

Regulation of Telematics in Dangerous Goods Transport

-
WP500 Data/Process Modeling

-
WG Telematics Meeting 7 – Bordeaux 2011/01/17



Josef Kaltwasser
AlbrechtConsult GmbH

Expected results

- ▶ **Models – extracted from and/or based on the *WHO DOES WHAT* spreadsheet – that can serve several needs in the future**
 - A high level overview of the domain of dangerous goods transport (“domain model” or “ontology” – basic concepts, not tied to particular applications)
 - An input to future standardisation / certification processes (on lower, more technical level)
 - A reference framework to align with external, relevant standardisation processes (i.e. mapping models used in external standards / systems into the logical domain of dangerous goods transport, without necessarily claiming syntactical structure)
- ▶ **Focus is on a data / information model, but processes will be included as far as possible under the given project constraints → only for processes dealing with DGT – modelling business processes in Freight & Logistics in general is out of scope**
- ▶ **Models shall be aligned with selected relevant external specifications (from standards or systems)**

Initial considerations

▶ What is a model?

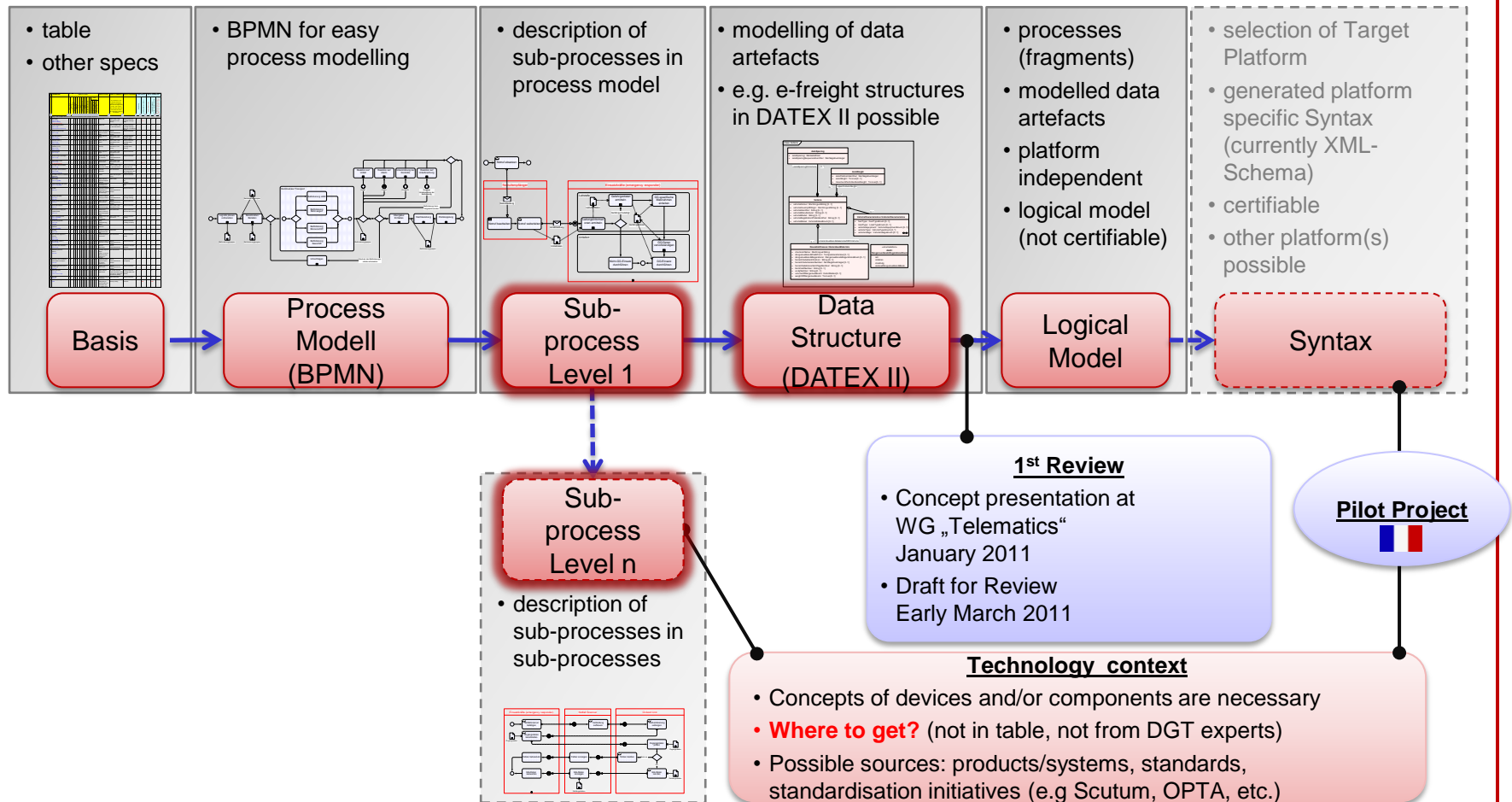
A model can be characterised by three features*

- **Mapping** – the model is a mapping of something real
 - **Reduction** – the model is (potentially) less complex than the original because it reflects only relevant features
 - **Pragmatism** – the model behaves like the original for the purpose of the modelling effort
- ▶ **Especially the latter is difficult since it requires good knowledge about the purpose of the modelling effort, whereas we are fairly vague on this point in this phase of the study and the modellers are no domain experts**
- ▶ **Suitable data modelling methods and approaches seem much clearer than for the process modelling domain, especially given that regulation of underlying business processes will only be partial, i.e. full modelling of these processes is out of scope**

