcName,
  java.lang.String TaskName);
  This method creates a Task. The ProcName is the Process that we wish to add to the task while TaskName is the required name of the task. This method is not used in the current version of the VPN.

- private void **TaskCode**(java.lang.String TaskName,
  java.io.BufferedWriter writer);
  This method writes to the xml file the required code in order to be a valid xml file. TaskName is the name of the task and writer is the writer of the file. This

method is not used in the current version of the VPN.

- public void **CreateSubProcess**(java.lang.String ParName,
  java.lang.String SubName,
  java.lang.String OrgName);
  This method creates a SubProcess. ParName is the name of the process that we want to create the SubProcess, SubName is the name of the SubProcess and OrgName is the name of the organization that the SubProcess belongs.

- public void **SubProcessCode**(java.lang.String ProcName,
  java.io.BufferedWriter writer,
  java.lang.String OrgName);
  This method writes all the required code to the xml file in order to be a valid SubProcess import. ProcName is the name of the SubProcess, writer is the writer to the xml file and OrgName is the name of the Organization that the SubProcess belongs.

- public void **CreateTaskConnection**(java.lang.String ProcName,
  java.lang.String From,
  java.lang.String To);
  This method creates a connection between tasks, subprocesses of both. ProcName is the name of the process that contains the two partners, From is the name of the first partner and To is the name of the second partner.

- private void **ConnectionTaskCode**(java.lang.String From,
  java.lang.String To,
  java.io.BufferedWriter writer);
  This method writes all the required code to the xml file in order to be a valid Connection import. From is the name of the first partner, To is the name of the second partner and writer is the writer to the xml file.

- public void **ItemExchange**(java.lang.String busItem,
  java.lang.String From,
  java.lang.String To,
  java.lang.String Proc);
  This method adds an item over a connection that is exchanged between 2 partners. busItem is the name of the item(must be an already created business item), From the name of the first partner, To the name of the second partner and Proc the name of the process that contains all the requested items.

3. public **PersonResource**(java.lang.String nm,
   java.lang.String fname,
   java.io.File fl);

This class creates a person resource. Nm is the name of the type of the person, fname is the name of the xml file and fl is a File pointer to that file. The methods included in PersonResource are the following:

- public void **CreationCode**();
  This method writes all the required code to the xml file in order to be a valid Person Resource import. It must be called after the call of the constructor so as to find the correct position of the declaration.

- private void **PersonResourceCode**(java.lang.String name,
  java.io.BufferedWriter writer);
  This method writes code to the xml file in order to be a valid Person Resource import. Argument name is the type of the resource and writer is a writer to the xml file.

- public void **AddOneTimeCost**(java.lang.String currency,
  double cost);
  This method adds One Time cost at the requested PersonResource object. Argument currency is the desired currency and cost is the required cost.

- public void **AddCostPerTime**(java.lang.String TmUnit,
  java.lang.String currency,
  double cost);
  This method adds Cost per Time at the requested PersonResource object. Argument TmUnit is the desired time unit, currency is the desired currency and cost is the required cost

4. public **EquipmentResource**(java.lang.String nm,
   java.lang.String fname,
   java.io.File fl);
   This class creates an equipment resource. Nm is the name of the type of the equipment, fname is the name of the xml file and fl is a File pointer to that file. The methods included in EquipmentResource are the following:

   - public void **CreationCode**();
     This method writes all the required code to the xml file in order to be a valid Equipment Resource import. It must be called after the call of the constructor in order to find the correct position of the declaration.

   - public void **EquipmentResourceCode**(java.lang.String name,
     java.io.BufferedWriter writer);
     This method writes code to the xml file in order to be a valid Equipment Resource import. Argument name is the type of the resource and writer is a writer to the xml file.

   - public void **AddOneTimeCost**(java.lang.String currency,
     double cost);
     This method adds One Time cost at the requested EquipmentResource

object. Argument currency is the desired currency and cost is the required cost.

