

ON THE COUPLING OF ARCHITECTURES LEVERAGING DEMO THEORY WITHIN THE ARIS FRAMEWORK

Danielle Strijdhartig BSc.
D.J.Strijdhartig@student.tudelft.nl
1053892

University:
Delft University of Technology

Faculty:
Electrical Engineering, Mathematics and Computer Science

Organization:
Rijkswaterstaat
Data-ICT-Dienst

Supervisor:
Prof.dr.ir. Jan L. G. Dietz, Delft University of Technology
Drs. Aart Vermetten, Rijkswaterstaat

September, 2008




TU Delft

This page is intentionally left blank.

Abstract

This thesis project has been carried out within the Data and ICT Service Centre of the Directorate-General for Public Works and Water Management (Rijkswaterstaat).

The project is aimed at exploring how architectures that are based on the DEMO methodology and the ARIS framework can be consistently interconnected in practice. The architectures that were used are the Dienstverleningsmodellen (DVL) and the Uniforme Primaire Processen (UPP) of Rijkswaterstaat, which are based on DEMO and ARIS respectively. The research culminates in the development of a consistent coupling between these two architectures.

The project is defined by the following question:

“How can DVL and UPP models be coupled in a consistent way?”

The following results of the project provide an answer to the research question:

- A description of the current state of coupling
- A description of the consistent state of coupling
- A method to achieve the consistent state of coupling
- Guidelines to maintain the consistency of the coupling

Prof. dr. ir. J.L.G. Dietz Delft University of Technology
Electrical Engineering, Mathematics and Computer Science

Dr. ir. J. van den Berg Delft University of Technology
Technology, Policy, and Management

Ir. H.J.A.M. Geers Delft University of Technology
Electrical Engineering, Mathematics and Computer Science

Dr. B. Knikman Directorate-General for Public Works and Water Management
Data-ICT-Dienst

Preface

The research project, of which this document is the result, was conducted as the finalization of the Master program Information Architecture at the Delft University of Technology.

First and foremost I would like to thank my supervisors for all their help and guidance. This report was not possible without the help of Prof.dr.ir. Jan Dietz my supervisor at the Delft University, and drs. Aart Vermetten my supervisor at Rijkswaterstaat.

I would like to express my gratitude to everyone who helped me complete my thesis work especially the attendees of my presentations for their contributions, which gradually improved the quality of my results. I would like to thank them for their patience, as my presentations were very long.

My thanks also go to dr. Martin Op 't Land of Capgemini whose practical inputs and discussions proved invaluable. I would very much like to thank drs. Ben Knikman of Rijkswaterstaat's Data-ICT-Dienst for all his constant help and unwavering support. And also drs. Martin Lemmen and ir. Fred Mulkens from Rijkswaterstaat's Corporate Dienst for their help and guidance on UPP.

For his guidance during the evaluation process, I thank Prof.dr.ing. Hans Mulder of VIAGroep. For their participation in the thesis committee I thank Ir. H.J.A.M. Geers and Dr. ir. J. van den Berg of the TU Delft. Some additional thanks to Dr. ir. J. van den Berg for his contribution of remarks on improving the quality of this thesis. Special thanks go to Ashna Rajaram BSc. for her openness and encouragement during the thesis project.

My appreciation would be incomplete without thanking my family and friends for all their support during my time at the TU Delft.

Danielle Strijdhartig
Delft, September 08, 2008

Table of Contents

1	Introduction	7
1.1	Conventions.....	7
1.2	Dutch Terminology	7
1.3	Project Context.....	8
1.4	Research Questions.....	9
1.5	Research Methodology and Report Structure	9
	Phase ONE – Seeking the Connection	11
2	Theoretical Background	12
2.1	DEMO Theory	12
2.2	ARIS Theory	14
3	Model Types.....	15
3.1	Dutch Terminology	15
3.2	DEMO	16
3.3	DVL Models.....	18
3.4	ARIS.....	19
3.5	UPP Models	21
3.6	Conclusion	23
4	Meta Object Types.....	25
4.1	DEMO Meta Object Types	25
4.2	ARIS Meta Object Types	27
4.3	Conclusion	30
5	The Substitution Test	31
5.1	Generating UPP Layer Four	31
5.2	Generating UPP Layer Five	33
5.3	Authentic UPP Process Logic	35
5.4	Conclusion	37
6	Analysis of Issuing Permits.....	38
6.1	Dutch Terminology	38
6.2	Delineation of Analysis Area	38
6.3	The DVL Model for Issuing Permits	40
6.4	The UPP Model for Issuing Permits	43
6.5	Correlation between the Models	48
6.6	Conclusion	50
	Phase TWO – Building the Connection.....	51
7	Method of Coupling	52
7.1	The 5-Stage Coupling.....	52
7.2	Benefits of the Coupling	57
7.3	Achieving Stage 1 and 2.....	58
7.4	Achieving Stage 3 and 4.....	65
7.5	Achieving Stage 5	69
7.6	Guidelines for Maintaining the Coupling	70
7.7	Conclusion	71
	Phase THREE – Assessing the Connection	72
8	Evaluation	73
8.1	The GDSS Session	73
8.2	Participants of the GDSS Session.....	73
8.3	Opportunities and Barriers.....	74
8.4	Technical Guidelines.....	77
8.5	Actions for the Realization	79
9	Discussion and Recommendations	80

9.1	Motivation for Coupling	80
9.2	Recommendations	80
10	Conclusion	81
11	Abbreviations.....	83
12	Appendices	84
12.1	DEMO Transaction Pattern	84
12.2	The Pizzeria Case.....	88
12.3	Proposed UPP models.....	89
12.4	GDSS Evaluation Output.....	96
13	Bibliography.....	116

1 Introduction

In this chapter I introduce the writing style conventions that I have applied throughout the thesis report. Next I explain the context in which the research has taken place. Then I introduce the research questions that the thesis attempts to answer. And last I explain the research methodology and the structure of the thesis report.

1.1 Conventions

Some of the readers of this document will be more accustomed to Dutch organization names, while others will be more comfortable with the English names. For convenience, each chapter that contains Dutch terms starts with a terminology section. This section contains a Dutch to English listing of the terms that are introduced in that chapter. In the remainder of the report only the Dutch terms are used but their abbreviations are used in place of the full names as much as possible. Dutch terms are given a different font face from the English text to make them more easily recognizable.

1.2 Dutch Terminology

Ministerie van Verkeer en Waterstaat (V&W)
Ministry of Transport, Public Works and Water Management

Rijkswaterstaat (RWS)
Directorate-General for Public Works and Water Management

Inspectie Verkeer en Waterstaat (IVW)
Transport, Public Works and Water Management Inspectorate

Koninklijk Nederlands Meteorologisch Instituut (KNMI)
Royal Netherlands Meteorological Institute

Staf DG
Directorate-General staff

Data-ICT-Dienst (DID)
Data and ICT Service Centre

Enterprise Architectuur & Consultancy (EAC)
Enterprise Architecture & Consultancy

Dienstverleningsmodellen (DVL)
Service Provision Models

Corporate Dienst (CD)
Corporate Service Centre

Beheerorganisatie Processen en Systemen (BPS)
Management Organization of Processes and Systems

Uniforme Primaire Processen (UPP)
Uniform Primary Processes

1.3 Project Context

The Ministerie van Verkeer en Waterstaat (V&W)ⁱ is one of the thirteen ministries which comprise the national government. The main goals of V&W are to protect the Netherlands against flooding and water erosion, and to ensure secure connections of international quality. The ministry consists of policy departments, executive departments, and three autonomous agencies namely:

- Rijkswaterstaat (RWS)ⁱⁱ
- Inspectie Verkeer en Waterstaat (IVW)ⁱⁱⁱ
- Koninklijk Nederlands Meteorologisch Instituut (KNMI)^{iv}

Of these three only RWS is of interest; RWS is concerned with the practical execution of the ministry's directives, mainly the construction and maintenance of roads and waterways. For a detailed description of the RWS organization see [RAJARAM].

The head of the organization consists of the management, the Directorate-General, the substitute Directorate-General and the Chief Financial Officer. They are supported by the Directorate-General staff (Staf DG).



The organization itself consists of the head, the Directorate-General staff, five national service centres, ten regional service centres, and three project management directorates. Two of the national service centres are of importance for the research project, namely the Data-ICT-Dienst and the Corporate Dienst.

The Data-ICT-Dienst^v is the knowledge and services centre of RWS and V&W. The Corporate Dienst^{vi} is the division that centrally supports the Directorate-General staff, the regional services and the national services, in terms of finance, legislation, real estate, facility management, HRM and communication.

On the side of the Data-ICT-Dienst there is the Enterprise Architectuur en Consultancy (EAC) group, that makes use of Dienstverleningsmodellen (DVL models) based on the Design and Engineering Methodology for Organizations (DEMO)^{vii}. On the side of the Corporate Dienst there is the Beheerorganisatie Processen en Systemen (BPS) that makes use of Uniforme Primaire Processen (UPP models) that are based on the Architecture of Integrated Information Systems (ARIS)^{viii}.

Both RWS architectures successfully fulfill their purposes. Hereby an architecture is taken to mean a set of models, including textual descriptions, by which the fundamental structure of an organization and its processes are captured. The DVL models were created to carry out the Application Portfolio Rationalization Project at RWS [OP 'T LAND ET AL.]. Since then they have been used to execute many more architectural projects, such as the Splitting of RWS/Deltares [OP 'T LAND, DIETZ]. The UPP models document those work processes that are uniform across V&W. Because of their practical nature, UPP models are used daily by V&W employees all across the Netherlands to aid them in executing their

work. Because of their successes both architectures enjoy a large amount of support within RWS and V&W. It is therefore expected that both architectures will continue to exist alongside each other for the foreseeable future.

Because DVL and UPP are about the same organization, and about related things of the organization, an expectation has grown that connecting DVL and UPP could be beneficial to both sides. It was also expected that such a connection could readily be made. My master's thesis project was carried out in response to this expectation, with the goal of documenting the DVL-UPP connection.

1.4 Research Questions

How can DVL and UPP models be coupled in a consistent way?

Consistency can be defined as *the lack of internal contradiction*¹, or as *the logical coherence among things or parts of things*². For the purpose of this research, consistency is defined as *the logical coherence among models and the object types within these models*.

The sub-questions that are asked in order to construct the answer for the main research question are:

- What is the current state of coupling between DVL and UPP?
- What is a consistent state of coupling between DVL and UPP?
- How can the consistent state of coupling be achieved?

9

By the state of coupling is meant the degree of logical coherence among the models and object types that are meant to be interconnected. In order to answer the main research question we must first know the current state of coupling. Then, we must make clear for ourselves what a consistent state of coupling entails; we must state in tangible terms how the two models and their object types should correlate in order to be considered consistent. The third question is the final piece of the puzzle. It is about stating in actionable terms what should be done to the models of DVL and UPP in order to arrive at the consistent state of coupling.

1.5 Research Methodology and Report Structure

The first phase of research was focused on deriving UPP models from their DVL counterparts. Given a DVL model I would substitute UPP object types for the object types present in DVL, in order to generate a matching UPP model from it. The goal was to develop a set of conversion rules that could be applied to all DVL models to generate their matching UPP models. Substitutions from UPP to DVL were also performed to see if the conversion process was bi-directional. This phase was carried out by studying literature on DEMO, ARIS, UPP and DVL, by conducting talks with EAC en BPS, and by testing various substitutions on a range of models. Results from this phase revealed that deriving UPP models from DVL models was possible in theory; one could substitute UPP object types for DVL object types to a degree. However, the UPP models that resulted did not resemble the UPP models that they were meant to match; the level of granularity of the generated UPP models was coarse in comparison to that of the actual UPP models. Substituting DVL object types for UPP object types also did not generate the expected DVL models; the object types in UPP used labeling schemes that did not

¹ <http://vandale.nl>

² <http://answer.com>

adequately transfer ontological meaning to DVL. Phase one is presented in chapters 2 up to and including chapter 6.

As a simple conversion between DVL and UPP did not seem effective, the second phase of research focused on using the strengths of UPP to compensate for the gaps in DVL and vice versa. Given a DVL model I would try to connect the parts of DVL to the more detailed counterparts in UPP. This was done to connect infological and datalogical information present in UPP to the ontological information present in DVL. At the same time this connection strengthened ontological understanding of UPP. This phase was carried out by analyzing and deconstructing a DVL model and a UPP model that both regarded the same area of the V&W organization. Steps were developed to match DVL and UPP models and make them consistent with each other through application of the DEMO theory. Phase two is presented in chapter 7.

The final phase of research focused on evaluating the resulting method of coupling DVL and UPP. The evaluation focused on the opportunities and barriers to applying the method. It focused on the technical guidelines of the method. It also focused on the feasibility and value of applying the method. This phase was carried out by presenting the method to EAC and BPS and letting them evaluate it in a group decision supported environment. Phase three is presented in chapter 8, chapter 9 holds my recommendations for further steps and chapter 10 holds my conclusion regarding the research questions.

Phase ONE – Seeking the Connection

2 Theoretical Background

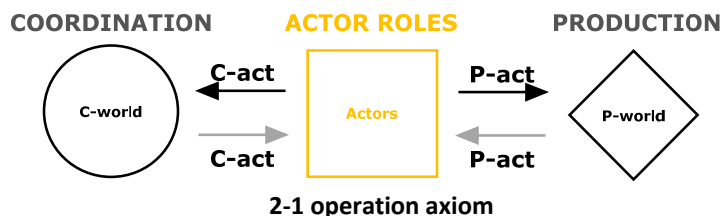
In this section the reader is introduced to the theoretical foundations of DEMO and ARIS. The theory discussed here is also the foundation of the coupling that is introduced in chapter 7.

2.1 DEMO Theory

DEMO stands for Design and Engineering Methodology for Organizations^{ix} and is based on the assumption that, to truly understand an organization, one must expel all implementation issues such as systems and applications. This is because DEMO is based on the PSI-theory (Performance in Social Interaction) of which the goal is to extract the essence of an organization from its actual appearance. PSI-theory's operation, transaction, composition and distinction axioms are used. Hereby the axioms are understood to be self-evident truths or principles that can be accepted as true without prior proof as the basis for argument.

Operation Axiom

The operation axiom regards the operation of enterprises. In PSI-theory the operation of an enterprise is constituted by the activities of actor roles. Hereby an **actor role** is an elementary chunk of authority and responsibility fulfilled by a subject and an **actor** is a subject in its fulfillment of an actor role. In fulfilling their actor roles, subjects may perform two kinds of acts namely: 1) production acts and 2) coordination acts. **Coordination acts** (C-acts) are communicative commitments related to a production act. **Production acts** (P-acts) are acts through which subjects contribute to bringing about goods and/or services that are delivered to the environment of the enterprise. Hereby the word *contribute* plays on the notion that PSI-theory, while necessary, is not sufficient to design and engineer enterprises; much more is required that PSI-theory does not cover. The result of a successful production act is a **production fact** (P-fact). Likewise, the result of a successful coordination act is a **coordination fact** (C-fact). Facts can be created but they cannot be undone. To rectify a mistake one can merely create an opposing fact to cancel out the negative effect. The set of P-facts that have been created up to a certain point in time are said to indicate the state of the P-world. Similarly the set of C-acts, that have been created up to a certain point in time, indicate the state of the C-world. The creation of a P-fact or C-fact leads to a **state transition** within, respectively, the P-world or C-world. The creation of an instance of a P-fact or C-fact causes an **event** in the P-world or C-world.



Transaction Axiom

An **enterprise** is a system of actors who perform C-acts and P-acts. These acts are structured into universal patterns called **transactions**. The transaction axiom regards these universal patterns. A transaction always takes place between two actor roles and consists of two conversations about a production act. The first conversation is about requesting and promising to perform a P-act. The second conversation is about stating

and accepting that the P-act has indeed been performed as was agreed. A transaction thus occurs in three phases. First, the order phase (O-phase) in which the first conversation takes place. Second, the execution phase (E-phase) where the production act is actually performed. And third, the result phase (R-phase) in which the second conversation takes place. Besides the basic pattern there is also a standard pattern which includes steps for declining a request and for rejecting a statement. And next to the standard pattern there are also cancellation patterns for requests, promises statements and acceptances. When these patterns are added to the standard pattern, then the complete pattern is achieved.

Composition Axiom

The composition axiom states that transactions can themselves be composed of transactions. A transaction can be initiated in three ways, namely: 1) enclosed in another transaction, 2) by an actor role in the environment, or 3) through self-activation. The relationships between transactions form a tree structure, which is broken down from left to right. When the executor of some transaction performs a request for a second transaction, this second transaction becomes an enclosed transaction of the first.

Distinction Axiom

PSI-theory states that actors have three distinct human abilities, the *forma*, *informa* and *performa* abilities. At the lowest level there is the *forma* ability. On the coordination side this ability is about ensuring that both the initiator and the executor agree on the language in which and/or the channel through which they are going to communicate. One can choose to *listen* to a customer or *read* an order form. The coordination acts on this level are called formative acts. On the production side the *forma* ability is about gathering and distributing data. The production acts here are called datalogical acts. These are P-acts such as storing transmitting, copying and destroying data and documents. There is an important difference between data and information. Data is raw whereas information only comes into existence when data is interpreted by someone. This leads to the next level and the *informa* ability. On the coordination side this ability is about synchronizing the thoughts of the initiator and executor, so that both of them understand what the other is thinking. Where the *forma* ability deals with “that they are speaking, or that they are writing”, the *informa* ability deals with “what they are saying, or what they have written”. The coordination acts on this level are called informative acts. Examples of such C-acts are formulating and interpreting thoughts. It is thus on this level that information comes into existence. On the production side this ability regards reasoning, computing, deriving and reproducing remembered knowledge. The production acts on this level are called infological acts. On the highest level stands the *performa* ability. On the coordination side this ability is about showing commitment. Coordination acts on this level are called performative acts. On the production side it is about creating new, original things in the form of products and/or services. On the production side it is the *performa* ability that is essential for doing business, and the corresponding acts are called ontological acts.

The Organization Theorem

The four axiomes discussed above are combined into one concise, comprehensive, coherent, and consistent notion of enterprise, namely the organization theorem. The organization theorem states that the organization of an enterprise is a heterogeneous system that emerges from the layered integration of three homogeneous systems: the B-organization (Business), the I-organization (Intellect), and the D-organization (Document). Each of the three homogeneous systems represent a different aspect of the total

organization. And each aspect contains a function and construction side. The construction of the D-organization is supported by the function of hardware. The construction of the I-organization is supported by the function of the D-organization. And the construction of the B-organization is supported by the function of the I-organization. Depending on the task, an actor can take the shape of a B, I or D-actor. On the production side, D-applications are generic software such as text processors and database management systems. I-applications are enterprise(type)-specific software such as accounting systems and human resource applications. B-applications are decision support systems. On the coordination side, D-applications are typically networking and e-mail systems. I-applications are workflow systems. And B-applications are process-management systems.

2.2 ARIS Theory

ARIS stands for Architecture of Integrated Information Systems^x and is an architecture solution for the management of business processes. The framework was developed by Prof. Dr. August Wilhelm Scheer of the University of Saarland, Germany. In 1984 he founded the company IDS Scheer based on his Y-Model of business information systems integration. He did this as a way of both rigorously testing his technique, and proving its value to the business world. Then in 1992 he created the Event-Driven Process Chain (EPC) modeling method in a joint effort with SAP AG. Currently EPC is the key component of the modeling concepts in both ARIS and SAP ERP.

Principles

When the complexity of a business process is described in a single model and with only one modeling style, the model will often be large and confusing. To lessen the complexity of business process models two basic principles are relied upon in ARIS, namely 1) the principle of descriptive levels and 2) the principle of views.

Descriptive Levels

Descriptive levels are determined by their proximity to information technology. In modeling business processes, each phase of development increases the amount of detail in the direction of an actual information system. By modeling only the needs of the phase at hand, a sufficient level of detail is achieved and maintained with minimal modeling effort.

Views

Views reduce the complexity of a business process by sectioning it into user specific aspects, and enabling the description of each aspect with its own optimal modeling technique. This means that each view can be optimized for a specific audience by leaving out details that are relevant only to other audiences. Furthermore each view can be modeled in the style that is most suitable for it. The division by views is visualized in the ARIS house (see figure 3-5). The house is founded on the output view. The data view and the function view are its pillars. All views are housed under the organization view and connected to each other via the control view. And all views are created one phase or level at a time.

3 Model Types

This section introduces the reader to the types of models that must be coupled. On the one hand there are the four aspect models of DEMO along with the DVL model; the DVL model is a direct descendant of one of them. On the other there are the five ARIS views along with the UPP model which is a remixed descendent of the views. The DVL and UPP models mentioned in this section are the ones that the further research is built upon.

3.1 Dutch Terminology

Primaire Processen Beleid

Primary Processes of Management

Primaire Processen RWS

Primary Processes of RWS

Primaire Processen KNMI

Primary Processes of KNMI

Primaire Processen IVW

Primary Processes of IVW

Ondersteunende Processen

Supporting Processes

Verkeersmanagement Hoofdwegennet

Traffic Management of Main Roadways

Verkeersmanagement Hoofdvaarwegennet

Traffic Management of Main Waterways

Watermanagement

Water Management

Beheer, Onderhoud & Ontwikkeling

Management, Maintenance & Development

Verlenen Vergunning

Issuing Permits

Handhaven

Enforcing

Functionaristype

Department, Position, Person, or Role / functionary type

Invoer

Input

Referentie

Reference

Uitvoer
Output

Technologie
Technology

3.2 DEMO

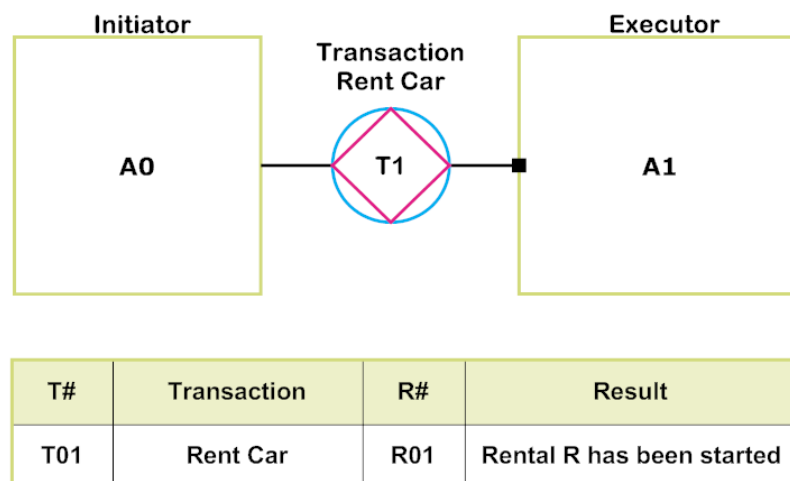
DEMO models the ontology of organizations and makes use of four aspect models namely the Construction Model, the Process Model, the Action Model and the State model.

The **Construction Model (CM)** specifies the construction of the organization. This relates to the operation axiom discussed in section 2.1. It is the highest level and most concise model of the DEMO methodology. The CM consists of two parts namely 1) the Interaction Model and the Interstriction Model. The **Interaction Model (IAM)** specifies the actor roles in the organization and the transactions that take place between them. For this it provides an Actor Transaction Diagram (ATD) and a Transaction Result Table (TRT). The **Interstriction Model (ISM)** specifies the relationship between the actor roles in the organization and the information banks used by them. The ISM provides an Actor Bank Diagram (ABD) and a Bank Contents Table (BCT). When they are merged into one, the ATD and ABD are called the Organization Construction Diagram (OCD).

16

The CM leaves out issues of process order and instead wraps the coordination and production acts and facts of a each transaction into one package. This capability makes it the most concise and easily understood of the DEMO models.

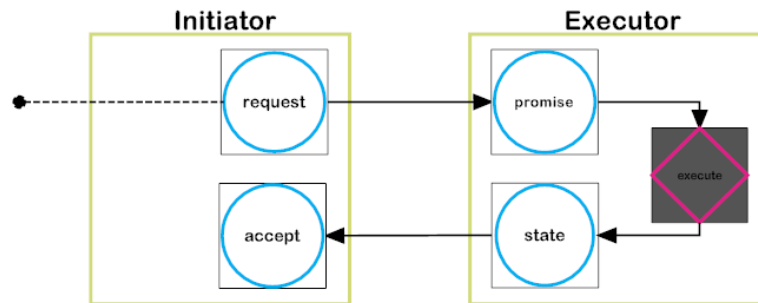
Part of a CM is shown in figure 3-1 where actor role A0 initiates a car rental transaction and actor role A1 executes it; execution is indicated by the square arrowhead. While the transaction contains coordination acts and facts and a production act and fact, these are not visible in the CM.



3-1 Sample of Actor Transaction Diagram and Transaction Result Table for Car Rentals.

The **Process Model (PM)** specifies the transaction patterns of each of the transactions in the CM. It also shows the causal and conditional relationships between transactions. As such it shows the state and transition space of the coordination world of the organization. The PM provides a Process Structure Diagram (PSD) and an Information

Use Table (IUT). The PM is thus a further detailing of the CM. Once one has achieved the comprehensive overview of the organization through the CM, the knowledge contained in the PM is overlaid on that knowledge and falls into place without confusing the analyst. Figure 3-2 shows the basic transaction pattern overlaid on the sample ATD in figure 3-1. For more information about the transaction pattern see section 4.1 and appendix 12.1.



3-2 Overlaying the Transaction Pattern

The **Action Model (AM)** is the specification of the action rules that actors follow in fulfillment of their actor roles. With the CM and the PM one attains knowledge of the actor roles and the transactions between them, and also of the causal and conditional relationships between those transactions. However, the actual guidelines that should be followed when transitioning from state to state are not yet explicit. The action rules serve as the guidelines that actors use but may also deviate from when necessary. The AM is specified in an Action Rule Specification (ARS), written in a pseudo-algorithmic language, which contains three clauses, namely 1) the *on clause*, 2) the *if clause* and 3) the *do clause*. The **on clause** is indicated by an on-no bracket pair, which encloses the action rule. The **if clause**, indicated by an if-fi bracket pair, specifies conditions for carrying out different actions. And the **do clause** specifies repeated actions by means of a do-od bracket pair. The AM is the most detailed of the four aspect models and contains sufficient detail to regenerate the CM and PM from it.

The **State Model (SM)** is the specification of the state space of the production world of an organization. It holds such things as object classes, fact types, result types, and existential laws. For visualizing these elements the SM provides an Object Fact Diagram (OFD), and for details which would make the OFD unnecessarily voluminous it provides an Object Property List (OPL).

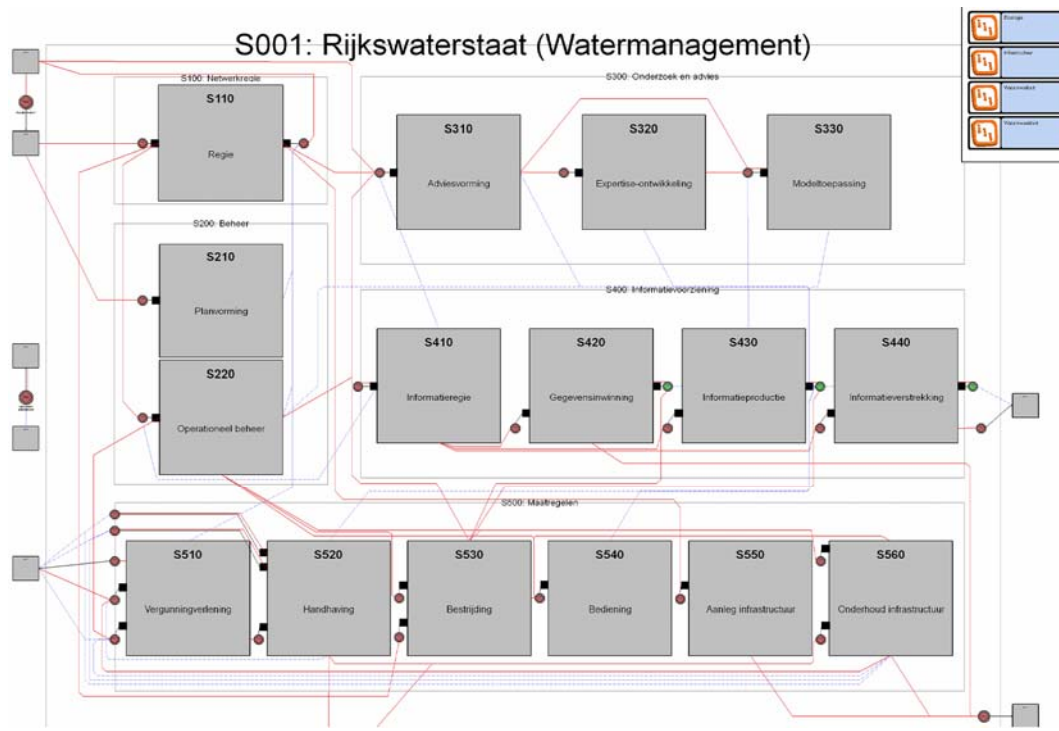


3-3 The Four DEMO Aspect Models

3.3 DVL Models

DVL models are based on the DEMO construction model. They are leveraged to show the various areas of the RWS organization, along with the services these provide to other areas. An example DVL model is shown in figure 3-4.

This diagram visualizes the structure of the watermanagement domain of RWS. Within the border of watermanagement there are 16 composite actor roles and outside of it there are seven. Each of these actor roles can be 'opened' to see the deeper structure. However, within the DVL document of watermanagement only the 16 internal actor roles have been expanded upon. To see the internal structure of the external actor roles one must consult the documentation of those actor roles. DVL documentation further contains textual descriptions and explanations of its diagrams. Typically the transactions, external actor roles, and internal actor roles are elicited.



3-4 DVL Model of RWS Water Management

3.4 ARIS

ARIS uses five views, namely the organization, function, data, output, and control views. These views depict different aspects of the same ARIS model.

The **Organization View** specifies who is responsible for a particular task. The responsible entities are called organizational units. An organizational unit can represent a department, a position within the company, or a subject. These relationships may be visualized in a hierarchical structure. For example, the highest level would indicate the organization as a whole, while the second level might show the various departments. The third level could then hold employee positions, while a fourth would present the actual subjects. The relationships may also be visualized in a context or interaction diagram. Such a diagram shows the organizational units with the output and communication relationships between them, but does not hold any sequencing information.

The **Function View** specifies the functions or activities that are performed as part of a task. This view may be visualized as a tree structure of functions. For example the highest level could indicate the business process as a whole, while the second level could show the various functional areas. The next levels could then be used to place tasks and subtasks. This view may also be visualized as a function flow diagram; a diagram of functions sequenced in the order of execution, along with the necessary logical operators. Logical operators are discussed below in the control view section.

19

The **Data View** is the specification of the data that is required to execute the various functions. There are different ways to construct the data view. If the Entity Relationship paradigm is used, then the view should contain the entities, their inter-relationships, and their attributes. If the Object Oriented paradigm is used, then the view should contain the classes, their attributes, their methods, and the generalization, aggregation and association relationships.

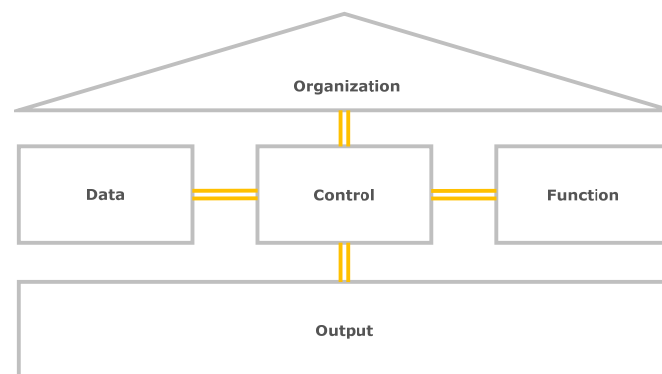
The **Output view** specifies the output that each business process produces and the equivalent value it receives. It also holds the intermediate outputs that follow from executing functions. Output can be material such as products, and immaterial when denoting services. The output view may be visualized as a hierarchical structure of outputs, or as an output flow diagram showing the outputs along with the functions that produce them.

The **Control view** which is the most complete of the five views, is built up by means of event-driven process chains (EPC). EPCs consist of, among others, alternating functions and events connected by control flow relationships. A function is an activity to be performed. It specifies the transformation from an initial status to a resultant status. An event is the status that results from executing a function, and triggers other functions. An EPC always starts with a triggering event and ends with a resultant event. Functions cannot trigger other functions. They can only result in events that in turn trigger new functions. For this reason functions and events will always be seen alternating. Besides events and functions, there is also the process interface. Process interfaces are navigational aids that show which processes the current process can lead to, and which processes the current process has followed from. Besides the control flow relationship, there are also the information flow and the organization unit assignment relationship. The information flow links data objects to functions while the assignment relationship

links organizational units to functions. Choices and synchronizations are indicated with AND, OR, and XOR (exclusive OR) logical operators.

- ✓ The **Opening AND**, or **Fork**, consists of a function or an event followed by an AND operator. There is one path into the AND, and multiple paths out of it. When the function is executed or when the event occurs, all outgoing paths are activated in parallel.
- ✓ The **Opening OR** consists of a function followed by an OR. It is not allowed to place an event before an OR, because decisions cannot be made during an event. There is one path into the OR and multiple paths out of it. When the function has been executed, one or more outgoing paths are activated depending on the decisions that were reached within the function.
- ✓ The **Opening XOR**, also called a **Branch**, is modeled as a function followed by an XOR with multiple outgoing paths. As with the opening OR, it is not allowed to place an event before the XOR because events cannot make decisions. Upon execution of the function exactly one of the possible paths is activated, based on the resultant state of the function.
- ✓ The **Closing AND**, also called a **Join**, consists of multiple events or functions followed by an AND with one outgoing path. There are multiple paths into the AND, and only one path out of it so that the AND synchronizes all incoming paths.
- ✓ The **Closing OR** consists of multiple events or functions followed by an OR with one outgoing path. If one or more of the incoming events occur, or if one or more of the incoming functions are executed, then the OR synchronizes the incoming paths.
- ✓ The **Closing XOR**, or **Merge**, is modeled as multiple events or functions followed by an XOR with one outgoing path. A merge is the opposite of a branch. It is about synchronizing one active path with several inactive paths.

In practice, there is a comprehensive set of representation techniques that can be applied in realizing the ARIS house, but not all were listed in the discussion above. For a more detailed listing see [\[KOZINA\]](#).



3-5 The Five ARIS Views

3.5 UPP Models

UPP models are based on ARIS and typically have six descriptive levels or layers. The first three layers are process layers showing a subjective distinction between process areas, main processes and actual processes. To better clarify the purpose of each layer, I have named each layer in a way that best demonstrates what one sees when one zooms in to a layer from an item at the previous higher layer. This is more intuitive for readers who will use the web-based UPP handbook to follow along. The handbook is discussed again in section 6.2.

Layer 1 : Process Areas

When viewing a UPP model one starts at the highest layer. Layer one is an overview of the organization in terms of process areas. The overview is presented in an added value diagram that clearly shows each process area of the organization. For V&W there are five process areas, namely: Primaire Processen Beleid, Primaire Processen RWS, Primaire Processen KNMI, Primaire Processen IVW, and Ondersteunende Processen. The first four regard the core business of V&W and its Directorates-General, while the fifth regards generic supporting processes such as finance, legislation, real estate, facility management, HRM and communication. The diagrams at this layer do not show which processes come before or after other processes.

Layer 2 : Main Processes

When viewing a process area one moves to layer two. Layer two is a further detailing of the organization in terms of the main processes that take place within the process areas. Layer two is also presented in added value diagrams. For Primaire Processen RWS, some examples of main processes are Verkeersmanagement hoofdwegennet, Verkeersmanagement hoofdvaarwegennet, Watermanagement (figure 3-4), and Beheer, onderhoud & ontwikkeling. The diagrams at this layer do not show which processes come before or after other processes.

Layer 3 : Processes

When viewing layer three one sees the next level of organization detail, namely the processes of the organization. The processes at layer three form the main processes at layer two. Layer three is presented in added value diagrams as well. Examples of processes taking place as part of Watermanagement are verlenen vergunning and handhaven. There is order information in the diagrams at this layer; one can see which processes come before or after other processes. The level of granularity used here is sufficient to be recognizable and executable by competent³ V&W employees.

Layer 4 : Sub-Processes

Layer four is the layer of sub-processes; the layer three processes are composed of such sub-processes. On this level a process is overviewed in a tree structure that contains nodes for each sub-process. Most processes of layer 3 are overviewed in this kind of tree structure but some are linked directly to their process logic, thus skipping layer 4.

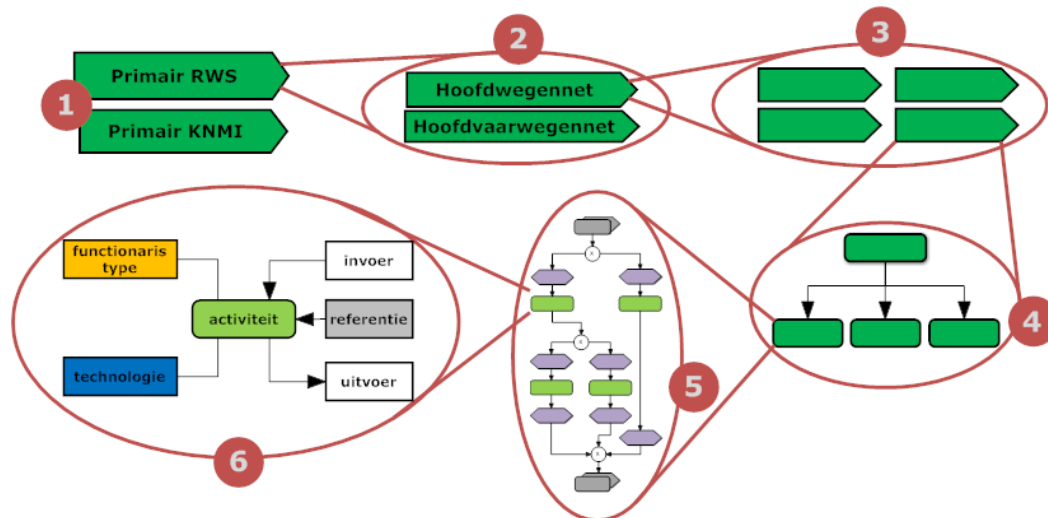
³Employees require a certain level of competence to comprehend and utilize the models.