Required input

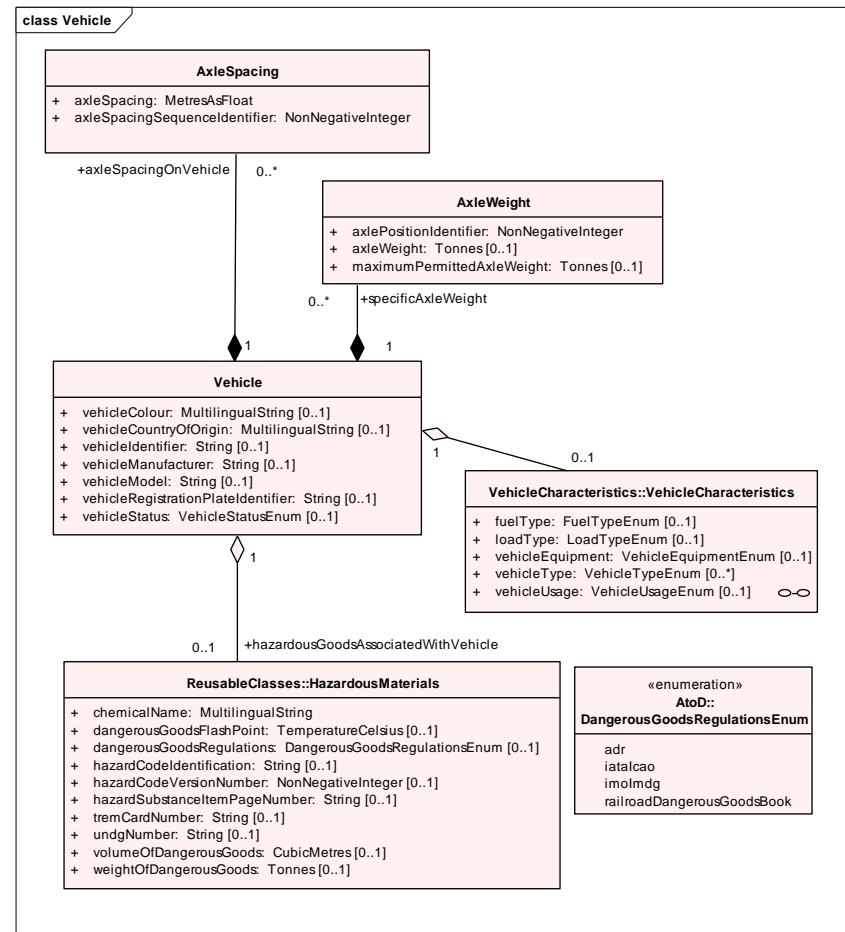
- ▶ **WP500's most important input is the *WHO DOES WHAT* table**
 - It contains many details about relevant metadata for data / information modelling (although most require additional detail to allow formal modelling)
 - It contains many important – direct or indirect – references to relevant sources of information that govern referenced underlying data / information model elements
 - References to the relevant sections of ADR / ADN / RID (column B)
 - “Transport document”, “Website UNECE”, “Road sign”... (mainly column T)
- ▶ **WP200 will have to provide the relevant set of standards and activities that potentially will provide new standards**
 - The table covers current knowledge, the WP200 output will provide relevant future conditions that should be taken into account
example: No. 49 (“*Alert-system for road traffic incident/accident*”) should in the future probably be aligned with eCall standards
- ▶ **WP300 & WP400 may set model restrictions due to security and certification reasons**
- ▶ **Modelling the *WHO DOES WHAT* table requires stakeholder feedback from WG Telematics and other relevant organisations / projects (e.g. eRailFreight, IATA e-freight, SCUTUM... – a potentially wide range that needs scope and focus!)**

Roadmap: From table to model

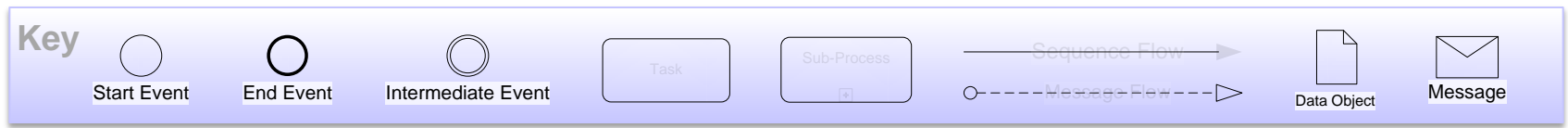
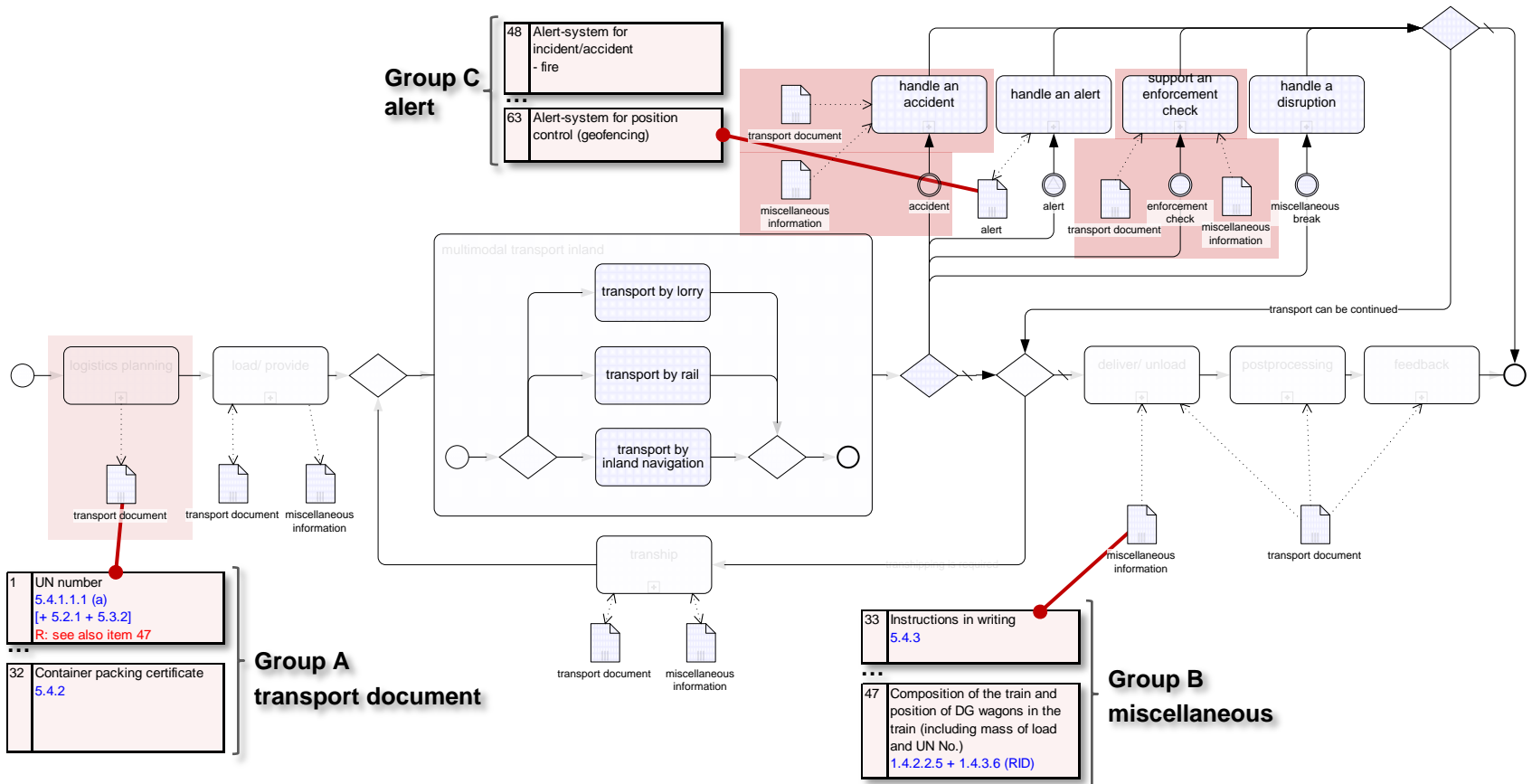


