Compliance Checking In Supply Chain Management
An approach to check regulatory compliance of business processes

Shuzheng Wang (4179595)

Thesis committee:
Chair: Prof. Dr. Marijn Janssen
First supervisor: Dr. Huib Aldewereld
Second supervisor: Dr. Michel Oey
Third supervisor: Jie Jiang
Abstract:

Companies in the business world have to make their business processes compliant with the governmental regulations, otherwise, they may suffer from the law issues and economic losses. Due to globalization, checking for the conflicts between governmental regulations and the business processes in various countries is necessary, yet quite difficult. One of the problems is that if articles of regulation are interrelated, verifying the compliance cannot be achieved by just checking the compliance of each article separately as done traditionally. For this reason, we introduce Norm Nets, which is a regulation model that taking into consideration of the effect of interrelation among regulations. By using the model of Norm Nets, we proposed an approach to check the regulatory compliance. In this thesis, we first made a meta model of Norm Nets. Then we designed an automatic transformation tool from Norm Nets to Colored Petri Nets (the analysis tool used in this thesis) in order to improve the efficiency of the compliance checking approach. In the end, we verify the compliance checking approach through a case study.

Key words: Regulatory Compliance, Norm Nets, Colored Petri Net
Executive summary

As the behaviors of a natural person in the society are regulated by all kinds of regulations and laws, the commercial activity of any organization in the business world is also restrained by the relevant regulations and laws. However, conflicts may occur in any step of the business process if the organizations happen to neglect but a miniscule clause in the governmental regulations. Such conflicts happening in supply chain may lead to delays and/or economic losses. The relationship between the organization’s business process and the governmental regulations is one of the most critical relationships existing in the global supply chain. For an organization to be successful, it is necessary to make sure that all its business processes are always compliant with the governmental regulations. Therefore compliance checking is a necessary and important step in supply chain management.

However, regulatory compliance checking is a complex process. Besides the complexity of law system itself, the interrelation between different articles of law also makes it difficult to model the system and further check the compliance. In this thesis, we propose an approach of compliance checking with considering the interrelation between different articles of law. We apply the model named Norm Net to model the regulations and laws in the real world.

To realize such an approach completely is quite a huge and comprehensive project. It is not feasible to accomplish within six months as a master graduation project. Therefore we strictly limit our research scope and phrase our three sub research question as

1) How do Norm Nets represent the structural elements in governmental regulations?
2) How can we detect the conflicts between the business processes and the governmental regulations by using Norm Nets?
3) What suggestion can be derived from the case study result?

Accordingly, our research will be carried out by focusing on

1) Analyzing the model of Norm Nets
2) Representing Norm Nets in CPNs and designing the transformation tool
3) Validating the compliance checking approach through a case study.

By accomplishing this project, we could answer the main research question:

*Can the conflicts between governmental regulations and business processes be detected by the use of the model of Norm Nets?*

We contribute mainly on three points through this project: the meta model of Norm Nets, the automatic transformation tool and the case study result. The meta model of Norm Nets describes clearly the model structure of Norm Nets. It also naturally offers the transformation tool a framework of its input interface. The transformation tool we designed in this project is capable to translate Norm Nets to CPNs. In the case study, the transformation tool functions
normally. It outputted the CPNs as we aimed to. In the case study, we compared our compliance checking result with the situation in reality, together with the feedback from the expert, we conclude the result of our checking approach well reflected the reality of the case.

This approach still have several limitations, for example, currently it is not suitable for a compliance checking to a large number of articles. We discuss the limitations and future research in the discussion section in the end.
Acknowledgements

I would like to take this opportunity to sincerely thank all the people who help me with this graduation project.

First of all, I would like to thank my graduation committee. Prof. Dr. Marijn Janssen (the chair) and Dr. Michel Oey (the second supervisor) offered me a lot of useful ideas. Their suggestions not only gave me fresh ideas from new perspectives to the same technical question, but also helped me get out of the pure technical thinking, broadened my view to managerial and other aspects. Especially in the last phase, their suggestions helped me to improve the thesis to a new level. My first supervisor Dr. Huib Aldewereld and third supervisor Jie Jiang are the supervisors I contacted most. They offered me this opportunity to do this project in ICT section. Along the process, we almost met every week (according to my notebook, we totally had 16 meetings!). Thank you for your patience and time, solving every problem I met timely and encouraging me to go forward.

Secondly, I want to specially thank Dr. Ziv Baida from IBM Netherland BV. Thank you for your great support to my thesis. Thank you for your kindness and time to reply every email timely and answer every question I had via email, phone call and meetings.

Last but not the least, I want to thank all my friends and family. The lack of the technical support may delay my graduation process for months, but without your support, I know I would never achieve it. To every member of our TBM (Think Big Management) group (A. P. Kamalapurkar, M. J. G. N. Boon, N. A. Hattangadi, Nandyka Yogamaya and S. Sugavanam), it is amazing to do so many group assignments and lovely discussions with all of you together in the past two years. It is beyond my expectation that even for this final individual thesis, all of you gave me great support. You are awesome! Thank all my Chinese friends here in the Netherlands and in China. Your encouragement and support make me persevere in this tough task and hardly feel lonely. At last, I would like to give my special gratitude to my parents and girlfriend. Thank you for letting me choose my own future freely and fully supporting every choice I made all the time. It is my luck to live in such a happy family with all of you.

Thank you all!

Delft, July 2013
Shuzheng Wang
Table of Content

Abstract: ........................................................................................................................... 1

Executive summary ........................................................................................................... 2

Acknowledgements ......................................................................................................... 4

1. Introduction ................................................................................................................. 10
   1.1. Problem Statement ............................................................................................... 12
   1.2. Research Question ............................................................................................... 13
   1.3. Research scope .................................................................................................... 15
   1.4. Research Methodology ......................................................................................... 15
       1.4.1. The relevance cycle ...................................................................................... 15
       1.4.2. The Rigor cycle ............................................................................................. 16
       1.4.3. The design cycle ........................................................................................... 16
       1.4.4. The specific research methodology of the project ......................................... 17
   1.5. The expected outcome and contribution ............................................................... 18
   1.6. Outline ................................................................................................................ 19

Part 1 ............................................................................................................................... 21

2. Conceptual model ...................................................................................................... 22
   2.1. Research domain and related work ....................................................................... 22
   2.2. Conceptual model ................................................................................................ 24
   2.3. Core concepts ....................................................................................................... 26
       2.3.1. Governmental regulations modeling .............................................................. 26
       2.3.2. Business process management ...................................................................... 27
       2.3.3. Deontic logic ................................................................................................. 28
       2.3.4. Norm Nets ..................................................................................................... 28
       2.3.5. Colored Petri Nets ........................................................................................ 29
       2.3.6. Meta model ................................................................................................... 30
   2.4. Conclusion ............................................................................................................ 31

3. Software design specification ..................................................................................... 32
   3.1. The design process ............................................................................................... 32
   3.2. Design requirements ............................................................................................ 32
       3.2.1. Functional requirements .............................................................................. 32
3.2.2. Platform ........................................................................................................... 33
3.2.3. Extensibility ................................................................................................. 33
3.2.4. Maintainability ............................................................................................. 33
3.3. Input and output .............................................................................................. 33
3.4. Reflection ........................................................................................................ 34
4. Analyzing Norm Nets ....................................................................................... 35
4.1. Role-Action Pair ............................................................................................. 35
4.2. Formula ........................................................................................................... 36
4.3. Norm ............................................................................................................... 37
4.4. Norm Nets ....................................................................................................... 38
4.5. Conclusion ....................................................................................................... 39
5. Creating CPN ..................................................................................................... 41
5.1. Introduction of CPN ......................................................................................... 41
5.2. The Role-Action Pair in CPN ......................................................................... 43
5.3. The Formula in CPN Tools .............................................................................. 44
5.3.1. RapAnd ....................................................................................................... 44
5.3.2. RapOr .......................................................................................................... 46
5.3.3. RapBefore .................................................................................................... 47
5.3.4. RapNot ........................................................................................................ 48
5.4. From Norm to CPN ....................................................................................... 48
5.4.1. Create a generic Norm without the deontic type ...................................... 48
5.4.2. Deontic type .............................................................................................. 49
5.5. From Norm Nets to CPN .............................................................................. 53
5.5.1. AND .......................................................................................................... 53
5.5.2. OR ............................................................................................................. 56
5.5.3. OE (or else) .............................................................................................. 58
5.6. Conclusion ....................................................................................................... 60
Part 2 ...................................................................................................................... 63
6. Case study ........................................................................................................... 64
6.1. Introduction of the case .................................................................................. 64
6.2. Mapping from regulations to Norm Net ...................................................... 65
6.3. Mapping from Norm Nets to CPN ............................................................... 68
6.4. Reflection .............................................................................................................. 69
7. Validation of the model .......................................................................................... 71
  7.1. The Event Trace ................................................................................................. 71
  7.2. The compliance checking process ...................................................................... 76
  7.3. Result ................................................................................................................ 77
  7.4. Expert feedback ................................................................................................ 78
  7.5. Evaluation ........................................................................................................ 79
  7.6. Conclusion ........................................................................................................ 80
8. Conclusion ............................................................................................................. 82
9. Discussion .............................................................................................................. 85
  9.1. Regulatory compliance checking in supply chain management ....................... 85
  9.2. Limitations ...................................................................................................... 86
  9.3. Future research ............................................................................................... 87
10. Reflection ............................................................................................................ 89
  10.1. MoT Perspective ............................................................................................. 89
  10.2. Background of this thesis .............................................................................. 90
  10.3. Dos and Don’ts ............................................................................................. 90
Appendix 1 .................................................................................................................. 95
Appendix 2 .................................................................................................................. 96
Figures

Figure 1 Design science research methodology for the project (modified from (Hevner, 2007)) ................................................................. 17
Figure 2 Visual outline of this thesis................................................................................................................................. 20
Figure 3 Conceptual model ........................................................................................................................................... 25
Figure 4 Meta level (Eysholdt, 2009) ........................................................................................................................................... 30
Figure 5 Diagram of the Meta model of Norm Nets ........................................................................................................... 39
Figure 6 Elements in a colored petri nets......................................................................................................................... 42
Figure 7 The four self-defined basic elements in the model.............................................................................................. 43
Figure 8 Rap in CPN ......................................................................................................................................................... 43
Figure 9 Formula And in CPN tools........................................................................................................................................ 45
Figure 10 Formula Or in CPN tools ........................................................................................................................................ 46
Figure 11 Formula Before in CPN tools .......................................................................................................................... 47
Figure 12 Norm in CPN tools ................................................................................................................................................ 49
Figure 13 Deontic type Obligation in CPN tools .................................................................................................................. 50
Figure 14 Deontic type Prohibition in CPN tools ................................................................................................................ 52
Figure 15 Norm AND in CPN tools ....................................................................................................................................... 54
Figure 16 Norm OR in CPN tools ......................................................................................................................................... 57
Figure 17 Norm OE in CPN tools................................................................................................................................................. 57
Figure 18 The CPN diagram of the case ............................................................................................................................ 69
Figure 19 Event trace 1.......................................................................................................................................................... 73
Figure 20 Event trace 2.......................................................................................................................................................... 73
Figure 21 Event trace 3.......................................................................................................................................................... 73
Figure 22 Event trace 4.......................................................................................................................................................... 74
Figure 23 Event trace 5.......................................................................................................................................................... 74
Figure 24 Event trace 6.......................................................................................................................................................... 75
Figure 25 Event trace 7.......................................................................................................................................................... 75
Figure 26 Flow chart of the simulation process .............................................................................................................. 77
Figure 27 The CPN analysis result ....................................................................................................................................... 78
Table

Table 1 Comparison between governmental regulations and business process .................. 10
Table 2 Functional requirements of the transformation tool ........................................ 33
Table 3 Raps in the example ....................................................................................... 36
Table 4 The instances of formulas in the example ...................................................... 37
Table 5 CPN representation result of Rap .................................................................... 43
Table 6 Comparison table of RapAnd .......................................................................... 46
Table 7 Comparison table of RapOr ............................................................................ 47
Table 8 Comparison table of RapBefore ................................................................. 48
Table 9 Comparison between logical result and CPN representation in obligation norm ... 52
Table 10 Comparison between logical result and CPN representation in prohibition norm .. 53
Table 11 Comparison table of AND ........................................................................... 56
Table 12 Comparison table of OR ............................................................................. 58
Table 13 Comparison table of OE .............................................................................. 60
Table 14 The actions in regulations ............................................................................ 72
1. Introduction

Nowadays, companies function in an increasingly complex and dynamic business environment. An emerging globalization, new economic challenges, rapid advancements in information technologies (IT), and the requirements of multi-facet skills are only some of the challenges facing businesses today (Mohamed & Lashine, 2003). The companies are also under pressure to comply with an increasing number of external regulations (zur Muehlen, et al., 2007). This condition especially holds for the high-tech companies which are serving in the global supply chain, as they need to first understand the governmental regulations both in their own country and in the other countries with which they are trading. The managers of these companies need design or adjust their own business process accordingly, making sure the business runs properly within the constraints of the relevant governmental regulations.

To understand why governmental regulations and business processes may conflict, we should first understand what purpose both are designed for and what different properties they have. Business processes are designed by business organizations to serve as guidelines for their business operations, which on the one hand improves the working efficiency, and on the other hand restricts the way that the relevant actors behave. Business processes focus on the action sequence. While governmental regulations are issued by the authorities to ensure the social security, protecting local interests and regulate the business activities. Normally, governmental regulations focus on state of affairs, which means they judge whether the particular state complies with the regulations or not. Table 1 shows us a comparison between the two conceptions. The politicians writing the laws are only interested in the intended effects. The laws and regulations are aimed at creating that effect (what to be done and what not to be done), without providing all the implementation details (how it to be done) (Gong & Janssen, 2012). The “how” question is supposed to be answered by the business organizations in their business processes.

<table>
<thead>
<tr>
<th>Governmental regulations</th>
<th>Business process</th>
</tr>
</thead>
<tbody>
<tr>
<td>Issued by authorities</td>
<td>Designed and implemented by organizations</td>
</tr>
<tr>
<td>National perspective</td>
<td>Organizational perspective</td>
</tr>
<tr>
<td>Focus on state of affairs</td>
<td>Focus on action sequence</td>
</tr>
<tr>
<td>Increase Social welfare</td>
<td>Increase the efficiency of the organizations to achieve the highest organizational benefit</td>
</tr>
<tr>
<td>Economics, safety, environment concern</td>
<td></td>
</tr>
<tr>
<td>Legal authority</td>
<td>Secondary legal authority</td>
</tr>
</tbody>
</table>

*Table 1 Comparison between governmental regulations and business process*

Regulatory compliance problem also exists in supply chain management. Supply chain is defined as “the systemic, strategic coordination of the traditional business functions and the tactics across these business functions within a particular company and across businesses
within the supply chain, for the purposes of improving the long-term performance of the individual companies and the supply chain as a whole” (Mentzer, et al., 2001). For the high-tech companies which locate part of their supply chain abroad, supply chain management have other properties. Such companies typically rely on the collaboration and information sharing with other actors in the supply chain (Hau L. Lee & Tang, 2004).

A large number of actors (e.g. suppliers, assemblers, governmental agencies, transporters, etc.) are involved in the supply chain. In such a system, regulations and laws stay on the top with their legal authorities. All the actors’ behaviors are ruled by the relevant regulations and laws, while they also have their own interests and values. Different actors may not aware of the other actors’ interests, or their interests and values are conflicting. In addition, the strategy related internal (e.g. the production plan and schedule) and external factors (e.g. the local culture and law issues) are also factors which organizations need to consider when they design and implement their business processes. Therefore managing and optimizing the whole supply chain is extremely complex. Ignoring any of those factors may lead to conflicts. In this thesis, we focus on conflicts checking between the business process and the governmental regulations. Regulatory compliance checking is vital for supply chain management as it can directly and indirectly influenced the main performance indicators of supply chain, such as cost, lead time, stock turnover, etc. Conflicts with regulations and laws may cause an unexpected delay or fine. Regulatory compliance is one of fundamental requirements for the success of supply chain management.

In this thesis project, we apply a new approach by using the model of Norm Nets (NNs). One of the deliverables (the design artifact) of the project is used to transform the contents in Norm Nets to Colored Petri Nets (CPNs, the analysis tool used for analysis in this project). The whole approach is later assessed through a case study. The case is from IBM Corporation. It is a project which aims to help the customers to set up their e-customs framework for the declaration management system.

The customs declaration process is a crucial step in the global supply chain. Any company whose supply chain involves in import and export activities needs to pay special attention on such process as this may affect its supply chain in terms of lead time, cost and reliability. The customs declaration process is a complex process which is subject to change (van de Aalst, 1996). Typical activities in the process are related to the registration, checking and control of movements of goods. The management of the processes is still lack of ICT system support. As a result, the processes are hard to manage. In such process, customs and the declarant are the two main parties. The customs department has the responsibility of checking and controlling on import and export goods. Their main interest is to make the import and export processes compliant with the relevant regulations, while the declarants have strong demand for a high speed of which the applications are processed. A fast and smooth customs declaration process can shorten the delivery time and improve the cost efficiency and reliability of the supply chain. Once mistakes or delay occur, various processes within the supply chain such as the purchase process, the production in factories, the storage and sales on the market etc. may all be affected. In addition, a high ratio of mistake and delay of the
customs process may directly lead to the financial loss of the declaring companies and even decrease the economic attraction of the region.

In the case study, the governmental regulations are represented by the legal articles which have been interpreted in the form “if... then...”, and the business processes are the process framework which IBM developed to design the declaration management system for the customs department. We apply our approach to check the conflicts. Based on the findings during the research and the final result, we validate the formalism of Norm Nets as well as the new compliance checking approach.

1.1. Problem Statement

Supply chains are crucial and complex network to manage for companies, especially for those high-tech companies which located part of their supply chain in foreign countries. Such companies have to run their business by following their own business process, while complying with the local regulations and the laws of the countries in which they operate.

