UN/CEFACT Smart Container Project

UN/CEFACT Smart Container Project

T & L Domain - Status Update

Hanane BECHA, Project Leader
Michael Schroeder, Editor
Todd Frazier, Editor

02/04/2019

Outline

I. Project Status Timing

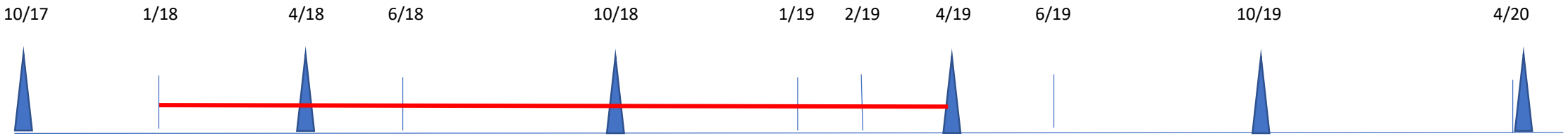
II. Project Steps

- Step 1. White Paper - completed
- Step 2. BRS Development – in process
- Step 3. Messaging Development – in process
- Step 4. API Development – planned

III. Summary

I. Project Status Timing

Project Time Line



PAST

- **UN/CEFACT Forum – 10/17, Rome, IT - Project proposal and head of delegations approval**
- **Interim – T & L Domain, – 1/18, Start of the Smart Container Project**
- **UN/CEFACT Forum – 4/18, Geneva, CH**
- Interim – 6/18, Work Session, hosted by CIF/David Roff, Liverpool, UK
- **UN/CEFACT Forum – 10/18, Hangzhou, CN**
- Interim – 1/19 Semantics and Message Structures, Smart container data model review , Paris, FR
- White Paper published, UNECE – 1/19
- Interim – 2/19, Smart Container data model and BRS document, Marseilles, FR
- **UN/CEFACT Forum – 4/19, Geneva, CH**

PLANNED:

- Interim – 6/19, Messaging and API Work Session, TBD
- **UN/CEFACT Forum – 10/19, TBD**
- API Development – date TBD
- BRS document, publication target – 4/20
- **UN/CEFACT Forum – 10/19, TBD**

NOTE: Between face-to-face meetings we have been conducting weekly conference calls to continue progress

II. Project Steps

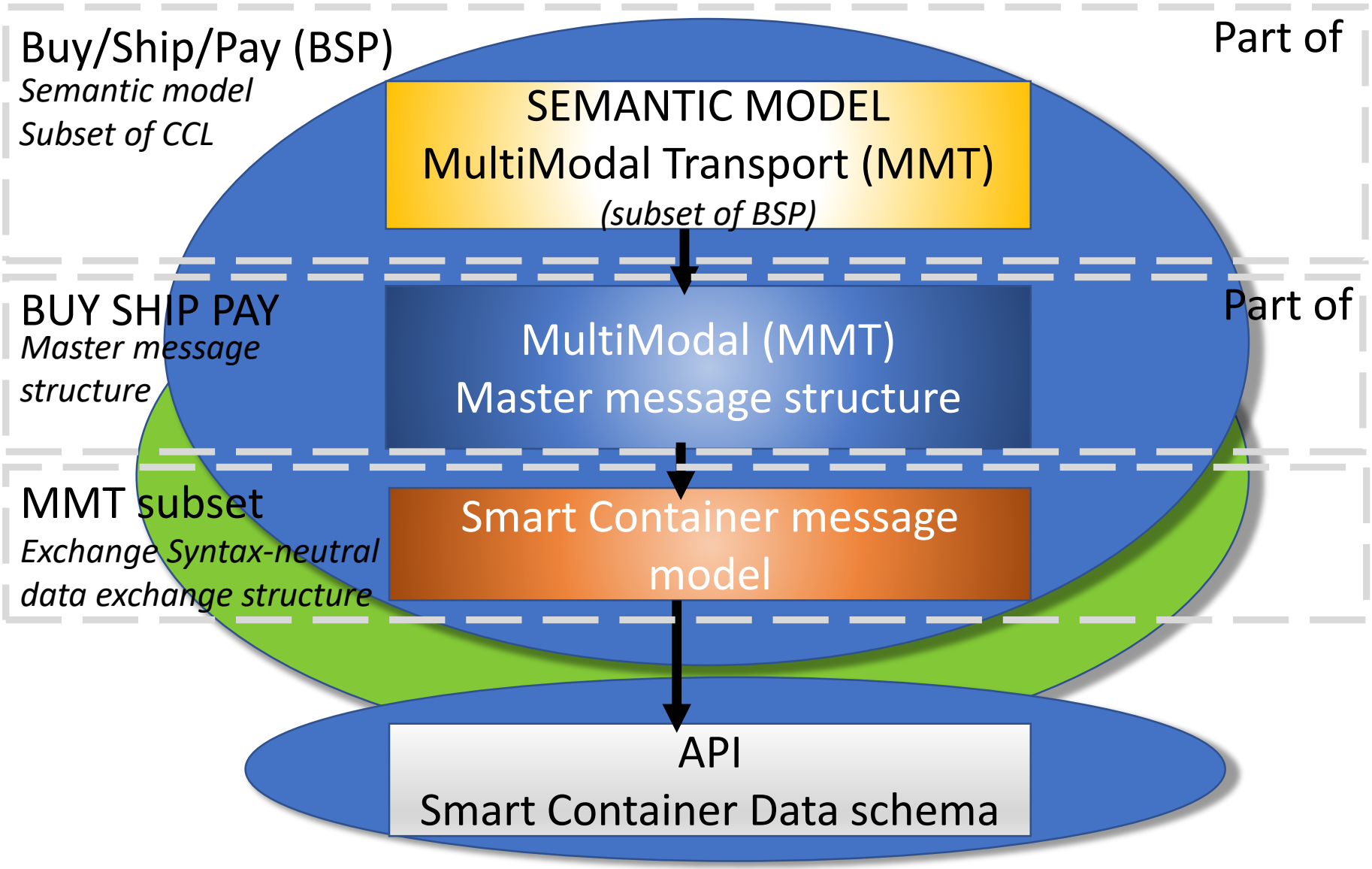
UN CEFACT T&L Domain Smart Container Project