- public void **AddCostPerQuantity**(java.lang.String currency,
  double cost);
  This method adds Cost per Quantity at the requested EquipmentResource object. Argument currency is the desired currency and cost is the required cost.

- public void **AddCostPerQuantityAndTime**(java.lang.String TmUnit,
  java.lang.String currency,
  double cost);
  This method adds Cost per Quantity and Time at the requested EquipmentResource object. Argument TmUnit is the desired time unit, currency is the desired currency and cost is the required cost.

4. A Case Study

In this section we describe an example and show how a value network is transformed into a business process model. Fig. 1 shows the network of Carpenter created by VNT. The network consists of a Wood Merchant, a Carpenter, a Technician and a Customer. The Customer contacts the Carpenter in order to buy a bookshelf. The Carpenter buys raw material from the Wood Merchant. The Technician produces the bookshelf for the Carpenter. Fig, 1 shows the relationships among the participants and the offerings (services, products, money) that are exchanged.

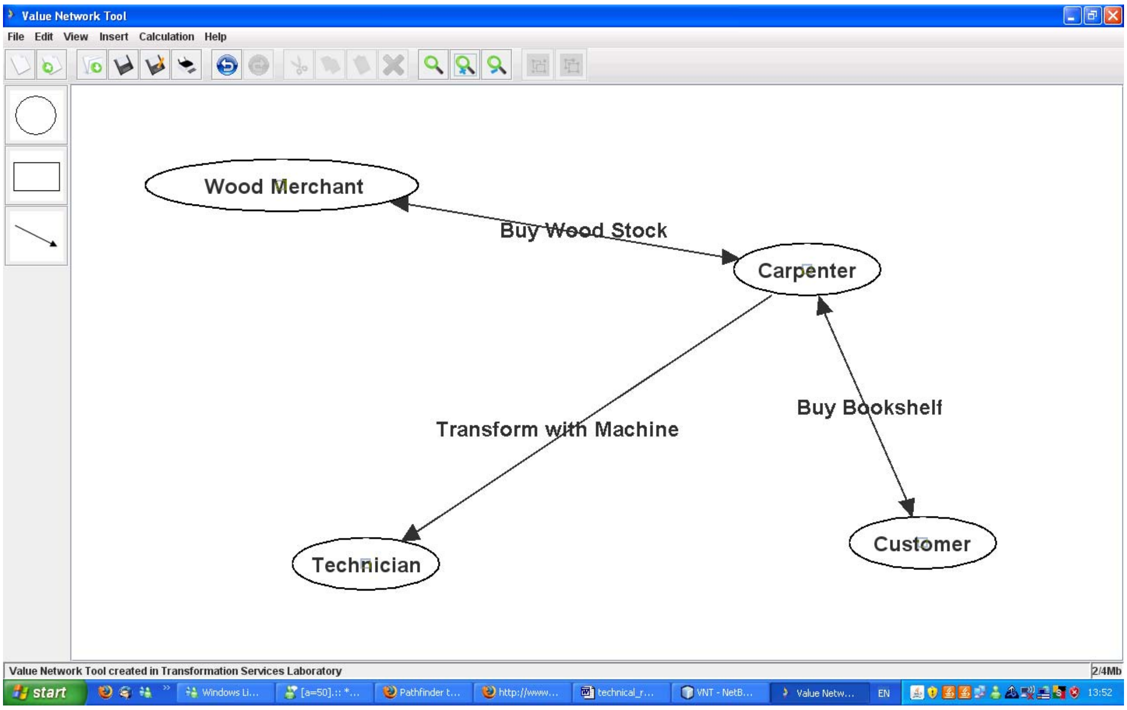
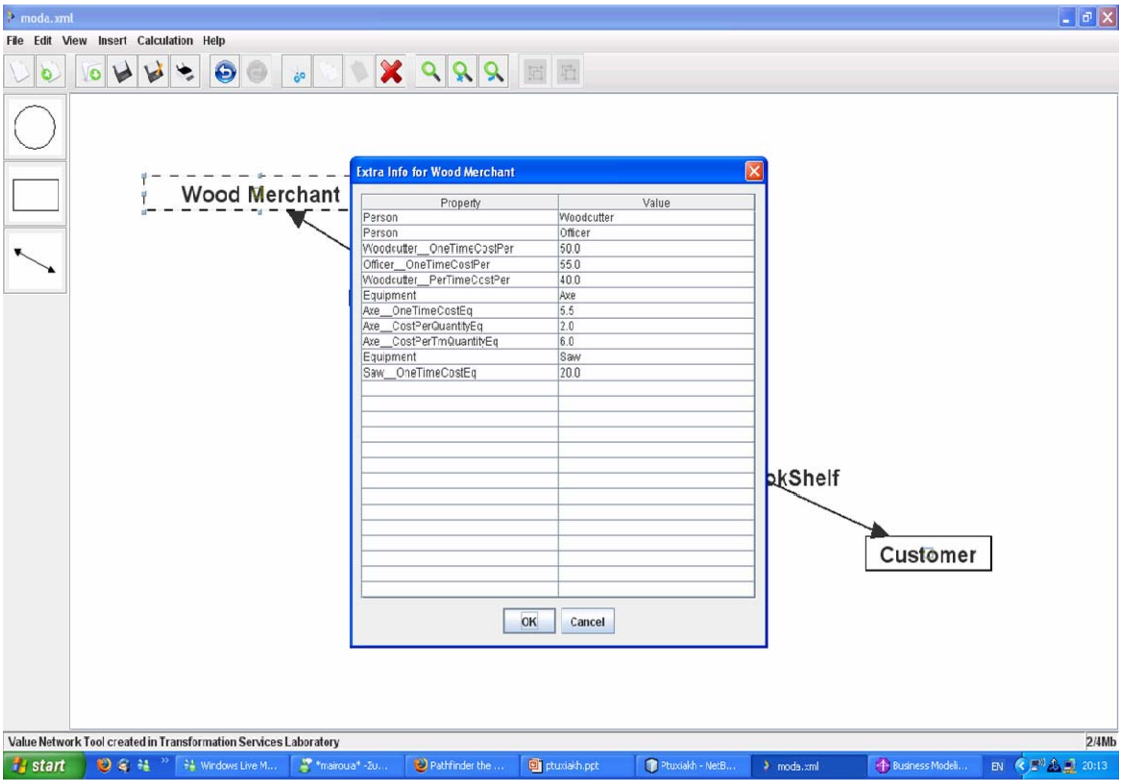