As mentioned in previous section, the regulations and laws state “what should be done” and the business processes solve the problem “how should it to be done”. Compliance is a complex area which requires the involvement of specialists from different functional departments, the compliance officers, legal experts, business process designers and operational managers, etc. (Boella, et al., 2013) The common way of relating the responsibilities of the various departments is called the Three lines of Defense model (Doughty, 2011). The first line of defense is in the business, which uses business process to mitigate regulatory risks. The second line of defense consists of risk managers, compliance officers and other specialists in a role of setting norms. The third line of defense consists of internal and external auditors, who verify whether the norms are adhered to and are effective.

According to such a model, companies are supposed to make their business process compliant with the governmental regulations to form the first defense line. However, because the regulative system is already fairly complex in terms of all kinds of pre-conditions, targets and deadlines, the companies’ task is made even more difficult by the fact that auditors can interpret norms more or less strictly, depending on the public opinion and the policies of the day (Boella, et al., 2013). Moreover, interdependencies between different regulations may further increase the complexity, inconsistencies between governmental regulations and business processes might occur. It has two patterns of manifestation: 1) Companies may find out that they cannot comply with all the governmental regulations no matter how they process their businesses, in such case, we need to inform the regulations makers to modify or improve their regulatory system. 2) There are one or more potential compliance path for companies, but the business processes of the organization does not choose the right one. What they choose conflicts with the governmental regulations. This may be caused by several reasons. In a complex business environment, the designer or the managers of the companies may have neglected some vital items in the governmental regulations unintentionally, or they
lack of the specific knowledge. In either situation, the business process may conflict with the governmental regulations and directly lead to law issues and economic loss to the companies. In this thesis, we leave the legislation system and interpretation problem out of the research scope and aim at the interrelation between articles of law.

Not just for avoiding the conflicts, it is also important for the companies to identify the reason, the type and the degree of the potential violations, to make proper decisions in order to get a certain compliance condition with lowest cost and highest benefit. (Governatori & Sadiq, 2009) defined three types of compliance, namely 1) ideal, where execution paths do not violate regulation; 2) sub-ideal, where violations are recovered via compensatory measures; 3) non-ideal, where violations are not repaired. In order to foresee and take measures to be in a more beneficial condition, we need to detect the potential conflicts between business processes and governmental regulations and identify the reasons of the conflicts. Solving this problem could not only help companies to check the conflicts in their existing business processes, and thus adjust themselves to avoid such problems, but also help the company to design regulation-compliant business processes. (Koehler, 2011) outlined different scenarios of interaction between process and regulations, that fall into four patterns: 1) business rules are embedded in the business process model as gateways; 2) business rules are kept separate and consulted in gateway decisions. In more complex variations, several domain-specific rule sets are maintained; 3) business rules consulted in gateways decisions depend on the outcomes of other processes or on previous instances of the current process; 4) business rules dynamically direct the business process flow, creating new business processes on the fly from given components, depending on the interplay between context-sensitive business rules. Some variations even generate business rules on the fly. In this thesis, we are going to apply our approach to check the compliance problem. We hope the model we generated and the software we developed in this project can help us analyze the interaction between process and regulations. Our work can also be base for the future study to assist process design and even directly generate compliant business process.

Moreover, it is also helpful to the governmental authorities and the policy-makers to check whether conflicts exist in the governmental regulation itself, especially for the regulative system with multiple hierarchies, e.g. regulative system which contains the state law, the local law, the industry regulation, etc. All the laws and regulations need to be reconciled with each other.

1.2. Research Question

This project implements an approach to check the compliance between business processes and governmental regulations with Norm Nets. It aims to validate and improve the model of Norm Nets (Jiang, et al., 2012) through a case study. We develop a transformation tool which is capable to transform Norm Nets into CPNs (Colored Petri Nets) for further analysis. As part of the model development of Norm Nets, we aim to assess and refine the model, making it a
reliable approach to check the conflicts between governmental regulations and business processes.

Accordingly, we formulate the research question as below:

*Research question: Can the conflicts between governmental regulations and business processes be detected by the use of Norm Nets?*

We propose an approach of conflicts checking between governmental regulations and business processes based on the model of Norm Nets. In this project, we demonstrate the whole approach, especially focusing on automatizing the transforming process from Norm Nets to CPNs. And in the last section we validate the approach by a case study. After the analysis and the case study, we are able to give answer to this question.

We further split the main question into three sub questions as below:

*Sub-question 1: How do Norm Nets represent the structural elements in governmental regulations?*

There are several ways to represent governmental regulations. The model of Norm Nets is one of them and was developed by (Jiang, et al., 2013). The advantage of Norm Nets is that this model includes the interdependency among norms. However, the applicability of the model still need to be further validated. In Chapter 4, we perform a model analysis on Norm Nets. Through this analysis, the meta model of Norm Nets is established. The structure as well as the components of Norm Nets is demonstrated in the meta model. Then we can explain show how Norm Net represent the structural elements in governmental regulations by its formalism.

*Sub-question 2: How can we detect the conflicts between the business processes and the governmental regulations by using Norm Nets?*

To answer this question we chose the support of CPN tools (the name of the analysis software used in this research). CPN tools is able to represent multiple states of a model which consist all real-world concepts of behaviors, works, actors, regulatory requirements, and offers the necessary analyzing functions. Since Norm Nets is a formalism which is not capable to detect and analysis the conflicts, we aim to map the model of Norm Nets to the model of CPN tools. By this model-to-model analysis, we connect the representational formalism to the analysis tool. By processing the data in CPN tools, we get all the states of the case, therefore the conflicts are detected.

*Sub-question 3: What suggestion can be derived from the case study result?*

In Part 2 we run our case study. There are two goals we want to attain through the case study.

1) According to the analysis result of the case study, we want to find the potential conflicts between governmental regulations and the business processes in our target company.
2) Based on the research, we are also intended to validate Norm Nets to see whether this new model is sufficient to represent all the governmental regulations. By finding and solving the problems, we develop and improve the model further.

The result of this research aims to answers all formulated sub research questions, and subsequently providing an answer to the main research question.

1.3. Research scope

In order to detect the conflicts between governmental regulations and business rules, we apply the approach by 1) translating natural language into Norm Nets, 2) transforming Norm Nets to CPNs, 3) transforming business process to CPN tools and 4) analyzing the CPNs and get the compliance checking result in CPN tools. In this research, we focus on the transforming process from Norm Nets to CPN tools and the final analysis. Correspondingly, the scope of the project includes decomposing of the model of Norm Nets, representing Norm Nets in CPNs elements and the case study to validate the approach.

Decomposing the model of Norm Nets is a structural analysis on the model. We aim to identify all the elements in the model and the way how these elements are organized in Norm Nets. By doing this step, we have a clear view on the model of Norm Nets and get the meta model of Norm Nets. Then we analyze the model of CPNs and find its meta model. By linking these two meta models help us understand the relationship between Norm Nets and CPNs so that we can design a transformation tool which can transform Norm Nets to CPNs automatically. Following the result obtained from the steps above, we perform a case study, where our aim is to check the potential conflicts by using our model. This process is also a validation of the model as well as the compliance checking approach.

Norm Nets is mainly for modeling the regulations and laws. The translation from natural language to Norm Nets and the transformation of business process to CPN tools is also important. They are used to complete the compliance checking approach, but they are out of the research scope of this project. We take existing methods and theories to fulfill these tasks.

1.4. Research Methodology

In the project, we design a transformation tool, which helps us perform the approach of compliance checking by use of Norm Nets. As we aim to design an IT artifact, we apply the design science methodology for this research. According to Alan R. Hevner’s ‘Three Cycle View of Design Science research’ (Hevner, 2007), we illustrate the corresponding three cycles of design for this research in this section (as shown in Figure 1).

1.4.1. The relevance cycle

The relevance cycle is the cycle which connects the empirical foundation and the design project. It initiates design science research with an application context that not only provides
the requirements for the research as input but also designs the acceptance criteria for the evaluation of the research result. (Hevner, 2007)

In this part we define the application domain and the problem. As described in the section 1.1, we check the potential conflicts existing between governmental regulations with interdependencies and businesses processes. We introduce the approach of compliance checking by the use of Norm Nets. The artifact which we use that is capable to translate the information in Norm Nets into CPN tools. We believe this transformation tool could help us to make the checking process more efficient.

1.4.2. The Rigor cycle

The rigor cycle links the design research with the theoretical foundation. The rigor cycle provides past knowledge to the research project. As mentioned by Juhani, 'It is the rigor of constructing IT artifacts that distinguishes information systems as design science from the practice of building IT artifacts' (Juhani, 2007).

We define the kernel theory for our project as law modeling Norm Nets and compliance checking methodology. We argue that Norm Nets is a new and better model in the law modeling domain, because this model not only analyzes the effect of individual norm in the normative agent system but also takes into account the interdependencies and interrelations between norms. The transformation tool we designed in this project is a new functional module in the approach. Moreover, by performing the research during the design of the artifact, we aim to refine the theory and method, enriching the knowledge base.

1.4.3. The design cycle

The internal design cycle is the heart of the design science research approach. The cycle of research activities iterates the construction of the artifact, its evaluation, and subsequent feedback to refine the design further.

By applying the model from (Verschuren & Hartog, 2005), this design cycle step could be divided into six stages: 1) First hunch, 2) Requirements and assumptions, 3) Structural specifications, 4) prototype, 5) implementation and 6) Evaluation. The first five steps are the actual design and construction phases, while the last step evaluation step is also vital for the whole process. The other another (Juhani, 2007) argues that during the performance of the design cycle it is important to maintain a balance between the efforts spent in construction and evaluating the evolving design artifact. The evaluation could be catalogued into three main types: 1) Plan evaluation, 2) Process evolution and 3) Production evaluation (Verschuren & Hartog, 2005). The plan evaluation phase mainly covers the first three stages of the first model we proposed (first hunch, requirements and assumptions, structural specifications). The process evaluation mainly focuses on the stages of prototype and implementation. The production evaluation covers the last stage in the designing cycle, the evaluation.

Furthermore, a Design Science Research Methodology is implemented for the whole research as shown in Figure 1. After understanding the problem and the opportunities in the specific
domain which is the empirical foundation for this project, a literature review is performed to build up the necessary theoretical foundation. Meanwhile the necessary tools for solving the problem should be gathered and selected. Then, the main part of the project-design part starts. We start from the conceptual design, to make model analysis of Norm Nets and a model analysis of the CPNs, until connecting the two models and building up the transformation tool. At last, the validity of the model is tested by a case study. According to the result of the case study, the model could be further improved.

**Figure 1 Design science research methodology for the project (modified from Hevner, 2007)**

1.4.4. The specific research methodology of the project

Based on the three cycles of design science methodology (Hevner, 2007), this part illustrates explicitly the specific research methodology for this project.

The relevance cycle is the cycle which links the empirical foundation and the design project. Chapter 1 introduces the background of the thesis. Chapter 3 defines the requirements for the design artifact. These two chapters covers the content of the upper arrow in relevance cycle stands for (Figure 1). In chapter 8, 9 and 10, we reflect on the compliance checking approach to explain the impacts on the relevant parties in supply chain, which is represented by the lower arrow on this cycle (Figure 1).

The design cycle is in the central position of design science research. Chapter 5 illustrates how we develop the transformation tool, while chapter 6 and 7 evaluates the transformation tool as well as the compliance checking approach through the case study. They are shown by the upper and lower arrows in the design cycle, respectively (Figure 1).
The rigor cycle links the design research with the theoretical foundation. Chapter 2 demonstrate the relevant concepts and theories. It is the theoretical foundation of the whole thesis, while chapter 4 describes the Norm Nets model, which is one of the kernel theories of this project. These two chapters reflects the rigor cycle in this thesis (Figure 1).

The implementation during this project consists mainly out of two steps, the artifact programming which is used to transform the contents of Norm Nets to Colored Petri Nets, and the case study which implements the new compliance checking approach.

For the first step, we decide Java to be the programming language. Java is a general-purpose, concurrent, object-oriented computer programming language that is specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere" (WORA), meaning that code that runs on one platform does not need to be recompiled to run on another. We have two reasons to choose Java for this project. The first argument is that Java is able to describe the logical construction in Norm Nets and make connection between the model of Norm Nets and the model of CPNs. The second reason is that CPN tools has already offered part of its source code and useful methods in Java which are ready to be invoked by Java programmers. Therefore it is relatively easy for us to design our interface between Norm Nets and CPNs in Java.

For the second step, we run the case study base on a case which comes from IBM on customs declaration management system. In this project, the declaration management process which is designed by IBM for the customs department will be verified according to the corresponding articles of law. We chose a part of available case data from this case to run our model to validate and refine the model based on the result.

1.5. The expected outcome and contribution

For this project, we have three main expected outcomes – the meta model of Norm Nets, the transformation tool and the case study result.

The meta model shows us the framework of Norm Nets. It also help us with the model-to-model design of the transformation tool, as we are intended to transform Norm Nets to CPNs. We first clarify the meta model of Norm Nets and then we can find its correspondence with CPNs and link these two.

The transformation tool is used to transform Norm Nets into CPN automatically. Before this project, this step is done by the analysts manually. This transformation tool can increases the speed of this transformation and decrease the possibility of making mistakes. Success of doing this can improve the efficiency of our compliance checking approach.

There are two objectives to conduct the case study in this project. We want to validate Norm Nets as well as our compliance checking approach though the case study. The checking result can also offer the company its compliance information.
By accomplishing this project, we hope to contribute in:

1) Making our compliance checking approach one step forward to full automation.
2) Validating and refine Norm Nets as well the compliance checking approach.
3) Offering the organizations which are required to be regulatory compliance one approach to fulfill the compliance checking task.
4) Besides the signal of violation or conflicts, offering the organizations more information in terms of the reason, the type and the degree of the problem.
5) Helping the organizations to mitigate risks and increase the efficiency in their supply chain operation.

1.6. Outline

This thesis consists of two parts. Part 1 comprises of chapters from 2 to 5. Chapter 2 gives the conceptual model of the project and explanation of the core conceptions used in this thesis. Starting from Chapter 3, we structure the design part of this thesis by, defining the design requirement, analyzing the model of Norm Nets, introducing the meta model of CPNs and mapping Norm Nets to CPNs. Though the analysis of this part, we can answer sub-research question 1.

Part 2 comprises of chapters 6 through 10. In Chapter 6 we discuss the background of the case study, the procedure of implementation and the simulation result. Chapter 7 presents our validation results and answers sub-question 2 and 3. Then in Chapter 8 we conclude the thesis and answer main research question. Next in Chapter 9, we discuss the limitations of this research and possible directions of future work and our reflection. Finally in Chapter 10, we present our reflection of the whole graduation project.
Can the conflicts between governmental regulations and business processes be detected by the use of the model of Norm Nets?

Figure 2 Visual outline of this thesis
2. Conceptual model

In this chapter we give a whole picture of the conceptual structure of the project. In this section we first have an overview on the conceptual structure by introducing the research domain and the conceptual model of the project. Then we take a close look at every core concept related to this thesis. We clarify the specific meanings of the concepts so that the readers can understand them while we mention them in the following chapters.

2.1. Research domain and related work

In terms of conflicts checking in supply chain, there are several different perspectives. For instance, as what (Blackhurst, et al., 2008) describes, the conflicts may come out because that each individual entity may be working towards goals that optimizing their own revenue or efficiency, while decreasing the efficiency of the integrated system as whole. Other researches take different perspectives, e.g. contracts need to be coordinated, otherwise conflicts take place for contracts (Cachon, 2003). In this thesis, we focus on the compliance checking between business processes and governmental regulations.

As described in previous chapters, checking for the conflicts between governmental regulations and the business processes is necessary yet difficult. One of the difficulties occurs because of the interrelationship between actors and the context in the business environment. The actor in a multi-actor system may behave in different ways to complete their specified tasks in the business environment. Different behaviors of the actors create different states of the system and influence the final result. For the analysis and assessment of this system, a large number of combinations of actors' behaviors would mean that a large quantity of system states are present, which increases the complexity level for the analysis. Traditionally, organizations assure compliance through regular audits and reviews (zur Muehlen, et al., 2007) (i.e. one or a team of lawyers or process specialists design and check the business process item by item with the relevant governmental regulations). The large number of combinations of system states makes it difficult to check all the possibilities in this way. In addition, the whole system (which is combined by actors’ individual behaviors) is not static but dynamic. Since it is changing all the time, all the managing, checking work need to be done in a new manner to keep up with the progressive surrounding. The traditional way of compliance checking can hardly realize it. Moreover, the compliant work also need to be updated regularly, having compliance personnel spent lots of time to conduct high-level testing of individual processes does not deliver the required enhanced assurance (Turner & Florio, 2004). Therefore we need the support of the ICT tools.

In order to check the conflicts between the governmental regulations and business processes with ICT tools, there are several steps that need to be followed. First of all, the governmental regulations and the business processes need to be transformed in a conceptual model. Then we can use this model as the input of the ICT analysis tool. This model is able to keep all the logical features in a structural way, which can be an interface between the natural language
and the ICT analysis tool. Secondly, a proper analysis tool needs to be present. This tool is able to describe the multi-actor system described in Chapter 1.

Many efforts have been taken in modeling governmental regulations. However, some of the representation was only focused on individual norms. For instance, (Herbst, 1996) structured the business rules by four main elements, namely Event, Condition, Then-action and Else-Action. Although he connects different business rules towards the final stage by using Petri Nets, the formal representative formalism of the interdependencies is still missing. Munindar P. Singh proposed to use commitments to capture normative concepts in multi-agent system and define norms as a tuple including subject, object, context, antecedent, and consequent (Singh, 2012). This approach provides an approach to characterize the bounds of autonomy and interdependence between agents, but the contextual aspects of norms are not considered.

Taking the process view in terms of business process modeling, a formalism called Process Compliance Language (PCL) was proposed by G. Governatori and A. Rotolo (Governatori & Rotolo, 2009). PCL expresses of violation conditions and obligations which is important for formalizing norms to determine the compliance of a process with the relevant norms. PCL enables to represent exceptions as well as to capture violations and the obligations resulting from the violations, but it does not take a broader view on norm sets where relationships other than reparation of violation exist between norms (Jiang, et al., 2012).

