NIMBLE Collaboration Network for Industry, Manufacturing, Business and Logistics in Europe

Collaborative Network for Industry, Manufacturing, Business and Logistics in Europe

D3.4

Supply Chain Negotiation Support

Project Acronym	NIMBLE
Project Title	Collaboration Network for Industry, Manufacturing, Business and Logistics in Europe
Project Number	723810
Work Package	WP3 Core Business Services for the NIMBLE Platform
Lead Beneficiary	SRDC
Editors	SUAT GÖNÜL, SRDC YILDIRAY KABAK, DOĞUKAN ÇAVDAROĞLU
Reviewers	
Contributors	
Dissemination Level	PU
Contractual Delivery Date	31/12/2017
Actual Delivery Date	03/02/2018
Version	1.0

NIMBLE in a Nutshell

NIMBLE is the collaboration Network for Industry, Manufacturing, Business and Logistics in Europe. It will develop the infrastructure for a cloud-based, Industry 4.0, Internet-of-Things-enabled B2B platform on which European manufacturing firms can register, publish machine-readable catalogues for products and services, search for suitable supply chain partners, negotiate contracts and supply logistics. Participating companies can establish private and secure B2B and M2M information exchange channels to optimise business work flows. The infrastructure is being developed as open source software under an Apache-type, permissive license. The governance model is a federation of platforms for multi-sided trade, with mandatory interoperation functions and optional added-value business functions that can be provided by third parties. This will foster the growth of a net-centric business ecosystem for sustainable innovation and fair competition as envisaged by the Digital Agenda 2020. Prospective NIMBLE providers can take the open source infrastructure and bundle it with sectorial, regional or functional added value services and launch a new platform in the federation. The project started in October 2016 and will last for 36 months.

Abstract

This document provides details about business processes, which are the enablers of B2B communication of NIMBLE users. Although business processes may have graph-like structures often modelling complicated flows among the participant actors, we follow a simpler approach where the processes are treated as a sequence of message exchanges between the participants. Our primary argument for following this approach is to deliver early prototypes supporting the activities throughout the supply chain, thus initially yielding an intuitive, easy to use platform, which then enables identification of the most desired further capabilities.

Throughout the document, we refer to a generic and a sector-specific standard defining business process management namely UBL and MODA-ML. These are of interest for NIMBLE as they target the industrial supply chains. Built-in business processes supported by NIMBLE follow the standards section. Our idea is to have business process templates that can be concatenated to model more complex business processes as needed. The current platform supports a set of such templates addressing different activities like ordering, fulfilment and transport execution plan. We present the usage of these templates throughout a scenario.

After elaborating the supported templates, we present details about the backend services enabling management of business processes. In NIMBLE, management of business processes is considered along two dimensions: design and execution. We have integrated *Camunda*, an open source business process management platform into NIMBLE, for both the design and execution of business processes.

Document History

Version	Date	Comments
V0.1	28/11/2017	Initial structure
V0.2	10/12/2017	Feedback from IBM
V0.3	22/12/2017	First complete version
V0.4	26/12/2017	QA IBM/bm
V0.5	08/01/2018	Revised version SRDC
V0.6	28/01/2018	Final edits SRFG/wb
V1.0	03/02/2018	Submission (wb)

Table of Contents

Acronyms

Acronym	Meaning
API	Application Programming Interface
B2B	Business to business
BPMN	Business process modelling notation
GUI	Graphical User Interface
NIMBLE	Collaboration Network for Industry, Manufacturing, Business and Logistics in Europe
PPAP	Production part approval process
REST	Representational State Transfer
UBL	Universal Business Language
UI	User interface
XSD	XML Schema Definition

1 Introduction

The ultimate aim of NIMBLE is to optimize companies' B2B operations throughout the supply chain by orchestrating and automating data exchange among the participants of the chain in different phases such as sub-contracted manufacturing, transportation, etc. The orchestration of data exchange for achieving this automation is realized through business processes. Business processes involve at least two trading parties with specific roles feeding the process with relevant data.

Business processes mandate data exchange to happen through well-defined procedures with standards-based, structured messages. This enables NIMBLE to keep track of the communication history and paves the way for providing value-added services on the maintained data, such as trading partner comparison, transaction dashboards, and auditing.

We make use of available standards as much as possible for built-in business process definitions and messages associated with these processes. Current built-in processes are modelled as bilateral data exchange schemes called templates. The aim is to keep communication activities as simple as possible and let users choose whatever template meets their needs.

Despite utilizing standards as much as possible, we envision a customizable system providing alternative ways to override default mechanisms. The overriding options are related to the creation, processing and delivery of the information to recipients. Synchronization with legacy information systems with the information related to business processes, constraining with processes access-control rights, data sharing settings are examples of value-added capabilities enabled by the customization options. In addition to customization of built-in processes, we also aim to provide capabilities to define completely new processes addressing the custom needs of specific companies or industry sectors.

1.1 Scope

The aim of this document is to lay out the conceptual terminology used in NIMBLE related to business processes along with implementation details as a documentation of the provided functionalities and associated frontend / backend services.

Throughout this document, we first provide details about the standards used, namely, Universal Business Language (UBL) and MODA-ML, and introduce a set of business process definitions with varying complexities. After referring to some of the example process definitions, we give details about the built-in process templates in NIMBLE, extracted from the standards. We also introduce a logical sequence of processes considering the activities throughout a supply chain.

After a conceptual elaboration of a business process, we provide design decisions taken on the underlying technical architecture for maintaining the business processes and establishing the communication between trading partners; present the main parts of the associated software module and give an overview of the open API supported by the software module.

2 Standards-Based Business Processes

Heterogeneity of the industrial data landscape arising from sectorial differences, and range of information systems is a fact (Heaney, 2013). This makes relying on standards reasonable for any industry-targeting software. On the one hand our motivation is to comply with generic standards, but on the other hand we still want to be able to support sectorial standards as well. Sectorial support eases on-boarding of sector companies as NIMBLE users thanks to low-entry barriers and associated value-added services available on the platform.

To support this vision, we cover the UBL and MODA-ML standards in the subsequent sections. In NIMBLE, in addition to a library of reusable data elements as elaborated in D2.2 (Semantic Modelling of Manufacturing Collaboration Assets), UBL was also chosen as the generic standard providing a library of business processes. UBL covers entire supply chains in terms of available process definitions and messages to be exchanged in those processes.

As introduced before, we are also interested in the MODA ML standard, which is a UBL-driven standard specialized for the textile clothing and footwear sectors and used in NIMBLE's textile related use case. Like UBL, MODA-ML also includes a library of messages that are shared within business processes specific to textile sector supply chains. Below, we provide further details about the aforementioned two standards.

2.2 UBL Processes

UBL offers a wide range of business processes driven from an interlinked set of supply chain use cases as depicted in Figure 1. The root use case of the use cases tree is the purchase of an item, from which other use cases are branched. Each use case is associated with a business process definition specifying the actor roles involved in the process, activities through which information is exchanged between the involved actors and schemas defining the structure of information to be exchanged. Figure 2 exemplifies this for an ordering process. Figure 3 shows a section of the UBL specification with links to several resources including the XSD schema related to an Order document.