Figure 1: The Carpenter Value Network.

Notice that each participant may have its own equipment and human resources that are combined with input offerings in order to produce output offerings. We insert properties for each participant using the menu extra properties of VNT. In this example we define the resources and costs represented in Fig. 2. We define for the participant Wood Merchant two types of individual resources, Woodcutter and Officer, and two types of bulk resources, Axe and Saw. We also define costs for each resource. For the Woodcutter we define both *One Time Cost* and *Per Time Cost* while for the Officer we define only *One Time Cost.* For the Axe we define *One Time Cost, Per Quantity Cost and Cost per Time and Quantity* while for the Saw we define only *One Time Cost.*



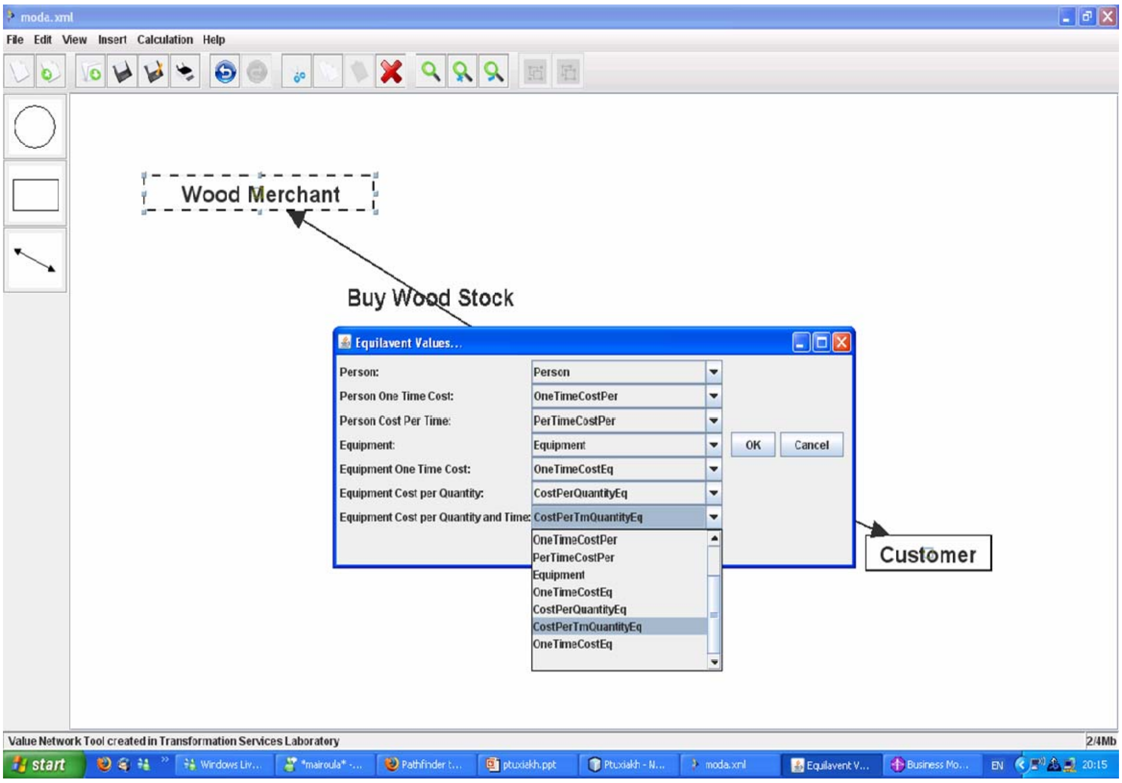
Extra Info for Wood Merchant

Property	Value
Person	Woodcutter
Person	Officer
Woodcutter__OneTimeCostPer	50.0
Officer__OneTimeCostPer	55.0
Woodcutter__PerTimeCostPer	40.0
Equipment	Axe
Axe__OneTimeCostEq	5.5
Axe__CostPerQuantityEq	2.0
Axe__CostPerTmQuantityEq	6.0
Equipment	Saw
Saw__OneTimeCostEq	20.0

Value Network Tool created in Transformation Services Laboratory

Figure 2: Definition of resources and their costs.

The next step for the transformation of this value network into a business process model is to use the Export for WBI Modeler function of VNT to create the xml file. We choose a name for the xml file and save it. Then, a JDialogWindow is opened asking if there are any resources to import. Those resources are the resources we defined in the extra properties column. Selecting yes, will provide us the menu shown in Fig. 3. In that menu, we have to select the equivalent definition of each type we defined in each extra property column. We have to use the same definition-mark for every similar type of property in each extra column of a participant. In case we did not define something, we can leave the definition empty.



Person: Person
Person One Time Cost: OneTimeCostPer
Person Cost Per Time: PerTimeCostPer
Equipment: Equipment
Equipment One Time Cost: OneTimeCostEq
Equipment Cost per Quantity: CostPerQuantityEq
Equipment Cost per Quantity and Time: CostPerTmQuantityEq