Another group of approaches is from the domain of normative multiagent systems. As presented by Felicissimo et al, models norms of multi-agent system has four levels of abstraction: Environment, Organization, Role and Interaction contexts. However, these contextual normative frameworks all concentrate on the effects of individual norms but ignore their relations (Felicissimo, et al., 2006). In order to regulate the behavior of agents in open and regulated multi-agent system in a distributed manner, (Gaertner, et al., 2007) presents a normative structure based on the propagation of normative positions as consequences of agents’ actions and provides a mapping into Colored Petri Nets for conflict checking. In Norm Nets which is used in this thesis, we approach the question with especially thinking of the interrelationships between norms.

Many analysis tools are currently are used to manage and analysis the workflow, e.g. WorkManager, Powerflow Echo .etc (van de Aalst, 1996). We choose Colored Petri Nets (Jensen & Kristensen, 2009) which is a graphical oriented language for design, specification, simulation and verification of systems. It is in particular well-suited for systems that consist of a number of processes which communicate and synchronize. (Boella, et al., 2013) argued that this type of process models are too general for use in legal settings. On the contrary, in (van de Aalst, 1996), Aalst argued three reasons to use Petri Nets for specification of workflow, namely 1) formal semantics despite the graphical nature; 2) State-based instead of event-based; 3) Abundance for analysis techniques.
As discussed in this section, many researchers are still working with different focuses in the domain of compliance checking, e.g. the contextual thinking, the PCL approach, the normative multiagent approach etc. In this thesis, we propose an approach by use of Norm Nets (Jiang, et al., 2012). The approach as the new framework is organized in such a way that they can be mapped to CPNs. The conflicts can be identified in CPNs with especially thinking of the interaction between different norms.

2.2. Conceptual model

Now let’s construct the conceptual model of this project. As we know it is not feasible to find the conflicts between governmental regulations and business process manually. We need the support of ICT and some tools to help us to analyze the whole process. However, the natural language text used in laws and business documents cannot be recognized by standard ICT tools. Both the governmental regulations and the business processes need to be processed before serving as inputs in the analysis tool. As shown in the Figure 3, by translating business processes and governmental regulations, the potential conflicts are finally analyzed using Colored Petri Nets. Because we believe the interrelationship among norms should be kept in the analysis, we represent laws by the use of Norm Nets. We believe the model with Norm Nets is a better approach to analysis the conflicts between the governmental regulations and the business process. Because the interrelationships which are ignored in previous researches are reflected in the formalism, this part of information is well stored in Norm Nets.

The arrows in the diagram reflect important relations between the concepts; the orange arrows indicate relations in scope of this research, the dashed arrows indicate the two types of conflicts (internal conflicts in regulations and the conflicts between governmental regulations and business processes). The remaining arrows are relations that are out of scope of the research (see section 1.3).
We explain the three orange arrows which represent the project scope in detail.

Firstly, the link between Norm Nets and Colored Petri Nets represents the translation work from Norm Nets to CPNs. It was previously done manually by the analysts themselves. We automatize this process in the project. A transformation tool is designed to fulfill this task, transforming content of Norm Nets to CPNs. This artifact is one of the key focuses in the project.

Secondly, the arrow from Colored Petri Nets to Conflicts stands for the compliance checking process. The analysis and the suggestions based on the result is offered to the company for improving their business process.

Last but not the least, the last arrow stands for the feedback step to the model of Norm Nets. This project is a validation for the model of Norm Nets. Through the real-world case we use in the case study, we intend to find whether the logical construction proposed in Norm Nets is sufficient to represent all the relations in governmental regulations. If problems were found
in representing some of the norms, we will further improve the model by either modifying the structure or adding new connections.

All the research questions are represented in the orange triangle in Figure 3. Sub research question 1 is included in the analysis of Norm Nets (the box “Norm Nets” in the conceptual model in Figure 3). We use the analysis result—the formalism of Norm Nets to show how Norm Nets represent the regulations. The arrow from Colored Petri Nets to conflicts in Figure 3 stands for the compliance checking process. The checking procedures answer the sub research question 2. The sub research question 3 is reflected in the analysis of the compliance checking result (the box “conflicts” in the conceptual model in Figure 3). Our recommendations are given according to the checking result. The box “Colored Petri Nets” is our analysis tool. The whole orange triangle defines the research scope of this project. The main research question can be answered by the analysis within this scope.

2.3. Core concepts

The conceptual model gives us an overview of the research domain and conceptual structure of the research. Next, we take a close look at every conceptual point one by one. All the main concepts used in this thesis are discussed in this part, including the discussion of different interpretations we referred from other literatures.

2.3.1. Governmental regulations modeling

The business-related governmental regulations in this thesis are typically phrased as “business rule” in other scientific literature. A business rule is a statement that aims to influence or guide behavior and information in an organization (Steinke & Colleen Nickolette, 2003). Just as the general governmental regulations, business rules can be categorized into (zur Muehlen, et al., 2007):

- Mandates: Published polices that must be followed, or consequences will ensure. The payment of tax belongs to this category.
- Policies: published policies that should be followed to adhere to company rules. Examples are budgets and mission statement.
- Guidelines: rules that may or may not apply, depending on circumstances. For instance, the management style and organization structure.

There are several ways to model business rules, as demonstrated in the following papers (M.Thorpe & C.Ke, 2001), (Demey, et al., 2002), (Herbst, 1996). Starting from Chapter 4, we will discuss in detail about how we model it with Norm Nets.

While talking about regulation modeling, we need to mention about the domain of normative system. Normative systems have been defined as sets of constraints (norms) on the behavior of agents in a system (Ågotnes, et al., 2007). In our conceptual model, it is related to the left branch (see Figure 3). The term “norm” stands for regulation in our case. “Norms prescribe
how the agents ought to behave in terms of obligations, permissions and prohibitions. Some examples of normative systems include corporate governance, legal and access-control systems, firewalls, and business application logic.” (Jones & Sergot, 1992) Norm (permissions, obligations and prohibitions) offers a useful abstraction with which to specify and present regulation system. Correspondingly, some researcher solve the compliance checking problem by using the normative multiagent approach (Vasconcelos, et al., 2009).

Next, we discuss the right branch in the conceptual model (see Figure 3), the business process part.

2.3.2. Business process management

The business process modeling is out of the research scope of this project, but in order to make our compliance checking process complete as an applicable approach, we need to transform the process into a particular form which can be recognized by our analysis tool (CPN tools). We discuss this concept which is on the right branch of the conceptual model (see Figure 3).

Business Process Modeling (BPM) in systems engineering is the activity of representing processes of an enterprise, so that the current process may be analyzed and improved. A business process is a collection of related, structured activities or tasks that produce a specific service or product (serve a particular goal) for a particular customer or customers. One important topic of business process modeling is the concept of norm compliance. Norm compliance aims at ensuring that business processes are in accordance with a set of norms. The complexities of business process modeling may come from 1) the specific obligatory actions may be ruled by the norms; 2) different types of obligations; 3) the different conditions we have for business process. (Governatori & Rotolo, 2009). In this thesis, we transform the business process into the form of event trace. We treat a process as a group of sequential events and compare them with the regulations in CPN tools. This will be discussed in detail in chapter 7.

Business processes are sets of activities that create value for a customer (Hammer & Champy, 2009). The research in business process management initially focused on the documentation and organizational governance of processes. Because of the complexity of the domain and the ICT revolution, more and more organizations are automating such processes using workflow systems and build management system around the system (zur Muehlen, et al., 2007). The Workflow Management Coalition (WfMC) defines workflow as “the automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules.” (P. Lawrence, 1997)

A number of modeling languages are used for modeling the processes, like process languages as BPMN, EPC, Net-based languages as Petri-nets, flow nets) and Workflow Programming Language as BPEL. (zur Muehlen, et al., 2007)
In this thesis, we choose Colored Petri Nets (a variation of Petri Nets) to model the relevant business rules. It is a graphical oriented language for design, specification, simulation and verification of systems. It is in particular well-suited for systems that consist of a number of processes which communicate and synchronize.

2.3.3. Deontic logic

Deontic logic is the field of logic that is concerned with obligation, permission, prohibition and related concepts. Alternatively, deontic logic is a formal system that attempts to capture the essential logical features of these concepts (Amini, 2010). Since the publication of von Wright’s paper ‘deontic logic’ in 1951 (Wright, 1951), the deontic logic has been development rapidly in many research fields. Currently, deontic logic is widely used in the field of AI and Law. For deontic logic, we have several definitions:

- Wide definition: Deontic logic is a symbolic logic concerned with the logic of normative expressions: a systematic study of the contribution these expressions make to what follows from what. (Amini, 2010)
- Narrow definition: Deontic logic is the logic of obligation, permission, and prohibition. (Amini, 2010)
- Precise definition: Deontic logic is that branch of symbolic logic that has been the most concerned with the contribution that the following notions make to what follows from what. (Amini, 2010)

As (Jones & Sergot, 1992) argued, there is no clear consensus about the role of deontic logic in legal knowledge representation. In this thesis, we add deontic type to Norm as one of its attribution to form a structure for a single article (see 4.3).

2.3.4. Norm Nets

Norm Nets is developed as a formalism to capture the structure of logical relations existing in the governmental regulations. Norm Nets provide rich expressiveness for single regulations based on formal normative system theories, and also capture the possible relations between different regulations in a modular way (Jiang, et al., 2013).

A norm is defined as a tuple \( n = (D, p, \delta, \sigma) \) where:

- \( D \) indicates the deontic type of the norm, i.e., Obliged, Forbidden, and Permitted;
- \( p \) describing the target to which the deontic is assigned;
- \( \delta \) describing the deadline;
- \( \sigma \) describing the precondition; (Jiang, et al., 2013)

The four components of the norm, target, deadline, precondition and deontic type is corresponding to a clause in a law. For example, we have such a clause “customers are not allowed to return the goods after 14 days after they received the goods”. We could formulate
the corresponding norm where the target is “the customers returned the goods”, precondition is “the customers have received the goods”, the deadline is “14 days since they received the goods”, and the deontic type is “forbidden”.

In the model of Norm Nets, Norm is composed of formulas which can either be a single role-action pair or sets of role-action pairs connected by And, Or and Before. Rap is short for “Role action pair”, which basically describes who does what. As its name describes, the two vital elements “Role” and “action” are defined in a single Rap. Raps connected by And, Or, Before are able to describe a complex situation where several Raps with interrelationships take place together.

A Norm Net can be a single norm or a nested structure composed of norms, which are connected by the three operators AND, OR, and OE (Or Else) in a certain context. (Jiang, et al., 2012). The interrelations between norms are reflected by these operators. By using different operators and different ways of constructing norm structure, the user could model complex regulations with interrelations.

More details about Norm Nets will be discussed in chapter 4.

2.3.5. Colored Petri Nets

Colored Petri Nets (Jensen & Kristensen, 2009) is a graphical oriented language for design, specification, simulation and verification of systems. It is in particular well-suited for systems that consist of a number of processes which communicate and synchronize. Typical examples of application areas are communication protocols, distributed systems, automated production systems, and work flow analysis.

In (van de Aalst, 1996), Aalst argued three reasons to use Petri Nets for specification of workflow. They are:

1) The business logic can be represented by a formal but also graphical language. The semantics of the classical Petri Nets and several variation, e.g. Colored Petri Nets have been defined formally.

2) In contrast with many other process modeling techniques, the state of a case can be modeled explicitly in a Petri Nets. Petri Nets is a state-based modeling technique instead of event-based one.

3) Abundance of analysis techniques.

By translating NNs to Colored Petri Nets (CPNs), we could use the standard features and tools of CPNs to realize compliance checking process. The graphical interface can show us the conflict result clearly. For analysis, it is relatively easy to find the reason of the conflicts by directly tracking the violation token (see 7.2) on the CPN diagram. These are the main reasons why we want to transform our data from Norm Nets into CPNs for further analysis.
2.3.6. Meta model

A model is an immaterial representation of a relevant part of the real world. It is created for
the purpose of a subject. It relates two systems, object system and model system. (Rosemann
& Muehlen, 1998). The object system represents the subjective interpretation of a selected
part of the real world including the relevant part of the environment. The model system
represents the subjective image of the object system. The complexity of the real world, which
is cause by all kinds of real world elements and relationships, is reflected by a cluster of
elements and relationships in a model.

The meta model defines a model structure. From a model’s perspective, a model is a meta
model’s instance. (Eysholdt, 2009) It can also be called as the model of the model. If a model
system M1 represents the object system of a model M2, then the model system M2
represents the meta model system of the object system which M1 is based upon (see Figure
4, object system is M0).

A meta model can be seen as "a design framework, that describes the basic model elements
and the relationships between the model elements as well as their semantic. The framework
also defines rules for using and specialization of model elements and relationship." (Rosemann & Muehlen, 1998) Figure 4 shows the meta level.

M3: the meta meta model layer; describes concepts that appears in the meta model.

M2: the meta model layer; describes concepts that make up a modeling language.

M1: the model layer; describes the object in the real world.

M0: the instance layer; the objects.

![Figure 4 Meta level (Eysholdt, 2009)](image)

Software development is a complex and difficult task that requires the investment of
significant resources and carries major risk of failure. Among all the software development
methodologies, Model-driven engineering (MDE) is considered to be a feasible approach
which could increase developer productivity and improve software reusability and makes
software more maintainable (Liddle, 2011). In this thesis, we build up a meta model of Norm
Nets and then connect this meta model to the meta model of CPNs. In this way, we could design our desired functions in the transformation tool without spending too much time on the functional modules which may have already existed in the model.

We built up a meta model by using Eclipse Modeling Frame (EMF) (The eclipse foundation, 2013). EMF Ecore is the language to formally define the data structure. In chapter 4, we will show the meta model of Norm Nets and how it can generate other models of a meta model for the specific case.

2.4. Conclusion

In this chapter, we reviewed the relevant literature in the research domain. The literature was discussed in such a way that several arguments/approaches for the same point were grouped and compared with our approach. And all the points together were linked to show the overview of the development in the domain. Then the conceptual model was presented. This diagram explained what our compliance checking approach is, and what are we focused on in this thesis. Moreover, the diagram also structures a framework for the next chapters. We will implement the analysis step by step according to this diagram. In the end, we discussed all the core concepts used in this thesis, building up a solid conceptual foundation for the further analysis in the following chapters.

In the next chapter, we define our software design specification for the design artifact in this project, the transformation tool.
3. Software design specification

The design specification of the transformation tool is defined in this chapter. The specification offers the instruction and requirements for the design work, describe what the designed artifact should be. We construct this chapter into three parts. The first part is the design process. In this part we illustrate the procedure we followed to design the transformation tool. The second part is the design requirements, including functional requirements and other requirements, where we define all the requirements for the software. Finally we define the input and output of the software.

3.1. The design process

There are four steps in this design project. Starting from Chapter 4, the detailed implementation of each step will be shown one by one as the procedure below.

1) Decomposing of the model of Norm Nets. In this step we do an analysis on the model of Norm Nets and build up its meta model.

2) Expressing Norm Nets by CPNs components. In this step, we find the corresponding relationships between models of Norm Nets and CPNs. We build up the CPN module for every Norm Nets components in its Meta model.

3) Automatizing the process of step 2. We develop the transformation tool to automatize the transformation process.

4) Validating the model and the transform tool through a case study. The whole approach is validated through a case study. Based on the result, we can give suggestions or refine the model as well as the transformation tool.

3.2. Design requirements

The design for the transformation tool used to transform Norm Nets to CPNs is one of the main deliverables of this project. Before defining the design requirements, we make an important assumption. We assume the users of the transformation tool have the basic knowledge of Norm Nets. They should at least know the structure of Norm Nets and the definition of all its components, so they should know how to translate the content from natural language into the form of Norm Nets. In other words, as what is shown in the conceptual model, the transforming work from natural language in regulations to Norm Nets is not the task for the designed artifact but for the users.

3.2.1. Functional requirements

In software engineering, a functional requirement defines a function of a system or its component. In Table 2, we summarize the functional requirements of the transformation tool.
### Functional requirements

<table>
<thead>
<tr>
<th><strong>Functional requirements</strong></th>
<th><strong>Content</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input function</strong></td>
<td>Users are allowed to feed the information as input the information in a Java based interface. The user interface support the function 1) choosing any Norm Nets element, 2) typing in the relevant content, 3) making connection between elements by the defined logical operator in order to keep all the information in Norm Nets.</td>
</tr>
<tr>
<td><strong>Transformation function</strong></td>
<td>The transformation tool is able to transform the regulations from Norm Nets to CPNs.</td>
</tr>
<tr>
<td><strong>Output function</strong></td>
<td>Output an XML file. The file contains the entire information of Norm Net and is able to input to CPN tools for the further analysis.</td>
</tr>
</tbody>
</table>

*Table 2 Functional requirements of the transformation tool*

3.2.2. **Platform**

Java is the platform for the transformation tool. Specifically, Eclipse is the software development platform for the program of transformation process. Eclipse Modeling Frame (EMF) is the development tool for the meta model of Norm Nets as well as the user interface.

3.2.3. **Extensibility**

Extensibility means that new capabilities can be added to the software without major changes to the underlying architecture. This is especially important because Norm Nets itself still need to be verified. It is quite possible that some new elements need to be added in. High extensibility allows us to relatively easily add new elements in the model.

The extensibility is mainly realized through the design approach of Model-Driven Engineering.

3.2.4. **Maintainability**

Maintainability is a measure of how easily bug fixes or functional modifications can be accomplished. High maintainability can be the product of modularity and extensibility. Maintainability in this project benefits from using more pre-defined methods in Java environment. In the future the software can be easily updated by modifying the parameters in the methods.

3.3. **Input and output**

The input to the transformation tool is all the content input from the user interface. Originally the information is in the form of Norm Nets, such as roles, actions, Raps (Role-action pairs), formulas, norms, logical operators and deontic types.
The main function of the transformation tool is to transform the content in the form of Norm Nets to the form of CPN. The information of every petri net is stored in XML in CPN tools. Therefore the function of the software is to generate an XML file which could be read by CPN tools and contained all the information in the Norm Nets model. The output is a file which could be expressed in CPN tools.

3.4. Reflection

To design the transformation tool, we demonstrate the design specification in this chapter. The design specification provides explicitly the requirements for the software. Designing the IT artifact is one of the main objectives in Part 1. In this chapter, we first presented the design process. Then we defined the functional requirements and other requirements of the transformation tool. At last, we defined the input and output.