Work package approach Methodology

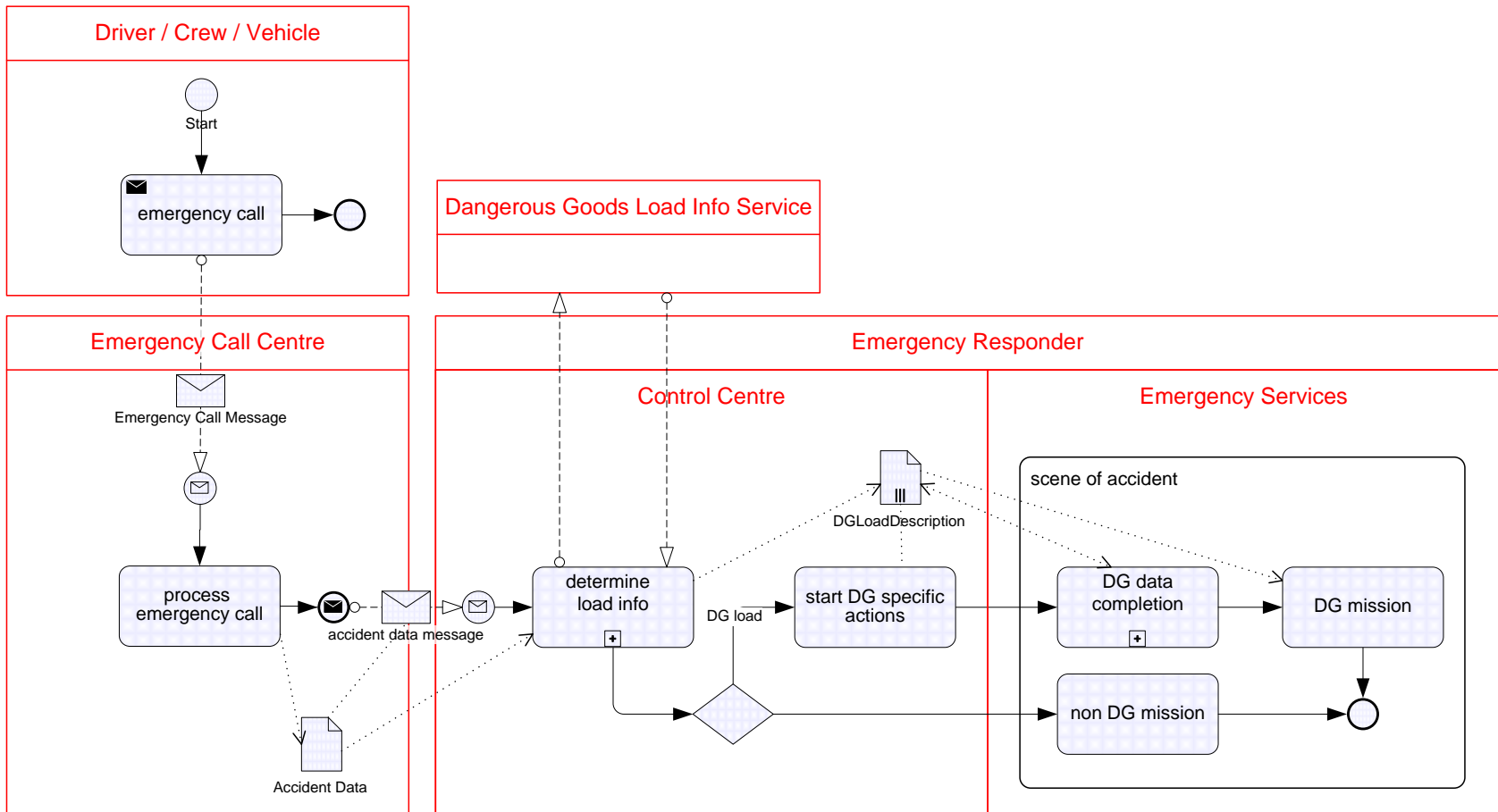
- ▶ **Data modelling will use up-to-date methods, in particular UML**
- ▶ **DATEX initiative provides starting point**
 - UML profile suitable for this type of modelling effort
 - Defined mapping to XML schema definitions (incl. Tool) (sometimes preferred by stakeholders; closer to implementation)
 - Detailed (technical) part of the resulting model can be fed back to DATEX for CEN standardisation, providing alignment of future versions of the interface standard for traffic management
- ▶ **Suitable accompanying methods for process modelling in WP510: BPMN – Business Process Modeling Notation (OMG)**



