

No. 2014-079

ADAPTATION: A PARTIALLY AUTOMATED APPROACH

By

T. Manjing, F.A. Bukhsh,
H. Weigand

18 December, 2014

ISSN 0924-7815
ISSN 2213-9532

Adaptation: A Partially Automated Approach

T. Manjing, F.A. Bukhsh and H. Weigand
Department of Management, Tilburg University, the Netherlands
f.a.bukhsh@uvt.nl, h.weigand@uvt.nl

December 18, 2014

Abstract

This paper showcases the possibility of creating an *adaptive auditing system*. Adaptation in an audit environment need human intervention at some point. Based on a case study this paper focuses on automation of adaptation process. It is divided into solution design and validation parts. The artefact design is developed around import procedures of M-company. An overview of the artefact is discussed in detail to fully describes the adaptation mechanism with automatic adjustment for compliance requirements. Validation part of this paper is based on an experimental validation of the artefact on simulated data.

Keywords: Adaptation, Compliance management, Auditing

Jel code: M15, M42

1 Introduction

As global trade grow into more and more complex, any organization that is involved in international trade is obligated to manage its Customs control compliance at various levels. The levels of controls can be categorized as follows:

- **Internal policies:** In order to control the internal processes organizations set compliance policies. These compliance policies are imposed by various parties along the organizational hierarchy ranging from top management to the responsible business units;
- **Policies of trade partners:** Trade usually involves several parties to make up a supply chain. Suitable policies are generated for number of trade partners along the supply chain and several specialized service providers such as financial service providers, Customs brokers, and shipping service providers;
- **Standards:** The organization may choose to adopt various standards and frameworks in order to improve different aspects of the organization. The examples of these standards include GS1, COBIT, etc.;
- **National laws and regulations:** Based on multiple factors government implements laws and regulations at national level. Whereas at the international level managing legal compliance of national laws is usually a complex affair;
- **International trade laws and agreements:** From last few decades international trade agreements have played important role. These are bilateral trade agreement. While majority of the agreements aim to assist the trade between partner nations, some agreements create more burdens to the trade organization in the form of additional standards.

When combined together, these different levels of policies and regulations make the logistic compliance management complicated. An example of the situation can be a company that wants to import goods and sell them in the European union (EU). As a Customs union, the EU imposes the same amount of Customs duty regardless of the point of entry into the union. On the other hand, value added tax (VAT) is imposed by the countries where the company markets and sells the products. Due to the freedom of movement within the EU, tracking of duties and VAT payment of these goods is complex for both the importer and the regulators. Moreover, the importer may choose to outsource some parts of the import process to third-party specialists. For example, the products may be handled by a shipping company while the payment of Customs duties is done by another agent that is specialized in Customs brokerage.

To handle the growing complexity in cross-border logistics compliance management, many organizations adopt information systems (IS) in their management practices. Technology adoption assists auditors in their work in automating the partial /whole compliance checking process. Underlying objective behind IS audits are safeguarding assets, maintaining data integrity, and achieving the organization's goals.

Even though automation of audit procedure is in place in many organizations, the ever-changing policies and regulations usually require human operators to adjust the system manually. Manual adjustment is time-consuming, expensive, and error-prone. A possible solution to this can be to create a system that configures and optimizes itself according to a change in compliance requirements. This can free the staffs from system configuration tasks to work on more productive assignments. By reducing human-involvement in the system possibility of errors introduced by human is also minimized.

2 Background

Compliance management refers to conforming to stated requirements. Such requirements can be rules of law, regulations, industry standards, organization policies, agreement, or even best practices (Silveira et al., 2011). This compliance management practice is a collection of procedures, methodologies and technologies which are used to assess the state of compliance and the risks related to the compliance requirements. For instance, SOX requirements mean that any electronic communication must be backed up and secured with reasonable disaster recovery infrastructure. Healthcare providers that store or transmit e-health records, like personal health information, are subject to HIPAA requirements.

The problem of compliance management is very complicated since each organization has to deal with several requirements on different levels. For example, a multinational public company has to comply with the rules set by the stock exchange governing body while abide by all the corporate laws imposed by all the countries that the company operates in. Adding complication to the current compliance governance is the fact that the company chooses to adopt a number of standards such as those set by international organization for standardization (ISO) in order to facilitate the organizational efforts in various aspects (Yates and Murphy, 2007). If we look inside the company itself, it can be noticed that each business unit has to manage its own segment-specific compliance; the production line has to maintain its safety standards while financial department has to ensure that all the financial transactions within the organization are verified.

To aid the compliance management effort, some organizations may choose to adopt frameworks such as committee of sponsoring organizations of the treadway commission (COSO). COSO is an initiative dedicated to improving organizational performance and governance through effective internal control, enterprise risk management, and fraud deterrence (McNally, 2013). In parallel to known framework there are some newly developed compliance requirement frameworks some of them are (a) based on process life cycle and provides an extensive discussion of process compliance (Sakr and Awad, 2010) (b) for compliance management it focuses on a policy-based approach for automatic semantic consistency checking (El Kharbili, 2012). In the following section we have discussed compliance management of import and export procedures from Customs' perspective with a focus on adaptation.

2.1 Customs Procedures in European Union

It is important to understand certain import and Customs declaration processes since the case study used in this work directly involves audit of these procedures. However, it is not practical to analyse all the Customs regulations of all the countries covered in the case study since that is not one of the main goals of this work. To demonstrate an import declaration procedure, the practice in the European Union (EU) is selected as an example. The reasons behind the choice are as followed:

- The EU is one of the Customs regions covered in the case study;
- As most advanced and the most integrated economic union in the world, the EU is a perfect example of a complex Customs' system
- Despite the ongoing effort to centralize its Customs procedures, there are still inconsistencies among different agencies and authorities across the EU. This reflects the complicated situations that businesses regularly have to deal with in Customs declaration processes.

The European Union is an economic and political union of 28 member states in Europe (European Commission, 2013). As an economic union, the EU is also a Customs union by definition. A Customs union is a trade bloc which is composed of a free trade area with a common external tariff. It implies that all the 28 Customs administrations of the EU states must act as though they were one (European Commission, 2014). The Customs authorities of EU member states implement EU policies in almost every field related to international trade. The Customs play a very important role in facilitating trade and protect the interest of the EU and its citizens (European Commission, 2014). Goods that are produced within the EU are automatically *community goods* which can circulate freely among the EU member states. On the contrary, goods that enter EU Customs territory from a non-EU country are referred to as *non-community goods*. These goods have to go through a Customs procedure before they can be marketed in the EU (Belastingdienst, 2014). Once all the conditions in the procedure have been satisfied, the *non-community goods* will become community goods which bestow on them the same status as goods that have been produced within the EU. These goods may be transported, stored or sold within the EU Customs territory without being subject to Customs formalities. To release the non-community goods into free circulation, all the following conditions have to be satisfied:

- A declaration should be filed for the goods: The goods have to be declared at the first point of entry into the economic union regardless of their final destination. For example, goods destined for Luxembourg city are imported through a seaport in Rotterdam. The goods must be declared to the Customs authority at the first point of entry which is Rotterdam. The Customs authorities may offer several options for making a declaration. Travellers can make declaration verbally when flying into an airport carrying luggage. Importers traditionally file written declarations for large amount of goods. Some Customs authorities such as the Dutch tax and Customs administration even offer an option to file declaration electronically (Belastingdienst, 2014). The fact that the messages are sent in standardized Electronic Data Interchange (EDI) format make it possible for the importers to integrate the Customs declaration process with their computer systems.
- Certain formalities also have to be fulfilled when the declaration is made: Such formalities include submitting invoices and ensuring that the goods available for inspection by Customs authority.
- The import duties and/or other import taxes owed should be paid: The Customs duties or tariff is common to all EU members regardless of the point of entry, but the rates of duty differ from one kind of import to another. The rate depends on the kind of good and its origin. Thus, obligation on a person to pay the amount of the import duties is referred to as Customs debt (European Commission, 1992, 2008). According to the Community Customs Code (CCC), the Customs debt should be paid within 10 days from the notification of an underpayment from the Customs authority. However, each Customs authority may choose to extend the enforcement period. For example, Revenue and Customs Commission in the United Kingdom allows an extra five days grace (making it 15 days in total) before

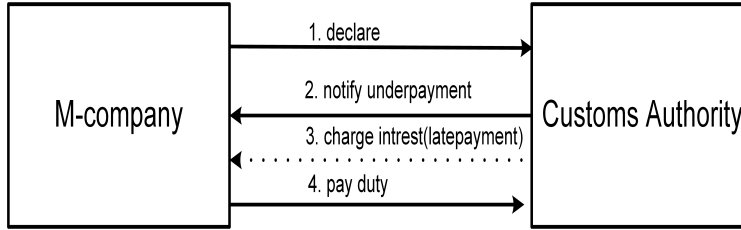


Figure 1: Interaction between M-company and a Customs authority

charging interest while Revenue Commissioners of the Republic of Ireland enforces 10-day period as prescribed by the CCC (European Union, 2004) (HMRC 2005; Revenue 2012).

2.2 Case Study

M-company is a multinational company. M-company imports goods mostly from Asia and distributes them to different clients retail stores all over the region of its operations. For the goods destined for the European market, the company has to declare imported goods to one of the member states Customs office and pay duty before the goods are allowed to be released into free circulation. M-company also has to perform similar procedures in the countries outside the EU. The practices in these countries may vary slightly but we assume that all of them follow the procedure depicted in Figure 1. The following summarizes several business practices of M-company that are basis for generation of simulated data. Test cases generated for validation purpose are factious.