NIMBLE does not cover the complete use case tree shown in Figure 1, but focuses on a subset of use cases including Quotation, Ordering, Shipment and Fulfilment. Integration of these processes into NIMBLE will be explained in detail in Section 3. The

complete list of process definitions and details about those can be found on the UBL specification page[1].
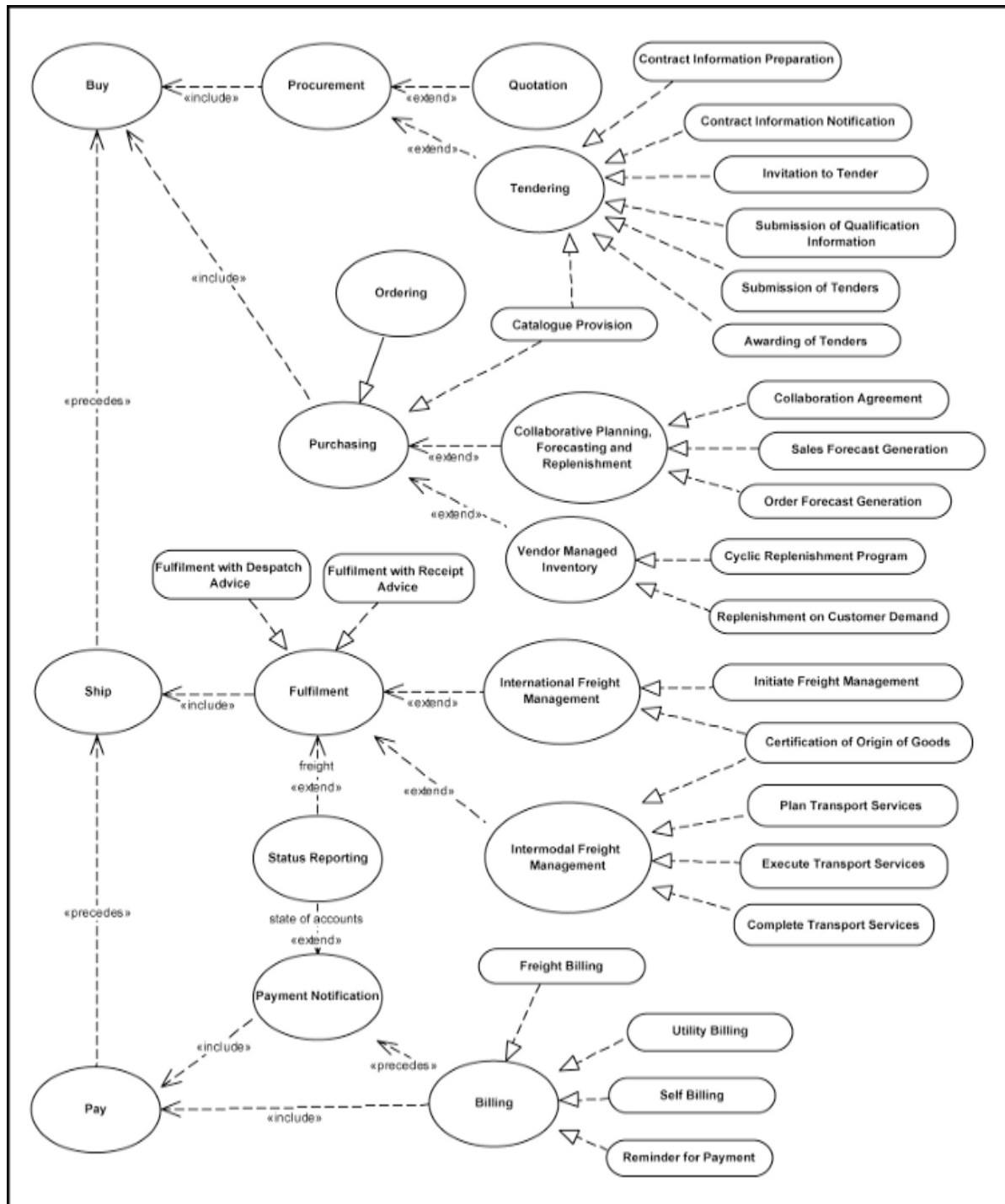

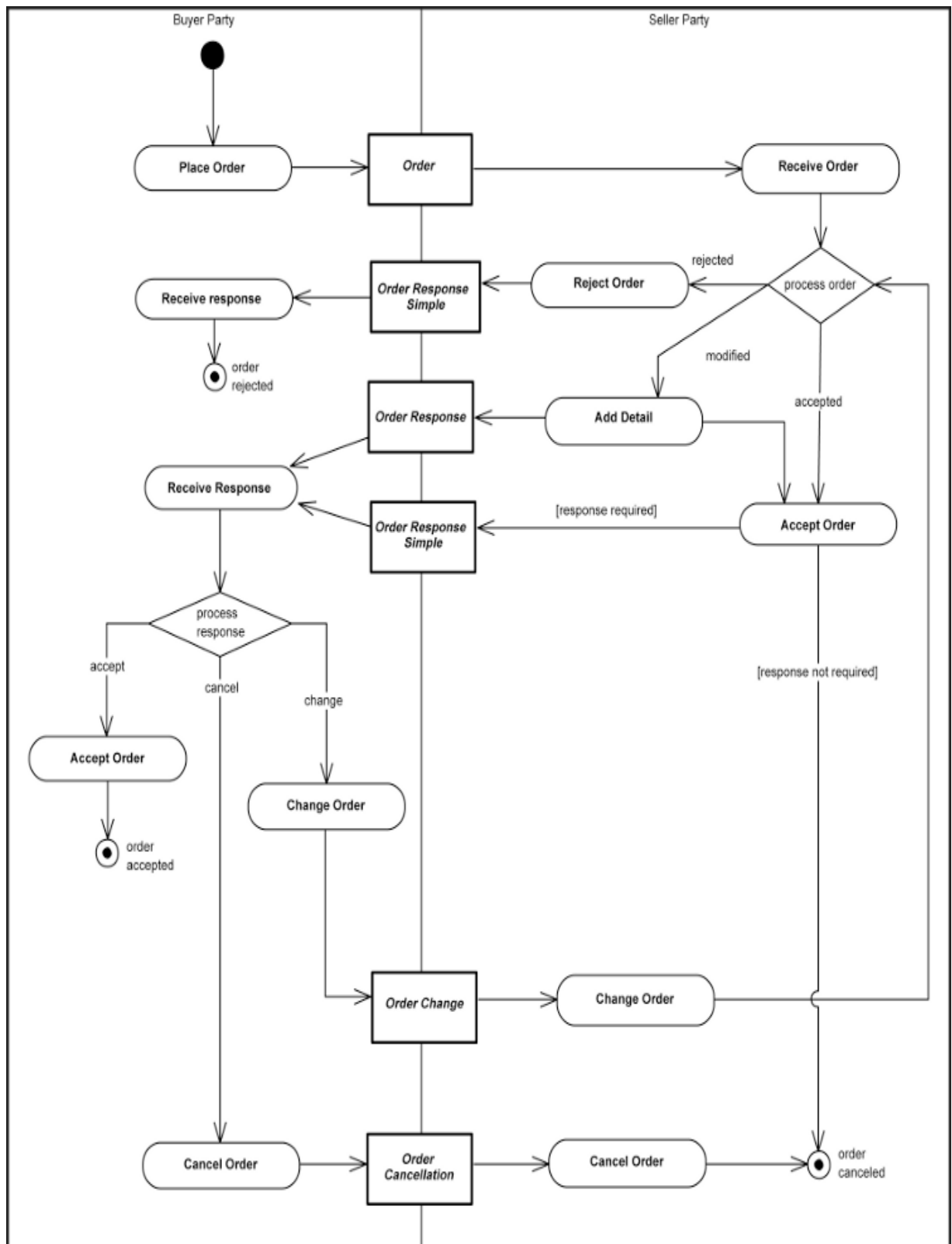**Figure 1 Interlinked supply chain use cases modelled by UBL[2]**

---

[1] http://docs.oasis-open.org/ubl/os-UBL-2.1/UBL-2.1.html.

[2] http://docs.oasis-open.org/ubl/os-UBL-2.1/UBL-2.1.html

Figure 2 UBL Ordering process[3]

3.1.32 Order

Description: A document used to order goods and services.