Step by Step: from Data Elements to APIs

2 Current efforts

	Steps	Deliverables	Resources
1	Share a common understanding of the Smart Container Business use cases & stakeholders: **SCOPE**	Smart Container **White Paper**	Project Working Group from different backgrounds
2	Define structured data elements generated by smart container and their qualifiers **TERMINOLOGY /SEMANTIC**	Business Requirements Specifications (BRS) & **Entities Relationship Diagrams**	UN/CEFACT CODES Lists & Multi Modal Transport Reference Model (MMT)
3	Select the data elements for a given use case	**Generic message structure** (Technology Neutral!)	Contextualized Notification Messages Structures
4	Choose the **SYNTAX** (language) to be used to communicate	**APIs**	Multi Syntax World

UN/CEFACT Smart Container Modeling



Step 1: White Paper on real time Smart Container data for supply chain excellence

United Nations ECE/TRADE/C/CEFACT/2019/10

Economic and Social Council

Distr.: General
17 January 2019

Original: English

Economic Commission for Europe

Executive Committee

Centre for Trade Facilitation and Electronic Business

Twenty-fifth session
Geneva, 8-9 April 2019
Item 7 (c) of the provisional agenda
Recommendations and standards:
Other deliverables for noting

White Paper on real-time Smart Container data for supply chain excellence

Summary

The Internet of Things (IoT) is the capability of devices to communicate information to a network or to stakeholders directly. Combining this technology with containers in international trade results in "smart" containers that can communicate a great deal of information to the rest of the supply chain and provides a great deal of benefits to all involved such as greater visibility, real-time tracking, less waste (linked to temperature or humidity variations), higher security and potentially faster border clearance. This White Paper outlines the benefits and potential use cases of Smart Container technology and establishes the basis for a future electronic standard on the subject.

Document ECE/TRADE/C/CEFACT/2019/10 is submitted by the UN/CEFACT Bureau to the twenty-fifth session of the Plenary for noting.

White Paper Excerpt

4. List of use cases of the smart container potential usage

Below is a summary of the Use Cases put together by the UN/CEFACT Smart Container work group. Each use case will be described in more detail and with the required data elements in a future Business Requirements Specifications (BRS) Document to be developed by the Smart Container Work Group.

Case Number /Type	Use Case	Description & Trigger	Receiver	Value Proposition
1 Operational	ETA Update	Message with new ETA at next point or at final destination can constantly be sent out. ETA calculation is based on comparing planned and actual time and distance.	Supply chain stakeholder (Carrier, Terminal, Forwarder, Authorities etc.)	Receiver can react proactively and plan container operations or cargo logistics accordingly
2 Operational and Security Awareness	Actual Executed Transit Time	Monitoring the execution of completed transports. For any leg of the trip, compare used time with initial estimation (e.g., the initial trip plan).	Supply chain stakeholder (Carrier, Terminal, Forwarder, Authorities etc.)	Determine bottlenecks / Delay causes along the trip for operations excellence. Collect historic data as basis for future trip calculation / prediction.

Contributors - Smart Container White Paper

1. Hanane BECHA, Smart Container Project Leader (TRAXENS, Marseille)
2. Michael Schroeder, Smart Container Project Editor (Hapag-Lloyd AG - Project Manager / IT Consulting, Hamburg)
3. Todd Frazier, Smart Container Project Editor (FedEx/IATA, USA)
4. Stellios Stratidakis, Marine Traffic, Athens
5. Jaco Voorspuij, GS1
6. Laurent Gonzalez, GS1 France
7. Bertrand Geoffray, BIC , France
8. Jorn Heerulff, BIC,
9. David Roff, CIF, UK
10. Christophe THUILLIER (Port of le Havre.. Unfortunately, he is deceased a day or two before Christmas!)
11. Victor Dolcemascolo, French Gouverment
12. Joelle Friedmann, Geneva
13. Cleiton Alves dos Santos Joao Simoes, Brazilian Customs
14. Rudy Hemeleers, E-CMR Project Editor, DTLF rep. and an Independent consultant Luxembourg
15. Lance Thompson, Secretary, UN/CEFACT - UNECE Geneva
16. Ian Watt, UN/CEFACT Vice Chair: International Supply Chain, Australia
17. Sue Probert, UN CEFACT Chair and an Independent consultant