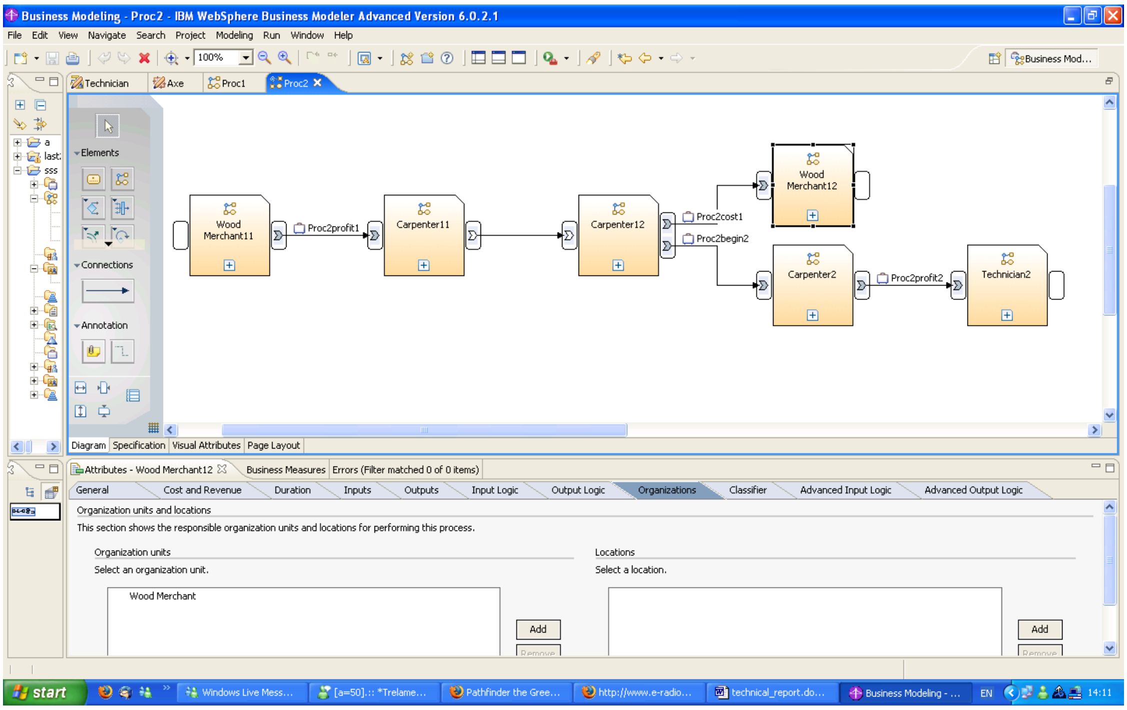
OneTimeCostPer
PerTimeCostPer
Equipment
OneTimeCostEq
CostPerQuantityEq
CostPerTmQuantityEq
OneTimeCostEq

Value Network Tool created in Transformation Services Laboratory
2/4Mb

Figure 3: Defining equivalent definitions of individual and bulk resources.

Then, we import manually the created XML file to a project of WBI Modeler. When the import is finished, we will be able to see the transformation of the value network into business processes. For each path a new process is created. In Fig. 5, the transformation of the relation between the Wood Merchant, the Carpenter and the Customer can be seen. The created sub processes represent all the procedures described above. The business items represent the goods or money that are sent from a participant to another participant. Each sub process belongs to an organization as we can see in the bottom of the screenshot (Fig. 4). In Fig. 5, we can also see the transformation of the second path of the Value Network.

In Fig. 6, the created resources and their costs are shown. In the menu resources in the left hand side, we see the Axe, the Saw, the Woodcutter and the Officer definition. In the right hand side we see all the costs defined for the bulk resource Axe.



Organization units and locations

This section shows the responsible organization units and locations for performing this process.

Organization units

Select an organization unit.

Wood Merchant

Locations

Select a location.

Figure 4: Modeling first path of an exchange between participants.