Processes involved	Ordering
Submitter role	Buyer
Receiver role	Seller
Normative schema	xsd/maindoc/UBL-Order-2.1.xsd
Runtime schema	xsdrt/maindoc/UBL-Order-2.1.xsd
RELAX NG schema	rnc/versions/UBL-Order-2.1.rnc
Document model (ODF)	mod/maindoc/UBL-Order-2.1.ods
Document model (Excel)	mod/maindoc/UBL-Order-2.1.xls
Document model (UML)	uml/UBL-Order-2.1.html
Summary report	mod/summary/reports/UBL-Order-2.1.html
UBL 2.0 example instance	xml/UBL-Order-2.0-Example.xml
UBL 2.1 example instance	xml/UBL-Order-2.1-Example.xml

Figure 3 Resources for Order document schema[4]

2.3 MODA-ML Processes

From a technical point of view, MODA-ML has the same set of conceptual elements as UBL. Business processes specify the actor roles, activities performed between the actors and schemas for the documents to be exchanged via the activities. As an example, Figure 4 shows details about the "Fabric subcontracted darning" process. In this case, two actors are involved in the process with the fabric producer and darning subcontractor roles. A set of documents of which names are given are exchanged between the two.

MODA-ML introduces processes for mainly supply and subcontracted manufacturing of textile products like fabric and yarn. Our priority is not to re-implement some of these MODA-ML processes on the NIMBLE platform. Nevertheless, we will reuse some of the document schemas to realize the collaborative design of the textile use case within a new business process designed for this specific scenario.
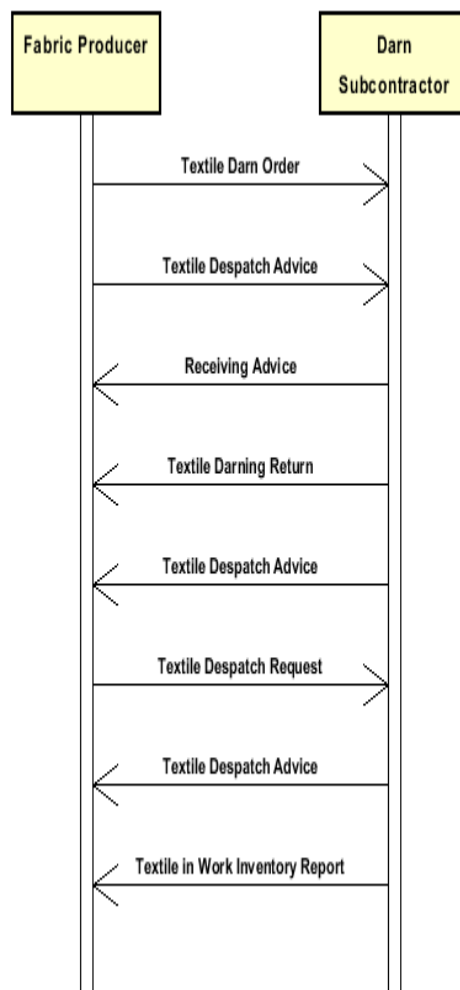
---

[4] http://docs.oasis-open.org/ubl/os-UBL-2.1/UBL-2.1.html#T-ORDER

Figure 4 Details of MODA-ML Fabric subcontracted darning[5]

3   Built-In Business Process Templates in NIMBLE

As Figure 2 indicates, even an ordering process is quite complex if all possible flows are handled. We followed a pragmatic approach considering the ease of use, understandability of the B2B data exchange capabilities envisioned on the NIMBLE platform as well as time constraints for delivery of the required technical functionalities.

We tried to extract the very core activities of business processes considering the main goal of the process. Again, considering the ordering process depicted in Figure 2, the main aim is to initiate an order by a buyer for some items and wait for a response from

---

[5] http://www.moda-ml.org/moda-ml/repository/uml-xmi/v2013-1/en/A_Subcontractedfabricdarning.png

the seller. So, we designed this process simply as a two-step, bilateral data exchange between a buyer and a seller as depicted in Figure 15.

Currently, NIMBLE supports six processes, most of which are adapted from the UBL standard. Each of them has the same data exchange structure as the order process. Below, we first present a supply chain scenario through which the supported business processes can be utilized and then provide further details about their intended use, involved actors and exchanged documents. Alongside these details, screenshots of the associated business process are also presented.

3.1 An Example Supply Chain Scenario

As depicted in Figure 6, the scenario involves three parties of three different types, namely: buyer, seller and transport service provider. The buyer looks for a product on the NIMBLE platform. Taking inspiration from the Micuna use case, let the buyer be Micuna looking for a specific type of varnish to be used to manufacture their new cradle, Kit-Desk (see Figure 5), which can be used as a blackboard by children. The other two entities are made up of companies *Pro Varnish*, from Germany, manufacturing the varnish, and *Hoffman Logistics*, also from Germany, providing road transport services throughout Europe. Below, we narrate the activities taking place between the parties throughout the scenario. On the right hand side, Figure 6 refers to the business process activated in each phase.

Figure 5: Micuna Kit-Desk

Before collaboration: Before elaborating the communication performed among those parties, it is assumed that all three parties are already registered in NIMBLE; and, Pro Varnish and Hoffman Logistics have already published their products / services on NIMBLE.

Product identification and requesting further details: After Micuna identifies a potential varnish, it would like to learn further details about the product that cannot be inferred from the published information; specifically, further details about the viscosity and adherence of the varnish. Micuna prepares a text-based document containing the parameters and requested values for each of them e.g. 30 M of viscosity and class 4-5 of adherence. Pro Varnish replies to this request by providing another document containing complete details about the varnish.

Requesting sample: Once being sure that the varnish meets Micuna's criteria, it asks for 2 liter of free sample varnish for testing it on premises as the final validation before ordering in bulk for serial production. Pro Varnish accepts sending the free sample to Valencia, Spain, where Micuna is based.

Fulfilment of sample shipment: Pro Varnish informs Micuna with delivery details after sending the sample with a regular courier service. As a response upon retrieval of the sample, Micuna informs Pro Varnish about the delivery.
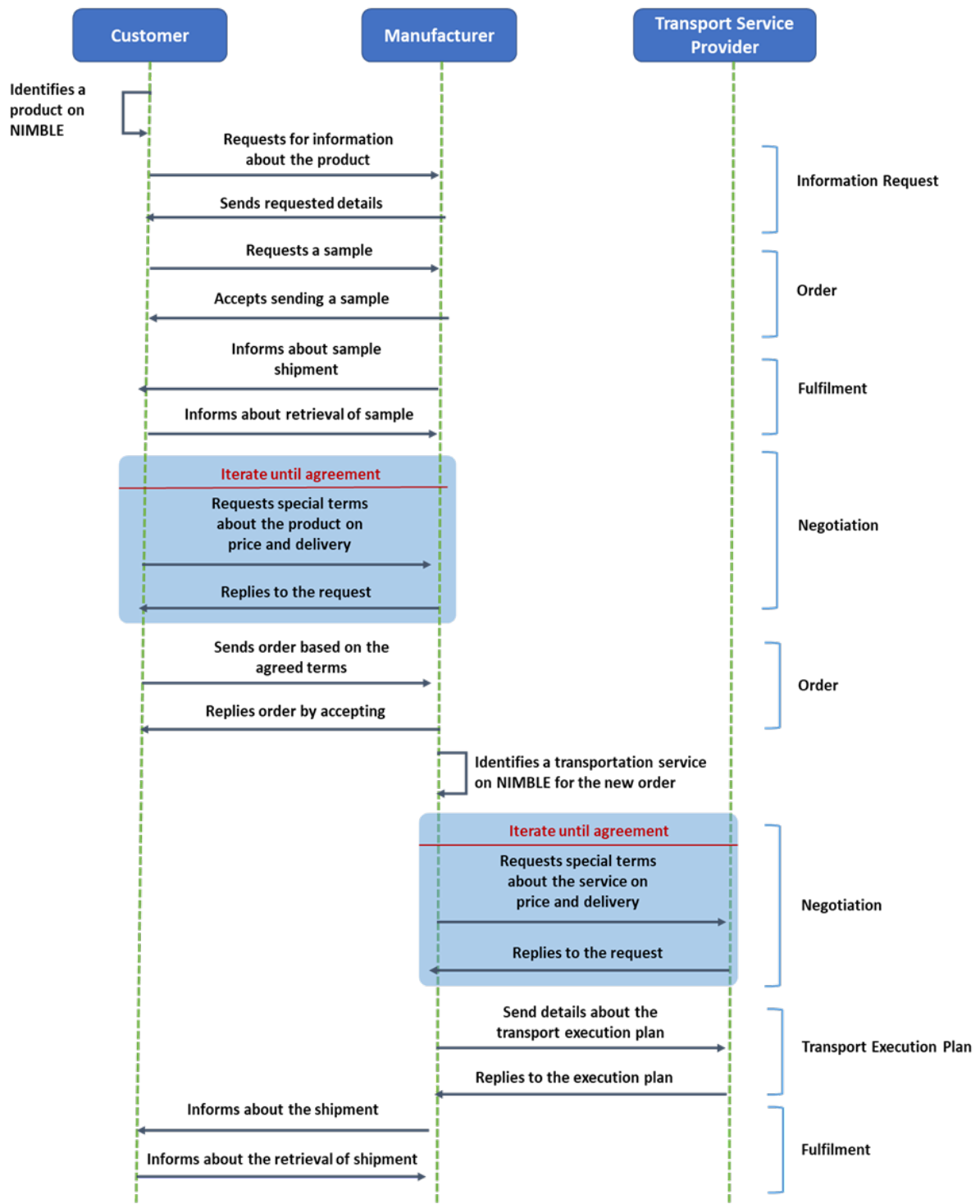
Figure 6 Sequence of activities performed throughout the example scenario

