2015

An Agent Based Approach for Simulating DEMO Enterprise Models

Mamadou Seck
Old Dominion University

Joseph Barjis

Repository Citation

Seck, Mamadou and Barjis, Joseph, "An Agent Based Approach for Simulating DEMO Enterprise Models" (2015). *Engineering Management & Systems Engineering Faculty Publications*. 13.
http://digitalcommons.odu.edu/emse_fac_pubs/13

Original Publication Citation

Seck, M., & Barjis, J. (2015). An agent based approach for simulating demo enterprise models. *Procedia Computer Science, 61*, 246-253. doi:10.1016/j.procs.2015.09.206

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 61 (2015) 246 – 253

Procedia Computer Science

Complex Adaptive Systems, Publication 5
Cihan H. Dagli, Editor in Chief
Conference Organized by Missouri University of Science and Technology
2015-San Jose, CA

An Agent Based Approach for Simulating DEMO Enterprise Models

Mamadou Seck[1], Joseph Barjis[2]

[1]Department of Engineering Management and Systems Engineering, 241 Kaufman Hall
Old Dominion University, Norfolk, VA, 23529
[2]Institue of Engineering and Management, The Netherlands

Abstract

DEMO (design and engineering methodology for organization) is a theoretically grounded methodology for rigorous enterprise modeling. It provides relevant concepts from a construction perspective. As the methodology gains a wider audience, there is a growing interest in simulating DEMO models. Most attempts to develop a simulation approach for enterprise models in general, and the DEMO methodology in particular, are based on process oriented conceptualizations that are typically implementation in a discrete event paradigm. In this paper, we propose a conceptual framework for the translation of DEMO models into an agent based simulation. We will describe the DEMO methodology in some detail and present a domain model that translates its most salient concepts in a form that makes it amenable to agent based simulation.

Keywords: DEMO Methodology, ontological modeli, enterprise modeling, agents simulation

1. Introduction

The phenomenon of an enterprise is of a complex system for the fact that it is comprised of both technical and social elements. From the myriad of theories that explain the enterprise phenomenon, we consider the enterprise ontology as best suitable for understanding operation and construction of an enterprise. According to the enterprise ontology (Dietz, 2006), the collective services that an enterprise provides to its environment are called the ‘business’ of the enterprise. This definition represents the functional perspective of an enterprise, which is not concerned with the construction aspects of the enterprise. The collective activities of an enterprise in which these services are delivered, along with the persons that carry out these activities, are called the ‘organization’ of an enterprise.

Enterprise processes are socially dominated as these processes extensively involve human interactions. As (Katzensten & Lerch, 2000) argue, a social context framework, which represents people’s motivations, social

doi:10.1016/j.procs.2015.09.206

relationships, and social constraints, allows analysts to see a richer picture of business processes and relationship between social and technical systems.

In the recent years, with enterprises growing in complexity, a new discipline of enterprise engineering emerged (Dietz *et all*, 2013), where modeling and simulation play important role to analyze and understand enterprise operations (Barjis 2011).

1.1. Enterprise Modeling and Simulation

In any system development, modeling plays a crucial role, especially during the analysis and design phase as it often allows representation of the design subject in a more visualized manner such as intuitive diagrams (Shannon, 1975). Models are used as an effective way of communication among the stakeholders.

One of the challenges of enterprise modeling is the complexity of capturing the entire breath and depth of an enterprise. Therefore, we emphasize that innovative approaches are needed to deal with complexity in a formal manner and make enterprise models intellectually comprehensible. One such a methodology, to be discussed in this section, is DEMO Methodology.

The objective of modeling is creating shared understanding and communicating design ideas and concepts among the stakeholders. Simulation plays a complementary role in this objective for a more advanced and extended study of the models. Examples of such advanced study could be analysis of performance of different design scenarios, or investigating a solution space in relation to certain metrics for optimization (Zeigler, Praehofer & Kim, 2000; Balci, 2001). Simulation builds a simplified replica of a system and conduct experiments for the purpose of understanding its behavior, evaluating various design options or strategies within the limits imposed by a criterion or set of criteria for the operation of the system (Shannon, 1975).