- Due to the variety of the origin and destination of *goods*, M-company imports goods through various ports, throughout its region of operation, in order to reduces the delivery time and cost. However, this strategy leads to number of compliance management complications since Customs procedure in different countries are independent and differ slightly;
- The finance department of M-company usually summarizes and authorizes Customs duty payments only twice a week on every Tuesday and Friday. These days are referred to as entry-closing days;
- M-company pays the duties through services provided by banks. The payment may take different amount of time to reach the Customs authorities in different countries.

The interaction between M-company and a Customs authority as illustrated by Figure 1 is explained below:

1. M-company declares the imported goods to the Customs authority. In response, the Customs inspects the goods to ensure that the declaration is valid and the goods comply with the standards and regulations of that country or EU.
2. The Customs notify M-company of duty underpayment. The company is compelled to pay the duty within limited amount of time which may vary from country to country. Failing to do so leads to penalty from the Customs authority such as interest accrued on the Customs debt.
3. The Customs charge interest on the Customs debt. This only occurs when the company fails to pay the duty within the period notified by the authority.
4. M-company pays the outstanding Customs debt to the Customs authority.

3 Artefact Overview

The artefact, *adaptive auditing system*, is designed to audit and manage compliance requirements of all the import Customs procedures in M-company. Even though the artefact discussed in Figure 2 is tailored

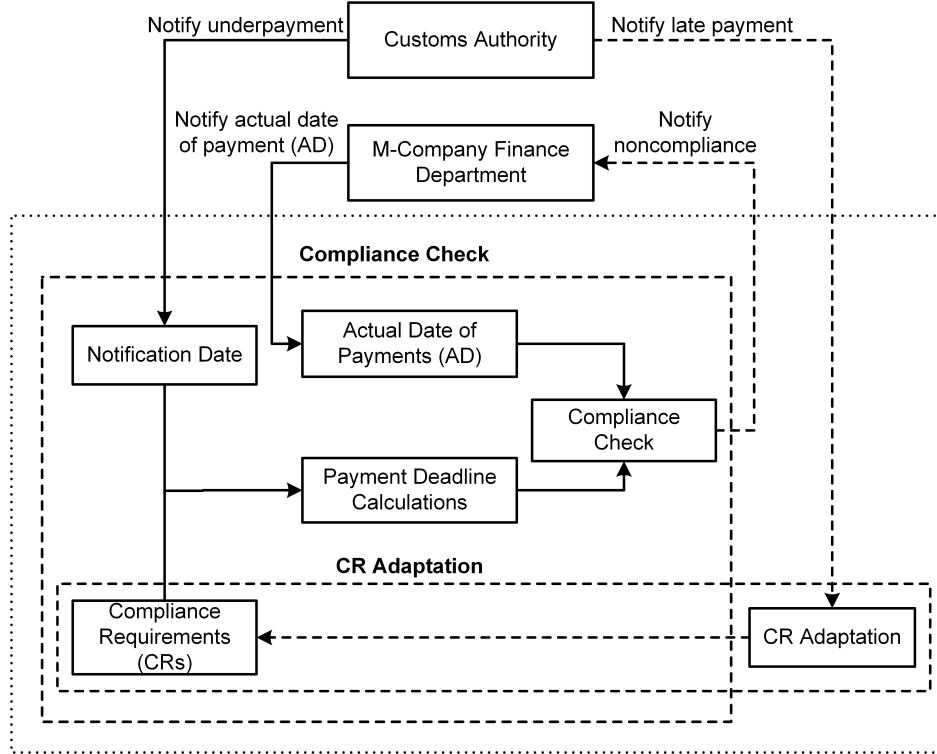


Figure 2: M-company *adaptive auditing system* overview

for M-company case study, its overall design is generic enough to be used in other compliance auditing systems as can be seen in Figure 3. Most of the components of the smart auditing *frameworking framework* (cf. Figure ??) can be seen in the *adaptive auditing system*. Input data consist of ‘payment record’ and ‘notification dates’. These inputs help to generate IST (i.e the ‘actual date of payment’) which is being compliant checked against SOLL consisting of ‘compliance requirement’ and ‘payment deadline calculator’. During compliance checking late payment notifications invoke the adaptation module. Decision making authorities are Customs authorities and M-company. Main components of smart auditing *frameworking framework* are IST, SOLL compliance checking and adaptation are fully covered in Figure 3.

- **Customs authority:** A Customs authority notifies the system about the underpayment (either by electronic message or being manually keyed in by staffs). In the case that M-company fails to pay the Customs debt within the notified period, the authority also informs the system about the situation and the penalty.
- **Finance department:** The finance department of M-company is responsible for paying the Customs debts to the Customs authorities. The department is also subject to auditing by the *adaptive auditing systems*.

Figure 2 shows that the *adaptive auditing system* is consists of two subsystems which share common compliance requirements (see subsection 3.1). These two subsystems are as follows:

- **Compliance check subsystem:** The main purpose of this subsystem is to check whether the finance department pays outstanding duty to the Customs within the *final date of payment* (FP) set by the system or not. The FP is calculated from compliance requirements (rules) and transaction information extracted from Customs underpayment notification. If actual date of payment is later than payment deadline, the system warns the finance department of the non-compliance. This procedure is described in more detail in subsection 3.2;

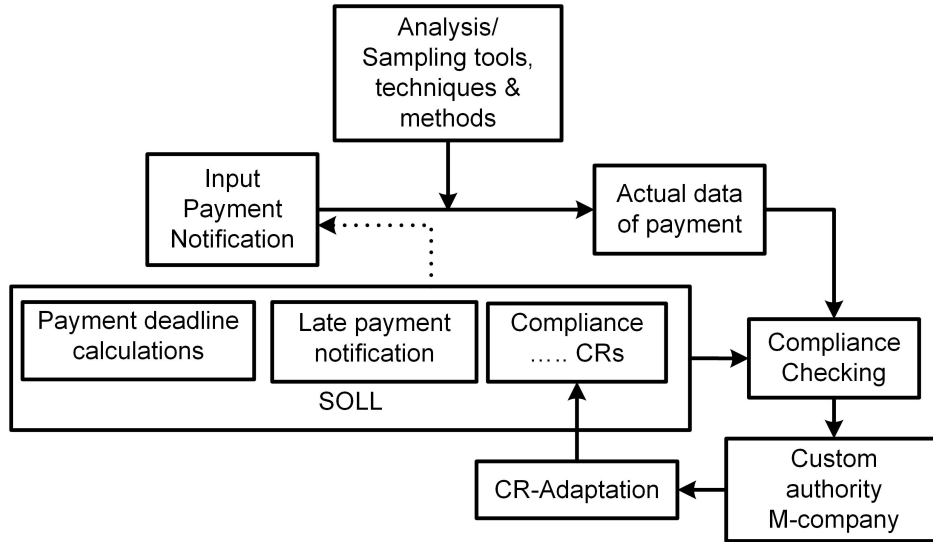


Figure 3: M-company *adaptive auditing system* against *smart auditing framework*

- **Adaptation subsystem:** This subsystem automatically adjusts the compliance requirements which are in turn used by *compliance check subsystem* for its audit task. The goal of this adaptation subsystem is to correct inaccurate compliance requirements, which may cause the compliance check subsystem to miscalculate FP. Adaptation subsystem is explained in subsection 3.3

3.1 Compliance Requirements

In this work, *compliance requirements* (CRs) refer to sets of controls that govern how compliance is checked in the *adaptive auditing system*. Compliance requirements are the core components of the *adaptive auditing system*. On one hand, the compliance check subsystem uses the dates calculated from these requirements to ensure compliance. On the other hand, the adaptation part of the system is designed to optimize these compliance requirements. These compliance requirements can be divided into three categories namely (i) *constraint*, (ii) *Customs specific requirement* (CSR), and (iii) *default Customs specific requirement* (default CSR). The CRs can also be easily translated into one of the rule formal languages. As explained earlier, rule language can help separate the representation of the requirements on business level from those on execution level.

3.1.1 Constraint

A constraint is a requirement that expresses a limitation or restriction on the behaviour of the system. A constraint can also represent an exact definition of a business term. In this scenario, three constraints have been identified. All these requirements are configurable which means they can be modified manually in case of a change in restriction.

- (1) A working day is a day that is NOT Saturday OR Sunday OR a public holiday

The first constraint defines the meaning of the business term ‘working day’. It describes that a working day is any day that is not a Saturday, a Sunday, or a public holiday. This constrain is used in the calculation of dates of payment which is explained in following section;

- (2) An entry-closing day is Tuesday OR Friday

An entry-closing day is a day that the finance department of M-company summarizes and authorizes Customs duty payments. This constraint specifies that M-company only closes its entries on Tuesday and Friday;

(3) *Transfer Period* (TP) must be less than or equal to 5 working days

This constraint specifies that TP must not be more than five days because the bank that provides money transfer service to M-company guarantees that the number of working days for funds to arrive (worldwide) is within five business days;

3.1.2 Customs Specific Requirement (CSR)

CSR is a requirement customized to each Customs authority. CSR is designed to maintain independent requirements for each Customs authority based on a principle that Customs in different countries are governed by different rules and these rules may change independently. CSR is the only category of compliance requirement that the CR adaptation subsystems can modify automatically. CSR is based on two terms which are explained below:

- *Interest-Free Period* (IFP): IFP is a period in which M-company can pay off Customs debt without being subjected to interest. The period varies from country to country depending on the Customs code of that specific country.
- *Transfer Period* (TP): Fund transfer service provided by a bank usually takes a few days for the fund to arrive at the designated Customs authority. The period in which the bank takes to transfer the fund is referred to as TP in this work. TP may differ from one Customs authority to another because the fund transfers may take different amount of time to reach Customs in different countries. Here we emphasize that the period-counting systems of TP and IFP are different. TP only accounts for working days within the period while IFP constitutes every day with in the period.