Negotiation: Once the on premise-testing lead to a positive decision, Micuna negotiates on the prices and delivery terms considering the bulk amount to be ordered. A series of terms exchanges take place between the parties as a negotiation process until an agreement is reached on all the terms.

Ordering: Micuna initiates an order based on the terms agreed in the negotiation. Pro Varnish replies to this order request positively.

Transport service identification and negotiation on the terms: After accepting the order, Pro Varnish looks for a transport service provider that might provide cheaper prices than a regular courier service for delivering the ordered varnish to Spain. Pro Varnish finds a road transport service covering whole Europe, provided by Hoffman Logistics.

Pro Varnish would like to arrange the transportation service based on the delivery terms of Micuna. Therefore, it initiates a negotiation with Hoffman Logistics, which lasts until an agreement is reached by both parties. Parties agree on prices, delivery means based on the period and physical characteristics of the package to be shipped.

Send transport execution plan: Parties agree on many terms during the negotiation, but precise timing of the shipment is specified via a transport execution plan. This plan is initiated by Pro Varnish and accepted or rejected by Hoffman Logistics.

Fulfilment of the order: Once the shipment is started, Pro Varnish informs Micuna about the details of the shipment and vice versa Micuna informs Pro Varnish about the safe arrival of the shipment.

In the subsequent sections, we present details about the business process templates currently supported by NIMBLE. We refer to the scenario described hereby where relevant i.e. in the UI screenshots.

3.2 Sequence of Processes

As a summary of the aforementioned business processes, we introduce the logical sequence of templates as depicted in Figure 7.
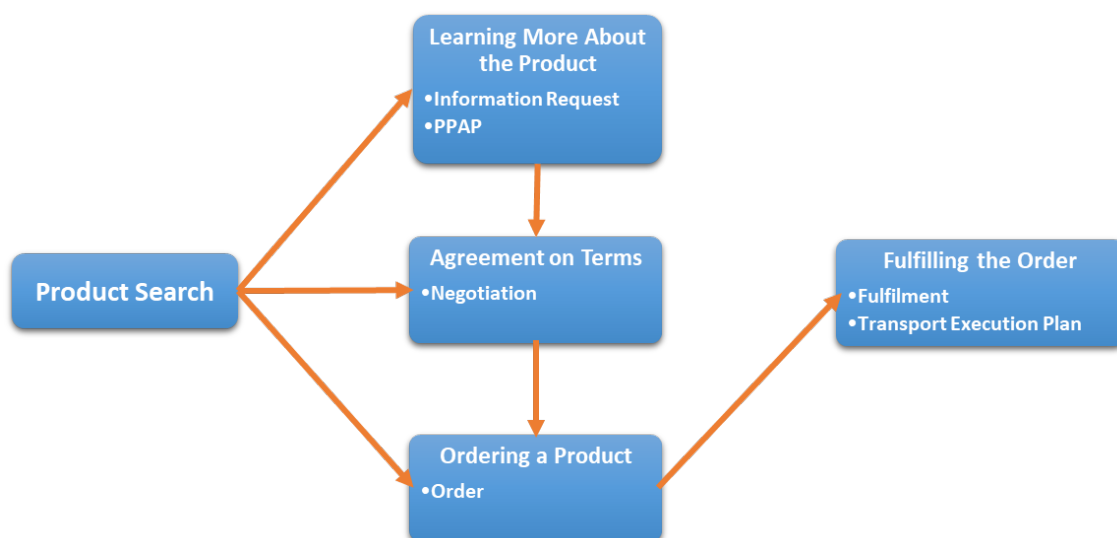
Figure 7 Sequence of business processes

Following the search phase, users have three options: They can either ask more information about the product or negotiate on the trading terms of the product or directly order it.

The first group of business processes contains Information Request and Production Part Approval Process (PPAP), which can be used to get detailed information about product characteristics, its design / production processes, measurement results and so on. Negotiation may take place right after the search or following the information retrieval processes. Similarly, ordering may take place right after the search or following the negotiation phase. As the last step completing the supply chain, fulfilment related processes follow the ordering.

3.3 Information Request

Information Request process is supposed to be used to request detailed information for an item (product / service) traded on NIMBLE before ordering the item. Figure 8 illustrates the involved actors in the process and messages exchanged between them.
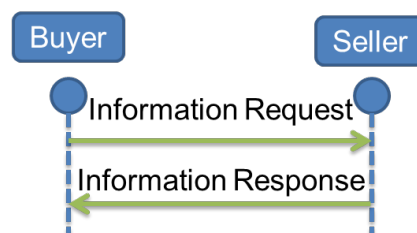
Figure 8 Information Request business process

Information Request: The buyer party can provide an arbitrary document (text document, image, etc.) specifying the detailed information pieces requested along with an explanatory note. Figure 9 shows the relevant UI part where the buyer has uploaded a requirements document and provided a note.

Common characteristic of the business process template UIs: As seen in the subsequent figures, the top part of the screenshot and the first two tabs "Product Characteristics", "Product Trading & Delivery Terms" are fixed for any business process concerning *products*. As can be seen in Figure 13, the first two tabs are replaced with a single one, named Service Characteristics, in case of *transportation services*. Turning back to the product case, the first two tabs contain the product details as provided by the seller. The third and subsequent tabs, if there is any, are specific to the activated business process.

Document schema used in this step: https://github.com/nimble-platform/common/blob/master/data-model/ubl-data-model/src/main/schema/NIMBLE-UBL-2.1/maindoc/UBL-ItemInformationRequest-2.1.xsd

Information Response: Similar to the first step, the seller party can provide an arbitrary document including the requested details along with an explanatory note. Figure 10 shows a screenshot of the platform, again from the buyer point of view when the buyer has already received a response from the seller. This means that the Information Request process is completed. The figure shows both the requirements document sent by the buyer and the detailed technical specification document sent by the seller as a response.

