ENTERPRISE ONTOLOGY
DEVELOPMENT BASED ON
DEMO METHOD

Notice of Originality
I declare that this paper is my own work and that information derived from published or unpublished work of others has been acknowledged in the text and has been explicitly referred to in the list of references. All citations are in the text between quotation marks (“”). I am fully aware that violation of these rules can have severe consequences for my study at Utrecht University.

Signed : A.D.Nagarajan  
Name : Anitha Devi Nagarajan

Date : April 25, 2016  
Place : Utrecht
Table of Contents

1.0 INTRODUCTION
  1.1 ABOUT THE ORIGINATOR OF DEMO  3
  1.2 UNDERSTANDING THE ENTERPRISES WITH DEMO  3

2.0 EXAMPLE  4

3.0 META-MODEL OF DEMO  6
  3.1 ENTERPRISE ONTOLOGY  6
    3.1.1 CONSTRUCTION MODEL  7
    3.1.2 PROCESS MODEL  7
    3.1.3 ACTION MODEL  7
    3.1.4 STATE MODEL  7
  3.2 PROCESS DELIVERABLE DIAGRAM (PDD)  7
    3.2.1 ACTIVITY TABLE  9
    3.2.2 CONCEPT TABLE  10

4.0 RELATED LITERATURE  12

5.0 REFERENCES  14
1.0 Introduction

Enterprise Engineering (EE) was defined as, “body of knowledge, principles, and practices having to do with the analysis, design, implementation and operation of an enterprise” (Liles, Johnson & Meade, 1995, p.45). There are several researchers investigating on the methods and models of EE. This paper will be focusing on one such EE method called Design and Engineering Methodology for Organizations (DEMO). Albani and Dietz (2011) described that DEMO is the method for modeling, redesigning and reengineering business processes of an organization by helping to understand the notion of business processes in it. DEMO is very helpful for the Information Architects, Managers and Enterprise Architects or similar roles to obtain the broad overview as well as deep insight on the construction and operations of the organization (Dietz, 2006b).

1.1 About the Originator of DEMO

DEMO method was developed by Dr. Jan L.G. Dietz, who is a professor of the Information Systems Design at the Delft University of Technology in The Netherlands. His research and educational interests are in modeling, (re)designing and (re)engineering organizations with the support of advanced ICT- applications. He has published over 200 scientific and professional papers as well as several books. He is a co-founder and chairman of the Enterprise Engineering Institute (http://www.ee-institute.org/en). To promote his ideas on enterprise engineering, he has set up an international community CIAO! Enterprise Engineering Network (http://ciaonetwork.org) (Jan Dietz, n.d.).

1.2 Understanding the enterprises with DEMO

According to Dietz (2006b; 2008a) a DEMO transaction is the basic element of a business process in DEMO method. As it is seen in Figure 1, a DEMO transaction is a pattern of two kind of acts performed by the social individuals (also called as subjects or actors), who are actually the active elements of an organization. Those two kinds of acts are namely: Production acts or P-acts and Coordination acts or C-acts.

![Figure 1 Basic DEMO transaction pattern as imitated from Hoogevorst and Dietz (2008d)](image)

As the names imply, production acts will produce something, which is either material (goods) or immaterial (services) and on the other hand, coordination act will be an act of compliance. The
outcome of any successful P-act will be called Production fact (P-fact) and the similar one for C-act will be called Coordination fact (C-fact). As mentioned earlier, these acts will occur in a generic coordination pattern called transaction and the actors involving in these transactions are called initiator and executor.

Any demo transaction occurs in three phases namely, Order Phase (O-Phase), Execution Phase (E-Phase) and Result Phase (R-Phase) (Dietz, 2006b). Because of this fact, these are also called as OER-patterns (Dietz, 1999). In the O-Phase, both the partaking subjects (initiator and executor) will come to an agreement about the P-fact to be produced. The corresponding C-acts in this phase are request and promise. In the E-Phase, the executor will produce the P-fact and so in the R-Phase, again both the actors will negotiate and settle with each other on delivered P-fact (Dietz, 2006b). These transaction patterns encircled in each other to form the business processes of the organizations.

As witnessed in Figure 2, DEMO considers the organization as an eco-system comprising the actors from Coordination world (C-world) and Production world (P-world). C-world actors have the responsibility to perform C-acts and hence yield in C-facts whereas, P-world actors have the competence to do P-acts that produces P-facts (Dietz, 2008a).