To elaborate the difference between IFP and TP, the following is an example of a CSR for the Customs authority of a country ‘A’:

```
IF customs is 'A'  
THEN IFP is 10 days AND TP is 3 working days
```

The CSR states that Interest-Free Period allowed by the Customs authority of ‘A’ is 10 days while Transfer Period which the bank takes to transfer the debt payment to the Customs authority of ‘A’ is three working days.

3.1.3 Default Customs Specific Requirement

Default Customs specific requirement (Default CSR) is the default setting for a newly created CSR. When the *adaptive auditing system* detects a new Customs authority that does not match any current CSR, it automatically creates *Customs specific requirement* for that specific Customs. Like CSR, default CSR is made up of 2 terms; default IFP and default TP. As the name suggests, default IFP and default TP are the default values of IFP and TP respectively. Both the default terms can be set manually by the system administrators. The following example illustrates the default CSR where the value of default IFP and default TP are set to 20 days and zero working day respectively.

```
IF customs is UNKNOWN  
THEN IFP is 20 days AND TP is 0 working day
```

3.2 Compliance Check Subsystem

Compliance check is one of the two subsystems of the *adaptive auditing system*. It checks if outstanding Customs debt is paid on time or not. This is done by comparing an actual date that the finance department pays off the debt with an estimated *final date of payment* (FP) which is calculated from

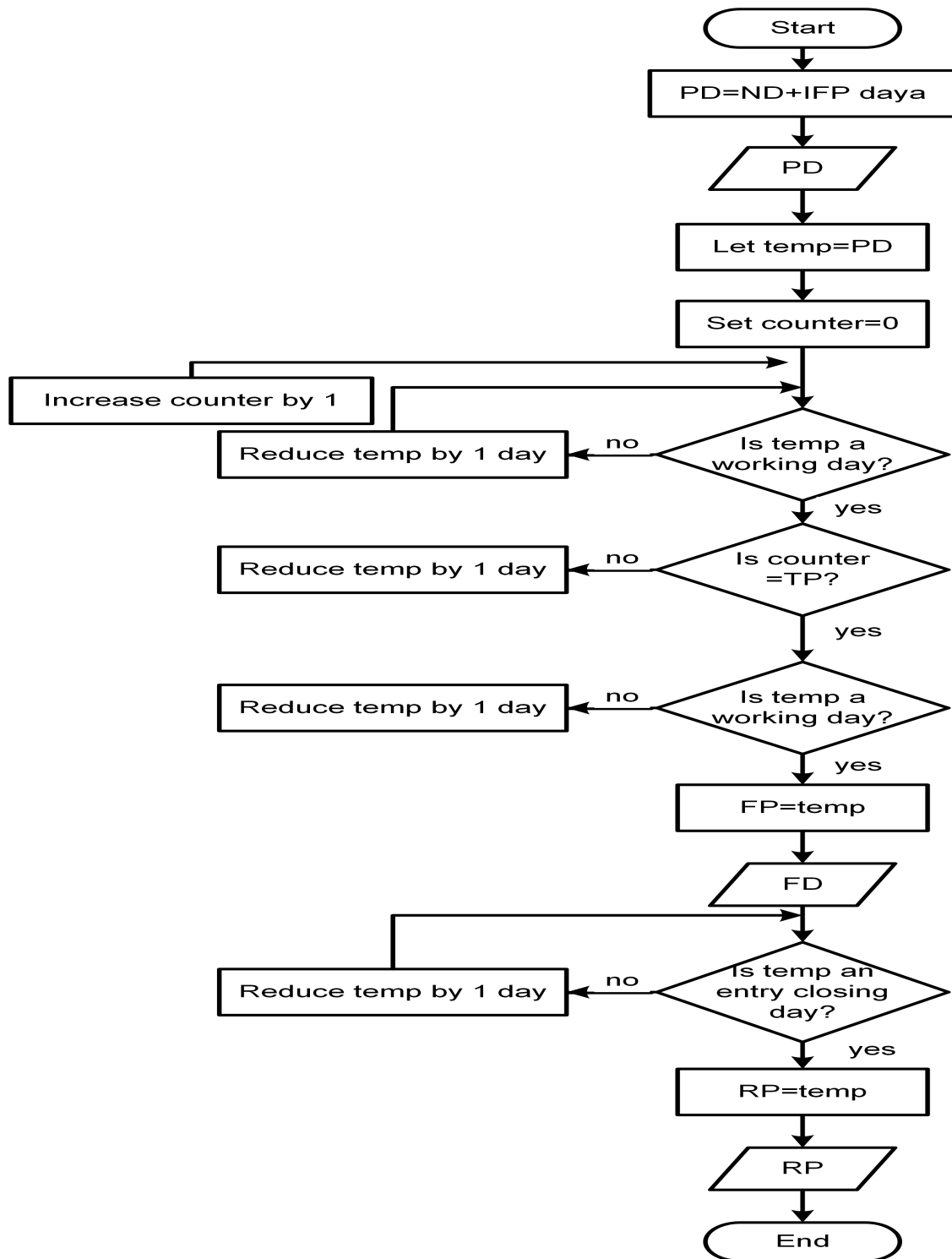


Figure 4: Payment Deadline (PD), Final Date of Payment (FP), and Recommended Date of Payment (RP) calculation algorithm

Input	
Term	Description
Interest-Free Period (IFP)	The period in which M-company can pay off the Customs debt without being subjected to interest.
Transfer Period (TP)	The period in which the bank takes to transfer the fund to the Customs.
Entry-Closing Day	Every Tuesday or Friday.
Working Day	Every day except Saturday, Sunday, and public holidays.
Notification Date (ND)	The date that the Customs notify the company of the debt. ND is extracted from Customs underpayment notification
Output	
Term	Description
Payment Deadline (PD)	The latest date possible that the finance department can pay off the Customs debt without being subjected to interest
Final Date of Payment (FP)	The latest date possible that the finance department can transfer a fund to pay off the custom debt
Recommended Date of Payment (RP)	The date that the finance department should transfer a fund to the Customs debt according to entry-closing days

Table 1: Inputs and outputs for Payment Deadline (PD), Final Date of Payment (FP), and Recommended Date of Payment (RP) calculation algorithm

transaction information extracted from underpayment notification and CSR for that specific Customs (see Figure 2).

The system uses an algorithm shown in Figure 4 to calculate the FP. The algorithm does not only produce FP which is the latest date possible that M-company can transfer a fund to pay off the custom debt. It also generates PD and RP. PD is the latest date that the finance department can pay off the Customs debt without being subjected to interest. It is used to determine whether or not *Interest-Free Period* (IFP) should be adjusted by CR adaptation subsystem. Another output from the algorithm, RP, is the closest entry-closing day that comes before the FP. RP is recommended to the finance department as a date that it should transfer the fund to pay off the Customs debt. This ensures that the company will pay off the debt on time while maintaining its two-days-per-week entry closing schedule. Since RP is also the latest entry-closing day in which the company can transfer a fund to pay off the Customs debt, it makes certain that the company holds on to its fund as long as possible which in turn improves the company liquidity.

The algorithm also requires a number of inputs to calculate PD, FR, and RP. These include *Customs specific requirements* (both IFP and TP), entry-closing days, working days, and *notification date* (ND). The first four inputs; IFP, TP, entry-closing days, and working days are documented as compliance requirements while ND is extracted from underpayment notification received from Customs authorities. Table 1 lists and describes all the inputs and outputs for the algorithm. Figure 4 illustrates the calculations of PD, FP, and RP. These calculations are explained below:

- Payment Deadline (PD): PD is a result of adding a number of days in the *Interest-Free Period* (IFP) to the notification date.
- Final Date of Payment (FP): To derive FP from PD, the system has to check whether PD is a working day since the fund can arrive at the Customs authority only on working days. If PD is not a working day, the algorithm will find the earliest working day before PD. The system then subtracts a number of days in the TP from the PD or the earliest working day in case PD is not a working day. Note that TP only counts the number of working days and FP has to be on a working day. The final result of this subtraction is FP.

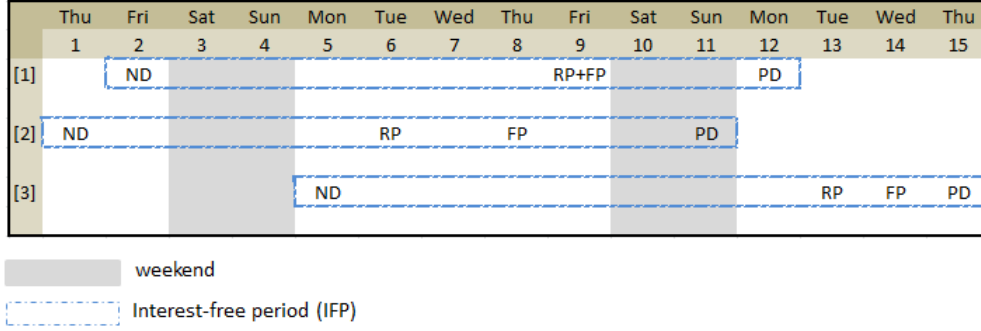


Figure 5: Examples of Payment Deadline (PD), Final Date of Payment (FP), and Recommended Date of Payment (RP) calculation (IFP is 10 days and TP is 1 working day)

- Recommended Date of Payment (RP): To derive RP from FP, the system check if RP is a closing-entry date or not. If the answer is yes, the FP itself is also RP. Otherwise, RP is the closest entry-closing day that comes before the FP.

