Upgrading DEMO / PSI Transactions to Asynchronous and Distributed Networks with: Multi-Actor Roles; Improved Expressiveness; more Flexibility with Incomplete Requests; and Handling Change in Cascade of Transaction

PhD. Student: Duarte Gouveia[1],
PhD. Supervisor: David Aveiro[1]

[1] University of Madeira & Madeira Interactive Technologies Institute,
Caminho da Penteada 9020-105 Funchal, Portugal
duarte.gouveia@m-iti.org, daveiro@uma.pt

Abstract. This work reports progress in the author's PhD program. The author aims at contributing to the evolution of Enterprise Engineering (EE) theory, more specifically on DEMO/PSI theory, namely: a) improving the transaction pattern so that it can be used in asynchronous and distributed systems; b) allowing more than two actor roles in the transaction pattern; c) adding expressiveness through the missing conversation patterns that are currently omitted in the transaction pattern (initial agreement; discussion states); d) adding flexibility on the constraint that request act must contain all the required information for the transaction, allowing incomplete requests; e) improving the change problem on a cascade of transactions by reducing its impact. The author presents the identified problems and the research questions that guides this research. The current work is on the development stage for validation purposes. The experiments planned for validation are described in a succinct way, as well as the expected outcomes, based on the chosen research method.

Keywords: Information Systems; Organizational Engineering; Enterprise Engineering

1 Introduction

This author's PhD program aims at giving a contribution to "Design Engineering and Method for Organizations" (DEMO) [1] as the focus for this work. The first issue we should address is: Why DEMO?

The author choose DEMO over alternatives to model business transactions, like modelling business processes with notations like BPMN [2], ArchiMate [3] or others [4], because DEMO provides a consistent and concise way to summarize organizations based on the splitting between Ontological, Infological and Datalogical transactions [1].

The prevailing paradigm in software development is a data-centric approach. Ontologies try to model data into to concepts with attributes, classify concepts in types and establish relationships between concepts. DEMO also models data (in the State Model), but that is not it's focus or starting point. DEMO uses a communication-centric approach, based on transactions where acts are performed to produce accepted results. This is an approach that is not the main stream, but the many reports on the use of DEMO method provide the insight that this might be a better approach for modelling organizations.
The usage of DEMO method in practice results in diagrams that describe the organizations that are smaller than those produced with alternative methods, organized by concerns and modular architecture. "ArchiMate is typically used for high-level processes and their relations (…), but not for detailed workflow modelling" [5]. On the contrary, "BPMN supports detailed subprocess and task modelling down to the level of executable specification but lacks the broader enterprise context" [5].
As any theory in science, DEMO is not perfect. We describe in Section 2 some of the constraints that currently exist on DEMO theories, that we wish to address in this work, both from the author's previous work and from others references in the literature. In Section 6 the proposed changes are presented and discussed, including their benefits, shortcomings and possible future research developments.
With this work, the author tries to address constraints that currently exist on DEMO's PSI theory and for more than a decade.
This document structure is as following: Section 2 contains the problem statement as the starting point for the research questions, presented in Section 3. Section 4 presents the chosen research method for validation of this research program. Section 5 contains a literature review with a succinct summary of DEMO. Section 6 presents and discusses the theoretical contributions to DEMO/PSI theory by previous work by the author. Section 7 presents the plans for validating the proposed improvements to the DEMO / PSI theory. Finally, section 8 presents the conclusions for the current document, which has the goal of presenting the report on progress on current PhD program.

2 Problem Statement

Although DEMO is only succinctly introduced in the literature review in Section 5.1, this problem statement section needs to address its constraints to support the research questions presented in Section 3. Literature review should only appear as needed to support the research questions. Please refer to Section 5.1 for additional information on DEMO.
In this section we list some of the problems with DEMO set of theories, that we wish to address in this PhD research project:

1. DEMO was designed with a centralized and synchronous approach in mind. Both Delta Theory [6] and PSI theory [7] prove the centralized and synchronous approach. In Delta theory, acts are performed with a blocking execution and a shared clock, which can only happen on a centralized and synchronous system.

Using the Delta theory prescription on a distributed system, a node would have to block waiting for the answer of a remote node, independently of the transmission time and execution time. This is not acceptable for a distributed system but might be acceptable for a centralized system if execution time is sufficiently fast.
When a coordination act is performed in the PSI transaction pattern it becomes instantly known by its counter-party. That can only occur with a centralized and synchronous system approach. On a distributed approach there are transmission times, risks that the message is lost, etc. The performance of a coordination act should only become a social fact when the counter-party acknowledges its existence. This position is backed by Portuguese and European law as presented in previous work [8]. The absence of acknowledge acts in the transaction pattern is a symptom that it was designed with a centralized view in mind.
It is not easy to adapt a system designed with a centralized and synchronous view to an asynchronous and distributed approach. The Internet operates as a distributed and asynchronous network. To be able to have DEMO transactions as a protocol and have persons and organizations communicating with it over the Internet we need to adapt it to an asynchronous and distributed approach.
The author has argued in favour of a distributed and asynchronous approach since 2015, in his initial papers in this PhD program [9][10].