Step 2:

- Described Use Cases in More Detail
- Developing Semantic
- Data Elements and Descriptions
- Data Elements Matrix
- Use Case Relationship Diagrams

Business Use Case 8: Operational - Short-shipped: Forgotten containers on the peer or ramp

BRS Excerpt

Priority 2

Value proposition: It happens that the cargo is manifested but not loaded. It could happen as well during transshipment that the cargo is loaded on the inbound transport means but not unloaded [HB1] at the transshipment location or unloaded from the inbound transport means but not loaded onto the outbound transport means. Smart containers could detect this short-shipped event before arriving to the next port of call or port of discharge.

How:

Container is still sending its signal from the port of loading after the vessel has sailed where it should have been loaded. In addition, meshing technology used by smart containers could assist shipper and consignee in understanding which containers are not associated with the ship during its current voyage. A certain range of location from the current ship's position would identify that it is not on that ship. Or if containers are reporting in and a reliable signal is obtained through the use of meshing from containers not on the manifest, it would be known that a container was loaded on the wrong ship.

Example:

A ship left Southampton and proceeds 10 nautical miles en route to the Mediterranean, but two containers shown on the manifest are not transmitting a signal from the current ship's position, rather they are identified by position as still in Southampton. Also, one reefer is transmitting a signal from the ship, but it is not recognized as a container that was to have been on that ship's manifest.

Conclusion/Benefits:

The stakeholders including the shipper, the Vessel operator can take corrective operational action and correct the manifest or stowage plan.

Use Case	Short-shipped: Forgotten containers on the peer or ramp
Sender	Smart Container Solution Provider
Receiver	Shipping lane (vessel operator), Shipper, terminal operator
Trigger	Exception Driven: distance between the AIS position of the vessel and GPS position of the container (e.g., over one mile)
Preconditions	Trip plan entered, ID and AIS of the vessel
Data Transmitted	Container ID, booking ID, GPS, timestamp, Alert

Semantic: Smart Container

This was a start on defining the semantic, currently in progress, for the Smart Container Project. This is based on the Multi-Modal Transport Data Model.

Sensor

- Sensor ID
- Sensor Owner ID (optional)
- Sensor Manufacturer ID (optional)
- Sensor Position (optional)
- Sensor type Enum
- Sensor Unit Enum
- Characteristics Range of Values [X, Y] (optional) ??
- Expected Values Setting Configuration range [A, B] (optional)

TimeStamp

- Universal UTC
- Date
- Hour
- Minutes
- Seconds

BRS Excerpts

ZOI (Zone Of Interest are the places in "the itinerary/trip plan" for geofencing/facility) ??

- Place Type Enum {Depot pick-up, Master Contract Consignor Place Of Acceptance, POL, POT, POD, FDD, Depot return} (optional)
- Indicator Public information OR Private information (optional)
- Place Code (optional)
- Place Name (optional)

Responsible Container Party

- Responsible Container Party ID
- Contact
- Email (optional)
- Phone number (optional)
- Indicator ZOI or Leg
- Linked to the leg/ZOI of trip plan

Booking Reference??

- Booking Reference ID (optional, example empty container)
- Transport Equipment Operator (optional)
- Transport Equipment ID
- Trip plan {ordered set of legs (start ZOI + end ZOI) + mode of transport + Party+ ETD/ETA}??
- House Transport Contract Reference(s)
- Indicator Smart Booking
- Requested/Expected measurements for the different sensors (e.g., humidity, temperature settings)

Data Elements and Descriptions

BRS Excerpt

4. Shipment/Booking

Data elements related to the shipment context (synonyms: booking, conveyance) that the container belongs to at the time of event reporting.

Definitions – proposal:
Trip = end-to-end routing from first pick-up to last drop-off e.g. from Shanghai to Gothenburg
Leg = single route from one point in the trip to the next point e.g. from SGSIN to LKCMB.
One trip consists of many legs.
Synonym could be "Segment"
➔ *to be aligned with CCL/MMT wording, or with Data Pipeline*

Data Element	Description	Format	Example
Booking number	Booking reference issued by the container operator, typically the shipping line. Synonym: Trip-ID. ⇨ Leave empty if container not in mission at time of event reporting	Char(15)	12345678
Container Operator	The party that issued the booking reference and that operates the	Char(3)	HLC