Document schema used in this step: https://github.com/nimble-platform/common/blob/master/data-model/ubl-data-model/src/main/schema/NIMBLE-UBL-2.1/maindoc/ItemInformationResponse.xsd

Figure 9 Requesting additional information

Figure 10 Screenshot of the Information Request process

3.4 Negotiation

The negotiation process enables a buyer party to request different/additional terms than the default terms specified by the manufacturer / service provider. For example, a customer may request some discount for large quantities of the product, request shorter delivery period for higher prices or ask for the price for delivery to a specific location.
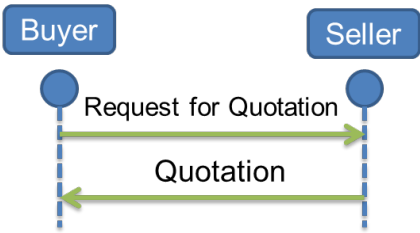
Figure 11 Negotiation business process

Request for Quotation: In the first step of a negotiation process, the buyer party presents its terms on the product to be ordered or service to be used. This step is named Request for Quotation as the name of the UBL document exchanged. In fact, we followed the same approach while naming the steps in the other business processes as well.

NIMBLE supports different options concerning the negotiation parameters about physical products and transportation services. While Figure 12 shows the currently available negotiation parameters for products, Figure 13 shows the ones for transportation services. Pricing and delivery terms are available for both types of traded

items. Further parameters related to handling unit and shipment are available in case of transportation services.

Figure 12 Negotiation parameters for products

Product Identifier   4a414e40-5122-4058-9e4c-428fa079944e

Name   Door to door logistics

Description   D2D logistics throughout Europe

<div style="text-align:center">

Back to Product Details ⌃

</div>

Item_Information_Request | Negotiation

Service Characteristics | Request for Quotation Details

Price / Requested Quantity

Amount	100	Currency:	EUR
Price Base Quantity	100	Unit:	pallet

Delivery Terms

Incoterms			
Requested Delivery Period		Unit:	
Delivery Location	Valencia, Spain		+
Special Terms			

Handling Unit Details

Transport Handling Unit			
Handling Unit Length		Unit:	
Handling Unit Width		Unit:	

Consolidated Shipment

Goods Item			
Quantity		Unit:	
Gross Volume Measure		Unit:	
Gross Weight Measure		Unit:	

Note	

Send Requested Terms

Figure 13 Negotiation parameters for transportation services

Document schema used in this step: https://github.com/nimble-platform/common/blob/master/data-model/ubl-data-model/src/main/schema/NIMBLE-UBL-2.1/maindoc/UBL-RequestForQuotation-2.1.xsd

Quotation: In this step, the manufacturer/transportation service provider party (as the seller) replies to the terms requested. In addition to simply accepting/rejecting the requested terms, it is also possible for a seller to update the terms. Figure 14 depicts the Quotation screen, which is initialized using the requested terms. The responding party can indicate the response status by choosing one of the response labels along with an explanatory note. The negotiation process can be carried on if the requested terms are updated or rejected by the seller by initiating a new request for quotation step. If the seller updated or accepted the terms, the buyer can initiate an order process using the terms sent by the seller.

Document schema: https://github.com/nimble-platform/common/blob/master/data-model/ubl-data-model/src/main/schema/NIMBLE-UBL-2.1/maindoc/UBL-Quotation-2.1.xsd

Product Identifier   varnish1-provarnish

Name   Waterbased clear varnish

Description   Touch dry in 30 minutes, for interior use

Commodity Classifications   Varnish

Product Characteristics   Product Trading & Delivery Terms   Request for Quotation Details   Quotation Details

Price / Requested Quantity

Amount	50	Currency:	EUR
Price Base Quantity	1	Unit:	lt
Requested Quantity	50	Unit:	lt

Delivery Terms

Incoterms

Requested Delivery Period   Unit:

Delivery Location   +

Special Terms

Warranty

Validity Period   Unit:

Response   Terms Updated ⇕

Response Reason   OK, but can only provide 50 lt with this price

Note

Send Quotation

Figure 14 Quotation screen

3.5 Order

An Order business process can be used for ordering products with a certain price or requesting free samples. As a response the seller party can approve the request by accepting the buyer's terms or decline the request by rejecting the terms.
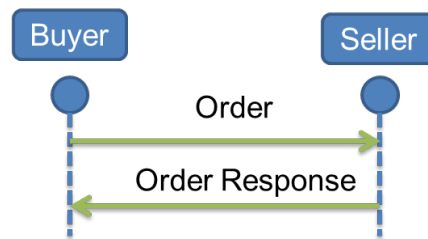
Figure 15 Order business process

Order: This step can be initiated in two ways. First, as a continuation of the Negotiation process; second, directly after identifying a product through the search phase. In addition to the price and delivery related terms, a very basic set of payment related terms can be specified by the buyer. As already mentioned, the Order process can be used to request a free sample. This can be realized, for example by setting the price *base quantity parameter* to zero.

Document schema: https://github.com/nimble-platform/common/blob/master/data-model/ubl-data-model/src/main/schema/NIMBLE-UBL-2.1/maindoc/UBL-Order-2.1.xsd

Order Response: The terms specified in the Order step can be accepted or rejected in this step as depicted in Figure 17.

Document schema: https://github.com/nimble-platform/common/blob/master/data-model/ubl-data-model/src/main/schema/NIMBLE-UBL-2.1/maindoc/UBL-OrderResponseSimple-2.1.xsd

"

Product Identifier		varnish1-provarnish
Name		Waterbased clear varnish
Description		Touch dry in 30 minutes, for interior use
Commodity Classifications	Varnish	

Product Characteristics | Product Trading & Delivery Terms | Order Details

Quantity	50	Unit:	lt

Price

Amount	50	Currency:	EUR
Price Base Quantity	1	Unit:	lt

Delivery Terms

Requested Delivery Period		Unit:	
Delivery Location			+

Payment Means

Payment Means	
Payer Account	
Payee Account	

Order Note	

Send Order

Figure 16 Order step

Figure 17 Order Response step

3.6 Fulfilment

The Fulfilment business process is initiated by the seller party to inform the buyer that the order has been processed, and the ordered products are shipped. The buyer on the other hand informs the seller about the delivery of the shipment. This process can be initiated by the seller only after an order is accepted by the seller.

Figure 18 Fulfilment business process

Dispatch Advice: In this step, the seller can provide detailed information about the shipment including delivered quantity, handling instructions, estimated delivery date, and carrier party. See Figure 19 depicting the UI where a Dispatch Advice is sent.

Document schema: https://github.com/nimble-platform/common/blob/master/data-model/ubl-data-model/src/main/schema/NIMBLE-UBL-2.1/maindoc/UBL-DespatchAdvice-2.1.xsd

Receipt Advice: As a response to the Dispatch Advice, the buyer can send a Receipt Advice informing the buyer about the status of the delivered products. In case any problem with the delivered products, the buyer has the option to reject a certain amount of the products shipped as depicted in Figure 20.

Document schema: https://github.com/nimble-platform/common/blob/master/data-model/ubl-data-model/src/main/schema/NIMBLE-UBL-2.1/maindoc/UBL-ReceiptAdvice-2.1.xsd

Figure 19 Dispatch Advice

Figure 20 Receipt Advice

3.7 Transport Execution Plan

The Transport Execution Plan process is used to plan the exact dates for the transport service provider to pick up the products from the manufacturer's site and start shipment. As a response, the transport service provider can accept or reject the terms. Note that the party in demand of a transportation service is called a transport service user in the scope of this process.
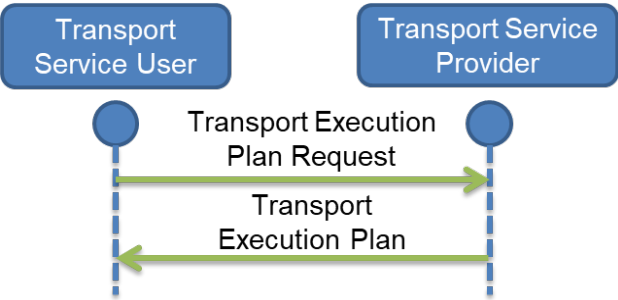

Figure 21 Transport Execution Plan business process

Transport Execution Plan Request: Currently, this step can be initiated in two ways. First, it can be initiated by a seller party after an order is accepted. An accepted order reveals an option for the user to look for a transport service provider to execute the transportation related to the order. Second, after negotiating on the terms of the transportation service a suitable option is presented to the user so that s/he can initiate the process.