Layer 5 : Process Logic

The fifth layer holds the process logic of the sub-processes at layer four in terms of the order in which activities and events can take place within them. Each sub-process is worked out into an event-process chain (EPC) showing the activities (ARIS functions) and events along with the choices and synchronizations on which they depend.

Layer 6 : Activity Realization

The last layer holds the documental details of the activities at layer 5. There are five categories of added detail, namely 1) functionaristype, invoer, referentie, uitvoer, and technologie. The first category, functionaristypen⁴, shows the parties involved with the activity in the upper left corner. Examples are departments, persons, or positions within the organization. The parties can be involved with the function in various capacities, for example they can be responsible for, contribute to, or execute the function. Other examples include giving advise, and needing information about a function. Next is invoer, the inputs for the activity, for example request forms. Underneath the inputs there is the referentie, information being referenced. An example of these are laws and regulations that must be complied with during execution of the function. Underneath the references is the uitvoer, the outputs. These can be documents such as permits. And finally in lower left corner there are the real world technology by which the abstract activity is realized, for example a system or an application. The figure below is a simplified depiction of the six layers of UPP.



3-6 The Six Layers of UPP Process Models

Each diagram is accompanied by textual descriptions and explanations. Next to the diagrams described above UPP also provides organization charts, system charts, document lists and a listing of definitions. These are not relevant for the further research. For a detailed walkthrough on modeling in UPP see [[CONARIS](#)].

⁴ At the GDSS session of 25 June 2008 it was determined that the Dutch term for the category of departments, company positions and persons would be functionaristype.

3.6 Conclusion

Based on the purpose of the models some conclusions can be made. The way in which DEMO and DVL models correlate is summed up in table 3-1.

DEMO	DVL	Reasoning
Construction Model	DVL	Of the four aspect models only the construction model is currently documented in DVL. Of the construction model, only the actor transaction diagram and the actor bank diagram are provided; the transaction result table and bank contents table are not provided.

3-1 DEMO-DVL Model Correlation

The way in which ARIS and UPP models correlate is summed up in table 3-2.

ARIS	UPP	Reasoning
Organization View	Organization Charts	Next to the layered process models UPP provides organization charts.
Function View	Layer Four	Layer four clearly shows a tree of activities for (almost) every process. Arguably layers 1 through 3 are the higher parts of these trees even though they are visualized through added value diagrams.
Data View	None	There is no separate data view in UPP.
Output View	Document lists	RWS outsources physical work as much as possible and so the immediate outputs are documents.
Control View	Layer Five & Six	Layer five holds an EPC with activities, events and logical operators. On layer six the connection is made with organizational units and data objects.
(Resource View)	System charts	[FERDIAN] mentions a resource view for systems.

3-2 ARIS-UPP Model Correlation

The way in which ARIS and DEMO models correlate is summed up in table 3-1

ARIS	DEMO	Reasoning
Organization View	Construction Model	The organization view, like the construction model, holds information about who is responsible in the organization. While the construction model shows roles and transactions, the organization view can show departments, positions, roles and persons.
Function View	Construction Model	The function view specifies the activities that are performed as part of a task. The construction model specifies the transactions that take place in the organization.
Data View	State Model	The data view and the state model provide the same kind of information such as object classes and their interrelationships.
Output View	None	The products and services visible in the output view are also defined in the state model but the state model does not show how one input follows from another or what the equivalent value is.

Control View	Process & Action Model	<p>The control view shows a process as a string of alternating activities and events connected by control flow relationships. This is similar to the transaction pattern of the process model. Both alternate between action and result and both show the conditional and causal relationships between activities. The difference is that the transaction pattern has a specific set of activities and events and the control view allows arbitrary activities and events. The control view also contains the guidelines that should be followed when transitioning from state to state. In DEMO these are specified in the action model.</p>
--------------	------------------------	---

3-3 ARIS-DEMO Model Correlation

4 Meta Object Types

This section introduces the reader to the meta object types that are present in the models that were introduced in section 3. This analysis is carried out to get an even deeper understanding of how the models are related to each other. This analysis strongly influences the direction that is chosen in phase two of the research.

4.1 DEMO Meta Object Types

In this section the meta object types of DEMO are analyzed so that they can be compared with ARIS object types later on. Note that in this analysis relationship types, such as information links, are not the primary concern. Rather the actor role object types, between which relationships exist, are the main focus. Types that are mere visual elements of DEMO diagrams are not taken into account. So the “boundary” is not analyzed in this section because it only exists as a visual aid within the diagrams of the CM and PM.

In DEMO an **actor role** is the unit in which the authority, responsibility, and competence for being the executor of a transaction is expressed. To fulfill an actor role, is to perform acts. An **act** is defined here as an atomic unit of activity. A **factum** is defined as a fact that is the result of an act. Hereby a **fact** is an elementary state of affairs. The factum “membership M has been started” only comes into existence after the transaction for starting new club memberships has been successfully completed. Related to the factum is the **statum**. A **statum** is defined as a fact that is not the result of an act. A statum can be a fact regarding what an object is. It can also regard the relationships between objects. For example the statum “Dutch is a Language” exists without first executing a transaction; assuming of course that defining new languages is not one’s core business. Every fact is related to one or more object classes. An **object class** is the domain of a variable in a fact. For factum “membership M has been started”, membership is the object class that serves as the domain of variable M.

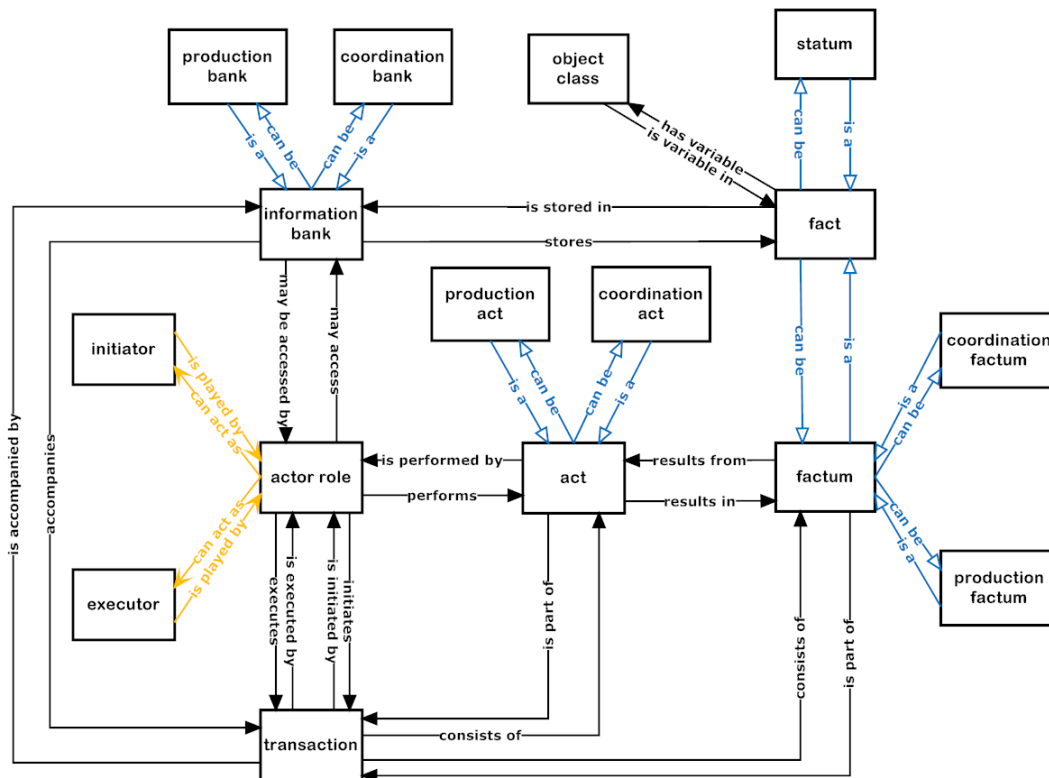
There are two kinds of acts, namely 1) production acts and 2) coordination acts. **Coordination acts** are communicative commitments related to a production act. **Production acts** are acts through which subjects contribute to bringing about goods and/or services that are delivered to the environment of the enterprise. Similarly, there are coordination facta and production facta. The result of a successful coordination act is a **coordination factum**, and the result of a successful production act is a **production factum**.

Production and coordination facta are stored in **information banks** to which only certain authorized actor roles have access. A **production bank** is a conceptual store of production facta. A **coordination bank** is a conceptual store of coordination facta. Each type of transaction has a coordination and production bank associated with it.

Thus all the transactions of type ‘deliver restoration materials’ have a shared coordination and a shared production bank associated with them. Each transaction for delivery of restoration materials adds stata and facta to these two banks. Similarly the transactions of type ‘perform restoration’ have a different pair of production and coordination banks, in which their corresponding stata and facta are stored.

When actor roles perform coordination acts and production acts, these acts are sequenced according to universal patterns called **transactions**. A transaction takes place

between two actor roles; the **initiator** is the actor role that starts the transaction and the **executor** is the actor role that performs the production act that the transaction is about. Thus the initiator is authorized to initiate a transaction, and is responsible for performing the corresponding coordination acts, according to the transaction pattern. The executor is authorized to execute a transaction, and is responsible for performing the production act and the corresponding coordination acts, according to the transaction pattern.



4-1 DEMO Meta-Model

Figure 4-1 visualizes how the DEMO meta object types are related to each other. Follow the direction of the arrows to interpret the relationships e.g. an actor role performs acts and an act is performed by an actor role. The relationships indicated by blue arrows with open arrowheads emphasize that the information bank, the act, the fact and the factum object types cannot be directly instantiated. This means that you can make an instance of a production bank, or you can make an instance of a coordination bank. But you cannot make an instance of an information bank, because this is a abstract object type. The same holds for the act, the fact and the factum. You cannot make an instance of a fact. You must choose to either make a statum instance, or to make a coordination factum instance, or a production factum instance. The relationships indicated by orange arrows with winged arrow heads emphasize that the actor role can be instantiated and that the initiator and the executor cannot be instantiated. So you can make an instance of actor role, but you cannot make an instance of a initiator.

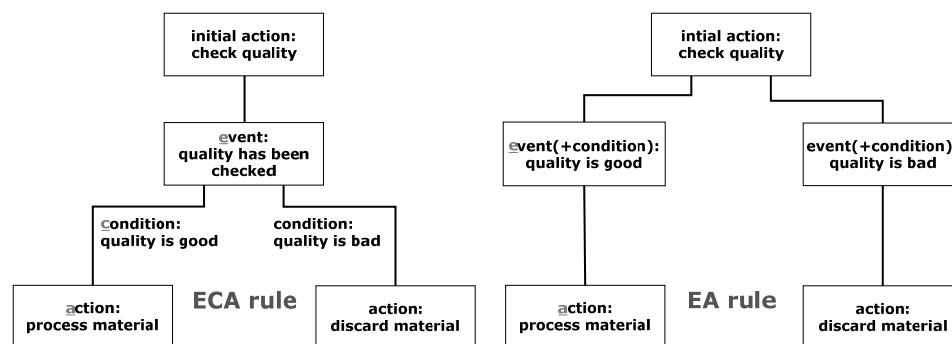
4.2 ARIS Meta Object Types

In this section the meta object types of ARIS are analyzed to determine how they are related to those of DEMO.

The **organizational unit** is the unit of responsibility within ARIS. The organizational unit shows who is responsible for a specific function. Examples are the sales department and the sales manager.

The **function** is the unit of activity within ARIS. Processes can be broken down into the individual functions of which they are composed. The **event** denotes a state change. A state change can trigger the execution of a function, and can itself result from the execution of a function.

Note: In ARIS an event is actually a combination of the resultant state of a function and the condition under which that state is relevant for taking further action. In computer science a state or an event is said to lead to an action on satisfaction of a condition. This is commonly referred to as the event-condition-action (ECA) rule. In ARIS the ECA rule is reduced into the EA rule by combining the event with the condition. Rather than having an event that leads to one action under one condition and another action under a different condition, an event+condition leads to one action and a different event+condition leads to another action. This concept is visualized in figure 4-2.



4-2 ECA vs. EA rule

According to the ECA rule, the initial action 'check quality' results in the event 'quality has been checked'. There are two conditions under which this new state is relevant for further action; either the quality is good, or the quality is bad. If the quality satisfies the first condition, then further action is to process the materials. If the second condition is satisfied, further action is to discard the materials. The point is that there is a clear separation between 1) the event, 2) the possible actions, and 3) the conditions that lead to them. According to the EA rule however, the initial action 'check quality' leads to two events which include conditional information. The first event is that the quality check determined good quality, and the second event is that the quality check determined bad quality. The first event leads to materials being processed and the second event leads to materials being discarded. There is no separate place for events and conditions; these are blended together.

Besides the functions, events and organizational units, there is in essence only one more meta object type in ARIS, namely the deliverable. The deliverable meta object type applies to anything that is delivered upon successful execution of a process. It can be a physical product; for example salt is the deliverable of a salt mine, food is the deliverable of a restaurant. Information can also be a deliverable; for example a weather service returns data values for wind speed and temperature. A deliverable can also be a service that is performed; for example a lawyer performs a legal service. This is neither a product nor mere information.

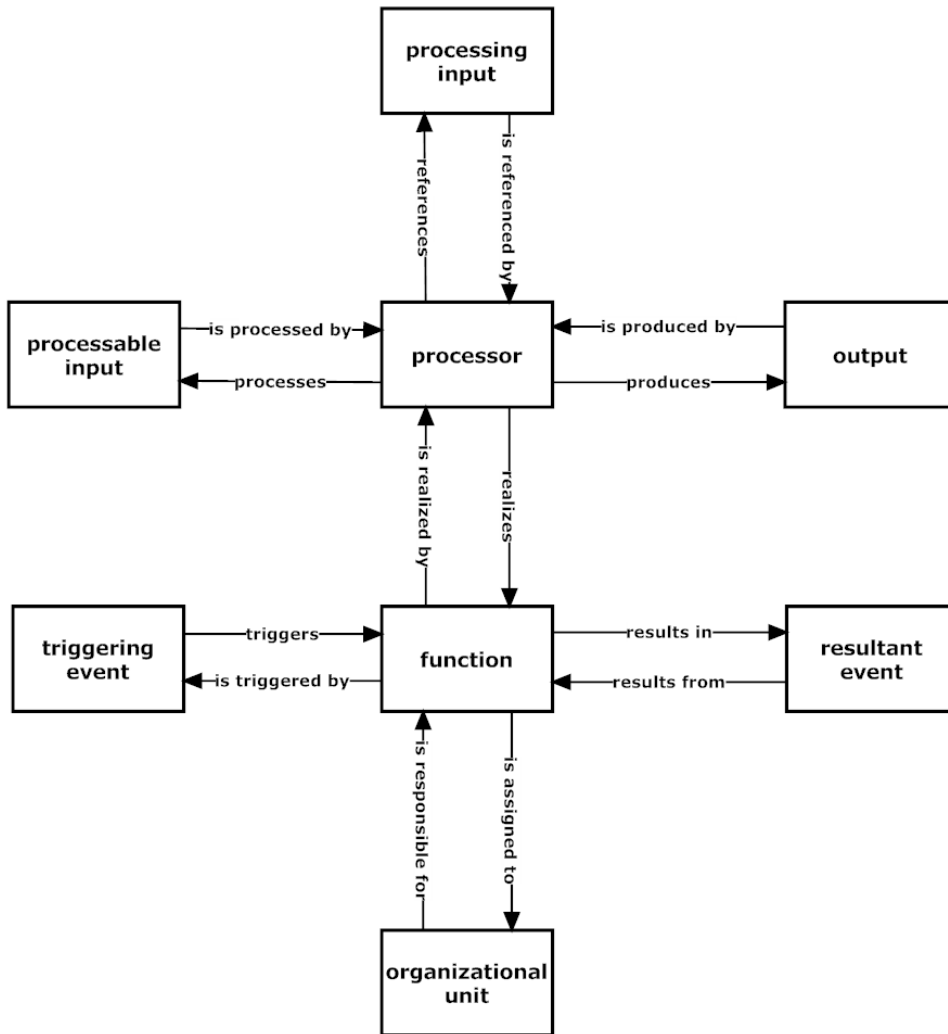
However, ARIS determines its meta object types based on the way they are used within the process under consideration, rather than on what they *are* in general. This statement is clarified with the following example. A brick is the deliverable of a baking process in a brick factory. No matter where in the world a brick is, in a truck on its way to a customer or on a build site on its way to becoming part of wall, it was at one point delivered after a baking process. However while it is in the truck it serves as a product being transferred to a customer to complete a sales process. And while it is on the build site it is regarded as material in a construction process. Thus the process under consideration determines whether a brick is a product or a material⁵.

Using this concept of focusing on the process under consideration and the purpose of the objects used within it, there are eight identifiable meta object types in ARIS. First there are the functions and organizational units as before, but the event can be split into the triggering and the resultant event. Further there is no universal deliverable. Instead there are the processor, the processable input, the processing input and the output. I have devised this naming scheme because the meaning of the original ARIS terms, such as resource, data, material etc, has become blurred and ambiguous over time.

A **processor** relates to the Dutch notion of 'materieel'. It regards a limited amount of machines and programs (the use of programs can be limited by the amount of available licenses) by which functions are realized. **Processable input** ('materiaal') is thus the raw material that enters a processor and is transformed into a product. It can also be data that is transformed into meaningful information, or that is used to perform a service. Hereby the product, the information and the service are all examples of the **output** object type. **Processing input** is the material or information that is used to operate the processor itself. For example a machine may require grease to keep it running smoothly but this is not an input in the same sense that a raw material is an input. The machine that requires grease can be replaced by a more modern machine with no moving parts and yet deliver the same product. The grease was not transformed into a part of the end product; it was merely required by the specific processor used at the time. In the example addition $2+5=7$, the numerals 2 and 5 are the processable inputs because they are transformed into the output 7. But the rules of addition are the processing inputs, because they are not transformed into a part of the output.

These meta object types are the essential ones in ARIS. Additional meta object types taken from UPP include but are not limited to the **role** and the **position**. These are both related to organizational units. Further the **system** is a kind of processor, because it indicates the computer or manufacturing system that realizes a function. **Documents, document lists** and **laws & regulations** can be inputs and/or outputs depending on the process under consideration. When they are produced as part of executing a function they are said to be outputs. When they are referenced, or transformed during execution of functions they are considered to be inputs.

⁵ Contrast this with the object types of DEMO such as the object class, the fact, and the act; there is no product object type or material object type in the modeling language of DEMO.



4-3 ARIS meta-model

Figure 4-3 visualizes how the ARIS object types are related to each other. Follow the direction of the arrows to interpret the relationships: a function is realized by a processor and a processor realizes a function. The bottom half of the diagram shows *the abstract* and the top half shows *the realization* of the abstract.

4.3 Conclusion

The way the DEMO and ARIS object types correlate has been summed up in table 4-1.

DEMO	ARIS	Reasoning
act	function	Since the act is the unit of activity in DEMO and the function is the unit of activity in ARIS, DEMO acts are comparable to ARIS functions.
factum	event	Since a factum is a fact that results from an act, and an event is the result of a function, the DEMO factum is comparable to the ARIS event. However facta and events are labeled differently from each other. A factum is labeled as something that happened at a certain point in time. The ARIS event is labeled as something that happened at a certain point in time combined with the condition under which it is relevant for further action (see ECA vs. EA rule).
statum	none	There is no ARIS equivalent of the statum object type.
transaction	functions and events	A transaction in DEMO is a sequence of acts and facta arranged according to the transaction pattern. Therefore, if directly converted, a transaction would map to a sequence of functions and events in ARIS.
actor role	organizational unit	The DEMO actor role can be substituted by the ARIS organizational unit but the organizational unit cannot always be substituted by an actor role. This is because the actor role is not directly equivalent to entities like departments and managers.
(contents of) information bank	processing input	The information bank is a container of facta and stata that may only be accessed by actor roles with the proper authorization. The equivalent of those facta and stata in ARIS is the processing input since it contains information necessary to execute a function.
none	processor	Implementation issues such as systems and applications are not provided in DEMO.
object class	processable input	Whether something is being used as input or output is not the focus of DEMO. Instead there are object classes and they are related to each other in various ways.
object class	output	Whether something is being used as input or output is not the focus of DEMO. Instead there are object classes and they are related to each other in various ways.

4-1 DEMO-ARIS Correlation

5 The Substitution Test

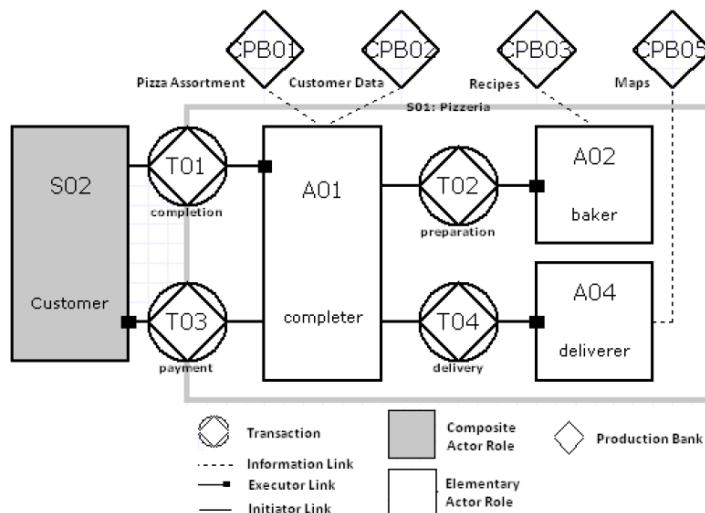
In this section an attempt is made to couple the models of DVL to the related models of UPP by using the knowledge gathered in sections 2, 3, and 4. Knowing that DVL and UPP are about the same topics there is an expectation that, to an extent, information modeled in DVL has also been stored in models of UPP and vice versa. This expectation is tested with small exercises in object type substitution.

Basically if there is some degree of coupling between DVL and UPP then substituting ARIS object types for the DEMO object types in DVL should generate a matching UPP model of a certain quality. In addition to this, substituting DEMO object types for the ARIS object types in UPP, should generate a matching DVL model of a certain quality. This kind of “cross-generation” should then show where UPP and DVL converge and where they diverge, thus exposing the coupling.

5.1 Generating UPP Layer Four

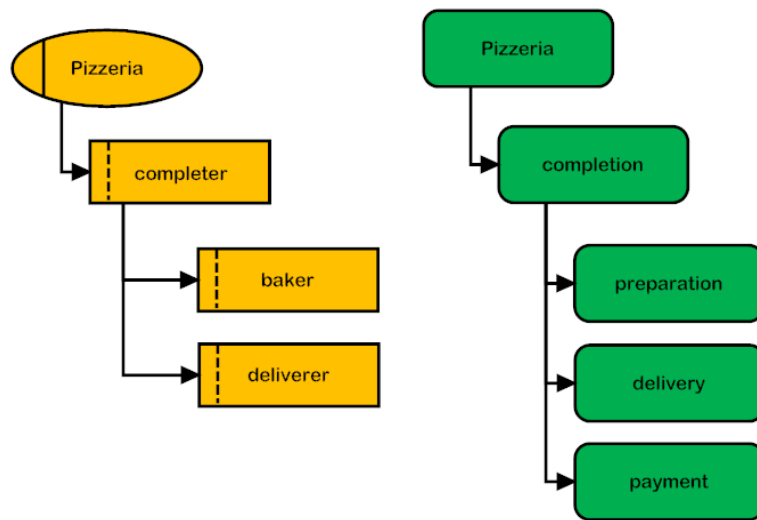
We know from section 3.6 that DVL matches the construction model of DEMO and that the construction model matches the organization and function views of ARIS. These in turn match the organization charts and layer four of UPP. To find the coupling that exists between the models of DVL and UPP one could generate UPP sub-processes from a DVL model, and the reverse, to analyze the results.

31 At this stage in the research no DVL and UPP models had been sufficiently studied to meaningfully interpret the results of the tests. Therefore the well-known Pizzeria example was used instead. While many small exercises were performed during this phase of research, this particular example was taken from [ALBANI ET AL.]. The given construction model shows that a Customer can initiate T01 Completion after which T02 Preparation, T03 Payment and T04 Deliver can be initiated. The production banks associated with the pizzeria are CPB01 Pizza Assortment, CPB02 Customer Data, CPB03 Recipes, CPB05 Maps. For the complete Case of the Pizzeria see appendix 12.2.



5-1 Construction Model of the Pizzeria

When substituting organizational units for actor roles as discussed in section 4.3 the organization chart of figure 5-2 is achieved. When relating the function view to the transactions, as stated in section 3.6, then the function view in figure 5-2 is achieved.



5-2 Organizational Chart and Function View Generated From the Pizzeria CM

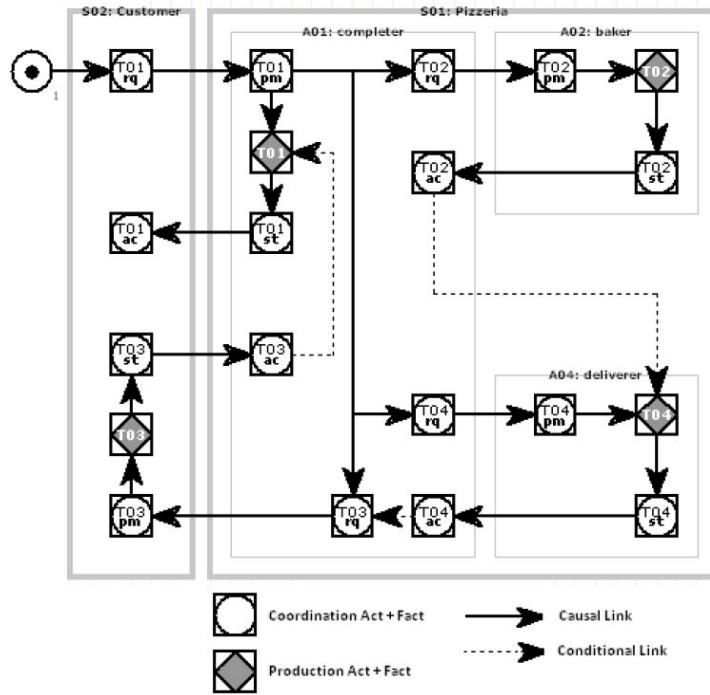
Using the same substitution rules as before the DEMO CM can be regenerated from the organization and function views that are present here. Root nodes map to the border of the organization, the composition of transactions is recreated, and the delegation structure of the internal actor roles is also recreated. However the knowledge of the external actor role is lost.

32

Note: [\[ALBANI ET AL.\]](#) did not provide an authentic pair of organization and function views to which the generated views could be compared; in this example there is no way to judge how closely the generated views match the authentic views.

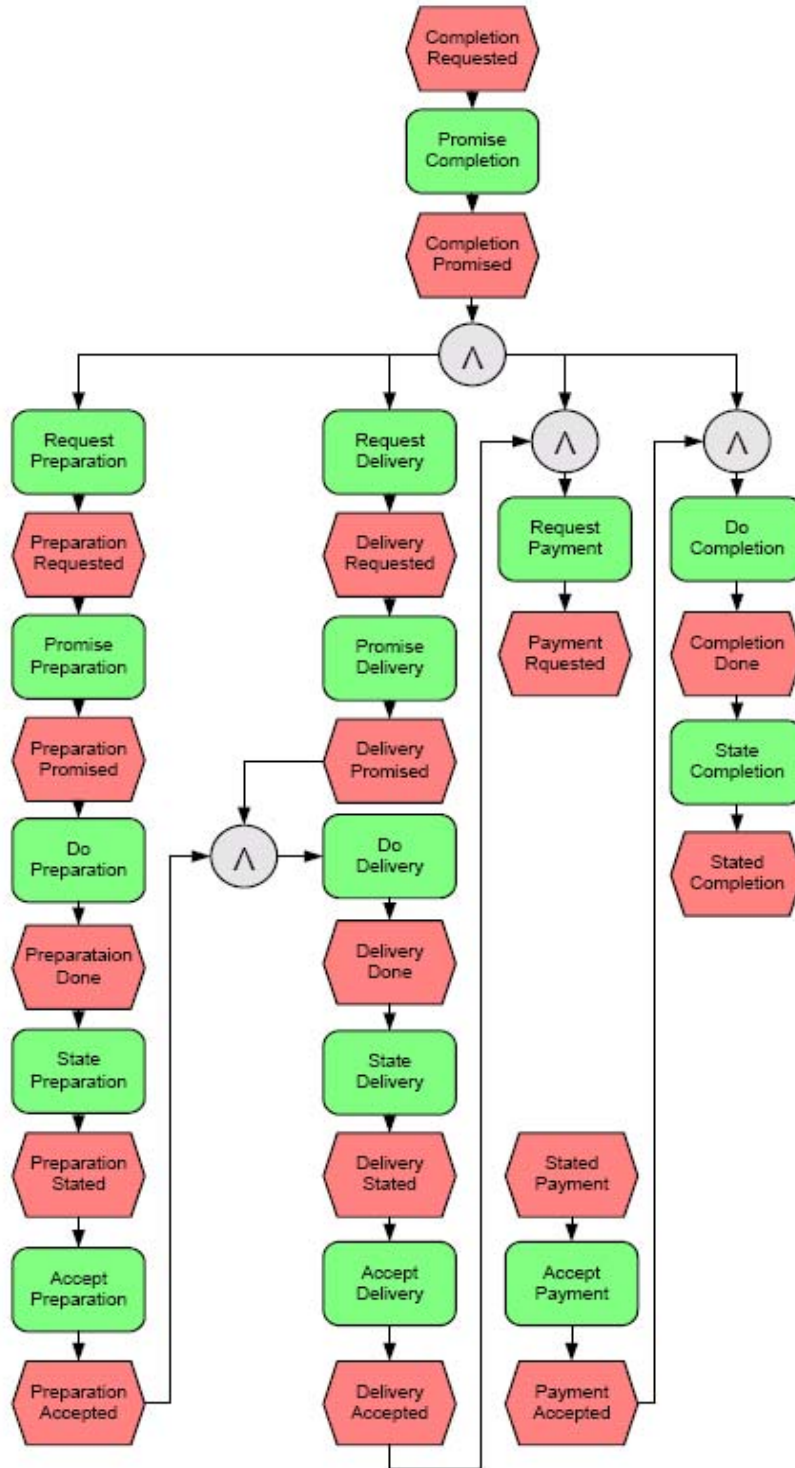
5.2 Generating UPP Layer Five

Mapping high level functions to transactions is fairly straightforward, but substituting activities and events for the transactions to built up the process logic is somewhat more complex; each transaction must be substituted according to the transaction pattern. This requires the DEMO process model since that shows the order in which the acts and facta of the transactions in the construction model follow each other. The PM is provided in figure 5-3.



5-3 Process Model of the Pizzeria

Now that the sequencing information and causal relationships of the Pizzeria's process steps are known, the UPP process logic can be generated. The following example shows the results of generating UPP process logic from a construction model by replacing the transactions with a series of activities and events as described in section 4.3. Using the process model as a guide, every act is replaced with a function and every factum is replaced with an event while logical operators are positioned as needed.

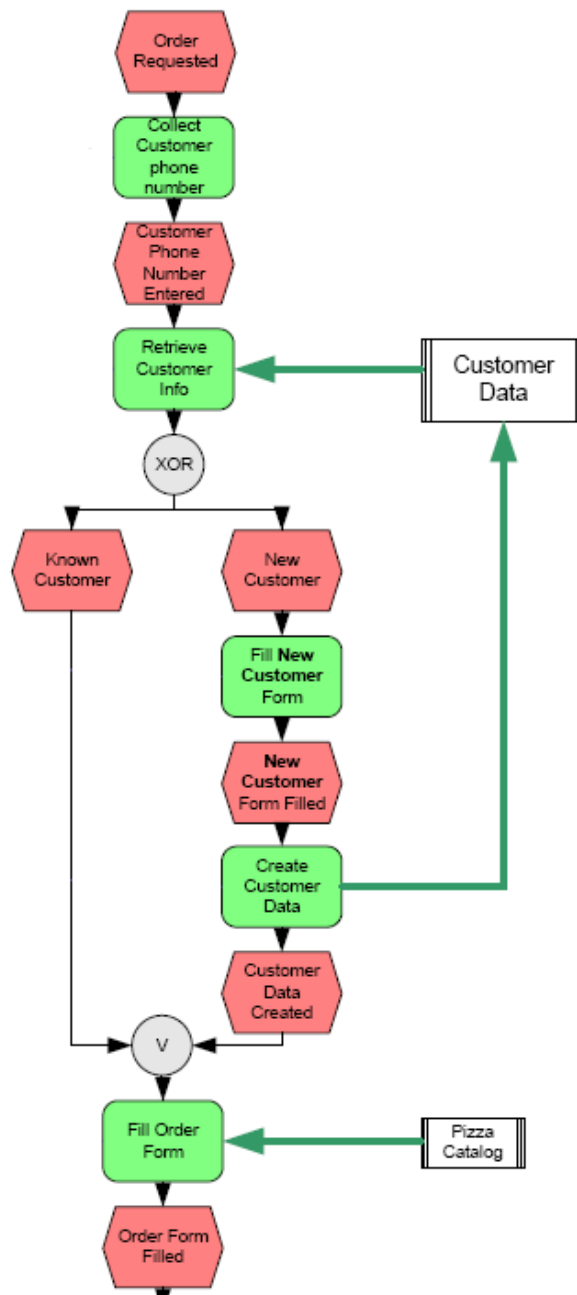


5-4 Process Logic Generated from the CM and PM

This resultant diagram is a replica of the process structure diagram in figure 5-3, but in terms of ARIS object types. One can see the request, promise, execute (do), state and accept steps following the same pattern as before. Because of this symmetry, recreating the process structure diagram from this diagram would be straightforward.

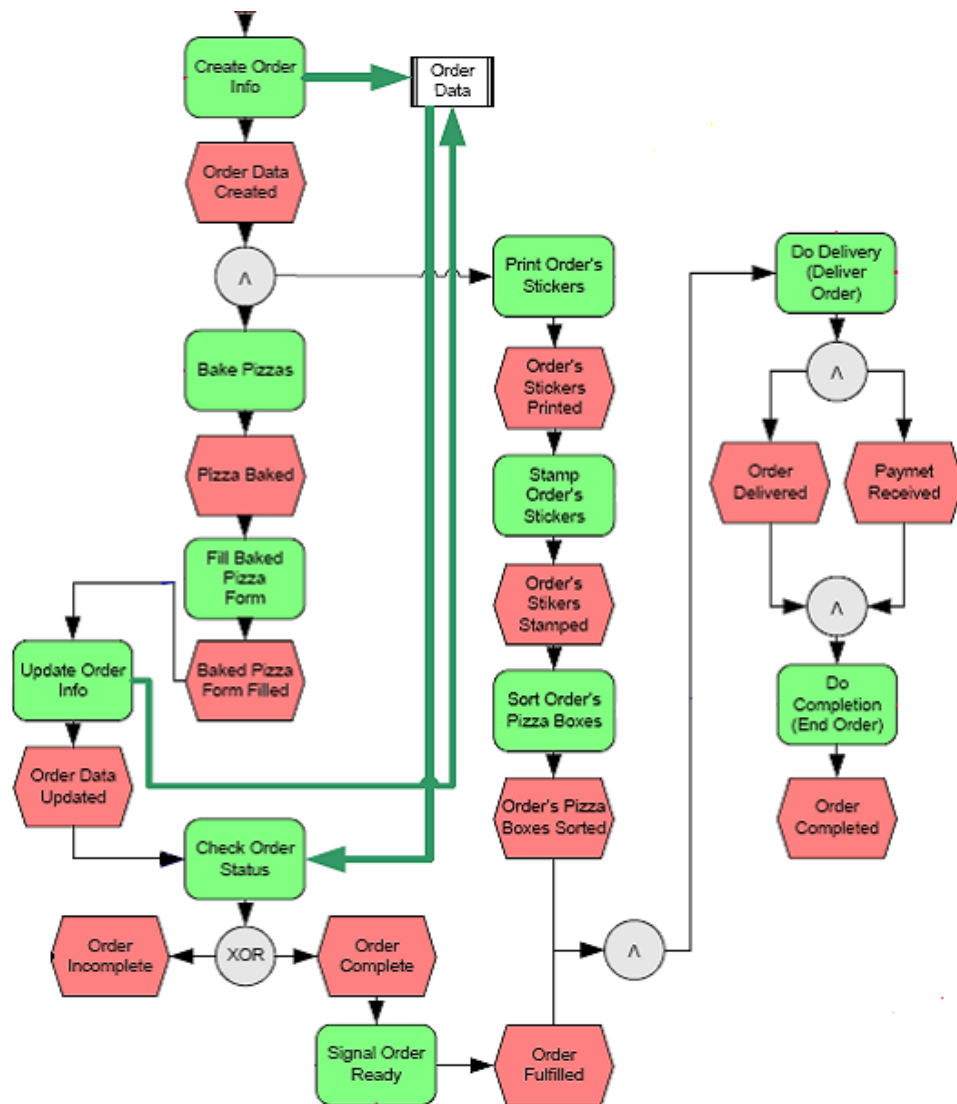
5.3 Authentic UPP Process Logic

The diagram produced from following the rules of EPC is shown in figure 5-5. It holds functions such as *collect customer phone number*, *retrieve customer info*, *fill new customer form*, and *create customer data*. The first request leads to getting the customer's phone number. This leads to retrieving information about the customer. If the customer is known, function *fill order form* is executed. Otherwise *fill new customer form* and *create customer data* are executed first. For the case description of the Pizzeria see appendix 12.2.