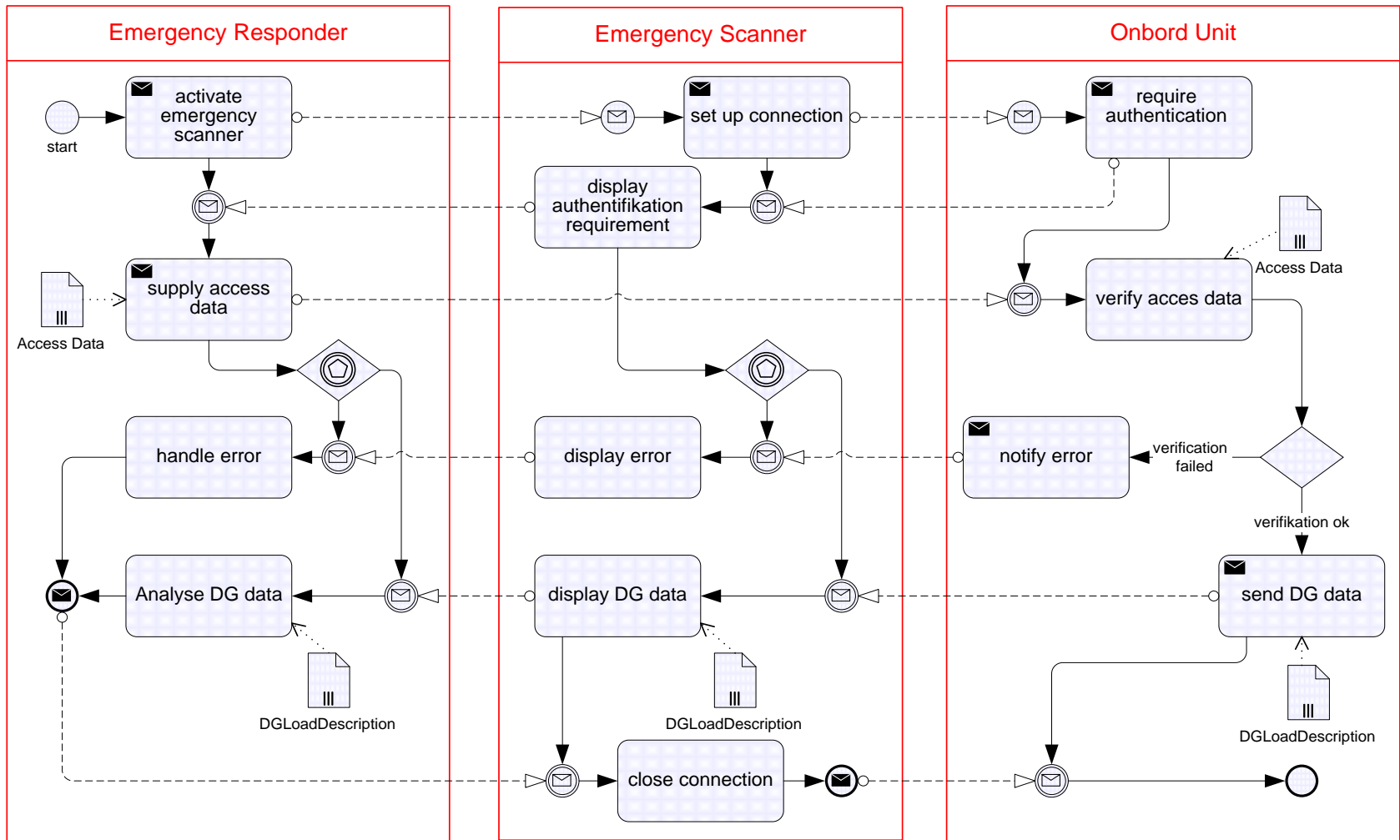
Process Dangerous Goods Transport



Sub-Process „handle an accident“



Sub-Process „DG Data completion“

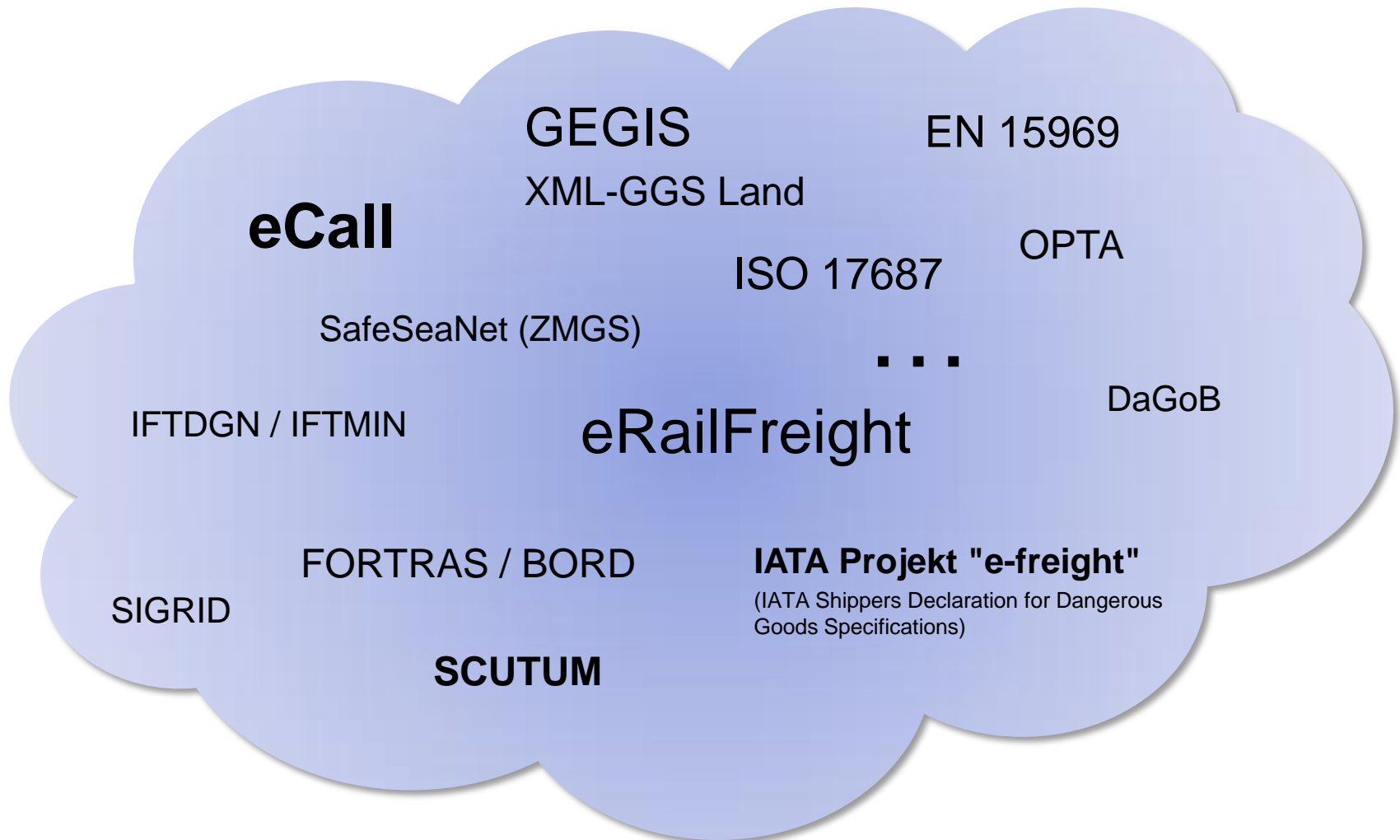


Work package approach

Basic modelling requirements

- ▶ **The *WHO DOES WHAT* spreadsheet needs to be processed further to provide more suitable input into standardisation and certification processes**
 - Human language must be transformed into a formal description language
example:
(No. 1) “UN number” → *UnNumberType ::= Digit[4] or better {0001..10000}*
 - Domain concepts taken from other specifications need to be obtained from the quoted source and compiled into self-sustained (IT-)definitions
example:
(No. 46) “Tunnel category (road)” → *TunCatType ::= {'A'-'E'} // ADR 8.6.2*
 - Entities, relations and attributes need to be fully qualified
 - Which elements are related to which other elements (e.g. *PressureSensor* related to *Tank*)
 - What is the nature of the relationship in the model (association, aggregation, composition...)
 - In which direction is the relationship navigable in the model?
 - Multiplicities
 - Permissible values for (primitive) attributes
 - Domain concepts not fully specified (either inside the table or in the quoted sources) need to be expanded and fully modelled (relying on expert advice or external input from other projects, e.g. e-freight, eRailFreight, SafeSeaNet, etc.)

