Benefits of Enterprise Ontology in Governing Complex Enterprise Transformations

Martin Op 't Land[1,2,3] and Jan L.G. Dietz[2]

[1] Capgemini Netherlands, P.O. Box 2575, 3500 GN Utrecht, the Netherlands
[2] Delft University of Technology, Delft, the Netherlands
[3] Antwerp Management School, Antwerp, Belgium
Martin.OptLand@capgemini.com, J.L.G.Dietz@tudelft.nl

Abstract. Enterprises increasingly execute complex transformations, such as mergers and splits, chain redesign, sharing and sourcing, and the rationalization of products, processes and applications. Its consequences are seldom sufficiently timely and completely discerned, while this is essential for governing such a transformation. Already in a small example we can see why Enterprise Ontology, as defined in DEMO (Design & Engineering Methodology for Organizations) delivers a clear and valuable instrument for transformation governance. Especially DEMO's Construction Model, which is the most abstract ontological aspect model, combines high expressiveness with a high Return On Modeling Effort (ROME).

Keywords: Enterprise Transformation, Governance, Enterprise Ontology, DEMO, Enterprise Engineering, ROME

1. Introduction

Heraclitus' saying "The only constant is change" (500 BC) [13] certainly applies to enterprises[1]. We observe an increasing need to perform better in "profit – people – planet", especially in the recent economic crisis. Enterprise activities are shared, in-sourced, out-sourced, off-shored and back-sourced again. They split up and merge. After giving room for regional autonomy, application portfolio rationalization programs are executed. All these attempt to save costs, improve quality and increase agility, which should ultimately lead to unique advantages in customer intimacy, product leadership and operational excellence [31].

The impact of such change-decisions is generally complex and large. Many stakeholders have to be reckoned with, the constraints and opportunities of laws and regulations have to be taken into account, and relevant parties in the chain need to be actively involved. The changes will not only affect products, processes and ICT[2] applications, but also the enterprises' economy and organization. Therefore, such changes call for a well-governed enterprise transformation. To this end, a fundamental and

[1] We use the term "enterprise" for any goal-oriented cooperative of people.
[2] ICT stands for Information and Communication Technology.

essential insight in organizations is needed, such that these transformations become intellectually manageable.

Enterprise Ontology, as adopted in DEMO[3] [11], reveals the essential, deep structure of an organization. A series of case studies [e.g., 16, 22, 23] of real-life reorganizations, organizational splits, post-merger integration and application portfolio rationalization has been published, in which DEMO is the core modeling method to support decision-making. These studies also report a tremendous ROME (Return On Modeling Effort). We found particularly that DEMO's systematic and reproducible abstractions from the realization and implementation of organizations are beneficial. Taking OMG's EU-Rent Case [20, 18] as an example, we will illustrate the identified benefits and make plausible that applying DEMO provides the required understanding for intellectually manage enterprise transformations, and to do this in 10% of the time commonly used.

The remainder of this paper is structured as follows. Section 2 categorizes impact of change in enterprise transformation and presents requirements for better governance of transformations. Section 3 briefly introduces DEMO and its underlying PSI-theory, and hypothesizes causes for the achieved benefits. In section 4 we discuss these hypotheses, using the EU-Rent transformation example. Finally, section 5 provides the conclusions as well as directions for further research.

2. Enterprise transformation – needs for improvement

Rouse defines enterprise transformation as a "… *fundamental* change that substantially alters an organization's relationships with one or more key constituencies, e.g., customers, employees, suppliers and investors" [26 p279, italics ours]. But what makes a transformation fundamental? For example, firing 40% of your people can add value from the perspective of the organization striving for operational. Is this a fundamental change? For the people involved, this will certainly be the case. For the organization the change could be large – e.g. when locations or operating countries are shut down, or complete products are retired. But the change could as well be small – keeping the product portfolio and the processes the same, just operating at a lower volume with the same mechanisms.

In his *Law of Requisite Variety*, Ashby [1] states that each system has to deal with complexity – expressed in variety - in its environment, answering it by sufficient internal variety. Beer [4] elaborates then *how* organizations should deal with complexity to stay viable. His *Viable System Model* (VSM, summarized in Table 1) states that each *viable* system – autonomous system, adaptable to its environment – (1) is recursive and (2) is composed of five interacting subsystems (see Fig. 1), of which the highest level has the lowest variety.

Using VSM, one can now order typical changes on a scale from low-impact (and frequent) to high-impact (and few) – the latter ones we consider to be fundamental.

To what extent are fundamental enterprise transformations successful? Many enterprises have undertaken transformations, only some flourished [26 p292]. Charan & Colving estimate [6] that 70% of CEO-firings are caused by not getting their enter-

[3] Design & Engineering Methodology for Organizations (www.ee-institute.com)

prise to implement agreed upon and reasonable strategies. As another example, in spite of the popularity of Mergers & Acquisitions [8], researchers commonly [24] suggest that approximately half of all M&A have proven unsuccessful.