5-5 Authentic Model of Pizzeria - I

The second part of the process shows the creation of order data, updating order information after the pizza has been baked, printing order stickers and so on up to the point of order completion.



5-6 Authentic Model of Pizzeria – II

Generation the process structure diagram or even the construction model from this diagram will be more of a challenge because one must first extract the ontological meaning from it.

5.4 Conclusion

The kinds of concepts present in the authentic diagram and those present in the generated one differ substantially. The generated model is more abstract and coarse in comparison to the authentic model.

In the DEMO-generated model visible in figure 5-4, the request from the customer leads to a function *promise completion*. The resulting event triggers the function *request preparation*.

In contrast, the authentic ARIS model does not follow the transaction pattern. Instead it holds information that is immediately actionable such as *fill new customer form*.

While this cannot be said with certainty about layer four, the overall conclusion is that the coupling, that exists between DEMO and ARIS, is more complex than an object type mapping. It is possible that the CM and the function and organization views map to each other fairly well, but this is definitely not the case for the DEMO PM and UPP process logic.

6 Analysis of Issuing Permits

In the previous section a basic DEMO CM and a matching ARIS EPC were used to serve as DVL and UPP models. In this section actual DVL and UPP models are analyzed and compared. Given the insights from section 5.4 the focus now is to identify the content related correlation between DVL and UPP.

6.1 Dutch Terminology

Beheer en Onderhoud (B&O)

Management and Maintenance

V&W handboek Administratieve Organisatie (handboek AO)

V&W handbook of the Administrative Organization (handbook AO)

Beheer Infrastructuur

Management of Infrastructure

Onderhoud Infrastructuur

Maintenance of Infrastructure

Dienstverleningen

Service Provisions

38

Vergunningverlening en handhaving

Issuing and Enforcing Permits (as one)

Verlenen vergunning and Handhaving

Issuing Permits and Enforcing Permits (separately)

Watermanagement en Beheer & Onderhoud (WM B&O)

Management and Maintenance (within the area of) Water Management

Vorbereiden Afhandeling Vergunning

Preparation for issuing permits

Afhandeling Vergunningaanvraag 3.4 Procedure

Process Permit Request (in accordance with) Procedure 3.4

Afhandeling Vergunningaanvraag 4.1 Procedure

Process Permit Request (in accordance with) Procedure 4.1

Bezwaar en Beroep

Objection and Appeal

6.2 Delineation of Analysis Area

In order to develop a coupling between DVL and UPP model types, the similarities and differences between the two model types had to be assessed in greater detail. In order to do that, an area needed to be found that had been modeled in both DVL and UPP.

This section explains the areas that were chosen as the basis for developing a beneficial coupling between DVL and UPP and the steps that led to their selection.

Talks with members of EAC and BPS revealed that the area Beheer en Onderhoud (B&O) in DVL would likely have a matching process area within UPP. The reason they gave for this was that the DVL model of B&O, unlike the previous DVL models, was developed from the UPP models present in the V&W handboek Administratieve Organisatie (handboek AO). The handboek AO is a web-based handbook that contains and interconnects all of the UPP models, and is updated on a monthly basis. The handbook is more than the sum of its parts; it is the heart of the organization, in the sense that it is the definitive word on how the members of the organization perform their work. A second reason they gave was that during the making of the B&O DVL model, the teams EAC en BPS worked together to determine potential transactions (or dienstverleningen as they are called by EAC). After taking these key issues into account, I chose B&O as the starting point of my search for matchable areas.

The document [APPConnAT/B&O] revealed that B&O regards maintenance of infrastructure. The task of the subdivision Beheer Infrastructuur is to keep the coherence between the varying management and maintenance activities. The task of Onderhoud Infrastructuur is to coordinate and execute individual maintenance tasks. There is also the subdivision Vergunningverlening en handhaving which regulates the use of assets by issuing and enforcing permits. Next to these three, there are five more subdivisions in B&O. Considering that each of the subdivisions has its own DVL model, I realized that B&O as a whole is too large to analyze in a meaningful way. For this reason I decided to select and use a single subdivision of B&O.

The handboek AO was instrumental in determining which subdivision to use. In the handbook a main process named Beheer, Onderhoud & ontwikkeling (BOO) was quickly found. Within this main process, reference is made to two process clusters namely Verlenen vergunning and Handhaving. It was very convenient to have Vergunningverlening en handhaving in DVL and then to find Verlenen vergunning and Handhaving in UPP. Due to this convenience, the choice was made to match the two process clusters Verlenen vergunning and Handhaving with the work field Vergunningverlening en handhaving of DVL.

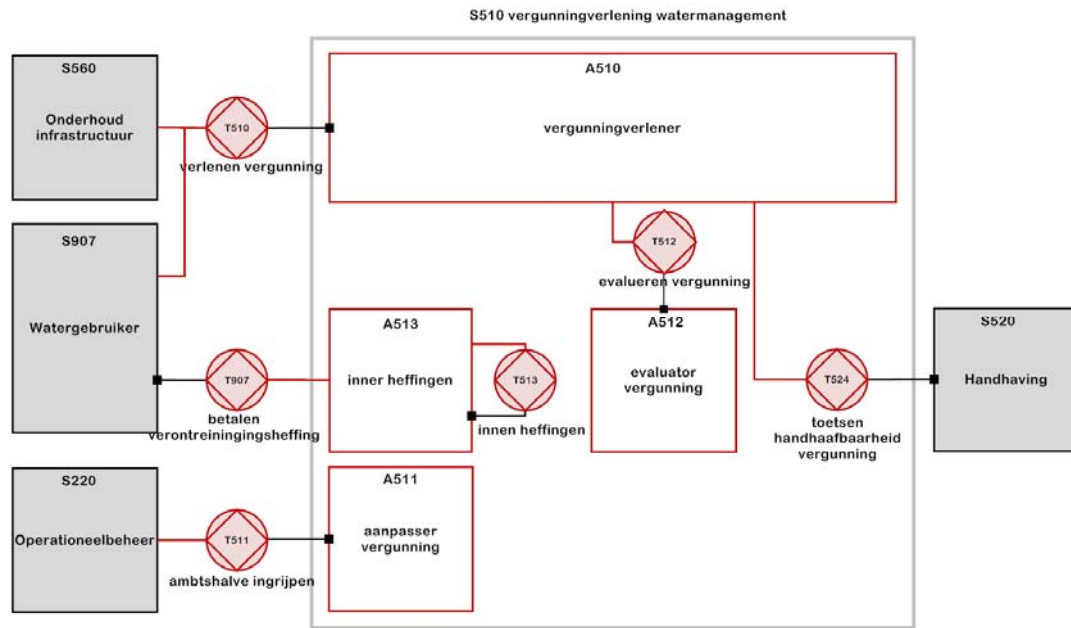
Upon further examination it turned out that the DVL model Watermanagement en Beheer & Onderhoud (WM B&O) was more recent than the B&O model. What was convenient about the newer model was that it contained the area Vergunningverlening en handhaving as two separate areas, namely Vergunningverlening and Handhaving. Having the separation of Vergunningverlening and Handhaving in both DVL and UPP made it possible to analyze an even smaller delineated area, by selecting only one of the two.

For this reason my final choice was to match 'DVL > WM B&O > Vergunningverlening' with 'UPP > WM(BOO) > Verlenen vergunning' in order to develop the linkage.

Note: in the handboek AO, the path UPP > BOO > Verlenen vergunning leads to exactly the same model as UPP > WM > Verlenen vergunning. There is a third path that leads to the model Verlenen vergunning, namely, UPP > Verkeersmanagement hoofdvaarwegennet > Verlenen vergunning.

6.3 The DVL Model for Issuing Permits

The DVL model for issuing permits, Vergunningverlening, consists of a DEMO actor transaction diagram accompanied by textual descriptions of the transactions, external actors and internal actors. It originates from the document Dienstverleningsmodel Watermanagement en Beheer & Onderhoud Nat dd: 20-12-2007 15:52 . It is specific to the water management area of V&W. Note that in the diagram below information links have been omitted.



6-1 DVL model – Actor Transaction Diagram

Transaction #	Transaction Type	Result #	Result Type
T510	verlenen vergunning	R510	vergunning V is verleend
T511	ambtshalve ingrijpen	R511	vergunningtype VT is ambtshalve gewijzigd.
T512	evalueren vergunning	R512	evaluatie van vergunning V is voltooid
T513	innen heffingen	R513	heffingen voor vergunning V over periode P zijn geïnd.
T524	toetsen handhaafbaarheid vergunning	R524	handhaafbaarheid van vergunning V is getoetst
T907	betalen verontreinigingsheffing	R907	verontreinigingsheffing voor vergunning V over periode P is betaald

6-1 Transaction Result Table

External Actors

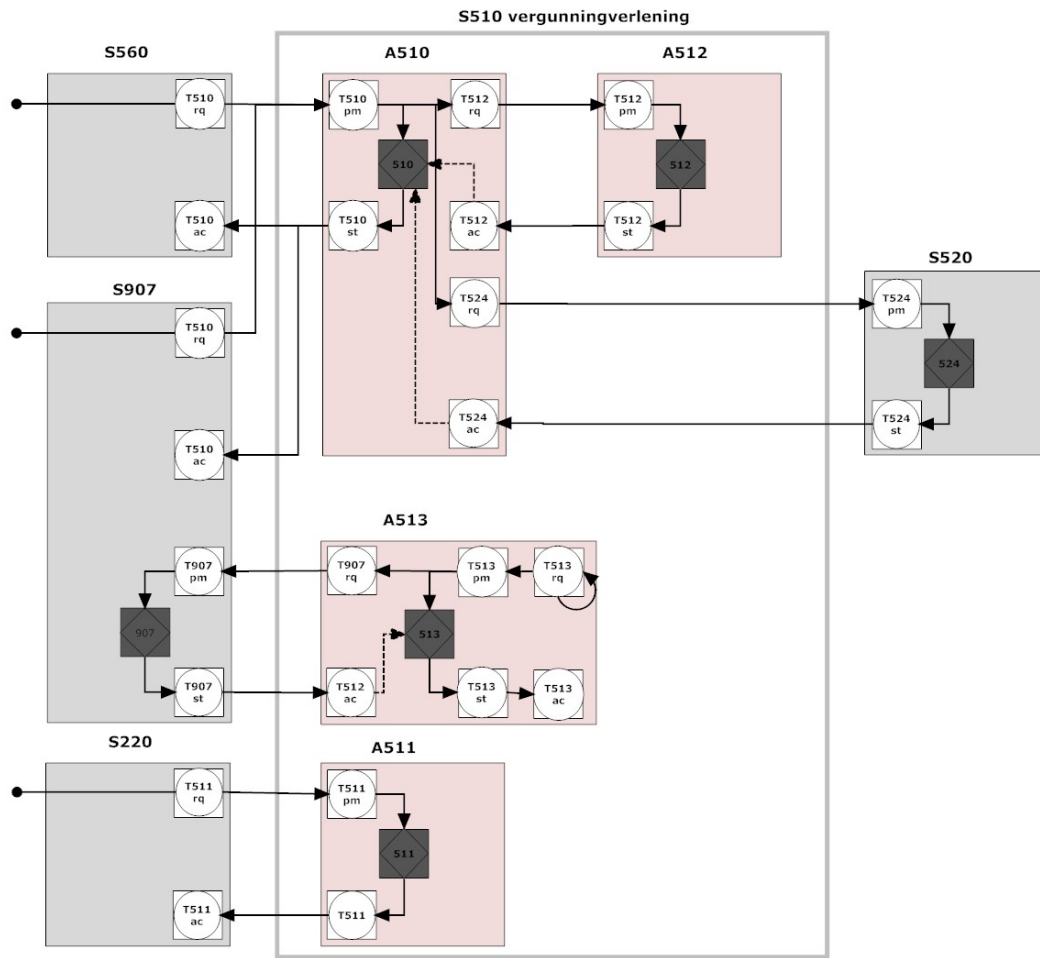
As part of coordinating and executing individual maintenance tasks on infrastructure elements, the Onderhoud Infrastructuur must sometimes request permits for use of the waterways. The Watergebruiker represents those persons or organizations that use water for drinking, recreation and commercial shipping and request permits for those activities. The Operationeelbeheer is responsible for monitoring and managing the chemical, physical, biological, and morphological state of the Dutch water system and

related “wet” infrastructure. Their insights can lead to the revision of the content of permit types. For example if they find that water quality has dropped, currently used permit types become obsolete and new permits are issued with lower tolerances for drainage pollution. On the right hand side Handhaving is responsible for enforcing permits. They perform such tasks as educating the public on relevant laws and permits, patrolling in order to deter would-be offenders, checking that laws and permits are being obeyed, and imposing sanctions on offenders. Since it is Handhaving that must eventually enforce the permit, the permit issuer delegates the checking of enforceability to them. The idea is that permit requestors provide certain proofs that they have done everything necessary to qualify for the permit which means that the permit can be properly enforced on them.

Business Processes and Transactions

Note that in figure 6-1 initiator links are colored red while executor links are colored black. The diagram depicts two transactions initiated from outside its border, namely T510 verlenen vergunning and T511 ambtshalve ingrijpen. It shows one self-initiating transaction, T513 innen heffingen. Therefore there are three business processes in the given CM.

The first business process is the process verlenen vergunning. This business process is about issuing permits for use of the waterways. In this process, T510 initiates the sub-transactions T512 evalueren vergunning and T524 toetsen handhaafbaarheid vergunning. The second business process is innen heffingen where T513 initiates sub-transaction T907 betalen verontreinigingsheffing. This process is about taking periodic payment related to issued permits. The third business process, ambtshalve ingrijpen, is about updating permit types when they become outdated.



6-2 Process Structure Diagram

6.4 The UPP Model for Issuing Permits

The UPP model for the issuing permits, *verlenen vergunning*, has been taken from the handboek AO of November 2007. Unlike the DVL model in section 6.3, it is a general model of issuing permits and is not specific to the area of water management.

The model consists of three layers and each layer has its own diagramming technique and its own set of textual explanations. The diagram in figure 6-3 shows the contents of *verlenen vergunning*.

The sub-processes that occur as part of *verlenen vergunning*, are structured into a three. In the three, the root node *verlenen vergunning* branches off into three sub-nodes. The first sub-node, *voorbereiden afhandeling vergunning*, holds the preparatory activities of issuing permits. The second sub-node, *afhandeling vergunningaanvraag 3.4 procedure*, indicates one possible procedure for processing a permit request. The third sub-node indicates the second possible procedure for processing a permit request. In the next sections the process logic of the sub-processes represented by these nodes is elaborated.



6-3 Process Verlenen Vergunning

Preparation

This diagram depicts the process logic of the sub-process voorbereiden afhandeling vergunning.

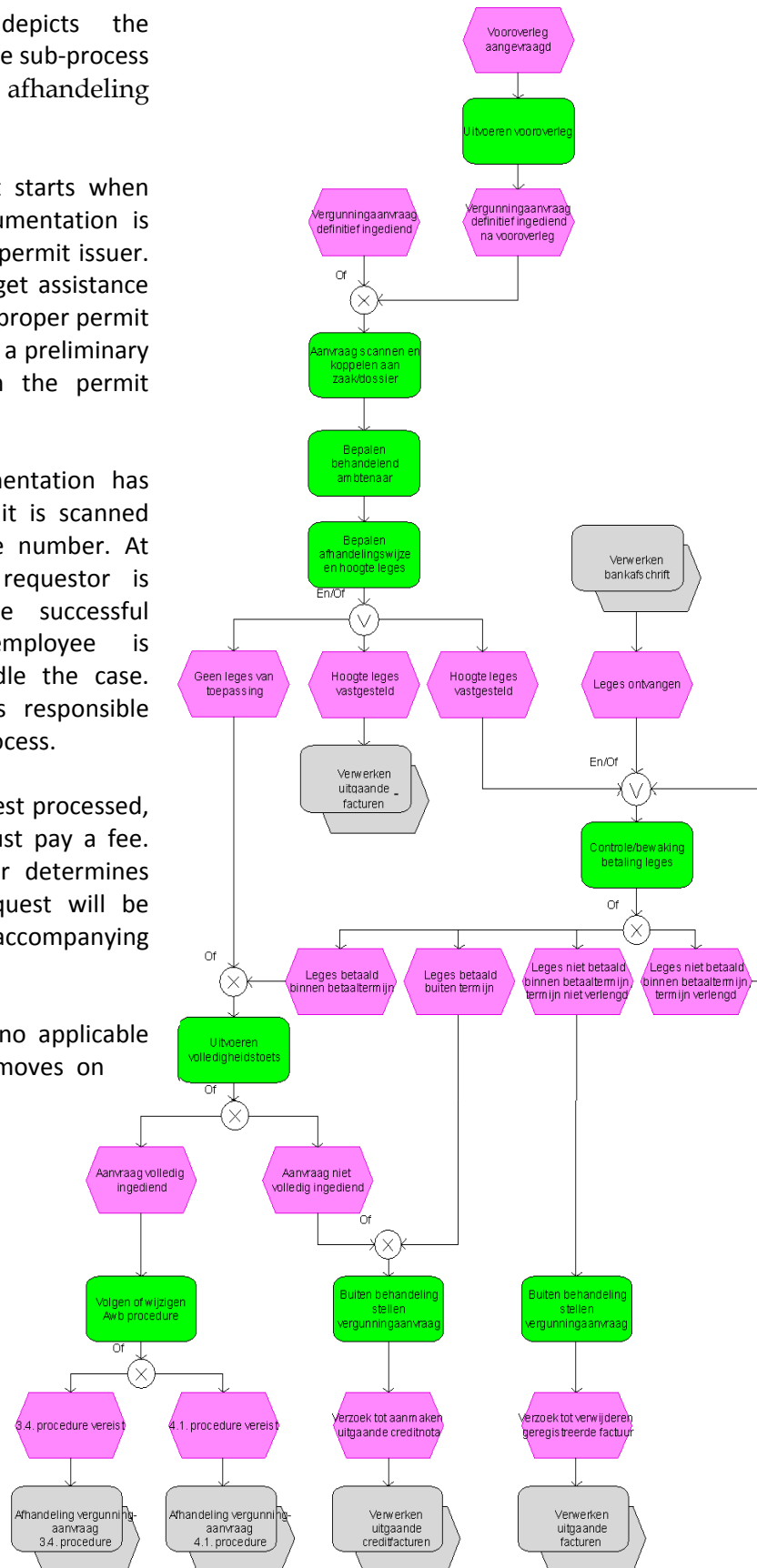
A permit request starts when the request documentation is submitted to the permit issuer. A requestor can get assistance in submitting the proper permit request by having a preliminary consultation with the permit issuer.

Once the documentation has been submitted, it is scanned and given a case number. At this point the requestor is informed of the successful receipt. An employee is assigned to handle the case. This employee is responsible for the further process.

To have the request processed, the requestor must pay a fee. The permit issuer determines the way the request will be handled and the accompanying fee.

When there are no applicable fees, the issuer moves on to check the completeness of the submitted documentation. When a fee is applicable, the issuer sees to it that an invoice is sent to the requestor.

In the diagram to the right, the process seems



6-4 Preparation

to jump away at the point of billing and jumps back once the bank statements have been processed.

The bank statements show the fees that have been received in the current period, so a check is carried out to see if the fees for the specific case have been paid already.

If the fee has been paid on time, the process moves on to check the completeness of the documentation. If the fee has not been on time, there are three possibilities. Either the requestor has asked for more time, or the requestor has paid too late or the requestor has made no payment at all. In the first case the fee payment is checked again when the next batch of bank statements arrive. In the second case the permit request process is ended and the payment is returned to the requestor. In the third case the permit request process is ended as well but there is no payment to return, so the invoice records are simply discarded.

If all has gone well and the documentation is complete, then the permit issuer decides the AWB procedure that should be followed. At this point in the diagram, the process is seen to jump away to either the 3.4 procedure or the 4.1 procedure.

Handling Procedure 3.4

This diagram depicts the process logic of afhandeling vergunningaanvraag 3.4 procedure.

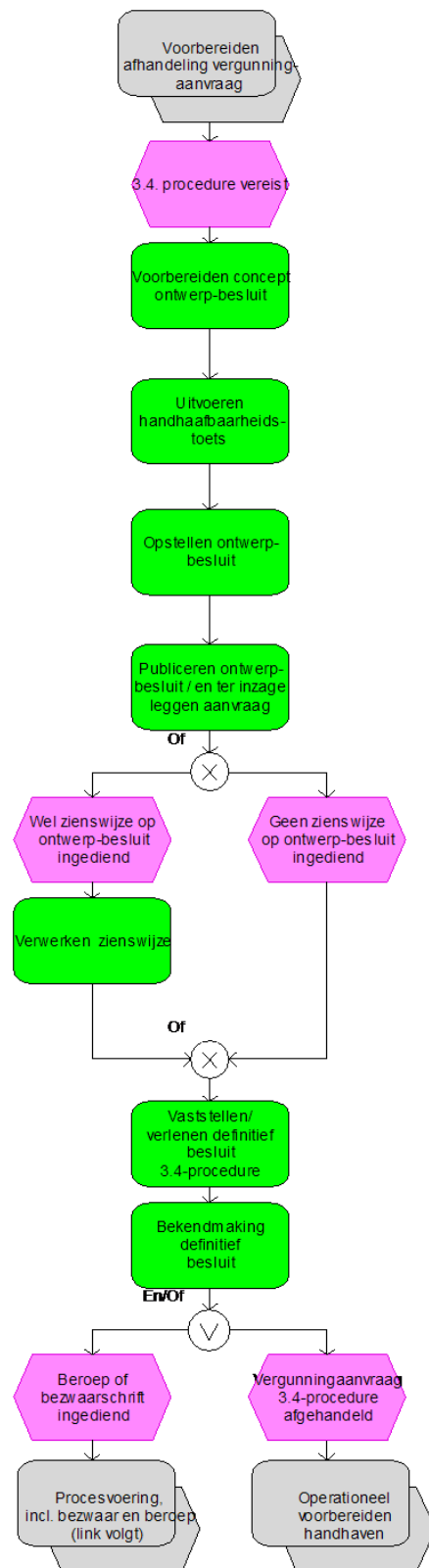
Procedure 3.4 is somewhat lengthier in execution than its 4.1 counterpart. It starts with the creation of a concept of the resolution. At this point, the permit issuer may ask internal and external parties for advise.

A copy of the concept is sent to the department Handhaving so that the enforceability of the permit can be checked. Based on the outcome of the check, the resolution is drawn up. If the permit is partly or completely denied, the permit requestor is informed and given the opportunity to provide contrary views. The resolution is also published in one or more newspapers. The publication is meant for stakeholders that cannot be reached directly and pertains to: the intended activities, the location of the activities, and the period during which (contrary) views on the permit can be submitted.

When no views are submitted, the resolution passes. When views are submitted, copies are sent to the requestor for a reaction. The views and the reaction are taken into account in the next version of the resolution.

The definitive resolution is drawn up and the decision is published in newspapers along with the changes due to the views and with mention of the period for objection. Once the period for objection has passed, the dossier is closed and the request is considered done. The dossier is passed on to the department Handhaving. In the diagram the 3.4 process is seen to end with a jump to the process for preparing enforcement. In case there were objections, the process bezwaar en beroep is followed instead.

Afhandeling vergunningaanvraag 3.4-procedure



6-5 Procedure 3.4

Handling Procedure 4.1

Procedure 4.1 starts with determining whether or not views should be gathered on the permit request. A stakeholder analysis is performed to determine who, if anyone, should be invited to present views.

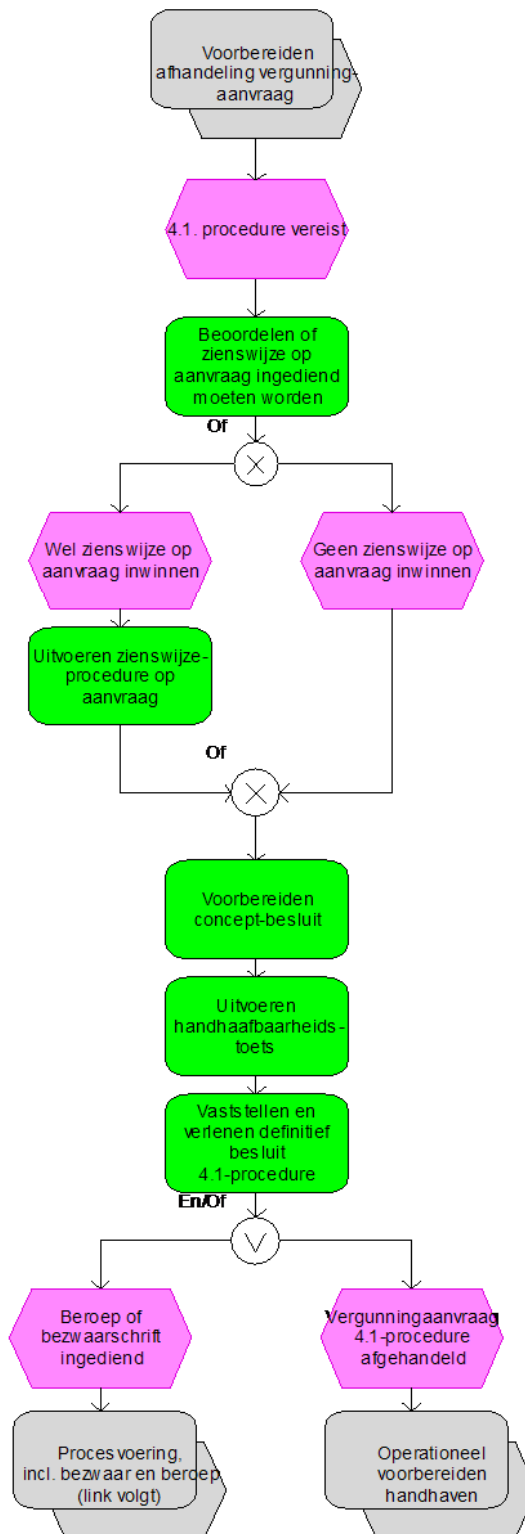
If no views are required a concept resolution is drawn up. Otherwise, a copy of the permit request is sent to the stakeholders. For stakeholders that cannot be reached in such a direct way, a public announcement is made in the newspapers.

If there are views, they are sent to the requestor for a response. Then all feedback is incorporated into the new concept.

Once the concept has been drawn up, a copy is sent to Handhaving to check the enforceability.

This result is taken into account when documenting the final decision. In case of environmental permits, the head of the permit department is required to sign the permit. In other cases it is the head of the district that must sign. Once the document has been signed, the procedure is finished and operational preparation of enforcing starts.

Afhandeling vergunningaanvraag 4.1- procedure



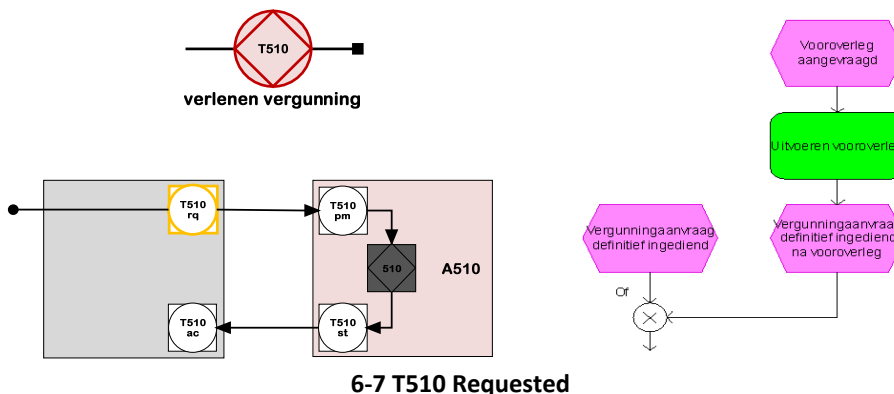
6-6 Procedure 4.1

6.5 Correlation between the Models

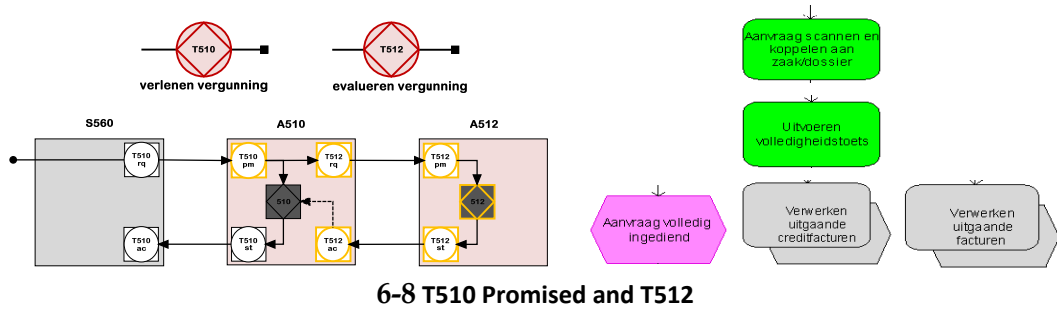
It is known that the DVL model of issuing permits and the UPP process of issuing permits are about the same organization area. So there should be some kind of correlation between what is present in the DVL model and what is present in the UPP model. In this section an attempt is made to clarify where the models correlate and where they do not.

First, the issuing permits DVL and the issuing permits UPP are related to each other at layer four. In terms of labeling the DVL model uses Dutch equivalents of *issuing*, *evaluating*, and *testing* the enforceability of permits. Also collecting tax on permits, and updating permit types. Instead, the UPP model shows terms equivalent to *preparing* and *handling* permit requests. The relationship between the content of the DVL model and that of UPP layer 4 is thus not obvious from the labels that are used.

Next, the transactions are related to the activities at layer 5. Here some correlation is observed. Transaction **T510** verlenen vergunning can be considered **requested** once the permit request documentation has been definitively submitted. Any preliminary consultation is thus part of requesting T510. In figure 6-7 the request phase is visible as all events and activities up to and including the event vergunningaanvraag definitief ingediend or up to and including the event vergunningaanvraag definitief ingediend na vooroverleg.



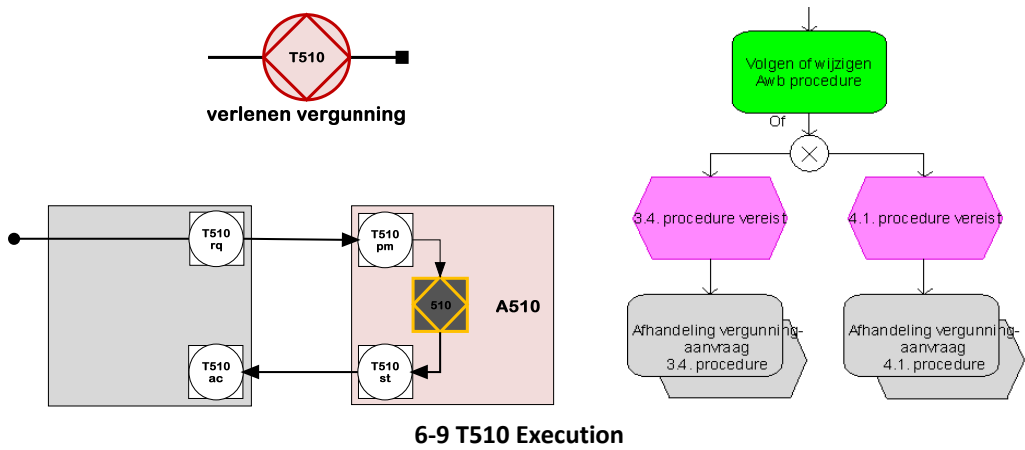
T510 becomes **promised** if 1) the processing fees, if applicable, have been paid, and 2) the permit request documentation is complete. This second criteria can be related to T512 evalueren vergunning. If the fees are not paid or if they are paid too late, or if the documentation is incomplete, then T510 is **declined**. In figure 6-8 the promise phase starts with the activity aanvraag scannen en koppelen aan zaak /dossier. When all goes well it ends with the event aanvraag volledig ingediend. When things go wrong it ends with a jump to either verwerken uitgaande creditfacturen or verwerken uitgaande facturen.



6-8 T510 Promised and T512

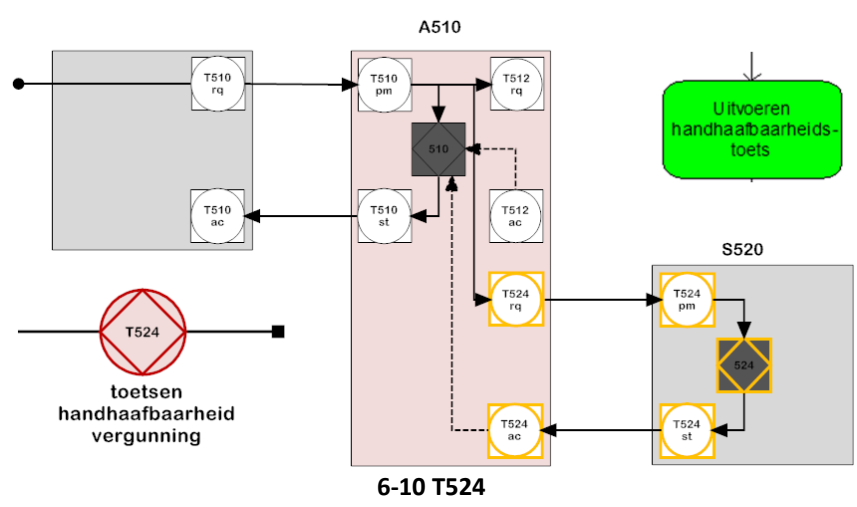
Once T510 has been promised, execution begins. The permit issuer determines if the AWB procedure indicated in the request is indeed appropriate. If it is appropriate, the permit is issued in accordance with the indicated procedure. Otherwise, the permit issuer follows the other procedure. Therefore **execution** of T510 starts with determining the right handling procedure, activity volgen of wijzigen AWB procedure, and continues in one of the two processes afhandeling vergunningaanvraag 3.4 procedure, or afhandeling vergunningaanvraag 4.1 procedure.

49



6-9 T510 Execution

During the execution of T510, in both the 3.4 and the 4.1 procedure, the enforceability of the permit is checked. This corresponds with T524 toetsen handhaafbaarheid vergunning. This is visible in both figure 4 and 6-6 as activity uitvoeren handhaafbaarheidstoets.



6-10 T524

None of the activities of the three processes correspond to **T513** innen heffingen, **T907** betalen verontreinigingsheffing and **T511** ambtshalve ingrijpen. Talks with BPS revealed that while there is a DVL model specifically for water management permits, there is no UPP model specifically for water management permits. This is why the two business processes in the water management DVL of issuing permits are missing in the general UPP model of issuing permits.

6.6 Conclusion

In this chapter the DVL model for issuing permits and the UPP model for issuing permits were analyzed and a listing was made of the content related correlations between them.

The business processes for collecting taxes and updating permit types are not reflected in the UPP cluster-functions and activities. The business process for issuing permits is reflected in the activities of the accompanying UPP processes, but this is not done in a systematic manner. By 'not systematic' is meant that one cannot say that because there is a transaction A with a sub-transaction B, one will also have a process A with a sub-process B. Nor can one say that because there is a transaction A, there will be a set of activities that reflect the process steps of A.

For example some of the process steps of T510 were found and listed in the section above, but there is no such listing for the process steps of T512. In fact, the whole of T512 is represented by exactly one UPP activity.

The conclusion that is reached based on this analysis, is twofold:

1. firstly, one cannot document the systematic coupling that exists between DVL and UPP because there currently is no systematic coupling between DVL and UPP. The coupling that currently exists is of the 'gut feeling' variety.
2. secondly, in order to provide a systematic coupling between the models of DVL and those of UPP regardless of the previously mentioned limitation, changes need to be made to both the DVL and UPP models to make them consistent with each other.

The next chapter regards the process of making DVL and UPP models consistent with each other.

Phase TWO – Building the Connection

7 Method of Coupling

After the first phase of research the DVL-UPP connection proved to be more complicated than a model-to-model and type-to-type mapping. While connected in some ways the architectures were also disconnected in others. Therefore the second phase of research moves away from seeking a coupling and moves towards developing one. In this section the reader is introduced to a new kind of coupling which is consistent and takes place in five stages. Next, the reader learns the steps to achieve this coupling and the guidelines by which the coupling can be kept in a consistent state.

7.1 The 5-Stage Coupling

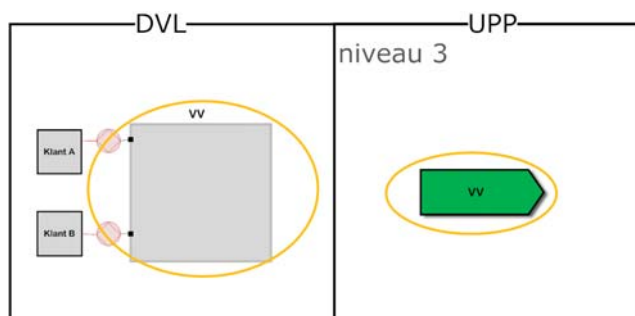
In chapters 5 and 6 it was shown that some correlation exists between the models of the architectures but that some work is required to make them fully consistent. This consistent state is now discussed in tangible terms.