From the next chapter until the end of Part 1, we did a research to design the transformation tool. We analyze the structure of Norm Nets and then the CPN representation of Norm Net. The design specification described in this chapter is implemented in the process of programing the transformation tool. We obtain the final checking result by use of the transformation tool. However, the detailed program framework and the code will not show in the thesis.
4. Analyzing Norm Nets

In order to make the connection between the model of Norm Nets and CPNs, we first need to decompose the Norm Nets model and clarify its structure. This is also the requirement of building the meta model of Norm Nets. Therefore before we start programming, we perform a detailed analysis on the model of Norm Nets in this chapter. The model can be decomposed into four layers: Rap, Formula, Norm and Norm Nets. They are explained one by one in this chapter. The decomposition result of Norm Nets is shown in the end in a meta model diagram. The diagram is generated by EMF (The eclipse foundation, 2013).

To explain the abstract concepts, we made an example which includes three interrelated regulations.

1) The committee is obliged to certify an applicant if an applicant has two referrers and holds a certification. To get the certification, an applicant can either get an old version certification validated or get a new one. To get a new certification, the applicant must join in the training program. Applicants who pass the training program could get the new certification.

2) The training can be passed by passing the exam within three months after the training finished; however, if the applicant did not pass the exam within three months, the applicant can still pass it by handing in an assignment and get a pass for the assignment within 2 month since the last exam.

3) An applicant is forbidden to plagiarize in the assignment, otherwise, the applicant fails the whole certification and forbidden to apply for the same certification within two years.

Next we use the example to introduce the four layers of the Norm Net model.

4.1. Role-Action Pair

A Role-Action pair (Rap) is the basic component in the whole model of Norm Nets. It is atomic element of the Norm Net model and used as building blocks to create the other structures. As its name tells, a Rap is composed of two elements, namely Role and Action.

**Definition 4.1:** Rap=\( (r, a) \), where “r” indicate an atomic role can be assigned to agents, “a” indicate an atomic action that agents can perform. (Jiang, et al., 2013).

A Rap represents a simple case of “who does what”. The action “a” is available for the role “r” and any actors enacting the role might perform that action. For instance, “customs releases the goods” can be formulated in Rap as (customs, releasing_the_goods). The roles and actions are, on this level of modelling, merely a label for both elements. Both of the Role and the Action are of the type of string in the meta model.

Take the third regulation in the certification example, we find several Raps in it.
An applicant is forbidden to plagiarize in the assignment (applicant, plagiarize in the assignment), otherwise, the applicant fails the whole certification (applicant, fail to get the certification) and forbidden to apply for the same certification (applicant, get a new certification) within two years.

We could find 15 Raps in the example above (see table 3).

<table>
<thead>
<tr>
<th>Rap No.</th>
<th>Role</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>committee</td>
<td>Issue certification to an applicant</td>
</tr>
<tr>
<td>2</td>
<td>applicant</td>
<td>Have two referrers</td>
</tr>
<tr>
<td>3</td>
<td>applicant</td>
<td>Hold a certification</td>
</tr>
<tr>
<td>4</td>
<td>applicant</td>
<td>Get the old certification validated</td>
</tr>
<tr>
<td>5</td>
<td>applicant</td>
<td>Get a new certification</td>
</tr>
<tr>
<td>6</td>
<td>applicant</td>
<td>Join a training program</td>
</tr>
<tr>
<td>7</td>
<td>applicant</td>
<td>Pass the exam</td>
</tr>
<tr>
<td>8</td>
<td>applicant</td>
<td>Hand in an assignment</td>
</tr>
<tr>
<td>9</td>
<td>applicant</td>
<td>Pass the assignment</td>
</tr>
<tr>
<td>10</td>
<td>applicant</td>
<td>Fail the assignment</td>
</tr>
<tr>
<td>11</td>
<td>applicant</td>
<td>Plagiarize in the assignment</td>
</tr>
<tr>
<td>12</td>
<td>applicant</td>
<td>Fail to get the certification</td>
</tr>
<tr>
<td>13</td>
<td>time</td>
<td>Pass three months since the end of the course</td>
</tr>
<tr>
<td>14</td>
<td>time</td>
<td>Pass two months since the last exam</td>
</tr>
<tr>
<td>15</td>
<td>time</td>
<td>Pass two years since the application date</td>
</tr>
</tbody>
</table>

*Table 3 Raps in the example*

### 4.2. Formula

The formula is based on Rap but can be more complex in terms of the structure and the number of components. In the meta model, in the meta model, Formula is the super type of Rap, that is, every Rap is also a Formula. Moreover a formula can also be a set of formulas connected by four logical operators, Before (<), Or (∨), And (∧) and Not (¬). In the example which is shown in the beginning of this chapter, all the Raps listed in Table3 are formulas. Some of the formulas could be connected with the logical operators and together express a compound meaning. For example, according to the given regulations, to get a certification, an applicant can either validate an old certification (Rap 2) or get a new one (Rap 3). Therefore we say Rap2 ∨ Rap3 (i.e. get the old certification validated or get a new certification), together is the prerequisite of the formula “getting a new certification” as interpreted in Regulation 1.

**Definition 4.2**: Formula::=\text{Rap} | \text{Before} (\text{Left}, \text{Right}) | \text{Or} (\text{Left}, \text{Right}) | \text{And} (\text{Left}, \text{Right}) | \text{Not} (\text{left}, \text{Right})
Left and Right components are the type of formula.

In the meta model, the two components of a formula on both sides of the logical operator are named “Left” and “Right”, respectively. Before means the left formula occurs earlier than the right formula. And means the right and left formula both occur. Or means that either the left component or the right component takes place. Not means the formula does not happen.

<table>
<thead>
<tr>
<th>Logical relationship</th>
<th>The example in the case</th>
<th>Formula components</th>
<th>The expression in Norm Net</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; (Before)</td>
<td>The applicant join the training before passing the exam</td>
<td>Left: Join the training Right: pass the exams</td>
<td>Rap 6&lt; Rap7</td>
</tr>
<tr>
<td>∨ (Or)</td>
<td>Validate an old certification or get a new certification</td>
<td>Left: validate an old certification Right: get a new certification</td>
<td>Rap4 ∨ Rap5</td>
</tr>
<tr>
<td>∧ (And)</td>
<td>Have two referrers and hold a certification</td>
<td>Left: have two referrers Right: hold a certification</td>
<td>Rap2 ∧ Rap3</td>
</tr>
<tr>
<td>¬ (Not)</td>
<td>Not pass the exam</td>
<td>Pass the exam</td>
<td>¬ Rap7</td>
</tr>
</tbody>
</table>

Table 4 The instances of formulas in the example

Note that both the left and right components are still of the type of formula which means they can be either a single Rap or a combination of Raps linked with the logical operators. Therefore formula can actually be complex in terms of the number of components. Theoretically, the number of nested formulas of a formula can be infinite, as every "Left" and "Right" component can be another formula which contains new components. For the analysis in this thesis, we assume that 1) the number of nested formulas is always finite, and 2) that formulas will not contain itself as a nested formula.

4.3. Norm

Either single Raps or complex Formulas contain several other Formulas, they state a matter of fact without judgment in terms of compliance or violation. While for one complete article in a law, it normally rules explicitly what is obliged to do and what is forbidden to do. It is not only a combination of roles and actions, but also involving in conditions of compliance or violation. Correspondingly, we set norm in the meta model.

Definition 4.3: A norm is defined as a tuple n = (D, ρ, δ, σ) where:

- D ∈ {O;F;P} indicates the deontic type of the norm, i.e., Obliged, Forbidden, and Permitted;
- ρ ∈ Rap, describing the target to which the deontic is assigned;
- δ ∈ Formula, describing the deadline;
We model norms with a temporal nature. They are not always active, and have to be achieved/fulfilled under a particular precondition and within a particular time. In a norm, the precondition determines whether the target should be initiated, and the deadline determines when the target has to be ensured. For instance, if the deadline has passed when the obligation action was not realized while the deontic type is obliged, the violation occurs. The violation also happens while the forbidden action is done when deontic type is forbidden. The three elements, a target is a rap; precondition, deadline are of the type of formula.

Returning to the earlier example, we can identify two instances of norms. These are $n_1 = (\text{obliged, Rap 5, Rap 13, Rap 7})$, which describes the regulation that to get a new certification, the applicant is obliged to pass the exam within three months. $n_2 = (\text{obliged, Rap 5, Rap 14, Rap 9})$, which describes the regulation that to get a new certification, the applicant is obliged to pass the assignment within two months since the last exam.

4.4. Norm Nets

There are always a number of articles instead of a unique one in a law. These articles are interrelated to each other. Reflecting on the meta model of Norm Nets, norms also need to come with relations. We need to model these relations. Hence we have another structure element, Norm Nets. A Norm Net can be a single Norm or a nested structure composed of Norm connected by And, OR and OE. In the meta model, Norm Net is the super type of Norm. We name the two components (Norm Net) which are in two sides of the operator “Left” and “Right”.

**Definition 4.4:** Definition 4. Norm Net::= n| AND (Left, right)| OR(Left, Right)| OE(Left, Right))

Left and Right components are the type of norm net.

We have three connectors in Norm Nets. The first one is AND, meaning that compliance of this Norm Net requires both of the Left and Right components to be complied. When even only one of the left or right norm net is violated the AND is violated. The second one is OR, meaning that the complied situation can be achieved by doing either of the two norm nets, and the violation will only occur when both of the two norm nets are violated; the last one is OE (Or Else). The activation of the Right component in this relation has a prerequisite which is the violation of its Left component (Else). The Right component will not be activated until the left one (Or) is violated. Only the violation of both of the two norms will lead to a violation of the whole structure. Complying with either of the two norms is considered to be a complied.

A regulative norm net is supposed to represent all of the regulative articles, and can therefore be either a single norm or a set of norm nets related via AND, OR and OE. Until here we construct the meta model of Norm Nets. Figure 5 shows the diagram of this meta model. All the components and logical constructs are shown in this diagram.
4.5. Conclusion

In this chapter we decomposed the model of Norm Nets, analyzing all the relevant components and connections. The four layers “Rap”, “formula”, “Norm”, “Norm Net” are explained individually. And in the end we build up the meta model of Norm Nets.

After making the meta model of Norm Net, we generate an editor of the meta model through EMF (The eclipse foundation, 2013). In the editor, the users of the transformation tool could design their customized Norm Net model. In other words, this meta model is the structure of the user input interface in the transformation tool. This part is the basis of the software design. Our objective of the design part is to transform the information from Norm Nets to CPNs. The analysis in this chapter makes Norm Net part clear and structured. In the next chapter we introduce CPN tools, and show how elements of a Norm Net are translated into CPN structures.

Through the analysis on the model of Norm Nets in this chapter, we know how the regulations are represented in the model. We can now answer the sub research question 1.
**Sub-question 1: How do Norm Nets represent the structural elements in governmental regulations?**

The formalism of Norm Net we discussed in this chapter is the answer for this question. Let us explain why.

The typical normative vocabularies used in laws and regulations are “must”, “ought”, “may”, “should”, “shall”, etc. (Susskind, 1987) We reflect these words in Norm by the deontic type; permission, prohibition and obligation. Generally speaking, words like “may”, “can” correspond to the deontic type of permission; words as “must”, “ought”, “should”, “shall” correspond to the deontic type of obligation; and the words as “forbidden”, “not allowed” are the typical signs for the deontic type of prohibition. However, we do not mean the deontic type and the vocabularies are strictly one to one correspondence relationship. Words like “must”, “ought”, “allowed” can have a variety of meanings, and their occurrence in legislative text is not in itself sufficient reason for determining a deontic type of a norm. Besides the semantics analysis, sometimes we need to consult the experts to get the correct interpretation.

Besides those vocabularies, the other elements of an article of law like applied condition, deadline, and the behavior which is ruled by this article also have their corresponding components in a Norm, namely precondition, deadline and target, respectively. Moreover, we designed formulas to represent the complex combinations within a norm component. For example, “if the student passed the exam or handed in the assignment and get sufficient grade before the teacher uploads the grade, the teacher is obliged to give the student a pass”, the precondition part is more complex than a single action; we need operators And, Or, Before to construct a composed structure to represent it in Norm Nets, as (student, pass exam) Or (((student, hand in assignment) And (student, get sufficient grade)) before (teacher, upload grade)). The Role-Action Pairs and formulas make it possible to describe these kinds of composed relationship.

Additionally, Norm Nets capture the relationships between different legal articles by using the structure of AND, OR and OE (or-else). These operators represent different meanings (see 4.5), thus they connect different articles in an interrelated net. This net could represent a set of interrelated governmental regulations.

Therefore all the logical features in regulations are captured by the formalism of Norm Nets in this way as discussed above.
5. Creating CPN

From the analysis in Chapter 4, we have already built up a Meta model for Norm Nets. This meta model gives us a view on the framework of Norm Nets. In this chapter, we investigate the relationship between the Meta model of Norm Nets and the CPN model. This is the step from Norm Nets to Colored Petri Nets in the conceptual model (see Figure 3). In this step, we create the CPN construction for every single Norm Nets structure. This is a model-to-model approach. It means our coding did not focus on realizing the functions, but making connection on the model level.

In this chapter, we first introduce CPN. For the convenience of describing a CPN model of Norm Nets, besides the four basic elements existing in CPN model, we define a pattern and four other elements (section 5.1). Next we show our interpretation of Norm Nets elements by using CPN components (section 5.2- section 5.5). Once we defined the transformation for each of the individual elements of Norm Net, we can form a standard transformation process from Norm Nets to CPN. Then we are able to use those defined transformation to build up the Norm Nets model in CPN Tools¹ for our case study and generalize to any other use cases. In the end of this chapter, we conclude and reflect the transformation process from Norm Nets to CPNs.

5.1 Introduction of CPN

In CPN model, there are four basic elements, Transition, Place, Arc and Token. The first three elements are our focuses in the transformation process from Norm Nets to CPN as we are intend to automatize this transformation by designing a transformation tool. While for tokens, it is not a part of the automatic transformation tool. In the case study, we added tokens manually, because adding token is related to the business process which is out of the scope of this project.

In Chapter 2, we introduced the background of CPNs. Here in this section we discuss its basic elements and operation mechanism.

Arrows connect the places and transitions. Places can contain tokens; the current state of the modeled system (the marking) is given by the number of tokens in each place. Transitions model activities which can occur (the transition fires), thus changing the state of the system (the marking of the CPN). Transitions are only allowed to fire if they are enabled, which means that all the preconditions for the activity must be fulfilled (there are enough tokens available in the input places). When the transition fires, it removes tokens from its input places and adds some at all of its output places. (Westergaard & Verbeek, 2013)

We conclude the operation mechanism of CPN model as below:

---

¹ CPN Tools is a tool for editing, simulating, and analyzing Colored Petri nets. It is the analysis tool in this project for analyzing the compliance result.
a) Connections are directed.

b) No connections between two places or two transitions.

c) Places may hold zero or more tokens.

d) A transition is enabled if each of its input places contains at least one token.

In Figure 6, a graphical representation of a CPN is shown (taken from the CPN Tools software). In this representation the transition is represented as a rectangle with a name, a place is a cycle or ellipse. A place could hold zero or more the green dots which represent tokens in it. Places and Tokens are marked by predefined colors. Colors are used to identify different roles. Only the token with the same color as of the place can transfer in/out that place. An Arc is a directional arrow in CPN tools.

![Figure 6 Elements in a colored petri nets](image)

We discussed the four basic elements already existed in CPN. In the transformation process, we find that some elements appearing in particular position of CPN model play import role in the structure and are quite often used. For the convenience of description of the transformation in the following sections, we self-define a pattern and four extra elements related to the pattern.

A pattern is a construction of the CPNs elements. We define here a pattern starts and ends both with places. It represents a part or the whole CPN modeled system. The four important elements in a pattern are the interface of this construction block of Pattern to the external structures, controlling the input and output flow go in/out the pattern. We define these elements with particular function in a pattern and list them as below.

- **The First Transitions**: the first transitions of a whole module. They are connected to the first Places.

- **The Last Transitions**: the last transitions in a whole module. They are connected to the Last Places.

- **The First Places**: the input points of a pattern. They are also the first elements in a pattern.
• **The Last Places**: the output points of a pattern. They are also the last elements in a pattern.

A pattern and its related elements are shown in Figure 7.

![The four self-defined basic elements in the model](image)

**Figure 7** The four self-defined basic elements in the model

### 5.2 The Role-Action Pair in CPN

As the base layer in the whole model structure, Role-Action Pair (Rap) is the basic element we use to build up the whole model. In the next sections we will show how to use Rap as an atom to build up other more complex structures.

Rap describes “who does what”. The role (who) is reflected in CPN model (Figure 8) by the color of the places. The action (what) is reflected by the name of the transition. A token passing through the structure means the action in the Rap is realized by the role, and vice versa.

![Rap in CPN](image)

**Figure 8** Rap in CPN

The CPN representation for different Rap scenarios is illustrated in Table 5. If an action was taken by the role, then the token will move to the last place, and vise versa. We use CPN behavior to reflect the scenarios in reality.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>CPN representation</th>
</tr>
</thead>
<tbody>
<tr>
<td>The action was taken by the role</td>
<td>Token in the last place</td>
</tr>
<tr>
<td>The action was not taken by the role</td>
<td>Token not in the last place</td>
</tr>
</tbody>
</table>

*Table 5* CPN representation result of Rap

The process of building up one Rap in CPN is as below:
1) Create two places and one translation;
2) Connect the transition with the two places by two arcs;
3) Name the translation with the action name “Rap. Action. name” ;
4) Set the color of the First and Last places role name “Rap. Role. name”.

5.3 The Formula in CPN Tools

A single Rap is a representation of whether a role is able to do an action, while in the real world the modeled case can be more complex. More roles and their actions, especially the interrelationship among different Raps need to be considered in a network. Therefore we introduce formula and their module in CPN model.

As discussed in Chapter 4, a formula can be either a single Rap of a set of related Raps. A formula which consists of only one Rap is treated just as a single Rap in CPNs (see 5.2), while the others are structured as below.