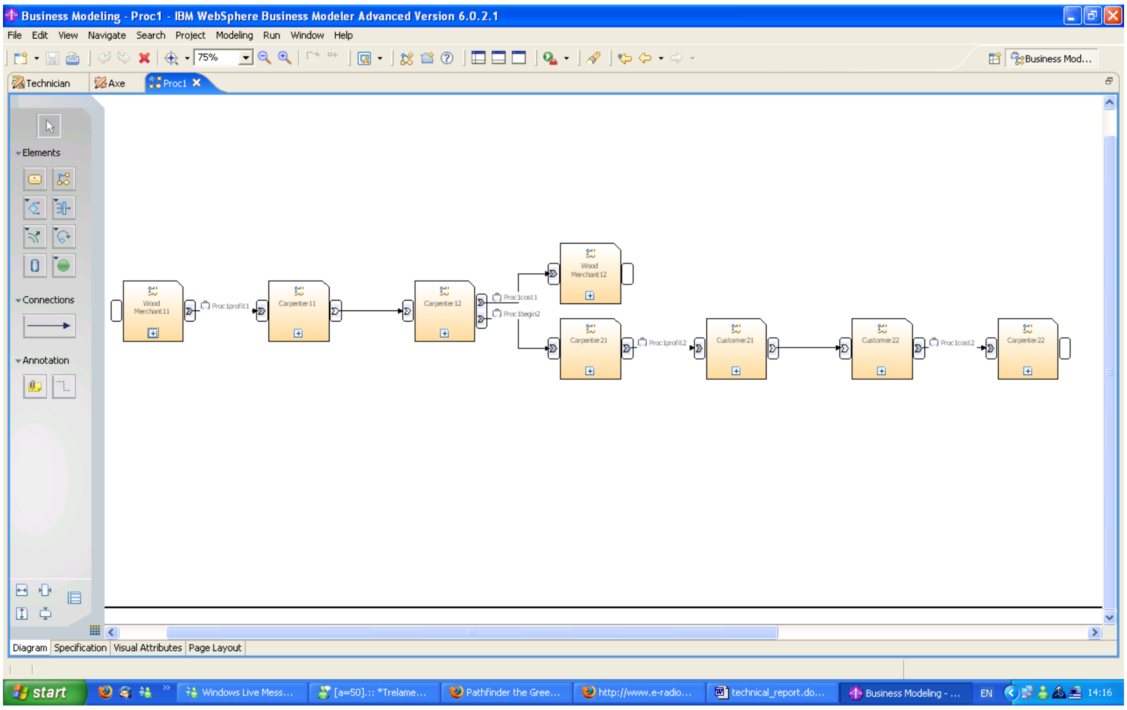
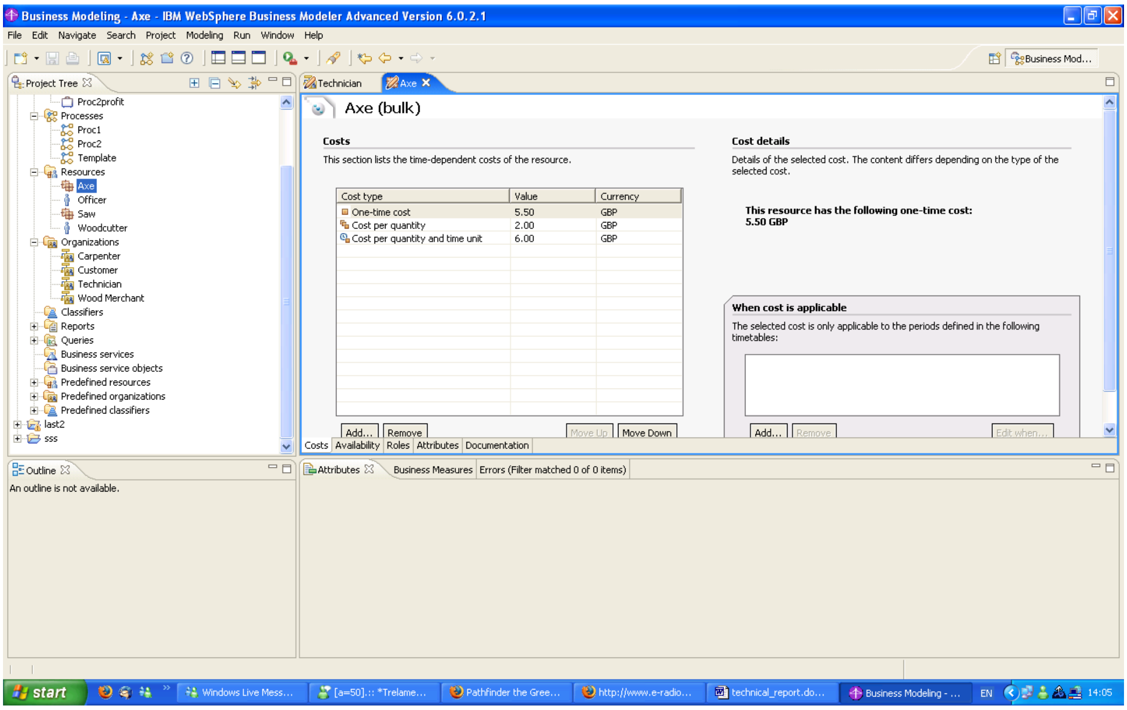