The challenge of simulation is not tools however, but to carry out the simulation study in the right way. Only a conceptually well-designed model will result in a simulation study that yields success. However, despite its importance for simulation success, conceptual modeling is poorly studied and understood (Brooks, 2006). Conceptual modeling is still largely a matter of creativity and multiple interpretations of the problem domain.

We believe that a profound enterprise conceptual model can be obtained based on solid theoretical notions, adoption of a suitable paradigm, and taking the right worldview. In this paper, although we will frequently refer to conceptual model as an ontological model, but the underpinning is the same. We refer to the discussed models in this paper as ontological model because the derived models are based on enterprise ontology.

2. DEMO Methodology

DEMO stands for ‘design and engineering methodology for organization’. This methodology is based on the ontological notion of an enterprise. Enterprise ontology is concerned with the study of the operation of an enterprise and its construction (Dietz, 2006).

According to enterprise ontology, an enterprise is a social system with social individuals its elements.

Social actors in organization perform two kinds of acts: *production act* (P-acts, for short) and *coordination act*s (C-acts, for short). By engaging in P-acts, the actors bring about new results or facts, e.g., they deliver service or produce goods. Examples of P-acts are: produce a new product; sign a new contract; make a payment; judge or decide. By engaging in C-acts, the actors enter into communication, negotiation, or commitment towards each other. These two types of acts constitute an ontological transaction (see Figure 1). Below follows more explanation of what is an ontological transaction. For more reading, the interested reader is referred to (Barjis, 2011).

An ontological transaction is (explicitly or implicitly) a chain of three phases where a series of coordination and production acts take place. The coordination act (or C-Acts) actually repeats in two phases – prior to the production act (or P-Acts) when actors communicate and agree upon some actions and after the production act when the same actors discuss the result and outcome of the production act. Thus, there are three phases that constitute a generic pattern in which a transaction is carried out (see Figure 2):

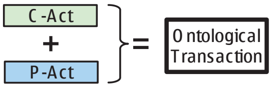

Figure 1: An ontological transaction consists of coordination act (C-Act) and production act (P-Act)

These three phases are indicated as O, E, and R. The three phases of an ontological transaction involve two actor roles. The actor role that initiates a transaction is called *initiator*. The actor role that carries out a production act is called *executor*. The generic process diagram of a transaction is illustrated in Figure 2.

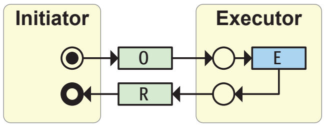

Figure 2: Basic transaction concept

The following discussion clarifies how the actor roles are played and how a social interaction and communication (coordination acts) result in an action (see Figure 3):

- The *Order phase (O)*: an actor, who plays the initiator role, makes a 'request' for a product towards another actor. This phase represents a number of interactions (coordination acts). This phase ends with a 'promise' (commitment) made by the second actor, who will deliver the requested product.
- The *Execution phase (E)*: an actor, who plays the executor role, i.e., 'produce' the product.
- The *Result phase (R)*: the actor, who played the executor role, 'present' the first actor with the product produced. This phase ends with the 'accept' of the product by the first actor, who initiated the transaction.

In Figure 3, the order phase and the result phase are spread over the boundaries of the two involved actors, which implies that these phases are carried out in collaboration (explicitly or implicitly) between the two actors. In contrary, the execution phase merely involves the executor.

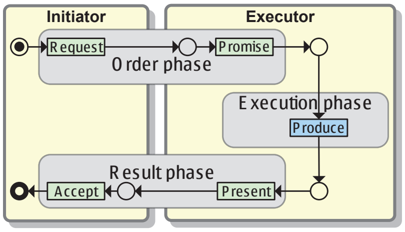

Figure 3: A generic form of ontological transaction

We use a table where all aspects of a transaction are described. For example, in Table 1, the transaction represents an activity of 'designing a new product'. The *initiator* of this transaction is a customer, and the *executor* of this transaction is a designer, and the *result* this transaction creates is the original fact that a new product is designed according to the customer requirements and accepted by the customer.