Figure 5 shows three examples of PD, FP, and RP on a time line extracted from business processes of M-company with fictitious date and day setting. All example cases are from the same Customs authority with its IFP set to 10 days and TP set to one working day.

M-company is informed of the Customs debt in example [1] (see Figure 5) on Friday the 2nd. Since the IFP is 10 days, the PD is on Monday the 12th. Because TP is one working day, the fund has to be transferred at least one day before the PD. Unfortunately, the day before PD is a Sunday so the FP is moved to Friday the 9th instead. In the case of [1], RP and FP is the same day because Friday is an entry-closing day.

Based on Figure 5 in example [2], the debt is notified on the 1st and the PD is on the 11th. Since PD is on the weekend, TP has to be counted from Friday which is the closest working day. Therefore, FP is put on Thursday the 8th. RP is on the 6th because it is the closest entry-closing day.

The notification is received on Monday the 5th in example [3](see Figure 5). PD is, therefore, on Thursday the 15th. As a result, FP and RP are on the 14th and the 13th respectively since there is no weekend or public holiday on or between them. After the system calculates an FP using the algorithm, the FP is compared with *actual date of payment* (AD) that the finance department transfers a fund to a Customs authority. The following rule clarifies the comparison between the two dates:

```

IF AD is earlier or on the same date as FP
THEN AD complies with the CRs
ELSE AD does NOT comply with the CRs

```

In the case that AD does not comply with the *compliance requirements* (CRs), the finance department will be informed of the non-compliance so that it can amend its process to address the compliance issue.

The design of compliance check subsystem can support both traditional auditing and continuous auditing model. To summarize, the dates of payment can be audited periodically in batch as in traditional auditing. While on real-time basis, the system can be implemented using continuous auditing model. The subsystem can use arriving duty payment information from the finance department as a trigger of each auditing process.

3.3 Adaptation Subsystem

The word ‘adaptive’ in the *adaptive auditing system* refers to the fact that the system can sense changes in its environment. The system later uses the information about the changes to make a decision about its adaptation strategy. Finally, compliance requirements are modified according to the decision.

The adaptation target of *compliance requirement adaptation* subsystems are the CSRs. The subsystem monitors and manages both IFP and TP in each CSR. Adaptation is designed with only a single goal in mind; to make sure that CSR of each Customs authority is as accurate as possible.

Interest Free Period (IFP) checking algorithm	
Term	Description
Notification Date (ND)	The date that the Customs notify the company of the debt. ND is extracted from Customs underpayment notification
Actual Payment Deadline (APD)	The actual latest date possible that the finance department can pay off the Customs debt without being subjected to interest as informed by the Customs authority in the notification of late payment
TP checking algorithm	
Term	Description
Actual Date of Payment (AD)	The date that the finance department transfers a fund to a Customs authority
Actual Received Date (ARD)	The date that the Customs authority receives the fund as informed by the Customs authority in the notification of late payment

Table 2: Inputs for Interest Free Period (IFP) and Transfer Period (TP) checking algorithms

Ideally, the IFP setting should be the same as that prescribed by the Customs code. However, this information is sometimes not accurate, not available, or changed without notice. On one hand, CSR should not be longer than the actual period that the Customs allows M-company to pay off its Customs debt or else the company may have to pay interest for the debt. On the other hand, CSR should neither be shorter than the period prescribed by the authority. This is because M-company wants to pay off the Customs debt as late as possible in order to maximize its liquidity benefit. Later the company transfers the fund, the longer the company can hold on to that fund which can be used for other financial purposes. Therefore, this IFP period is comparable to a credit period granted to M-company by the Customs authority. This principle also applies to the adaptation mechanism of TP.

In order to start the adaptation process, the CR Adaptation subsystem needs to be triggered by a notification of late payment from a Customs authority (see Figure 2). Consequently, the subsystem investigates the CSR which belongs to that authority to check the accuracy of the current IFP and TP. The checking mechanism of IFP and that of TP are, in fact, independent from each other and are performed using different algorithm. Table 2 lists the inputs for both algorithms.

- **Interest Free Period (IFP) checking algorithm:**

```
LET NEWIFP equals the number of days between APD 'AND' ND
IF NEWIFP is NOT equal to IFP
THEN LET IFP equals NEWIFP
```

In IFP checking algorithm, the new IFP value is equal to the number of days between actual payment deadline (APD) and notification deadline (ND) (the count includes AD). The new value of IFP is then compared to the current value of IFP. If the new value is not equal to the original value, the system concludes the original IFP is inaccurate and replace it with the new IFP instead.

- **Transfer Period (TP) checking algorithm:**

```
LET NEWTP equals the number of working days between ARD
'AND' AD
IF NEWTP NOT is equal to TP AND NEWTP is less than or equal to 5 working days
THEN LET TP equals NEWTP
```

The new value of TP is calculated by counting the total number of working days between ARD and AD (the count includes AD). Later, the new and the original TP are compared whether the numbers

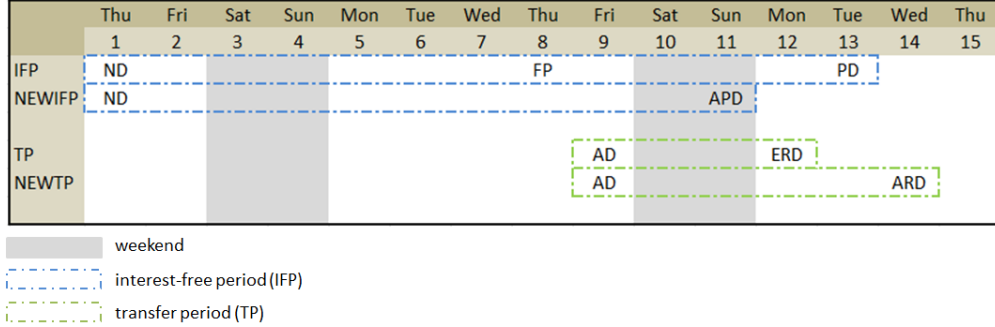


Figure 6: An example of Interest Free Period (IFP) and Transfer Period (TP) checking (IFP is 12 days and TP is 1 working days)

of working days are different. If the answer is yes, the original value is replaced by the new value. The system also ensures that TP is not more than 5 working days as prescribed by the constraints. The difference of RD and ARD counts only the number of working days because the *transfer period* (TP) only includes working days.

Figure 6 shows an example of inaccurate IFP and TP. In the example, the original IFP and TP of this particular Customs were set to 12 days and 1 working day respectively. On the contrary, the adjusted IFP (NEWIFP) is 10 days (calculated using APD and ND) and the adjusted TP (NEWTTP) is 3 working days (calculated using ARD and AD).

4 Adaptive Auditing System Validation

For *adaptive auditing system*, an experimental simulation has been designed which mimics the solution design processes explained in previous Section 3. We will introduce the setup of this simulation and gives explanations of all the test cases used in this simulation.

4.1 Simulation Configuration

The solution artefact simulation is executed by an application called *Adaptive Auditing Simulator*. The first task of this simulator is to load sets of data into the system. This set of input data is called ‘case’, which represents an individual import activity. The inputs include name of the Customs office, ND, AD, APD, and ARD. Adaptive auditing simulator is designed to take in input data in batch, meaning that a number of cases are loaded into the system at the same time. Later, the simulator starts executing the cases in its two subsystems. Despite the fact that the solution design does not require the two subsystems, compliance check and CR adaptation, to be executed at the same time; the simulator is implemented in a way that CR Adaptation process starts right after compliance check in every case. This is to simplify the preparation of the test data.

Adaptive auditing system simulator is implemented in Java programming language. The source code of the simulator is shown in Appendix A(Section A.1). The simulator is programmed in a class-based manner. These classes are explained below:

- AdaptiveAuditingSystem: Both compliance check and CR adaptation logics are embedded in this class. These two subsystems produce log files called audit_result.txt and adapting_result.txt that contain the result of the simulation. AdaptiveAuditingSystem class is the only class in this application that contains a main method which is a standard method used to start execution of any Java program. Additionally, *Customs specific requirements* values are coded directly in this class so that they can be easily modified during the validation process;

- CSR: This class defines the two properties of *Customs specific requirements*: IFP and TP. The class also provides a constructor which allows the system to create a new *Customs specific requirement* with default IFP and TP values whenever there is no corresponding *Customs specific requirement* for the Customs being audited;
- TestData: To simplify the way the system loads input data, inputs for each test case are coded directly in this class. The class also provides methods for AdaptiveAuditingSystem to gain access to the different elements of the case and check if there is a late-payment notification;
- Logwriter: Log writer provides a method for AdaptiveAuditingSystem to write log files which contain the result of the simulation;
- Constraint: All the constraint-type requirements are defined in this class. These include definition of working days, definition of entry-closing days, and the upper limit of TP.

4.1.1 Test Cases

This section explains all the test case scenarios in the simulation. The word ‘test case’ is not to be confused with the term ‘case’, which represents an individual import activity. A test case, on the other hand, is a set of cases under which the simulation will determine whether the system is working as it was originally predicted. The test cases are divided into two main groups according to the subsystems in *adaptive auditing system*. The first group of test cases are designed to validate compliance check subsystem while the other group is formulated to validate CR adaptation subsystem. All the test data used in this simulation are listed in Appendix A(Section B).