2. DEMO's PSI theory was designed with two actor roles in mind, – the initiator and the executor - performing fixed roles in the transaction pattern. This author's approach wishes to relax these constraints allowing more than two actor roles and more flexibility. Agreements with more than two participants are quite common in society, although the majority of agreements are between two participants. The limitation on two actor roles has been described in the literature as a constraint since 2002 [11][12][13] and also on author's previous work [9][10][14].

3. DEMO's PSI theory was designed with a single pattern in mind, based on the happy flow sequence (request – promise – declare – accept), with additional revoke conversations for each of these acts.
Searl established [15] that the minimal unit of human communication is not a sentence or other expression, but a certain kind of language acts, called speech acts, such as state, assert, describe, warn, remark, comment, command, order, request, criticize, apologize, censure, approve, welcome, promise, express approval, and express regret.
Speech acts are much more general than those acts made available in the PSI transaction pattern. The PSI transaction pattern does not handle the agreement that happens before a request is settled, nor the discussion that might happens after a decline or after a reject. In [8], a model for how the initial negotiation takes place according to the law. Other kinds of acts have been suggested in previous work. For example, in author's previous work [9] the tell act and agree act are proposed, as well as other were suggested as future work: knowledge acts; normative acts; identification acts; question acts; meaning acts; material acts; ask for advice acts. In the previous work by the author [10], a decision act was proposed, as well as other acts were suggested, like informative acts, meaningful acts, question acts, material acts, delegation acts, advice acts and identification

acts. In the author's previous work [8] the following additional acts are identified based on the law: negative acts, information acts, question acts, enaction of rights, acts performed based on time and acts that can no longer be performed after a certain time, acts enforced by a court of law, delegation acts, preference pact acts. Although all these acts have been identified they were not handled in a coherent way and with concrete proposal on how they should be handled in a systematic way, like with a state machine or Petri-net.

4. DEMO/PSI transactions require that the full information needed to get into an accepted state is provided when the request act is performed. This is a very strict requirement that does not match reality in organizations. Options that have a minimal impact on the executor costs are usually left for a later stage agreement, namely close to the declare act. This topic has been discussed in the literature [16][17][18], as well as addressed by the author's previous work [10].

5. According to current theory, when we have a cascade of transactions with dependencies among them, if a revoke request or revoke promise is allowed all the transactions that have dependencies on that transaction where the revoke occurred must also be revoked and later redone step by step. For dependent transactions where no act has been performed this has no impact, however on dependent transactions where some acts have been performed or even worse when an accepted state has been reached, the impact is tremendous. This problem has been addressed in 2017 EEWC Technical Session on PSI Theory [][] with a proposed solution.

3 Research Questions

Based on the problem statements presented in Section 2, we present the following research questions:

1. How to model DEMO/PSI transactions in an unambiguous way, that enables its use on asynchronous and distributed networks?

2. How to model DEMO/PSI transactions in such a way that allows more than two actor roles?

3. What is the minimal set of communication protocols that are needed to model the most common conversations in an organization, as elicited in the literature review?

4. How to model transactions in such a way to loosen the constraints of having the full information required at the request act?

5. In a cascade of transactions with dependencies among them, how to handle a revoke request or revoke promise on one of the transactions at the top with minimal impacts on the transactions bellow that have been accepted or that are in progress?

4 Research Method

We use Ulrich Frank "pluralistic conception of research methods in information systems" [21] as a reference for the scientific validation of the research contributions, combined with Design Science [19][20].

Independently of the specific research method used, there are some essential characteristics of scientific knowledge that must be met a-priori. We shall evaluate each research contribution based on originality, abstraction and justification, which are the three essential characteristic for the results of scientific research, as defined by Ulrich Frank [21].

- The originality claim argues that each research contribution is novel as well as superior when compared to existing knowledge, as it should be more powerful.
- The abstraction claim argues that each contribution has essential common features, providing a more comprehensive explanation of natural phenomena and their evolution, and well as with a broad application to a whole range of instances, going beyond the application to just some single instances.
- The justification characteristic supports each research contribution with convincing reasons producing evidences for its truth. There are different conceptions for truth in science, as we lack a common notion of truth [21]. Research contributions should, first of all, "be coherent with a given body of knowledge, and be regarded as plausible until further evidence rejects this assumption" [21]. Truth claims can be achieved by empirical tests on reality. For research contributions that, due to its nature, cannot be tested empirically, we can claim truth by consensus, using discursive evaluations on their supporting judgements.

To validate the IS artefacts referenced in validation, Section 7, we will use Design Science research method [19][20] within the context of [21].

5 Literature Review

Jan Dietz proposed [22] a classification scheme for the set of theories on the field of study of organizational engineering. The arrows among the classes in Figure 1 signify the support they typically provide to each other to assure soundness or rigor by being well supported in theoretical terms.

- Philosophical theories concern is the more basic foundational building blocks about knowledge in general (conception and perception). At the Philosophical level DEMO has the FI and TAO theories. FI stands for Facts and Information and addresses the information theory. TAO stands for Teleology, Affordance, Ontology and addresses function-construction theory.