Stage 1 - Delineation

In section 6.2 it was shown that the first step to couple DVL and UPP is to find in DVL and in UPP a model that is about the same part of the organization; the DVL model Vergunningverlening had a matching process in UPP. We know from section 3.5 that this process resides on UPP's layer 3. However, this match was inexact because the DVL model encompassed a larger area than its UPP counterpart.

And so in the ideal situation *the delineation of DVL models should be consistent with the delineation of UPP processes, UPP layer 3.*

This recommendation starts the first stage of improving consistency between DVL and UPP. Figure 7-1 depicts this notion; the responsibilities captured in the DVL model named VV, are the same as those captured in the UPP model named VV.



7-1 Stage 1 - Delineation

The benefit that this stage of coupling would give architects is that it would allow them to test their assumptions of what responsibilities belong together by comparing their own delineations with those of the other team.

This stage of coupling gives users the benefit of finding related areas in the different architectures. For example, a domain in DVL can be used to find the corresponding organization and function view in UPP. Conversely, the name of a process in UPP can be used to find its respective DVL model.

Stage 2 - Structure

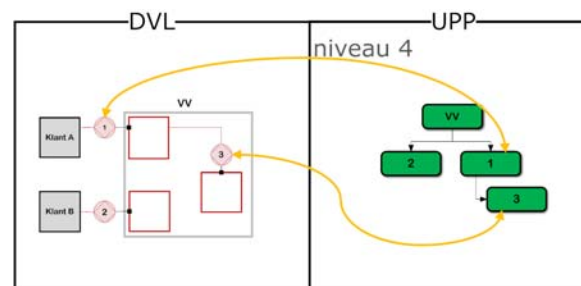
Section 5.1 showed that there is a promising way to consistently couple DVL with UPP layer 4. It requires the transactions and actor roles of DVL to match the function and organization view of UPP. However section 6.5 showed that the structure of the Vergunningverlening DVL model and the structure of the accompanying UPP model (layer 4) were different; the architectures considered different concepts essential. The following recommendation starts the second stage of improving consistency between DVL and UPP and coupling DVL and UPP.

Ideally, *the structure of DVL models should be consistent with the structure of UPP processes, UPP layer 4.*

By saying that the structure of different models is consistent, we mean that:

- There is a clear correspondence between the elements across the models.
- Relationships between elements within one model are also present in the other model.

Figure 7-2 depicts this notion.



7-2 Stage 2 – Structure

In the example of figure 7-2 the orange pointers indicate that:

- On the side of DVL there are transactions #1, #2, and #3. Correspondingly, on the side of UPP we see sub-processes #1, #2, and #3.
- On the left hand side, one can see that transactions #1 and #2 are initiated from outside of the border of the model, while transaction #3 is a “child” of transaction #1. On the right hand side we see the same relationship; sub-processes #1 and #2 are coming directly from the root node, and sub-process #3 is a child node of sub-process #1.

It is an established fact that the DEMO theory, on which DVL is based, is well-suited to determine the completeness of a model. This is the case because finding a single step belonging to a transaction is sufficient to determine that transaction exists. Finding a single step belonging to a different transaction is sufficient to determine that an additional transaction exists. By asking targeted questions about other possible transactions a DEMO Professional⁶ can systematically track down and document every transaction. The chance that a professional DEMO practitioner could miss all clues leading to the existence of a transaction are very small.

By relating DVL and UPP as described at this stage, architects can leverage the virtue of completeness to test the completeness of UPP, e.g. knowing that all necessary transactions

⁶ A DEMO Professional is someone who is certified in to develop DEMO models and consult on DEMO theory.

are present in DVL, they can verify whether all transactions in DVL have been accounted for in UPP.

Moving from the other direction, this stage makes the hierarchical structure of UPP available to the architects of DVL. DVL shows transactions and actor roles and how they are interrelated. But it cannot show the larger the business processes at a glance; the reader must analyze the model in order to determine the business processes. The tree structure however, due to its lack of actor roles and cyclical structures, does provide an overview of business processes at a glance.

The benefit that this stage gives to users is fine-grained control over how they move between architectures. Users can now select specific transactions or specific sub-processes and look up their counter parts in the other architecture. For example, an employee working on sub-process #3 in UPP can relate this to transaction #3 in DVL. And an employee working on transaction #2 in DVL can find sub-process #2 in UPP.

Stage 3 - Full Circle

On the side of UPP, the relationship between the sub-processes at layer 4 and the process logic at layer 5 is clear from sections 3.5 and 6.4. Each sub-process at layer 4 has process logic at layer 5 and vice versa.

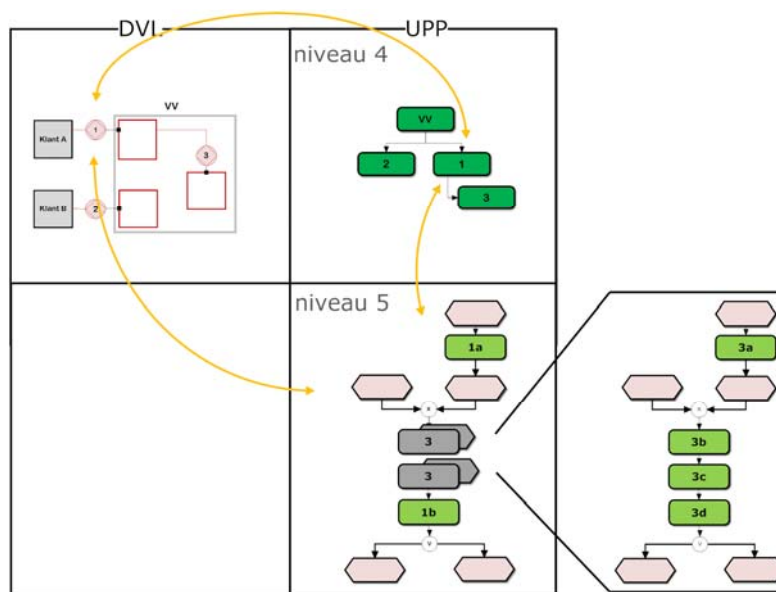
Ideally, this consistency between UPP layers 4 and 5 is maintained. *The structure of UPP processes, layer 4, should be consistent with the structure of UPP process logic, layer 5.*

54

Note that when stages 1, 2, and 3 are combined that *the structure of UPP process logic, layer 5, should be consistent with the structure of DVL models.*

What this means is that:

- When there is a one-to-one match between DVL concepts and UPP layer 4 concepts, and
- there is a one-to-one match between layer 4 concepts and layer 5 concepts,
- then there is also a one-to-one match between layer 5 concepts and DVL concepts. This achieves a full circle of consistency.



7-3 Stage 3 – Full Circle

See the example depicted in figure 7-3 where the orange pointers indicate the following:

- Transaction #1 has a one-to-one relationship with sub-process #1 at UPP layer 4.
- Sub-process #1 has a one-to-one relationship with the process logic at layer 5.
- And thus the process logic at layer 5 has a one-to-one relationship with transaction #1 in DVL.

The benefit of this stage is the ease with which consistency between the models is achieved and maintained. The architect knows how an addition to the process logic would lead to an addition at layer 4 and in DVL. Any change that has not been properly propagated will stand out. It also prevents architects from having to model in DVL what has already been modeled in UPP.

From the user's perspective, the main value of this stage is the ability to navigate from the ontology in DVL to the UPP implementation details, all the way down to the applications and work instructions. The opposite is also true, that one can understand ones place in the whole organization not just the current process.

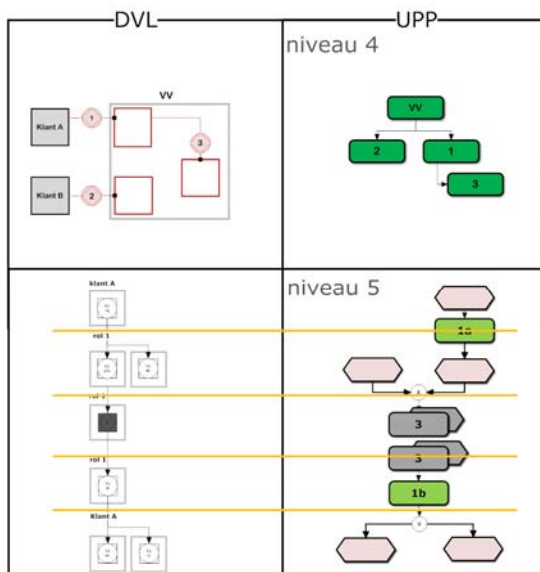
Stage 4 - Reflecting the Transaction Pattern

Every transaction in DVL has the transaction pattern within it. If there is a systematic coupling between the DVL transactions and the UPP process logic then this pattern can be juxtaposed on the process logic in a systematic manner as well.

The fourth stage of consistency is achieved when *the notion of process steps as put forth by the DEMO theory is reflected in the process logic presented in UPP.*

In figure 7-4 the orange line separators represent the sectioning of the UPP process logic on the right hand side in accordance with the DEMO process steps on the left hand side. From top to bottom these are: the request, promise/decline, execute, state, and accept/reject steps of the standard transaction pattern.

For information on the quit and stop steps of the standard transaction pattern see section 7.4. For a brief discussion on why the standard transaction pattern was chosen, rather the basic or the complete pattern, also see section 7.4.



7-4 Stage 4

The Benefit that this stage brings to architects of UPP is the immediate view on the parts of the transaction pattern that have been modeled. An architect can see, at a glance, if the request, promise/decline, execute, state, and accept/reject logic has been modeled. The standardized form of the process logic is enough to see this information without first reading the text. This gives architects an additional way to check the completeness of their models.

The user can also see at a glance if there are circumstances under which they must decline a request or reject a result. They do not need to read details before this information is clear to them. This will allow them to oversee their processes and do their work with more ease.

Stage 5 - Roles to Units

The last recommendation is to make sure that *organizational unit assignments in UPP are in accordance with actor role assignments in DVL*. DVL shows which roles are responsible for certain transactions. By determining which organizational units could take on a particular role, one could better evaluate the correctness of responsibility assignments in UPP.



7-1 Stage 5

The benefit of this stage is to have a systematic way to assign responsibility based on what is best for the organization, and to eliminate any “this job has always been ours” kinds of assignments.

The Five Stages of Coupling

Five stages have been identified where DVL and UPP can and should be made consistent with each other. By making the models consistent with each other a systematic coupling is built up and each stage strengthens this coupling.

In short, the ideal coupling between UPP and DVL occurs in the following five stages:

1. The delineation of DVL models should be consistent with the delineation of UPP processes, UPP layer 3.
2. The structure of DVL models should be consistent with the structure of UPP processes, UPP layer 4.
3. The structure of UPP processes, layer 4, should be consistent with the structure of UPP process logic, layer 5 and therefore the structure of UPP process logic, layer 5, should be consistent with the structure of DVL models.
4. The notion of process steps as put forth by the DEMO theory should be reflected in the process logic presented in UPP.
5. Organizational unit assignments in UPP should be in accordance with actor role assignments in DVL.

7.2 Benefits of the Coupling

The benefits that were discussed in section 7.1 are listed here for clarity. They regard new productivity-related capabilities for architects and architecture users.

The architects will benefit from the coupling in the following ways:

- The recommended modeling practice will make the **consistency** between DVL and UPP verifiable. For instance, it will be visible when an area has been delineated differently in UPP than in DVL. It will also be visible when the relationships between processes in UPP are different from the relationships between transactions in DVL. By applying the recommended modeling practice, changes in one model will spur changes in the other. By acting on the indicators architects can maintain the consistency of the models.
- By leveraging the DEMO theory, UPP architects will also be able to test the **completeness** of their models. For example, it will be visible whether all transactions in DVL have counterparts in UPP. It will also be visible whether all aspects of the transaction pattern have been accounted for in UPP process logic. The recommended modeling practice thus makes the completeness of UPP models verifiable in a new way.

The users of the architectures will benefit from the coupling in the following ways:

- Coupling the transactions in DVL to the processes in UPP prevents employees from doing **unnecessary work**. The users of DVL will know how to get related information out of UPP and will not need to gather, analyze and processes data which is already been compiled into UPP. The users of UPP will be able to see the context of the processes they carry out in DVL and will no longer need to keep track of that themselves.
- Because it leverages the DEMO theory, the recommended modeling practice will increase the **clarity** of UPP models; employees will better understand why they perform certain activities (as part of promise, or execute, etc) and what the context of the work they are doing is (part of a larger business process).
- Because the relationship between actor roles and process logic is documented, managers can better decide who should be responsible for the execution of each process and for performing the activities within them; by determining which organizational units can take on a particular role, managers can determine who is eligible to perform the task. In this way they can eliminate assignments that would be detrimental for the organization and instead assign **responsibilities** based on what is best for the organization.

Note that these capabilities are only the immediate benefits; the real value of the coupling comes when employees start to leverage the capabilities of the coupling, and start to perform tasks and achieve results which would not have been possible before. At least, not without a large amount of time and effort. To be documented, this *real* or *greater* value would require a thesis of its own.

Nevertheless, this document provides an inventory of the opportunities that the coupling is likely to create, along with an assessment of these opportunities. The inventory was developed during the GDSS session discussed in chapter 8.

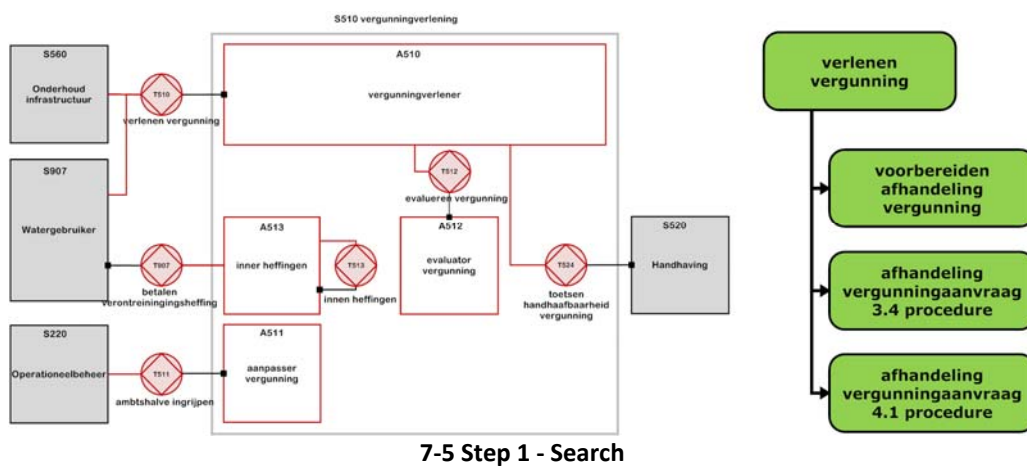
7.3 Achieving Stage 1 and 2

Building up the five stage coupling explained in section 7.1 requires certain modifications to be made to existing models. In this section we go over the required actions to achieve stages 1 and 2 of the coupling.

Step 1 - Search

The first step is to search for two areas, one in DVL and one in UPP, that are about the same part of the organization. We need to compare these areas to each other in order to make the first stage of coupling. In this case we take the DVL model vergunningverlening and UPP model verlenen vergunning.

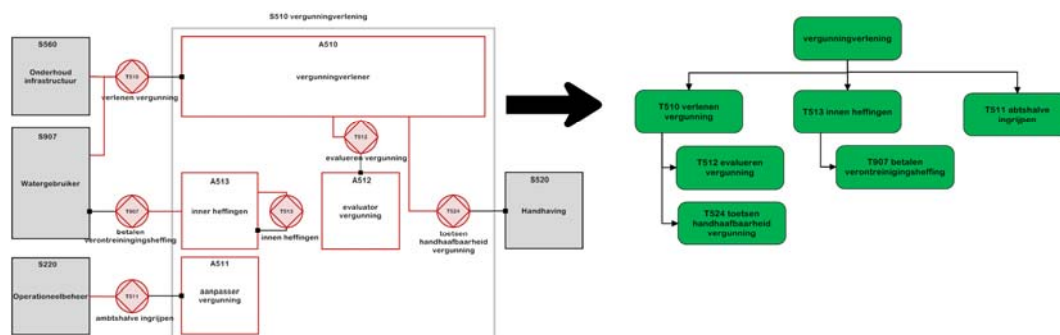
58



7-5 Step 1 - Search

Step 2 - Generate Tree

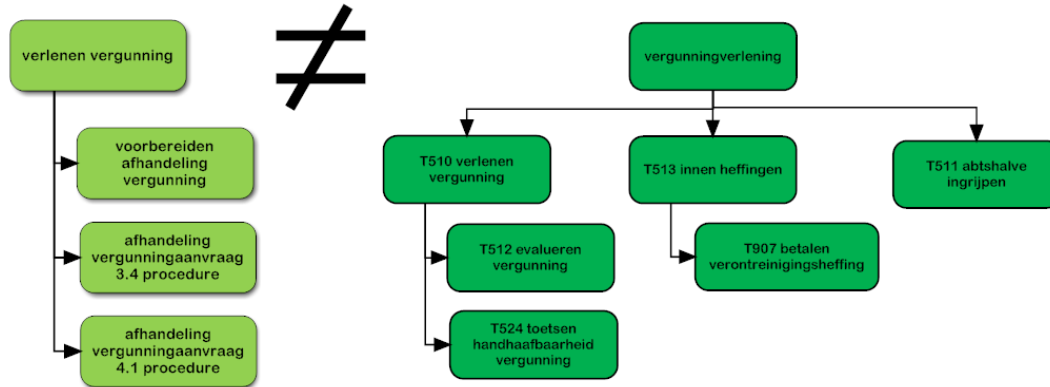
Now there are two entirely different kinds of models to compare. In order to determine if the DVL model is delineated similarly to the UPP model the following trick is used: convert the DVL model into a tree. Two trees can easily be compared. Make the designation of the DVL model into the root of the tree. Then make each business process into a branch of the tree.



7-6 Step 2 - Generate Tree

Step 3 - Compare (and Unite)

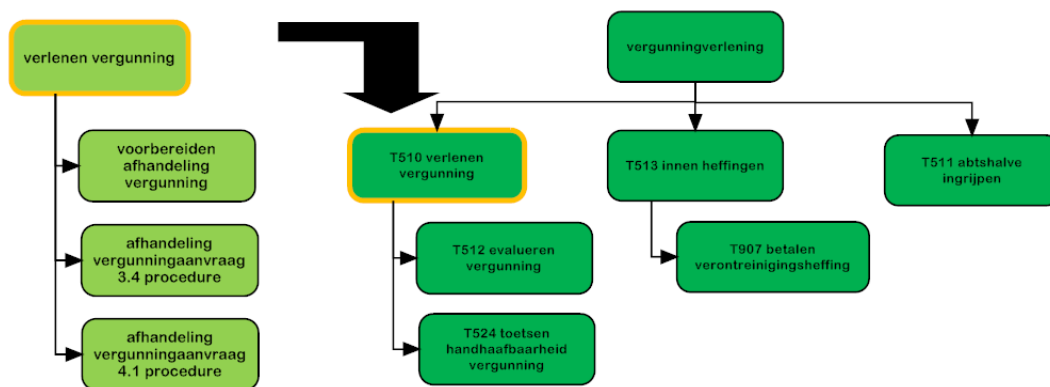
If the generated tree looks like the original tree then the DVL and UPP models are delineated and structured in the same way. Unfortunately the resulting tree clearly does not resemble the original tree.



7-7 Step 3 - I

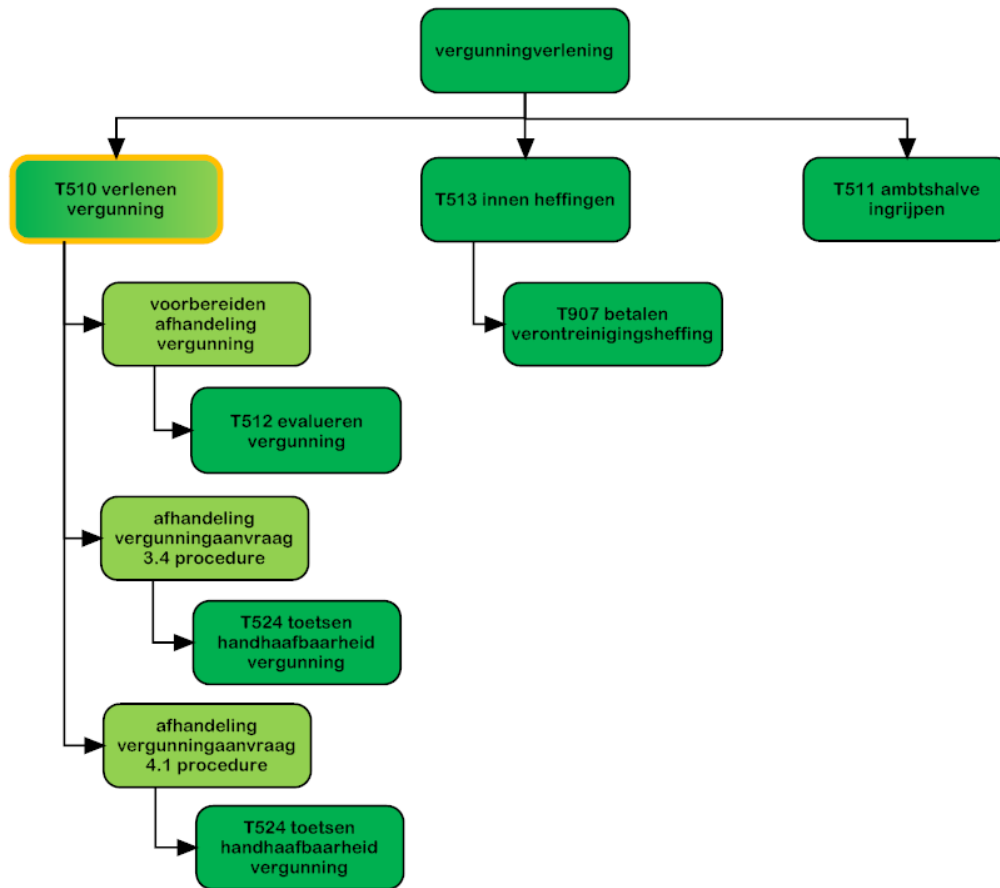
Using the analysis in section 6.5 about the correlation between the DVL and UPP models of Vergunningverlening, we can show that the original tree matches only the left column of the generated tree.

59



7-8 Step 3 - II

Now that is known how they fit together, the next step could be to make a union of the trees or to select the cross-section. Since we do not want to risk losing any essential information yet, we choose to unite the trees.



7-9 Step 3 - III

Step 4 - Ontologize

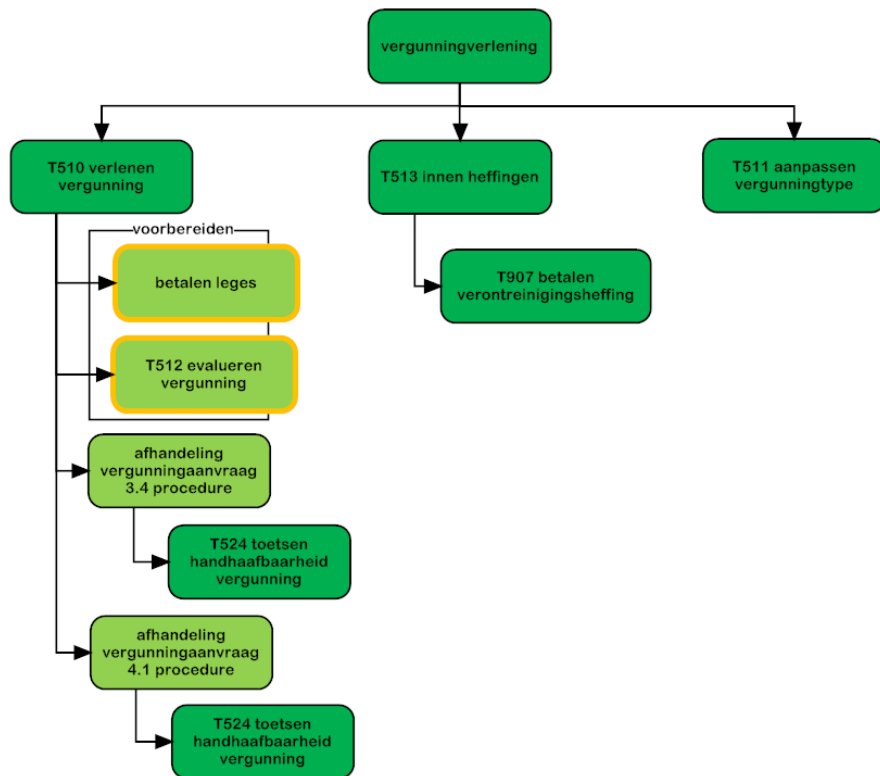
Prior to step 3 the delineation of the DVL model was larger than that of the UPP model. Unifying the models ensures that any DVL model generated from the new tree would match it exactly. However, there is still some unevenness in the tree. Since the DVL model is based on DEMO its nodes are ontological in nature. The UPP nodes are a mix of ontological, infological and datalogical nodes with the greater majority being infological. The next step is thus to “ontologize” the tree. This means that we must:

1. Remove non-ontological nodes
2. Add missing ontological nodes
3. Improve labels to accentuate their ontological meaning
4. Provide transaction numbers for unnumbered nodes.
5. Build up the Transaction Result Table.

In figure 7-10 node *voorbereiden afhandeling vergunning* is removed as it is quite vague and can be replaced with a clearer node. In section 6.5 *evalueren vergunning* was shown to correlate with only one check activity in UPP. It did not correlate with a range of activities which could justify marking it as a transaction. It should therefore be removed as well.

Note however that the reverse is also possible. It is possible that this is an example of a transaction in DVL that is insufficiently worked out in UPP. Such situations should be judged by the EAC en BPS teams. For now the node is left in the tree because it will provide interesting process logic later on in appendix 12.3. Further figure 7-10 depicts the adding of node *betalen leges* in the place of node *voorbereiden afhandeling*

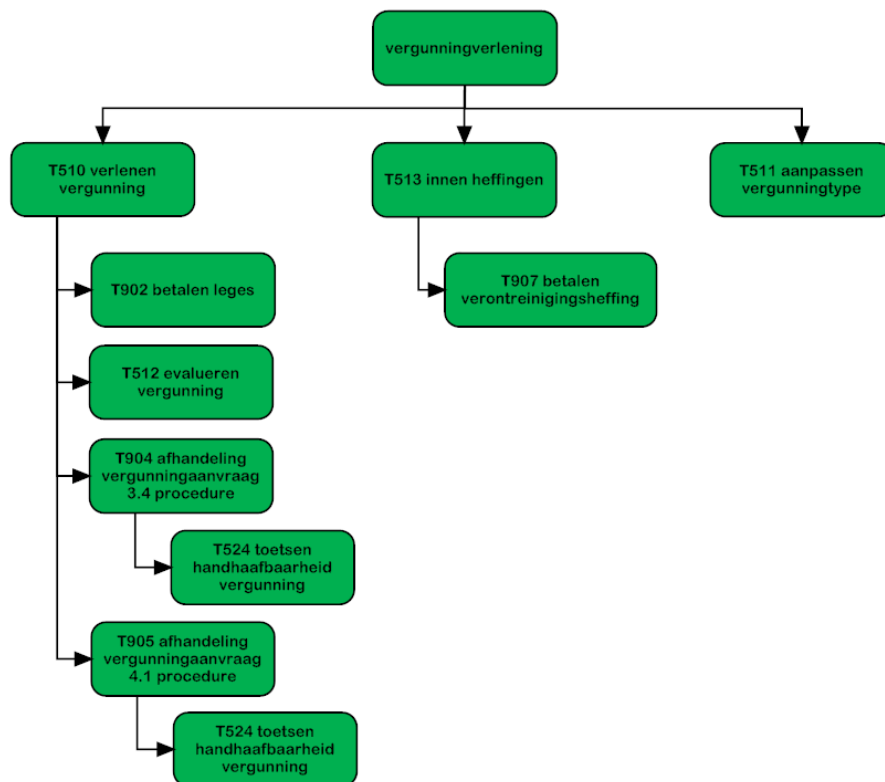
vergunning. Node ambtshalve ingrijpen could be modified into aanpassen vergunningtype, making its ontological meaning more obvious.



7-10 Remove Non-Ontological Nodes and Add Ontological Nodes

61

After adding transaction numbers to the new nodes the following result emerges.



7-11 Provide transaction numbers for unnumbered nodes.

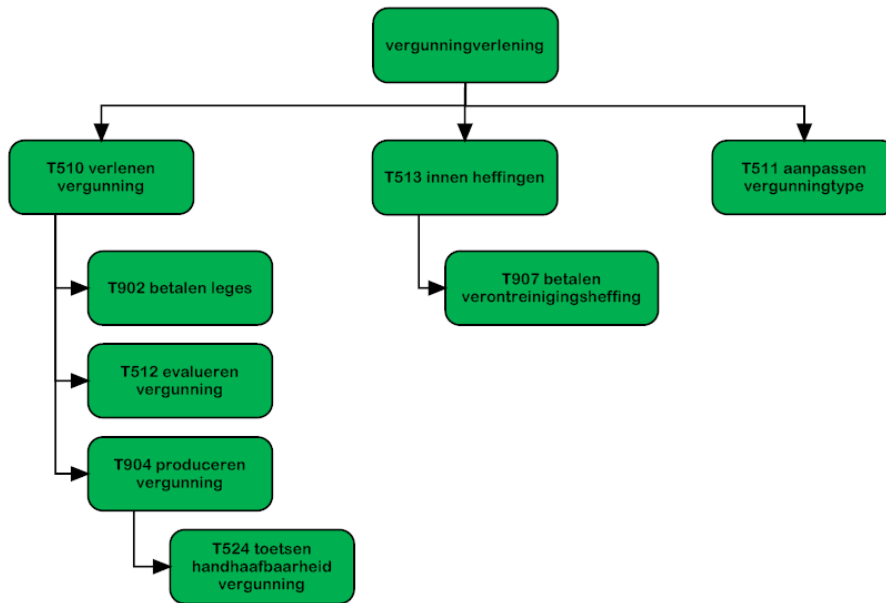
When the tree has been updated and is completely ontological then it is essentially ready to be reverse engineered into a DVL model. Building the TRT is an effective way to test if all nodes are indeed ontological. If the transaction result cannot be properly defined, than the node is likely not ontological and must either be modified or removed.

For example The TRT shows us that the result of *afhandeling vergunningaanvraag 3.4* procedure is the same as the result of *afhandeling vergunningaanvraag 4.1* procedure, namely *vergunningaanvraag V is afgehandeld*. Whether the handling has occurred according to procedure 3.4 or 4.1 is not ontological. The TRT helps us to see that.

Transaction #	Transaction Type	Result #	Result Type
T510	verlenen vergunning	R510	vergunning V is verleend
T511	aanpassen vergunningtype	R511	vergunningtype VT is aangepast.
T512	evalueren vergunning	R512	evaluatie van vergunning V is voltooid
T513	innen heffingen	R513	heffingen van vergunning V over periode P zijn geïnd
T524	toetsen handhaafbaarheid vergunning	R524	handhaafbaarheid van vergunning V is getoetst
T902	betalen leges	R902	Leges voor vergunning V over periode P zijn betaald
T904	afhandeling vergunningaanvraag 3.4 procedure	R904	vergunningaanvraag V is afgehandeld
T905	afhandeling vergunningaanvraag 4.1 procedure	R905	vergunningaanvraag V is afgehandeld
T907	betalen verontreinigingsheffing	R907	verontreinigingsheffing volgens vergunning V over periode P is betaald.

7-2 New Transaction Result Table

After the TRT test, nodes *afhandeling vergunningaanvraag 3.4* procedure and *afhandeling vergunningaanvraag 4.1* procedure are replaced with a single ontological node *produceren vergunning*. The following tree emerges.



7-12 After the TRT Test

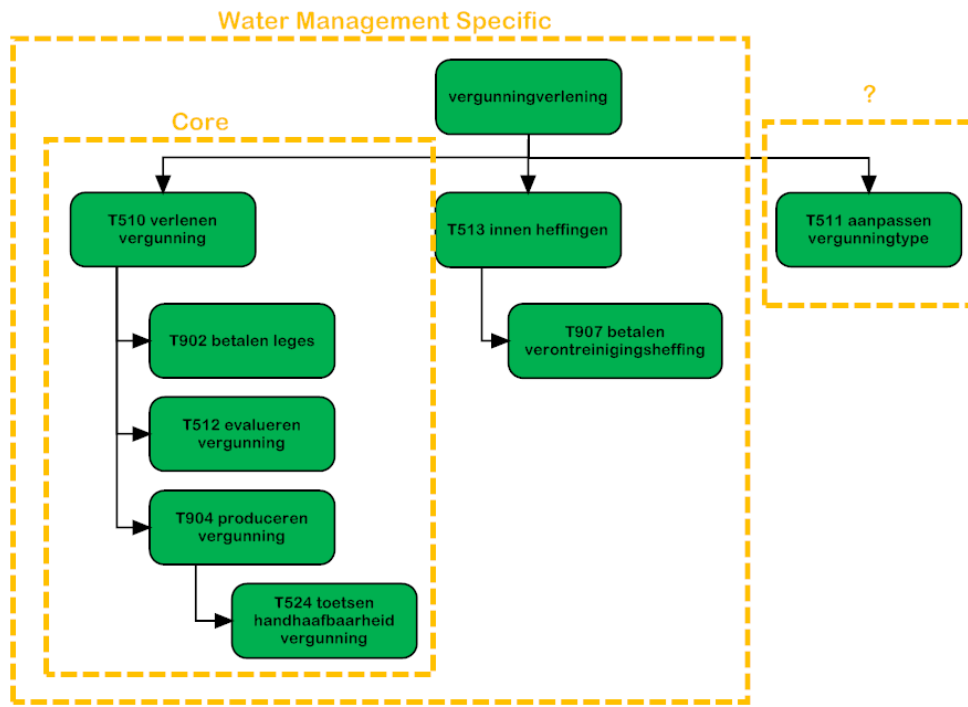
Step 5 - Trim

In the previous step we had taken the union of both trees to insure no information was inadvertently lost. Now we must determine if this delineation serves our goal.

63

For example, Issuing permits in the area of water management requires the paying of taxes, but outside of the area of water management this is not necessarily the case. Therefore, when developing the core model of issuing permits only the left branch should be taken into account. However, when making a model specific to water management, the left and the middle branches should be taken into account.

For both mentioned cases, it is questionable whether the third branch should be taken into account. Is *aanpassen vergunningtype* a matter of issuing permits or a matter of enforcing them, or something else entirely? These kinds of decisions are up to the people of BPS and EAC.

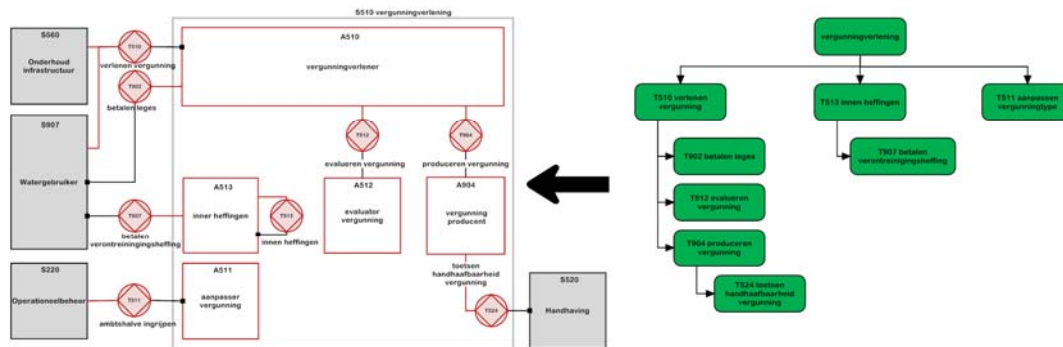


7-13 There is more than One Way to Trim a Tree

Step 6 - Reverse Engineer

64

Once the tree has been properly trimmed of unwanted branches, the last step is to reverse engineer a DVL model out of it. The root of the tree becomes the designation of the DVL model and the branches of the tree become the business processes in the DVL model. In the example below the full tree is used.



7-14 Regenerate a DVL model that Exactly Matches the UPP Model

Step 7 - Add information links

The last step is to add information links to complete the DVL model.

The Seven Steps to DVL-UPP Harmony

To provide an overview, the seven steps to achieving stage 1 and 2 of the coupling are repeated here:

1. Find related DVL models and UPP processes
2. From each DVL model, generate a tree.
 - a. Make the designation of the DVL model into the root of the tree
 - b. Make every business process into a branch of the tree
3. If the trees are not identical, make the union of the trees.

4. Make the new tree ontological:
 - a. Remove non-ontological nodes
 - b. Add missing ontological nodes
 - c. Improve labels to accentuate their ontological meaning
 - d. Provide transaction numbers to unnumbered nodes.
 - e. Build up the Transaction Result Table.
5. Trim the tree.
6. Reverse engineer the tree into a DVL model.
7. Add information links to the DVL model.

Note that in cases of uncertainty the method of Evidence Based Splitting of Organizations (EBSO) can be used. EBSO is a comprehensive tool based on DEMO that consists of organization construction rules and allows managers to calculate a plausible organization split. It gives insights into the costs and benefits of an organization split and has additional value as a method to determine contract topics for working in the aftermath of a split. For more on EBSO see [Op t Land].