An abundance of (potential) influences on the model



Data model

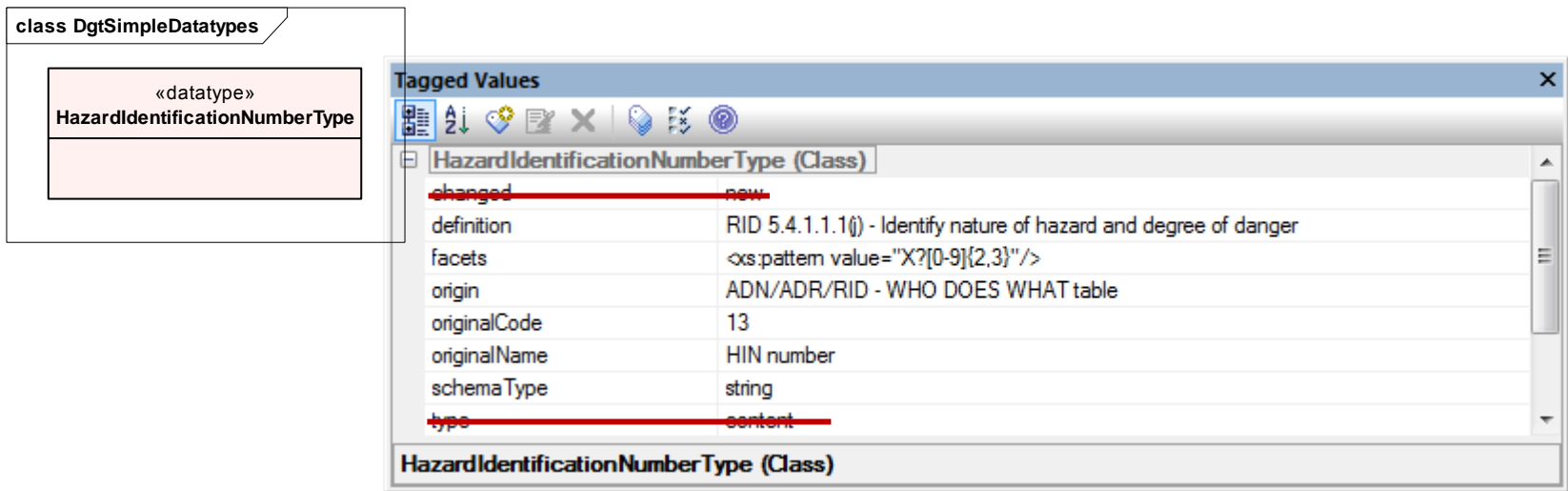
First results and modelling issues

► Simple data types

- Define the permissible value space of atomic data structures (“attributes”)
- Typical examples are *Boolean* (0/1), *NonNegativeInteger* (0, 1, 2, ... ∞), *String* (character sequence from given alphabet), etc.
- DATEX II reuses the primitive data types from the XML Schema standard (IT domain)
 - Ensures interoperability with current single implementation platform XML / XML schema
 - Reflects broad consensus in the EDI world, supported by many other B2B standards
- Are not canonical – the more constrained they are, the more prescriptive (= suitable for certification) is the specification.
- Example 1: UN number
 - Define an appropriate data type for *UnNumberType*
 - Proposal 1: Float – “2.3” would be a permissible value
 - Proposal 2: *nonNegativeInteger* – “123456789” would be a permissible value
 - Proposal 3: *nonNegativeInteger* + *minInclusive*/*maxInclusive* facets – only values from 0000 to 9999 allowed!
- Careful modelling of simple data types is important to ensure that the power of syntax checking is exploited as far as possible!

Simple data types derived from WDW table

Example – Hazard identification number



- ▶ **Simple types are modelled as classes with a «datatype» stereotype**
- ▶ **Additional metadata is captured in tagged values**
 - definition – reference to regulations taken from column B – with definition taken from column R of the WDW table added after “-” character
 - origin – “AND/ADR/RID – Who DOES WHAT table”
 - originalCode – line number in WDW table
 - originalName – black text in column B of the WDW table
 - schemaType – primitive type taken from the XML Schema Definition standard (part 2)
 - facets – additional mechanism from XML schema to further restrict the lexical space for permissible values; used here to specify an optional ‘X’ plus 2 to 3 digits – only where required