- Ontological theories focus on the nature of things, how to model them, how to explain then, how things relate to each other and use them to predict outcomes. At the Ontological level DEMO has the PSI and DELTA theories.

PSI stands for Performance in Social Interaction and addresses the Organization Theory. DELTA stands for Discrete Event in Linear Time Agent and addresses the systems theory.

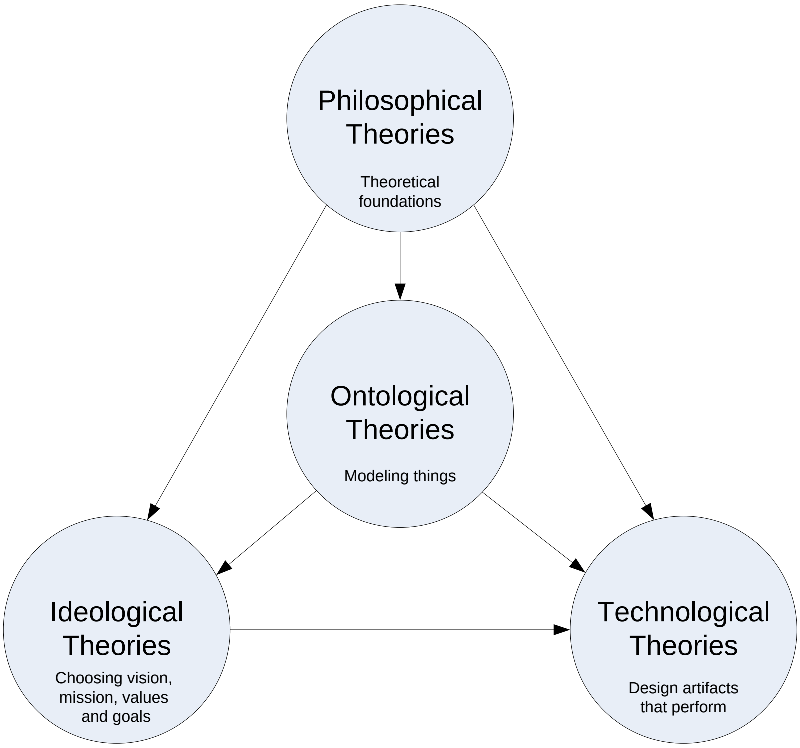

Figure 1 – Foundational Theories for EE, adapted from [23]

- Ideological theories are focused on the why we choose to do things in certain ways, based on our vision, mission, values and goals. At the Ideological level DEMO has the SIGMA theory. SIGMA stands for Socially Inspired Governance and Management Approach and addresses the governance and management theory.

- Technological theories are concern with the actual design of artifacts that perform as intended and on the methods to get to those results. At the Technological level DEMO has the MU and BETA theories. MU stands for Model Universe and addresses the model theory. BETA stands for Binding Essence, Technology and Architecture, and addresses the design theory.

Although these theories are interrelated and constitute a cohesive all, the main focus on this PhD program is on the PSI theory – the organization theory. It is on this ontological level that most theoretical contributions will take place, although on the practical side, when implementing the software artefact for validation, all theories will be used.

PSI Theory

A core idea of DEMO is that to model business interactions we should use a communication-centric approach, instead of the data-centric approach which is the dominant approach in the design of information systems.

Figure 2 - Action Workflow Loop [24]

The communication-centric approach has its roots in the Action Workflow Loop [24] presented in Figure 2, being "general and universal", models the core pattern of all successful interactions.

According to Denning and Medina-Mora [24], "Incomplete workflows invariantly cause breakdowns, and if they persist, they give rise to complaints and bad feelings that interfere with the ultimate purpose of work – to satisfy the customer."

DEMO extends this core loop through Performance in Social Interactions Theory (PSI) [1][25]. It describes the world through a model based on transactions, each producing a single result, initiated by a set of actor roles and executed by one particular actor role. This result is the simplified pattern presented in Figure 3 and 4 which uses a sequence of coordination acts surrounding a production (execute) act.

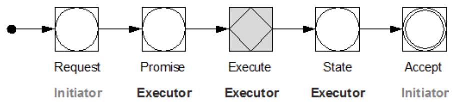

Figure 3 - Simplified pattern for a PSI transaction [1]

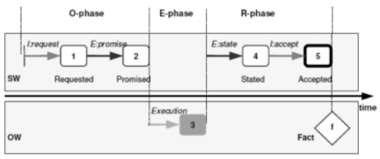

Figure 4 - Order, Execution and Result phases [26]

As depicted in Figures 3, 4 and 5, the transaction starts with a request (rq) by the initiator which includes the desired outcome in full detail. If the executor can fulfill that request, he will promise (pm) a delivery and then produce/execute the expected result and state (st) its completion to the initiator. Assuming that the delivered result is as requested, the initiator will finish the transaction by accepting (ac) the result, otherwise it can be rejected (rj). Therefore, this pattern assigns different acts to the initiator and the executor actor roles. These core acts can be split into three phases, as can be seen in Figure 3: order, execution and result [26]. In 2017 the state (st) act was renamed to declare act (de). There might appear both names in this work depending on the year of the content they reference.