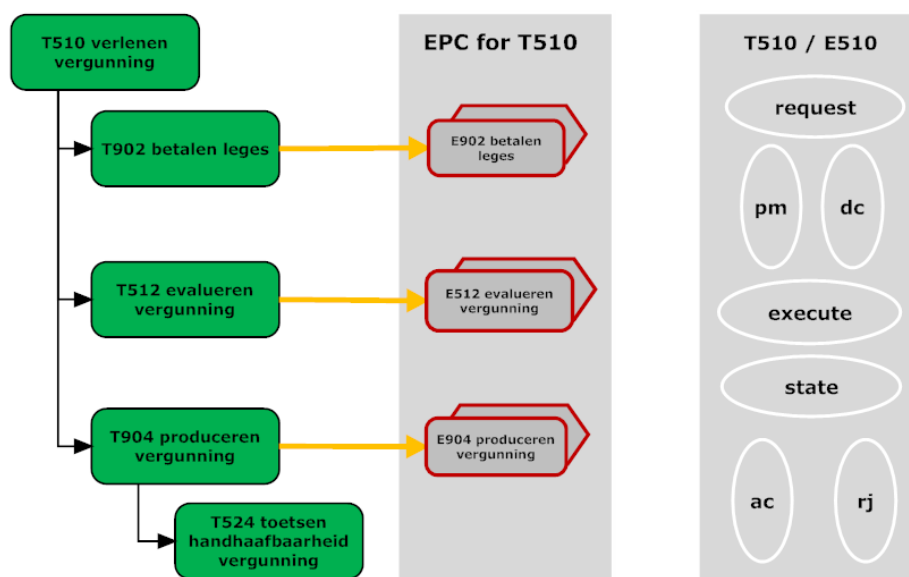
When applying EBSO the alternative steps are:

5. Reverse engineer the tree into a DVL model.
6. Add information links to the DVL model.
7. Split the DVL model according to EBSO.
8. Generate the function tree.

65

7.4 Achieving Stage 3 and 4

In this section stages 3 and 4 of the coupling are demonstrated by building up the process logic for the left branch of figure 7-13 (also visible below in figure 7-15). Nodes T510, T902, T512, T904, and T524 must all be represented in the process logic for the left branch. And the process logic must follow the same structure as in the tree (also see figure 7-3). Therefore we must first create an event-process chain or EPC for T510. Next we must insure that EPCs for T902, T512, T904 are linked to from within the EPC of T510. See the visualization below. And third there must be a link to the EPC for T524 from within the EPC for T904.



7-15 Stage 3 and 4

Furthermore each EPC must be built up according to the transaction pattern. Appendix 12.1 shows the basic, standard and universal transaction patterns. The analysis in section 6.5 showed the presence of request, promise and execute steps in *verlenen vergunning*. The basic transaction pattern would suffice to organize process logic according to these steps. But the analysis also showed the decline process step. So, we can deduce that at least the standard transaction pattern is required. Although the search was not exhaustive, there were no examples that required the universal transaction pattern. Therefore the standard transaction pattern will be used to organize UPP process logic in this document.

Notice that on the far right of figure 7-15 the default locations for the request, promise/decline, execute, state, and accept/reject steps have been highlighted. So, an EPC should start with activities and events for requesting something. Activities for promising or declining should follow underneath. Hereby the left hand side is reserved for promising and the right hand side is reserved for declining. The region for state related activities and events follows underneath that, and the accept/reject region underneath the state region.

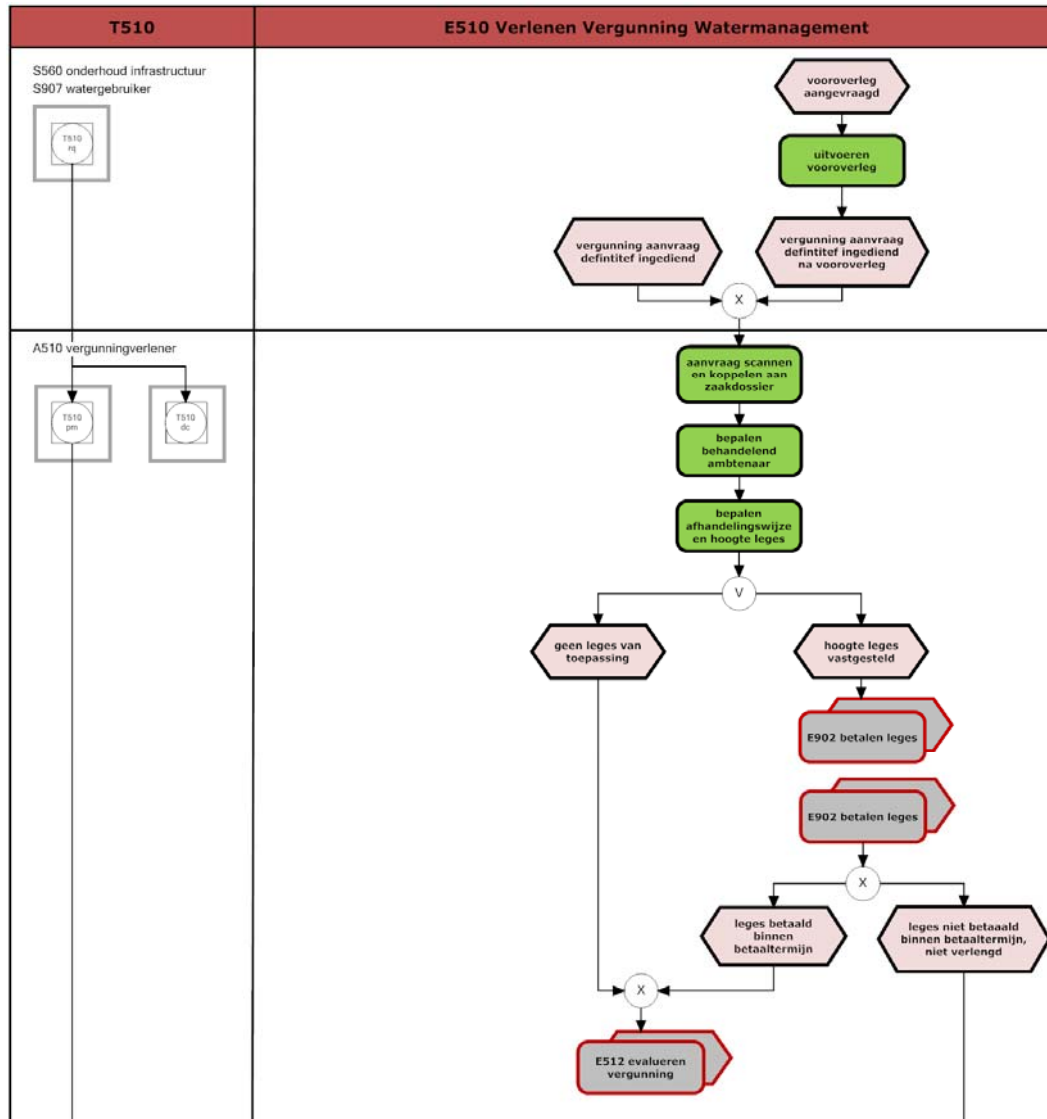
While the transaction pattern also possesses a *quit* and a *stop* step these have not been highlighted for practical reasons; they would clutter the example. I will however *say* that I recommend placing stop and quit process logic within the request and the state sections respectively.

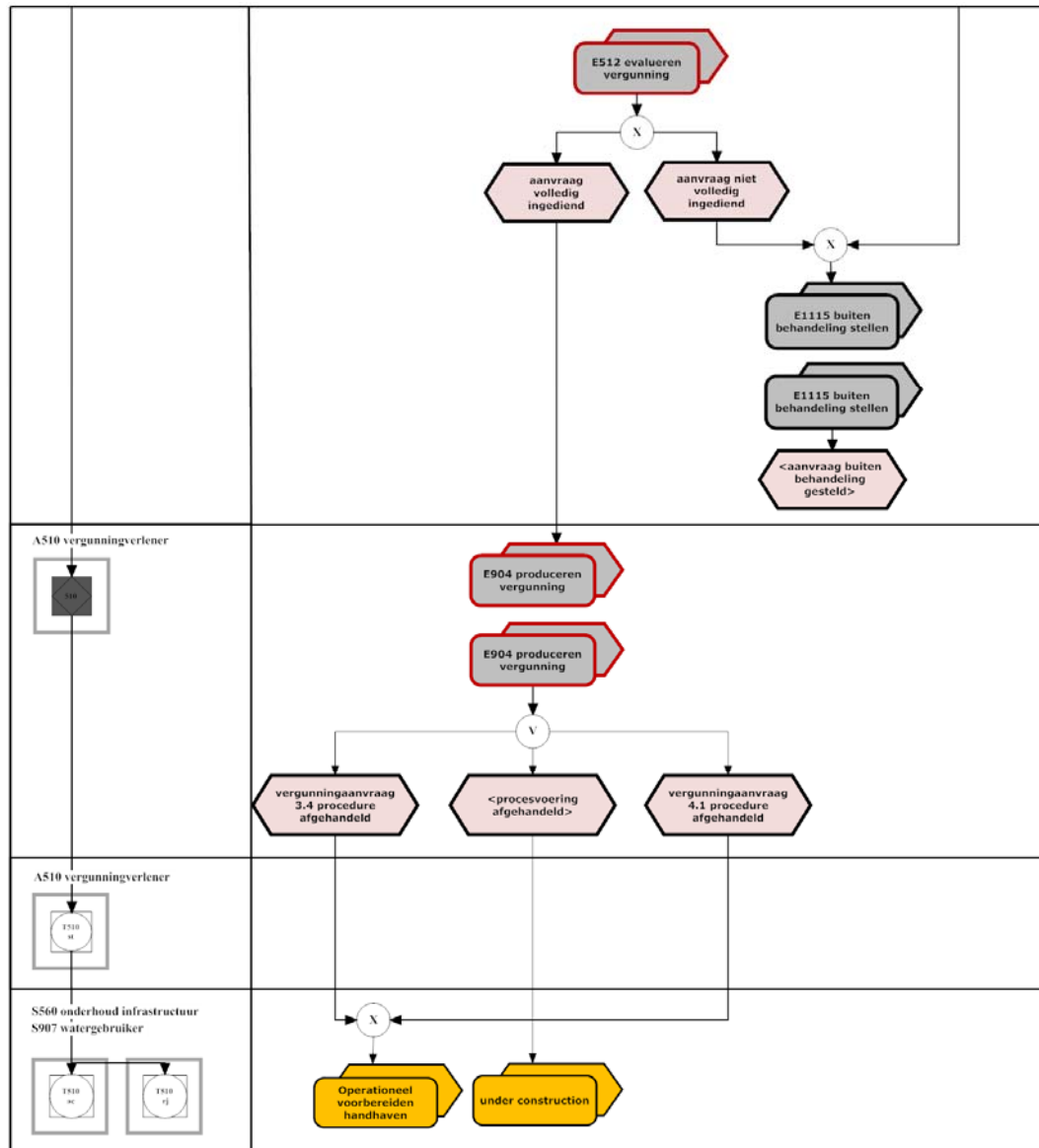
Thus when organizing the process logic it should be clear where the request, promise/decline, execute, state, and accept/reject steps take place. And like in the DEMO process structure diagram, the EPCs should be connected to each other in a systematic way.

Proposed UPP model E510

Using the idea of organizing EPCs according to the standard transaction pattern and interlinking them accordingly, a new layout is proposed for the models of section 6.4.

Below is an example of applying stages 3 and 4 of the coupling to verlenen vergunning. The EPC is organized by process steps and the links to other EPCs are visible with red edge; the edges are colored red to indicate their ontological relationship with DVL. The correlations between DVL and UPP that were discussed in section 6.5 are now visible.





Maintaining the Structure of T510 in E510

The link to **E902** coincides with the sub-node T902. Once event hoogte leges vastgesteld has occurred E902 is entered. After all the activities of E902 have been completed behind the scenes E902 is exited and the process continues. When returning from E902 one of two events might have occurred: either the fee was paid on time, leges betaald binnen termijn, or it was not, leges niet betaald binnen betaal termijn, termijn niet verlengd.

Successful payments lead to **E512** evalueren vergunning. Unsuccessful ones end with a link to buiten behandeling stellen. This contains verwerken uitgaande creditfacturen and verwerken uitgaande facturen which are therefore not visible. When all goes well the promise step ends with the event aanvraag volledig ingediend.

Link **E904** which coincides with node T904 can also be seen within E510.

Reflecting the Transaction Pattern in E510

The **request** step starts with the event vooroverleg aangeraagd and ends with the event vergunningaanvraag definitief ingediend or with the event vergunningaanvraag definitief ingediend na vooroverleg.

The **promise** step starts with the activity aanvraag scannen en koppelen aan zaak /dossier. It should be considered good form to end a process step with an event and start the next step with an activity as that more closely resembles ending a step with a factum and starting a next step with an act in the DEMO process structure diagram. Notice that both the promise and the decline are represented.

Execution starts after aanvraag volledig ingediend. Depending on what happened during execution it either ends with vergunningaanvraag 3.4 procedure afgehandeld, vergunningaanvraag 4.1 procedure afgehandeld, or with procesvoering afgehandeld. This event was added to account for situations where the 3.4 and 4.1 procedures were interrupted by protests. The actual process logic for these situations was still being developed by UPP and could not be added to this document.

No activities and events were found that represented the **state** step. This could indicate one of two things: either the statement was mistakenly left out, or no explicit statement is made by RWS. Whatever the case, the point is that it is now quite visible that the statement has not been modeled and this is an effective way to spot gaps in the process logic.

69

The **accept** step depends on operationeel voorbereiden handhaving which was outside of the scope of the selected research area. Its details are not essential for further explanation of the proposed modeling method. For more examples of proposed EPCs see appendix 12.3.

Note that each step of the transaction pattern is linked to actor roles and not to company positions. Company positions only come into focus at layer 6.

The Two Steps to Completing the Circle

To provide an overview the two steps to achieving stage 3 and 4 of the coupling are repeated here:

1. For the root process of the tree, create an EPC.
2. In the EPC of the process currently under consideration:
 - a. place process interface pairs to the EPCs of its sub-processes.
 - b. Create the EPCs of its sub-processes.
 - c. Repeat step 2 as needed.

7.5 Achieving Stage 5

Stage 5 of the coupling is achieved by assigning organizational units in terms of the actor roles they are authorized to play. The role that makes the request must be the role that accepts the result. The role that promises must be the role that executes and states. But the actual functionaristype that fulfils the role in each step may differ. This arrangement prevents roles from being assigned to a functionaristype that is convenient rather than appropriate.

The steps to achieving stage 5 are listed here:

1. Ensure that the role that makes the request is the role that accepts the result.
2. Ensure that the role that promises the result is the role that executes and states.
3. Ensure that the company position assigned to the roles is authorized to take on those roles.

7.6 Guidelines for Maintaining the Coupling

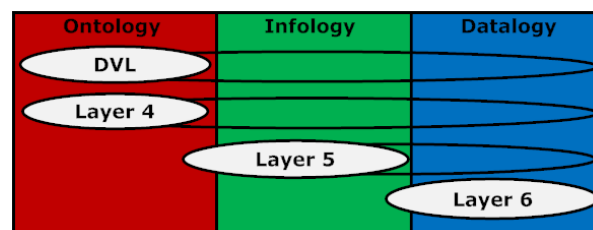
The steps in sections 7.3, 7.4 and 7.5 help the architect to create a consistent delineation and structuring of DVL models and accompanying UPP models. The following guidelines are meant to help the architect to maintain this consistency after it has been created. They are based on the four axioms and the organization theorem of DEMO. A listing of the guidelines in Dutch can be found in appendix 12.4.6.

The organization theorem was discussed in section 2.1 and is summarized here in a simplified form as: *an organization is the layered integration of business, intellect, and document*. This theorem should be combined with the distinction axiom to explain the need for separate layers when modeling the organization.

The distinction axiom was discussed in section 2.1 and is summarized here in a simplified form as: *data processing, information processing, and decision making are three distinct capabilities*.

The distinction axiom (and organization theorem) leads to the following four guidelines.

- DVL models should hold ontological transactions, decisions.
- UPP processes (layer 4) should hold ontological nodes, decisions.
- UPP process logic (layer 5) should primarily hold the infological activities that support the ontological nodes, the information processing on which decisions are based.
- UPP activity realizations should primarily hold the datalogical entities that support the infological activities, the documents and systems that facilitate information processing.



7-16 Distinction Guidelines

The composition axiom was discussed in section 2.1 and is summarized here in a simplified form as: *transactions can be composed of other transactions*.

The composition axiom leads to the following two guidelines.

- A DVL model should be accompanied by a UPP process model, and a transaction should be accompanied by a UPP sub-process node. Hereby externally and self-initiating transactions relate to branches that sprout from the root of the sub-process tree. Enclosed transactions relate to sub-process nodes that sprout from branches.
- UPP nodes at layer 4 should be accompanied by EPCs at layer 5, and a sub-node on layer 4 should be accompanied by a sub-EPC at layer 5.

The transaction axiom was discussed in section 2.1 and is summarized here in a simplified form as: *a transaction is a pattern in which someone is requesting and someone else is promising. After which the second person declares and the first person accepts. At least, when all goes well.*

The transaction axiom leads to the following guideline.

- UPP process logic at layer 5 should reflect the transaction pattern.

The operation axiom was discussed in section 2.1 and is summarized here in a simplified form as: *actor roles have responsibilities and execute transactions.*

The operation axiom leads to the following two guidelines.

- UPP process logic should be based on actor roles.
- UPP activity realizations at layer 6 should translate actor roles into company positions.

These guidelines are like a checklist that must be completed in order for the coupling to be considered consistent. For example if an EPC has been added to layer 5 and there is no node representing that EPC at layer 4, then there is inconsistency between layer 5 and layer 4. By regularly going over this list architects can find and fix inconsistencies that tend to sneak into the architectures over time.

7.7 Conclusion

In this chapter I have put forth a description of a consistent coupling between DVL and UPP. The coupling takes place in five stages and ensures consistency of the delineation and structure of DVL and UPP models.

I have also listed the immediate benefits that the architects and users of these architectures would receive if the coupling were realized. These benefits are: a new way to verify consistency, a new way to test completeness, prevention of doing unnecessary work, increased clarity of models, and a systematic way of assigning responsibilities. One must keep in mind that the listed benefits only scratch the surface of what is possible with the coupling in place; those who utilize the coupling will truly unleash its value.

This chapter also contains actionable steps for achieving the coupling. There are seven steps for achieving consistency between DVL and UPP layer 4. There are two additional steps for extending this to layer 5, and there are three additional steps that deal with UPP layer 6.

In order to demonstrate its capabilities, the coupling has been applied to the issuing permits area of RWS. The resultant DVL model was seen in section 7.3, and the resultant UPP models were seen in section 7.4, and in appendix 12.3.

The chapter is concluded with a set of guidelines that serve as a checklist for maintaining the coupling once it is in place.

This chapter has explained, in terms of modeling, what it means for DVL and UPP to be consistently coupled, how this coupling is beneficial, and how this coupling can be achieved and maintained.

Phase THREE – Assessing the Connection

8 Evaluation

In this section I discuss the results of the final phase of the research project, namely the evaluation of the proposed coupling by a group of people who are in the position to influence the course of the architectures. I introduce the GDSS used for the evaluation as well as the participants of the session. And finally I discuss the session and the results that were achieved.

8.1 The GDSS Session

The research results were evaluated in a session held on July 2nd 2008 using a Group Decision Support System (GDSS). The GDSS that was used consisted of a number of laptops running the GDS meeting support software MeetingWorks and was operated by Prof.dr.ing. Hans Mulder of VIAGroep^{xi}. The purpose of a GDSS is to gather, exchange, analyze, process, and evaluate information, in order to make decisions.

A GDSS has five important characteristics:

1. Anonymity: all input is anonymous which means that every idea is decoupled from the person who inputs it, and so every idea can be evaluated objectively by the group.
2. Parallel input and communication: participants can enter comments and ideas at the same time, drastically shortening the time required for everyone to have their say.
3. Group memory: all inputs are stored and can later be retrieved for processing.
4. Visual consensus: the degree of agreement and disagreement can be visualized in real-time.
5. Immediate reporting: the system can generate reports of the meeting immediately after the meeting has been concluded.

To best accommodate the participants all content was in Dutch and the results are presented in this document in Dutch as well. I do not translate the results into English because I do not wish to inadvertently change the intention that the participants were trying to convey.

The session facilitated the making of three results, namely 1) a valuation of the opportunities and barriers to coupling DVL and UPP, 2) a valuation of the guidelines by which the coupling is established, and 3) an inventory of further actions for implementing the coupling.

8.2 Participants of the GDSS Session

Ton Kroon [3]	Ton.Kroon@rws.nl
Aart Vermetten [1]	aart.vermetten@rws.nl
Jan Dietz [3]	j.l.g.dietz@tudelft.nl
Hans Spek [1]	hans.spek@rws.nl
Lemmen, MHLM [2]	martin.lemmen@rws.nl
Ben Knikman [1]	ben.knikman@rws.nl
John Hof [2]	john.hof@rws.nl
Xander de Gans [2]	xander.de.gans@rws.nl
Danielle Strijdhartig [1]	dstrijdhartig@rws.nl
Fred Louwen. [2]	fred.louwen@rws.nl
Martin Op 't Land [3]	martin.OptLand@gmail.com

[1] = EAC,

[2] = BPS,

[3] = external expert



8.3 Opportunities and Barriers

The session started out with a presentation on the work and the results that were achieved. This introduction gave the participants a good impression of possible benefits of the coupling. After the introduction the participants were asked to provide their views on the opportunities that the coupling would create and the barriers that will be faced. This part of the session was handled as a brainstorm step where the participants could enter as many views as they wanted into the GDSS. The system automatically gathered and displayed the views. For a listing of these views see appendix 12.4.1.

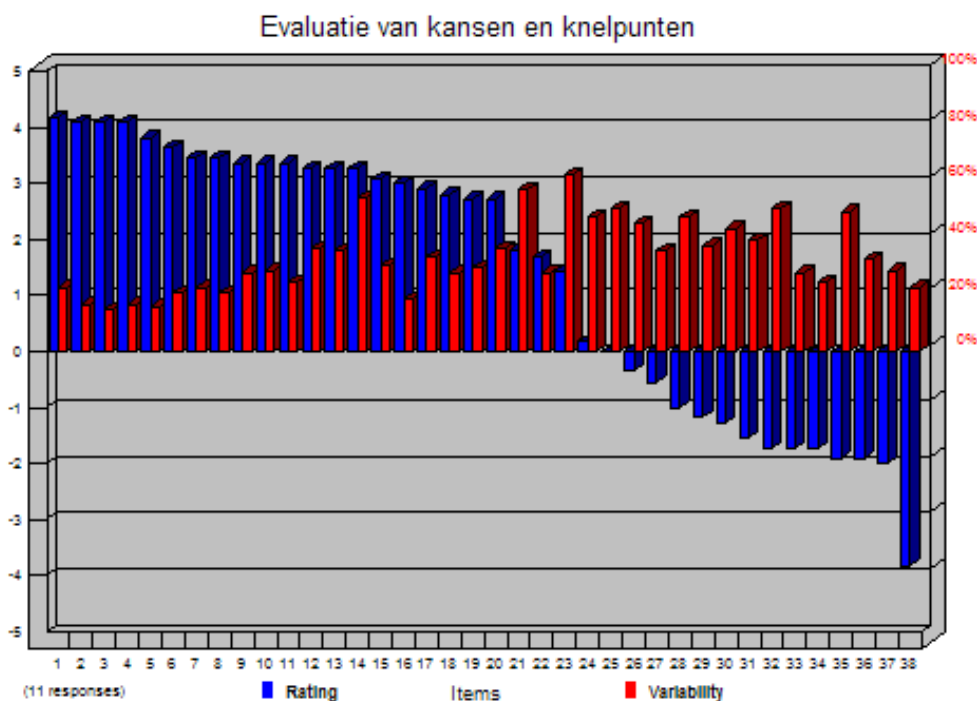
To eliminate duplicates, the views were organized into groups of related views that are visible in appendix 12.4.2. This step was handled by selecting the views one by one and discussing with the participants if they thought that the view was related to a previously discussed view. If yes, the view was added to the group of that previous view. If no, then the view received its own new group.

The next step was to find out how the participants perceived the presented views and to poll the amount of agreement among the participants. The participants were given the opportunity to rate the grouped views on a scale of -5 to 5, whereby negative values indicate a barrier and positive values indicate an opportunity. The idea behind this is that one persons barrier may be another persons opportunity. Thus -5 is a serious barrier, 0 is neither a barrier nor an opportunity and 5 is a big opportunity. Participants were allowed to abstain when they did not have an opinion.

74

Figure 8-1 shows an overview of the results of the rating. The blue bars above the horizon show where views are considered opportunities and those below the horizon show where views are considered barriers. The red bars show the variability of the values that were given to each view.

Figure 8-1 Rating of the Views



The first view is considered a strong opportunity and there is little variability in the values. This means that the participants largely agree that the first view is a good opportunity. The last view is clearly considered a barrier and again there is little variability. So the participants agree that the last view is indeed a barrier. But view 25 is neither a barrier nor an opportunity. The red bar shows that there is much variability and so much disagreement. The conclusion must be that some participants considered it an opportunity while others considered it a barrier. The views are presented in table Table 8-1 along with the average of the ratings and the variability. For the exact values that were entered by the participants see appendix 12.4.3.

Table 8-1 Average Rating and Variability of the Views

# View	Average Rating	Variability
1. Het uniformeren van begrippen / dezelfde taal spreken binnen RWS, w.o. tussen CD en DID	4.2	23%
2. door de samenhang in beeld te brengen wordt ook duidelijker hoe belangrijk de verschillende onderdelen zijn (vb: gegevens op orde)	4.1	17%
3. impact van wijzigingen binnen bestaande processen eenvoudiger te vertalen naar consequenties (vanuit interne verandering)	4.1	15%
4. Verbetering van de kwaliteit en consistentie van zowel DVL als processen	4.1	17%
5. duidelijkheid tot welke dienstverlening een proces bijdraagt	3.8	16%
6. impact van nieuwe ontwikkelingen zijn eenvoudiger te vertalen naar (1) documentgerelateerde en (2) ICT-gerelateerde consequenties	3.6	21%
7. impact van nieuwe ontwikkelingen zijn eenvoudiger te vertalen naar organisatie consequenties (bijv. nieuwe wetgeving)	3.5	23%
8. Mogelijkheid om maatschappelijk vitale processen te kwalificeren op basis van dienstverleningen en de processen die ze ondersteunen (hogere eisen aan bepaalde processen)	3.5	21%
9. Voor integrale besluitvorming biedt het een kans om op basis van ontwikkelingen te prioriteren	3.4	28%
10. applicaties die niet aan een proces gekoppeld kunnen worden zichtbaar maken	3.4	29%
11. Door de koppeling worden de verantwoordelijkheden in UPP-modellen helder.	3.4	25%
12. Duidelijker verband tussen "missie" (de opdracht) en de feitelijke uitvoering	3.3	37%
13. Door mijn proces door een DEMO-bril te bekijken, kan ik het proces logischer, dus begrijpbaarder maken.	3.3	36%
14. Kans: t.b.v applicatieconsolidatie: applicaties makkelijker inventariseren door gebruik te maken van de koppeling tussen transacties in DVL en de applicaties & systemen op niveau 6 van UPP.	3.3	55%
15. Kans: Het management begrijpt beter wat zijn	3.1	31%

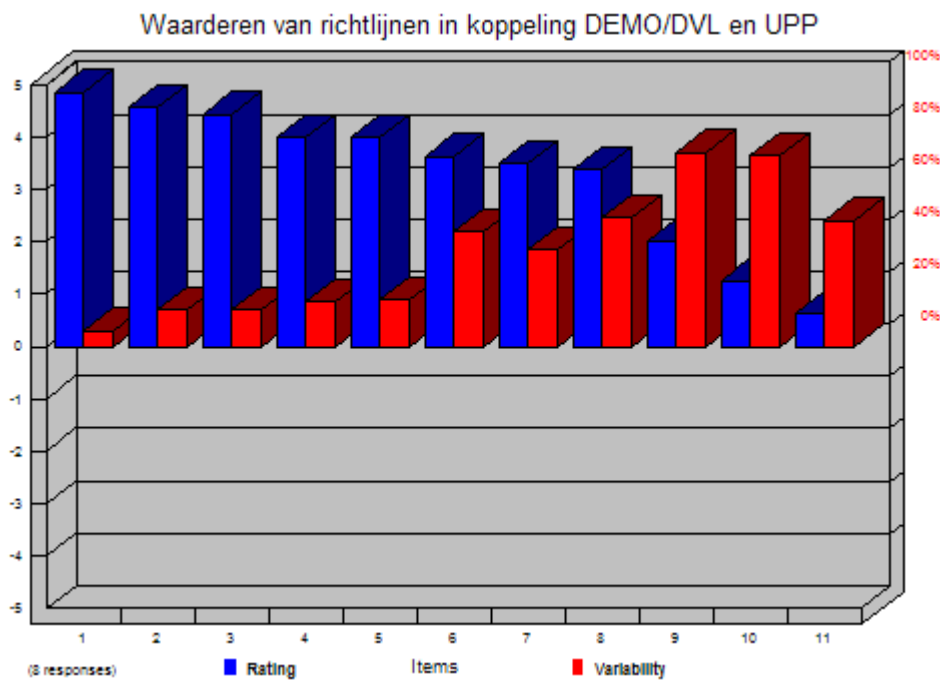
personeel doet.		
16. Het ondersteunen van de koppeling tussen 'procesbeheer' (traditioneel BPS werk) en informatiemanagement / functioneel beheer (nieuw BPS werk)	3.0	19%
17. Kans: werknemers zullen niet alleen hun eigen proces begrijpen maar ook de context van het proces binnen de organisatie en kunnen meer initiatief nemen.	2.9	34%
18. stapsgewijze besluitvorming over functionaristypen en organisatirollen faciliteren	2.8	28%
19. besluitvorming binnen directieteam RWS wordt integraler en daardoor eenduidiger	2.7	30%
20. scherper duidelijk krijgen welke informatiebehoefte achter de UPP-raadplegingen van documenten, input en processor zitten	2.7	37%
21. Een ideale combinatie van DVL en UPP zou m.i. zijn dat EPC's worden toegepast voor het in detail beschrijven van een processtap, ofwel voor de actieregels in DEMO. Dan is er geen overlapping en kunnen er dus geen inconsistenties ontstaan.	1.8	58%
22. koppeling ondersteunt de afbakening (BPS-DID) tussen 'vraag' / functionaliteit en 'aanbod' / applicaties; wie gaat waarover	1.7	28%
23. het streven naar een 1 op 1 vertaling van transacties naar functies	1.4	63%
24. koppelen vraagt altijd om nauwe samenwerking en afstemming	0.2	48%
25. medewerkers voelen zich nog meer gebonden aan vaste werkwijze, niet alleen proces, maar ook het daarvoor te gebruiken systeem	0.0	51%
26. Er is in principe geen een-op-een verband tussen transacties in DVL en functies en actoren en organisatorische eenheden in UPP.	-0.3	46%
27. Koppeling vertroebelt mogelijk het onderscheid tussen vraagfunctie (IV, LD'en) en aanbodfunctie (ICT, DID)	-0.5	36%
28. de structuur van de UPP processen moet worden aangepast	-1.0	48%
29. UPP is "onderworpen" aan V&W-conventies, deze kunnen strijdig zijn met uitgangspunten DEMO	-1.2	38%
30. Knelpunt: het is nodig de (afbakening en indeling van de) huidige modellen te wijzigen.	-1.3	44%
31. Knelpunt: er is DEMO training nodig voor procesbeheerders	-1.5	40%
32. De kosten van de aanpassing van de DVL-modellen en de UPP-modellen	-1.7	51%
33. Het kan overkomen alsof het beheer van DEMO-modellen en UPP-modellen dubbel werk is	-1.7	28%
34. Doelstelling van verschillende modellen kan verwarren	-1.7	25%

35. Gescheiden beheer van DVL en Processen; noodzaak tot directe koppeling/gezamenlijke tooling (of regelmatige maar redundante bijwerking)	-1.9	50%
36. UPP opgezet obv PDCA-cyclus; dit kan in combinatie met co"ordinatie-patroon tot "opblazen" van het proces leiden	-1.9	33%
37. het detail niveau van koppeling wordt te groot (bijv. van een applicatie, gaat dat tot niveau MS-Word?)	-2.0	29%
38. Is hier een opdrachtgever van hogerhand voor te vinden (binnen het procesdenken)	-3.8	23%

8.4 Technical Guidelines

After evaluating the opportunities and barriers the guidelines of the coupling were presented. During the presentation there were opportunities for participants to rate the guidelines. The result of this process is visible in figure Figure 8-2. The blue bars are all above the horizon which means that on average the guidelines were well-liked. The red bars show the variability of the values that were given to each guideline.

Figure 8-2 Rating of the Guidelines



The conclusion that can be drawn from figure Figure 8-2 is that acceptability of guidelines 1 to 5 is high. Guidelines 6, 7 and 8 are generally acceptable but there are some participants who are critical of them. Guidelines 9, 10 and 11 are disputed significantly despite their overall acceptability. The guidelines are listed in tableTable 8-2 along with the average rating and the variability. For the exact values that were entered by the participants see appendix 12.4.6.

Table 8-2 Rating and Variability on Guidelines

# Guideline	Average Rating	Variability
1. Bij een DVL, hoort een functieboom en bij een transactie hoort een functie. Hierbij starten extern geïnitieerde en zelf initiërende transacties nieuwe takken vanuit de boomwortel. Ingesloten transacties starten sub-functies in de bestaande takken.	4.9	6%
2. Bij een functie hoort een EPC, en bij een sub-functie hoort een sub-EPC.	4.6	14%
3. Een UPP proces is hangt af van actor rollen.	4.4	14%
4. Een DVL model bevat ontologische transacties.	4.0	17%
5. In een UPP FAD worden rollen omgezet naar functionaristypen.	4.0	18%
6. Een DVL model is op gelijke wijze afgebakend als de bijbehorende UPP cluster.	3.6	44%
7. Een UPP functieboom (niveau 4) bevat de bijbehorende ontologische functies.	3.5	37%
8. Een DVL model is op gelijke wijze gestructureerd als de bijbehorende UPP cluster.	3.4	49%
9. Een UPP proces wordt ingericht volgens het standaard transactiepatroon.	2.0	74%
10. Een UPP FAD (niveau 6) bevat de datalogische ondersteuning van de infologische activiteiten.	1.3	73%
11. Een UPP proces (niveau 5) bevat de infologische ondersteuning van de ontologische functies.	0.6	48%

8.5 Actions for the Realization

Finally the participants were asked to determine the actions that they thought would be necessary to realize the coupling. It was a brainstorm session so some actions may contradict others. The actions are listed here:

- Introduceren koppeling voorlopig alleen bij nieuwe modellen
- Introduceren koppeling bij bestaande toetsbare modellen
- Gemeenschappelijke training van EAC en BPS
- Jaarlijks handmatig synchroniseren van modellen
- Koppelen Troux en ARIS systemen
- Afstemmen met V&W qua conventies
- Introduceren koppeling bij modellen, die op dit moment belangrijk zijn voor RWS (b.v. prestatiebestekken, samenwerking Droog VM van RWS met prov NB, etc.)
- Uitwerken aansprekend voorbeeld en presenteren op Landelijke Netwerkdag oktober/november dit jaar
- Inventarisatie modellen EAC/BPS
- Definiëren extra views op DVL voor specifieke doelgroepen
- Vaststellen nieuwe conventies
- Synchroniseren resultaten UPP-processen met diensten DVL's (korte termijn)
- Resultaten van deze sessie uitdragen binnen EAC en BPS
- Consensus op hoog niveau voor realisatie koppeling en uitvoer van de gevolgen
- Nagaan noodzaak van infologische en datalogische commentaren op UPP, gezien de UPP-doelgroep
- Samenvoegen scripties Danielle en Ashna; voorwoord BPS EN EAC toevoegen
- Verder onderzoeken doorwerkingen informatieaspecten (bv feitenbanken in DVL) naar infologische aspecten op niveaus 5

79



8-3 End of the Session

9 Discussion and Recommendations

This section is used to place the remarks that have not found a place in the previous sections and the recommendations for further steps.

9.1 Motivation for Coupling

First I would like to provide some remarks on why I believe the coupling should be accomplished in the first place. Thus why do we not merge the two architectures completely or why do we not leave them completely separate or in the extreme case abandon one in favor of the other.

I will address the last case first: the two architectures cater to different audiences and they do this to such an extent that neither one would be suitable to replace the other in its entirety.

That said, the motivation for keeping two separate architectures about the same organization rather than completely merging them into one architecture is to allow specialization. Certain user groups will need certain kinds of information represented in a certain way while other user groups will need different kinds of information or the same kinds of information but represented differently. Allowing specialized architectures ensures that each group of users gets the information they need without compromise.

The motivation for coupling architectures rather than leaving them as separate entities is the ability to update them simultaneously. If a change takes place in the organization and it is only visible in one architecture than the persons using the other architecture may make decisions based on an outdated view of the organization. It is important to keep architectures consistent with each other so that all persons see the same organization at all times, even if they see it from a slightly different angle. It is thus important to remember that the architectures are different views on the same organization and for the members of the organization to act with one mind towards one vision these architectures must be consistent with each other at all times.

9.2 Recommendations

The first recommendation is to maintain the momentum generated during the project. The benefits for the organization should be communicated to higher management in order to secure long term support for coupling the architectures.

The second recommendation is to follow up on the actions presented in section 8.5. and create a task force consisting of those who are knowledgeable of the architectures and have the authority to change them. This task force should asses the range of projects that could benefit from the coupling. It should then determine how deep the coupling should be implemented to support those projects. Thus, should the coupling be achieved through conventions between people, or through interconnecting systems and applications. And based on these insights the task force should start to mobilize those parties that are needed to realize the coupling.