Table 1: Ontological transaction structured description

Transaction (e.g., T01)	Activity (e.g., designing a new product)
Initiator	Name of the role that initiates the transaction (e.g., *customer)*
Executor	Name of the role that executes the transaction (e.g., *designer)*
Result	The result created as the transaction is carried out (e.g., *a new product is designed)*

3. Agent Based modeling of DEMO Models

To implement the DEMO concepts in the agent based simulation paradigm, we begin with an object oriented conceptualization that captures the most significant domain notions introduced in section 2. In the interest of brevity, however, the conceptualization here presented is abstracted and knowingly omits a number of entities and relationships that are accounted for in DEMO theory. Figure 4 shows a highly simplified view of the proposed domain model.

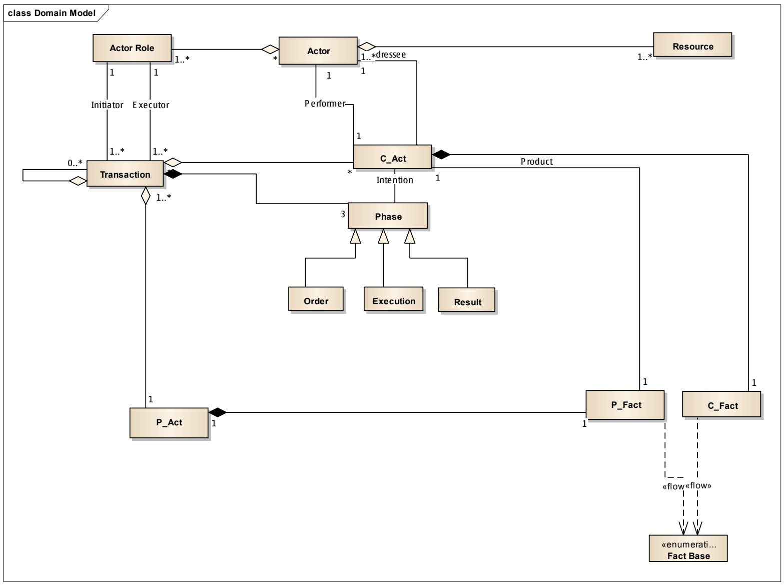

Figure 4: Abstract domain model for DEMO.

The main agent that brings about the behavior in the simulation is the Actor. An Actor can embody various Actor Roles, which specify the competencies and responsibilities that regulate the behavior. An Actor can play different Actor Roles in an organization and conversely, an Actor Role can be embodied by different Actors. All behavior is mediated through Transactions, each of which mobilizes two Actor Roles: the Initiator and the Executor. These are conceptualized as an association that links the Actor Role with the Transaction. A Transaction is a concatenation of acts, the performance of which leads to the observable behavior. These acts are of two sorts, the P-Acts and the C-Acts. The performance of acts brings about new facts, which are again of two types, P-Facts and C-Facts. A C-Act has a Performer and an Addressee, which are both Actors with the appropriate roles, as well as a Product. This Product is a P-Fact. Actors generate the behavior by performing Acts in the sequence prescribed by the transaction pattern. A C-Act also includes an Intention, which is determined by the Phase in which the Transaction is situated: Order, Execution, or Result. P-Facts and C-Facts are stored in a Fact Base. To simulate scarcity or competition for resources, which are responsible for most delays and queuing phenomena in organizations, we also introduce the concept of a Resource, which can be exclusive to a given Actor or shared amongst Actors, with a specified capacity. Both Transactions and Actor Roles can be organized hierarchically in composition relations.

Section 4 introduces a case that exemplifies the DEMO methodology.

4. DEMO Modeling: Patient Examination Process

In the family health care center (FHCC), in order to be examined by a doctor, a patient needs to make an appointment. A patient, upon arrival, signs in on the "Check In" sheet at the front desk and waits in the waiting room to be called. Meanwhile, the front desk forwards the patient's chart and a face-sheet to the nurse's desk so that the first available nurse may deal with the patient.