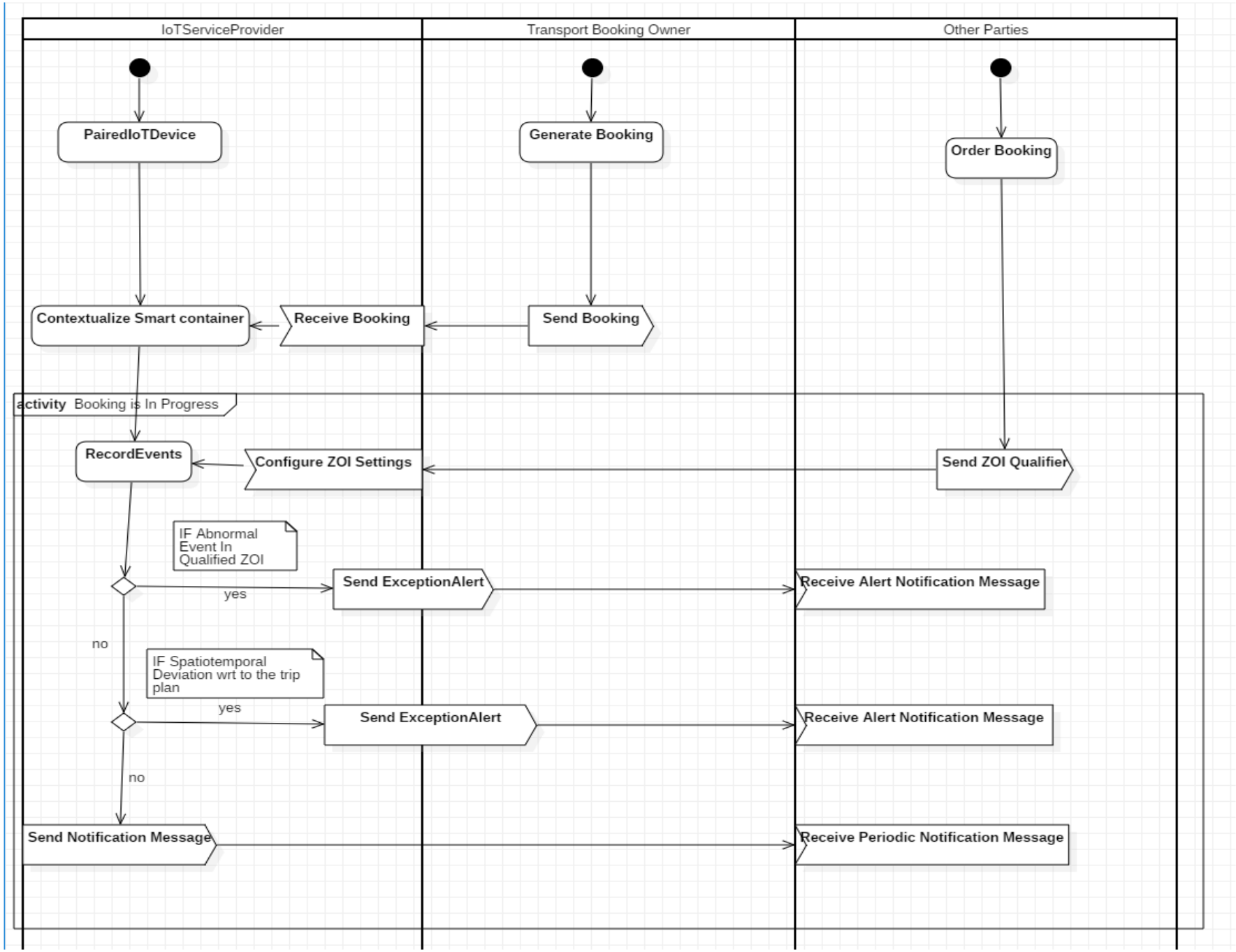
Data Elements Matrix

BRS Excerpt

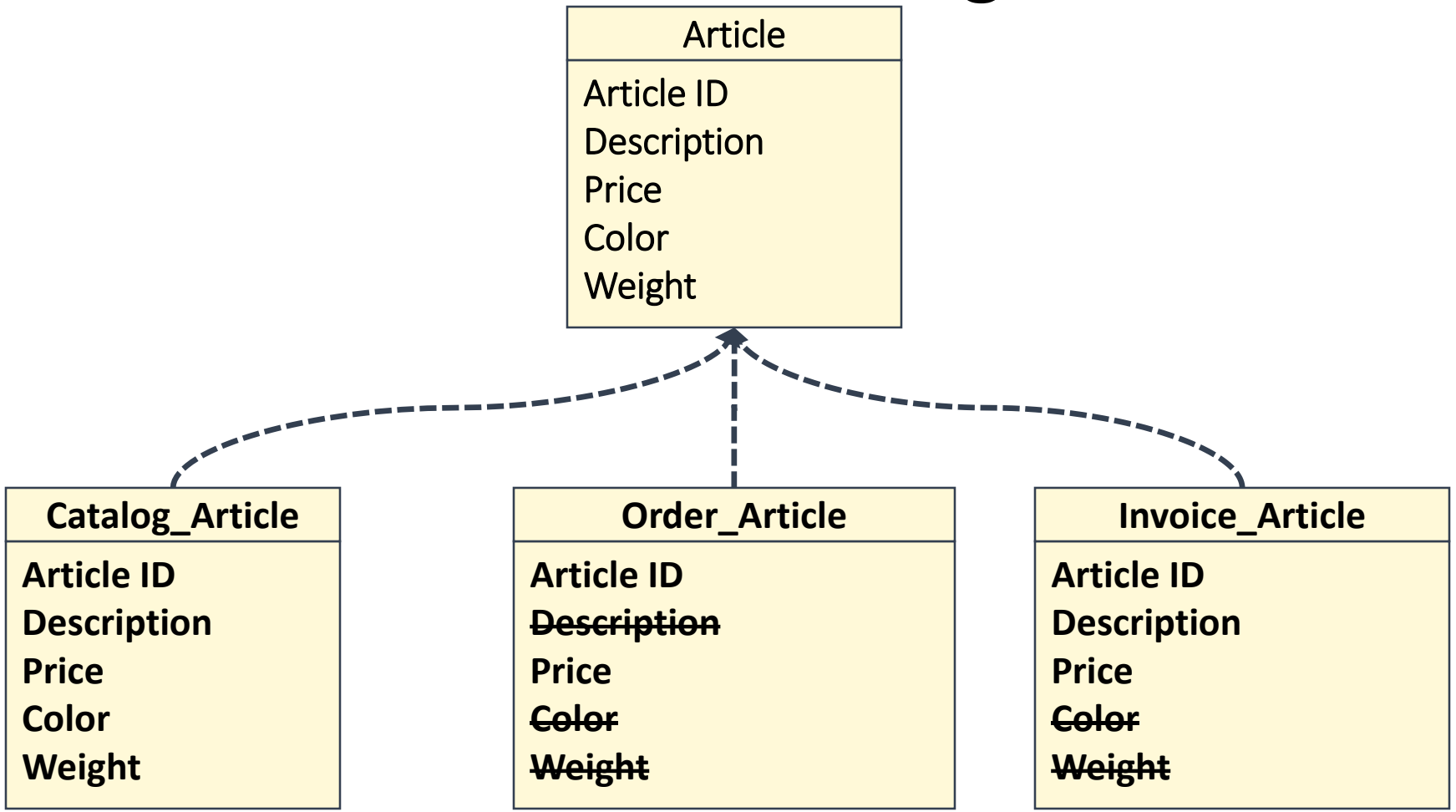
Use Case # Data Elements	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Device position	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Device life time indicator	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
2. Sensor	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Sensor ID	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Sensor Manufacturer	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Sensor Owner	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Sensor position	