What has been done for two architectures can also be done for three. There should be follow up projects that seek to develop the same kind of coupling between DVL, UPP and the RIS/MARNIS architecture. Only when all three architectures are consistent will V&W be able to act with one mind towards one vision.

10 Conclusion

The first sub-question that was asked in the introduction regards the current state of coupling between DVL and UPP. The first phase of research was geared towards answering that question. Because the architectures were based on DEMO and ARIS the first step was to read books and papers about those two topics. This gave a good insight in what their purpose is and how they achieve this purpose.

DEMO is very much about getting at the ontology of organizations and transferring this knowledge to decision makers. ARIS is an architecture solution for the management of business processes.

That said, DVL is used to support decision makers in carrying out large scale organizational changes, and UPP is used to support the uniform execution of processes within the existing organizational situation.

DVL is an implementation of the DEMO construction model, and UPP is an implementation of the ARIS views. There are four DEMO models and five ARIS views and there is some overlap between them. The DEMO construction model provides the same information as the ARIS organization and function views. Of course due to the difference in purpose of DEMO and ARIS the information is presented differently in both. The DEMO process and action models are wrapped up in UPP's control view. DEMO's state model and ARIS' data view both have the intention of sharing with their audience the data types that are important to the organization. This is extent of coupling on the level of model types.

Looking within the models to the object types that the models contain some more overlap can be spotted. Functions vs. acts, events vs. facts. But there is also some things that they do not have in common such as DEMO's focus on what things are and how they are related to each other vs. ARIS' focus on how things are used within the organization.

Testing my assumptions by substituting object types of ARIS into DEMO models and vice versa was one of the most valuable steps in my search for the coupling. It showed which information was missing in the models on the other side and which information was present.

But the true test came when I compared an actual DVL model with an actual UPP model. Because these models were real world examples they had complexities that textbook examples lacked. In the real world there is never a single right answer for a problem. And this comparison showed that sometimes transactions in DVL were matched by sections of EPC in UPP. Other times a single activity in UPP could match an entire transaction in DVL. Some parts of DVL were not represented in UPP and there were more of such inconsistencies.

Basically, we know that the architectures describe aspects of the same organization and that there should be a systematic way to move from one architecture into the next and back again but the current way of modeling does not support that kind of transparency.

The second sub-question regards the desired state of coupling. The desired state should allow architects to present information in a way that is most effective for their user group but at the same time consistent with the other architectures of the organization. Based on the exploration done in the first phase of research a five-stage coupling was developed that

would accomplish this goal. The coupling ensures consistency of delineation and structure of the two architectures, and has the benefit of allowing their users to move freely from one architecture to the other and see both views of the organization quickly and easily. Users no longer need to start initiatives to gather needed information that happens to be outside of the scope of their own architecture. They can simply relate the content of the other architecture to that of their own and focus on their business tasks. This capability is likely to unleash a multitude of new opportunities for V&W.

The third question was about how the coupling could be achieved. I was able to answer this question in modeling terms; there are steps that architects can take to make their architectures consistent with each other, and there is a set of guidelines to help them maintain that consistency. However the managerial side of this question was touched upon only lightly in the GDSS evaluation. Participants were allowed to say what they thought would be needed to accomplish the coupling in an organizational setting. And this is now the most important part of further action. To get support from higher management and to keep the momentum that was built up during the project.

To answer the problem thesis, which is:

How can DVL and UPP models be coupled in a consistent way?

Consistency, as it relates to DVL and UPP, entails the proper delineation and structuring of models, reflecting the transaction pattern and assigning responsibilities in accordance with actor roles. This thesis provides a description of the consistent state in tangible terms. By keeping this state in mind, architects can align their goals with each other and start working together towards consistency.

Since DVL and UPP are currently not in a consistent state of coupling, their models need to be re-delineated and restructured to arrive at that consistent state. This thesis provides a set of actionable steps with which the consistent state can be achieved. By applying these steps the architects of UPP and DVL can bring their models into the consistent state.

Once consistency has been achieved the architects must keep the architectures in this state. The thesis provides a set of guidelines to aid them in that respect.

Thus, the way in which DVL and UPP models can be consistently coupled is by envisioning the coupling, growing support for the coupling, applying the steps to achieve the coupling, and following the guidelines to maintain the consistency of the coupling.

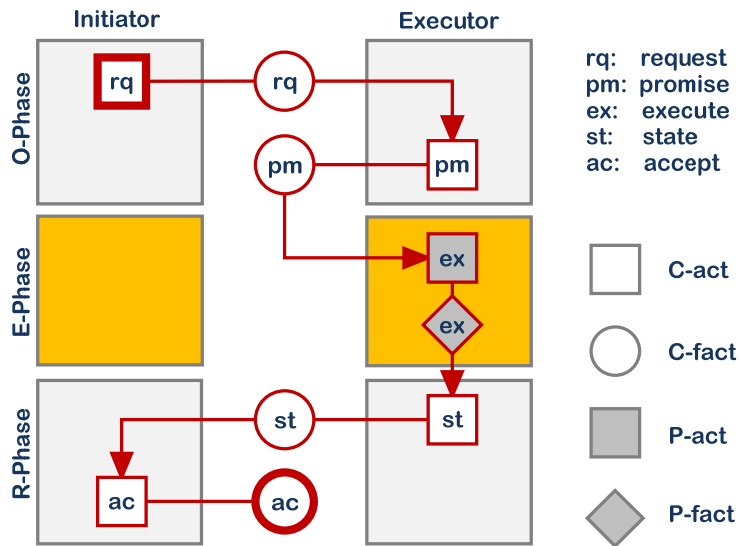
11 Abbreviations

AM	- Action Model
ABD	- Actor Bank Diagram
ARIS	- ARchitecture of Integrated Information Systems
ARS	- Action Rule Specification
ATD	- Actor Transaction Diagram
BCT	- Bank Contents Table
C-act	- Coordination act
C-fact	- Coordination fact
CM	- Construction Model
CPB	- Composite Production Bank
DEMO	- Design and Engineering Methodology for Organizations
ECA	- Event-Condition-Action
EPC	- Event-Process Chain
GDSS	- Group Decision Support System
IAM	- InterAction Model
ISM	- InterStriction Model
IUT	- Information Use Table
P-act	- Production act
P-fact	- Production fact
PM	- Process Model
PSD	- Process Structure Diagram
OCD	- Organization Construction Diagram
OFD	- Object Fact Diagram
OPL	- Object Property List
SM	- State Model
TRT	- Transaction Result Table
XOR	- exclusive OR

12 Appendices

12.1 DEMO Transaction Pattern

12-1 The Basic Transaction Pattern

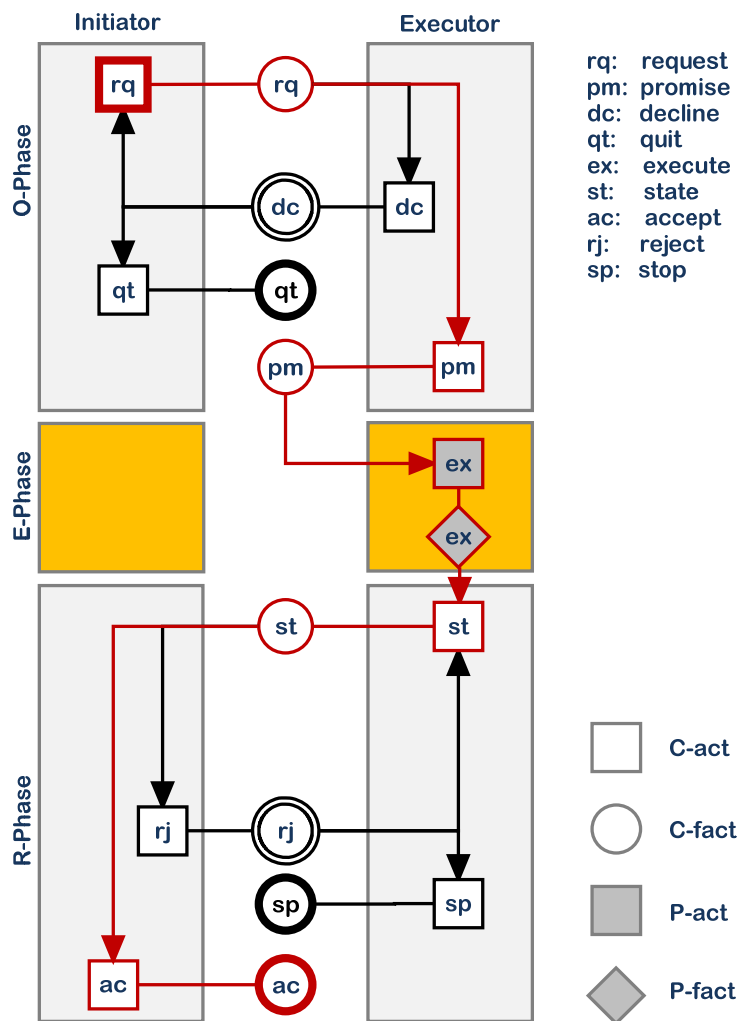


84

The diagram in figure 12.1 depicts that a transaction always occurs between two actors, the initiator and the executor. Every transaction occurs in three phases, namely the Order phase or O-phase, the Execution phase or E-phase and the Result phase or R-phase.

In the order phase the initiator requests something and the executor promises to deliver the requested item or service. In the execution phase the executor produces that which was agreed upon. In the result phase the executor states that the work has been completed and offers the result to the initiator. The initiator accepts the result and the transaction is completed. The basic transaction pattern is the most concise representation of the transaction pattern. The standard and universal transaction patterns show the additional paths that be taken during a transaction.

12-2 The Standard Transaction Pattern

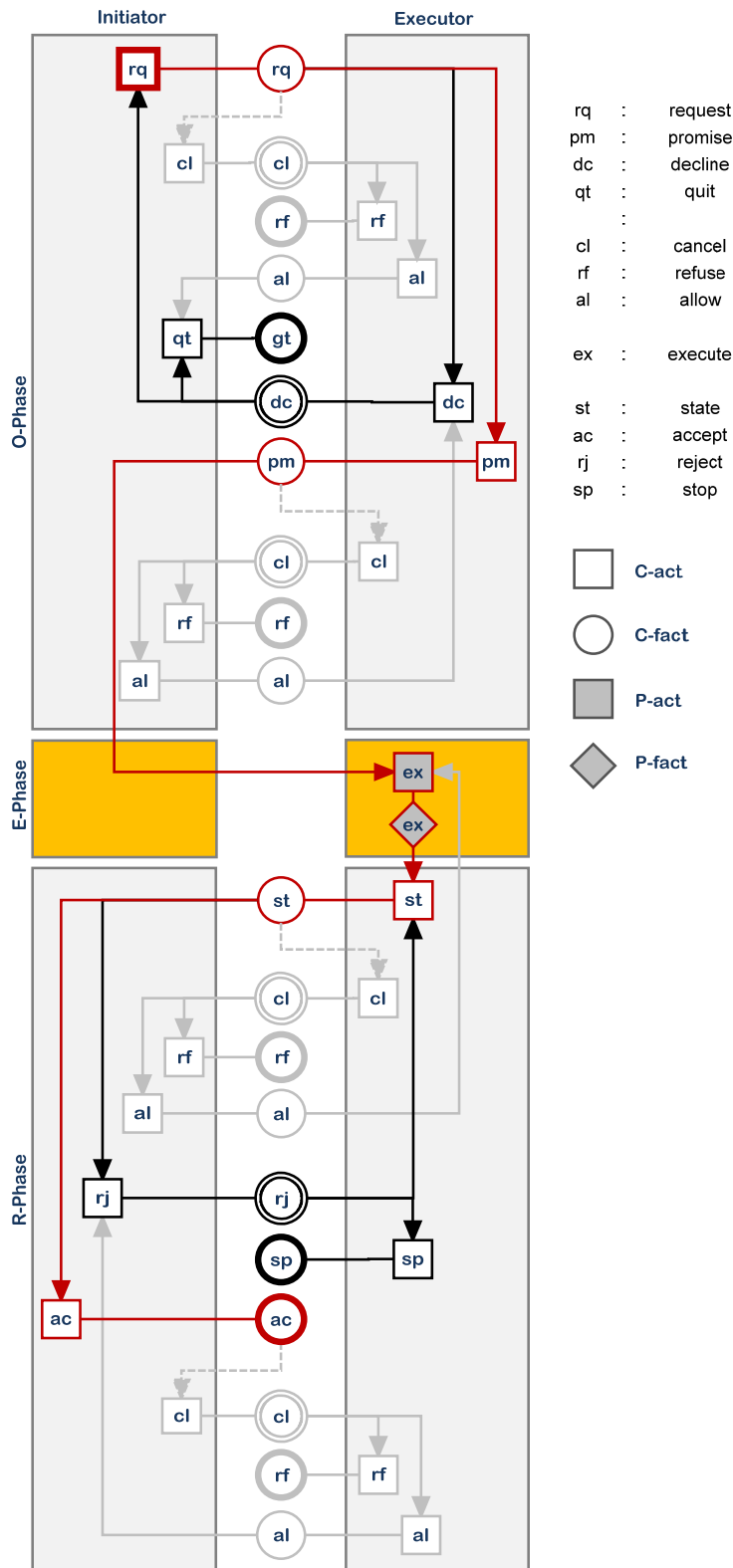


85

As in figure 12-1, the diagram above depicts that a transaction always occurs between two actors, the initiator and the executor. Every transaction occurs in three phases, namely the Order phase or O-phase, the Execution phase or E-phase and the Result phase or R-phase.

In the order phase the initiator requests something and the executor either promises or declines to deliver the requested item or service. Upon declining, the initiator may either quit the transaction or request something different. In the execution phase the executor produces that which was agreed upon. In the result phase the executor states that the work has been completed and offers the result to the initiator. The initiator either accepts or rejects the result. If the result is rejected the executor may either re-do the work or stand firm and stop the transaction.

12-3 The Complete Transaction Pattern



86

As in figure 12-1, the diagram above depicts that a transaction always occurs between two actors, the initiator and the executor. Every transaction occurs in three phases, namely the Order phase or O-phase, the Execution phase or E-phase and the Result phase or R-phase.

In the order phase the initiator requests something and the executor either promises or declines to deliver the requested item or service. Upon declining, the initiator may either quit the transaction or request something different. The initiator can cancel the request at any moment before the promise or decline. The executor can cancel the promise at any point before execution. In the execution phase the executor produces that which was agreed upon. In the result phase the executor states that the work has been completed and offers the result to the initiator. This statement can be cancelled for instance if the executor discovers that there is a problem with the batch of products that is about to be delivered. The initiator either accepts or rejects the result. If the result is rejected the executor may either re-do the work or stand firm and stop the transaction. But even after the transaction has been accepted, the initiator can feel buyers remorse and return the product and so cancel the accept.

12.2 The Pizzeria Case

The pizzeria belongs to a chain Domino. The pizza selection as well as the prices are determined by the central office. One can choose out of 12 kinds of pizza in three sizes: 25 cm Classic Medium, 25 cm Pan Pizza Medium, and 35 cm Classic Large. The selection is printed on colorful flyers, which are given to customers and dropped at locations where one may expect gatherings of young people.

Giovanni manages about ten youngsters on a part-time basis. Their function is either baker or deliverer. Baking is not much of a job anymore; the pizzas are produced in automatic ovens. Both the bakers and the deliverers take orders from customers, of which over 100% are now by telephone. There is a fully automated order entry system that is integrated with other administrative functions. The address data of customers are recorded in the system, so that they can easily be retrieved if the customer's telephone number is entered.

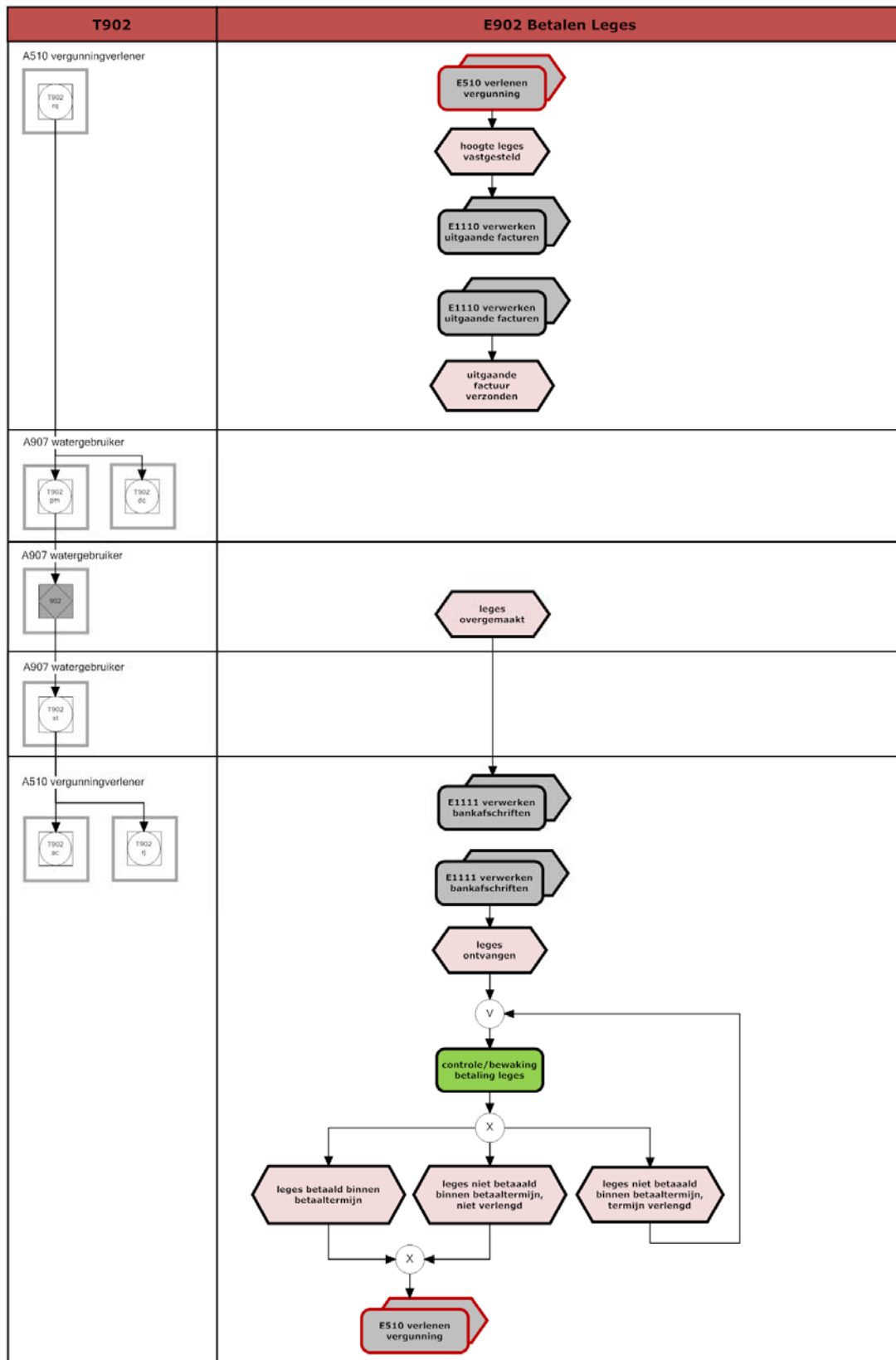
After entering an order, stickers are automatically printed, which an employee sticks on empty boxes. These boxes are then put in a row on a rack. On the sticker are printed the ordered pizza, the order number, the total price, the name of the customer, and the delivery address.

As soon as a pizza is discharged by the baking machine, one of the bakers puts it in the right box, which he or she takes from the rack. The baker then enters a record of the pizza being baked in the automatic order system (there is a PC next to the baking machine). The boxes of the same order are kept together on a large table with infrared lamps hanging above.

If an order is complete, this is signaled on the PC for the deliverers. One of them then takes the boxes to the customer address. On return, he or she puts the received money in the pink box, which has a slot for every deliverer.

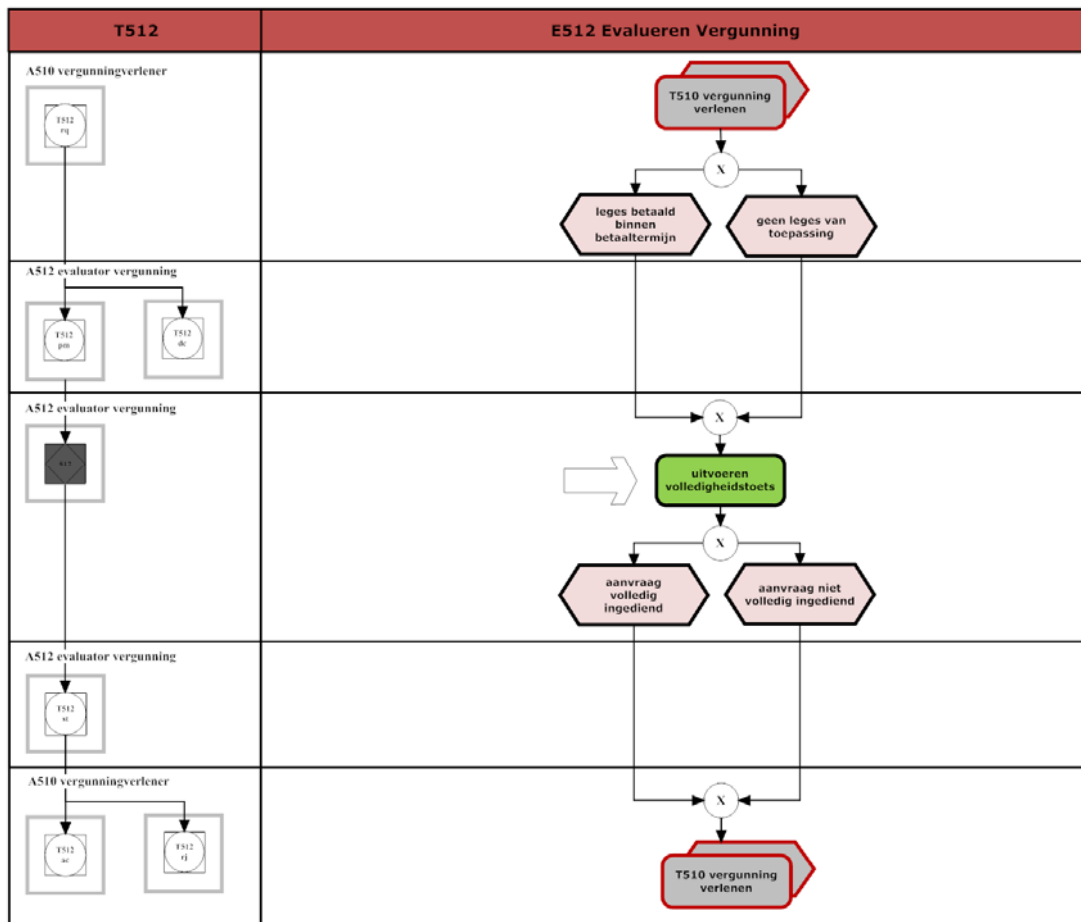
12.3 Proposed UPP models

Figure 12-4 E902 Betalen Leges



E902 betalen leges is reached via E510 verlenen vergunning. It wraps the infological EPCs E1110 verwerken uitgaande facturen and E1111 verwerken bankafschriften into a single ontological EPC. By hiding the infological EPCs within an ontological one, layer 4 and the DVL can avoid infological nodes and transactions. This limits the complexity of maintaining the coupling. The ontological transaction will always be betalen leges but the infological activities that make it possible may change. Keeping fully infological EPCs on a “second tier” allows those changes to happen without affecting layer 4 and DVL. This means less overhead in maintaining consistency between the architectures and thus more freedom to update it. This does not forbid the adding of infological nodes in layer 4 and in DVL, it merely presents a transparent way to avoid it.

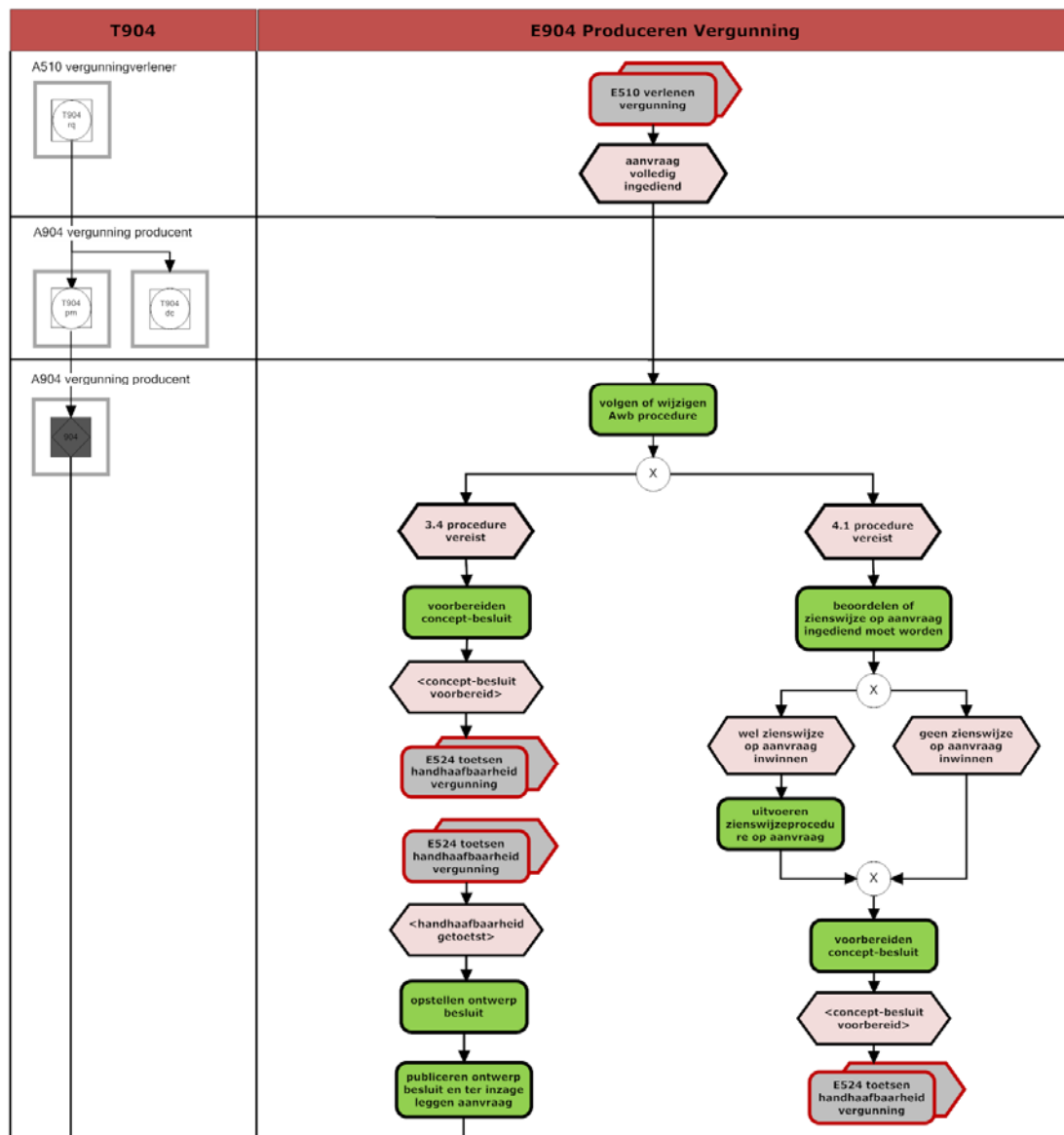
Figure 12-5 E512 Evalueren Vergunning



91

EPC 512 evalueren vergunning is reached via E510 verlenen vergunning. It was included to demonstrate the result that is achieved when the ontologize step of the coupling is performed poorly. Assuming that T512 is a valid ontological transaction then node T512 must be worked out into a proper EPC. However since T512 correlates with a single activity the resulting EPC looks as figure Figure 12-5. It is merely a “skeleton”; it opens with a process interface from E510 verlenen vergunning, followed by the two events that could have triggered it, followed by the single activity, its resultant events and finally a closing process interface back to E510. Processes such as these are an indicator of poor execution of the ontologize step and should encourage the architect to reevaluate the coupling at layer 4. If the coupling is correct and this is the achieved result, one must conclude that a significant proportion of process logic is missing. Next steps are to speak with the users of the process to develop the missing process logic.

Figure 12-1 E904 Produceren Vergunning I

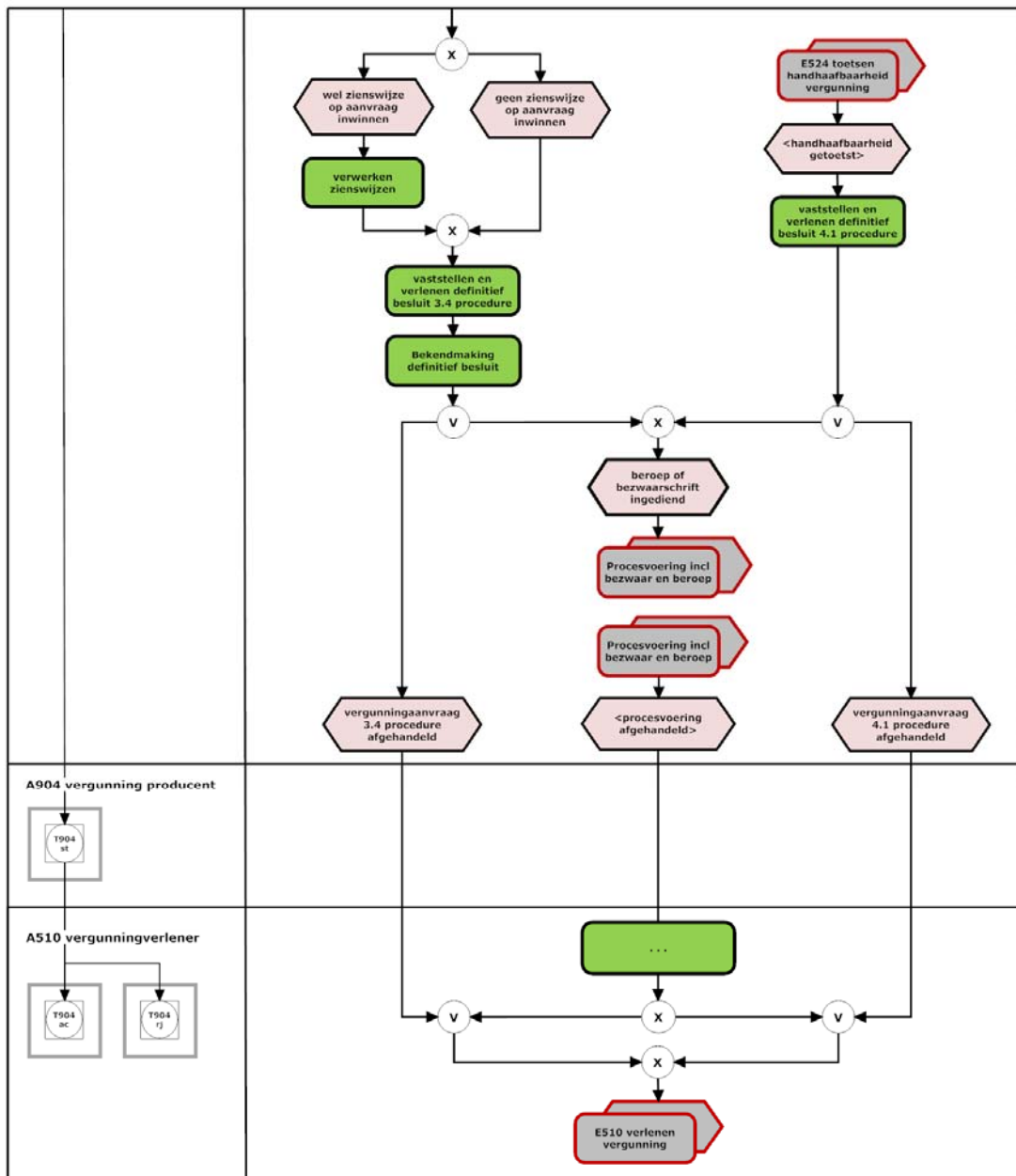


92

E904 produceren vergunning is also reached via E510 verlenen vergunning. In it, the 3.4 and 4.1 procedures are still intact. But as with E902 these infological EPCs are hidden within the ontological EPC Produceren Vergunning. In fact they are now simply two branches within E904. It is possible to place the procedures into their own EPCs and link to them from within E904 (as was done in E902), but one must ask if there is any value in doing that. The way to answer that is to ask if there is a separate actor role who performs the procedures. If there is not, there is also no viable reason to place them into separate EPCs. In the case of E904, juxtaposing the transaction pattern reveals that the requesting and promising party are the same role. Whenever this is the case there is no reason to make separate EPCs. A good place for a separate EPC is at E524 toetsen handhaafbaarheid vergunning. The DVL clearly shows it being performed by an external actor role.

The emptiness of the promise/decline region attracts the architect's attention. Does the vergunning producent really accept every request?

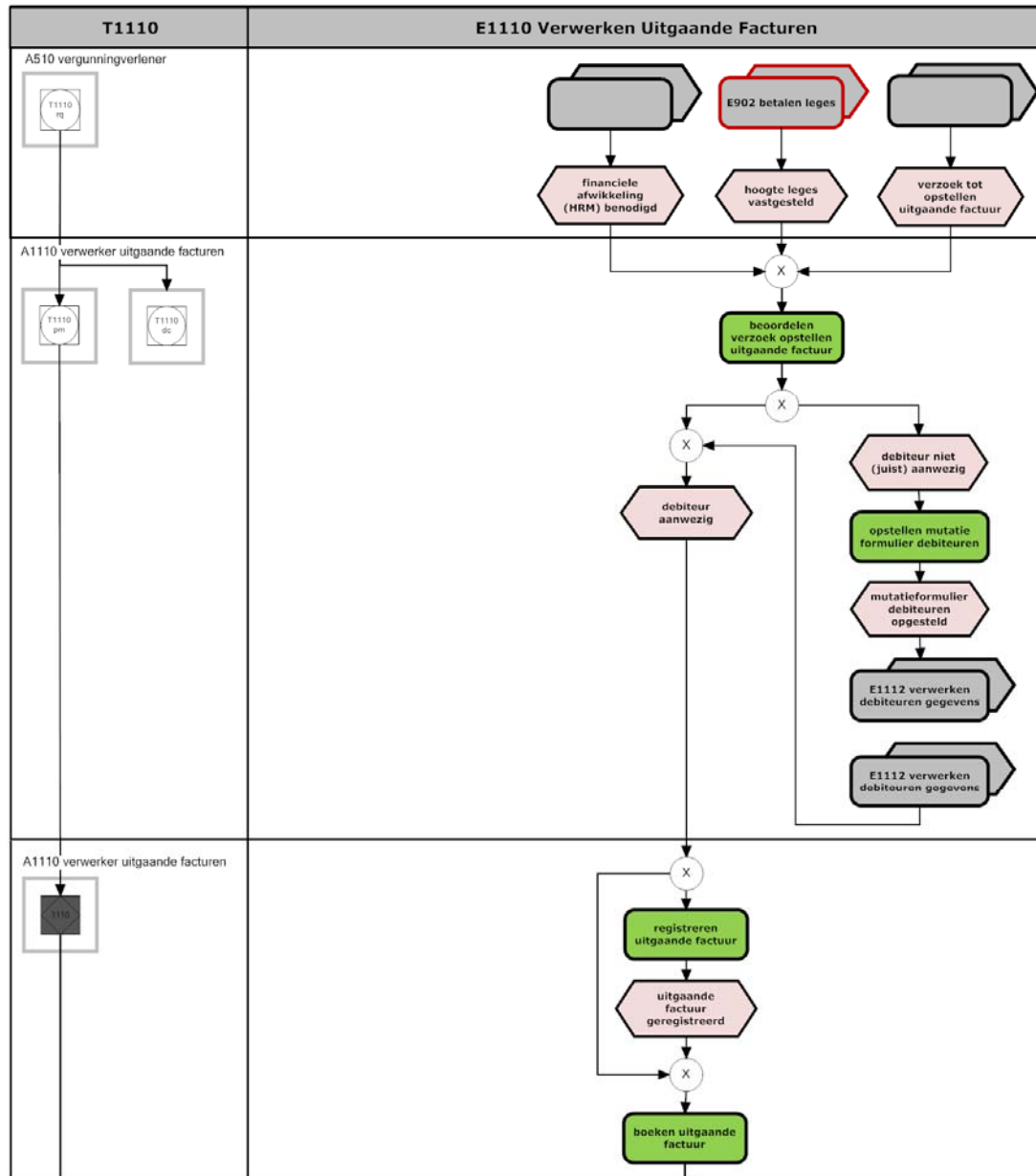
Figure 12-2 E904 Produceren Vergunning II



93

The lower part of the diagram shows that whether the 3.4 or the 4.1 procedure is followed, at some point beroep of bezwaar is possible. There is no explicit statement. E904 ends with vergunning aanvraag 3.4 procedure afgehandeld, or with vergunning aanvraag 4.1 procedure afgehandeld, or with a combination of one of them with procesvoering afgehandeld. In order to properly route the process logic an additional activity is required. For the rules of modeling with EPCs see the choices and synchronizations in section 3.4.