The nurse calls the patient and conducts preliminary general checkup (blood pressure, EKG, basic lab work) and records chief complaint, and reason for the visit. After completing this preliminary checkup, the nurse escorts the patient to an available examination room and places the chart into the designated box at the door of the examination room. The doctor examines the patient and updates the patients chart if any prescription is issued, diagnosis is made, referral is given, or if any other notes are taken. After completing the examination, the patient goes to the side-desk to check out, to make payment relevant to the service delivered, and, if needed, to make a follow-up appointment.

The FHCC is capable of providing most of the services and treatments a patient may need; however, in rare cases, patients may need further examination by external healthcare providers (sub-specialist) or advanced diagnostic equipment such as a CAT scan, available elsewhere.

In this case, the FHCC, after providing a preliminary diagnosis, schedules an appointment with an external healthcare provider based on the availability of the network provider. Some procedures such as a CAT scan may require the insurance company's pre-approval in which case the FHCC first requests pre-approval and then makes the appointment arrangement.

Transaction Identification:

Now, we identify all essential transactions that constitute the FHCC healthcare processes. The first transaction occurs when a patient makes a new appointment. It involves two actor roles (patient, receptionist), its execution creates a new fact – a new appointment.

Delivering healthcare (examination) to the patient is the second transaction. It starts with a request by the patient and concludes when the patient is examined. However, as we will see later in when constructing a model, this transaction is a composite transaction, that is, this transaction nests the transaction of general check up by a nurse. According to the regulations in FHCC, doctor's examination always implies that the patient should be seen by a nurse for general check up. However, what a patient asks is not the general check up but examination by his/her physician. Therefore, the examination request takes place first and then it leads to the general checkup transaction before the examination transaction completes. In the same manner, we identify all the relevant transactions form the plain description of the previous section.

T1: Appointment scheduling
Initiator: Patient; Executor: Receptionist (FHCC); Result: A new appointment is made

T2: Patient examination
Initiator: Patient; Executor: FHCC (Physician); Result: The patient is examined

T3: General checkup
Initiator: Physician (FHCC); Executor: Nurse (FHCC); Result: The patient general checkup is completed

T4: Patient payment
Initiator: Business office (FHCC); Executor: Patient; Result: The service is paid

T5: Lab works
Initiator: Physician (FHCC); Executor: Technician; Result: Lab works are done

T6: Specialist appointment arrangement
Initiator: Physician (FHCC); Executor: Nurse; Result: An arrangement for appointment is made

T7: Specialist appointment approval
Initiator: Nurse (FHCC); Executor: External specialist; Result: An external appointment is approved

T8: Insurance pre-approval
Initiator: Nurse (FHCC); Executor: Insurance company; Result: Pre-approval is granted

T9: Insurance payment
Initiator: Business office (FHCC); Executor: Insurance company; Result: payment is granted

Now that we have all the essential transactions identified, we move on to model development where all the identified transactions are put in relation to each other as a whole structure.

In this step, we build a conceptual model that should represent all essential transactions of the FHCC.

A complete model of the FHCC is represented in Figure 5.

The model should be read in a left-right and top-down manner. The model illustrates several sub-processes: “appointment process”, “examination process”, “referral process”, “patient payment process”, “insurance payment process”.

Transaction T1 (appointment scheduling) is an atomic process that does not involve any further transactions. Therefore, T1 by itself is a sub-process with start and end points. The result of T1 is a condition for the patient examination sub-process to start, which starts with initiation of T2.

Lets explain the modeling elements used. In the “appointment process” or T1, the process start is represented by a little circle with black dot in it. It proceeds to the order phase of the transaction, T1/O, one the order phase is completed, it proceeds to execution phase, T1/E, then to the result phase, T1/R, and finally it reaches end state, represented by small circle with a hole in it. Also, from this segment one can notice that all actions are represented by boxes , e.g., T1/O, T1/E, T1/R, and logical flow of these actions are represented by arrows connecting these boxes. The circles following each box represent the result of each action or a state in the model.

In a similar fashion, the entire model of Figure 5 can be read. In the model of Figure 5, each sub-process has its starting and ending points, represented by circles with a black dot and hole.