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Sensor life time indicator	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Sensor Type	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
GPS	X	X	X									X	X			X	X	X	X	X
Temperature					X						X					X			X	
Humidity						X										X			X	
Shock										X						X			X	
Gases																X			X	
Door Latch				X											X	X				
Active							X	X	X					X				X		

Relationship Diagrams

BRS Excerpt

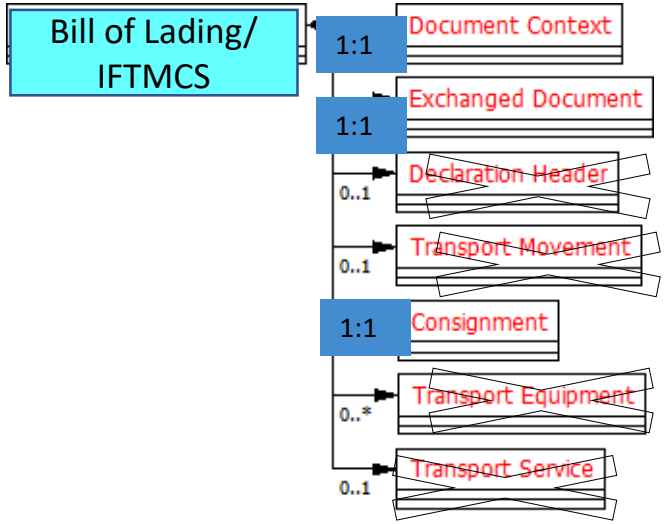


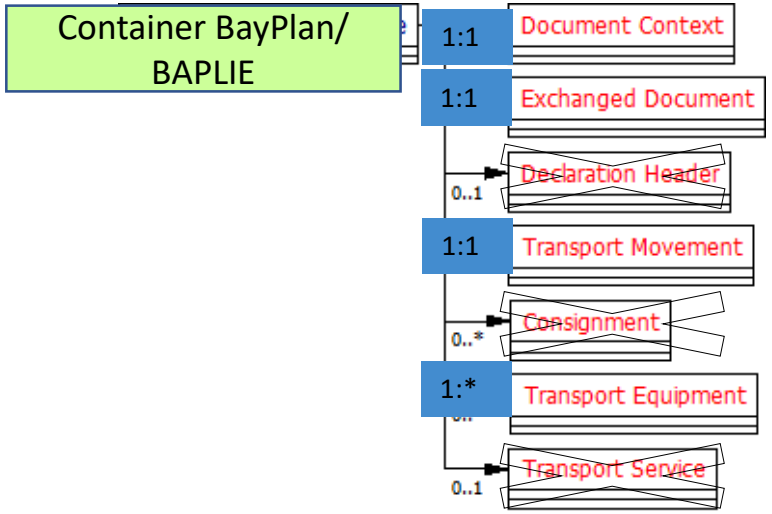
Component Library Approach: Reduction of complexity through reusable document building blocks