In this step, the service user party provides detailed information about the consolidated shipment such as volume/weight of the shipment, handling instructions, estimated

delivery date and carrier party. The other main group of information is related to the execution plan. Start and end locations of the transportation as well as the periods when the transportation must start and end are provided as the execution plan details. Figure 22 presents the UI where this step is initiated.

Document schema: https://github.com/nimble-platform/common/blob/master/data-model/ubl-data-model/src/main/schema/NIMBLE-UBL-2.1/maindoc/UBL-TransportExecutionPlanRequest-2.1.xsd

Transport Execution Plan: As a response to the Transport Execution Plan, the service provider party can reply to the request with an indication of acceptance/rejection along with some reasons/notes. Figure 23 depicts the relevant UI.

Document schema: https://github.com/nimble-platform/common/blob/master/data-model/ubl-data-model/src/main/schema/NIMBLE-UBL-2.1/maindoc/UBL-TransportExecutionPlan-2.1.xsd

Product Identifier	f4ff1e8b-58c7-4b9b-a9df-3a24868974e3
Name	Door to door logistics
Description	D2D logistics throughout Europe Back to Product Details

Back to Product Details ⤊

Transport_Execution_Plan

Transportation Service Details | Transport Execution Plan Request Details

Consolidated Shipment

Goods Item	Waterbased clear varnish	
Gross Volume Measure	5	Unit: m3
Gross Weight Measure	120	Unit: kg
Handling Instructions	The pallet must be handled such that it would always stand vertically	
Estimated Delivery Date	2017-12-24	
Carrier Party		
Carrier Party Contact		

Execution Plan

Service Start Date	2017-12-17
Service End Date	2017-12-24
From Location	Berlin +
To Location	Valencia +

Use Service

Figure 22 Transport Execution Plan details

Figure 23 Responding to Transport Execution Plan

3.8 Production Part Approval Process (PPAP)

Production Part Approval Process (PPAP) is a process used for ensuring the product quality of semi-finished products / intermediate goods like car spare parts, raw materials such that the supplier meets the buyer's requirements in terms of design and production processes. A buyer might request several documents in the scope of this process including process flow diagrams, dimensional results, quality inspection reports and so on.
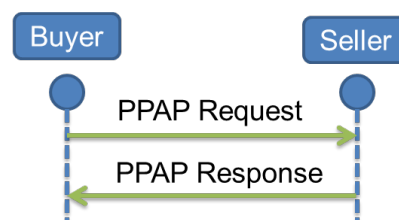
Figure 24 PPAP business process

PPAP Request: This step can be initiated directly after identifying a product through the search phase. The buyer can choose a PPAP submission level from ranging from 1 to 5. Each level has its own submission requirements, which requires provision of more information by the seller as the level increases. For example, level 1 requires information about "Part Submission Warrant" and "Appearance Approval Report", level 5 requires information about all the elements presented in Figure 25. PPAP levels are summarized as follows:

- Level 1 – Part Submission Warrant (PSW) only submitted to the customer
- Level 2 – PSW with product samples and limited supporting data
- Level 3 – PSW with product samples and complete supporting data
- Level 4 – PSW and other requirements as defined by the customer

---

- Level 5 – PSW with product samples and complete supporting data available for review at the supplier's manufacturing location

Document schema:

https://github.com/nimble-platform/common/blob/ppapProcess/data-model/ubl-data-model/src/main/schema/NIMBLE-UBL-2.1/maindoc/UBL-PpapRequest-2.1.xsd

PPAP Response: As a response to the PPAP Request, the seller should submit the requested documents to the buyer. The seller can choose one or more documents for each element specified in the request.

Document schema:

https://github.com/nimble-platform/common/blob/ppapProcess/data-model/ubl-data-model/src/main/schema/NIMBLE-UBL-2.1/maindoc/UBL-PpapResponse-2.1.xsd

Figure 25 PPAP Request step

Figure 26 PPAP Response step

4   Implementation Details

Concerning the technical implementation, we take business processes into account mainly on two dimensions: design and execution. Design activities include the structured representation of processes and their user (company) specific configurations. Execution covers instantiation of processes and keeping track of each instance. As explained in the next section, we utilize Camunda for both design and execution of business processes.

4.1   Camunda Integration

We have integrated Camunda, which is an open source platform, as the behind-the-scenes enabler of business process management in NIMBLE. Concerning the deployment architecture, Camunda is natively integrated into the microservice architecture of NIMBLE thanks to its Spring Boot Starter[6]. Specifically, it is embedded

---

[6] https://github.com/camunda/camunda-bpm-spring-boot-starter

inside the business process microservice and is launched simultaneously along with the microservice itself.

Camunda offers several capabilities that we utilize in NIMBLE. First, it provides a standalone modelling environment with graphical UIs to design new business processes[7]. (Note that this a standalone tool, which is not shipped inside Camunda Spring Boot Starter). We utilized this capability in order to design the built-in process supported by NIMBLE so far. Figure 27 shows the design of the Negotiation process on the modeller. Details of the design approach will be given in the subsequent sections.
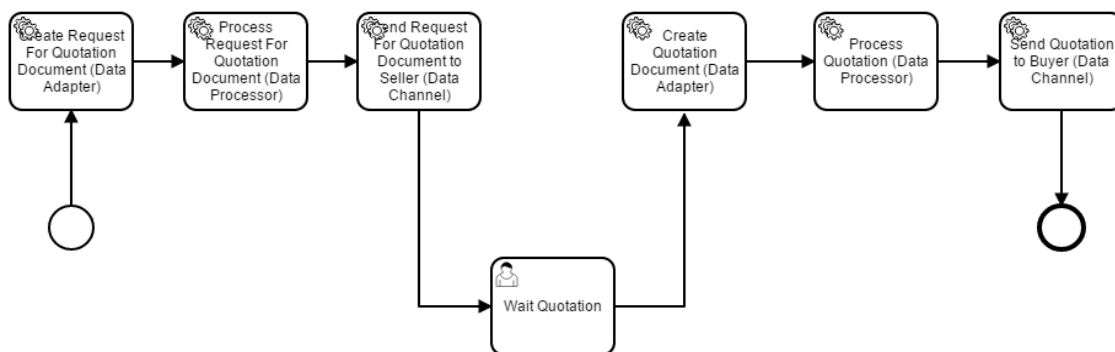

Figure 27 Design of the Negotiation process on Camunda

Camunda Modeller automatically generates a BPMN representation of the designed process, which is then imported to the Camunda Engine. The BPMN representations can be seen as general definitions of processes that would be instantiated for each individual activity e.g. negotiation on NIMBLE. Once the BPMN representation of a business process is imported to the Camunda Engine, it is then able to keep track of individual instances.

In addition to the modelling environment, Camunda provides a REST API through which all business process related activities can be realized. These REST services are deployed onto the web application created for the business process microservices.

4.2 Business Process Execution

In NIMBLE, business process execution is mainly focused on triggering the steps that need manual intervention so that the flow of the process can move forward to the next activity. Taking Figure 27 as an example, there are two steps requiring an external trigger. The first one is the start state (the circle on the left) and the "Wait Quotation" step in the middle. Indeed, the latter one is a *User Task*, which is a BPMN construct that is used in business process definitions where an explicit confirmation is expected from a human being. The other steps in the process i.e. the ones with a "setting icon"

---

[7] https://camunda.org/download/modeler/

on the top-left are executed automatically. The reason for having separate automatically executed processes are given in the next section.

