PUBLIC TRANSPORT
NETWORK TIMETABLE EXCHANGE
(NeTEx)

DESIGN METHODOLOGY

CEN TC278/WG3/SG9 NeTEx PT001

Nicholas JS Knowles

OCTOBER 2015

TABLE OF CONTENTS

3 INTRODUCTION
CORRESPONDING NeTEx DOCUMENTATION
MODEL DRIVEN DESIGN
4 THE TRANSMODEL CONCEPTUAL MODEL
CONSISTENT TERMINOLOGY
UML NOTATION
3 XML
XML BENEFITS
6 XML DRAWBACKS
7 TOOLS & TECHNOLOGY
MODULARISATION OF THE FRAMEWORK

8 ACKNOWLEDGEMENT

FURTHER READING
THE NeTEx STANDARD
OTHER NeTEx WHITE PAPERS
OTHER REFERENCES
FURTHER INFORMATION

INTRODUCTION

The NeTEx (Network Timetable Exchange) standard is a CEN standard for exchanging public transport data [N1], [N2], [N3]. This paper provides a short discussion of some aspects of the design methodology underlying NeTEx, an understanding of which may help with interpreting and implementing NeTEx systems.

CORRESPONDING NeTEx DOCUMENTATION

The paper is intended to convey a high level view sufficient for a technical manager to appreciate the capabilities of NeTEx, and omits all detailed considerations - for a detailed description please see the full CEN NeTEx specification, in particular Part 1 [N1] from which sections of this paper are taken.

MODEL DRIVEN DESIGN

NeTEx uses a “Model Driven” approach to design, that is, the fundamental design is described as a high level conceptual model that represents the problem domain as entities and relationship that have been identified by a set of use cases and existing systems covering the desired business scope. This conceptual model is implementation independent, but is then elaborated to create a more detailed design for a physical model that can subsequently be transformed into a software implementation, either automatically or semi-automatically, using a specific technology – in NeTEx’s case “XML (other” implementation languages are also “possible).”
The use of high level models allows designs to be reviewed and validated by interested parties and to be fully documented with narrative text that describes the intention of the design. Implementing a data exchange format represents a significant investment by many different stakeholders and having such a model and documentation facilitates long term use of the model and schema by many different parties. Modelling in particular helps identify common abstractions and components that simplify the implementation. It also allows dependencies between components to be understood so that the system can be modularised in a way that minimises coupling and optimises flexibility. This in turns makes it easier for implementers to select just those components needed for a given purpose. It is also valuable for future evolution as the dependencies between components can be properly understood and the effects of a change evaluated.

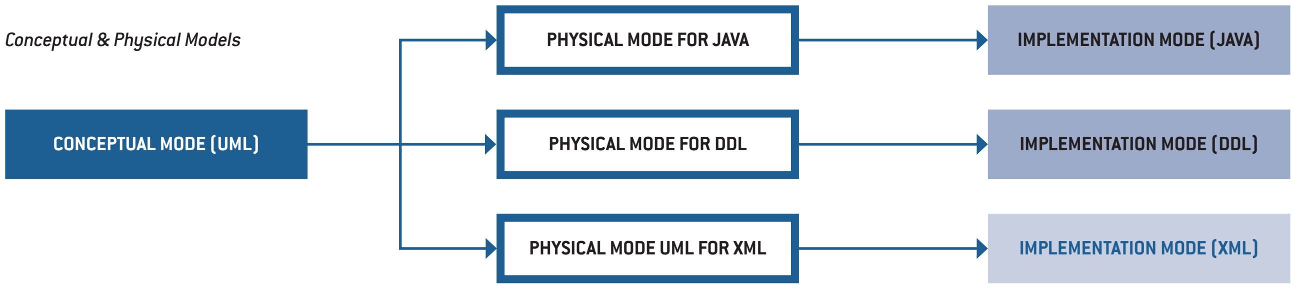


Figure 1 – Conceptual and Physical models

The Transmodel Conceptual model

The conceptual model used in NeTEx is based on Transmodel, the CEN Reference model for Public Transport [T1], [T2], [T3], [T4], developed over the course of the past two decades from a harmonisation and systemisation of a number of proven European National standards. Transmodel focuses on Transport concepts at a high level design level and as an object model. It does not need to address detailed implementation points such as the use of name-spaces, or how elements should be organised when serialised in a linear format such as XML.
NeTEx, as a concrete physical format which can be implemented, must make technology specific decisions on such points and also adds in functional attributes for a number of elements based on existing European transport systems