4.2 Validation of Compliance Check Subsystem

To validate compliance check subsystem, a number of criteria are set up for the test data. The test data for the compliance check subsystem are composed of 50 cases, in which their *notification dates* (NDs) are spread throughout 2014. Out of these 50 cases: 25 cases contain *actual dates of payment* (ADs) which comply with the CRs while the other 25 cases contain ADs which do not comply with the CRs. Each compliance check test case consists of a sub-population of these cases. Sub-populations in different test cases may overlap which means a case can be used in verification of more than one test cases. The following lists all the test cases related to compliance check subsystem:

4.2.1 Test Case 1: Customs-Specific Requirements (CSR) Setting

The goal of this test case is to verify the accuracy of the PD, FP, and RP calculation algorithm given different CSR settings for different cases. Table 3 shows CSRs setting for Customs A, B, and C used in this test case. Each of these CSR setting is assigned to 10 cases, making it 30 cases in total. Examples of these cases are illustrated in Figure 7 as A, B, and C. In order to check the validity of this simulation, the following outputs are observed:

- Payment Deadline (PD): PD has to be equal to ND plus the number of IFP dates;
- Final Date of Payment (FP): Since FP is calculated using TP, the test data include various scenarios where TPs are over holiday and weekend periods. Sample A and B in Figure 7 show the cases in which the TPs are over weekends while sample C gives an example of a case where TP doesn’t include weekend;
- Recommended Date of Payment (RP): Since RP has to fall on one of the *entry-closing days*(ECs), which are Tuesday and Friday, the test data have to cover both the scenarios that RPs are on Tuesday and the cases that RPs are on Friday. Sample A and B in Figure 7 show the cases in which the RPs fall on Fridays while sample C gives an example of a case where RP is on Tuesday.

	Customs	A	B	C	D	E
	IFP	15	12	10	N/A	N/A
	TP	3	0	1	N/A	N/A
Tue	12-Aug			ND	ND	
Wed	13-Aug					
Thu	14-Aug					
Fri	15-Aug					
Sat	16-Aug					
Sun	17-Aug					
Mon	18-Aug					
Tue	19-Aug			RP		
Wed	20-Aug					
Thu	21-Aug			FP		
Fri	22-Aug			AD,PD		
Sat	23-Aug					
Sun	24-Aug					
Mon	25-Aug		ND			ND
Tue	26-Aug	ND			RP	
Wed	27-Aug				PD & FP	
Thu	28-Aug					
Fri	29-Aug				AD	
Sat	30-Aug					
Sun	31-Aug					
Mon	01-Sep					
Tue	02-Sep					
Wed	03-Sep					
Thu	04-Sep					
Fri	05-Sep	FP,RP,AD	FP, RP			AD
Sat	06-Sep		PD			
Sun	07-Sep					
Mon	08-Sep					
Tue	09-Sep		AD			PD,FP,RP
Wed	10-Sep	PD				
Thu	11-Sep					

Figure 7: Samples of compliance check cases on a timeline

Customs Name	IFP	TP
A	15	3
B	12	0
C	10	1

Table 3: *Customs specific requirements* setting for Customs A, B, and C

4.2.2 Test Case 2: Default Customs-Specific Requirements (Default CSR) Setting

This test case verifies that new *Customs specific requirements* are initiated correctly according to the default CSR setting; in the case that CSRs are not available for specific Customs authorities. Additionally, this test case verifies that these newly-created *Customs specific requirements* can be used to calculate the PD, FP, and RP. The IFP and TP of the default CSR are set to 15 and 0 respectively in this simulation. In this test case, 10 cases from Customs D and another 10 cases from Customs E are configured to require CSR initiation. Examples of these cases are illustrated in Figure 7 as D and E. Similar to *Customs specific requirement* setting test case, the following outputs are observed:

- Payment Deadline (PD): PD has to be equal to ND plus the number of IFP dates;
- Final Date of Payment (FP): Since FP is calculated using TP, the test data include various scenarios where TPs are over holiday and weekend periods;
- Recommended Date of Payment (RP): Since RP has to fall on one of the entry-closing days (ECs) which are Tuesday and Friday, the test data have to cover both scenarios namely RPs are on Tuesday and that RPs are on Friday.

4.2.3 Test Case 3: Compliance Validation

The aim of compliance validation test case is to verify the accuracy of the compliance check mechanism. On one hand, FPs in 25 cases are pre-set to be compliant according to their ADs and *Customs specific requirement* settings (as shown in sample A and E in Figure 7). On the other hand, FPs in another 25 cases are configured to be non-compliant when compared to their ADs (as shown in sample B, C, and, D in Figure 7). The compliance check mechanism is proved to be valid if the results from all the test data match the predefined outcomes.

4.3 Validation of Adaptation Subsystem

The test data for validation of adaptation subsystem, contrary to the individual cases used in compliance check validation, are groups of carefully selected cases referred to as ‘case sets’. The following describes the composition of a case set:

- Each case set is composed of six cases;
- The first case in each case set is called ‘control case’. It represents the situation in which IFP, TP, or both of them are manipulated in the simulation in order to observe the effect on the other cases in the same case set;
- The other five cases in the same case set are referred to as ‘outcome cases’. These cases represent import activities with different Customs authorities, namely A, B, C, D, and E. The IFPs, TPs, as well as compliance check results of the outcome cases are observed in order to analyse the consequences of the control case adaptation.

In this simulation, there are 15 ($15 \times 6 = 90$ cases) case sets in total. To ensure that the algorithm works with different CSR setting, these case sets are divided into five groups with three case sets in each group. The control cases in different group are assigned with different Customs authorities. The Customs settings of A,

CN	ND	AD	LP	APD	ARD	IFP		TP	
						Before	After	Before	After
D	24/11	09/12	Yes	06/12	12/12	15	12	0	3
D	25/11	09/12	No	N/A	N/A	12	12	3	3
A	08/07	18/07	No	N/A	N/A	15	15	3	3
B	03/04	15/04	No	N/A	N/A	12	12	0	0
C	26/09	03/10	No	N/A	N/A	10	10	1	1
E	25/08	09/09	No	N/A	N/A	15	15	0	0

Table 4: A sample case set for CR adaptation validation (CN stands for *Customs name* while LP represent *late payment*)

B, and C are shown in Table 3 while D and E use default CSR setting (IFP is 15 days and TP is 0 working days). Table 4 shows a sample of a case set where its control case belongs to Customs D. Both IFP and TP of the control case are incorrect. Therefore, the IFP of the Customs D is changed 12 and the TP to 3 after adjustment. If the simulation is working as it was originally designed to; the compliance check in the first outcome case, which also belongs to Customs D, should use the adjusted IFP and TP in its checking process. On the contrary, the compliance check in the rest of the outcome cases, which belong to other Customs offices, should not be affected by the adjustment. The following lists all the test cases related to CR adaptation subsystem:

4.3.1 Test Case 4: IFP Adjustment

The goal of this test case is to verify if the adaptation subsystem can adjust IFP correctly according to case-specific ND and APD. Out of the 15 case sets, 10 of them are designed to verify IFP adaptation mechanism. In this test case, the following are observed:

- The IFP of the corresponding Customs authority must be adjusted correctly;
- The IFP of Customs authorities, which are not part of the control case, must not be affected by the adjustment;
- The outcome case that belongs to the same Customs as the control case must use the adjusted IFP in its compliance check process.

4.3.2 Test Case 5: TP Adjustment

The goal of this test case is to verify if the adaptation subsystem can adjust TP correctly according to case-specific AD and ARD. Ten of the case sets are designated to verify TP adaptation mechanism. In this test case, the following are observed:

- The TP of the corresponding Customs authority must be adjusted correctly;
- The TP of Customs authorities, which are not part of the control case, must not be affected by the adjustment;
- The outcome case that belongs to the same Customs as the control case must use the adjusted TP in its compliance check process;
- TP upper limit constraint must be enforced. In this simulation, the upper limit for TP is set to 5. The simulator must not modify TP value if the calculated new value exceeds five.

Test Case	Result
TC1: <i>Customs specific requirement</i> setting	<ul style="list-style-type: none"> • PD of all 30 cases were calculated correctly; • FP of all 30 cases were calculated correctly; • RP of all 30 cases were calculated correctly
TC2: Default CSR setting	<ul style="list-style-type: none"> • PD of all 20 cases were calculated correctly; • FP of all 20 cases were calculated correctly; • RP of all 20 cases were calculated correctly
TC3: Compliance validation	<ul style="list-style-type: none"> • compliance check simulation result of all 50 cases were correct

Table 5: Compliance check validation result

4.3.3 Test Case 6: IFP and TP Adjustment In-Dependency

The goal of this test case is to ensure that IFP and TP adjustments are independent. In other word, even if both IFP and TP need to be modified in the same case; the two adaptation mechanisms must not interfere with each other. Five of the case sets are set to verify this in-dependency. In this test case, the following are observed:

- The IFP and TP of the corresponding Customs authority must be adjusted correctly;
- The IFP and TP of Customs authorities, which are not part of the control case, must not be affected by the adjustment;
- The outcome case that belongs to the same Customs as the control case must use the adjusted IF and TP in its compliance check process;
- TP upper limit constraint must be enforced. In this simulation, the upper limit for TP is set to 5. Therefore, the simulator must not modify TP value if the new value exceeds five.