2.0 Example

In this section, the DEMO way of modeling of an organization has been illustrated with a speculative example of a take-away fast food shop called ‘Burger Queen’. Let us assume that Burger Queen is run by two people, whereas one (hereafter called ‘processor’) processes the orders from a customer and the other one (hereafter called as ‘chef’) makes the burgers. Every customer who comes to the shop will provide the order to the processor and he will inform the chef to prepare the burger. Once the burger is ready, the chef hands it over to the processor and the processor hands it over to the ordered customer.

For the above scenario of the fast food shop, let us try to construct the process step diagram. First, there are three actors identified from the scenario as given below.

Actor A0: Customer; Actor A1: Processor; Actor A2: Chef.

For every burger ordering process, there are two transactions involved as following.

Transaction T1: Between A0 and A1 – Customer (A0) placing the order to the processor (A1)

Transaction T2: Between A1 and A2 – Processor (A1) placing the order to the chef (A2)

Based on the above information about actors and transactions involved, the actor transaction diagram can be found in Figure 3.
In Figure 3, the link between the A0 and T1 is called the information link and the one between T1 and A1 is called the executor link.

Figure 3 Actor Transaction Diagram of Burger Queen

Figure 4 shows the Process Step Diagram of Burger Queen, which details the transaction flow among the actors of the organization. Following steps explain the transaction flow shown in Figure 4. The encapsulation of the second transaction within the first one has been justified with different format of numbering used.

Step 1. The Actor A0 (Customer) requests a burger (T1 in state ‘Request’) to Actor A1 (Processor).

Step 2. Once the order was accepted by the processor, the T1 moves to state ‘Promise’.
   a. Then the second transaction T2 is coming into place, when A1 places the burger order to A2 (Chef).
   b. Here the Transaction T2 is in state ‘Request’ and it moves to state ‘Promise’ when the request was accepted.
   c. Then the chef makes the burger (which is the P-fact of T2) and gives it to A1 so that T2 enters into ‘state’ T2.
   d. Once the burger was taken by the processor the Transaction T2 moves to the state called ‘accept’ and completes there.

Step 3. The processor takes the burger (now it is the P-fact of T1) and gives it to customer so that T1 enters into ‘state’.

Step 4. Once the customer accepts the burger, the transaction T1 enters into ‘accept’ and completes at that point.

From Figure 4, it can be seen that there is an intermediate flow (called as conditional link) which is not including the transaction T2. This will be taken for example, when the customer orders a burger and if it is not available in the shop, the transaction T1 will be completed without involving the chef (and so no T2).
Let us assume that the Burger Queen is being expanded with home delivery option to the customer and so there is one more employee included in the organization to perform the burger delivery to the customers. In this situation, the actor transaction diagram will be expanded to include the new actor (A3 – Deliverer) and transaction (between A1 and A3) as shown in Figure 5.

![Figure 5 Extended Actor Transaction Diagram of Burger Queen](image)

The Process Step Diagram can be modified accordingly in order to show the complete transaction flow for the extended Burger Queen and it is not shown here due to space limitations.

### 3.0 Meta-Model of DEMO

In this section, a high level overview of the enterprise ontology construction process using DEMO method is depicted through meta-modeling technique called ‘Process Deliverable Diagram’, which was introduced by Weerd and Brinkkemper (2008). PDD provides a compact and standardized view of both involved activities and obtained deliverables with the help its two models namely Meta-Process model and Meta-Data model. Meta Process Model will be on the left hand side of the PDD and it will describe the activities and sub-activities to be carried out in the method involved. Meta Data Model will be on the right hand side of the PDD and it depicts the corresponding deliverables/concepts which are obtained as an output from the activities involved. At the end, a concept table and an activity table is included in order to explain the corresponding items. These tables are further explained clearly below when they are brought up.

### 3.1 Enterprise Ontology

‘Enterprise Ontology is defined as the implementation independent understanding of the operations of organizations, where organizations are systems in the category of social systems.’ (Geskus and Dietz, 2009, p. 133). Albani and Dietz (2011) described that an enterprise ontology is a business domain model with the characteristics namely coherence, consistency, comprehension, conciseness and essentiality. The ontological model constructed with DEMO satisfies these characteristics completely. As per DEMO, the entire ontological model of an enterprise comprises four aspect models i.e. Constructional Model (CM), Process Model (PM), State Model (SM) and Action Model (AM) as seen in Figure 6.
3.1.1 Construction Model

The Construction Model of an organization details the model of its construction and it consists of internal and external actor roles of an organization, the transaction and information links between the actor roles. This model is represented with *Object Construction Diagram, Transaction Result Table* and *Bank Contents Table*.