Data types based on *Strings*

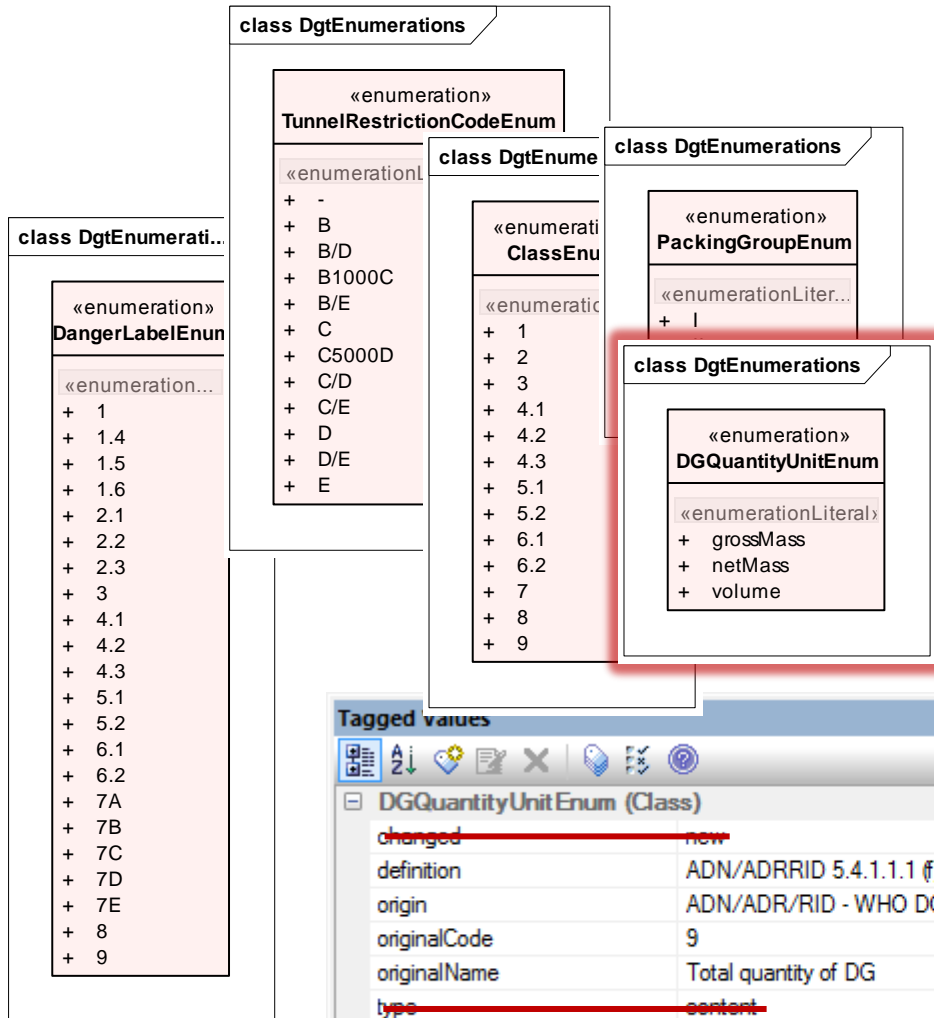
- ▶ The WDW table contains many definitions of value spaces for attributes that are canonically represented by *strings* (sequences of characters from a given alphabet)
- ▶ Some of these can have – more or less – arbitrary values (e.g. *consignorAdress.streetName*)
- ▶ Some string-based data types have only one representation (example: *streetName* – see above)
- ▶ Some string-based data types may have multiple instances that express the same value in different languages (see for example ADR 5.1.2.1 or 5.2.1.5)
- ▶ Some string-based data types require limitations of length and permissible characters
- ▶ DATEX II modelling features:
 - Strings based on the XML schema *string* primitive data type
 - The use of *facets* allow limiting length and defining *regular expressions*
 - DATEX II itself adds the concept of a *MultilingualString*, a string that has multiple instantiations in different languages inside one data structure



Strings with limited number of fixed values: *Enumeration* types

- ▶ In many cases, value spaces for attributes can only take a limited number of permissible values (examples: danger labels, tunnel restriction codes, etc.)
- ▶ The values look like strings, but in reality they are (self-describing) codes
- ▶ Assigning a value not found in the list of permissible *literals* is an error, the value for the attribute is invalid and interoperability is not guaranteed
- ▶ Input documents from external sources often do not use the power of restricting the values space via enumerations and specify attribute types as *string* instead
- ▶ **DATEX II modelling features:**
 - *Enumerations* and *EnumerationLiterals* as part of the meta-model
 - Used intensively to increase the quality of syntax checking in information processing (traffic management & information model: ~1500 literals)

Enumeration examples

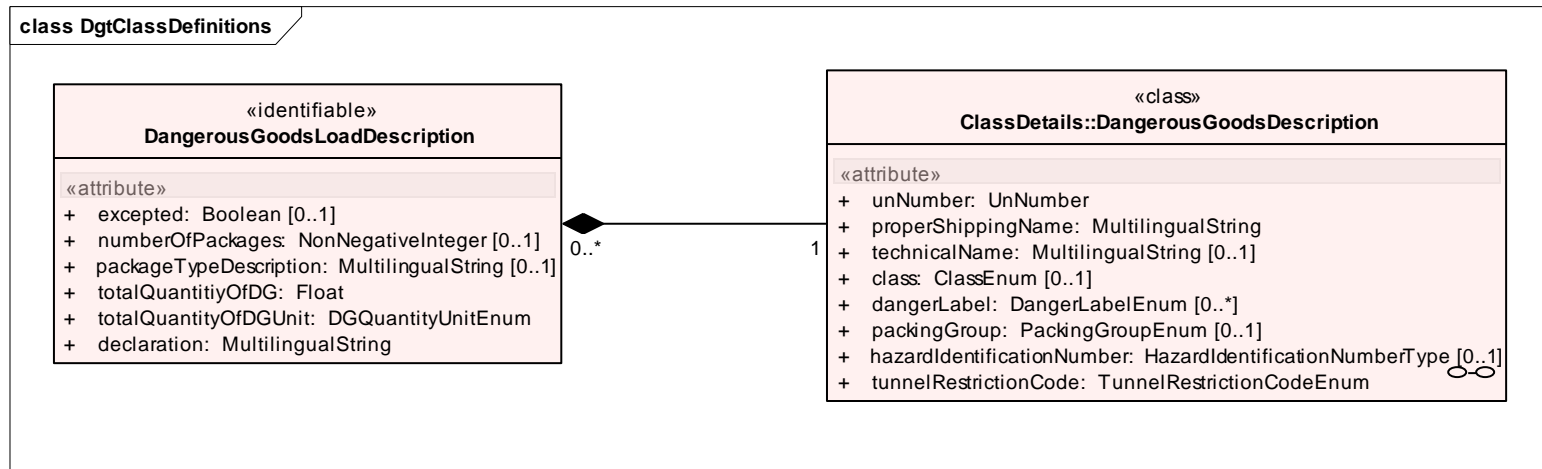


- ▶ Same tagged values as for data types (except for schema type, which is obvious)
- ▶ Same mapping from WDW table to tagged values proposed
- ▶ Feedback to DATEX: naming convention violated

Tagged values

DGQuantityUnitEnum (Class)	
changed	new
definition	ADN/ADRRID 5.4.1.1.1 (f) - Indicate quantity of individual DGs
origin	ADN/ADR/RID - WHO DOES WHAT table
originalCode	9
originalName	Total quantity of DG
type	content

Complex data types



- ▶ **Complex data types can be modelled as UML classes with attributes (e.g. Address consists of name, street, city...)**
- ▶ **Complex data types have their own metadata (e.g. definition) and can be re-used throughout the model**
- ▶ **Complex data types are recursive, e.g. CDTs can contain other CDTs that again contain CDTs, etc.**
- ▶ **Complex data types are used via class associations - they are named automatically (in DATEX II) if no explicit name is given**
- ▶ **Same options for multiplicities like in the case of attributes**

Complex data type description

class DgtClassDefinitions

«class»
ClassDetails::DangerousGoodsDescription

«attribute»

- + unNumber: UnNumber
- + properShippingName: MultilingualString
- + technicalName: MultilingualString
- + class: ClassEnum [0..1]
- + dangerLabel: DangerLabel
- + packingGroup: PackingGroup
- + hazardIdentificationNumber: HazardIdentificationNumber
- + tunnelRestrictionCode: TunnelRestrictionCode