4.4 Discussion

For the test cases in the compliance check group, the results are shown in Table 5. In test case 1, all 30 PDs, FPs, and RPs were calculated correctly. A similar result is also obtained in test case 2 where PDs, FPs, and RPs of the cases with default CSR setting were all proved to be correct. Finally, the simulator could check the compliance of all 50 cases correctly. In test case 4, 5, and 6, a few inaccurate *Customs specific requirements* could not be detected (see Table 6). The reason behind is the *actual dates of payment* (ADs) come before *actual received dates* (ARDs) in these cases, which means the Customs authorities actually receive the Customs debt payments before the deadlines. Therefore, the authorities do not send notification of late payment to the M-company since there is no late payment in the first place. After further analysis, we found out that the gap between *recommended date of payment* (RP) and *final date of payment* (FP) in some cases were too big because FP was far away from the closest entry-closing date. The cause of this problem was not a flaw in the design artefact. On the contrary, it was the result of the limited scope of the case study. In the case study, M-company can only gain access to *actual payment deadline* (APD) and *actual received date* (ARD) from the notification of late payment. In the real world, however, a company might be able

Test Case	Result
TC4: IFP adjustment	<ul style="list-style-type: none"> • In three case sets, the IFP inaccuracy in the control case could not be detected. This was due to the lack of notification of late payment from the corresponding Customs authority; • The IFPs of Customs authorities, which were not part of the control case, were not affected by control case IFP adjustment; • In all the detected case sets, the outcome case that belongs to the same Customs as the control case used the adjusted IFP in its compliance check process
TC5: TP adjustment	<ul style="list-style-type: none"> • In one case set, the TP inaccuracy in the control case could not be detected. This was due to the lack of notification of late payment from the corresponding Customs authority; • The TPs of Customs authorities, which were not part of the control case, were not affected by control case TP adjustment; • In all the detected case sets, the outcome case that belongs to the same Customs as the control case used the adjusted IFP in its compliance check process • All new TPs that exceed 5 working days were detected.
TC6: IFP and TP adjustment in-dependency	<ul style="list-style-type: none"> • In two case sets, the <i>Customs specific requirement</i> inaccuracy in the control case was undetected. This was due to the lack of notification of late payment from the Customs authority; • The IFPs and TPs of Customs authorities, which were not part of the control case, were not affected by control case adjustment; • In all the detected case sets, the outcome case that belongs to the same Customs as the control case used the adjusted <i>Customs specific requirement</i> in its compliance check process

Table 6: CR adaptation validation result

to obtain this information by other means such as inquiring APD directly from the authority or extracting ARD from duty payment receipt. If that is the case, the company can detect and adapt inaccurate *Customs specific requirement* in every case without a trigger from notification of late payment.

In all other case sets that *Customs specific requirement* inaccuracy were detected, the adaptation of both IFP and TP went as expected. The adjusted *Customs specific requirements* were used in the outcome cases that belong to the same Customs as the control case while the *Customs specific requirements* of Customs authorities, which were not part of the control cases, were not affected by control case adjustments; Finally, TP upper limit constrain was enforced effectively in all the cases.

5 Conclusion

In this paper we have developed an adaptive system for compliance management in cross-border logistic. Design of this specific system can be used with other domain as well. To evaluate the flexibility in *adaptive auditing system*, we have mapped it to *smart auditing framework* as can be seen from Figure 2 and Figure 3). Design artefact is made up of two main components. The first component is compliance check subsystem. Its goal is to verify if outstanding Customs debt is paid on time. The other component is compliance

requirement adaptation subsystem. This subsystem aim is to observe the accuracy of CSRs. If these CSRs are inaccurate, the subsystem must correct them accordingly.

Adaptive auditing system, offer numerous benefits compared to conventional audit method. First of all, it can potentially help to reduce costs. The automation of both compliance checking and rule adjustment can remove most of the human involvement from the operation. This brings down the required number of staffs and increases efficiency of the process at the same time. The rest of the staff will play more passive roles such as monitoring instead of involving directly in auditing process. By reducing human involvement, it also helps minimize possible errors introduced by human. This system can help decrease cost further by keeping CSRs as precise as possible using its adaptation mechanism. Maintaining accurate CSRs results reduce the chance of late duty payment, hence, lowering the amount of interest charged on the Customs debt. Moreover, since late payment incidents may damage the *reputation* of the organization; it is reasonable for the company to try to pay all the debts on time in order to maintain a good credit history. The accuracy of CSRs also ensures that the company can pay off the Customs debt as late as the corresponding authority allows in order to maximize its liquidity benefit. This extended payment period can be considered as an extra credit period for the company.

References

- Belastingdienst (2014). Customs procedure for release into free circulation. http://www.belastingdienst.nl/wps/wcm/connect/bldcontenten/belastingdienst/customs/import/customs_procedures_for_import/release_into_free_circulation.
- El Kharbili, M. (2012). Business process regulatory compliance management solution frameworks: A comparative evaluation. In *Proceedings of the 8th Asia-Pacific Conference on Conceptual Modelling-Volume 130*, pages 23–32. Australian Computer Society, Inc.
- European Commission (1992). Council regulation (EEC) no 2913/92 of 12 october 1992 establishing the community customs code. *Official Journal of the European Union*.
- European Commission (2008). Regulation (Ec) no 450/2008 of The European parliament and of the council of 23 april 2008 laying down the community customs code (modernised customs code). *Official Journal of the European Union*, 51:L145/1–L145/64.
- European Commission (2013). *The European Union Explained-How the European Union Works*. The European Union Directorate-General for Communication Publications, Belgium.
- European Commission (2014). European Union customs policy issues. http://ec.europa.eu/taxation_customs/customs/policy_issues.
- European Union (2004). Modern custom code. available at: http://ec.europa.eu/taxation_customs/resources/documents/458rev_en.pdf retrieved august 2014.
- McNally, J. (2013). The 2013 COSO framework and SOX compliance: One approach to an effective transition. *Strategic Finance*.
- Sakr, S. and Awad, A. (2010). A framework for querying graph-based business process models. In *Proceedings of the 19th international conference on World wide web*, pages 1297–1300. ACM.
- Silveira, P., Rodriguez, C., Birukou, A., Casati, F., Daniel, F., D’Andrea, V., Worledge, C., and Taheri, Z. (2011). Aiding compliance governance in service-based business processes. *Non-Functional Properties for Service-Oriented Systems: Future Directions*, 16:17–75.
- Yates, J. and Murphy, C. N. (2007). Coordinating international standards : the formation of the ISO. Sloan working papers, MIT Sloan School of Management, Cambridge, Massachusetts, USA.

A Simulation Code & Test Data: Adaptive Auditing System

A.1 Adaptive Auditing System

```
package adaptiveauditingssystem;

import java.io.IOException;
import java.text.DateFormat;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.List;

public class AdaptiveAuditingSystem {

    public static void main(String[] args) throws ParseException, IOException{
        DateFormat formatter = new SimpleDateFormat("dd/MM/yy");
        TestData test_data = new TestData();
        LogWriter audit_result = new LogWriter("audit_result.txt");
        LogWriter adapting_result = new LogWriter("adapting_result.txt");
        List<CSR> CSR_list = new ArrayList<CSR>();
        CSR_list.add(new CSR("A", 15, 3));
        CSR_list.add(new CSR("B", 12, 0));
        CSR_list.add(new CSR("C", 10, 1));
        Constraint constraint = new Constraint();

        String customs_name;
        CSR current_CSR, default_CSR;
```

```

int IFP, TP, temp_IFP, temp_TP, counter;
Calendar AD, ND, PD, FP, RP, ARD, APD, date_pointer;

for(int i=0; i<test_data.getNumberOfCase(); i++){
//START Compliance Check
IFP = -1; //reset IFP
TP = -1; //reset IFP
counter = 0; //reset counter
customs_name = test_data.getCustomsName(i);
for(int j=0; j<CSR_list.size();j++){
current_CSR = CSR_list.get(j);
if(current_CSR.getCustomsName().equals(customs_name)){
IFP = current_CSR.getIFP();
TP = current_CSR.getTP();
break;
}
}
//create a new CSR using default constructor
if(IFP == -1){
default_CSR = new CSR(customs_name);
CSR_list.add(default_CSR);
IFP = default_CSR.getIFP();
TP = default_CSR.getTP();
}
ND = (Calendar) test_data.getND(i).clone();
AD = (Calendar) test_data.getAD(i).clone();
date_pointer = (Calendar) ND.clone();
date_pointer.add(Calendar.DATE, IFP);
PD = (Calendar) date_pointer.clone();
while(counter<TP){
while(!constraint.isWorkingDay(date_pointer)){
date_pointer.add(Calendar.DATE, -1);
}
date_pointer.add(Calendar.DATE, -1);
counter++;
}
while(!constraint.isWorkingDay(date_pointer)){
date_pointer.add(Calendar.DATE, -1);
}
FP = (Calendar) date_pointer.clone();

while(!constraint.isECDay(date_pointer)||
!constraint.isWorkingDay(date_pointer)){
date_pointer.add(Calendar.DATE, -1);
}
RP = (Calendar) date_pointer.clone();
audit_result.writeLog("Case : " + (i + 1), 1);
if(AD.after(FP)){
//non-compilant
audit_result.writeLog("result: non-compliant", 1);
}
else{
//compliant
audit_result.writeLog("result: compliant", 1);
}
audit_result.writeLog("customs: " + customs_name, 1);
audit_result.writeLog("IFP = " + IFP, 1);
audit_result.writeLog("TP = " + TP, 1);
audit_result.writeLog("AD = " + formatter.format(AD.getTime()), 1);
audit_result.writeLog("ND = " + formatter.format(ND.getTime()), 1);
audit_result.writeLog("PD = " + formatter.format(PD.getTime()), 1);
audit_result.writeLog("FP = " + formatter.format(FP.getTime()), 1);
audit_result.writeLog("RP = " + formatter.format(RP.getTime()), 2);
//END Compliance Check

//START CR Adaptation

if(test_data.checkLatePayment(i)){
APD = (Calendar) test_data.getAPD(i).clone();
ARD = (Calendar) test_data.getARD(i).clone();
if(ARD.after(APD)){
adapting_result.writeLog("customs: " + customs_name, 1);
//START IFP Adaptation
temp_IFP = (int) ((APD.getTimeInMillis()-ND.getTimeInMillis())/86400000);
if(IFP!=temp_IFP){
IFP = temp_IFP;
adapting_result.writeLog("IFP has been changed to " + IFP, 1);
}
}
}