This simplified description becomes more complex, as can be seen in Figure 5, as additional revoke acts are needed and so are added to each phase [25]:

- The ***initiator*** can change his mind and **revoke** the request (rv rq) at any time.
- The ***executor*** can **decline** (dc) the initial request if he does not wish, is not able, or can't deliver in the conditions requested by the initiator.
- The ***executor*** can **revoke** his previous **promise** act (rv pm).

- The ***executor*** can **revoke** his previous **state** act (rv st).
- The ***initiator*** may **reject** (rj) the **state**d (st) result.
- The ***initiator*** may **revoke** a previous **accept** (rv ac).

Revoking acts contradict previously established expectations. They may be initiated by any of parties and the counterparty may allow the revoke or refuse it.

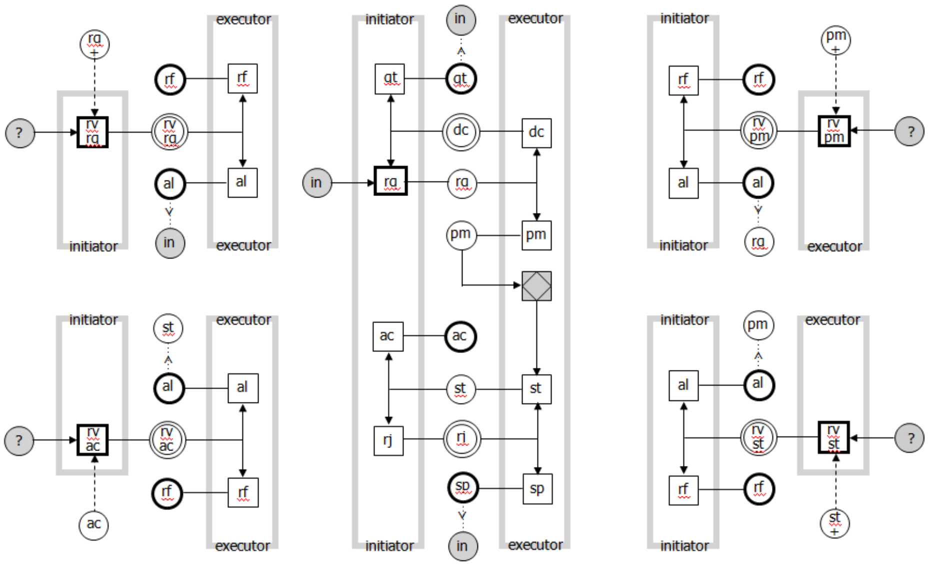

Figure 5 - DEMO 3.4 complete transaction pattern [7]

6 Theoretical Contributions to DEMO/PSI Theory

The author's contributions to the evolution of the DEMO/PSI Theory are the following:

1. In 2015, in the author's previous work [9], four proposals were presented to improve the DEMO/PSI transaction pattern. Three of those proposals addressed the stop and quit acts/states and the author's proposal was to remove them from the transaction pattern. In 2017 these three proposals were adopted by the EE community and those acts/states were removed from the transaction pattern, as reported in [26].
   The fourth proposal in [9] addressed the issue of allowed revokes for state and accept acts, as well as after the reject act, should lead to the promised state allowing the executor to execute and state again. Also, the allowed revokes for request and promise acts, should lead to the initial state. This proposal was presented again in the 2017 EEWC Technical Session on PSI Theory [2][7][28], but not adopted by the community. More on this topic bellow.

2. In 2015, in the author's previous work [9], the two reference implementations of DEMO, [29][30] and [31][32][33][34] were challenged by the author's as non-compliant with what was prescribed by the PSI Theory. At that time, it was not clear for the EE community that those implementations were not compliant. Since then, that position has become adopted in the EE community because of the author's work in [9].
3. In 2015, in the author's previous work [9], a deterministic state machine was produced, as well as the corresponding grammar in Backus-Naur form, to model what was then the understanding by the author of the full transaction pattern, in two variants - with and without the proposed changes by the author. Like the other implementations, addressed in the previous item, this solution did not fully comply with the prescribed PSI theory. Awareness was raised about the existence of different interpretations in the EE community on the diagram of the full transaction pattern.
4. In 2015, in critical evaluation of the author's work [9], five problems were identified as constraints to the proposed transaction pattern, although no solution was then provided to solve them:
   a. In many states of the state machine, only one of the actor roles can act. If he refused to act, that would block the transaction forever, which is not in accordance with Habermas ideal speech situation [35];
   b. The author's proposed solution for the state machine used a shared-memory approach (centralized approach);
   c. The request act is presumed to have all the details required for the two actor roles to complete the transaction, but the transaction pattern does not provide any clue on how that negotiation and agreement is reached. The same situation happens on the discussion states after a decline or a reject. This problem was referenced in the literature [16][17][18]. This is a big challenge for developing a software artefact based on DEMO, as that negotiation, if to be performed within the Information System must also be modelled. It was also questioned in [9] if that design constrain was adequate to reality as configuration details are many times postponed to delivery stage (state-accept).
   d. The fixed roles between initiator and executor actor roles were also identified as a constraint, i.e., the initiator is assigned to the acts request and accept, and the executor to the acts promise and state without any flexibility. This problem is referenced in the literature in [12][13]. Sometimes acts are performed by agency or delegation, but there is no common understanding on how those mechanisms can be brought into practice.
   e. PSI transactions definition assume that in a transaction there is a unilateral transfer of value. The executor transfers something of value for the initiator. Only through the full set of transactions the value network makes sense.
5. In 2015, in the author's previous work [9], the Tell&Agree pattern was proposed with a state machine that is simpler (less acts, less states) [36] than the DEMO/PSI pattern for a simpler problem: How to model a negotiation without the requirement for shared memory? It takes advantage of tell and agree acts, and acknowledge to those acts, to implement a two-phase commit on an agreement