CONSISTENT TERMINOLOGY

One of the design goals of NeTEx is to uphold a consistent set of terminology for a given Public Transport data concept wherever it is used. The names of public transport concepts in everyday language are often fuzzy, the same word having different meanings in different contexts depending on whether one is discussing the passenger, the vehicle or the timetable, or describing a process or an entity. For example, consider "Stop", "Service", "Route", "Journey", etc, each of which has multiple meanings in English. This problem is compounded at a European level since different languages may have further differences in their semantic categories as well (that is to say a given term may cover a subtly different set of connotations in each specific language – as say with Reise/ Viaggio/ Voyage/ Journey/ Potovanje/ Reis, etc). Transmodel tries to follow a restricted, consistent terminology, using a single term for a single concept each with its own well-defined description. Quite often this results in a rather cumbersome technical vocabulary, for example a VEHICLE JOURNEY, SCHEDULED STOP POINT, ACCESS RIGHT ASSIGNMENT (Transmodel terms are given in uppercase by convention), and sometimes a term is used somewhat artificially to exclude some colloquial senses (for example in Transmodel TRIP is used only for passengers while JOURNEY is used only for vehicles), but the approach has significant advantages in precision and, once understood and adopted, in reducing overall complexity. Readers should however to be aware of this convention in reading NeTEx documents.

UML NOTATION

NeTEx represents the underlying model using Unified Modelling Language (UML) class diagrams [U1], stored alongside an electronic representation of the model in a repository. UML is an industry standard notation for describing complex software models and supports a number of different types of relationship between software components (associate, aggregation, inheritance etc). Two levels of UML model are provided – a high level conceptual UML model which is implementation independent, and a physical model which includes detailed attributes and details needed to support implementation as XML schema. Textual definitions are attached to the NeTEx schema elements as well. Each model is carefully modularised into packages within the NeTEx Part1, Part2, and Part3 parts, with a given package having correspondences at each level (UML dependency diagrams are provided to document the relationship between modules).

In practice, the use of a UML model requires the use of modern design tool (such as Enterprise Architect) that offer powerful navigation and visualization capabilities to examine a model and its documentation interactively in many different views. Such views may be included as illustrations in a document such as this but represent only a static snapshot. For in depth study of NeTEx use of a tool is recommended.

XML

NeTEx uses W3C XML schema to describe data. XML Schema Definition (XSD) is a general purpose language for describing data model elements in a form that can be serialized and transmitted between different computer system.

XML Benefits

XML allows for a semantically rich representation of data and has several important advantages over a flat file technology.

- **Validity checking:** a schema allows not only the structure of the model to be described but also many integrity constraints. These can cover not only basic data types (dates, times, numbers, allowed values for enumerations, etc) but also complex referential integrity conditions that can be used to ensure that a coherent data set is delivered, such as unique identifiers and satisfied cross references. This automates much of the data quality checking process and assists problem resolution between different participants.
- **Reuse mechanisms:** XML is a modern Object Orientated Language and includes powerful mechanism such as inheritance and embedded groups that simplify representations and improve the maintainability of implementations.
- **Programming language and Software Tool support:** there is widespread support for XML in many different tools and programming languages, making it relatively straightforward to implement import and export procedures. In particular the complex task of parsing and reassembling data is largely done by the standard XML parsers without any further programming effort being required.
- **Flexibility:** normally standards need to be able to evolve over time to support changes in the business requirements. XML is a self-describing format that can include optional elements and it is possible to have successive, but backwards compatible versions in concurrent use, distinguishable by different schema version attributes. This is valuable in a distributed implementation where there are many different systems using different versions of the system at the same time and that will choose to upgrade to new releases at different times.

XML Drawbacks

Use of XML schema technology also has some drawbacks.

- **Document size:** XML implementations are relatively verbose compared to flat file formats, requiring more bandwidth to transmit and greater computational resource to process. In mitigation, the semantically richness may also be used to condense content to avoid unnecessary repetition (as is found say in TAP price data) and data can in any case be compressed for transmission using normal zip techniques. Where size is critical additional optimisations can be made ("It is easier to make a correct model fast than a fast model correct").
- **Multiplicity of components:** the use of discrete, functionally orthogonal, reusable components, implemented with the various inheritance mechanism of XML (subtyping, embedding of groups etc) so as to be highly modular, means that the specification is quite fragmented and has a large number of small elements, making it harder to comprehend. This can be mitigated by the use of powerful editing and visualisation tools (such as XML SPY or Oxygen) which are able to reassemble the elements into views for users to inspect and edit schemas.
- **Complexity of Interpretation:** the semantic richness of NeTEx means that sometimes there is more than one way of potentially encoding a concept. When the difference is purely syntactic this does not matter in practice (for example, in-lining elements rather than declaring them separately and cross-referencing them), as the parser technology available for XML in any case does most of the work to reassemble the serialize objects from the document using the information provided by the schema, regardless of the actual encoding. A more serious problem is that in certain cases there may appear to be more than one plausible way of encoding data. Usually there will be a preferred "more correct" way, but deciding exactly the correct representation to use may involve expert consideration of subtle aspects of the model. For example, does a temporal condition apply to a whole SALES PACKAGE or just to one particular parameter such the PURCHASE WINDOW? Or should a condition be attached to a FARE PRODUCT or to a SALES PACKAGE based on the FARE PRODUCT? To mitigate this, users concerned to exchange a particular set of data (for example VDV timetables or TAP/TSI fares) typically specify a "profile" that spells out the preferred choice of elements, and examples (as cited in this document) are also valuable. A profile can be further described by a TYPE OF FRAME which is can indicate as "metadata" which elements should or should not be present.
- **Limited constraints:** although XML schemas support many important types of integrity constraint to check data, certain more complicated constraints cannot be expressed and must be checked programmatically in an incoming program. For example, for the sequence of stops making up the calling pattern of a vehicle journey, it might be required that there is only a departure time at the origin, only an arrival time at the destination; but at least one departure or arrival times for all the intermediate stops. NeTEx in fact makes most properties optional in the schema so that the same schema can be used for many different applications, in effect not even using all of XML's capabilities to express multiplicity of occurrence.