Table 1. The five systems of the Viable System Model (VSM) - summary

	content	example
1	Primary or production system; to deliver products and services to its environment.	changes impacting the daily primary operations, such as solving an unexpected customer complaint, trying to speed up one order
2	Coordination, by mutual adjustment between the Systems 1, to prevent oscillation; e.g. by scheduling, common standards, procedures etc.	(re-)scheduling resources such as people and space, so that not all pupils try to have Math in the same room, while the Chemistry teachers have nobody to teach
3	Control or Cohesion, manages resources (men, materials, machines, money) and performance requirements of Systems 1. System 3 is supported by System 3* Audit or Monitoring (sporadic, in-depth). Focus: "Are we doing the things right?"	given that we still are delivering the same products to the same markets, resources are re-assigned to anyhow perform, or the performance-targets are adapted
4	Intelligence or Future, looking outwards to the environment to understand how the organization needs to adapt in order to stay viable. Focus: "are we doing the right things?"	new products or services are proposed, or current products or services are going to be delivered with a different quality, or products & services are retired
5	Policy or Identity, balancing current and future demands of the organization.	contemplating our identity, we decide we should no longer stay an airline company, but gradually and controlled become a connector of people

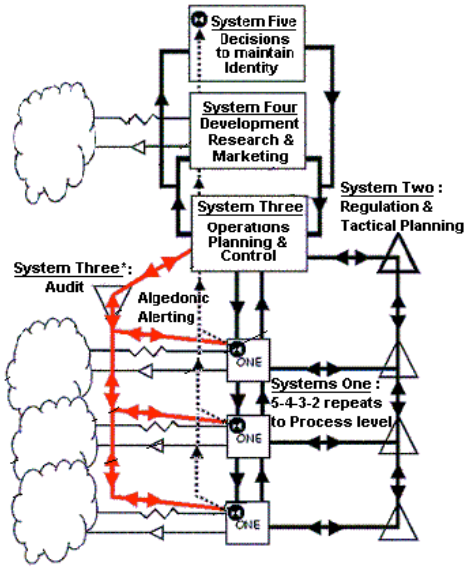

Fig. 1. Subsystems of a viable system, according to the Viable System Model (VSM) [11]

Rarely is the impact of the intended change discerned sufficiently timely, coherently and completely. This has a destructive impact on many stakeholders. Indeed, the lack of shared objective insight gives rise to speculations, interpretations, unnecessary polarizations and the unintended overlooking of stakeholders and their interests. On one hand stakeholders know that providing enough time for building support and buy-in is essential for a successful transformation. On the other hand they feel pressure for fast decisions to protect personal positions or to be able to give well-founded answers to customer markets, stock exchanges and labor unions. As a consequence decision-making often tends to oscillate between slow moving and ad hoc.

Therefore, to make the fundamental change of enterprise transformation successful, good governance needs to be in place, using an *enterprise transformation dashboard* [21, 12]. First and foremost, this should be derived from the *goals* of the enterprise transformation, by clear answers to questions like "what are the intended benefits and the exploitation costs" and "what durability, scalability and sourceability should we aim for". Next to that, guidance is needed on the *process* of the enterprise transformation, monitoring transformation costs, duration and risks. Thirdly, the basis of all this is a shared and objective insight in the *content* of the enterprise transformation. Applying de Leeuw's governance paradigm [15] with "enterprise transformation" as the target system, several authors propose [21 p29, 12] to use a *dashboard* as a means for such an *informed governance* for enterprise transformation, containing at least (1) *indicators* – e.g., models, views, performance measurements – giving insight into the enterprise's current state, the enterprise's current performance, the enterprise's future (expected) performance, and the (selected) direction and progress of its transformation processes, and (2) *controls* – e.g., (enforced) reference models, design principles, standards – allowing the transformation processes to be influenced, such as the enterprise's desired future state, plateaus of intermediary stages and overall regulations.

To enable coherent and consistent enterprise (re)design, resulting in unified and integrated enterprise operations, we propose as extra requirements for this dashboard:

- it should give an holistic view of the enterprise, clarifying coherence between its components;
- the views should be necessary and sufficient to evaluate different future implementations – with each different people/organizations and different (a/o ICT-) means;
- therefore implementation-independent views of the enterprise should be available, allowing for mapping or comparing different implementations of the enterprise;
- finally, making and maintaining the enterprise views should be cost-effective in relation to the issues to be solved or prevented; i.e., the views should have a good *Return on Modeling Effort* (ROME).

3. The DEMO Methodology

A complete, so-called *essential model* of an organization consists of four aspect models: Construction Model (CM), Process Model (PM), Action Model (AM), and State Model (SM). The CM specifies the composition, the environment and the structure of the organization. It contains the identified *transaction types*, the associated *actor roles*

as well as the *information links* between actor roles and transaction banks (the conceptual containers of the process history). The PM details each transaction type according to the universal transaction pattern. In addition, it shows the structure of the identified business processes, which are trees of transactions. The AM specifies the imperatively formulated *business rules* that serve as guidelines for the actors in dealing with their agenda. The SM specifies the *object classes*, the *fact types* and the declarative formulations of the *business rules*.

Let us briefly introduce the concepts of the CM and the way in which it is represented (Fig. 2). A Construction Model shows the network of identified transaction types and the corresponding actor roles. E.g., transaction type T01 delivers a business service to actor role A00. A00 is called the initiator (consumer) and A01 the executor (producer). The executor of a transaction is marked by a small black diamond on the edge of the actor role box. The solid line between A00 and T01 is the initiator link; the solid line between A01 and T01 is the executor link. Fig. 2 also shows that some other actor role (A07) needs to have access to the history of transactions T01 (production facts as well as coordination facts (e.g., status "requested", "promised", "stated", "accepted")). This is represented by the dashed line between A07 and T01.