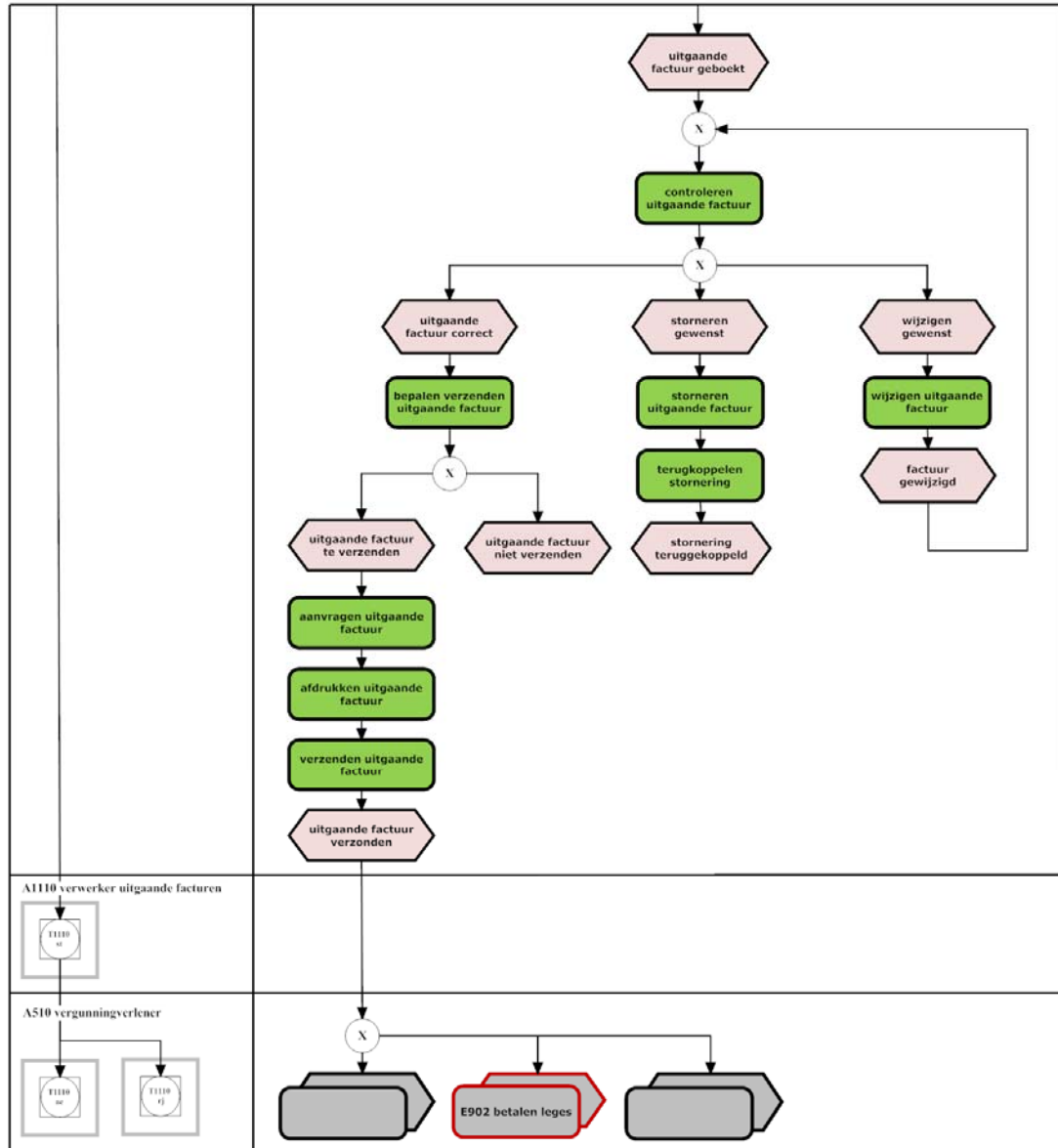
Figure 12-3 E1110 Verwerken Uitgaande Facturen I



94

This EPC shows what happens after E1110 verwerken uitgaande facturen is entered from within E902 betalen leges. Multiple EPCs lead to E1110, but E902 leads here after hoogte leges vastgesteld. The promise/decline step is interesting because it seems as though both possibilities are represented. First beoordelen verzoek opstellen uitgaande factuur is performed. In the case that the debiteur is known a promise follows immediately. In the other case the debiteur information must first be entered into the system by another party. Only then can the promise follow. Because the requestor is not contacted in anyway, this should not be considered a decline. It should be considered a call to a second infological transaction. One on which the promise step must wait before it may complete.

Figure 12-4 E1110 Verwerken Uitgaande Facturen II



95

After the execution there is no state step and the EPC ends by re-entering E902. Of course, if E1110 had been entered from a different EPC control would return to *that* EPC.

12.4 GDSS Evaluation Output

12.4.1 Listing of Opportunities and Barriers

Opportunities

- t.b.v applicatieconsolidatie: applicaties makkelijker inventariseren door gebruik te maken van de koppeling tussen transacties in DVL en de applicaties & systemen op niveau 6 van UPP.
- werknemers zullen niet alleen hun eigen proces begrijpen maar ook de context van het proces binnen de organisatie en kunnen meer initiatief nemen.
- duidelijkheid tot welke dienstverlening een proces bijdraagt
- impact van nieuwe ontwikkelingen zijn eenvoudiger te vertalen naar organisatie consequenties
- Mogelijkheid om maatschappelijk vitale processen te kwalificeren op basis van dienstverleningen en de processen die ze ondersteunen
- Het management begrijpt beter wat zijn personeel doet.
- Het ondersteunen van de koppeling tussen 'procesbeheer' (traditioneel BPS werk) en informatiemanagement / functioneel beheer (nieuw BPS werk)
- impact van wijzigingen binnen bestaande processen eenvoudiger te vertalen naar consequenties
- inzichtelijk maken welke applicaties er voor welke transacties zijn.
- Het uniformeren van begrippen / dezelfde taal spreken tussen CD en DID
- Door mijn proces door een DEMO-bril te bekijken, kan ik het proces logischer, dus begrijpbaarder maken.
- impact van nieuwe ontwikkelingen zijn eenvoudiger te vertalen naar (1) documentgerelateerde en (2) ICT-gerelateerde consequenties
- 1. Duidelijker verband tussen "missie" (de opdracht) en de feitelijke uitvoering
- besluitvorming wordt integraler en daardoor eenduidiger
- Verbetering van de kwaliteit en consistentie van zowel DVL als processen
- Er is in principe geen een-op-een verband tussen actoren in DVL en functies in UPP. Dat kan worden opgelost d.m.v. een actor-functie-matrix.
- Relatie tussen applicaties en processen wordt ontsloten.
- applicaties die niet aan een proces gekoppeld kunnen worden zichtbaar maken
- Eventuele hiaten zullen opvallen.
- het biedt een kans om op basis van ontwikkelingen te prioriteren
- herkenbaarheid van proces-resultaten uit UPP wordt sterker
- Compleetheit van de processen is goed duidelijk
- door de samenhang in beeld te brengen wordt ook duidelijker hoe belangrijk de verschillende onderdelen zijn (vb: gegevens op orde)
- Een ideale combinatie van DVL en UPP zou m.i. zijn dat EPCs worden toegepast voor het in detail beschrijven van de details in een processtap, ofwel voor de actieregels in DEMO. Dan is er

geen overlapping en kunnen er dus geen inconsistenties ontstaan.

- koppeling ondersteunt de afbakening tussen 'vraag' / functionaliteit en 'aanbod' / applicaties
- Zowel de DVL-modellen als de UPP-modellen worden verbeterd als voorstel wordt gevolgd.
- Door de koppeling worden de verantwoordelijkheden in UPP-modellen helder.
- scherper duidelijk krijgen welke informatiebehoefte achter de UPP-raadplegingen van documenten, input en processor zitten
- stapsgewijze besluitvorming over functionaristypen en organisatierollen faciliteren

Barriers

- het is nodig de (afbakening en indeling van de) huidige modellen te wijzigen.
- er is DEMO training nodig voor procesbeheerders
- Het kan overkomen alsof het beheren van DEMO-modellen en UPP-modellen dubbel werk is
- Ondersteunt de tooling de koppeling wel?
- Wat is de afbakening van verantwoordelijkheden / wie is eigenaar van samengestelde procesmodellen?
- Is hier een opdrachtgever van hogerhand voor te vinden
- UPP is "onderworpen" aan V&W-conventies, deze kunnen strijdig zijn met uitgangspunten DEMO
- medewerkers voelen zich nog meer gebonden aan vaste werkwijze, niet alleen proces, maar ook het daarvoor te gebruiken systeem
- Wat gaat de aanpassing van de DVL-modellen en de UPP-modellen kosten?
- Gescheiden beheer van DVL en Processen; noodzaak tot directe koppeling/gezamenlijke tooling (of regelmatige maar redundante bijwerking)
- koppelen vraagt altijd om nauwe samenwerking en afstemming
- Er is blijvende afstemming nodig tussen beide modellen
- UPP opgezet obv PDCA-cyclus; dit kan in combinatie met coördinatie-patroon tot "opblazen" van het proces leiden
- de structuur van de UPP processen moet worden aangepast
- tot op welk detail niveau koppel je een applicatie, gaat dat tot niveau word?
- Capaciteit aan EAC zijde om DVL's te completeren
- Koppeling vertroebelt mogelijk het onderscheid tussen vraagfunctie (IV, LD'en) en aanbodfunctie (ICT, DID)
- wie heeft de regie
- een koppeling kan leiden tot stammenstrijd
- Toegevoegde waarde van koppeling tot op het diepste niveau nog onvoldoende helder
- Doelstelling van verschillende modellen verwacht

12.4.2 Grouping of Opportunities and Barriers

- t.b.v applicatieconsolidatie: applicaties makkelijker inventariseren door gebruik te maken van de koppeling tussen transacties in DVL en de applicaties & systemen op niveau 6 van UPP.
 - inzichtelijk maken welke applicaties er voor welke transacties zijn.
 - Relatie tussen applicaties en processen wordt ontsloten.
- werknemers zullen niet alleen hun eigen proces begrijpen maar ook de context van het proces binnen de organisatie en kunnen meer initiatief nemen.
- duidelijkheid tot welke dienstverlening een proces bijdraagt
- impact van nieuwe ontwikkelingen zijn eenvoudiger te vertalen naar organisatie consequenties (bijv. nieuwe wetgeving)
- Mogelijkheid om maatschappelijk vitale processen te kwalificeren op basis van dienstverleningen en de processen die ze ondersteunen (hogere eisen aan bepaalde processen)
- Kans: Het management begrijpt beter wat zijn personeel doet.
- Het ondersteunen van de koppeling tussen 'procesbeheer' (traditioneel BPS werk) en informatiemanagement / functioneel beheer (nieuw BPS werk)
- impact van wijzigingen binnen bestaande processen eenvoudiger te vertalen naar consequenties (vanuit interne verandering)
- Het uniformeren van begrippen / dezelfde taal spreken binnen RWS, w.o. tussen CD en DID
- Door mijn proces door een DEMO-bril te bekijken, kan ik het proces logischer, dus begrijpbaarder maken.
- impact van nieuwe ontwikkelingen zijn eenvoudiger te vertalen naar (1) documentgerelateerde en (2) ICT-gerelateerde consequenties
- Duidelijker verband tussen "missie" (de opdracht) en de feitelijke uitvoering
 - herkenbaarheid van proces-resultaten uit UPP wordt sterker
- besluitvorming binnen directieteam RWS wordt integraler en daardoor eenduidiger
- Verbetering van de kwaliteit en consistentie van zowel DVL als processen
 - Zowel de DVL-modellen als de UPP-modellen worden verbeterd als voorstel wordt gevolgd.
 - Eventuele hiaten zullen opvallen.
 - Compleetheid van de processen is goed duidelijk
- Er is in principe geen een-op-een verband tussen transacties in DVL en functies en actoren en organisatorische eenheden in UPP.
- het streven naar een 1 op 1 vertaling van transacties naar functies
- applicaties die niet aan een proces gekoppeld kunnen worden zichtbaar maken
- Voor integrale besluitvorming biedt het een kans om op basis van ontwikkelingen te prioriteren
- door de samenhang in beeld te brengen wordt ook duidelijker hoe belangrijk de verschillende onderdelen zijn (vb: gegevens op orde)

- Een ideale combinatie van DVL en UPP zou m.i. zijn dat EPCs worden toegepast voor het in detail beschrijven van een processtap, ofwel voor de actieregels in DEMO. Dan is er geen overlapping en kunnen er dus geen inconsistenties ontstaan.
- koppeling ondersteunt de afbakening (BPS-DID) tussen 'vraag' / functionaliteit en 'aanbod' / applicaties; wie gaat waarover
 - Wat is de afbakening van verantwoordelijkheden / wie is eigenaar van samengestelde procesmodellen?
- Door de koppeling worden de verantwoordelijkheden in UPP-modellen helder.
- scherper duidelijk krijgen welke informatiebehoefte achter de UPP-raadplegingen van documenten, input en processor zitten
- stapsgewijze besluitvorming over functionaristypen en organisatirollen faciliteren
- Knelpunt: het is nodig de (afbakening en indeling van de) huidige modellen te wijzigen.
- Knelpunt: er is DEMO training nodig voor procesbeheerders
- Het kan overkomen alsof het beheren van DEMO-modellen en UPP-modellen dubbel werk is
- Is hier een opdrachtgever van hogerhand voor te vinden (binnen het procesdenken)
 - wie heeft de regie
 - een koppeling kan leiden tot stammenstrijd
- UPP is "onderworpen" aan V&W-conventies, deze kunnen strijdig zijn met uitgangspunten DEMO
- medewerkers voelen zich nog meer gebonden aan vaste werkwijze, niet alleen proces, maar ook het daarvoor te gebruiken systeem
- De kosten van de aanpassing van de DVL-modellen en de UPP-modellen
 - Capaciteit aan EAC zijde om DVL's te completeren
- Gescheiden beheer van DVL en Processen; noodzaak tot directe koppeling/gezamenlijke tooling (of regelmatige maar redundante bijwerking)
 - Ondersteunt de tooling de koppeling wel?
- koppelen vraagt altijd om nauwe samenwerking en afstemming
 - Er is blijvende afstemming nodig tussen beide modellen
- UPP opgezet obv PDCA-cyclus; dit kan in combinatie met co"ordinatiepatroon tot "opblazen" van het proces leiden
- de structuur van de UPP processen moet worden aangepast
- het detail niveau van koppeling wordt te groot (bijv. van een applicatie, gaat dat tot niveau MS-Word?)
- Koppeling vertroebelt mogelijk het onderscheid tussen vraagfunctie (IV, LD'en) en aanbodfunctie (ICT, DID)
- Doelstelling van verschillende modellen kan verwarren
 - Toegevoegde waarde van koppeling tot op het diepste niveau nog onvoldoende helder

12.4.3 Detailed Rating of Opportunities and Barriers

The tables below show the scores that the participants of the GDSS evaluation session gave to the (grouped) views. In these tables yellow columns contain scores by EAC, cyan columns contain scores by BPS, and blue columns contain scores by the external experts. Hereby GEM is the group's average (gemiddelde) and STD is the groups standard deviation. To read comments from the participants regarding their reasoning for the scores they have given see appendix 12.4.4.


12-6 Detailed Rating of Opportunities and Barriers

Opportunities and Barriers	1A	1A	1A	1A	GEM	STD	2A	2A	2A	2B	GEM	STD	3A	3A	3A	GEM	STD
1 Het uniformeren van begrippen / dezelfde taal spreken binnen RWS, w.o. tussen CD en DID	5	5	2	4	4.00	1.41	3	2	5	5	3.75	1.50	5	5	5	5.00	0.00
2 door de samenhang in beeld te brengen wordt ook duidelijker hoe belangrijk de verschillende onderdelen zijn (vb: gegevens op orde)	4	5	3	3	3.75	0.96	5	4	5	5	4.75	0.50	3	3	5	3.67	1.15
3 impact van wijzigingen binnen bestaande processen eenvoudiger te vertalen naar consequenties (vanuit interne verandering)	4	5	2	4	3.75	1.26	4	4	5	4	4.25	0.50	4	4	5	4.33	0.58
4 Verbetering van de kwaliteit en consistentie van zowel DVL als processen	4	5	4	5	4.50	0.58	2	4	4	3	3.25	0.96	4	5	5	4.67	0.58
5 duidelijkheid tot welke dienstverlening een proces bijdraagt	3	3	4	4	3.50	0.58	5	3	5	3	4.00	1.15	4	3	5	4.00	1.00
6 impact van nieuwe ontwikkelingen zijn eenvoudiger te vertalen naar (1) documentgerelateerde en (2) ICT-gerelateerde consequenties	3	5	2	2	3.00	1.41	5	3	5	4	4.25	0.96	4	4	3	3.67	0.58
7 impact van nieuwe ontwikkelingen zijn eenvoudiger te vertalen naar organisatie consequenties (bijv. nieuwe wetgeving)	3	4	2	2	2.75	0.96	4	3	5	5	4.25	0.96	2	3	5	3.33	1.53
8 Mogelijkheid om maatschappelijk vitale processen te kwalificeren op basis van dienstverleningen en de processen die ze ondersteunen (hogere eisen aan bepaalde processen)	4	2	4	2	3.00	1.15	4	4	2	5	3.75	1.26	3	3	5	3.67	1.15
9 Voor integrale besluitvorming biedt het een kans om op basis van ontwikkelingen te prioriteren	3	5	1	2	2.75	1.71	3	4	5	4	4.00	0.82	1	5	4	3.33	2.08
10 applicaties die niet aan een proces gekoppeld kunnen worden zichtbaar maken	2	5	4	4	3.75	1.26	5	1	5	2	3.25	2.06	1	4	4	3.00	1.73

100

11 Door de koppeling worden de verantwoordelijkheden in UPP-modellen helder.	3	5	2	5	3.75	1.50	2	3	5	2	3.00	1.41	2	3	5	3.33	1.53
12 Duidelijker verband tussen "missie" (de opdracht) en de feitelijke uitvoering	3	5	3	0	2.75	2.06	5	2	5	5	4.25	1.50	0	3	5	2.67	2.52
13 Door mijn proces door een DEMO-bril te bekijken, kan ik het proces logischer, dus begrijpbaarder maken.	4	5	1	5	3.75	1.89	2	5	0	4	2.75	2.22	1	4	5	3.33	2.08
14 Kans: t.b.v applicatieconsolidatie: applicaties makkelijker inventariseren door gebruik te maken van de koppeling tussen transacties in DVL en de applicaties & systemen op niveau 6 van UPP.	5	5	4	5	4.75	0.50	4	-5	4	2	1.25	4.27	4	3	5	4.00	1.00
15 Kans: Het management begrijpt beter wat zijn personeel doet.	3	5	1	2	2.75	1.71	3	5	2	5	3.75	1.50	1	2	5	2.67	2.08
16 Het ondersteunen van de koppeling tussen 'procesbeheer' (traditioneel BPS werk) en informatiemanagement / functioneel beheer (nieuw BPS werk)	3	4	2	1	2.50	1.29	3	3	3	4	3.25	0.50	2	4	4	3.33	1.15
17 Kans: werknemers zullen niet alleen hun eigen proces begrijpen maar ook de context van het proces binnen de organisatie en kunnen meer initiatief nemen.	3	3	3	0	2.25	1.50	2	5	0	4	2.75	2.22	2	5	5	4.00	1.73
18 stapsgewijze besluitvorming over functionaristypen en organisatirollen faciliteren	3	5	1		3.00	2.00	2	2	3	1	2.00	0.82	2	4	5	3.67	1.53
19 besluitvorming binnen directieteam RWS wordt integraler en daardoor eenduidiger	2	4	1	2	2.25	1.26	3	2	4	5	3.50	1.29	0	2	5	2.33	2.52
20 scherper duidelijk krijgen welke informatiebehoefte achter de UPP-raadplegingen van documenten, input en processor zitten	4	3	2	2	2.75	0.96	2	-2	5	2	1.75	2.87	3	4	5	4.00	1.00

21	Een ideale combinatie van DVL en UPP zou m.i. zijn dat EPCs worden toegepast voor het in detail beschrijven van een processtap, ofwel voor de actieregels in DEMO. Dan is er geen overlapping en kunnen er dus geen inconsistenties ontstaan.	4	5	-2	5	3.00	3.37	2	3	-3	0	0.50	2.65	-2	3	5	2.00	3.61
22	koppeling ondersteunt de afbakening (BPS-DID) tussen 'vraag' / functionaliteit en 'aanbod' / applicaties; wie gaat waarover	1	5	1	2	2.25	1.89	3	2		2	2.33	0.58	1	0	0	0.33	0.58
23	het streven naar een 1 op 1 vertaling van transacties naar functies	3	5			4.00	1.41	2	4	-3	-1	0.50	3.11	2	-4	5	1.00	4.58
24	koppelen vraagt altijd om nauwe samenwerking en afstemming	-2	0	2	-2	-0.50	1.91	-1	4	5	-2	1.50	3.51	1	-2	-1	-0.67	1.53
25	medewerkers voelen zich nog meer gebonden aan vaste werkwijze, niet alleen proces, maar ook het daarvoor te gebruiken systeem	-2	5	0	-2	0.25	3.30	-1	4	-2	0	0.25	2.63	-4	2	0	-0.67	3.06
26	Er is in principe geen een-op-een verband tussen transacties in DVL en functies en actoren en organisatorische eenheden in UPP.	-2	3	-1		0.00	2.65	2	3	-3	0	0.50	2.65	-2		-3	-2.50	0.71
27	Koppeling vertroebelt mogelijk het onderscheid tussen vraagfunctie (IV, LD'en) en aanbodfunctie (ICT, DID)	0	0	0	0	0.00	0.00	0	-5	-2	-1	-2.00	2.16	-1	3	0	0.67	2.08
28	de structuur van de UPP processen moet worden aangepast	0	-3	-2	-2	-1.75	1.26	0	-5	-3	0	-2.00	2.45	-2	2	4	1.33	3.06
29	UPP is "onderworpen" aan V&W-conventies, deze kunnen strijdig zijn met uitgangspunten DEMO	0	3	-1	-1	0.25	1.89	0	-5	-2	-2	-2.25	2.06	-2	0	-3	-1.67	1.53
30	Knelpunt: het is nodig de (afbakening en indeling van de) huidige modellen te wijzigen.	-1	-5	-2	-3	-2.75	1.71	-1	2	-3	0	-0.50	2.08	-3	3	-1	-0.33	3.06
31	Knelpunt: er is DEMO training nodig voor procesbeheerders	-1	-5	-2	-3	-2.75	1.71	-1	-5	0	0	-1.50	2.38	-1	2	-1	0.00	1.73
32	De kosten van de aanpassing van de DVL-modellen en de UPP-modellen	-1	-5	-2	-5	-3.25	2.06	-1	3	-3	-5	-1.50	3.42	-2	2	0	0.00	2.00
33	Het kan overkomen alsof het beheren van DEMO-modellen en UPP-modellen dubbel werk is	-2	-2	0	-5	-2.25	2.06	0	-2	0	-1	-0.75	0.96	-3	-2	-2	-2.33	0.58
34	Doelstelling van verschillende modellen kan verwarren	-2	-2	0	-2	-1.50	1.00	-1	-4	-2	-3	-2.50	1.29	-3	0	0	-1.00	1.73
35	Gescheiden beheer van DVL en Processen; noodzaak tot directe koppeling/gezamenlijke tooling (of regelmatige maar redundante bijwerking)	-3	-4	-2	-5	-3.50	1.29	-2	-2	5	0	0.25	3.30	-3	-2	-3	-2.67	0.58
36	UPP opgezet obv PDCA-cyclus; dit kan in combinatie met coördinatie-patroon tot "opblazen" van het proces leiden	-3	-1	-1	-2	-1.75	0.96	0	-5	-5	-2	-3.00	2.45	-1	0	-1	-0.67	0.58
37	het detail niveau van koppeling wordt te groot (bijv. van een applicatie, gaat dat tot niveau MS-Word?)	-2	0	-1	-2	-1.25	0.96	-2	-4	-5	-3	-3.50	1.29	-2	0	-1	-1.00	1.00
38	Is hier een opdrachtgever van hogerhand voor te vinden (binnen het procesdenken)	-3	-3	-4	-5	-3.75	0.96	-3	-5	-5	-5	-4.50	1.00	-2	-2	-5	-3.00	1.73

aandachtspunten 

The areas marked in red are areas of interest (AoI). Having the proper experience Prof.dr.ing. Hans Mulder, the GDSS facilitator, carefully selected these AoIs as follows: the scores were grouped by organization, thus EAC, BPS, and external. Then the average scores of the views were calculated per organization group. So the averages of EAC, BPS and of the externals. Then the averages were compared to each other in order to find AoIs. Some of the AoIs are briefly discussed below. I do not make any assumptions about the causes of the AoIs. The causes should be discovered through discussions between EAC and BPS and perhaps the external experts. I only offer a brief explanation of how to interpret the AoIs by giving some examples.

AoI Scores EAC / BPS / External

View 7: 2.75 / 4.25 / 3.33

“Impact van nieuwe ontwikkelingen zijn eenvoudiger te vertalen naar organisatie consequenties (bijv. Nieuwe wetgeving).”

On average all groups thought this view was a good opportunity but BPS seems to find it better than EAC and the experts.

View 12: **2.75 / 4.25 / 2.67**

“Duidelijk verband tussen missie (de opdracht) en de feitelijke uitvoering.”

As with view 7 BPS has marked view 12 higher than EAC and the external experts.

View 14: **4.75 / 1.25 / 4.00**

“Kans: t.b.v applicatieconsolidatie: applicaties makkelijker inventariseren door gebruik te maken van de koppeling tussen transacties in DVL en de applicaties & systemen op niveau 6 van UPP.”

Both the external experts and EAC considered view 14 a strong opportunity. BPS considered it a weak opportunity.

View 20: **2.75 / 1.75 / 4.00**

“Scherper duidelijk krijgen welke informatiebehoefte achter de UPP-raadplegingen van documenten, input en processor zitten.”

The external experts saw a strong opportunity which was not seen by EAC and BPS.

12.4.4 Participant Comments on Opportunities and Barriers

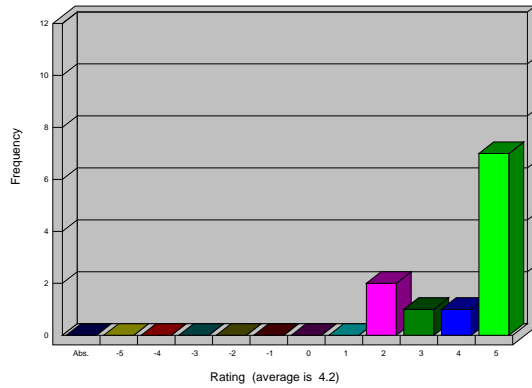
Bold statements are opportunities or barriers, the number value is the score given to the statement by the commenting participant, and next to the score is the comment.

- 1 **Mogelijkheid om maatschappelijk vitale processen te kwalificeren op basis van dienstverleningen en de processen die ze ondersteunen (hogere eisen aan bepaalde processen)**
 - 3.0 mits bij dienstverleningen ook de Quality of Business (QoB) wordt gedocumenteerd
- 2 **koppeling ondersteunt de afbakening (BPS-DID) tussen 'vraag' / functionaliteit en 'aanbod' / applicaties; wie gaat waarover**
 - 0.0 het gesprek tussen BPS en DID is absoluut een kans. EAC heeft een corporate rol (zowel vraag- als aanbodzijde / dixit CFO).
- 3 **medewerkers voelen zich nog meer gebonden aan vaste werkwijze, niet alleen proces, maar ook het daarvoor te gebruiken systeem**
 - 2.0 dat zou juist reden kunnen zijn of UPP-procesbeschrijvingen te versimpelen en geen dingen te laten voorschrijven die niet resultaat-/QoB-gericht zijn. Zie de exercitie "productdefinitie" van de Martins Lemmen en Op 't Land
- 4 **Koppeling vertroebelt mogelijk het onderscheid tussen vraagfunctie (IV, LD'en) en aanbodfunctie (ICT, DID)**
 - 3.0 juist een kans om dit onderscheid te verhelderen! Overigens is DID ook een LD ...
- 5 **UPP is "onderworpen" aan V&W-conventies, deze kunnen strijdig zijn met uitgangspunten DEMO**
 - 0.0 qua wijze van documenteren wel, qua achterliggende denkwijze (abstractie en transactiepatroon axioma's etc) verwacht ik geen groot probleem
- 6 **Knelpunt: er is DEMO training nodig voor procesbeheerders**
 - 2.0 heeft kwaliteitsverhoging tot gevolg
- 7 **De kosten van de aanpassing van de DVL-modellen en de UPP-modellen**
 - 2.0 door de gebieden op een slimme manier te kiezen o.b.v. RWS-prioriteiten kunnen deze kosten telkens gekoppeld worden aan rendement
- 8 **UPP opgezet obv PDCA-cyclus; dit kan in combinatie met co"ordinatie-patroon tot "opblazen" van het proces leiden**
 - 0.0 hoeft niet; er ligt theorie ('the Art of Mgt', Marcel Nieuwenhuis) over koppeling PDCA en DEMO-transactiepatroon

12.4.5 Dispersion diagrams for Valuation of Opportunities and Barriers

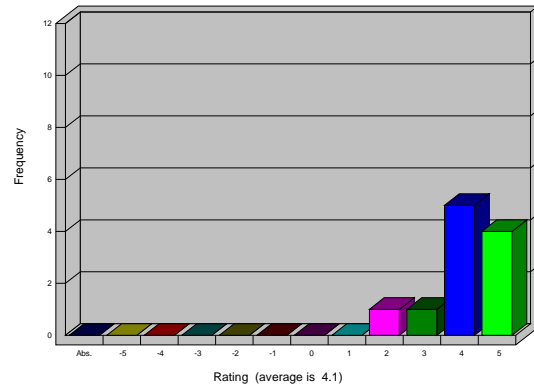
Each view has a dispersion diagram associated with it. The dispersion diagrams show the number of participants that rated the view a particular value. For example, in the first diagram two participants have voted 2, one voted 3, another participant voted 4, and seven participants considered the view worthy of 5 points.

Het uniformeren van begrippen / dezelfde taal spreken binnen R...



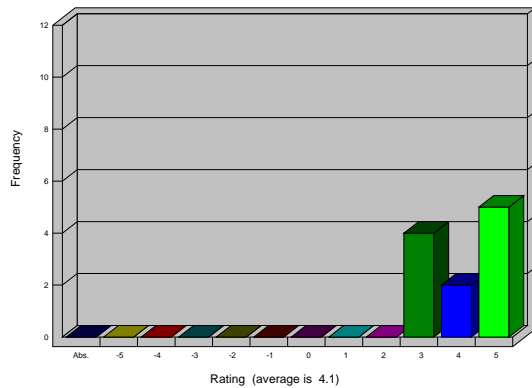
(11 responses)

Verbetering van de kwaliteit en consistentie van zowel DVL als...



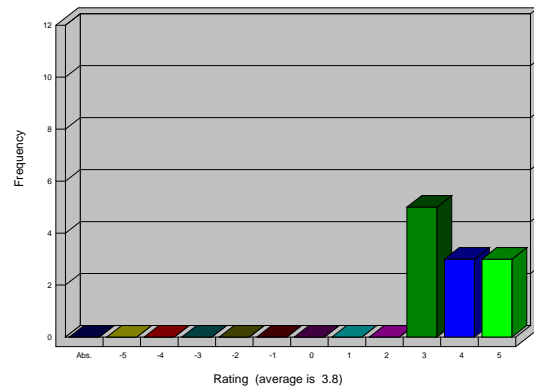
(11 responses)

door de samenhang in beeld te brengen wordt ook duidelijker ho...



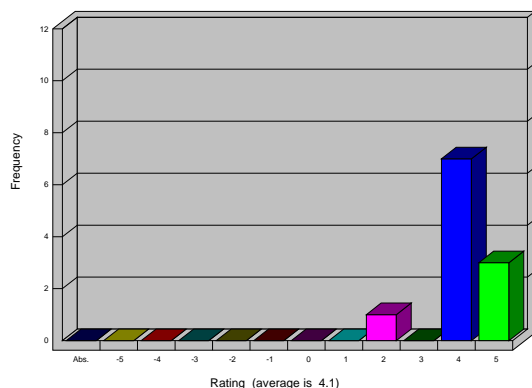
(11 responses)

duidelijkheid tot welke dienstverlening een proces bijdraagt



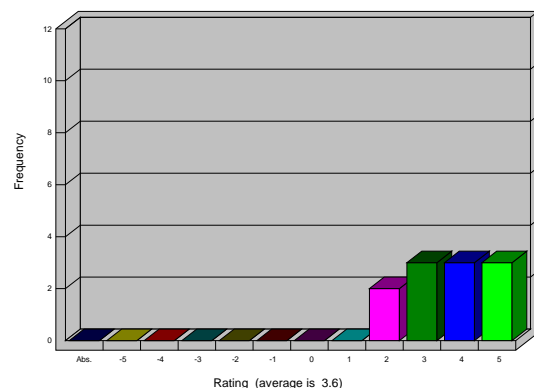
(11 responses)

impact van wijzigingen binnen bestaande processen eenvoudiger ...



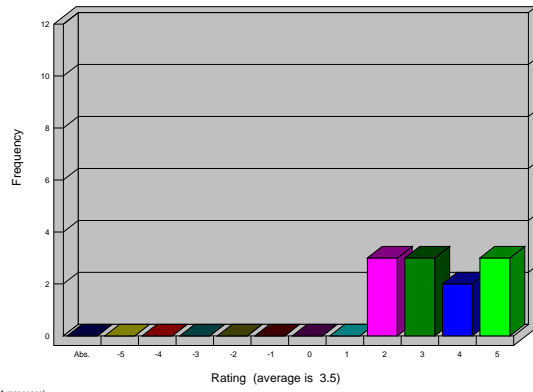
(11 responses)

impact van nieuwe ontwikkelingen zijn eenvoudiger te vertalen ...



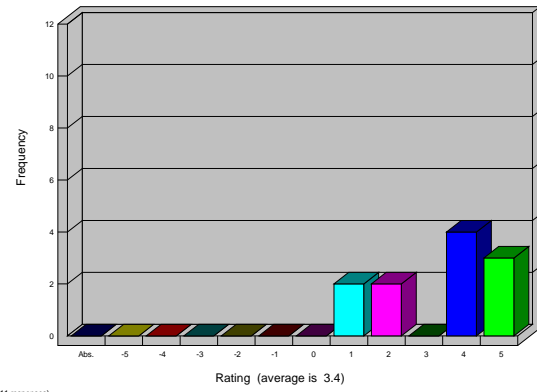
(11 responses)

impact van nieuwe ontwikkelingen zijn eenvoudiger te vertalen ...



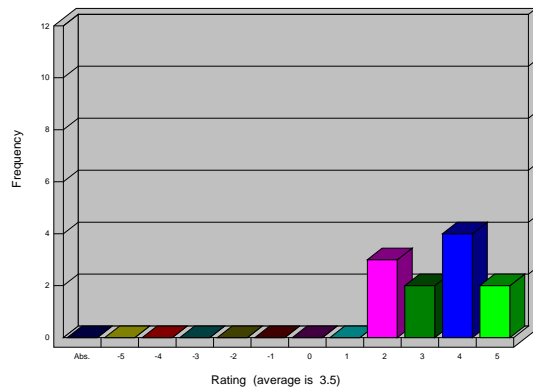
(11 responses)

applicaties die niet aan een proces gekoppeld kunnen worden zi...



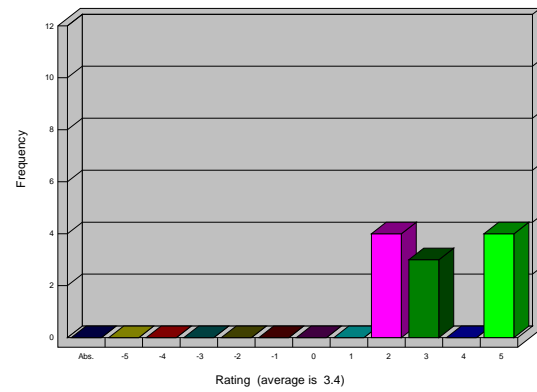
(11 responses)

Mogelijkheid om maatschappelijk vitale processen te kwalificer...



(11 responses)

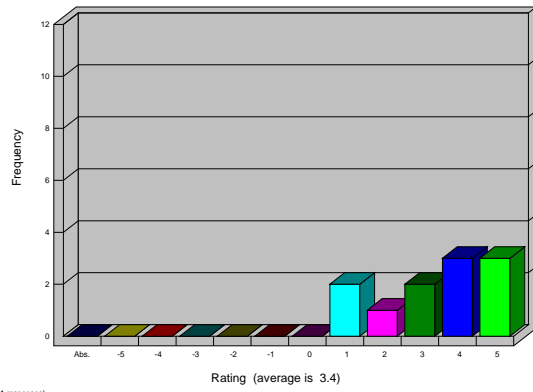
Door de koppeling worden de verantwoordelijkheden in UPP-model...



(11 responses)

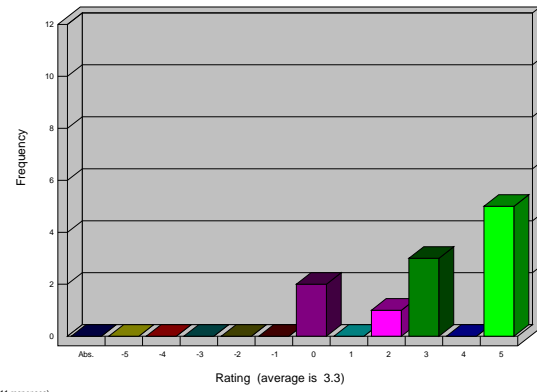
105

Voor integrale besluitvorming biedt het een kans om op basis v...