![Diagram of the four aspects ontological model](image)

*Figure 6 The four aspects ontological model as adopted from Albani & Dietz (2011)*

3.1.2 Process Model

The Process Model of an organization specifies all the identified transactions from CM with the help of the DEMO transaction pattern explained earlier. This model also details the relationships among the transactions. These tree structure of enclosed transactions define the business processes of the corresponding enterprise. This PM model is represented by *Process Step Diagram* and may be complemented with an *Information Use Table*.

3.1.3 Action Model

The Action Model of an organization consists of a set of action rules which specify the production and/or coordination acts that must be performed and the production and/or coordination facts whose presence in the state of the world must be assessed. This model is represented in *Action Rule Specifications* document.

3.1.4 State Model

The State Model describes the fact types, object classes and ontological coexistence rules in the production world. This model is represented using *Object Fact Diagram* and possibly complemented by *Object Property Table*.

3.2 Process Deliverable Diagram (PDD)

The overall process deliverable diagram given in Figure 7 has been constructed based on the steps described by Albani, Dietz & Zaha (2006c). Figure 8 depicts the PDD constructed for the sub-activity ‘Create actor transaction diagram’.

Please note that the role names are not mentioned since the enterprise ontology will be developed by DEMO specialists (internal or external) and the situation may be very different for every organization.
In the overall PDD, most of the sub-activities within phases are depicted as ‘closed complex activities’ since all the detailed steps of performing those activities are not found in the identified literature documents and so those are not depicted elsewhere. In the ‘Create actor transaction diagram’ PDD, the sub-activities are kept as unordered since those can be carried out in any sequence.

Figure 7 PDD of the Enterprise Ontology Development Method with DEMO
3.2.1 Activity Table

The activities and sub-activities identified from the left-hand side of both the PDDs are shown in Table 1. Each of the activity is associated with the corresponding sub-activities and the relevant description of those sub-activities. Please note that the activities are presented in the same sequence as given in the PDDs.

Table 1 Activity Table

<table>
<thead>
<tr>
<th>Activity</th>
<th>Sub-activity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Construction</td>
<td>Create actor transaction diagram</td>
<td>This sub-activity is to create the ACTOR TRANSACTION DIAGRAM after identifying the actors and transactions within the organization.</td>
</tr>
<tr>
<td>Model</td>
<td>Create actor bank diagram</td>
<td>Production banks are conceptual stores of production facts and in the same way, Coordination banks are conceptual stores of coordination facts. This sub-activity will be expanding the ACTOR TRANSACTION DIAGRAM by including the ACTOR BANK DIAGRAM in it. Hence the resulting complete diagram will be the ORGANISATION CONSTRUCTION DIAGRAM.</td>
</tr>
<tr>
<td></td>
<td>Derive transaction result table</td>
<td>This sub-activity will construct the TRANSACTION RESULT TABLE along with the ACTOR TRANSACTION DIAGRAM.</td>
</tr>
<tr>
<td></td>
<td>Build bank contents table</td>
<td>This sub-activity will construct the BANK CONTENTS TABLE which is corresponding to the ACTOR BANK DIAGRAM.</td>
</tr>
<tr>
<td>Process</td>
<td>Construct process step diagram</td>
<td>This sub-activity will construct the PROCESS STEP DIAGRAM based on the basic transaction pattern of DEMO. The basic transaction pattern consists of three phases namely Order Phase, Execute Phase and Result Phase. Also it is multiple states called request, promise, execute, state, accept etc.</td>
</tr>
<tr>
<td>Model</td>
<td>Create</td>
<td>This activity is related to the transaction processes however this</td>
</tr>
</tbody>
</table>
Design and Engineering Methodology for Organizations