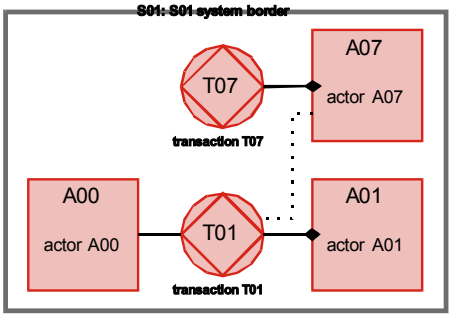

Fig. 2. Typical constructs of a DEMO Construction Model

We now want to hypothesize the next benefits of DEMO, compared to common current modeling approaches such as BPMN [19]:

- It offers a significant reduction of complexity (over 90% in terms of the size of documentation);
- It is an instrument for detecting tacitly performed coordination acts.

In order to test the hypotheses, we will first model EU-Rent, OMG's standard SBVR- and BMM-example ([20, 18]), in its current situation with DEMO. Then we will compare the DEMO model with a common way of process modeling. Finally, we will model a transformation for EU-Rent in DEMO, namely the introduction of loyalty rewards, and explain the use of DEMO in governing this transformation. The next narrative description applies:

EU-Rent is a company that rents cars to persons, operating from geographically dispersed branches. The cars of EU-Rent are divided in car types (brands and models); for every car type there is a particular rental tariff per day.

A car may be rented by a reservation in advance or by a 'walk-in' customer on the day of renting. A rental contract specifies the start and end dates of the rental, the car type one wishes, the branch where the rental starts (called the pick-up branch), and the branch where the rental will end (called the drop-off branch). Rentals have a maximum duration. The person who rents the car is called the renter. The one who is going to drive is called the driver. A rental will only be started if the driver has a valid driving license. In addition, a car of the requested type must be available.

As soon as the car of a rental has been dropped-off, the rental can be ended, after the incurred charge has been paid. This charge may consist of several elements. First, there is the basic charge (number of days times the tariff per day). Next, there may be a penalty charge for exceeding this duration (number of extra days times the late return penalty tariff). Lastly, a location penalty charge is added if the car has been dropped-off at another branch than agreed (this charge depends on the distance between the branches).

Apparently, the relevant unit of service of EU-Rent is the rental of a car for some period. In the case description this notion was already designated by "car rental". The rental of a car is a space-time notion, like e.g. the loan of a book from a library, or the rental of a hotel room. Basically, it is the right to use a space-bound service for some time. The usage of such a service has to be started and to be ended explicitly.

So, we identify two transaction kinds in the B-organization of EU-Rent, which we will call rental start (B-T01) and rental end (B-T02). The transaction results are respectively "[rental] has been started" (B-R01) and "[rental] has been ended" (B-R02). In the formulation of these results "[rental]" is a placeholder for concrete individual instances of the type rental. By convention, the executor of B-T01 gets the actor role number "B-A01"; let us call this actor role "rental starter". Similarly, the executor of B-T02 is designated by "B-A02"; let us call it "rental ender". Moreover, we call the initiator of both B-T01 and B-T02 "renter"; let us give this external (and by convention composite) actor role the number "B-CA01". In the period between the creation time of B-R01 and the creation time of B-R02 of a rental, the rental is considered to be alive. It means that during the lifetime of a rental, B-CA01 (the renter) has the right to make use of the rented car. Note that booking in advance seems to be a separate transaction but it is not. Booking in advance means only that the requested creation time of B-T01, i.e. the contracted start date, is some time in the future.

When the rental start is promised, the rental starter will proceed with requesting the driver to pick up the selected car at the contracted pick-up branch (B-T03), as well as to drop off the car at the contracted drop-off branch on the contracted end date (B-T04). So, the initiator of B-T03 and B-T04 is B-A01. The executor of B-T03 as well as the executor of B-T04 is an elementary actor role within the (external) composite actor role B-CA02, which we will call "driver". At some time, the driver will drop-off the car at some branch, and the renter will subsequently request to end the car rental.

Before completing the rental end (B-T02), however, the costs of the rental have to be paid. So, we identify the last transaction kind, B-T05 (rental payment). The initiator is obviously B-A02 (rental ender) and the executor is (by convention) the elementary actor role B-A05 within the external (composite) actor role B-CA03, which we will name "payer".

When the renter initiates the rental end (B-T02) the rental ender will check whether the car has been dropped off, i.e. whether the car drop-off has been accepted. Note

that it may be the case that B-T05 will not be initiated at all because the car pick-up (B-T03) has not been executed! Although strange of course, this may happen. The normal case, however, is that the car has been picked-up and been dropped-off. As soon as B-T05 is accepted, B-T02 will be continued and completed.

Fig. 3 shows EU-Rent's Construction Model (CM), Table 2 is its Transaction Result Table (TRT). The executor of a transaction is marked by a small black diamond on the edge of the actor role box. The other linked actor role is the initiator.

Table 2. Transaction Result Table EU-Rent, current situation

Transaction kind	Transaction result
B-T01 rental start	B-R01 [rental] has been started
B-T02 rental end	B-R02 [rental] has been ended
B-T03 car pick-up	B-R03 the car of [rental] has been picked-up
B-T04 car drop-off	B-R04 the car of [rental] has been dropped-off
B-T05 rental payment	B-R05 [rental] has been paid

From the action rules (not discussed here) we derive the information links (dashed lines) in the CM. It means that the actor role is allowed to access the contents of the connected C-bank and/or P-bank. Note that there are two external P-banks; they contain facts that are the result of transactions outside the boundary of the organization we focus on (the bold grey rectangle named "EU-Rent").