that enable a communication centric protocol that does not require shared memory.

To model negotiations we used a state chart where for each message exchanged, both actors would keep two instances of the Tell&Agree pattern: one that contains his own acts, and another that contained the counter party acts.

This pattern also introduced an unexpected benefit – the ability to model agreements where more than two participants taking part, which was discussed in that work [36].

This Tell&Agree pattern is an evolved pattern when compared with the two-phase commit that is so common in database operations. It introduces the ability to handle more than two parties and revoke commitments also based on agreement between the parties.

The main drawbacks for Tell&Agree are:

- Tell&Agree is able to model the phases (order, execution, result) of a DEMO/PSI protocol independently with distinct messages, but there isn't a clear relation/dependency between those messages. This flexibility however, allows this model to provide a solution to handle change, that is, to reach new agreements that modify the previous ones without impacting the execution and result phase.
- The Tell&Agree protocol was represented by a non-deterministic state machine. The author was not able to transform it into a deterministic state machine, nor presented it as a BNF grammar.

6. In 2016 at the 16th CIAO! Doctoral Consortium, in Madeira, the author presented the work *Core Components of Communication* (CCC) [10]. In that work we proposed a unification of DEMO/PSI and Tell&Agree. We have done so based on the finding that there were a set of common core components that always exist in communication like tell acts, ask acts, agree acts. Both Tell&Agree and DEMO/PSI were modelled in this work using the CCC. The Tell&Agree was modelled with just two CCC and the DEMO/PSI with eight CCC.

   The CCC allows the modelling of business processes using a higher layer of abstraction. Each component in CCC has its own state machine with internal states and possible acts from those states, which can only be seen if we look at it as a white box model. When we look at each CCC in a black box perspective we only have one initial state, and several possible result states. There are a few generic CCC, but many possible configurations for them. The most important configurations are the existence or not of certain result states, and if those result states lead to a new state, or get back to the original initial state, after adding the new fact to the database.

   Using the CCC to model DEMO/PSI fully matched what was required by the theory, however there was a problem regarding the request and state acts that where hidden within the state machine, and therefore not a visible state when viewed as a black box. The solution that we did not realized back then is rather simple – to use a grey box, where the internal states that correspond to the request and state are made visible from the outside, but do not correspond to a state that makes the business transaction move forward, as they are an intermediary step.

   We have use the metaphor of puzzle pieces, where each piece has one intrusion – the initial state, and several possible result states as extrusions. We realize that the puzzle pieces as a metaphor has its constraints. It is a 2D approach, which makes it easy to print. It provides the affordance of combining the puzzle pieces. However, it also requires hubs to allow several puzzle pieces to be mounted in the same result state, and that, we have now found, consistently creates confusion in the interpretation of the diagrams, because it breaks the metaphor.

7. In 2017 the author produced the paper DEMO/PSI and the Law of the Land [8] work we analysed two laws, one section of Portuguese Contract Law in Civil Code, and the European Common Sales Law. Looking at the mandatory requirements imposed by these laws, we could identify several problems with the existing DEMO/PSI theory. Some of the requirements are very hard, if not impossible, to properly model with current version of DEMO/PSI theory and with existing Core Components of Communication. Some of the identified components were already suggested on previous work [10].
8. In 2017 the author produced the paper Modeling Exchange Agreements with DEMO/PSI and Core Components of Communication [14]. In this work a concrete and real world example was studied – the exchange agreement, where two actors exchange their home for vacations. Both parties agree on the terms of their exchange, and both parties are executors of their part of the deal within the context of the same exchange agreement. The work fully analyses the possibilities of modelling this business transactions with DEMO/PSI, concluding that node of them provides a useful modelling for this business transaction. The same problem is modelled successfully with Core Components of Communication.
9. The author presented several proposal for improving the DEMO/PSI Theory on the 2017 EEWC Technical Session on PSI Theory [27][27].
10. In 2018, in this conference, the author presented two papers "Modeling the system described by the EU General Data Protection Regulation with DEMO" [39] and "Colored Petri-Net for Implementing DEMO/PSI Transactions for N Actor Roles (N>=2)" [40]. As these works will be presented in this conference we will not summarize them here.