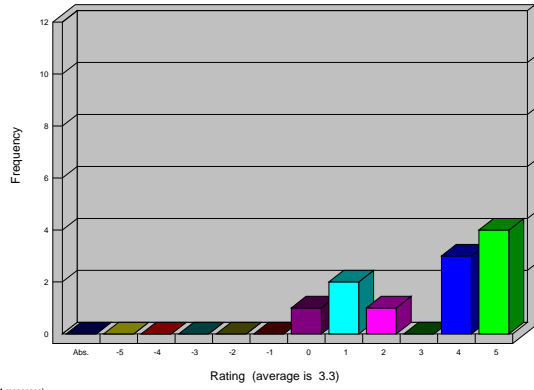
(11 responses)

Duidelijker verband tussen "missie" (de opdracht) en de feitel...

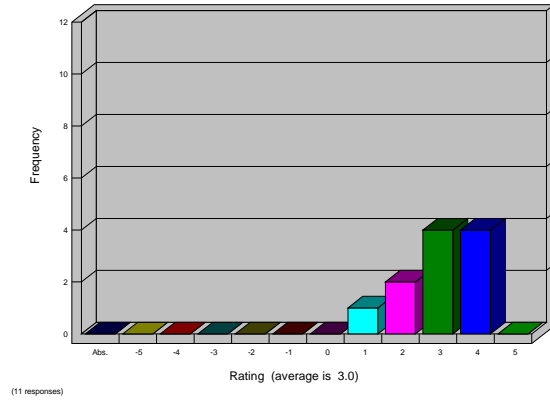


(11 responses)

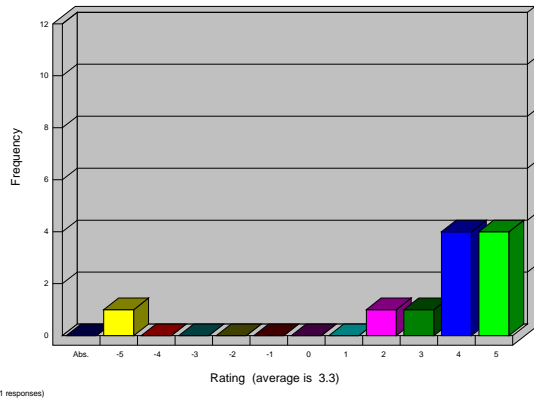
Door mijn proces door een DEMO-bril te bekijken, kan ik het pr...



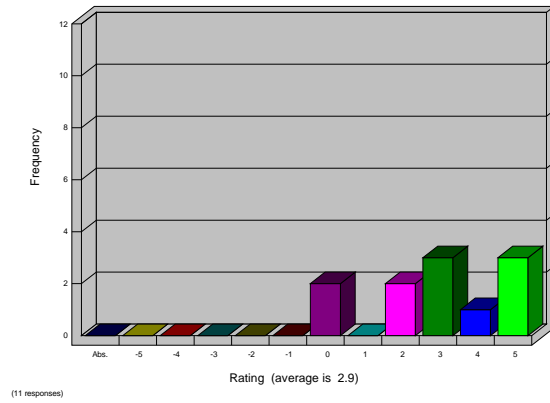
Het ondersteunen van de koppeling tussen 'procesbeheer' (tradi...



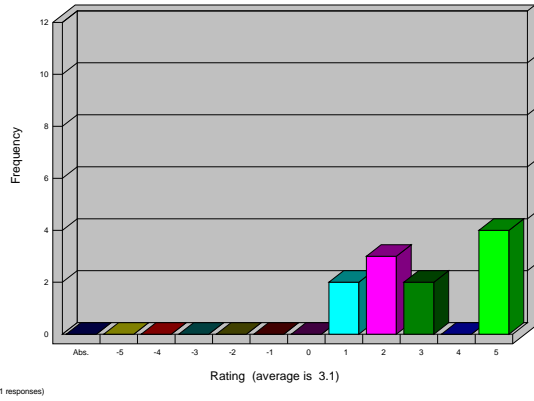
Kans: t.b.v applicatieconsolidatie: applicaties makkelijker i...



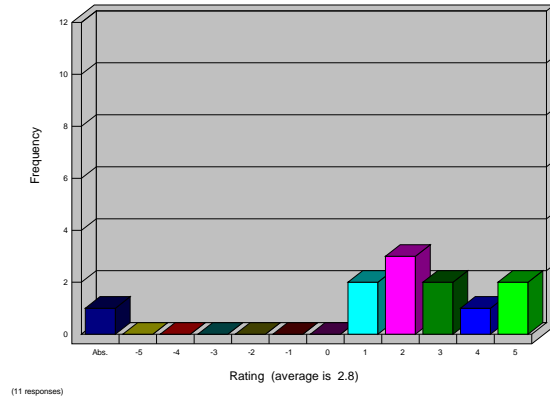
Kans: werknemers zullen niet alleen hun eigen proces begrijpen...



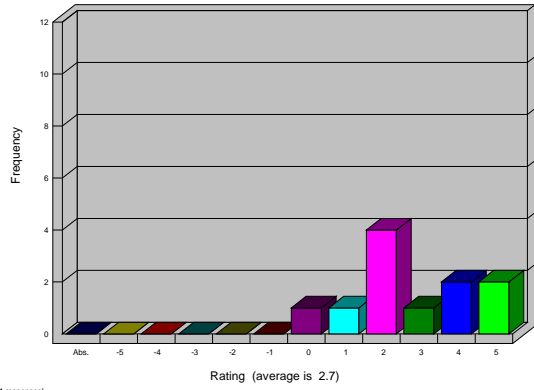
Kans: Het management begrijpt beter wat zijn personeel doet.



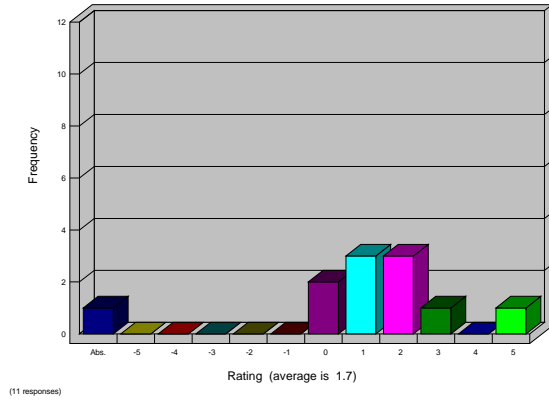
stapsgewijze besluitvorming over functionaristypen en organisa...



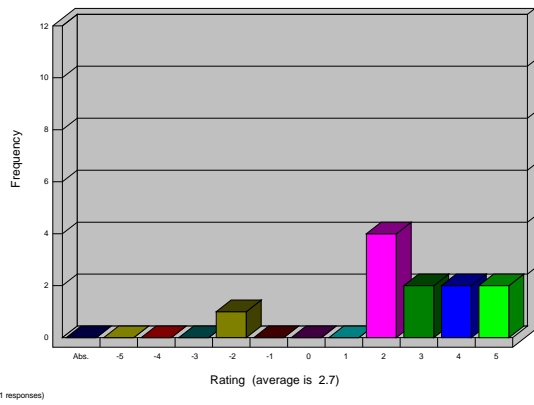
besluitvorming binnen directieteam RWS wordt integraler en daa...



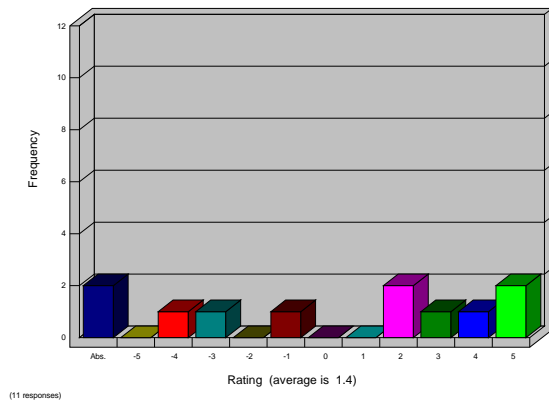
koppeling ondersteunt de afbakening (BPS-DID) tussen 'vraag' /...



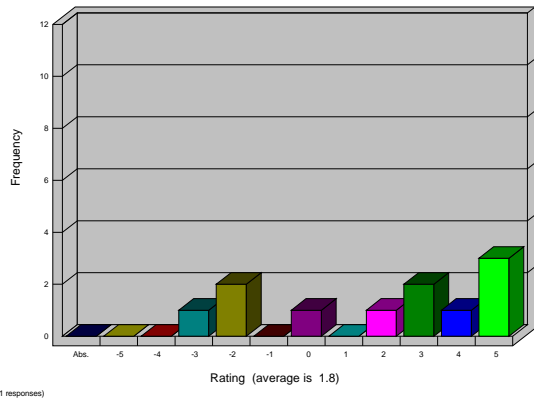
scherper duidelijk krijgen welke informatiebehoefden achter de...



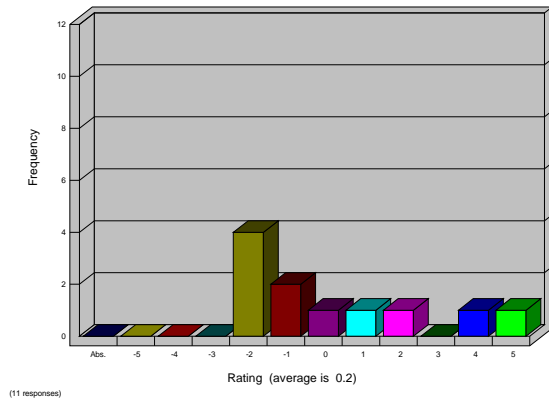
het streven naar een 1 op 1 vertaling van transacties naar fun...



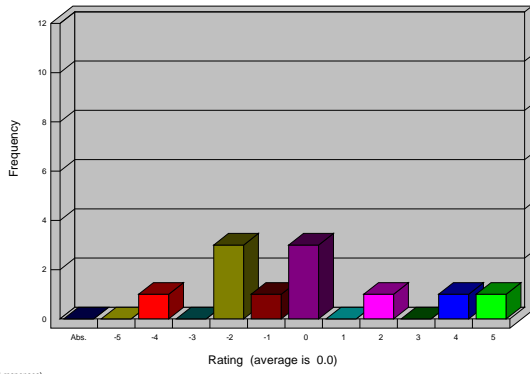
Een ideale combinatie van DVL en UPP zou m.i. zijn dat EPCs wo...



koppelen vraagt altijd om nauwe samenwerking en afstemming

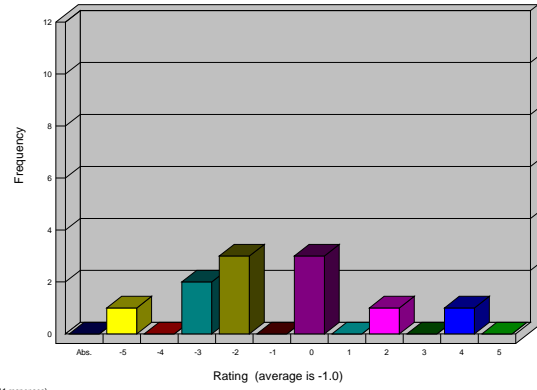


medewerkers voelen zich nog meer gebonden aan vaste werkwijze,...



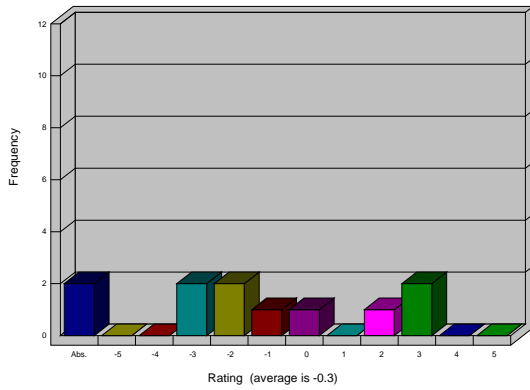
(11 responses)

de structuur van de UPP processen moet worden aangepast



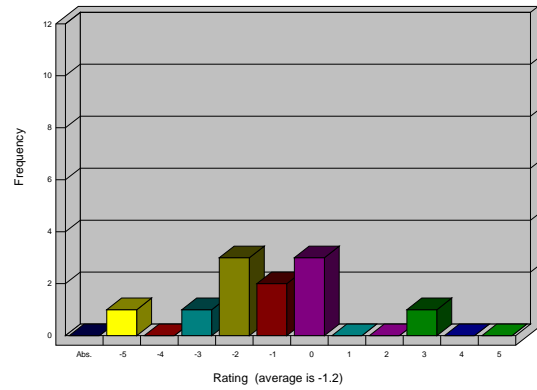
(11 responses)

Er is in principe geen een-op-een verband tussen transacties i...



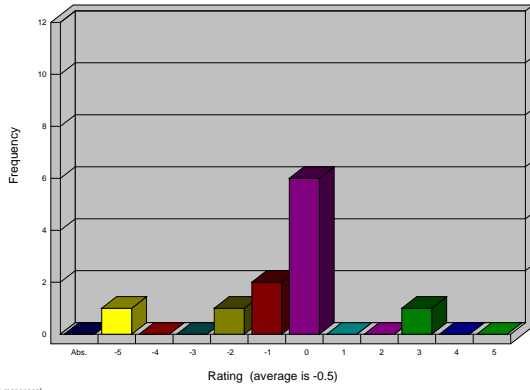
(11 responses)

UPP is "onderworpen" aan V&W-conventies, deze kunnen strijdig ...



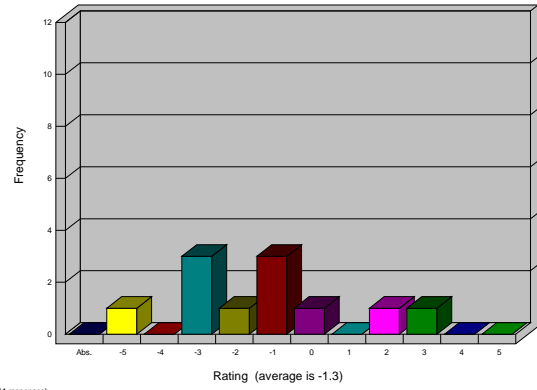
(11 responses)

Koppeling vertroebelt mogelijk het onderscheid tussen vraagfun...



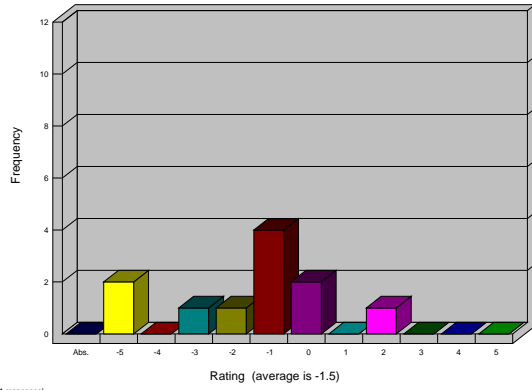
(11 responses)

Knelpunt: het is nodig de (afbakening en indeling van de) huid...



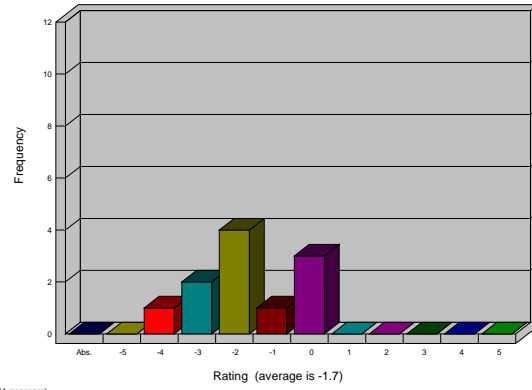
(11 responses)

Knelpunt: er is DEMO training nodig voor procesbeheerders



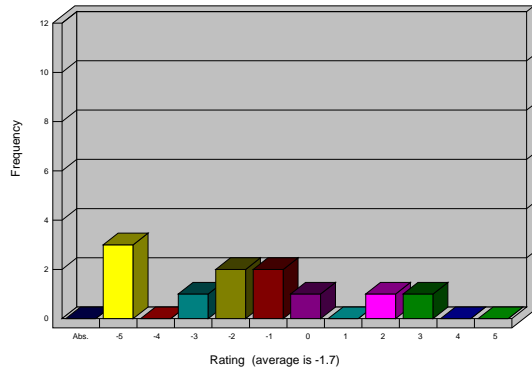
(11 responses)

Doelstelling van verschillende modellen kan verwarren



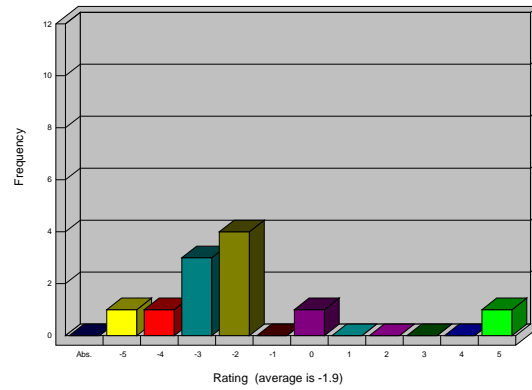
(11 responses)

De kosten van de aanpassing van de DVL-modellen en de UPP-mode...



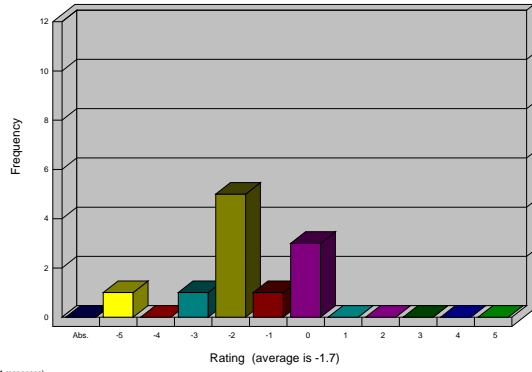
(11 responses)

Gescheiden beheer van DVL en Processen; noodzaak tot directe k...



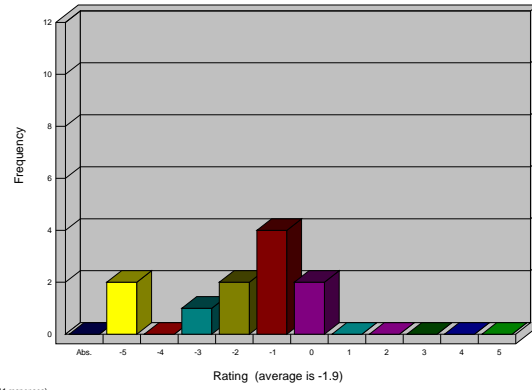
(11 responses)

Het kan overkomen alsof het beheren van DEMO-modellen en UPP-m...



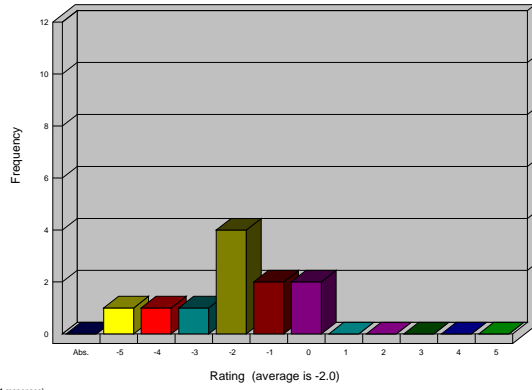
(11 responses)

UPP opgezet obv PDCA-cyclus; dit kan in combinatie met co'ordi...



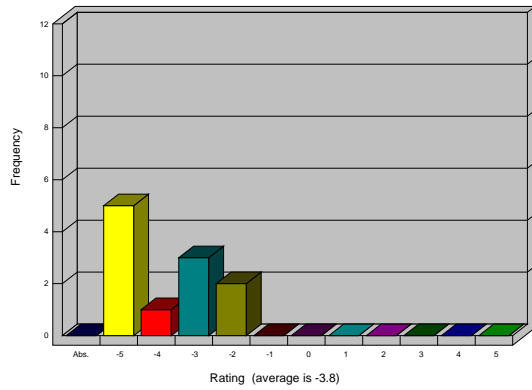
(11 responses)

het detail niveau van koppeling wordt te groot (bijv. van een ...



(11 responses)

Is hier een opdrachtgever van hogerhand voor te vinden (binnen...



(11 responses)

12.4.6 Detailed Rating of Guidelines

The table below shows the scores that the participants of the GDSS evaluation session gave to the guidelines. In these tables yellow columns contain scores by EAC, green columns contain scores by BPS, and blue columns contain scores by the external experts. GEM shows the average value (gemiddelde) and STD shows the standard deviation. To read comments from the participants regarding their reasoning for the scores they have given see appendix 12.4.7.

12-7 Detailed Rating of Guidelines

	Richtlijnen	1A	1A	1A	GEM	STD	2A	2A	GEM	STD	3A	3A	3A	GEM	STD
1	Bij een DVL, hoort een functieboom en bij een transactie hoort een functie. Hierbij starten extern geïnitieerde en zelf initiërende transacties nieuwe takken vanuit de boomwortel. Ingesloten transacties starten sub-functies in de bestaande takken.	4		5	4.50	0.71	5	5	5.00	0.00	5	5	5	5.00	0.00
2	Bij een functie hoort een EPC, en bij een sub-functie hoort een sub-EPC.	5		4	4.50	0.71	3	5	4.00	1.41	5	5	5	5.00	0.00
3	Een UPP proces is hangt af van actor rollen.	4		3	3.50	0.71	5	5	5.00	0.00	4	5	5	4.67	0.58
4	Een DVL model bevat ontologische transacties.	4	3	3	3.33	0.58	5	4	4.50	0.71	3	5	5	4.33	1.15
5	In een UPP FAD worden rollen omgezet naar actoren.	3		3	3.00	0.00	5	5	5.00	0.00	3	4	5	4.00	1.00
6	Een DVL model is op gelijke wijze afgebakend als de bijbehorende UPP cluster.	4	-2	3	1.67	3.21	5	5	5.00	0.00	5	5	4	4.67	0.58
7	Een UPP functieboom (niveau 4) bevat de bijbehorende ontologische functies.	4	-1	3	2.00	2.65	5	4	4.50	0.71	3	5	5	4.33	1.15
8	Een DVL model is op gelijke wijze gestructureerd als de bijbehorende UPP cluster.	3	-3	4	1.33	3.79	5	5	5.00	0.00	4	5	4	4.33	0.58
9	Een UPP proces wordt ingericht volgens het standaard transactiepatroon.	4		5	4.50	0.71	-5	-2	-3.50	2.12	2	5	5	4.00	1.73
10	Een UPP FAD (niveau 6) bevat de datalogische ondersteuning van de infologische activiteiten.	4	-3	4	1.67	4.04	-5	-2	-3.50	2.12	3	5	4	4.00	1.00
11	Een UPP proces (niveau 5) bevat de infologische ondersteuning van de ontologische functies.	2	-2	2	0.67	2.31	-3	-2	-2.50	0.71	1	3	4	2.67	1.53

111

The areas marked in red are areas of interest (Aoi). Having the proper experience Prof.dr.ing. Hans Mulder, the GDSS facilitator, carefully selected these Aois as follows: the scores were grouped by organization, thus EAC, BPS, and external. Then the average scores of the views were calculated per organization group. So the averages of EAC, BPS and of the externals. Then the averages were compared to each other in order to find Aois. Some of the Aois are briefly discussed below. I do not make any assumptions about the causes of the Aois. The causes should be discovered through discussions between EAC and BPS and perhaps the external experts. I only offer a brief explanation of how to interpret the Aois by giving some examples.

Aoi Scores **EAC / BPS / External**

Guideline 6: **1.67 / 5.00 / 4.67**

“Een DVL model is op gelijke wijze afgebakend als de bijbehorende UPP cluster.”

On average all groups thought this view was an opportunity but EAC seems to find it a small opportunity in comparison to BPS and the external experts.

Guidelines 7 and 8 show the same pattern of low, high, high scores as guideline 6.

Guideline 9: 4.50 / -3.50 / 4.00

“Een UPP proces wordt ingericht volgens het standaard transactiepatroon.”

EAC and the external experts seem to find this a strong opportunity in comparison to BPS that seems to find it a significant barrier.

Guideline 10 and 11 show a pattern of good from EAC, very good from experts and very bad from BPS.

12.4.7 Participant Comments on Guidelines

Bold statements are guidelines, the number value is the score given to the statement by the commenting participant, and next to the score is the participant's comment.

1 Een DVL model bevat ontologische transacties.

3.0 idealiter wel; voor de communicatie moet het toch mogelijk zijn af en toe infologische en zelfs datalogische transacties toe te voegen

2 In een UPP FAD worden rollen omgezet naar actoren.

5.0 Het zou raar zijn als het niet zo werkt. Hier komt de DVL-actor/UPP org. functie -matrix in beeld.

3 Een UPP functieboom (niveau 4) bevat de bijbehorende ontologische functies.

3.0 idealiter wel; voor de communicatie moet het toch mogelijk zijn af en toe infologische en zelfs datalogische functies toe te voegen

4 Een UPP proces wordt ingericht volgens het standaard transactiepatroon.

2.0 vaak een kwestie van zichtbaar maken wat er al impliciet in UPP zit

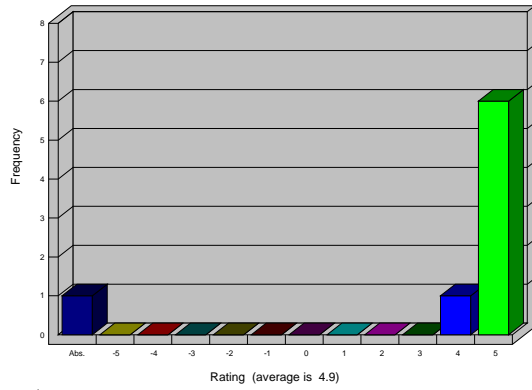
5 Een UPP proces (niveau 5) bevat de infologische ondersteuning van de ontologische functies.

1.0 wie heeft behoefte aan deze infologische beschrijving in UPP, gezien de UPP-doelgroep (vakvolwassen medewerker, die ondersteund wordt in zijn operationele proces)?

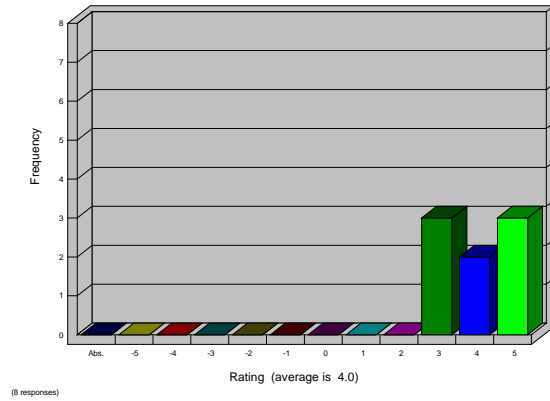
12.4.8 Dispersion Diagrams for Valuation of Guidelines

Each guideline has a dispersion diagram associated with it. The dispersion diagrams show the number of participants that rated the guideline a particular value. The first diagram shows that one participant abstained, one voted 4, and six participants valued the guideline with 5 points.

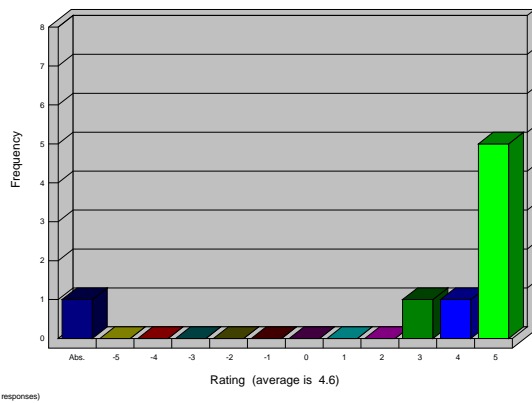
Bij een DVL, hoort een functieboom en bij een transactie hoort...



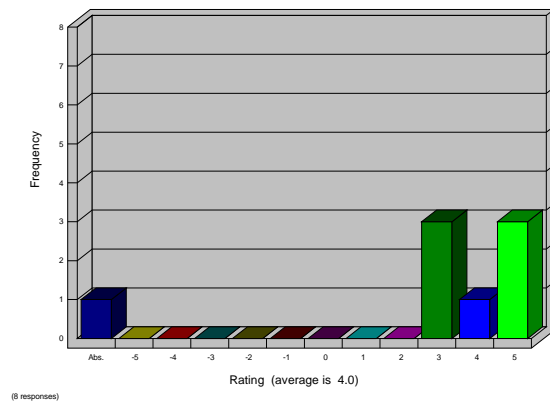
Een DVL model bevat ontlogische transacties.



Bij een functie hoort een EPC, en bij een sub-functie hoort ee...

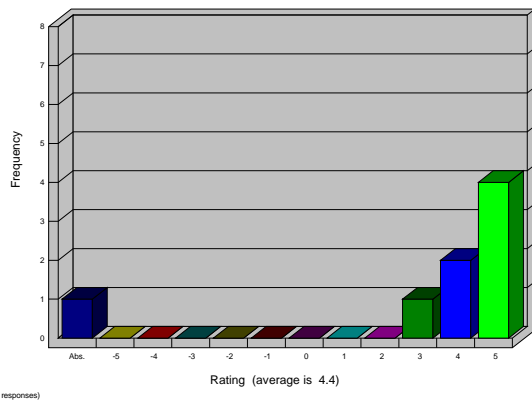


In een UPP FAD worden rollen omgezet naar actoren.

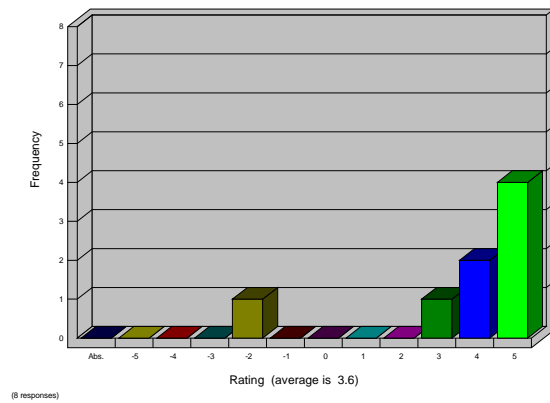


113

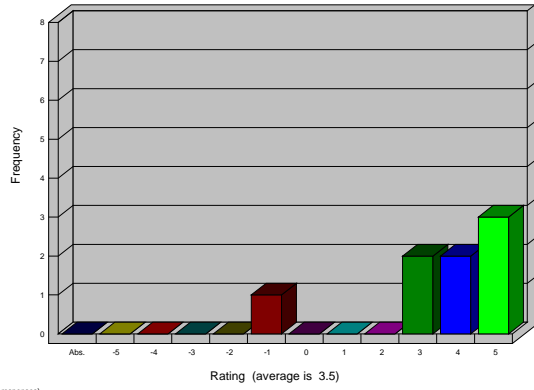
Een UPP proces is hangt af van actor rollen.



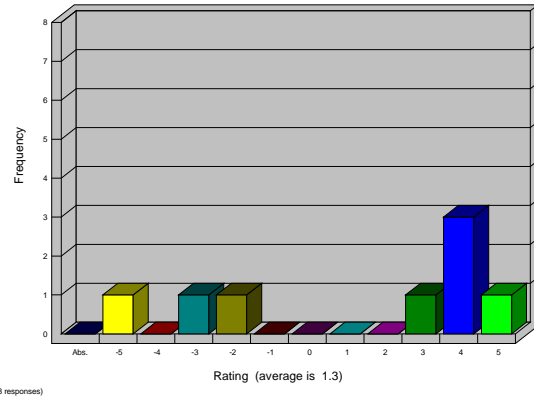
Een DVL model is op gelijke wijze afgebakend als de bijbehoren...



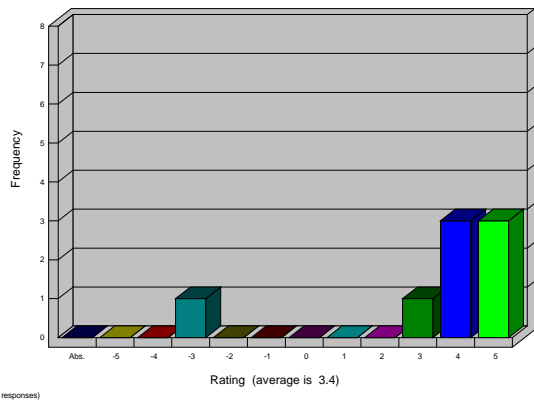
Een UPP functieboom (niveau 4) bevat de bijbehorende ontologi...



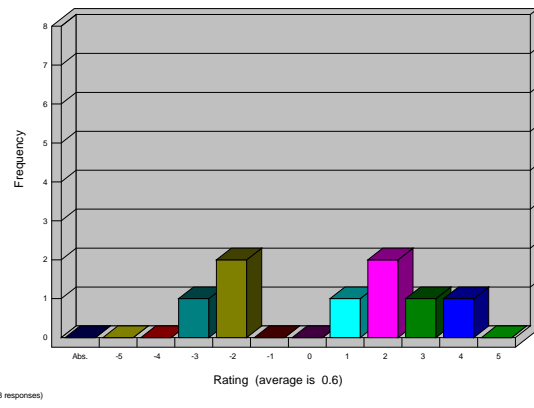
Een UPP FAD (niveau 6) bevat de datalogische ondersteuning van...



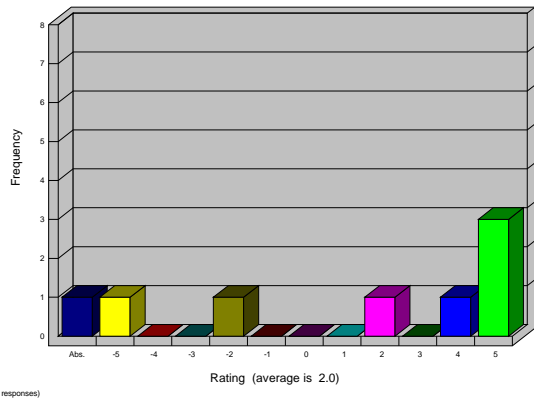
Een DVL model is op gelijke wijze gestructureerd als de bijbeh...



Een UPP proces (niveau 5) bevat de inflogische ondersteuning ...



Een UPP proces wordt ingericht volgens het standaard transacti...



12.4.9 Evaluation of the Session

After the GDSS session the participants had the opportunity to express their opinion on the usefulness of the meeting:

- Glashelder verhaal
- Zeer geslaagd
- Nuttige discussies over inhoud, meer begrip aan beide zijden
- Goed verhaal, praktisch en toepasbaar
- Duidelijk verhaal
- Kan zeker leiden tot succesvolle implementatie
- Consensus over noodzaak om dit door samenwerking verder te brengen
- Helpt RWS verder
- Het waardeverhaal, wat is het NUT van deze samenvoeging/samenwerking, is belangrijk om nog uit te voeren, wil je draagvlak krijgen voor de geïnventariseerde acties
- UPP ten voeten uit na realisatie
- Zelfs zonder koppeling levert het inhoudelijk nuttige resultaten
- Meerwaarde zit hem in het op gang houden van de discussie

13 Bibliography

Books

Dietz, J.L.G.	Enterprise Ontology, Theory and Methodology. 2006.
Mulder, J.B.F.	Rapid Enterprise Design. 2006.
Barjis, J.	Enterprise Information Systems, Automatic business process analysis and simulation based on DEMO. 2007.
Op 't Land, M.	Applying Architecture and Ontology to the Splitting and Allaying of Enterprises. 2008.
Dumas, M. Van der Aalst, W. Ter Hofstede, A.	Process Aware Information Systems. 2005.

Papers

Op 't Land, M., Middeljans, K., Buller, V.	Enterprise Ontology Based Application Portfolio Rationalization at Rijkswaterstaat. 2007.
Op 't Land, M., Dietz, J.,	Enterprise Ontology Based Splitting and Contracting of Organizations. 2008.
Albani, A., Dietz, J., De Matos, M.	DEMO and the improvement of ARIS Business Modeling. 2007.
Kozina, M.	Evaluation Of Aris And Zachman Frameworks As Enterprise Architectures., M. Kozina., 2006
Ferdian	A Comparison of Event-driven Process Chains and UML Activity Diagram for Denoting Business Processes

Handbooks

ConARIS	Conventie handboek Procesmodellieren in ARIS. Mei 2007.
BA	Raamwerkdocument Business Architectuur. December 2006.
DVL/WM B&O	Dienstverleningsmodel Watermanagement en Beheer & Onderhoud Nat. December 2007.
AppConNat/B&O	Applicatieconsolidatie NAT – Voorbereiding Besluitvorming. November 2007.

Related Work

Rajaram, A.	ARIS and DEMO, A Way of Working, Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology. 2008.
Strijdhartig, D.	Research Assignment, Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology. 2008.

Online Resources

-
- ⁱ www.minvenw.nl Ministerie van Verkeer en Waterstaat
- ⁱⁱ www.rijkswaterstaat.nl Rijkswaterstaat
- ⁱⁱⁱ www.ivw.nl Inspectie Verkeer en Waterstaat
- ^{iv} www.knmi.nl Koninklijke Nederlands Meteorologisch Instituut
- ^v www.rijkswaterstaat.nl/organisatie/adressen_en_diensten/data_ict_dienst/
Data-ICT-Dienst
- ^{vi} www.rijkswaterstaat.nl/organisatie/adressen_en_diensten/170-rws_cd
Corporate Dienst
- ^{vii} www.demo.nl DEMO knowledge base
- ^{viii} www.ids-scheer.nl IDS Scheer Nederland B.V.
- ^{ix} www.demo.nl DEMO knowledge base
- ^x www.ids-scheer.nl IDS Scheer Nederland B.V.
- ^{xi} www.viagroep.nl Venture Informatisering Adviesgroep N.V.