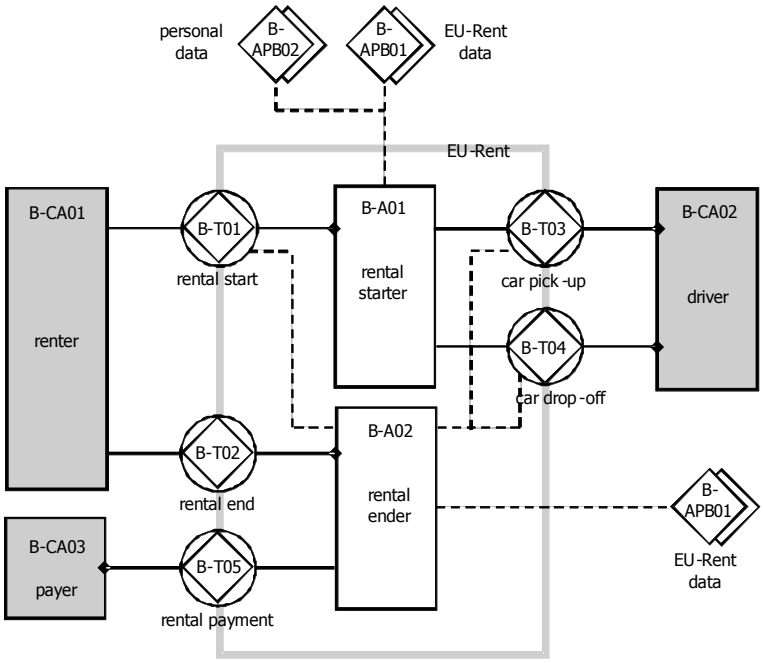

Fig. 3. Construction Model of EU-Rent, current situation

Fig. 4 reuses and shows a Business Process Diagram (BPD) according to the OMG-standard Business Process Modeling Notation (BPMN) [19], elaborated for the same case EU-Rent [18, 20], for the business process "advance reservation" [27 p89].

How does this BPD deal with the identified coordination acts? In the diagram we noticed two of them explicitly, namely the *request* ("rental request"), the *reject* ("rejection") and the *accept* ("accept payment"). For many other coordination acts, we are left more or less guessing if they are implicit, asking questions such as "is the *accept* of the rental request embodied in the hand-over of the paper rental contract?", "is the pick-up of the car always promised & stated (even if the driver appears to be drunken and unable to show his driver's license)?" and "is payment of the car ever requested, promised etc.?". So, a BPD like the one made in BPMN does not enable checking the completeness of transactions.

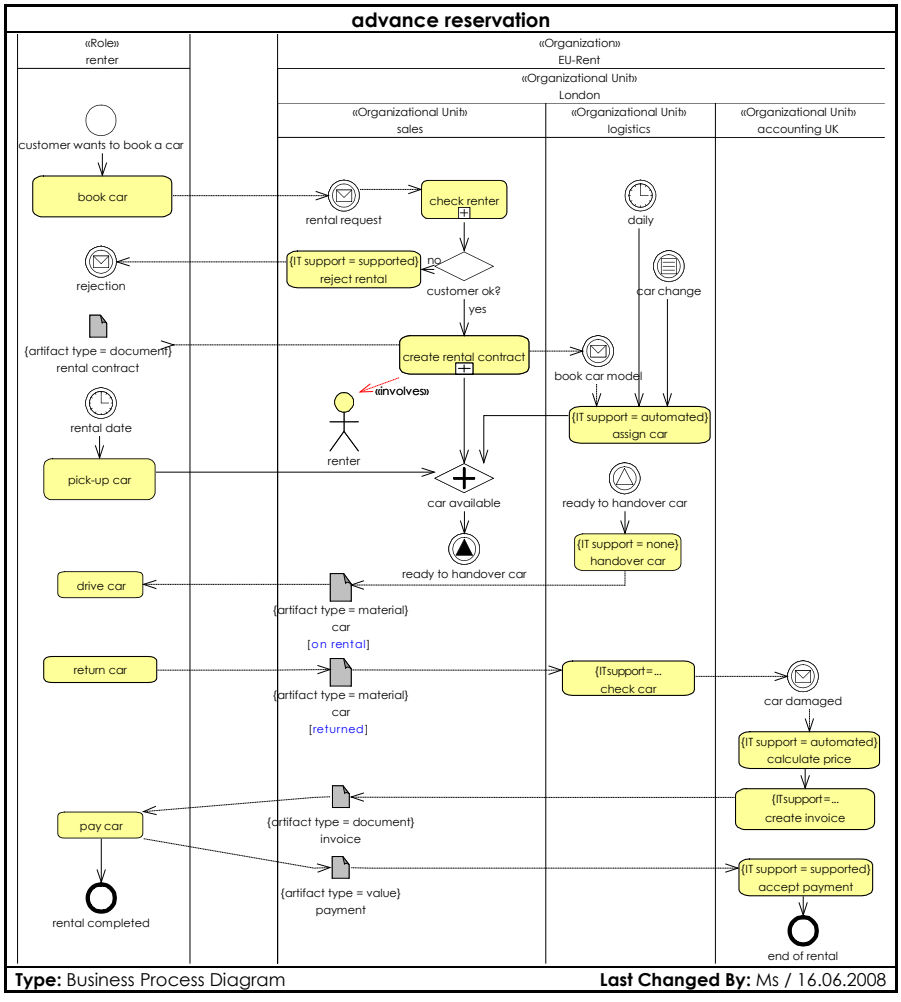

Fig. 4. Business Process Diagram "advance reservation EU-Rent" [27 p89]

How does this BPD deal with infological and datalogical aspects? As infological examples, it shows calculations, such as “calculate price” and “assign car”. On the datalogical level, it shows a “rental contract” and an “invoice”. We are now left guessing to what extent this BPD is complete here. For instance, are no calculations done elsewhere, such as “calculate credit risk” in deciding whether to accept the renter? Or when the car has been damaged, will a damage report be made, recorded in EU-Rent and a copy been given to the customer? Again, a BPD like this does not enable checking for completeness of I- and D-actions.