7 Plan for Remaining Research and Validation

In this section we summarize the strategy and the plan for trying to achieve successful answers for each of the five research questions.

For the first research question, regarding modelling DEMO / PSI transactions that enables its use in asynchronous and distributed networks we rely on the Petri-Net model presented in 2018 in this conference [40]. A specific prototype is being developed for its initial validation, that will be followed with its use in the other artefacts described below, also with the intent of achieving validation.

The second research question, regarding the use of more than two actor roles in a DEMO/PSI transaction is also based on the same work [40]. The artefacts used for validating this hypothesis are a subset of the artefacts to be used in validation, because only two of then have transactions with more than two actors, namely, the one that are based on the previous works regarding Exchange Agreements [14] and General Data

Protection Regulation [39]. The prototype mentioned on the first research question can also be used as initial validation for this second research question.
Regarding the third research question, with the minimal set of communication protocols to model the most common conversations, although they have been mentioned before, we still miss some systematization. As a report on the work in progress we shall present in a very succinct way the six conversations we believe are the adequate hypothesis for getting a positive answer to this question. Those conversations are: **Tell**, **Ask**, **Result**, **Agree**, **Choose** and **Plan**.

- The **Tell conversation** is the simplest one, with only two possible acts in sequence: the Tell act and the Acknowledge act. The Tell conversation has one sender and can have one or more receivers. Each receiver independently acknowledges the tell act with its contents. The Acknowledge act for the Tell conversation is not mandatory. A Tell act can be configured to produce changes in the Petri-Net, either on the Tell Act or on the Acknowledge act, namely adding or removing tokens (black or colored) to specific states (the Permit and Configure states). Check [40] for additional references. These states do not modify the inter-social state of a transaction but enable (or disable) the possibility of having actor roles performing acts. Tell acts can be scheduled to be performed automatically with pre-authorization at a predefined time, or after a certain time has passed after a certain act in the Petri-net has been performed. Scheduled act can be cancelled by the actor role that authorize them.
- The **Ask conversation** can be used by one of the actor roles to provide clarifications. It can be used for configuration issues, or late decisions on how to provide a result. This addresses the forth research question and will be address bellow. The Ask conversation has five acts: Ask, Acknowledge Ask, Answer, Acknowledge answer and Result. Contrary to the Tell conversation, Acknowledge acts in the Ask conversation are mandatory. The Ask conversations can be established with one or more actor roles. An Ask conversation is not just a sequence of Tell conversations, as the sender may configure several questions with closed or open answers, and require the results to be analysed with descriptive statistics (an implicit Infological transaction). The Result act is not mandatory, as the Ask conversation might be configure to have private results, or to send them to participant actor roles when chosen by the asker actor role.
- The **Result conversation** corresponds to the Petri-Net presented in [40] regarding the Declare-Accept portion of the current DEMO/PSI transaction. The portion that corresponds to the Request-Promise, is the **Agree conversation**. The agree conversation is also based on previous work [9] the Tell&Agree protocol. The split between the order phase of the transaction and the result phase have also been address in previous work, namely on 2017 EEWC Technical Session on PSI Theory [27][28]. When a Result conversation reaches an accepted state, or when an Agree conversation reaches a promised state there might by configured changes to states in the Petri-Net, just like described in the Tell conversation.
- The **Choose conversation** allows participants to choose, or vote. Choosing is a more complex conversation than just asking, as the goal of the transaction might not just get an opinion in secrecy, but can also be an attempt to reach a consensus among participant actor roles. Participants may change their previous choice as the debate is happening with Tell conversations within the context of choose

conversations. The existing of this contextual conversations is also optional by configuration. Therefore in a Chose conversation participants might be notified of each participant decision, or just that one has chosen, depending on the configuration options. The actor role that starts the choosing conversation can also have the right to choose, or just do it as tie-breaker if needed. Choose conversation has many other detailed options that have already been addressed in previous work [10] – Core Components of Communication. Decisions might be achieved by unanimous consent or using a configurable majority. Just like in the Tell conversation, the predefined resulting choice might be charged with "intructions" to add or remove tokens from certain states. This will be used in the artefact for modelling Exchange Agreements, presented in previous work [14].

- The **Plan conversation** is a bundle of Result, Agree, Tell and Choose conversations, globally encompassed by an Agree conversation. Participant actor roles, that can be one or more, add and configure the inner conversations, dependencies among them and even changes in the Petri-Net as certain acts occur. A Plan conversation be looked at as a mix of ATD and PSD in PSI transactions, as they set actor roles, conversations, and dependencies among them. By allowing several Result conversations within the same plan conversation we can combine distinct Result conversation within the same logical plan. This allows, for example, to combine the delivery of a service and its payment as Result conversations where actor roles occupy opposite positions on the transfer of value. Plans can be instantiated with different actor assigned to the actor roles. A Plan conversation will be used to implement the Exchange Agreement based on previous work [14]. The option of modelling Plan conversation has to do with the fith research question, as discussed below.