<table>
<thead>
<tr>
<th>Action Model</th>
<th>Find action rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information use table</td>
<td>The table will be produced as the last activity after the complete ORGANISATION CONSTRUCTION DIAGRAM is obtained.</td>
</tr>
<tr>
<td>Action Model</td>
<td>Find action rules</td>
</tr>
<tr>
<td>Find action rules</td>
<td>There is an action rule for every agendum kind for every internal actor role. An action rule specifies the (production and/or coordination) acts that must be performed, as well as the facts in the production world and/or the coordination world whose presence or absence in the state of the world must be assessed. This sub-activity will be creating the ACTION RULE SPECIFICATIONS once after the PROCESS STEP DIAGRAM is produced.</td>
</tr>
<tr>
<td>State Model</td>
<td>Draw object fact diagram</td>
</tr>
<tr>
<td>Draw object fact diagram</td>
<td>The OBJECT FACT DIAGRAM will be developed as a result of this sub-activity based on the action model of the scope of interest. If the action model is missing, one has to rely on the available information for determining which fact types are used and/or created.</td>
</tr>
<tr>
<td>State Model</td>
<td>Derive object property table</td>
</tr>
<tr>
<td>Derive object property table</td>
<td>This sub-activity will derive Property type, Object class and scale details from OBJECT FACT DIAGRAM and represent it in the OBJECT PROPERTY TABLE.</td>
</tr>
</tbody>
</table>

3.2.2 Concept Table

Table 2 depicts the concept table of the method that lists all the concepts shown in the right-hand side of the given PDD. As given in the table, each of the concepts is explained with a short description and associated with the corresponding reference to it. Please note that the concepts are ordered in the same way as it is in given in the PDD in order to improve the readability. So the concepts from overall PDD are given at the top, whereas concepts from the second PDD are provided at the later part of the table.

Table 2 Concept Table

<table>
<thead>
<tr>
<th>Concept</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENTERPRISE ONTOLOGY</td>
<td>ENTERPRISE ONTOLOGY is the business domain model of an organization. All the remaining concepts mentioned in this table together constitute the complete ENTERPRISE ONTOLOGY of an organization. This concept in combination with business process components help for the development of information systems (Albani et al., 2006c).</td>
</tr>
<tr>
<td>ORGANISATION CONSTRUCTION DIAGRAM</td>
<td>ORGANISATION CONSTRUCTION DIAGRAM shows the structure and environment of an organization. The ACTOR TRANSACTION DIAGRAM and the ACTOR BANK DIAGRAM together creates the ORGANISATION CONSTRUCTION DIAGRAM (Albani &amp; Dietz, 2011).</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>ACTOR TRANSACTION DIAGRAM</td>
<td>ACTOR TRANSACTION DIAGRAM demonstrates the identified transaction types, which are executed by associated actor roles (Albani &amp; Dietz, 2011).</td>
</tr>
<tr>
<td>ACTOR BANK DIAGRAM</td>
<td>ACTOR BANK DIAGRAM diagram shows the information links from actor roles to production and coordination banks, which are the conceptual stores of production and coordination facts respectively. This will be drawn as an extension of the ACTOR TRANSACTION DIAGRAM (Albani &amp; Dietz, 2011).</td>
</tr>
<tr>
<td>TRANSACTION RESULT TABLE</td>
<td>The TRANSACTION RESULT TABLE is a table of transaction kinds and the specification of resulting fact types (Geskus &amp; Dietz, 2009).</td>
</tr>
<tr>
<td>BANK CONTENTS TABLE</td>
<td>The BANK CONTENTS TABLE is a table that shows the fact kinds of which instances are contained in the transaction banks of the listed transaction kinds (Geskus &amp; Dietz, 2009).</td>
</tr>
<tr>
<td>PROCESS STEP DIAGRAM</td>
<td>PROCESS STEP DIAGRAM shows how the distinct transaction types are related. All the transactions are based on the basic transaction pattern of DEMO (Albani &amp; Dietz, 2011).</td>
</tr>
<tr>
<td>INFORMATION USE TABLE</td>
<td>INFORMATION USE TABLE consists of all the process step names and the corresponding shortcuts assigned to it (Albani &amp; Dietz, 2011).</td>
</tr>
<tr>
<td>ACTION RULE SPECIFICATIONS</td>
<td>ACTION RULE SPECIFICATIONS specify the action rules that serve as guidelines for the actors in dealing with their agenda. It contains one or more action rules for every agendum type. These rules are grouped according to the actor roles that are distinguished. The action rules are expressed in a pseudo-algorithmic language, by which an optimal balance is achieved between readability and preciseness (Albani et al., 2006c).</td>
</tr>
<tr>
<td>OBJECT FACT DIAGRAM</td>
<td>OBJECT FACT DIAGRAM specifies all object types and fact types occurring in the action rules from the ACTION RULE SPECIFICATIONS. This diagram is the ontological variant of the Object Role Model diagram (Albani &amp; Dietz, 2011).</td>
</tr>
<tr>
<td>OBJECT PROPERTY TABLE</td>
<td>OBJECT PROPERTY TABLE represents the properties and scale. Properties are binary fact types that happen to be pure mathematical functions, of which the range is a set of, usually ordered, values, called a scale (Albani &amp; Dietz, 2011).</td>
</tr>
<tr>
<td>ACTOR</td>
<td>ACTORS are the subject roles of the organization who is having a specific responsibility to perform one or more of the organizational activities. These ACTOR roles can be either of type ELEMENTARY ACTOR ROLE or COMPOSITE ACTOR ROLE (Dietz, 2008a).</td>
</tr>
</tbody>
</table>
ELEMENTARY ACTOR ROLE