5.3.1 RapAnd

RapAnd means two formulas are connected by And. The composed action of the structure can only be done by finishing doing both of the actions of the two formulas which the RapAnd structure contains.

By applying the operation mechanism d) of CPNs, the last transition of RapAnd will be enabled only if all of its input places hold at least one token. In CPN model of RapAnd (Figure 9), token appearing in the last places of its left and right branches means both of the two corresponding action are finished. Only in this case, can the tokens move to the last place of RapAnd which means the composed action is performed. This mechanism reflects the logical meaning of operator “And” in Norm Nets. Therefore the structure of RapAnd is realized in CPNs in this way.

The process of building up a RapAnd structure in CPN tools is as below:

1) Create two formulas corresponding to the right and left element of “RapAnd”,
2) Create one extra transition and one extra place.
3) Connect the last places of the two formulas to the created transition with Arcs and then connect this transition to the extra place.
4) Name the extra transition “auto+left.ID+right.ID”.

---

2 In the meta model we created, Action is one reference of Rap, and name is one reference of Action. Therefore here we mean that we use the name of action in the Rap to name the transition. In the following sections, we will use the same representation to refer the elements in the meta model of Norm Nets.
5) Get the color of the last place of the right element (colorR), get the color of the last place of the left element (colorL). Set the color of the last place of RapAND with the color union of colorR and ColorL.

![Diagram of RapAND in CPN](image)

**Figure 9 Formula And in CPN tools**

A comparison between different RapAnd values and the corresponding CPN results is shown in Table 6. The Boolean value in the table illustrates whether the formula is performed or not. Value 0 means that the formula is not performed completely, while value 1 means the formula has been performed. Different value combinations of left formula and right formula show all the possible scenarios of a RapAnd. Correspondingly, in its CPN model, these scenarios are reflected by the token in the last places of left and right formulas. From the table we can see that only both of the left and right formulas are performed can make the value of RapAnd to be 1 (logical meaning). In such situation, the token in the CPN model will arrive in the last place of RapAnd (CPN representation). The other three scenarios all have a Boolean value 0 as result (logical meaning). In CPN model, token will not move to the last place (CPN representation).

Taking the third row in table 6 as an example, we show how to find the correspondence relationship between the Expected Boolean value of RapAnd and the CPN representation. The value of left formula is 0. That means the action included in left formula is not performed. The right formula is 1. That means the action included in right formula is performed. Because these two parts are connected with a logical operator And, we can get the result of the whole structure is 0. Correspondingly in CPN, the action of left formula not being performed makes the token does not move to the last place of left formula (see table 6). The right formula behaves just oppositely —the token move to the last place of right formula. The auto transition does not fire because one of its input place is still empty. Then no token moves to the last place of this structure. The logical meaning is coincident with the CPN result. This is what shows in the third row of Table 6. Other scenarios can get the correspondence relation shown in table 6 in the same way.

---

3 Color union means the union of two colors. In CPN tools, the places with a color union can transfer tokens of both colors which belongs to the color union.
5.3.2 RapOr

RapOr means two formulas are connected by Or. The composed action of this structure could be finished by doing either of the actions of the left and right elements which RapOr contains.

According to the CPNs operation mechanism d), in the CPN model of RapOr (Figure 10) accepting a token from either of its left or right formula will transfer a token to the place in the joining place. This token could enable the following transition and in the end arrive at the last place of the structure, which mean, this RapOr has been performed. Finishing either of the two formulas or both of them lead to the finishing result of the whole structure. This mechanism is exactly the logical meaning of operator “Or” in Norm Nets. The component of RapOr is realized in CPN in this way.

The process of building up a RapOr structure in CPN tools is as below:

1) Create two formulas corresponding to the left and right elements of “RapOr”, create two extra places and three extra transitions.

2) Connect the output place of the two formulas to the two extra transitions, respectively.

3) Connect the two transitions to one place and then connect the transition and the last place in serial order.

4) Name the last transition of the structure “auto+ leftFormula.ID+ rightFormula.ID”.

5) Set the color of the output last place “left Formula. role+ right Formula. role”.

A comparison between different RapOr values and the corresponding CPN representations are shown in Table 7. The Boolean value in the table illustrates whether the formula is performed or not. The same as what we did for RapAnd, we can find the correspondence
relationship between the logical meaning of RapOr and its CPN result. From the table we can see that either the left or right formula is performed can make the value of RapOr to be 1 (logical meaning). In such situation, the token in the CPN model will arrive in the last place of RapOr (CPN representation). Only if the values from both of the left and right formula are equal to 0, then the result in the value of the structure is to be 0. Correspondingly, token will not move to the last place in the CPN model.

<table>
<thead>
<tr>
<th>Boolean value of Left formula</th>
<th>Boolean value of right formula</th>
<th>Expected Boolean value of RapOr</th>
<th>CPN result</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Token not in the last place</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>Token in the last place</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>Token in the last place</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Token in the last place</td>
</tr>
</tbody>
</table>

*Table 7 Comparison table of RapOr*

5.3.3 RapBefore

RapBefore represents a relationship that one formula happens before the other formula. We could also understand the structure in this way: one formula (right Formula) has a precondition formula (left Formula). The right Formula can only be activated after the left Formula finished. It is a structure designed to model two events in order.

As described above, the left formula could be seen as a precondition of the right formula. The left formula determines the enable condition of the right formula. However, it does not change the role and the action of the right Formula. Therefore the color and the transition’s name of the right formula do not change. According to the rule of CPNs, we know the right formula will not be enabled until all the input places hold at least one token. We also know that a token in the last place of the left formula means the action in the left formula has already finished. Therefore the right formula has to be processed after the left Formula. And this is exactly the meaning of “before” in Norm Nets.

The process of building up a RapBefore structure in CPN tools is as below:

1) Create two formulas corresponding to the left and right elements of “Before”.

2) Connect the last places of left Formula to the first transitions of right Formula.

*Figure 11 Formula Before in CPN tools*
A comparison between different RapBefore values and the corresponding CPN representations are shown in Table 8. The same as what we did for RapAnd, we can find the correspondence relationship between the logical meaning of RapBefore and its CPN result. This table is similar to the table of RapAnd. The difference is in its last two rows. Not only the value, but also the order of the occurrence of the two formulas determines the result. Only when the left and right formulas occur and the left formula occurs before the right one can make the result value to be 1 (logical meaning). In this situation, the right formula is fired. The CPN model responds by moving one token to the last place of the structure (CPN representation). All the other four scenarios have the result value 0, correspondingly no token arrives at the last place (see table 8, the order of the occurrence are marked by “first” and “second” in the bracket).

<table>
<thead>
<tr>
<th>Boolean value of Left formula</th>
<th>Boolean value of right formula</th>
<th>Expected Boolean value of RapBefore</th>
<th>CPN result</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Token not in the last place</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>Token not in the last place</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>Token not in the last place</td>
</tr>
<tr>
<td>1 (second)</td>
<td>1 (first)</td>
<td>0</td>
<td>Token not in the last place</td>
</tr>
<tr>
<td>1 (first)</td>
<td>1 (second)</td>
<td>1</td>
<td>Token in the last place</td>
</tr>
</tbody>
</table>

Table 8 Comparison table of RapBefore

5.3.4 RapNot

RapNot represents a relationship that one formula does not occur. It is a new element we proposed in this thesis compared to the original version ([Jiang, et al., 2012]). This relationship is necessary in Norm Net because we found some instances in the regulations. Additionally, and, or, not are the basic operators in logistics. Unlike other relationships, the representation of RapNot by CPN is not only determined by the logical relationship (Not), but also effected by its position in a Norm (e.g. RapNot in target and RapNot in precondition can have different CPN representations). The relevant representation is still need further analysis and coordination with other components in Norm Nets. Therefore for RapNot, we only propose it in this thesis but we will not discuss about its representation in CPNs. We leave this discussion open to the future research (see section 9.2).

5.4 From Norm to CPN

A Norm is a structure which contains a deontic type, target, deadline and precondition. In this section we first create a basic CPN structure for a generic Norm without consideration of the deontic type. Then we set different parameters for the output places according to the deontic type.

5.4.1 Create a generic Norm without the deontic type

Without considering the deontic type, a norm consists of three other elements: precondition, target and deadline. The target is in the form of Rap in CPN tools, and the other two are of the formula type. Therefore this step is relatively simple-- create one Rap corresponding to the
target, and two formulas that corresponding with the precondition and the deadline using the procedures described in the previous section.

Figure 12 represents a norm in CPN. The action performing the precondition in CPN means removing a token from the first place of Pre and adding one in its last place. This makes it possible to the Target and DL to fire. Which one will fire first depends on which action occurs first. If Target occurs first, then the Target transition fires, and if DL occurs first, then the DL transition fires. We call the state of performing an obliged action or not performing a forbidden action Compliance, on the contrary, we call the state of performing a forbidden action or not performing an obliged action Violation. The compliance result of the norm differs according to the deontic type of the norm which will be discussed in 5.4.2.

Now we can connect Pre, DL and Target to represent a norm in CPN. The steps are shown as below.

1) Create norm components.
   - Precondition formula (name it Pre)
   - Target rap (name it Target)
   - Deadline formula (name it DL)

2) Connect the last places of Pre to the first Transition of Target rap and the first transitions of DL formula.

3) Connect the first place of Target to the first transitions of DL formula (see figure 12)

Figure 12 Norm in CPN tools

5.4.2 Deontic type
Norms can have several deontic types (see section 2.3.3), namely, obligation, prohibition, and permission. Here we only model obligation and prohibition. The other deontic type, permission is not considered, because permission does not constrain the actions of an actors. The result of a permission type norm will never lead to the final result of “violation” no matter whether the actor performed the action or not. For instance, if the rule says that it is permitted
to change the order before payment, which means no matter whether the relevant actor change the order or not, it never violates the norm.

In CPN model, we use the name of the last places of a norm to identify the compliance and violation result of the norm. Next, we decide the name of the last places according to the deontic type of the norm.

**Obligation**: The CPN representation of an obligation means that, if the precondition was fulfilled, and if the target rap was performed, only then should the token in the first place of Target should move to the last place of Target. This scenario describes the compliance situation. Therefore we name the last place of Target C, representing compliance with the norm. However, if the precondition was fulfilled before the deadline expired, and if the target rap was not performed before the deadline expired, then the token in the first place of Target rap will move to the first transition of DL, and then moves to the last place of DL. This scenario describes the violation situation. Therefore we name the last place of DL violation V, expressing a violation of the norm.

For the deontic type is obligation, we name the output places as in Figure 13. The output place of target rap is “C” (compliance), while the output place of DL formula is “V” (violation).

![Figure 13 Deontic type Obligation in CPN tools](image)

As what we did for formulas (section 5.3), we illustrate the relation between the regulatory scenarios and the CPN representations in Table 9. In this table, instead of Boolean value, we use the sequence of the event occurrence to indicate the scenarios and the result of compliance or violation to illustrate the result of the structure (“No effect” means the structure has not processed completely yet, therefore the result yielded no compliance or violation result comes out). Three scenarios result in compliance, “Precondition -> Target -> Deadline”, “Deadline -> Precondition -> Target” and “Precondition -> Target”. All of them illustrate that 1) the Precondition and the Target occurs, 2) the Precondition occurs before the Target, 3) no Deadline occurs in between. This is the request for a compliance result of a Norm (logical meaning). For these three scenarios, one token is moved to the last place C, which reflects the result of these compliance scenarios in CPN model (CPN representation). The deadline occurring after the Precondition sequentially results violation (logical meaning), correspondingly, one token is moved to the last place V to indicate such result (CPN
representation). For all the other scenarios, the given conditions are not sufficient to judge the result, therefore no result is presented, while the CPN model reflects by no token being in C or V (Table 9).

We use row 2 to row 5 in table 9 as three examples to explain the correspondence relationship between the expected result and the CPN result.

The sequence of events occurrence in row 1 is “Precondition -> Target -> Deadline”. From the interpretation of Norm Nets we know the result of this norm is compliance, because Precondition presents and Deadline occurs after Target. In CPN, when precondition presents first, one token move to its last place, which can enable the first transitions of Target and Deadline. Only then Target occurs. The first transition of Target fires and moves one token to the last place C (Figure 13).

The sequence of events occurrence in row 2 is “Precondition -> Deadline -> Target”. From the interpretation of Norm nets we know the result of this norm is violation, because Deadline occurs before Target. In CPN, when precondition presents first, one token move to its last place, which enables the first transitions of Target and Deadline. Then Deadline occurs. Its first transition fires and move one token to the last place V (Figure 13).

The sequence of events occurrence in row 3 is “Target -> Deadline -> Precondition”. From this sequence, we can draw no conclusion in terms of compliance and violation, because the start of a norm --Precondition, occurs in the end, and no other events follow. In CPN, the first transitions of Target and Deadline cannot be fired in the beginning as the input place from Precondition is empty. Until Precondition occurs, one token moves to the last place of Precondition. In the end, no token appears in the last place of C or V (Figure 13).

The other scenarios can also be explained in the similar way as above. The expected results are coincident with the CPN result (Table 9).
<table>
<thead>
<tr>
<th>Scenario (the sequence of event occurrence)</th>
<th>Expected Result</th>
<th>CPN result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precondition -&gt; Target -&gt; Deadline</td>
<td>Compliance</td>
<td>Token in C</td>
</tr>
<tr>
<td>Precondition -&gt; Deadline -&gt; Target</td>
<td>Violation</td>
<td>Token in V</td>
</tr>
<tr>
<td>Target -&gt; Deadline -&gt; Precondition</td>
<td>No effect</td>
<td>No token in C nor V</td>
</tr>
<tr>
<td>Target -&gt; Precondition -&gt; Deadline</td>
<td>Violation</td>
<td>Token in V</td>
</tr>
<tr>
<td>Deadline -&gt; Target -&gt; Precondition</td>
<td>No effect</td>
<td>No token in C nor V</td>
</tr>
<tr>
<td>Deadline -&gt; Precondition -&gt; Target</td>
<td>Compliance</td>
<td>Token in C</td>
</tr>
<tr>
<td>Precondition -&gt; Target</td>
<td>Compliance</td>
<td>Token in C</td>
</tr>
<tr>
<td>Precondition -&gt; Deadline</td>
<td>Violation</td>
<td>Token in V</td>
</tr>
<tr>
<td>Target -&gt; Precondition</td>
<td>No effect</td>
<td>No token in C nor V</td>
</tr>
<tr>
<td>Target -&gt; Deadline</td>
<td>No effect</td>
<td>No token in C nor V</td>
</tr>
<tr>
<td>Deadline -&gt; Target</td>
<td>No effect</td>
<td>No token in C nor V</td>
</tr>
<tr>
<td>Deadline -&gt; Precondition</td>
<td>No effect</td>
<td>No token in C nor V</td>
</tr>
<tr>
<td>Only one event occurs</td>
<td>No effect</td>
<td>No token in C nor V</td>
</tr>
</tbody>
</table>

*Table 9 Comparison between logical result and CPN representation in obligation norm*

**Prohibition:** The CPN representation of a prohibition expresses that, if the precondition was fulfilled, and if the target was performed, then the token in the first place of Target moves to its last place. This scenario describes a violation situation, and therefore we name the last place of Target V which represents a violation of the norm. However, if the precondition was fulfilled, and the target was not performed before the deadline expired, then the token in the first place of Target moves to the first transition of DL, and then moves to the last place of DL. This scenario describes the compliance situation. Therefore we name the last place of DL C, expressing compliance with the norm.

If the deontic type is obligation, then name the last places as in Figure 14. The last place of target rap is “V” (violation), while the last place of DL formula is “C” (compliance).

![Figure 14 Deontic type Prohibition in CPN tools](image)

Figure 13 and figure 14 show CPN models of a generic norm with deontic type. Until this step, we represented all the information in a single norm in Norm Nets by CPNs. Next we will discuss the relationship between norms.
The same as we explained for obligation norm, the correspondence relationship between the expected results of different scenarios and the corresponding CPN results for prohibition norm is shown in Table 10.

<table>
<thead>
<tr>
<th>Scenario (the sequence of event occurrence)</th>
<th>Expected Result</th>
<th>CPN result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precondition -&gt; Target -&gt; Deadline</td>
<td>Violation</td>
<td>Token in V</td>
</tr>
<tr>
<td>Precondition -&gt; Deadline -&gt; Target</td>
<td>Compliance</td>
<td>Token in C</td>
</tr>
<tr>
<td>Target -&gt; Precondition -&gt; Deadline</td>
<td>Compliance</td>
<td>Token in C</td>
</tr>
<tr>
<td>Target -&gt; Deadline -&gt; Precondition</td>
<td>No effect</td>
<td>No token in C nor V</td>
</tr>
<tr>
<td>Deadline -&gt; Target -&gt; Precondition</td>
<td>No effect</td>
<td>No token in C nor V</td>
</tr>
<tr>
<td>Deadline -&gt; Precondition -&gt; Target</td>
<td>Violation</td>
<td>Token in V</td>
</tr>
<tr>
<td>Precondition -&gt; Target</td>
<td>Violation</td>
<td>Token in V</td>
</tr>
<tr>
<td>Precondition -&gt; Deadline</td>
<td>Compliance</td>
<td>Token in C</td>
</tr>
<tr>
<td>Target -&gt; Precondition</td>
<td>No effect</td>
<td>No token in C nor V</td>
</tr>
<tr>
<td>Target -&gt; Deadline</td>
<td>No effect</td>
<td>No token in C nor V</td>
</tr>
<tr>
<td>Deadline -&gt; Target</td>
<td>No effect</td>
<td>No token in C nor V</td>
</tr>
<tr>
<td>Deadline -&gt; Precondition</td>
<td>No effect</td>
<td>No token in C nor V</td>
</tr>
<tr>
<td>Only one event occurs</td>
<td>No effect</td>
<td>No token in C nor V</td>
</tr>
</tbody>
</table>

*Table 10 Comparison between logical result and CPN representation in prohibition norm*

5.5 From Norm Nets to CPN

After representing a generic norm, our next step is to represent Norm Nets with CPNs. In the meta model, norm net is the super type of Norm, that is, norm net can either be a single norm or a set of Norms connected by logical operator AND, OR, OE. These logical operators are used to represent the relationship between norm nets. In each composed norm net, we use left norm net and right norm net to represent the two sub norm net components of the composed structure. Note that each of the sub norm net can consist of other norm net structures. Therefore, one norm net can be very complex and large in terms of the number of the contained norms and the relationships among them.