The forth research question, regarding loosening the constraints that currently exist on the request act in the DEMO/PSI transaction. By splitting the ordering phase (Agree conversation) from the result phase (Result conversation), as well as having Tell and Ask conversations at anytime within the context of both the Agree and Result conversation, we provide a much higher level flexibility, allowing to reschedule acts, delegate or changing actors in actor roles (with agreement by counterparty), and other more flexible configurations.

The fifth research question addresses the problem of change in a cascade of transactions. Through the split of the order phase from the result phase, with distinct transaction results for each conversation, we can revoke an agreement without having to revoke the results that have already been produced. We believe that this is actually a combinatorial effect as the ones predicted in the Normalized Systems Theory [41][42]. Revoking an agreement may lead to an initial state of a Plan conversation, but already starting with a context. Some conversations might be needed to add, like compensations and remedies, others might be just removed, especially if they were not started, and other should be stored as historic. When a new plan is in position to be approved, the corresponding agreement conversation of the plan is set on motion, and when approved, the new set of conversations can be used. Notice that even when an agreement is revoked, the participants can continue to work on the conversations of previous plan (unless there is an agreement to remove those conversations from production). This also allows a much higher degree of flexibility.

Validation Plan

The validation plan for this PhD program includes several tasks corresponding to different artefacts with an increasing level of complexity.

After concluding the development of the prototype for the Petri-Net presented in [40], we shall implement simple artefacts from the DEMO literature like the Pizzeria, the Library and the Rent-a-Car examples. This should address the purpose of validation by Correspondence Theory – matching the expectations in the literature of Enterprise Engineering research field.

Them we shall develop a conceptual framework, using Ulrich Frank definition on [21], for a general purpose organization, addressing many implementation challenges and options, namely those described in previous work [37][38][39]. This conceptual framework will then be reused for future artefacts, using this common base.

Then we develop three additional artefact, with a higher degree of complexity, and validate them with persons on real business context, using interviews and observation of their performance on prepared tasks after initial presentation of the artefact. The three real world are the Speech Club, the Exchange Club and the General Data Protection Regulation for a more general scope of organizations.

8 Conclusions

This is still a work in progress, currently in the development stage for the artefacts that will be used for validation. This document is a report on current progress. At current stage there is a limited number of valid research findings that can be reported.

The theoretical contributions already presented in previous work provides the author with the motivation for the remaining implementation and validation steps that should lead to the completion of this PhD program.

The major contribution aimed by this work is to improve DEMO/PSI transactions, namely: a) to improve the transaction pattern so that it can be used in asynchronous and distributed systems; b) to allow more than two actor roles in the transaction pattern; c) to add the additional conversation patterns that are currently omitted in the transaction pattern; d) to add flexibility on the constraint that request act must contain all the required information for the transaction; e) to improve the change problem on a cascade of transactions by reducing its impact.

arditi agência regional para o desenvolvimento da investigação tecnologia e inovação

This work was developed with financial support from ARDITI (Agência Regional para o Desenvolvimento da Investigação, Tecnologia e Inovação), in the context of project M14-20 09-5369-FSE-000001- Bolsa de Doutoramento.

References

1. Dietz, J.L.G. (2006) Enterprise Ontology – Theory and Methodology
2. http://www.bpmn.org/
3. http://pubs.opengroup.org/architecture/archimate2-doc/