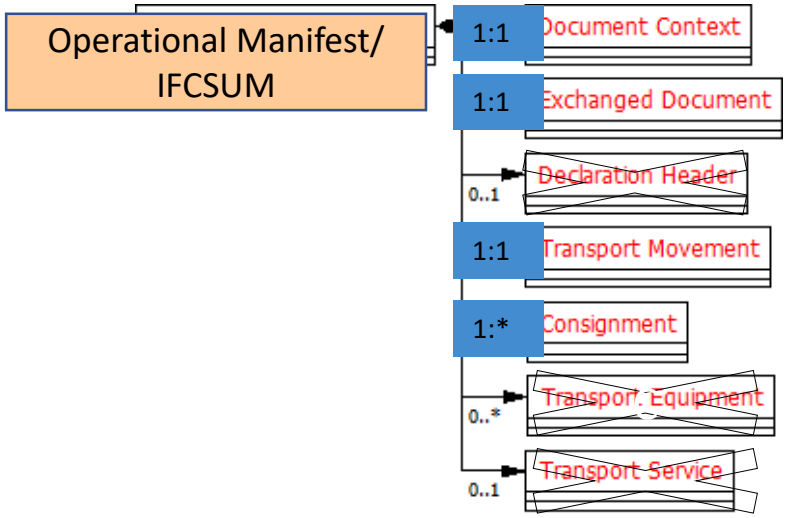


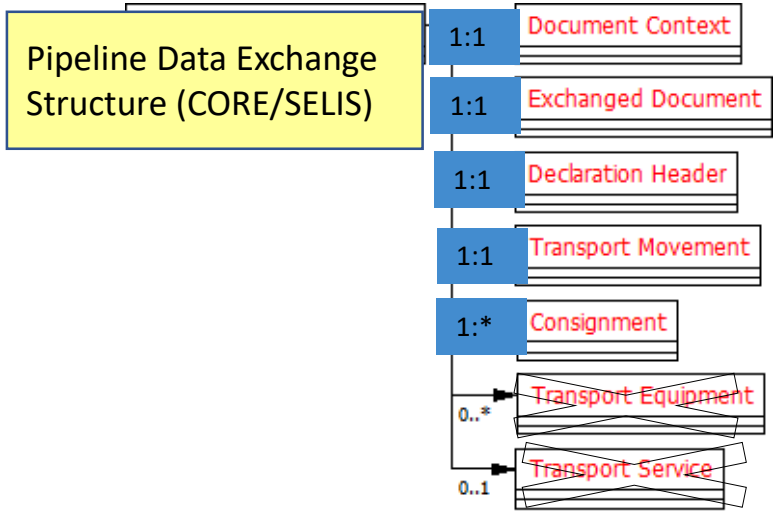
Contextualization by omitting non-used elements