5.5.1 AND

AND structure means that the two norm nets (Left norm net and right norm net) of the structure are connected by AND. A violation of either norm net is considered a violation of the composed norm net. For instance, two norm nets “It is forbidden to change the order after payment has been accomplished” and “It is obliged to pay within two days after receiving the goods”. If these two norms are connected with the Norm net operator “AND”, then only complying with both of the two norms is considered to be a compliance of the whole structure (not changing the order after finishing payment and paying before the deadline), the violation of either norm net is considered as a violation of the composed norm (changing the order or not paying before the deadline).
Taking the AND structure as a new norm net pattern, we need to represent the logical AND relationship between its right and left norm net components with CPN and use its last places to represent the compliance and violation result. Therefore we connect the left and right Norm Nets as shown in Figure 15. According to the CPN operation mechanism d), token can only reaches the finial compliance place (last place C of AND) when both of the compliance places of left and right Norm net contains a token. While either of token in the violation places of left and right Norm net could move to the final violation place (last place V of AND). This mechanism reflects the meaning of AND in Norm Net.

The steps to create an AND structure is as below:

1) Create two norm nets corresponding to the two elements of “AND”. Create three transitions and two places.

2) Connect the two compliance places C of each Norm to one transition, and connect the violation places of each Norm to the other two transitions, respectively.

3) Connect the transition, linked to two C, to one place and name it “C”; connect the other two transitions to the other place, and name it “V.

We show the correspondence relationship between the logical meaning and the CPN representation of AND in Table 11. Since the outputs of left and right Norm Nets are marked by compliance (C) or violation (V). We use these two symbols to represent different scenarios. Only two compliance result of its left and right Norm Nets can achieve a compliance result of the whole structure (logical meaning). Correspondingly, in the CPN model, one token can be transferred to the last place C if tokens appear in both the two place C of its left and right Norm Nets (CPN representation). A violation result of either left or right Norm Net results in a violation result of the whole structure (logical meaning). Correspondingly, one token appears in the place V in either left or right Norm Net result a token appearing in the last place V of the AND structure (CPN representation). The other scenarios are 1) only one compliance result in either left or right Norm Net without the result of the other one, 2) no result is
presented for both left and right Norm Nets. In such conditions, it is not sufficient to judge the final result. Therefore in this case the result of AND is null (logical meaning) and no token appears in either the last places C or V.

Taking the row 2, row 3 and row 4 as example, we explain correspondence relationship between the Expected compliance results of AND and its CPN results.

If both results of left and right Norm Nets are compliance (row 2), the result of AND structure is compliance according to the logical meaning. In CPN, result for the sub Norm Net is compliance means one tokens is in its last place C. The last places C of right and left Norm Net with token can enable the transition they connected and finally move a token to the last place C of the structure (Figure 15).

If the result of left Norm Net is compliance is C and the right one is V (row 3), then from the logical meaning we know the expected compliance result is now V. in CPN, the token in the last place V of right Norm Net can move to the last place V of AND (Figure 15). The token in the last place V of left Norm Net cannot move as the transition it connects can only be enabled if both its inputs places hold tokens.

Similar as previous one, but if the right Norm Net does not offer any result (row 4), then the expected compliance result is null as the given information is not sufficient to obtain a result. In CPN, the same reason as previous scenario, the last place C of AND is empty. As the right Norm Net has no output, the last place V is also empty (Figure 15).

The correspondence relationship of AND between expected compliance result and the CPN result for other scenarios can be found in the same way.
Compliance result of Left Norm Net | Compliance result of Right Norm Net | Expected compliance result of AND | CPN result
--- | --- | --- | ---
C | C | C | Token in the last place C, No token in the last place V
C | V | V | Token in the last place V, No token in the last place C
C | Null | Null | No token in the last place C or V
V | C | V | Token in the last place V, No token in the last place C
V | V | V | Token in the last place V, No token in the last place
V | Null | V | Token in the last place V, No token in the last place C
Null | C | Null | No token in the last place C or V
Null | V | V | Token in the last place V, No token in the last place C
Null | Null | Null | No token in the last place C or V

*Table 11: Comparison table of AND*

5.5.2 OR

Or structure means that the two norm nets (Left norm net and right norm net) of the structure are connected by OR. A compliance with either norm net is considered a compliance with the composed norm net. For instance, two norm nets “It is forbidden to change the order after payment has been accomplished” and “It is obliged to pay within two days after receiving the goods”. If these two norms are connected with the Norm net operator OR, then only violation of both of the two norms is considered to be a violation of the whole structure (not changing the order after finishing payment and not paying before the deadline), the compliance with either norm net is considered as a compliance of the composed norm (changing the order or paying before the deadline).

Taking the OR structure as a new norm net pattern, we need to represent the logical OR relationship between its right and left norm net components with CPN and use its last places to represent the compliance and violation result. Therefore we connect the left and right Norm Nets as shown in Figure 6. According to the CPN operation mechanism d), token only reaches the final violation place (last place V of OR) when both of the violation place of left and right Norm net contain at least one token. While either of token in the compliance places of left and right Norm net could move to the final compliance place (last place C of OR), meaning a compliance of the composed norm net. This mechanism reflects the definition of OR in Norm Net.

The steps to create an OR structure are shown as below:
1) Create two norms corresponding to the two elements of “Or”. Create three transitions and two places.

2) Connect the two “C” last places of the two Norms to one transition, and connect the “V” last places of the two Norms to the other two transitions, respectively.

3) Connect the one transition which linked to two “V” places to one place and name it “V”; Connect the other two transitions jointly to the other place, and name it “C”

**Figure 16 Norm OR in CPN tools**

The same as AND, we illustrate the correspondence relationship between the logical meaning and the CPN representation of OR in table 12.
5.5.3 OE (or else)

OE means that there are two opportunities to comply with the relevant regulation. The first one is to comply with the left normnet, if not, then complying with the right normnet is still considered to be a compliance. It is similar with the structure OR, the difference is the right part of OE (Else part) will not be taken into account until the left norm (Or part) failed to be complied with. In other words, the right norm (Else norm) is a complement to the left norm (Or norm). It offers an extra opportunity to achieve compliance. Only when both of the two norms are violated it is considered to be a violation of OE. For example, the two norms “the applicant is obliged to hand in a complete declaration” and “if the declaration is incomplete, then the applicant is obliged to hand in the complimentary material within two weeks after being informed”. If these two norms are connected with the operator “OE”, then if the applicant failed to comply with the first norm, that is the declaration is incomplete, then the applicant still has the opportunity to hand in the complimentary material to comply with the regulations. Only violating both of the norms is considered to be a violation of the whole structure.

To design a CPN model for OE, we need to represent the relationship between the left and right norm net described as above. The right norm net will be enabled by the token transferred from the violation place V of the left norm net. The violation of both of the norm net is a violation of OE, while a compliance with either norm net is a compliance result for the composed structure. We show the CPN model of OE in Figure 17. According to the CPN operation mechanism d), only if the violation place V of the left norm net contains a token, the right norm net can be enabled. The vr place is used as an interface between right and left part of OE. Only the violation of both norm nets will the token be transferred to violation place.
V of OE. A token moves to the compliance place C of OE when either of the two norm nets is complied. This mechanism reflects the definition of OE in Norm Nets.

The steps to create an OE structure are shown as below:

1) Create two Norm Nets corresponding to the two elements of “OE”;
2) Create two places named “C” and “V”, respectively;
3) Create one more place named “vr”;
4) Create four extra transitions, name two of them “C” and the other two “V”;
5) Connect the two “C” last places and two “V” last places of the two Norm Nets to the four transitions, C to C, V to V, respectively.
6) Connect the two “C” transitions to the “C” place;
7) Connect the first “V” transition which connects to the output place of left Norm Net to “VR” place;
8) Connect VR place to the first transitions of right Norm Net;
9) Connect the second “V” transition to the output “V” place;
Similar as AND, we illustrate the correspondence relationship between the expected compliance result and the CPN result of OE in table 13. One point needs to be noticed is that when the compliance result of the left Norm Net is violation (V), and the right Norm Net is not initiated (row 5), then one token is left in place \( vr \) (figure 17). If the compliance checking requests to do a “fully compliance” check, we should take this token as a violation. In other cases, it can also be ignored as it is a “half” violation (still have chance to repair).

<table>
<thead>
<tr>
<th>Compliance result of Left Norm Net</th>
<th>Compliance result of Right Norm Net</th>
<th>Expected compliance result of OE</th>
<th>CPN result</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>Null</td>
<td>C</td>
<td>Token in the last place C</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>No token in the last place V</td>
</tr>
<tr>
<td>V</td>
<td>C</td>
<td>C</td>
<td>Token in the last place C</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>No token in the last place V</td>
</tr>
<tr>
<td>V</td>
<td>V</td>
<td>V</td>
<td>Token in the last place V</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>No token in the last place C</td>
</tr>
<tr>
<td>V</td>
<td>N</td>
<td>Null</td>
<td>Token in ( vr )</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>No token in the last place C or V</td>
</tr>
<tr>
<td>Null</td>
<td>C</td>
<td>Null</td>
<td>No token in the last place C or V</td>
</tr>
<tr>
<td>Null</td>
<td>V</td>
<td>Null</td>
<td>No token in the last place C or V</td>
</tr>
<tr>
<td>Null</td>
<td>Null</td>
<td>Null</td>
<td>No token in the last place C or V</td>
</tr>
</tbody>
</table>

Table 13 Comparison table of OE

5.6. Conclusion

In this chapter, we created the CPN constructions for every Norm Nets elements. For every CPN construction, we listed the construction process and explained the CPN mechanism.

By doing the analysis in this chapter, we connected the two models, the meta model of Norm Net and the model of CPNs. After this, we implemented this connection in the transformation
tool. The coding for these elements in the transformation tool is designed similar to the construction processes described in this chapter. Thereby, the transformation tool is able to transform a Norm Nets model to CPNs.

In the following part, we use this transformation tool to do a case study. From this case study, we can further check our idea about the transformation process. It is also a validation process to the whole compliance checking approach.
6. Case study

In this chapter we apply our compliance checking approach in a case study. The case is from IBM Corporation and Intrasoft. It is a project on designing a declaration management system for the customs department. The material includes the designed framework of the declaration handling process and the relevant regulations which they refer to. Our task is to check whether the declaration handing process is compliant with the relevant regulations.

In this chapter, we first introduce the background of the case briefly (section 6.1). Then we formalize the business regulations into Norm Nets (section 6.2). In this step, we selected several articles from the business regulations. Then we translated the semantic of the chosen article to the Norm Nets model and compose them into a Norm Net of the chosen regulations. Next the Norm Nets are transformed to CPNs by the transformation tool automatically (section 6.3). We show the output of our transformation tool (Norm Net) in a diagram. At last, we conclude this chapter (section 6.4). The CPN resulting from the transformation tool is used in the next chapter for the analysis of the business process.

6.1. Introduction of the case

The case is a project, from the IBM Cooperation and Intrasoft International, which aims to help their customer (customs agencies) to design and set up their declaration management system.

The processing of a Customs declaration is a complex process which is subject to change. Activities that are needed to handle a declaration are typically related to the registration, checking and control of movements of goods. During the whole process, change may occur due to any differences between the declaration statement and the corresponding requirements. The quantity of the regulations is large and the relevant rules are complex. In the document of regulations we have, there shows near 350 articles. For some types of declarations more than 50 activities can be identified (van de Aalst, 1996). Customs always need to react to different conditions. High ratio of mistake and delay of the customs department process may directly lead to the financial loss of the declaring companies and decrease the economic attraction of the region.

The management system is designed to handle all the processes from receiving the declaration, validating the declaration, doing risk assessment until clearing the declaration, monitoring after release and finalizing process. It is supposed to offer the functionalities as sending reminders, monitoring the progress timeline, document storing etc. The management system is constructed according to the specification of the “manage declaration” process. The process features in the main process document are depicted by process diagrams, subprocesses, decision points, activities, messages and services. From this file, we could identify the logical sequence of the activities, decision points and sub-processes in the main process diagram. There are also more files to describe the sub-processes and other information. The
“manage declaration” process is designed to minimize trade delay, minimize human intervention, and realize risk based control and optimal national variability. This declaration management system has already been accomplished and implemented.

Above all of these design principles and requirements, such as a management framework, the “manage declaration” process needs to comply with the relevant customs laws and regulations. Actually it was designed according to the relevant regulations. However, as what was mentioned in Chapter 1, the complexity of the regulation and the interrelationship between articles make the design of the management process a highly systematic project. Neglecting a single article in the regulation or detailed relationship may lead to conflicts. In this chapter and the next chapter, we apply our compliance checking approach to check whether the declaration management process is compliant with the relevant regulation.

6.2. Mapping from regulations to Norm Net

To design such an IT system, IBM first collected all the relevant laws and regulations of the management declaration process. Then they asked the lawyers to interpret all the articles of law and translated them to the form of “if...then...” to make them understandable to IBM design engineers. The number of relevant articles in the form of “if... then” are near 350. Because of the time limitation, it is not feasible to cover all of these articles of law in our case study. Moreover, because what we want to achieve through this case study is to validate the compliance checking approach, it is not necessary to include too much articles with similar structures (And, Or, Before etc.) in the case. Therefore we chose a sub-module in the declarant management process called Acceptance Declarant. From these sub-module we have selected six articles which could reflected the most of logical relationships in the Norm nets model. Through this case study, we can validate whether the approach of compliance checking is feasible and applicable in the real world. If through this case study we find no error and nothing we cannot represent with Norm Net, the approach we used to analyze these six articles could later on be generalize to more articles.

The six chosen articles of law (including the original “if...then...” regulations in Dutch and the corresponding English translation made by us) are shown in Appendix 1. Next we interpret these six articles into Norm Nets.

Each article in the regulation corresponds to a Norm Net. The process to model a single article with Norm Nets is actually the process of defining the deontic types, targets, deadlines preconditions and logical connections between norms according to the semantics of the regulations. In this thesis, this transformation process is realized by a human. The correctness of the transformation is highly dependent on the person’s knowledge of Norm Nets and the knowledge of the relevant laws and regulations. Sometimes, expert consulting might be necessary, just like what we did for this case study. The true meanings of the articles of law were explained and confirmed by an expert from IBM.
We illustrate the transformation result of each the articles as below (the combination of letter and number in front of every regulation is the ID of that regulation in the IBM regulation File, we use the same ID to identify their corresponding norms in our norm Net model).

A2: If the applicant was informed about checking on some objects of control, then the application for correcting the data of controlled object is not allowed.

- Deontic type: Forbidden
- Target: (declarant, change data)
- Deadline: (customs, clear declaration)
- Precondition: (customs, inform control)

A3: If the customs finds out that the statement in the declaration are incorrect, then the changes are not allowed.

In text, this regulation means that once the error was found, both the declarant and the customs are forbidden to change the declaration. This article has to be understood in a context. According to the interpretation of the IBM expert, this norm only applies to the declarant. It is because that it is implied from the context of the article at hand in the legislative text. This article relates to a request for correction, and such a request is done by the declarant. Customs is generally allowed to make changes if incorrect information is found.

According to such an interpretation, A3 could be formed as:

Norm A3
- Deontic type: Forbidden
- Target: (declarant, change data)
- Deadline: (customs, clear declaration)
- Precondition: (customs, find errors)

A4: If goods are released, then change is not allowed.

For the same reason of A3, the forbidden action “change” here is applied to only to the declarant but not to the customs.

Therefore we form A4 as:

Norm A4
- Deontic type: Forbidden
- Target: (declarant, make changes)
- Deadline: (customs, clear declaration)
• Precondition: (customs, release goods)

**A21:** IF a customs debt is incurred AND the requested procedure in the declaration is not temporary admission with partial exemption, THEN the goods are only released AFTER:

• The amount of the customs debt has been paid,

OR

• Security has been provided for the whole amount.

Article A21 illustrates two conditions under which the goods should not be released. We could use two norms to represent this regulation. The relationship between the two conditions is indicated in the text by the word “OR” explicitly. Accordingly, we connect the two norms with OR in Norm Nets model.

Norm A21a

• Deontic type: Forbidden
• Target: (customs, release goods)
• Deadline: (declarant, pay customs debt)
• Precondition: (declarant, incur customs debt) and (declarant, not request temporary admission with partial exemption)

Norm A21b

• Deontic type: Forbidden
• Target: (customs, release goods)
• Deadline: (declarant, secure customs debt)
• Precondition: (declarant, incur customs debt) and (declarant, not request temporary admission with partial exemption)

Article A21 therefore should be represented as (A21a ∨ A21b)

**A62:** If the declaration is incomplete and goods are released, then it is allowed to submit the data or the document within one month at most after the acceptance of the declaration.

• Deontic type: Obliged
• Target: (declarant, supplement data)
• Deadline: (customs, accept declaration) BEFORE (time, pass 1 month)
• Precondition: (declarant, submit incomplete declaration) AND (customs, release goods)
B70: If incomplete declaration is submitted, then it must contain information which can identify the goods AND the information must contain a 'quantity measurement' ELSE the declaration will not be accepted.

- Deontic type: Obliged
- Target: (customs, reject)
- Deadline: (customs, accept declaration)
- Precondition: (declarant, submit incomplete declaration) AND (customs, find missing data)

We use 7 norms to represent the 6 legal articles. The 5 norms A2, A3, A4, A62, B70 are all compulsory for making the process, which means the process need to comply with each of them, therefore we connect them with AND. As explained before, the relationship between A21a and A21b is OR, therefore we connect them with OR. At last we use AND to link (A21a ∨ A21b) with (A2^A3^A4 ^ A62^B70). The Norm Net consists of these 7 norms is the Norm Nets model we construct for the case.

**Norm Net model for the case:** A2^A3^A4 ^ A62^B70 ^ (A21a ∨ A21b)

The illustration of this Norm Net in the transformation tool is shown in Appendix 2.