4. Bock, A., Kaczmarek, M., Overbeek, S., & Heß, M. (2014, November). A Comparative Analysis of Selected Enterprise Modeling Approaches. In *PoEM* (pp. 148-163).
5. Lankhorst, M. (2017). *Enterprise Architecture at Work: Modelling, Communication and Analysis* (The Enterprise Engineering Series).
6. (2017) Delta theory (Discrete Event in Linear Time Automaton) - Extended Summary - version 3.2
7. Dietz, J.L.G (2017) The PSI theory – understanding human collaboration (v3.3)
8. Gouveia, D., & Aveiro, D. (2017) DEMO/PSI Theory and the Law of the Land.
9. Gouveia, D., & Aveiro, D. (2015) Two Protocols for DEMO Engines: PSI or Tell&Agree. *CIAO DC.*
10. Gouveia, D., & Aveiro, D. (2016) Core Component of Communication. *CIAO DC.*
11. Lind, M., & Goldkuhl, G. (2002, June). Questioning two-role models or who bakes the pizza. In *Seventh International Workshop on the Language-Action Perspective on Communication Modeling* (LAP 2002) (p.44).
12. Goldkuhl, G. (2007). Beyond Communication Loops–Multi-Responsive Actions in Business Processes. *Systems, Signs & Actions, An International Journal on Communication, Information Technology and Work*, *3*(1), 9-24.
13. Goldkuhl, G. (2003). Conversational analysis as a theoretical foundation for language action approaches. In *Proceedings of the Eighth International Conference on the Language Action Perspective on Communication Modeling, LAP-2003, Tilburg, The Netherlands* (pp. 51-69).
14. Gouveia, D., & Aveiro, D. (2017) Modeling Exchange Agreements with DEMO/PSI and Core Component of Communication.
15. Searle, J. R. (1969). *Speech acts: An essay in the philosophy of language* (Vol. 626). Cambridge university press.
16. Medina-Mora, R., Winograd, T., Flores, R., & Flores, F. (1992, December). The action workflow approach to workflow management technology. In *Proceedings of the 1992 ACM conference on Computer-supported cooperative work* (pp. 281-288). ACM.
17. Rittgen, P. (2007, January). Negotiating models. In *Advanced information systems engineering* (pp. 561-573). Springer Berlin Heidelberg.
18. Weigand, H., & De Moor, A. (2001). A framework for the normative analysis of workflow loops. *ACM Siggroup Bulletin*, *22*(2), 38-40.
19. von Alan, R. H., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS quarterly*, *28*(1), 75-105.
20. Gregor, S., & Hevner, A. R. (2013). Positioning and presenting design science research for maximum impact. *MIS quarterly*, *37*(2), 337-355.
21. Frank, U. (2006). *Towards a pluralistic conception of research methods in information systems research* (No. 7). ICB-research report.
22. Dietz, J. L., & Hoogervorst, J. A. (2011). Enterprise engineering manifesto. *Advances in Enterprise Engineering I. LNBIP, 10.*
23. Dietz, J. L., Hoogervorst, J. A., Albani, A., Aveiro, D., Babkin, E., Barjis, J., ... & Winter, R. (2013). *The discipline of enterprise engineering. International Journal of Organisational Design and Engineering*, 3(1), 86-114.
24. Medina-Mora, R., Winograd, T., Flores, R., & Flores, F. (1992, December). The action workflow approach to workflow management technology. In Proceedings of the 1992 ACM conference on Computer-supported cooperative work (pp. 281-288). ACM.
25. Dietz, J.L.G (2009) DEMO-3 Way of Working
26. Van Reijswoud, V. E., Mulder, H. B., & Dietz, J. L. (1999). Communicative action-based business process and information systems modelling with DEMO.Information Systems Journal, 9(2), 117-138.
26. (2017) PSI theory - Extended Summary - version 3.2
27. 2017 EEWC Technical Session on PSI Theory (session 1)

https://www.researchgate.net/publication/325050369_2017_EEWC_Technical_Session_on_PSI-theory_day_1 (last visited 2018/05/10)
28. 2017 EEWC Technical Session on PSI Theory (session 2)
https://www.researchgate.net/publication/325050437_PSI-theory_Technical_Session_Day_2 (last visited 2018/05/10)
29. www.demoworld.nl
30. Van Kervel, S. J. H. (2012). *Ontology driven Enterprise Information Systems Engineering* (Doctoral dissertation, TU Delft, Delft University of Technology).
31. Op't Land, M., Krouwel, M. R., van Dipten, E., & Verelst, J. (2011). Exploring normalized systems potential for dutch mod's agility. In *Practice-Driven Research on Enterprise Transformation* (pp. 110-121). Springer Berlin Heidelberg.
32. Krouwel, M. R., & Op't Land, M. (2011). Combining DEMO and Normalized Systems for developing agile enterprise information systems. In *Advances in Enterprise Engineering V* (pp. 31-45). Springer Berlin Heidelberg.
33. Huysmans, P., Bellens, D., Van Nuffel, D., & Ven, K. (2010). Aligning the constructs of enterprise ontology and normalized systems. In *Advances in Enterprise Engineering IV* (pp. 1-15). Springer Berlin Heidelberg.
34. Van Nuffel, D., Mannaert, H., De Backer, C., & Verelst, J. (2010). Towards a deterministic business process modelling method based on normalized systems theory. *International journal on advances in software*, *3*(1 and 2), 54-69.
35. Habermas, Jurgen. "Discourse Ethics: Notes on a Program of Philosophical Justification." Moral Consciousness and Communicative Action. Trans. Christian Lenhart and Shierry Weber Nicholson. Cambridge: MIT Press, 1990, pp. 43-115.
36. Prototype for validation of several DEMO Engines: Tell&Agree, DEMOWorld, Normalized Systems, Current DEMO/PSI grammar e Improved DEMO/PSI grammar.
http://www.duarte-gouveia.info/protocols/ (last visited 2018/05/10)
37. Gouveia, D. (2017) Towards implementation of DEMO models as Software Artefact. *CIAO DC.*
38. Gouveia, D., & Aveiro, D. (2016) Things, References, Connectors, Types, Variables, Relations and Attributes – A Contribution to the FI and MU Theories.
39. Gouveia, D., & Aveiro, D. (2018) Modeling the system described by the EU General Data Protection Regulation with DEMO
40. Gouveia, D., & Aveiro, D. (2018) Modeling the system described by the EU General Data Protection Regulation with DEMO
41. Mannaert, H., & Verelst, J. (2009). *Normalized systems: re-creating information technology based on laws for software evolvability*.
42. Mannaert, H., Verelst, J., Bruyn, P. (2016). *Normalized Systems Theory – From Foundations for Evolvable Software – Towards a General Theory for Evolvable Design*, Koppa.