Each built-in process on NIMBLE is initiated by the user through either the relevant UI or the REST API provided by the business process microservice. By default, all the built-in business processes are triggered by a REST call to the *start* endpoint. This endpoint expects five parameters, namely:

- **processID:** identifier of the business process type inside NIMBLE e.g. *Negotiation*
- **processInstanceID:** identifier of the instance created for the specified business process
- **initiatorID:** identifier of the party initiating the process
- **responderID:** identifier of the target party
- **content:** message content to be sent

The same methods are used for resuming the process.

Another concept related to the execution of business processes is monitoring. Considering the initiation of business processes, they can be categorized as incoming or outgoing. While the user should be responding to an incoming activity, a response is expected from the target trading partner for an outgoing activity. Concerning monitoring the status of processes, it is possible to retrieve the list of incoming and outgoing processes and in turn query the status of each process via the REST services provided by a combination of the business process microservice and Camunda. However, the utilization of REST services is a suitable approach for software vendors to integrate data from NIMBLE into their own systems.

For users desiring to monitor the processes on NIMBLE, there is a dashboard page as depicted in Figure 28. NIMBLE provides a panel presenting a summary of incoming (Sales) and outgoing processes (Purchases). Each entry in the lists includes basic information about the type of process, initiation date of process, product, trading partner, process status and action required. Each entry also contains a context menu icon revealing further options regarding that process instance. Currently, a process can be canceled by the initiating party if it is not replied by the target party yet. Furthermore, the communication history about the process can be visualized by clicking on the *Business History* button. This action navigates the user to the UI of the related business process as depicted in Section 3. Users are presented with relevant options according to the status of the process and role of as well as the sequence of processes depicted in Figure 7.
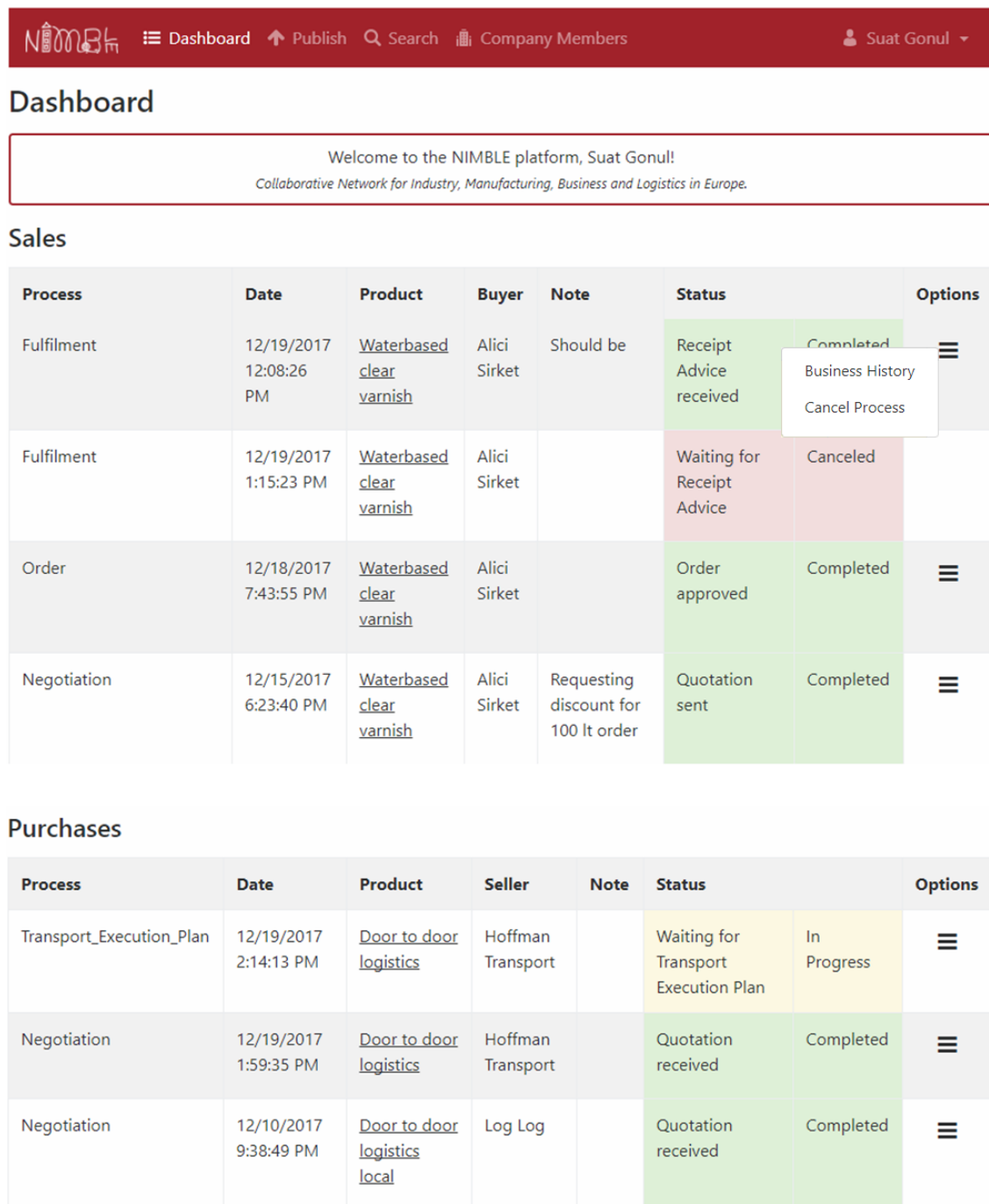
Figure 28 NIMBLE B2B dashboard

4.3 Business Process Design

In general, designing a process means describing a process in a structured manner. Specifically, design activities involve the specification of actors to be included in the process, the flow of message exchanges between these actors and the type of messages to be exchanged.

We used Camunda Modeler for designing the built-in business processes supported by NIMBLE and for generating the corresponding BPMN representations. The representations of built-in processes are located at [8]. Those business processes are imported into the engine at the start-up of the business process microservice.

Instead of forcing users to learn BPMN and design processes with BPMN constructs, we provide a more intuitive environment for process design. In this way, we both eliminate the dependency on an external design tool and more importantly introduce a modular method for process design, which allows customization of processes considering the level of integration of users with their own legacy systems.

4.3.1 3-Step Data Transmission

We have seen two depictions of the Negotiation process in Figure 11 and Figure 27. The former depiction is a sequence diagram showing the sequence of message exchanges between the trading partners. The latter figure shows how this bilateral communication is realized through the execution of the corresponding business process. As can be seen in Figure 29, each message exchange is realized via a 3-step sub-process, which is executed sequentially and automatically.
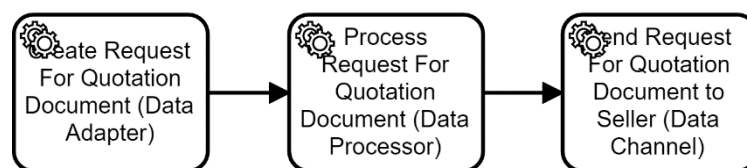

Figure 29 3-step data transmission for the first step of the Negotiation business process

Currently, the BPMN representations of built-in business processes do not explicitly specify the actor roles and message types that we need but they just mandate the sequential execution of the sub-process task for each message exchange activity. Each step in each sub-process is linked with a modular Java class, which is executed when the associated step is active during the process execution. The association of Java classes and process steps is shown in Figure 30. This Java application processes the input parameters and creates the necessary records, associating the process with both the initiator and responder parties.