Contextualized messages structures

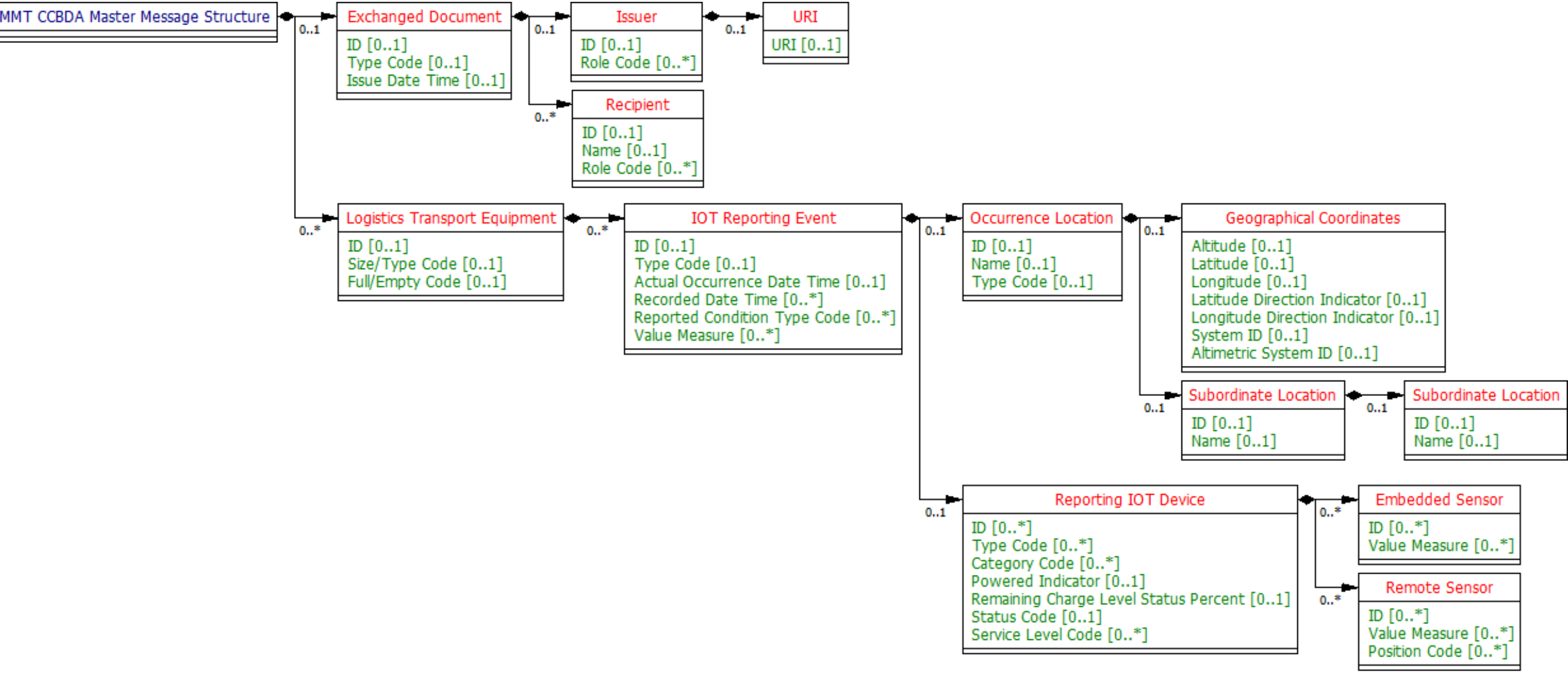








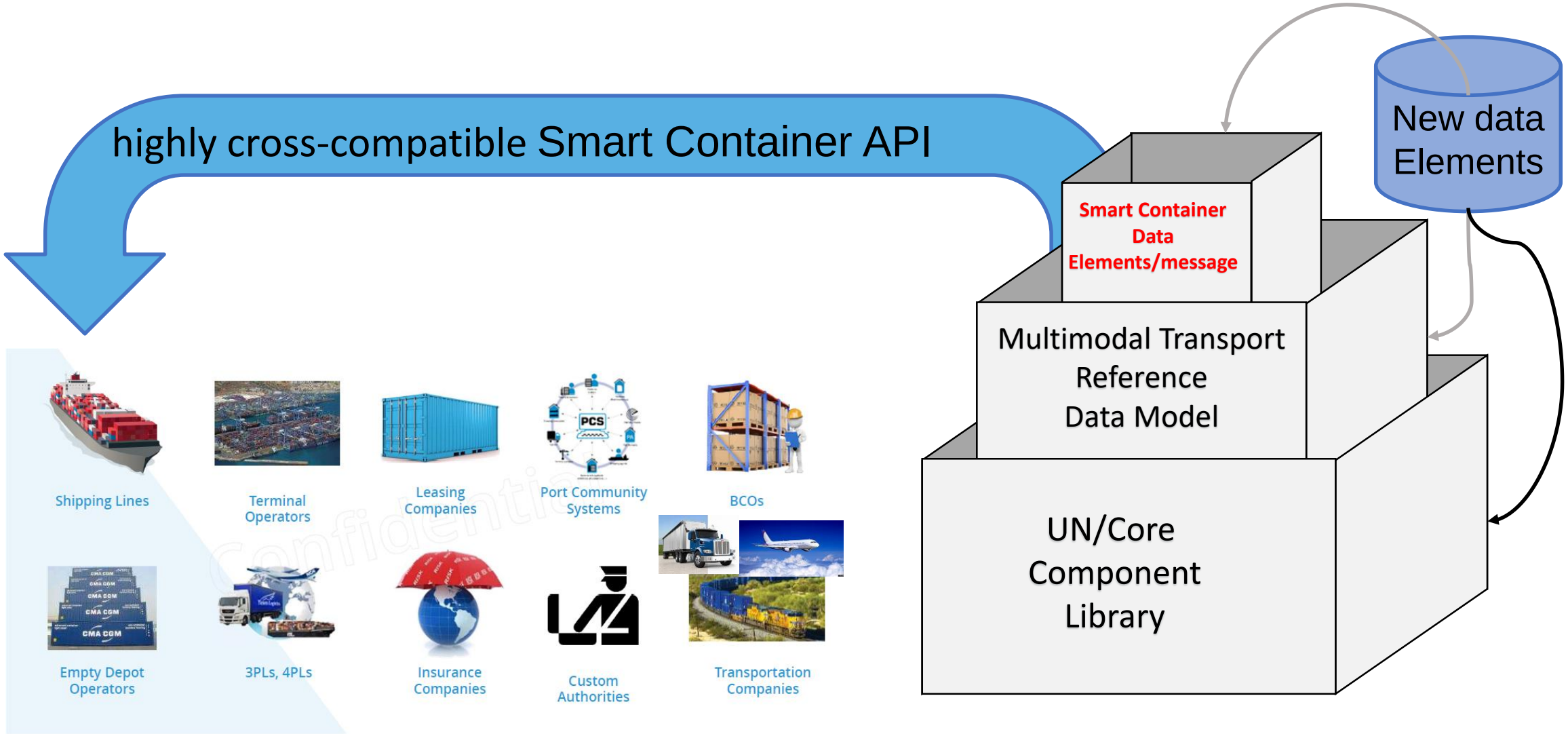
Step 3: Example Smart Container Message layout (work in progress)