This is one of the types of ACTOR roles and it represents the internal actor roles which present within the ORGANIZATION BOUNDARY (Albani & Dietz, 2011).

COMPOSITE ACTOR ROLE

COMPOSITE ACTOR ROLE is a type of ACTOR role which is involved in the business processes of the organization but is not present within the ORGANIZATION BOUNDARY (Albani & Dietz, 2011).

ORGANIZATION BOUNDARY

ORGANIZATION BOUNDARY concept defines the bounds of each of the enterprises involved in an organization, which is considered to have one or more enterprises (Albani & Dietz, 2011).

TRANSACTION

TRANSACTIONS are the communication between two ACTORs in the organizational process (Dietz, 2006a).

### 4.0 Related Literature

As mentioned earlier, DEMO method has been developed by Prof. Dr. Jan L. G. Dietz during the 1980’s (Bobbert, 2009). The theoretical basis of this method was drawn upon three scientific literatures namely Habermas’ Communicative Action Theory, Stamper’s Semiotic Ladder and Bunge’s Ontology (Dietz, 1999). To be specific, the DEMO transaction pattern which was introduced in the introduction section of this report is based on Habermas’ Theory of Communicative Action (Dietz, 2006b), which is one of the two base theories of Language Action Perspective (Dietz, 1999). Both Habermas’ theory and DEMO method are based on Ψ- theory (ψ is pronounced as PSI, standing for Performance in Social Interaction), which is classified as an Ontological Theory (Dietz, 2013; Dietz, 2006a).

DEMO method has been compared with other similar enterprise engineering methods in order to gain more insight. As an example, Ettema and Dietz (2009) have performed a theoretical and practical comparative analysis between DEMO and ArchiMate, which is an enterprise architecture method. It was concluded that these methods are following completely different approaches and so not comparable to each other. However the authors have indicated that these methods can be combined so that ArchiMate can include re-engineering projects in its scope.

Since DEMO method is yielding high level ontological models, various case studies with very different domains have been found. Huysmans, Ven and Verelst (2010) have used the DEMO method for Open Source Software Development (OSSD) process modeling. They mentioned that the obtained DEMO models were high in quality as well as in abstraction level, thereby complexity was very much reduced. They stated that the ontological modeling and transaction patterns are the unique features of DEMO method and they make it suitable for OSSD process models. Albani and Dietz (2011) have applied DEMO method to the Strategic Supply Network Development (SSND) domain to derive the business models. They mentioned that these business models helped to design the information models, which contain the crucial features and so well explainable to business people of the organization.

DEMO can be combined with other techniques or methods in order to meet the various needs of specific projects or an organization. For example, Thuan, Dietz and Tran (2010) have explained how the DEMO models can be combined with the Rapid Application Development (RAD) Techniques in the software development process. In this case, DEMO can be used to capture the business processes whereas RAD technique will link the identified business processes to the software development concepts. By this way, the business functions can be mapped to the requirements for the system to be built which will help to increase the understanding and communication of the system.
Design and Engineering Methodology for Organizations

a different study (Barijs, Chong, Dietz and Liu, 2002), DEMO has been extended with a semiotic method called NAM for norm analysis. DEMO has been used to capture the business rules and processes whereas the norm analysis helped to formalize the rules with associating the responsible roles and also to capture the exceptions.
5.0 References