### 6.3. Mapping from Norm Nets to CPN

After constructing the Norm Net manually, we transform this Norm Net to CPN automatically by the transformation tool.

The input of the transformation tool is shown in Appendix 2. After completing the input process, we run the program and get the output xml file. Opening the file with CPN tools, we get the corresponding CPN diagram of the Norm Net model. We add tokens to all the input places, waiting to be fired according to the Event Trace (which will be introduced in next chapter). And now the CPN is ready to be fired according to business process which we will demonstrate in next chapter. The CPN diagram with tokens is shown in Figure 18.
6.4. Reflection

In this chapter, we modeled the regulations in a use case with the Norm Nets Model. This is the left branch in the conceptual model (see Figure 3). By doing this step, we transformed the regulations into a CPN model and input the model into CPN tools. Once we have the input from the process part, we can apply the compliance checking approach to make a check for the case. This part will be shown in next chapter.

The Norm Nets model is applied to a real case. In this case, we modeled six articles of law. The number of the modeled articles is small. The limitation of this will be further discussed in section 8.1.

For the first step, the transformation from regulations to Norm Nets, we do it manually. This step is highly depended on the person who formalizes them. The knowledge of the person on Norm Nets and his understanding of the specific articles of law determine the correctness of the result. For example, regulation A62 states “...then it is allowed to submit the data or the document within one month...” We thought here the word “allowed” implies that it is a right for to submit the data or document, therefore the corresponding deontic type is “permission”,

Figure 18 The CPN diagram of the case
however, after consulting the expert, we knew it actually means “obliged”. Even for these six modeled articles, we need consult the expert a lot.

For the chosen legal articles, the model of Norm Nets is able to represent the content of the separate articles as well as the relationships among the different articles. We generated a CPN of the Norm Net model of the use case by using the transformation tool. The correctness of the transformation from regulations to Norm Nets and the transformation from Norm Nets to CPN will be further checked in Chapter 7.
7. Validation of the model

In previous chapter we built a Norm Net to represent the relevant regulations in our case and expressed it in a CPN model. With this step realized, the left branch in our conceptual model is fulfilled (see Figure 3). In this chapter, we transform and input the declaration management process to the CPN which we created in Chapter 6 and complete the compliance checking approach. We asked the expert to evaluate the final result. By doing so, we validated the Norm Nets model as well as the approach.

In this chapter, we first transform the declaration management process into the form which could be input in the CPN (section 7.1). After that we run the simulation (section 7.2) and get the checking result (section 7.3). Next, we illustrate the feedback from IBM expert (Section 7.4) and present our evaluation (section 7.5) At last we reflect on the approach as well as the validation result and answer sub research question 2 and 3 (section 7.6).

7.1. The Event Trace

The declaration management process is complex. Inside the main process there are a large amount of decision points, activities and sub-processes. Our task for this step is to abstract all the actions mentioned in the chosen regulations and find their corresponding positions in the declaration management process. Once we know all the possible positions of the actions in the process, we can link the actions to be a chain according to the order of the actions in the process. We call this chain Event Trace. Later on we use the event trace as input to the CPN model to indicate in which order the transitions should be fired and check the compliance.

To get the event trace, we first need to find all the actions in the regulations. The actions that appear in the selected regulations are show in Table 14 below.
<table>
<thead>
<tr>
<th>ID Number</th>
<th>Relevant Action in Regulations</th>
<th>Name in CPN model</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Change declaration</td>
<td>changeD</td>
</tr>
<tr>
<td>2</td>
<td>Clear declaration</td>
<td>clearD</td>
</tr>
<tr>
<td>3</td>
<td>Inform control</td>
<td>informControl</td>
</tr>
<tr>
<td>4</td>
<td>Find errors</td>
<td>fError</td>
</tr>
<tr>
<td>5</td>
<td>Release goods</td>
<td>release</td>
</tr>
<tr>
<td>6</td>
<td>Supplement data</td>
<td>supplementD</td>
</tr>
<tr>
<td>7</td>
<td>Accept declaration</td>
<td>acceptD</td>
</tr>
<tr>
<td>8</td>
<td>Submit incomplete declaration</td>
<td>submitIncompD</td>
</tr>
<tr>
<td>9</td>
<td>Pass 1 month</td>
<td>pass1M</td>
</tr>
<tr>
<td>10</td>
<td>Reject declaration</td>
<td>rejectD</td>
</tr>
<tr>
<td>11</td>
<td>Find missing yet mandatory data</td>
<td>findMissingData</td>
</tr>
<tr>
<td>12</td>
<td>Submit complete declaration</td>
<td>submitCompD</td>
</tr>
<tr>
<td>13</td>
<td>Incur customs debt</td>
<td>incurCDebt</td>
</tr>
<tr>
<td>14</td>
<td>pay customs debt</td>
<td>payCDebt</td>
</tr>
<tr>
<td>15</td>
<td>Request temporary admission with partial exemption</td>
<td>noPartialExemption</td>
</tr>
<tr>
<td>16</td>
<td>Secure the customs debt</td>
<td>secureCDebt</td>
</tr>
</tbody>
</table>

*Table 14 The actions in regulations*

By reviewing the declaration management process files and after having the discussion with IBM expert, we assigned the actions in Table 14 into different positions in the declaration management process and abstracted that into seven Event traces.

To describe the event trace clearly, we first explain the meanings of the terms and notations we used in the event trace diagram as below.

- **Regulated action**: the action in our chosen regulations (see Table 14)
- **Blue box**: The process step without regulated actions occurring
- **Orange box**: The process step with regulated actions occurring
- **Words in the box**: The name of the process step
- **Words in the bracket**: The name of the regulated actions in CPN model
- **Number in the bracket**: The ID number of the regulated action (defined in Table 14)

Then we list all the six event traces and explain the corresponding processes which they represented.

- **Event Trace 1**

  This event trace (shown in Figure 19) includes 4 actions in the regulation, action 8, action 12, action 4 and action 10. It describes the scenario which mainly relates to regulation A3. It describes that after customs receiving the declaration, they perform validation. If the customs finds major errors in the declaration, then the declaration is rejected.
This event trace (shown in Figure 20) includes 3 actions in the regulation, action 8, action 11 and action 10. It describes the scenario which mainly corresponds to regulation B70. It indicates that after customs receives an incomplete declaration, they perform validation. If the required data (information which can identify the goods and 'quantity measurement') is missing, then the declaration is rejected.

This event trace (shown in Figure 21) includes 5 actions in the regulation, action 8, action 7, action 1, action 3 and action 10. It describes the scenario which relates to regulation A2 and B70. This event trace indicates that after customs receiving an incomplete declaration, the process goes through Perform Validation, Accept Declaration, Notify Accept Declaration, Risk Assessment, Initiate Control and Perform Control, then if major discrepancies were found, the declaration will be invalidate after the procedure Determine Customs Position.
• Event Trace 4

This event trace (shown in Figure 22) includes 7 actions in the regulation, action 8, action 7, action 6, action 5, action 1, action 3 and action 2. It describes the scenario which relates to regulation A2, A4, A62 and B70. The main difference from the previous process is that after Determine Customs Position, customs Release Goods, Receive Supplementary Declaration, and then Process Supplementary Declaration, Mark Ex-incomplete Declaration and Clear Declaration in the end.

![Figure 22 Event trace 4](image)

• Event Trace 5

This event trace (shown in Figure 23) includes 7 actions in the regulation, action 8, action 7, action 6, action 1, action 3, action 5 and action 2. It describes the scenario which mainly relates to regulation A2, A4, A62 and B70. The main difference from the event trace 4 is the process step Receive Supplementary Declaration occurs earlier. In event trace 5, it follows Perform Control.

![Figure 23 Event trace 5](image)
• **Event Trace 6**

Event trace 6 and 7 involves in the events related to customs debt. Therefore they relate to regulation A21. Event trace 6 describes that if Customs Debt Incurs and the declarant Pay Customs Debt, then the customs can release the goods.

![Event Trace 6 Diagram]

**Figure 24 Event trace 6**

• **Event Trace 7**

Similar as event trace 6, this event trace illustrates that after customs debt occurs, if the declarant can “Secure The Customs Debt”, the customs releases the goods.

![Event Trace 7 Diagram]

**Figure 25 Event trace 7**

In the creation of the event traces, the information of the process part was already transformed into a form which can be used as input into the CPN model. This accomplishes the transformation work on the right branch of the conceptual model (see Figure 3). Next in the compliance checking, we run the CPN model and get result.
7.2. The compliance checking process

The CPN model is constructed based on the chosen regulations. The event traces were made according to the business processes. Now we converge these two information flows together by firing the corresponding transitions in the CPN model (which stands for regulation information) in the order of appearance in the event traces (which stands for process information), and check whether tokens will appear in the violation place of the CPN, thus indicating that a conflict between the process and the regulation has occurred.

Specifically, the operation procedure is go about like this. Firstly we take one event trace and find the earliest-occurred regulated action (these are the actions in the orange box in the event trace diagram) in the event trace. Then we find this action (represented by transitions) in the CPN diagram. From these transitions, we fire only those ones which are in the enable state. This step transfers one token from the input place of the transition to its output place. After firing 1 transition, we fire the next enabled transitions if their name are started with “auto”, these transitions are those without corresponding actions and should be fired automatically. That is because these represent the operations related to AND, OR, OE in the Norm Nets model which are corresponding to the regulations but not effected by the process inputs. We repeat this until no more transitions can be fired anymore. Next we repeat the procedure with the second regulated action in the event trace. If any token would arrive at the last violation place (V34 in Figure 18) during this procedure, we record this relevant action and analyze the reason. Finishing all of these step, we know whether conflicts exist between the process and the relevant regulations. If the last violation place contains tokens, then we know the recorded actions are the reasons of those conflicts. Reflecting on the relevant regulations, we can analyze the conflicts.

Figure 26 in next page illustrates the process we follow for this simulation process.

---

4 In CPN tools, a transition in enable state means that there are enough tokens available in the input places
7.3. Result

After running the checking procedures, we would found that there is one token in the last violation place (V34 in Figure 18) in the CPN diagram. This means that there is a conflict between the declaration management process and the six regulations we chose.

Analyzing the reason for this conflict, we found the token arrives in V34 is because of the Rap (declarant, change the declaration). Further investigating on the CPN diagram, we found this is a violation of regulation A2.
The CPN analysis result for this conflict is shown in Figure 27. The three list are the markings of the initial state, before the violating state and after the violating state. From the red marks we know the token in V34 in the third list changed after one action was taken. Form the execution process, we know that that action is Change the Declaration.

**Figure 27** The CPN analysis result

Going back to the process, in the process step “Determine Customs Position”, we found the description: “The supervisor will hand over the control result to a specialist who will take a customs position by determining whether the control resulted in [...] a minor discrepancy, in which case he will ask an assistant to correct the declaration or instruct the declarant to do so”. If we apply this regulation on Event Trace 4 or Trace 5, conflict will occur against regulation A2.

This conflict is because the rule says that if the declarant was informed about control, then the declarant is forbidden to change the declaration, while in the process step “Determine Customs Position”, however, the process still allows the declarant to make changes on the declaration.

### 7.4. Expert feedback

After accomplished the case study, we had an evaluation meeting with the IBM expert.

We explained Norm Nets model to the IBM expert. Our emphasis lays in explaining the structure of Norm, as the definition of the precondition, the deadline, the target and the deontic type. The representation of the norms for the selected articles was confirmed by him.
We also explained our verification process. We showed the actions we abstracted from chosen articles, and how did we formalize the event traces. The seven event traces were confirmed by the expert.

At last, we demonstrated the compliance checking result. The expert agreed that the conflict we found is indeed in the document. However, through the discussion, we realized that the reason for the conflict is actually because of the system designer’s concern for national variability. When they designed the process step “Determine customs position”, they realized that there are some countries that allow the declarant to change the declaration while some others do not. To keep the national variability, the designer deliberately left these two options open in the document. However, as the system is currently mainly applied in EU countries, whose regulations do not allow the declarant to make changes in this step, the system has blocked the possibility for the declarant to make any changes in the process step “determine customs position” by technical means. Therefore, this conflict has been avoided through the implementation process.

7.5. Evaluation

In this section, we evaluate this project, present the achievements and drawbacks. Moreover, we demonstrate the advantage and benefit for organizations to implement the compliance checking approach.

- The meta mode of Norm Nets is established. Not only the structure of Norm Nets is clear in the meta model, because of its maintainability and expandability, it also make the model to model design for other purpose easier.
- The transformation tool has been developed. The function of transforming Norm Nets to CPN automatically is realized.
- The case study is accomplished. We proved the compliance checking approach is feasible in the case study.

Though the verification, we demonstrated our compliance checking approach in CPN (see Figure 26). During the operation, we also find some drawbacks.

- Firing the transitions manually according to the event trace in the CPN diagram is not an easy and handy task. The complex diagram structure, limited display space on the screen and the limited name length of the transitions all increase that difficulty. Therefore the approach currently is applicable for compliance checking with a large amount of regulations.
- Both the process file and articles of law need the interpretation of the expert. The time we spent on this takes the large part of whole approach.

We will further discuss this part in limitation (section 8.2).
We propose a compliance checking approach by the use of analysis software CPN tools. By applying such a computer assisted approach, the requirement of compliance checking can be met. Not only the compliance result, but also the reason, the type and the degree of the violation can be identified by the organizations. The organizations can then take measures to modify the business accordingly, in this way, they can reconcile their time and cost efficiency with the requirements of the governmental authorities. Moreover, by the automatic transformation tool, the speed of such checking in organizations is increased. They do not need a group of lawyers to compare the process with regulations item by item which is typically how they did it in the past.

7.6. Conclusion

In this chapter, we transformed the process information to CPN and ran the simulation of compliance checking in CPN tools. The expert feedback confirmed our interpretation of the process (event trace) and the interpretation of the regulations (Norm Net). The misunderstanding on the checked conflicting point in the process was explained by the expert. Therefore we conclude the declaration management process is complied with the six relevant regulations.

By finishing this chapter we can now answer the sub research questions 2 and 3.

Sub-question 2: How can we detect the conflicts between the business processes and the governmental regulations by using Norm Nets?

The approach we applied in chapter 6 and 7 answered this question. Here we conclude it briefly.

To detect the conflicts between the business process and the governmental regulations, we need to transforms these two items into other forms which can be expressed in CPN tools, as they are not able to be input to CPN tools (the analysis tool) directly. We transformed the regulations into Norm nets and transformed the process into event trace.

We first interpret the governmental regulation into a Norm Net manually, and then transform the Norm Net to a CPN by the designed transformation tool.

For the business process part, we first list all the actions appearing in the regulations. Then we assign the actions to different places in a chain according to their positions in the process. This chain we call event trace. For one business process, we can abstract several event traces corresponding to different scenarios.

After completing the transformation for the regulations and the processes, we can run our CPN verification. The specific process is illustrated in 7.2 (see Figure 26).

In this compliance checking approach, we detect the conflicts between business processes and governmental regulations. Conflicts will appear as tokens in the final violation place of the CPN model.
Sub-question 3: What suggestion can be derived from the case study result?

We answer this question in two points.

a) For the company, as our result shows the declarant checking process is complied with the chosen regulations, there are no suggestion on adjusting the management process. However, the unclear point (see 7.3) where the conflicts occurs does alert us to pay attention to the document maintenance. We would suggest that if the system has already made the choice for some open points during the implementation, it might be better to update the document accordingly in time, or make a mark beside the relevant item.

b) For our meta model of Norm Nets, we would suggest that he sub-module “Not” deserve more consideration as in the case, we have “Not” existing in the articles. Currently we treat this logical relation during the operation process of CPNs. There could be other means.

In the next 3 chapters, we will present the conclusion of this thesis (Chapter 8), discuss the main limitation and the future work of this research (Chapter 9), and illustrate our reflection on the project and present the personal suggestion for the future graduate students (Chapter 10).
8. Conclusion

All the companies acting in the business world are requested to ensure that their business processes comply with the governmental regulations, otherwise they may suffer from the law issues, reputation impairment and economic loss. However, checking the conflicts between governmental regulations and the business processes is necessary yet difficult. The complex content and structure, the interrelationships among regulations, the regulations’ property of constantly changing, makes it difficult to be realize just by manual work.

This thesis proposes a compliance checking approach by the use of analysis software CPN tools. By applying such a computer assisted approach, the requirement of compliance checking can be met and the efficiency of such checking in organizations can be improved significantly. They do not need a group of lawyers to compare the process with regulations item by item any longer. Although the approach still has a number of limitations, we believe (together with the refinement to the approach and the future research) applying this new compliance checking approach can bring the organizations at least these benefits:

1) Standard regulation database can be established and shared. Once the database is built up, the relevant organizations can have a standard interpretation of regulations to support their organizational compliance work.

2) The workload of compliance departments in organizations can be reduced. Their work will change from multiple tasks related to compliance checking procedures to mainly focus on the maintenance of regulation database and analysis on the optimal process path. In this way, the change of the regulations can be monitored and treated.

3) The organizations can make their business processes more reliable in term of regulatory compliance. They could better foresee the opportunity and cost of the planed and ongoing projects.

4) The control and maintenance cost of the authorities can be reduced. It provides the possibility to provide the integrated service (regulation interpretation, consultation, risk management...).

5) As every actor in the business world has a better view on the regulations and regulatory compliance, cooperation among organizations is possible to be built. The efficiency of the supply chain and the relevant business activities can be improved.

In this graduation project, we analyzed the model of Norm Nets, designed a transformation tool, and verified the compliance checking approach through a case study.

In part 1, we first reviewed the research domain and presented the conceptual model of the project. Then we described the model of Norm Nets and how to use this model to represent the articles of law. Next, we defined the transformation process from Norm Nets to CPNs.
Through programming, at last we automatized this procedure by a designed transformation tool.

In part two, we validated the compliance checking approach by the use of Norm Nets through a case study. In this part, we first introduced and implemented the compliance checking approach. Then the validation result by the IBM expert and the reflection were presented in the end of the case study.