```

```

}
//END IFP Adaptation

//START TP Adaptation
date_pointer = (Calendar) ARD.clone();
temp_TP = (int) ((ARD.getTimeInMillis()-AD.getTimeInMillis())/86400000);
for(int k = 0; k<temp_TP; k++){
date_pointer.add(Calendar.DATE, -1);
if(!constraint.isWorkingDay(date_pointer)){
temp_TP--;
}
}
if(TP!=temp_TP){
if(constraint.checkTP(temp_TP)){
TP = temp_TP;
adapting_result.writeLog("TP has been changed to " + TP, 1);
}
else{
adapting_result.writeLog("Error: The new TP does not
comply with the constraints", 1);
}
}
}
//Modify CSR
for(CSR new_CSR: CSR_list){
if(customs_name.equals(new_CSR.getCustomsName())){
new_CSR.setIFP(IFP);
new_CSR.setTP(TP);
break;
}
}
//END TP Adaptation
}
//END CR Adaptation
}
audit_result.close();
adapting_result.close();
}
}

```

A.2 Constraints

```

package adaptiveauditingssystem;

import java.text.DateFormat;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.GregorianCalendar;

public class Constraint {
private String[] public_holiday;
Calendar calendar_date = new GregorianCalendar();
DateFormat formatter = new SimpleDateFormat("dd/MM/yy");
Date the_date;

public Constraint(){
public_holiday = specifyPublicHolidays();
}

private String[] specifyPublicHolidays()
{
String[] generalHolidays = new String[]{"01/01/", "18/04/", "21/04/",
"29/05/", "09/06/", "05/12/",
"25/12/", "26/12/"};
String currentYear =String.valueOf(Calendar.getInstance().get(Calendar.YEAR));
for(int i=0; i<= generalHolidays.length-1; i++)
{
generalHolidays[i]= generalHolidays[i]+currentYear;
}
return generalHolidays;
}
public Boolean isWorkingDay(Calendar date) throws ParseException{
if(date.get(Calendar.DAY_OF_WEEK) == Calendar.SUNDAY){

```

```

return false;
}
else if(date.get(Calendar.DAY_OF_WEEK) == Calendar.SATURDAY){
return false;
}
else{
for(int i=0; i<public_holiday.length; i++){
the_date = formatter.parse(public_holiday[i]);
calendar_date.setTime(the_date);
if(calendar_date.compareTo(date)==0){
return false;
}
}
}
return true;
}

public Boolean isECCDay(Calendar date) throws ParseException{
if(date.get(Calendar.DAY_OF_WEEK) == Calendar.TUESDAY){
return true;
}
else if(date.get(Calendar.DAY_OF_WEEK) == Calendar.FRIDAY){
return true;
}
return false;
}

public Boolean checkTP(int TP){
if(TP>5){
return false;
}
return true;
}
}

```

A.3 Test Data

```

package adaptiveauditingssystem;

import java.text.DateFormat;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.GregorianCalendar;

public class TestData {
private String[] [] test_data;
Calendar calendar_date = new GregorianCalendar();
DateFormat formatter = new SimpleDateFormat("dd/MM/yy");
Date the_date;

public TestData(){
//customs_name, ND, AD, APD, ARD
test_data = new String[] []{
{"E", "07/03/2014", "21/03/2014", "17/03/14", "28/03/14"},
{"E", "12/03/2014", "25/03/2014", "N/A", "N/A"},
{"A", "22/09/2014", "30/09/2014", "N/A", "N/A"},
{"B", "17/12/2014", "23/12/2014", "N/A", "N/A"},
{"C", "20/11/2014", "25/11/2014", "N/A", "N/A"},
{"D", "24/01/2014", "07/02/2014", "N/A", "N/A"},
};
}

public int getNumberOfCase(){
return test_data.length;
}

public String getCustomsName(int case_no){
return test_data[case_no][0];
}

public Calendar getND(int case_no) throws ParseException{
the_date = formatter.parse(test_data[case_no][1]);
calendar_date.setTime(the_date);
}
}

```



```

return calendar_date;
}

public Calendar getAD(int case_no) throws ParseException{
the_date = formatter.parse(test_data[case_no][2]);
calendar_date.setTime(the_date);
return calendar_date;
}

public Calendar getAPD(int case_no) throws ParseException{
the_date = formatter.parse(test_data[case_no][3]);
calendar_date.setTime(the_date);
return calendar_date;
}

public Calendar getARD(int case_no) throws ParseException{
the_date = formatter.parse(test_data[case_no][4]);
calendar_date.setTime(the_date);
return calendar_date;
}

public Boolean checkLatePayment(int case_no){
return !test_data[case_no][3].equals("N/A");
}
}

```

A.4 CSR

```

package adaptiveauditingssystem;

public class CSR {
private String customs;
private int IFP;
private int TP;
//Known customs
public CSR(String customs_name, int IFP, int TP){
customs = customs_name;
this.IFP = IFP;
this.TP = TP;
}
//Default CSR
public CSR(String customsName){
customs = customsName;
IFP = 15;
TP = 0;
}

public String getCustomsName(){
return customs;
}

public int getIFP(){
return IFP;
}

public int getTP(){
return TP;
}

public void setIFP(int IFP){
this.IFP = IFP;
}

public void setTP(int TP){
this.TP = TP;
}
}

```

A.5 Log Writer

```

package adaptiveauditingssystem;

```

```

import java.io.BufferedWriter;
import java.io.File;
import java.io.FileWriter;
import java.io.IOException;

public class LogWriter {
    BufferedWriter writer;

    public LogWriter(String file_name) throws IOException{
        writer = new BufferedWriter(new FileWriter(file_name));
    }

    public void writeLog(String result, int new_line) throws IOException{
        writer.write(result);
        for(int i=0; i<new_line; i++){
            writer.newLine();
        }
    }

    public void close() throws IOException{
        writer.close();
    }
}

```

B Compliance Check Test Data

Case no.	Customs name	ND	AD	IFP	TP	Expected result
1	A	06-01-14	14-01-14	15	3	compliant
2	A	08-01-14	21-01-14	15	3	non-compliant
3	A	27-05-14	03-06-14	15	3	compliant
4	A	02-06-14	13-06-14	15	3	non-compliant
5	A	03-06-14	13-06-14	15	3	compliant
6	A	08-07-14	18-07-14	15	3	compliant
7	A	14-07-14	22-07-14	15	3	compliant
8	A	26-08-14	05-09-14	15	3	compliant
9	A	22-09-14	03-10-14	15	3	non-compliant
10	A	02-12-14	12-12-14	15	3	compliant
11	B	29-01-14	14-02-14	12	0	non-compliant
12	B	10-02-14	21-02-14	12	0	compliant
13	B	26-02-14	07-03-14	12	0	compliant
14	B	03-04-14	18-04-14	12	0	non-compliant
15	B	04-06-14	17-06-14	12	0	non-compliant
16	B	14-07-14	29-07-14	12	0	non-compliant
17	B	25-08-14	09-09-14	12	0	non-compliant
18	B	17-09-14	26-09-14	12	0	compliant
19	B	27-10-14	11-11-14	12	0	non-compliant
20	B	17-12-14	23-12-14	12	0	compliant
21	C	20-01-14	28-01-14	10	1	compliant
22	C	21-02-14	07-03-14	10	1	non-compliant
23	C	03-03-14	11-03-14	10	1	compliant
24	C	09-04-14	22-04-14	10	1	non-compliant
25	C	30-04-14	09-05-14	10	1	non-compliant
26	C	08-05-14	13-05-14	10	1	compliant
27	C	12-08-14	22-08-14	10	1	non-compliant
28	C	26-09-14	03-10-14	10	1	compliant
29	C	15-10-14	24-10-14	10	1	non-compliant
30	C	20-11-14	25-11-14	10	1	compliant
31	D	24-01-14	07-02-14	N/A		N/A compliant
32	D	14-02-14	04-03-14	N/A	N/A	non-compliant
33	D	04-03-14	21-03-14	N/A	N/A	non-compliant
34	D	12-05-14	27-05-14	N/A	N/A	compliant
35	D	13-05-14	30-05-14	N/A	N/A	non-compliant
36	D	07-07-14	25-07-14	N/A	N/A	non-compliant
37	D	12-08-14	29-08-14	N/A	N/A	non-compliant
38	D	15-10-14	28-10-14	N/A	N/A	compliant
39	D	24-11-14	09-12-14	N/A	N/A	compliant
40	D	25-11-14	12-12-14	N/A	N/A	non-compliant
41	E	07-03-14	25-03-14	N/A	N/A	non-compliant
42	E	12-03-14	25-03-14	N/A	N/A	compliant
43	E	28-03-14	11-04-14	N/A	N/A	compliant
44	E	31-03-14	15-04-14	N/A	N/A	compliant
45	E	29-04-14	16-05-14	N/A	N/A	non-compliant