What signs of organizational and ICT-implementation do we see in the BPD? By using “swimming lanes”, the BPD clearly shows the way how the business activities are organizationally embedded – in this example a geographical unit (London) and three departments within such a geographical unit. If the organization changes – e.g., by outsourcing the car logistics and merging the sales and the accounting unit –, all diagrams need to be changed here as well. Also the ICT- and paper-based implementation are shown by texts ({IT-support = …}) and symbols in the BPD. When such an implementation changes, e.g. by electronic invoices and contracts or fully-automated damage-checking, the BPD needs to be changed.

The BPD shows a clear order of working for all people involved. This also means that the BPD has to be changed the moment the order of working changes, e.g. when we decide to ask for a partly payment in advance to decrease the amount of no-shows.

Finally, it is difficult to determine whether this BPD is consistent. E.g., by making no distinction between renter and driver, we run the risk that we ask the driver to be credible and the renter to have a driver’s license, which should be, as you will notice, the other way around. Also, by making different processes for “walk-in rental” and “advance reservation” we introduce an opportunity for unintended “unequal treatment” between a renter in these two scenarios [cp. 27 p85], while probably uniform processes were meant.

From our consulting practice, we sometimes notice user-appreciation of the BPD because it clarifies for all executors of a process the order of working and the current organizational and ICT-implementation. At the same time, our analysis shows several disadvantages of using a BPD for the purpose of governing transformation, because (1) completeness of coordination acts, and thereby completeness for the whole chain of actors and services cannot be checked, (2) completeness of infological and datalogical actions cannot be checked, (3) order of working is fixed, and (4) all organizational and ICT-implementation choices are “hard-coded”.

4. Modeling EU-Rent’s transformation

EU-Rent provides car rental service across Europe and North America for both business and personal customers. It operates nation-wide in each country of operation, focusing on major airports, competing head-to-head, on-airport, with other premium car rental companies such as Avis and Hertz.

EU-Rent wants to improve customer satisfaction by industry-leading customer service, well maintained cars and by having vehicles available for rental when and where customers expect them. This should contribute to external recognition of EU-Rent as

'premium brand' and to top ratings by parties such as A C Nielsen. EU-Rent herself monitors her progress in this area by a quarterly customer satisfaction survey.

As part of the customer satisfaction improvement program, EU-Rent has decided to introduce an attractive loyalty rewards scheme for frequent renters during 2012-H1. Frequent renters should become rewarded by EU-Rent with loyalty credits, especially when they extend their rental. Also customers should be able to spend loyalty credits at EU-Rent in special offerings, such as "three-days-for-the-price-of-two" or simple discounts. In her strategy, EU-Rent has also decided to join an established rewards scheme run by a third party – i.e., outsource rather than building own scheme.

Table 3. Transaction Result Table EU-Rent: extra transactions for future situation

Transaction kind	Transaction result
B-T06 credits awarding	B-R06 credits for [rental] have been awarded
B-T07 credits cashing	B-R07 credits for [rental] have been cashed