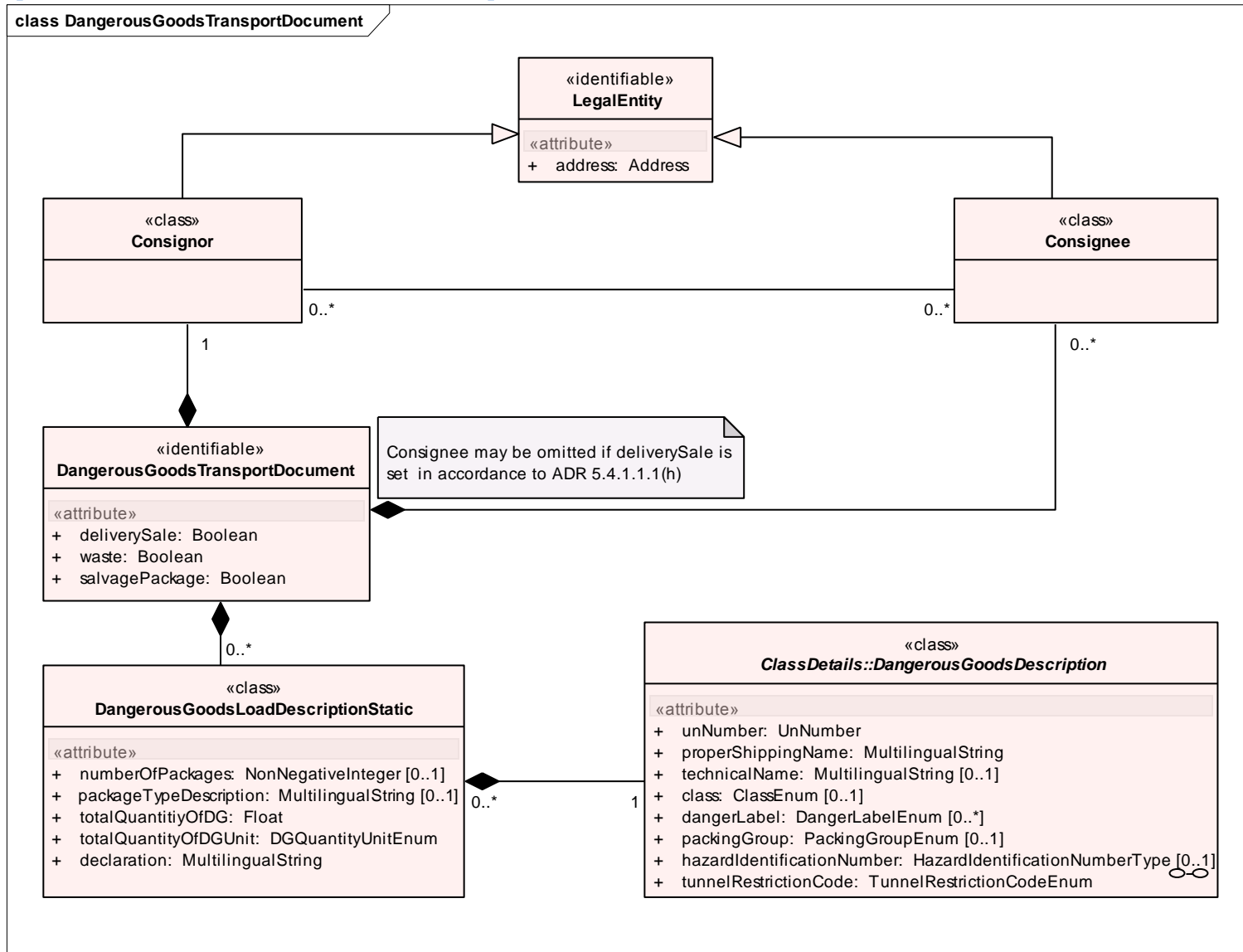
Tagged Values

DangerousGoodsDescription (Class)

changed	new
definition	???
origin	ADN/ADR/RID - WHO DOES WHAT table
originalCode	???
originalName	???
type	content

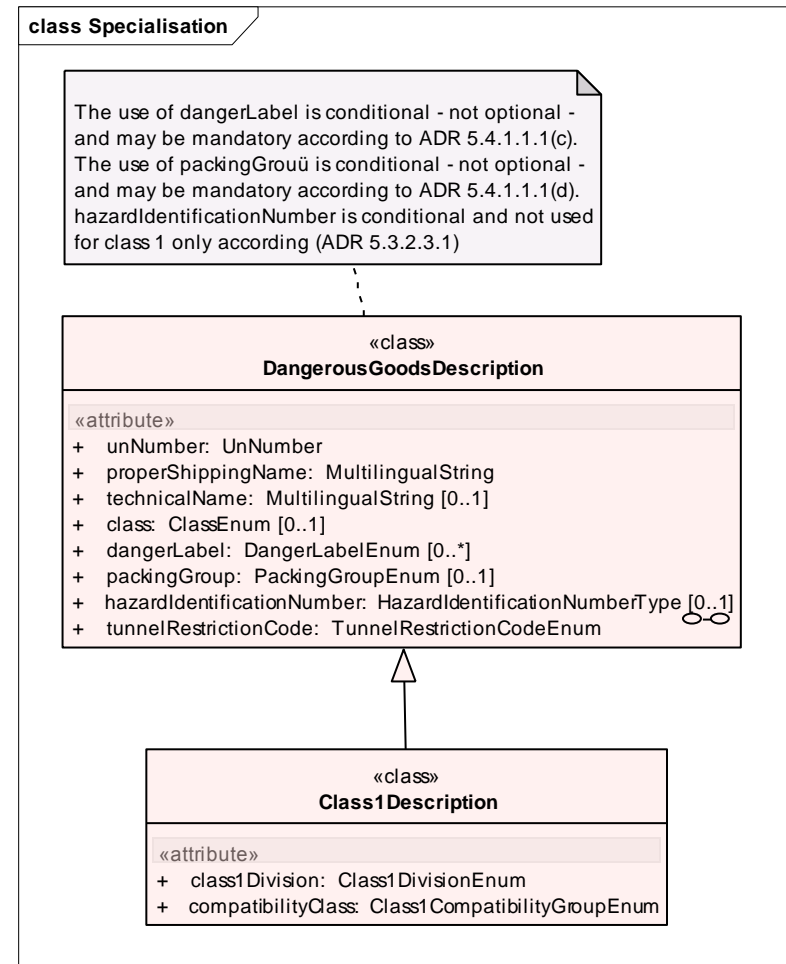
- ▶ The same set of descriptive metadata as for enumerations
- ▶ Problem: the data structures represented here are not systematically defined – neither in ADR, ADN & RID nor in the WDW table
- ▶ originalCode & originalName may be nulled – but a definition is mandatory and needs to be agreed with the WG