---

[8] https://github.com/nimble-platform/business-process-service/tree/master/src/main/resources/bpmn

Figure 30 Linking a Java class to a business process step

Having a closer look at the first step of the Negotiation business process, we can see the names of components in parenthesis: Data Adapter, Data Processor and Data Channel. The aim of having such separate activities for each data transmission is to be able to configure data generation, handling of the generated data and its transmission for custom needs. However, the exact extension mechanism has not been elaborated thoroughly yet, and will be shaped based on the emerging integration requirements of users with their legacy systems. For example, one of our end users indicated that synchronization of their internal inventory levels and inventory levels on existing eCommerce platforms like Amazon is a great problem. So, detailing the three data transmission modules is a way of opening up such integration possibilities.
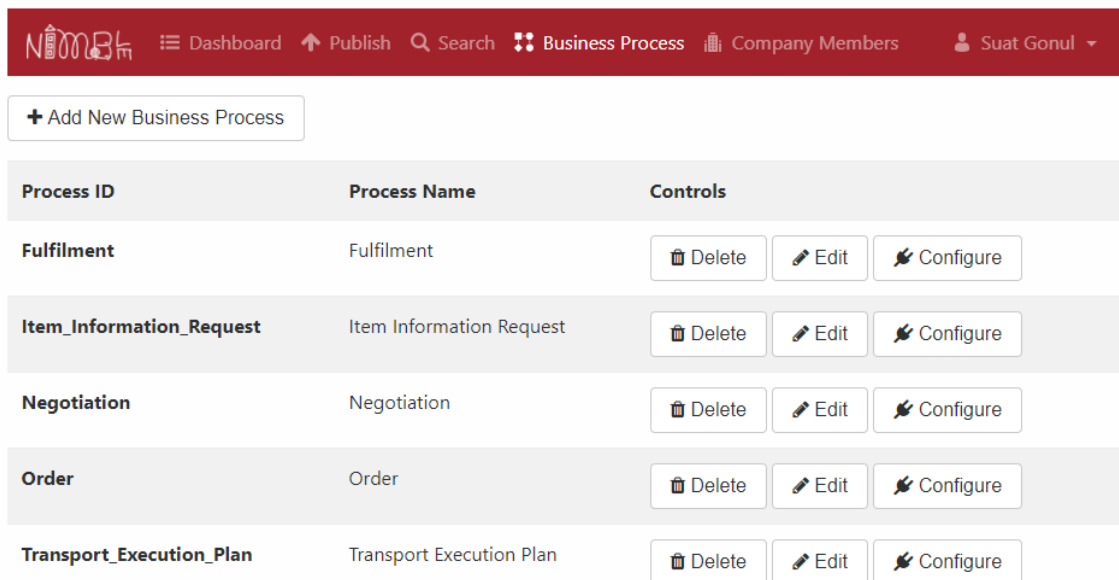
Data Adapter: A Data Adapter is supposed to generate the message to be sent to the trading partner. Concerning the built-in business processes, the data is provided to the default Data Adapters through the REST call to the */start* and */continue* endpoints as stated earlier. NIMBLE might be extended with custom adapters gathering the information to be transmitted from legacy information systems, or IoT-enabled devices through appropriate protocols. Furthermore, access-control mechanisms can be introduced for the generation / retrieval of information, initiation of the process and so on.

Data Processor: Data Processors are supposed to manage the data produced in the previous step by a Data Adapter. By default, the message content is associated with the relevant business process instance and then stored in a database. A possible extension in this step, for example, could be duplication of data into the company's legacy system to keep NIMBLE and the legacy system synchronized.

Data Channel: Data Channels are used to send the message to the recipient. In the current implementation, this component does not have a major functionality except associating the request message (e.g. a Request for Quotation) with the response message (e.g. a Quotation). However, this component can be configured for example to send the document to a remote endpoint or access-control rights could be configured for recipient parties.

4.3.2 Designing Custom Business Processes

We offer capabilities to manage business process definitions on NIMBLE. Navigating to Business Process modules, users can see the list of available business process definitions as depicted in Figure 31. While users have the option to delete, edit and configure them on the existing processes definitions, it is also possible add new ones.

Figure 31 List of available business process definitions

Clicking on the "Add New Business Process" button brings the user to the screen shown in Figure 32 where users can specify the details of the new business process including *id*, *name*, *type* and *content*. *ID* uniquely identifies the process definition throughout the NIMBLE platform. *Type* is mainly an indicator of the supply chain-related activity realized through the process e.g. ordering, fulfilment, negotiation, etc.

Order for Maintenance Services Details

Process ID:	order_for_maintenances_services
Process Name:	Order for Maintenance Services
Process Type:	ORDER ▼

Process Content:

```
Buyer -> Maintenance Service Provider: Order with Service Parameters
Note right of Maintenance Service Provider: Evaluate Order
Maintenance Service Provider -> Buyer: Order Response
```


Create

Figure 32 Business process creation

Processes Content is the actual definition of a business process represented with an intuitive formal language provided by a sequence diagram library[9], which we have integrated into NIMBLE. This library takes some text input (i.e. the process content) and transforms it to a sequence diagram as seen in Figure 32. It should be noted that NIMBLE does not totally support the language offered by this library but only a subset enabling message exchange associations between parties.
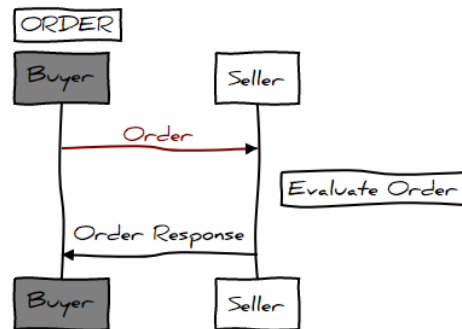
If we look at the process content in detail, there are three lines of two types. The first and third lines first specify the actors between which a message exchange will be realized. The syntax *Company A -> Company B* indicates that a message should be sent from Company A to Company B. Extending this expression, *Company A -> Company B: Order with Service Parameters* indicates that Company A should be sending an order document including service parameters. In the current implementation, each such line is transformed into the 3-step sub-process as explained earlier. The second line, on the other hand, is used to create a task where a manual intervention is required.

---

[9] https://bramp.github.io/js-sequence-diagrams/

Once a business process is created as described above, users have the *edit* and *configure* options. While the edit option enables users to update the description of the process, the configure option provides users with options to override the default configurations for the components of 3-step data transmission sub-process.

Figure 33 Configuration of a business process

4.4 API Overview

The way to communicate with the business process microservice is through its REST API, which contains the following main endpoints:

- /start: Starts a business process

- /continue: Resumes a business process following a user task

- /document: Provides management of messages exchanged within business processes

- /content: Provides management of business process definitions

The complete list of the REST API can be found in D2.3 - Design of an Open API for the NIMBLE Platform or at [10]. The content referred via the footnote can be visualized nicely on https://editor.swagger.io/

---

[10] https://github.com/nimble-platform/business-process-service/blob/master/src/main/resources/api.yml

5 Future Work

Although we have introduced an initial version of the **process design environment**, we have not activated it in the current version (Release 1, December 2017) of the NIMBLE software. We first wanted to collect user feedback for the available pre-defined business processes. After the initial feedback round, we will make the design capabilities available with a user interface that is tailored to the needs of NIMBLE.

Security and privacy is one of the primary topics for future work. Improvements related to this topic will address role-based rights to members of an individual company to initiate specific business processes, configuration of the target party / user who is allowed to receive the message to be exchanged in a process / specific process step and role-based access to processes.

The improvements mentioned above are partially related to ownership management of business processes, which is not yet in place in the first version of the NIMBLE platform. The ownership mechanism in combination with the extension approach will be clarified according to the emerging requirements. We will be working on both the back-end and front-end services based on the priority of the requirements.

We also aim to establish the required infrastructure to make B2B communication across distinct instances of the NIMBLE platform. New capabilities will be added in the scope of federation-related activities.

In case of any further activity, we will keep this document up-to-date.

6 References

Heaney, B. (2013). *Supply Chain Visibility - A Critical Strategy to Optimize Cost and Service.* Aberdeen Group. Retrieved from www.gs1.org/docs/visibility/Supply_Chain_Visibility_Aberdeen_Report.pdf