Case no.	Customs name	ND	AD	IFP	TP	Expected result
46	E	23-05-14	10-06-14	N/A	N/A	non-compliant
47	E	25-08-14	05-09-14	N/A	N/A	compliant
48	E	15-09-14	03-10-14	N/A	N/A	non-compliant
49	E	08-10-14	21-10-14	N/A	N/A	compliant
50	E	01-12-14	19-12-14	N/A	N/A	non-compliant

B.1 CR Adaptation Test Data

Case Set 1

Customs Name	ND	AD	Late	APD	ARD	IFP		TP	
						Before	After	Before	After
A	06-01-14	14-01-14	No	20-01-14	17-01-14	15	15	3	3
A	06-01-14	14-01-14	No	20-01-14	17-01-14	15	15	3	3
B	17-09-14	26-09-14	No	N/A	N/A	12	12	0	0
C	12-08-14	19-08-14	No	N/A	N/A	10	10	1	1
D	07-07-14	22-07-14	No	N/A	N/A	15	15	0	0
E	29-04-14	13-05-14	No	N/A	N/A	15	15	0	0

Case Set 2

Customs Name	ND	AD	Late	APD	ARD	IFP		TP	
						Before	After	Before	After
A	03-06-14	13-06-14	Yes	18-06-14	19-06-14	15	15	3	4
A	14-07-14	22-07-14	No	N/A	N/A	15	15	4	4
B	04-06-14	13-06-14	No	N/A	N/A	12	12	0	0
C	03-03-14	11-03-14	No	N/A	N/A	10	10	1	1
D	14-02-14	28-02-14	No	N/A	N/A	15	15	0	0
E	23-05-14	06-06-14	No	N/A	N/A	15	15	0	0

Case Set 3

Customs Name	ND	AD	Late	APD	ARD	IFP		TP	
						Before	After	Before	After
A	02-12-14	12-12-14	Yes	16-12-14	18-12-14	15	14	3	4
A	26-08-14	05-09-14	No	N/A	N/A	14	14	4	4
B	17-12-14	23-12-14	No	N/A	N/A	12	12	0	0
C	09-04-14	15-04-14	No	N/A	N/A	10	10	1	1
D	04-03-14	18-03-14	No	N/A	N/A	15	15	0	0
E	31-03-14	15-04-14	No	N/A	N/A	15	15	0	0

Case Set 4

Customs Name	ND	AD	Late	APD	ARD	IFP		TP	
						Before	After	Before	After
B	26-02-14	07-03-14	No	08-03-14	07-03-14	12	12	0	0
B	03-04-14	15-04-14	Yes	13-04-14	15-04-14	12	10	0	0
A	27-05-14	03-06-14	No	N/A	N/A	15	15	3	3
C	21-02-14	28-02-14	No	N/A	N/A	10	10	1	1
D	13-05-14	27-05-14	No	N/A	N/A	15	15	0	0
E	08-10-14	21-10-14	No	N/A	N/A	15	15	0	0

Case Set 5

Customs Name	ND	AD	Late	APD	ARD	IFP		TP	
						Before	After	Before	After
B	29-01-14	07-02-14	Yes	10-02-14	11-02-14	12	12	0	2
B	10-02-14	21-02-14	No	N/A	N/A	12	12	2	2
A	02-06-14	10-06-14	No	N/A	N/A	15	15	3	3
C	08-05-14	13-05-14	No	N/A	N/A	10	10	1	1
D	15-10-14	28-10-14	No	N/A	N/A	15	15	0	0
E	15-09-14	30-09-14	No	N/A	N/A	15	15	0	0

Case Set 6

Customs Name	ND	AD	Late	APD	ARD	IFP		TP	
						Before	After	Before	After
B	25-08-14	05-09-14	Yes	06-09-14	16-09-14	12	10	0	0
B	14-07-14	25-07-14	No	N/A	N/A	10	10	0	0
A	22-09-14	30-09-14	No	N/A	N/A	15	15	3	3
C	26-09-14	03-10-14	No	N/A	N/A	10	10	1	1
D	12-05-14	27-05-14	No	N/A	N/A	15	15	0	0
E	01-12-14	16-12-14	No	N/A	N/A	15	15	0	0

Case Set 7

Customs Name	ND	AD	Late	APD	ARD	IFP		TP	
						Before	After	Before	After
C	20-01-14	28-01-14	Yes	28-01-14	29-01-14	10	8	1	1
C	30-04-14	06-05-14	No	N/A	N/A	8	8	1	1
A	08-01-14	17-01-14	No	N/A	N/A	15	15	3	3
B	27-10-14	07-11-14	No	N/A	N/A	12	12	0	0
D	12-08-14	26-08-14	No	N/A	N/A	15	15	0	0
E	07-03-14	21-03-14	No	N/A	N/A	15	15	0	0

Case Set 8

Customs Name	ND	AD	Late	APD	ARD	IFP		TP	
						Before	After	Before	After
C	20-11-14	25-11-14	Yes	30-11-14	03-12-14	10	10	1	1
C	15-10-14	21-10-14	No	N/A	N/A	10	10	1	1
A	08-07-14	18-07-14	No	N/A	N/A	15	15	3	3
B	04-06-14	13-06-14	No	N/A	N/A	12	12	0	0
D	25-11-14	09-12-14	No	N/A	N/A	15	15	0	0
E	25-08-14	09-09-14	No	N/A	N/A	15	15	0	0

Case Set 9

Customs Name	ND	AD	Late	APD	ARD	IFP		TP	
						Before	After	Before	After
C	30-04-14	06-05-14	No	08-05-14	08-05-14	10	10	1	1
C	03-03-14	11-03-14	Yes	11-03-14	13-03-14	10	8	1	2
A	03-06-14	13-06-14	No	N/A	N/A	15	15	3	3
B	17-09-14	26-09-14	No	N/A	N/A	12	12	0	0
D	24-11-14	09-12-14	No	N/A	N/A	15	15	0	0
E	28-03-14	11-04-14	No	N/A	N/A	15	15	0	0

Case Set 10

Customs Name	ND	AD	Late	APD	ARD	IFP		TP	
						Before	After	Before	After
D	24-01-14	07-02-14	Yes	05-02-14	07-02-14	15	12	0	0
D	12-08-14	26-08-14	No	N/A	N/A	12	12	0	0
A	14-07-14	22-07-14	No	N/A	N/A	15	15	3	3
B	10-02-14	21-02-14	No	N/A	N/A	12	12	0	0
C	09-04-14	15-04-14	No	N/A	N/A	10	10	1	1
E	12-03-14	25-03-14	No	N/A	N/A	15	15	0	0

Case Set 11

Customs Name	ND	AD	Late	APD	ARD	IFP		TP	
						Before	After	Before	After
D	12-05-14	27-05-14	Yes	27-05-14	02-06-14	15	15	0	3
D	07-07-14	22-07-14	No	N/A	N/A	15	15	3	3
A	27-05-14	03-06-14	No	N/A	N/A	15	15	3	3
B	14-07-14	25-07-14	No	N/A	N/A	12	12	0	0
C	20-01-14	28-01-14	No	N/A	N/A	10	10	1	1
E	15-09-14	30-09-14	No	N/A	N/A	15	15	0	0

Case Set 12

Customs Name	ND	AD	Late	APD	ARD	IFP		TP	
						Before	After	Before	After
D	24-11-14	09-12-14	Yes	06-12-14	12-12-14	15	12	0	3
D	25-11-14	09-12-14	No	N/A	N/A	12	12	3	3
A	08-07-14	18-07-14	No	N/A	N/A	15	15	3	3
B	03-04-14	15-04-14	No	N/A	N/A	12	12	0	0
C	26-09-14	03-10-14	No	N/A	N/A	10	10	1	1
E	25-08-14	09-09-14	No	N/A	N/A	15	15	0	0

Case Set 13

Customs Name	ND	AD	Late	APD	ARD	IFP		TP	
						Before	After	Before	After
E	29-04-14	13-05-14	Yes	09-05-14	13-05-14	15	10	0	0
E	23-05-14	06-06-14	No	N/A	N/A	10	10	0	0
A	08-01-14	17-01-14	No	N/A	N/A	15	15	3	3
B	29-01-14	07-02-14	No	N/A	N/A	12	12	0	0
C	08-05-14	13-05-14	No	N/A	N/A	10	10	1	1
D	13-05-14	27-05-14	No	N/A	N/A	15	15	0	0

Case Set 14

Customs Name	ND	AD	Late	APD	ARD	IFP		TP	
						Before	After	Before	After
E	08-10-14	21-10-14	Yes	23-10-14	28-10-14	15	15	0	5
E	01-12-14	16-12-14	No	N/A	N/A	15	15	5	5
A	02-06-14	10-06-14	No	N/A	N/A	15	15	3	3
B	26-02-14	07-03-14	No	N/A	N/A	12	12	0	0
C	21-02-14	28-02-14	No	N/A	N/A	10	10	1	1
D	15-10-14	28-10-14	No	N/A	N/A	15	15	0	0

Case Set 15

Customs Name	ND	AD	Late	APD	ARD	IFP		TP	
						Before	After	Before	After
E	07-03-14	21-03-14	Yes	17-03-14	28-03-14	15	10	0	5
E	12-03-14	25-03-14	No	N/A	N/A	10	10	5	5
A	22-09-14	30-09-14	No	N/A	N/A	15	15	3	3
B	17-12-14	23-12-14	No	N/A	N/A	12	12	0	0
C	20-11-14	25-11-14	No	N/A	N/A	10	10	1	1
D	24-01-14	07-02-14	No	N/A	N/A	15	15	0	0