Transaction T2 (patient examination) is a composite transaction that nests several transactions. Therefore, the execution phase of T2 is stretched out to allow seeing how the nested transactions are related to T2. For instance, part of the patient examination (T2) is a general checkup (T3) that should be carried out by a nurse. In some cases, during the patient examination, minor lab works might also take place (T5). Notice, T5 is an optional transaction and therefore it is represented with a diamond symbol in the box that illustrates optionality. With the completion of T3 and (optionally) T5, the patient examination is completed (T2/R).

In Figure 5, the area, labeled with T2/E, represents the actor role ‘physician”. That means that executor of T2 is a physician.

In case, if the patient examination requires, the physician may ask his/her nurse to arrange an appointment with a specialist (T6). Transaction T6 is also a composite transaction and therefore its execution phases is stretched out to illustrate how the nested transactions are executed. If an insurance pre-approval is needed for making an appointment with a specialist, then the nurse contacts the insurance company to get a pre-approval (T8), but pre-approval is not always needed so T8 is an optional transaction. Next, the nurse approves an appointment with a specialist (T7).

There is also a payment process. Portion of the payment, called co-pay, is paid by the patient (T4) and portion of the service is paid by the patient’s insurance company (T9). Since not necessarily all patients are insured, T9 is optional transaction.

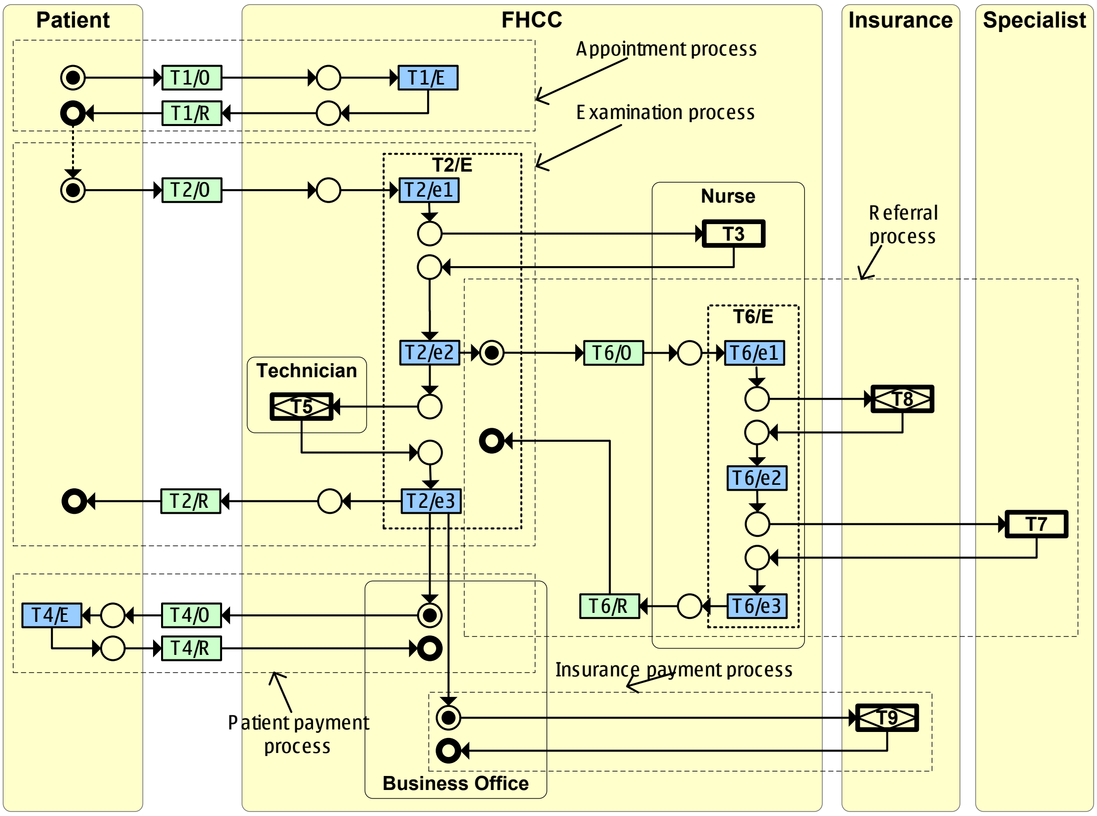

Figure 5: The FHCC DEMO model.