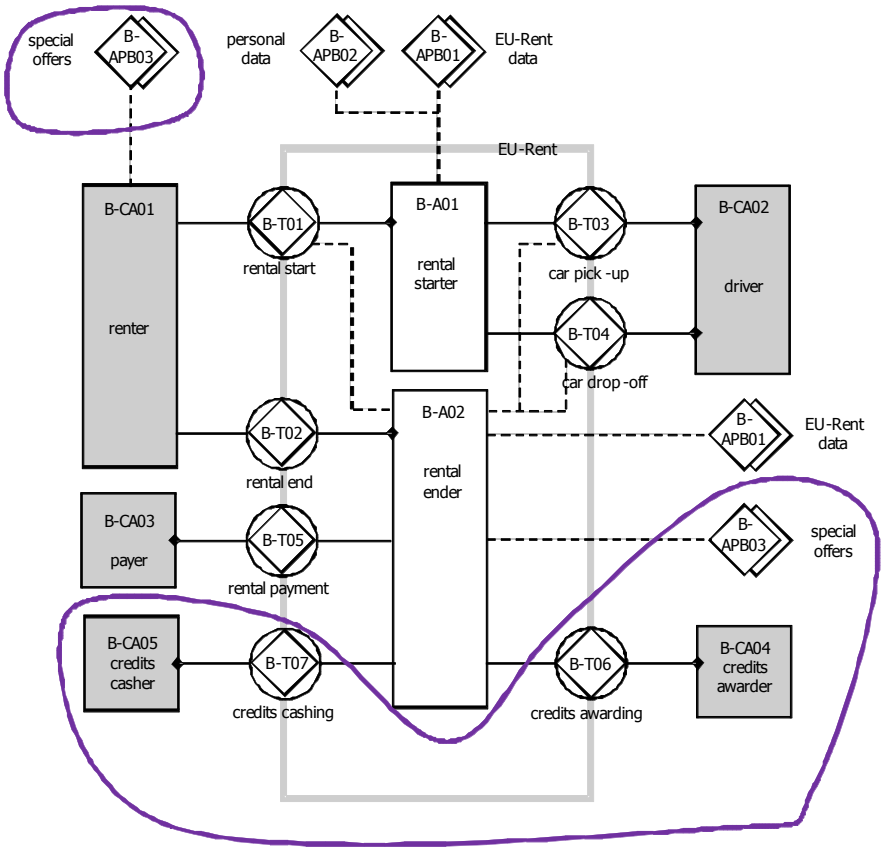

Fig. 5. Construction Model EU-Rent, future situation with loyalty credits

Wiser from some previous experiences, the CEO of EU-Rent wants to ensure integral governance before giving the final go for the loyalty rewards project. First of all, she wants to have a better insight in the ToBe-situation: what does it mean for the organization, the personnel, the ICT, the way of cooperation with third parties and the exploitation costs versus the extra business this should create for EU Rent. Also she wants to have an insight in the transformation involved: what migration should be undertaken, what are the risks, what are the transformation costs and what is the expected duration. And finally she wants to have an understanding about the evolvability of the solution, e.g. would it be possible to start with a "EU-Rent only" earning and spending of loyalty credits and then broaden it to save and spend loyalty credits with other parties as well – or the other way around.

With a similar line of reasoning we build the DEMO CM for the new situation, for which Fig. 5 shows the new complete CM (changes marked in purple) and the TRT in Table 3 shows the additional transactions. The renter (B-CA01) will be informed on special offers (B-APB03), such as discounts or extra loyalty rewards. The same information will be needed by the rental ender (B-A02), when (s)he has to ask for the right amount of money to be paid (B-T05), of which a part can be paid by cashing loyalty credits (B-T07). By the rental, the renter can also earn credits (B-T06).

From Fig. 5, we now can answer several questions asked by EU-Rent's CEO. First of all we immediately see several new B-actor roles appear, namely the *credits awarder* (B-CA04) and the *credits casher* (B-CA05). Next, the appearance of the new fact-banks *credits awarding* (B-T06), *credits cashing* (B-T07) and *special offers* (B-APB03) introduces also new D-actor roles, responsible for recording, distributing and copying data from these three fact-banks, such as the recording of special offers in B-APB03. And finally we see I-actor roles appear for each new explicit (dashed) or implicit (solid) information link in the CM, such as giving selections from *special offers* (B-APB03), directed towards specific customers, regions or periods.

For each of these actor roles we can consider and compare several alternative organizational implementations such as, for instance, the alternative "outsource loyalty rewards scheme" versus "build own loyalty rewards scheme". Each alternative puts up another organization border, thereby creating another cooperation with parties, ultimately resulting in other future contracting [23 pp77-92].

Looking back at the questions of the CEO, we see we have now answered some of her questions about the ToBe-situation, namely "what does the transformation mean for the way of cooperation with third parties and for the organization". In order to answer the "consequences for personnel and ICT", more insight in Quality of Services is needed – then also the exploitation costs can be estimated, as well as the transformation consequences (content, risks, costs, duration). "The extra business this should create for EU Rent" should be derived from market research, not from the CM.

Finally the question about the evolvability of the solution, e.g., would it be possible to start with a "EU-Rent only" earning and spending of loyalty credits and then broaden it to save and spend loyalty credits with other parties as well – or the other way around. The CM clarifies here that in a situation of outsourcing the loyalty rewards scheme this is a matter of a different Service Level Agreement (SLA) with the "loyalty rewards service provider". In this SLA agreement should be reached about which credits awarded elsewhere should "count" for EU-Rent and the other way

around. In case of insourcing the loyalty rewards scheme, all these considerations should be brought into scope and elaborated.

5. Conclusions and further research

As main characteristics of using DEMO and its Construction Model (CM) we found:

1. it ensures completeness in unambiguously discerning all activities – business, infological and datalogical – required to deliver a certain product or service;
2. this in turn helps considering and comparing different implementations of these activities and their executing actor roles in organizations, people and automation;
3. the models are able to express "just in time, just enough detail";
4. making the models is possible with an attractive ROME.

The first three characteristics are caused by three parts of the Ψ-theory, namely:

- the *distinction axiom*, stating that "there are three distinct human abilities playing a role in the operation of actors, called performa, informa en forma"; applied as a means for abstraction to the performa ("business") level, it offers a significant reduction (at least 70%[4]) of complexity; applied as a means for concretion, it ensures completeness in discerning responsibilities in information provision (infological actor roles) and data governance (datalogical actor roles);
- the implementation notion, stating that an organization should be made operational by means of *technology*, being organizations, individual people and other (a/o ICT-) means; applied as a means for abstraction, it offers the opportunity to look across current or envisioned organizations and other means; applied as a means for concretion, it offers the opportunity to compare many different implementations, first in terms of people, functionary types, organizations, next to that also in several ambition levels of automation;
- the *transaction axiom*, stating that "coordination acts (C-acts) are performed as steps in universal patterns"; applied as a means for abstraction to the level of transactions, it offers another significant reduction (at least 70%[5]) of complexity; applied as a means for concretion, it ensures detecting all C-acts, also the ones currently performed tacitly.