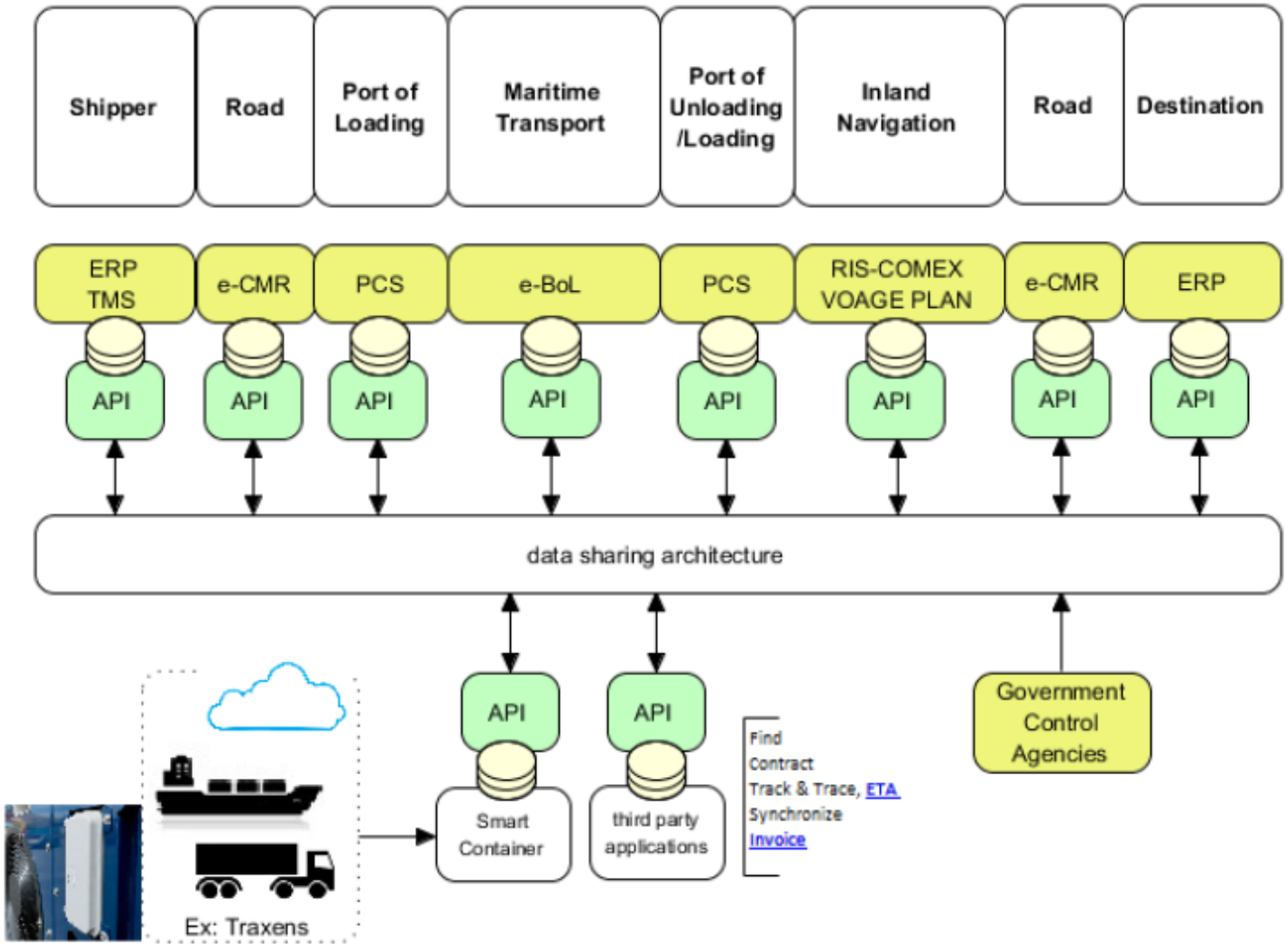
Step 4 - APIs for SOA

- **Service Oriented Architecture (SOA)** is an architectural methodology built upon the concept that capabilities should be implemented as services. The ‘Client’ can utilize any software component following the usage specification, irrespective of the technologies upon which the service was developed or upon which the 'calling client' was developed.
- **Application Programming Interface** (**API**) is a source code-based specification to be used as an interface by software components (services) to communicate with each other. Independent SOA services communicate using a common API.
- **APIs** can be created in any chosen syntax (Web Services) based on standardized syntax-neutral data exchange structures (Master data exchange structure)
- **API** is a layer for providing data access. API's should be architected with SOA support in mind (e.g., JSON, REST, SOAP).

API based on Generic Master Message structure to serve the whole ecosystem



APIs: JSON, REST, SOAP, etc.
Multi Syntax world



III. Summary

Summary

- CCL as semantic building blocks for the definition of business data for processes - **Context free** – reuse in multiple business sectors

- Customization of generic core components to specific business sectors and application domains

- Specific APIs (code-based specifications) to be used as interfaces by different services (software components) to communicate with each other and build a composite service.