To provide an indication of the implementation, the first 3 transactions (T1, T2, and T3) are translated in the agent based model as depicted in figure 6. Actor roles are shown as square icons. Actors are represented by person icons. Transactions are specified between actor roles and depicted as links that go from the initiator to the executor. Actors are connected to their roles by a link going from actor role to actor.

As the simulation executes, actors create temporary links between one-another, which are instances of the transactions defined between their roles. This creates an adaptive network that continuously evolves until all actors have completed all their transactions. Actors who play executor roles in transactions have links to resources, which may be of finite capacity, and will lead to queuing behavior. The establishment of the inter-actor links and the completion of the acts prescribed by the transactions are controlled by the action rules defined in the netlogo code.

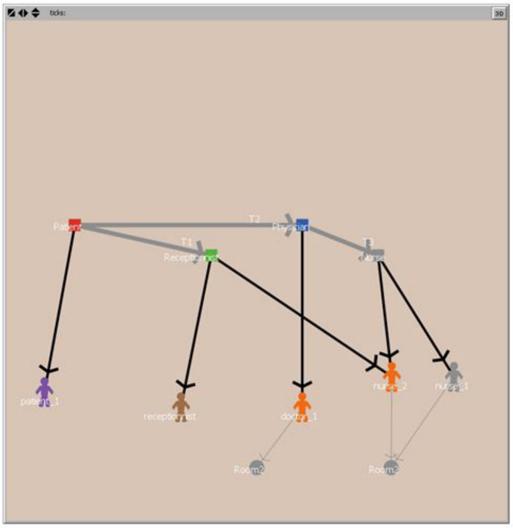

Figure 6: The FHCC DEMO model implemented in netlogo.

5. Conclusion

We have presented the DEMO methodology in some detail and translated some of its most salient concepts into the agent based modeling paradigm. DEMO has benefit of expressing complex organization in a form that retains their essence, while remaining rigorous. DEMO emphasizes the social nature of organizations and adopts a construction perspective. These two features make the agent based modeling approach well adapted for simulating DEMO models. Unlike "process oriented" discrete event languages, an agent based implementation of DEMO retains the social network view and retains the key insight that enterprises are adaptive networks of actors who engage in mutual commitments to bring about results.

Future works include a more complete implementation of the conceptual agent based model, including the other perspectives of the DEMO methodology and the simulation of a wider range of enterprises, to ascertain the applicability of the proposed concept.

References

Balci, O. 2001. A methodology for certification of modeling and simulation applications. ACM Transactions on Modeling and Computer Simulation, Vol. 11, No. 4, pp.352-377

Barjis, J. (2011). Enterprise Modeling and Simulation within Enterprise Engineering. Journal of Enterprise Transformation 1(3), pp. 185-207.

Brooks R. (2006). Some Thoughts on Conceptual Modelling: Performance, Complexity and Simplification. In the Proceedings of the 2006 OR Society Simulation Workshop (28-29 March 2006).

Dietz, J. (2006). Enterprise Ontology - Theory and Methodology. Springer-Verlag Berlin Heidelberg.

Dietz, J.L.G., Hoogervorst, J.A.P., Albani, A., Aveiro, D., Babkin, E., Barjis, J., Caetano, A., Huysmans, P., Iijima, J., van Kervel, S.J.H., Mulder, H., Op 't Land, M., Proper, H.A., Sanz, J., Terlouw, L., Tribolet, J., Verelst, J. and Winter, R. (2013) 'The discipline of enterprise engineering', Int. J. Organisational Design and Engineering, Vol. 3, No. 1, pp.86–114

Katzensten, G., & Lerch, J. (2000). Beneath the surface of organizational processes: a social representation framework for business process redesign. ACM Transactions on Information Systems Vol. 18 (4) 383-422

Shannon, R. (1975). System simulation: the art and science. Prentice Hall, Englewood Cliffs, New York.

Zeigler B. P., Praehofer, H., and Kim, T.G. (2000). Theory of modeling and simulation, 2nd Edition, Academic Press.