The fourth characteristic, an attractive Return On Modeling Effort (ROME), is caused by simultaneous abstraction according to the distinction axiom and the transaction axiom, as is done in DEMO's Construction Model. Its effect is a reduction of at least 90% of the time commonly used (the earlier mentioned 30%*30%, confirmed by practices such as the SGC-case [16 p77] which even mentions 95%), and a result that has a greater power of expression for evaluating implementation alternatives than commonly used – such as in common process models or flowcharts.

As typical benefits of using DEMO and its Construction Model (CM) we found:

---

[4] One business transaction needs already at least two infological transactions (one per information link), and each infological transaction at least one datalogical transaction. So only focusing on business transactions would theoretically give even a reduction of 80%.

[5] Assuming that at least the four standard coordination acts are generally described, summarizing those 4 C-acts in one transaction would give a reduction of 75%.

1. business-activities, currently executed by different organizations can be made unambiguously and fast comparable; this is very handy in post merger integration, implementation of shared service centers, insourcing, outsourcing, and as a first step in uniformizing processes when cross-region staffing is required;
2. ICT-applications, currently supporting different organizations or departments, can be made unambiguously and fast comparable; this is very handy in application portfolio rationalization.

In terms of the VSM subsystems, DEMO supports governing changes as follows:

- DEMO is not useful in directly governing changes in *System One*, so the impacting of the daily primary operations, though DEMO can be used to structurally improve exception handling and to embed systematic learning in an organization [2];
- for changes in Coordination, *System Two*, DEMO can clarify which information is really and minimally needed by which actor role;
- for changes in Control, *System Three*, DEMO supports re-assigning responsibilities, especially when organizational borders are crossed, such as in BPO or SSCs;
- for changes in Intelligence, *System Four*, DEMO clarifies for several alternative answers to environmental challenges in terms of changed products or services, what actor roles are needed in these alternative products or services;
- for changes in Identity, where *System Five* balances current and future demands of the organization, DEMO can support evaluating implementation alternatives when the DEMO CM is combined with specific mappings on organization and ICT.

In terms of the *enterprise transformation dashboard*, which part is now fulfilled by DEMO? The current, future and intermediate states of the enterprise can be expressed in DEMO models and their mapping to current and different future/intermediate (organizational and ICT-) implementations. This gives a complete insight in all actor roles needed in each state, and thereby a first order insight in the consequences in terms of organizations, people and required ICT for the future/intermediate state (see e.g. [22]).

To what extent does DEMO fulfill our additional requirements for the enterprise transformation dashboard? Indeed, DEMO gives an holistic and implementation-independent view of the enterprise (without names of organizations or functionary types), allowing to compare different (organizational and ICT-) implementations, such as in sharing and sourcing. Together with mappings on (organizational and ICT-) implementations, DEMO has appeared to be a necessary, useful and in some cases even sufficient instrument to evaluate these different implementations. DEMO also shows a complete insight in the information required by each actor role, clarifying also the origin of new facts in reality. By its attractive ROME, also the making and maintaining of views based on DEMO appear to be cost-effective.

Because of these unique characteristics, we propose to put DEMO, and especially its Construction Model, as a standard on the *enterprise transformation dashboard* as the prime model for getting first order insight in impact of change. As shown earlier: not because DEMO models answer all questions stand-alone. It will, as case studies have shown already, always need to be supplemented by, and connected with strategically (e.g. [17]) and functionally oriented models (e.g, with OMG's Business Motivation Model BMM [18]), with infological and datalogical aspects, with implementation-oriented process and ICT models and with principles guiding the transformation.

Examples of enriching implementation-oriented models with DEMO concepts and connecting these to DEMO models (e.g., with Lean Six Sigma [9], ArchiMate [10], ARIS [29], and BPMN [5, 14]) look promising and deserve further research effort.

At the same time, still richer insights are needed in enterprise engineering, e.g.:

- to what extent does enterprise engineering practice *empirically* confirm that the time savings found in earlier case studies can be ascribed indeed to the abstractions from (1) coordination acts and (2) infological and datalogical actions?
- how to apply simulations already on the level of the DEMO CM to support evaluating several implementation alternatives - as has been done already on the more detailed level of processes [3];
- how to support making first order estimations of (business- and ICT-) performance, *exploitation* costs and risks of implementation alternatives already on the level of the DEMO CM;
- how to support making first order estimations of *transformation* costs and risks of implementation alternatives already on the level of the DEMO CM;
- how could the other three DEMO aspect models (PM, SM, AM) contribute to supporting transformations?
- what VSM-subsystems do benefit most from applying DEMO?

Already Heraclitus (500BC) – with his statement "The only constant is change" – is seen focusing not on things as constantly changing, but on things as constant while changing. Enterprise Ontology offers an important contribution in discerning potentially constant parts in (chains of) organizations, enabling executive management to choose.