Figure 5: Modeling second path of exchange between participants.



Axe (bulk)

Costs

This section lists the time-dependent costs of the resource.

Cost type	Value	Currency
One-time cost	5.50	GBP
Cost per quantity	2.00	GBP
Cost per quantity and time unit	6.00	GBP

Cost details

Details of the selected cost. The content differs depending on the type of the selected cost.

This resource has the following one-time cost: 5.50 GBP

When cost is applicable

The selected cost is only applicable to the periods defined in the following timetables:

An outline is not available.

Figure 6: Imported costs from a Value Network.

5. Future Work

We have used VNT to model and transform a value network into a business process model. As a future work we will make additional modifications to the modeler package to increase functionality. In the case that the xml schema of the WBI Modeler changes, the modifications can be easily adopted because of the division of the class between positioning and source code.

An extra column could be added in the extra info list for each participant. That will result in making declarations of properties easier. A drop down list that will contain all the available declaration will allow the user to know what is available for declaration and hence they will not have to use the "__" to distinguish resources from their costs.

Another useful addition is to distinguish the difference between Tasks and Sub Processes. The code that creates Tasks is ready but it is not used and we can also define Tasks in Sub Processes dynamically. So defining the difference between Tasks and Sub Processes will result in a better representation of a value network in a business process model.

Finally, we should add a definition of exchanging items in the value network as we do not have enough information now about the exchanged items, apart from their costs and values. The relationships between participants are not all the same and we should realize that difference also from the exchanged items. These items then will be the created business items in the business process.

References

1. N. Caswell, C. Nikolaou, J. Sairamesh, M. Bitsaki, G. D. Koutras and G. Iacovidis. "Estimating Value in Service Systems – A theory and an example". IBM Systems Journal, VOL 47, NO 1, 2008.
2. J. Spohrer, P. P. Maglio, J. Bailey, and D. Gruhl, “Steps Toward a Science of Service Systems,” IEEE Computer 40, No. 1, 71–77 (2007).
3. Allee Verna, “Reconfiguring the Value Network”, Journal of Business Strategy, Vol. 21, N 4, July-Aug 2000, also at http://www.vernaallee.com/value_networks/Reconfiguring_the_Value_Network.pdf
4. Jaap Gordijn, Hans Akkermans, “Designing and evaluating e-business models”, in IEEE Intelligent Systems, Vol. 16(4):11-17, 2001.
5. http://vnt.tsl.gr/
6. http://www-306.ibm.com/software/integration/wbimodeler/index.html
7. Papathanasiou V. “ Υλοποίηση Πλατφόρμας μοντελοποίησης, προσομοίωσης και αναπαράστασης δικτύων αξίας”. Technical Report, University of Crete.
8. Zarras A. “Value Network Tool: A Tool for Modeling, Simulating and Representating Value Networks”. Technical Report, University of Crete.
9. Xalkiadaki M. “ Υπολογισμός και Γραφικές Παραστάσεις Αξιών στο Εργαλείο Δικτυών Αξίας”. Technical Report, University of Crete.