The whole thesis describes what Norm Nets is, how to apply the compliance checking approach by the use of Norm Nets and what the validation result is concluded through the case study. From the “three cycle view of design science research” (Hevner, 2007), we can structure the thesis into three cycles. The relevance cycle covers chapter 1, 3, 8. This cycle connects the empirical foundation and the design project. The design cycle consists of chapter 5, 6, 7. This chapters contain our design process and the evaluation step. The rigor cycle includes chapter 2 and chapter 4, which is our theoretical foundation.

In this project, our main contribution is:

1) Established the meta model for Norm Nets. The meta model not only demonstrates the formalism of Norm Net, but also, offers an extensible model for future research.

2) Developed a transformation tool (design artifact). The transformation tool generates the CPN diagram automatically which dramatically increases the transformation efficiency.

3) Validated Norm Net as well as the compliance checking approach through a case study.

Concluding the three sub research questions 1 through 3, we can finally answer the research question:

Research question: Can the conflicts between governmental regulations and business processes be detected by the use of Norm Nets?

First of all, Norm Nets is a model we developed to model the governmental regulations. We argue that this model is capable to represent the information of a single article of law and the interrelationships among multiple articles. A set of regulations with interrelations can be modeled by using Norm Nets.

The logical features in regulations are captured by the formalism of Norm Nets. The typical normative vocabularies used in laws and regulations are “must”, “ought”, “may”, “should”, “shall”, etc. (Susskind, 1987) We reflect these words in Norm by the deontic type, permission, prohibition and obligation.

Besides those vocabularies, the other elements of an article of law like applied condition, deadline, and the behavior which is ruled by this article also have their corresponding components in a Norm, namely precondition, deadline and target, respectively. Moreover, we designed formulas to represent the complex combinations within a norm component. The
Role-Action Pairs and Formulas (RapAnd, RapOr, RapBefore, RapNot. See 4.3) make it possible to describe these kinds of composed relationship.

Additionally, Norm Nets capture the relationships between different legal articles by using the structure of AND, OR and OE (or-else). These operators represent different meanings (see 4.5), thus they connect different articles in an interrelated net. This net could represent a set of interrelated governmental regulations.

Therefore Norm Nets is capable to model a set of governmental regulations with interrelations.

Secondly, only formalism itself cannot check the conflicts. It is just one step in our conceptual model of the whole approach (Figure 3). To detect the conflicts between the business process and the governmental regulations, we need to transform business processes into other forms which can be expressed in CPN tools (the analysis tool), as it is not able to be input to CPN tools directly. We transform the process into event trace.

We first interpret the governmental regulation into a Norm Net manually, and then transform the Norm Net to a CPN by the designed transformation tool automatically. While for the business process part, we first list all the actions appearing in the regulations. Then we assign the actions to different places in the event traces according to their order in the process. For one business process, we can abstract several event traces. At last, we can run our CPN verification. The specific process is illustrated in 7.2 (see Figure 26). In this compliance checking approach, we detect the conflicts between business processes and governmental regulations. Conflicts appear as tokens in the final violation place of the CPN model.

In this way, we can detect the conflicts between governmental regulations and business processes.

In this thesis, we have left a number of issues and research tasks out of scope. In the next chapter, we give an overview of limitations and future research.
9. Discussion

In this chapter, we first discuss the regulatory compliance checking in supply chain management. We discuss the main problem and relevant topic in this domain (section 9.1). Then we discuss the limitations of this project and the future research. Some of the limitations have already been mentioned in previous chapters, some others are what we found while we apply the compliance checking approach. We conclude them all in section 9.2. The future research mainly discussed the possible research can be done in the future to improve the compliance checking approach and the complete the model of Norm Nets. The future research in section 9.3 is closely related to the limitations in section 9.1.

9.1. Regulatory compliance checking in supply chain management

Supply chains are crucial and complex network to manage for companies, especially for those high-tech companies which located part of their supply chain in different countries. It is requested for such companies to make their own business process compliant with the local regulations and the laws. The regulations can come from different sources, many of which are updated constantly (Boella, et al., 2013), it is very difficult for companies to keep track of relevant legislations.

Supply chain management relates to multiple actors in the supply chain. The compliance checking in the supply chain also need to take multiple shareholders into account. As we stated in the very beginning of the thesis (Table 1), the two main parties of compliance checking, the governmental authorities and the companies, have different interests. Such differences may influence their decisions and strategies. Not only may those two, from the perspective of a high-tech company, more stakeholders played a part in its regulatory compliance, e.g. the vendor may influence the product quality, the logistics contractor may influence the delivery time. Checking the compliance with regulations and knowing the effect of potential violation in advance can help the companies better perform the shareholder management and risk management. From this point of view, compliance checking can help companies to make specific management strategies to different stakeholders, foresee the risks and arrange corresponding backup plans.

Compliance-related decision can also be influenced by organizational strategy if considering about the relevant cost. Occasionally, companies may decide it is not worth conforming to certain norms when the cost of implementation is great compared to the fine or loss of reputation they would face in case of non-compliance (Boella, et al., 2013). This situation was also convinced by our IBM expert when he explained their customers’ requirements to us.

This thesis proposes a compliance checking approach. It is an attempt to solve the compliance checking problem. However, it still need further research to enrich and optimize its functionality. This limitation and future research are discussed in the next sections.
9.2. Limitations

Through this project, we have already proved that the compliance checking approach is a feasible approach. However, there are several limitations we found during the implementation of the approach.

1) Although we proposed RapNot should be included in the Norm Nets model, due to the limited time, we did not succeed to define a formal CPN representation for RapNot as other formula connectors (RapAnd, RapOr, and RapBefore). For the work next step, we want to suggest two points while making such a RapNot. The first one is RapNot can have different CPN representations according to its different positions in a Norm (Target, deadline, precondition). The second one is that not just by CPN model, RapNot can also be reflected in the compliance checking phase by not firing the corresponding transitions.

2) The limitation in the compliance checking process. Firstly, the transformation from regulations to Norm Nets is done by hand and highly dependent on the person who transforms it. Secondly, deontic type and the vocabularies in regulations are not strictly one to one related. Words like “must”, “ought”, “allowed” can have a variety of meanings, and their occurrence in legislative text is not itself sufficient reason for determining a deontic type of a norm. Besides the semantics analysis, sometimes we have to rely on the experts’ interpretation. Lastly, we automatized the transformation process from Norm Nets to CPNs, however, for several other steps we still do them manually. This makes the compliance checking approach is currently not feasible to handle too many regulations. That is because a) the transformation from to Norm Nets need knowledge of both regulation and Norm Nets, manually do it is inefficient; b) manually firing the transitions according to the event trace is not suitable for large number of regulations.

3) Not all regulations have explicit deadlines. What we do currently is if we cannot find a deadline in the regulation, then we will just set the deadline to be the last step in the process. However, again, it varies from case to case, mistakes and misunderstanding could happen if we do not interpret the process correctly. In this case, we again need the support of the expert.

4) As what we stated in section 3.2, the user of the transforming tool need to have sufficient knowledge about Norm Net and CPN. This requirement implies the approach we proposed currently still need more work to get the level “easy to use” to other users. The compliance information need to be communicated in the organization across disciplines. All in all, CPN tools is not such a software which can easily get started and be quickly handled.

5) The limitation of the case:
Firstly, although we intended to cover all the constructions in Norm Nets (RapAnd, RapOR, AND, OE .etc.) in this case study, we did not find the OE relation in the given material. Because the regulations in the IBM files mainly apply to one organization—the customs department, while the OE relation would more tend to appear in the cross-organization regulations.

Secondly, what we have for the case study is the high-level process document instead of the code of the declaration management system. We do not know how exactly the process was implemented. It is a black box for us. We cannot ensure logical features in the documents are exactly the same as the declaration management system. In this case, we rely on the interpretation of the expert.

Thirdly, we only chose six articles in the case study, although most of the components in Norm Nets model are included, we do not know whether problems would occur if a large number of articles are transformed and apply in the compliance checking approach as inputs.

Lastly, through the discussion with the IBM expert, we found sometimes there is a difference between the reality and the design, e.g. theoretically it is possible to make such a group of event traces, but some event trace abstracted from the process file can never occur in the reality. Therefore those selected ones would actually make no sense.

Next, we discuss the future research corresponding to these limitations.

9.3. Future research

In the future work, to avoid the mistakes made by people and to improve the efficiency of the approach, the transformation from regulations to Norm Nets can be automatized by applying semantic identification and AI techniques. We know it is another challenging research domain, maybe we need external support or cooperation, but by doing this, we could increase the efficiency of the compliance checking approach dramatically. Another step need to be automatized is the transition firing process. By exploring other functions of CPN tools, we can fire the transitions in the order of the actions in the event traces without human intervention. Once we complete these two steps, the whole approach is automatized, so that we can process more regulations.

Back to our Norm Net model, I think it is necessary to find a proper way to represent RapNot relation in Norm Nets model as well as the transformation tool. It is because and, or, not are the basic logical operators. It is relatively easy for users to use these logical operators to describe regulations. In addition, from the IBM process file, we found several articles contains the relation of RapNot.

Right now it is possible for us to check the compliance between existing regulations and processes, we believe afterwards it is possible and promising to generate new compliant
processes with the model of Norm Nets based on the relevant regulations. It would be a useful assistant tool for business process design, offering the process designers more optional processes and help them to make better choice.

It would require the user to have considerable knowledge of Norm Nets and CPN in order to check the compliance by applying this approach. This requirement also implies that more future work can be done to improve the user interface and automation level.

Finally, future research should make use of other properties of CPNs, such as time. If we have absolute time to be the deadline, then we do not need to assume inexistent deadlines in regulations. In addition, multi-level Norm Nets together with its hierarchical CPN model is also one of the research direction in the future.
10. Reflection

This chapter starts with a reflection from MoT (management of technology) perspective (section 10.1). We discuss the relation of this project with my master study. The next two sections contain my personal reflection on the process and result of the 6-month graduation project behind this thesis. I will start by explaining how my master program led to this particular graduation project (section 10.2). Then I list a number of dos and don’ts which I realized in hindsight (section 10.3).

10.1. MoT Perspective

My master program in Delft University of Technology is MoT (Management of Technology). In this program, the main topic is how to explore and understand technology as a corporate resource to make the corporates competitive in its business domain. The courses in the first year helped in learning various techniques on how to observe, analyze, make forecasts and manage the impact of technology while adjusting the management accordingly. In the second year of my master program, I chose Supply Chain as my specialization. The specific knowledge of supply chain management was discussed, including the relevant concepts, the models, the methods, etc.

While I was doing the case study, I was very impressed that various new technologies are really changing today’s world. I realized that such changes took place even in the customs department, which is a typical bureaucratic governmental agency in my previous impression. They are implementing the new IT system to help them increase the efficiency and mitigate the risks. This is from the aspect of technology push. On the other hand, market and social impact also affect the development new technology. We talked to our IBM expert and got to know that while they tried to sell their new management system to the customs departments in different countries, some of the business ministers may ask the department to compare the price the new system with the possible fine and lost they would get without the system – a realistic cost–benefit analysis. This interaction between technology and market reflects the courses which I took in the first year like technology dynamics, innovation management, and technology strategy. While we develop new technology, we have to always take the social reaction into consideration, especially if the technology is developed for a specific market, then it may be important to make the technology have other properties besides advancement and originality, as easy to use, easy to maintain acceptable price, etc.

Supply chain is another perspective to understand the practical value of this graduation project. We know the simple truth that that with one link broken, the whole chain is broken. Supply chain is also composed of a large amount of links. The compliance condition between regulations and the business processes is one of such links. The violation in any step of business process can affect other linked steps and the performance of the whole chain, as lead time, storage cost, stock turn over, etc. In addition, for the supply chain managers, governmental influence and the law & regulation issues are inevitable aspects need to be keep
in mind. Foreseeing conflicts before it happens could help the supply chain managers to have an overview on the potential regulatory conflicts in the whole process and make better decision to mitigate the risks and maximize the benefit. Therefore, a compliance checking approach could be a part of the supply chain management system, managing and optimizing all kinds of supply chains.

10.2. Background of this thesis.

Before I started this period of master study, I have been working for four years in different companies. I delivered several self-designed automotive testing equipments when I was a test engineer. It is true that I learned a lot of new theoretical knowledge during this master study, however, besides those abstract theories, I really missed the feeling to get some concrete artifacts being done. Therefore in my graduation project, I was eager to apply those theories in the real world to “make something” instead of “suggest something”. I hope the deliverable of the project could be a combination of the theoretical model and some designed artifact which can be used to solve practical problems. Eventually I got this project from ICT department of TBM faculty. When I got to know in the project we collaborate with IBM and one of the main deliverable is an IT program, I think it met almost all the demands and wishes I had for a graduation project.

When I started, I found the difficulty is far beyond my imagination. Because most of the software (such as EMF, Eclipse, CPN tools) used in this project is totally new for me, I have to spend a lot of time to learn from basic. Fortunately, I got great support from my graduation committee. I am really grateful to all my supervisors for their help and patience. Although some times (during various stages of this thesis process) were quite tough, in the end I learn a lot. And I am especially happy that the transformation tool works at last which makes me regain the fantastic feeling I had when my first testing software worked three years ago.

10.3. Dos and Don’ts

This is the first time I take such a graduation project—limited time, unfamiliar domain, scientific research, etc. With hindsight, I have quite a few dos and don’ts for both the future graduate students and for myself.

I think probably for every master student, the importance of making plan and schedule is nothing new and has been practiced for many times. However, I think it might be different to make a plan for the graduation project, because it has a strict timeline (if you do not want to delay your graduation) on one hand and so much uncertainty on the other hand in terms of what exactly you need to do. For the latter one, I mean before starting the project, no matter how hard you have worked on the proposal, how many articles you have reviewed, it is almost impossible to ensure there won’t be a new area you need to investigate during the process. Especially for such a design science project, you will never know where you may get stuck during the process and how long you need to spent on that specific problem. Therefore it is
very difficult to forecast the workload beforehand. During my project, I spent a lot of time in making the meta model until early May, without considering that I planned to finish programming and start cast study since mid-May. I am just luckily that I got great support from my committee and catch up in time. My suggestion for the schedule issue is besides making a schedule, probably it is more important to keep on checking it with the current progress. Always discuss the schedule issue in the meeting with the supervisors and make adjustment accordingly. Do not just work and work without thinking about the schedule, and all of a sudden it is too late to catch up the schedule.

Another issue I want to mention here is the role of the second supervisor. The first time I met my second supervisor is when I was about to start the project. Until the mid-term review I did not contact him again. After that, I was reminded by my first supervisor to hold a meeting with him. In that meeting, I found actually my second supervisor offered me a lot of useful ideas from a totally new perspective. I think not only for myself, might it be common for other students that the contact to the second supervisors is sufficient. With hindsight, I would strongly recommend to meet the second supervisors for more times, because they can really give valuable and fresh suggestions from a totally new perspective.

Lastly, finalizing this thesis in formal, scientific English cost both me and my first supervisor a lot of time, far beyond my estimation. I would suggest students who are not really good at English, especially English scientific writing, to take some courses before starting the thesis work. I never thought it would be a main problem for my thesis, but it turned out that it is truly a problem. Besides normal English words and grammar, there are a large number of implicit rules and hints of scientific writing in terms of scientific words, structure, layout etc. For students like me, without a systematic training, it is really hard to management it in a short time. For this, I would greatly appreciate my first supervisor. He explained me a lot patiently and corrected all the mistakes one by one in this thesis. Therefore the last don’t I would suggest is do not underestimate the difficulty of English scientific writing.
BIBLIOGRAPHY


Amini, M., 2010. Deontic logic-- introduction and application in computer science, s.l.: s.n.


Eysholdt, M., 2009. EMF Ecore Based Meta Model Evolution and Model Co-Evolution, s.l.: s.n.


<table>
<thead>
<tr>
<th>ID</th>
<th>Business rule (origineel)</th>
<th>English translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>A2</td>
<td>INDIEN aangever weet dat goederen fysiek worden gecontroleerd</td>
<td>If the applicant was informed about checking on some objects of control, then the application for correcting the data of controlled object is not allowed</td>
</tr>
<tr>
<td></td>
<td>DAN wordt wijziging niet meer toegestaan</td>
<td></td>
</tr>
<tr>
<td>A3</td>
<td>INDIEN douane constateert dat vermeldingen in de aangifte onjuist zijn</td>
<td>If the customs finds out that the statement in the declaration are incorrect, then the changes are not allowed</td>
</tr>
<tr>
<td></td>
<td>DAN wordt wijziging niet meer toegestaan</td>
<td></td>
</tr>
<tr>
<td>A4</td>
<td>INDIEN goederen zijn vrijgegeven</td>
<td>If goods are released then change is not allowed</td>
</tr>
<tr>
<td></td>
<td>DAN wordt wijziging niet meer toegestaan</td>
<td></td>
</tr>
<tr>
<td>A21</td>
<td>INDIEN een douaneschuld ontstaat EN er geen sprake is van tijdelijke invoer met gedeeelte</td>
<td>If customs debt is incurred and there is no temporary importation with partial exemption, then the goods can only be released after</td>
</tr>
<tr>
<td></td>
<td>lijijke vrijstelling DAN worden de goederen pas vrijgegeven NADAT:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Het bedrag van de douaneschuld is betaald; of</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Voor het gehele bedrag zekerheid is gesteld</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A62</td>
<td>INDIEN sprake van onvolledige aangifte EN goederen zijn vrijgegeven DAN is de periode</td>
<td>If the declaration is incomplete and goods are released, then it is allowed to submit the data or the document within one month at most after the acceptance of the declaration.</td>
</tr>
<tr>
<td></td>
<td>voor overlegging gegevens en/of bescheiden maximaal een maand na aanvaarding van de aangifte</td>
<td></td>
</tr>
<tr>
<td>B70</td>
<td>INDIEN een onvolledige aangifte is ingediend DAN moet deze de gegevens voor de identificatie van de goederen bevatten EN moet deze de gegevens voor de hoeveelheid bevatten ANDERS wordt de aangifte niet aanvaard</td>
<td>If incomplete declaration is submitted then it must contain information which can identify the goods AND the information must contain a 'quantity measurement' else the declaration will not be expected.</td>
</tr>
</tbody>
</table>
Appendix 2

The user interface of the transformation tool