6. References

1. Ashby, W.R.: An Introduction to Cybernetics. Chapman & Hall, London (1956)
2. Aveiro, D. S. A. de, Silva, A. R., Tribolet, J. M. N. S.: Modeling the Function Perspective in ODE: Improving Unknown Exception Handling and Change. In proceedings of *International Workshop on Organizational Design and Engineering – IWODE09 (2009)*
3. Barjis, J.: The importance of business process modeling in software systems design. Science of Computer Programming 71, pp73–87. Elsevier Science Direct (2008)
4. Beer, S.: Brain of the Firm; Allen Lane, The Penguin Press, London, Herder and Herder, USA (1972)
5. Caetano, A., Assis, A, Tribolet, J.: Using Business Transactions to Analyse the Consistency of Business Process Models. In C. Møller, editor, Proceedings of CONFENIS 2011, CIP Working Papers. Center for Industrial Production, Aalborg University (2011)
6. Charan, R., Colvin, G.: Why CEOs fail. Fortune Vol. 139 no. 12, pp68-78 (1999)
7. Dietz, J.L.G.: Enterprise Ontology – theory and methodology. Springer, Heidelberg (2006)
8. Eigruber, S.: Mergers & Akquisitionen: Der Hunger ist vorüber. Wirtschaftsblatt, 2 July 2008 (http://www.wirtschaftsblatt.at/home/zeitung/aktuell/333341/index.do)
9. Ettema, R. Diagnosing the enterprise, towards a true causal claim. Forthcoming.
10. Ettema, R., Dietz, J.L.G.: ArchiMate and DEMO – Mates to Date? In A. Albani, J. Barjis, J. L. G. Dietz, W. Aalst, J. Mylopoulos, M. Rosemann, M. J. Shaw, and C. Szyperski, editors, *Advances in Enterprise Engineering III*, volume 34 of *Lecture Notes in Business Information Processing*, pp 172–186. Springer Berlin Heidelberg, 2009.

11. Green, N. Wikipedia: Vsm.gif. http://en.wikipedia.org/wiki/File:Vsm.gif
12. Harmsen, F., Proper, H.A., Kok, N.: Informed Governance of Enterprise Transformations. In: *Advances in Enterprise Engineering II* (Proceedings of the First NAF Academy Working Conference on Practice-Driven Research on Enterprise Transformation, PRET 2009, held at CAiSE 2009). Springer, Berlin Heidelberg, pp 155-180 (2009)
13. Heraclites of Ephesus (535 BC - 475 BC), as quoted/interpreted by Plato in Cratylus. The quote is rather free; it should originally read "Πάντα ῥεῖ" (panta rhei), which literally means "everything flows" or a bit more free, "everything moves". Plato quotes "Πάντα χωρεῖ" (panta chōrei), meaning "everything changes" – from *chōros*, to change.
14. Horst, P.v.d.: From business transactions to business processes work flows; Using DEMO and BPMN. Master's thesis, Delft University of Technology (2010)
15. Leeuw, A.C.J. de: Organisaties – management, analyse, ontwerp en verandering; een systeemvisie. In Dutch. van Gorcum, Assen, The Netherlands (1982)
16. Mulder, J.B.F.: Rapid Enterprise Design. PhD thesis, Delft University of Technology (2006)
17. Öberg, C., Henneberg, S. C., and Mouzas, S.: Changing network pictures: Evidence from mergers and acquisitions. Industrial Marketing Management, 36(7):926 – 940 (2007)
18. OMG: Business Motivation Model, v1.0 (2008)
19. OMG: Business Process Modeling Notation (BPMN), v1.2 (2009)
20. OMG: Semantics of Business Vocabulary and Business Rules (SBVR), v1.0 (2008)
21. Op 't Land, M., Proper, E., Waage, M., Cloo, J., Steghuis, C.: Enterprise Architecture – Creating Value by Informed Governance. In: Enterprise Engineering Series. Springer-Verlag Berlin Heidelberg (2009)
22. Op 't Land, M., Zwitzer, H., Ensink, P., Lebel, Q.: Towards a Fast Enterprise Ontology Based Method for Post Merger Integration. In: Proceedings of the 2009 ACM symposium on Applied Computing, Honolulu, Hawaii, USA. ACM New York, NY, USA, pp 245-252 (2009)
23. Op 't Land, M.: Applying Architecture and Ontology to the Splitting and Allying of Enterprises. PhD thesis, Delft University of Technology (2008)
24. Pautler, P.A.: The Effects of Mergers and Post-Merger Integration: A Review of Business Consulting Literature. Federal Trade Commission (2003) http://www.ftc.gov/be/rt/businesreviewpaper.pdf , last visited August 29th, 2009
25. Recker, J.: Opportunities and constraints: the current struggle with BPMN. Business Process Management Journal, 16(1):181 – 201 (2010)
26. Rouse, W.B.: A Theory of Enterprise Transformation. Systems Engineering, Vol. 8, No. 4, pp 279–295. Wiley Periodicals, Inc. (2005)
27. Schacher, M.: Mini EU-Rent; Business Model v23.06.2008. KnowGravity (2008), http://www.knowgravity.com/pdf-e/Mini%20EU-Rent%20BU.pdf
28. Stein, S.: What is ARIS? Blog of the ARIS BPM Community, last visited January 24th, 2012 http://www.ariscommunity.com/users/sstein/2009-10-23-what-aris
29. Strijdhaftig, D.: On The Coupling Of Architectures: Leveraging DEMO Theory Within The ARIS Framework. Master's thesis, Delft University of Technology (2008)
30. The Open Group: ArchiMate® 1.0 Specification. Technical report. The Open Group, Thames Tower, 37-45 Station Road, Reading, Berkshire RG1 1LX, United Kingdom (2009)
31. Treacy, M., Wiersema, F.: The Discipline of Market Leaders – Choose Your Customers, Narrow Your Focus, Dominate Your Market. Addison-Wesley, Reading (1977)