Top level model for transport document



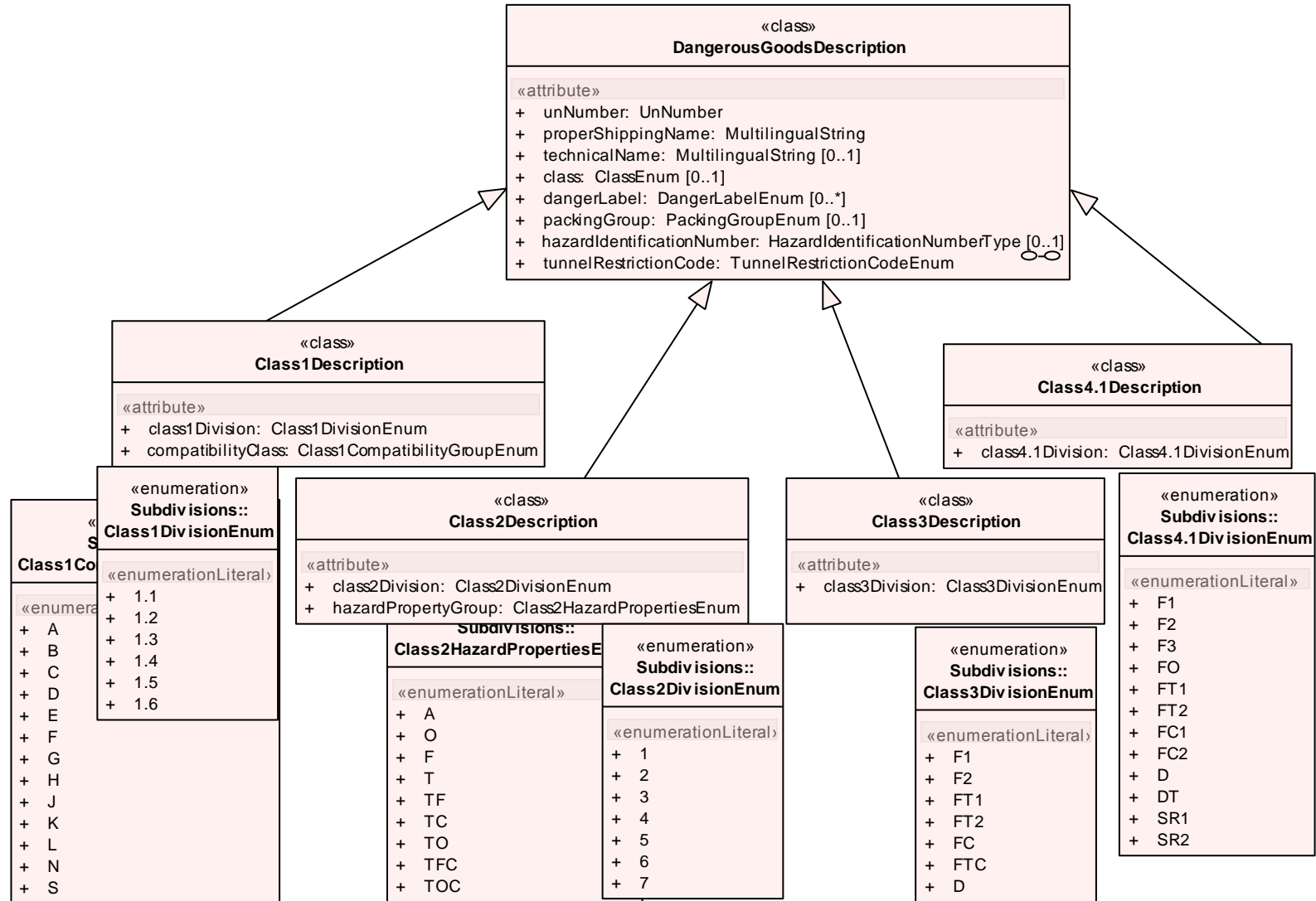
Advanced modelling – specialisation

- ▶ **Classes can be specialised into sub-classes**
- ▶ **These sub-classes “inherit” the data structure of their “parent class”**
- ▶ **Sub-classes can be put into all places where super-classes are allowed (“is a” relationship: a *Class1Description* is a *DangerousGoodsDescription*)**
- ▶ **Powerful tool to foster reuse in modelling**
- ▶ **Can be used e.g. to model the disparate sub-divisions inside the DG classes**



Specialisation example: sub-division of different DG classes

class Specialisation



Conclusions / observations

- ▶ **The DATEX II modelling method has proven during the first modelling steps to be a sound basis for modelling a Dangerous Goods model based on the WHO DOES WHAT table – plus additional input from ADR, RID & ADN**
- ▶ **Extra input from external stakeholders is helpful in providing input for data structure choices, but also bears potential for divergence and inconsistencies**
- ▶ **The overall amount of input material is overwhelming and stretching the project's resources**
- ▶ **From the first round already we can expect the resulting model to become substantial**
- ▶ **The current DATEX II model provides a mapping to XML schema which may prove to be helpful for alignment with other backbone specifications like Dakosy, eRailFreight, e-freight, etc.
(beware that other channels – esp. Radio – need different mapping)**

Modelling challenges

- ▶ **How to verify the ‘correctness’ of the model?**
 - The modelling experts are no DGT experts and get overwhelmed by the width and depth of the input material ⇒ model prone to errors and inconsistencies
 - The domain experts may feel uncomfortable with the modelling methodology and tools – also: a thorough review of an extensive UML model needs substantial time
 - Some metadata cannot be derived systematically from the table and/or the ADR/ADN/RID – where to get these from?
- ▶ **Data type modelling – *Enumerations* in particular – is a powerful tool, but implies maintenance to keep aligned with the regulations**
 - There might be a need in the future to change the structure of the ASR/ADN/RID to better support the link to modelling – e.g. an index on table 3.2.1
- ▶ **Difficult to find appropriate ‘pragmatism’ of model**
 - The table is a world model with a claim to capture the full width of the domain
 - Models for particular purposes need to be restricted to elements relevant for the purpose of the model
 - How to determine the elements that need to be present in the model – e.g. do we need to model labels, placards, yellow plates, etc. if their ‘content’ (the description of the load of dangerous goods) is already represented in the model?
- ▶ **How to cope with the abundance of input material and potential inconsistencies**

Thank you!

**Josef Kaltwasser
AlbrechtConsult GmbH**



**direct contact
Tel: +49 1520 877 04 02
Fax: +49 241 500 718
Josef.Kaltwasser@albrechtconsult.com**