TOOLS & TECHNOLOGY

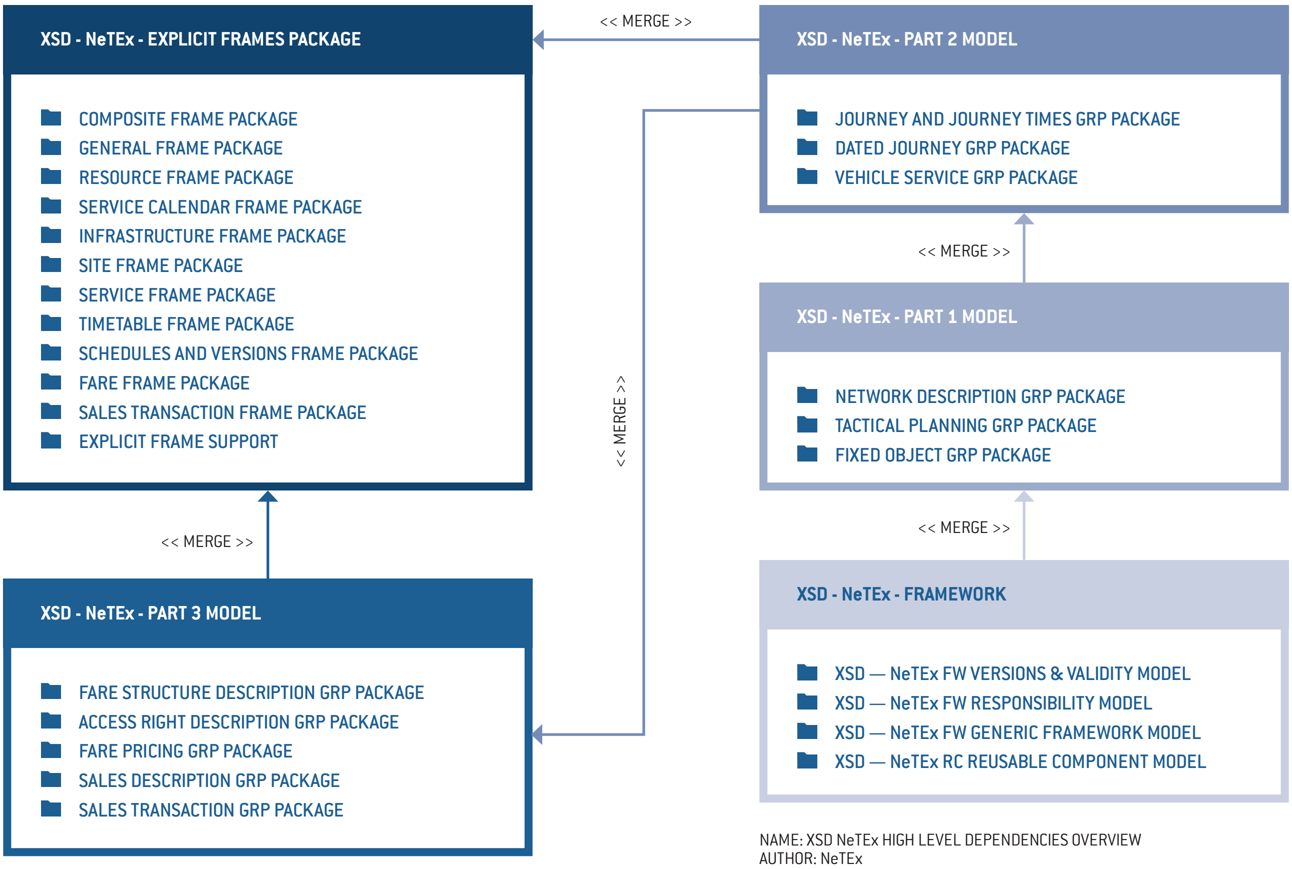
NeTEx's technology choices (XML, UML) represent mainstream technologies with widespread tool and platform support and a large pool of people with the necessary technology skills.

MODULARISATION OF THE FRAMEWORK

A large conceptual model and schema such as NeTEx (which has several hundred entities) needs to be modularised into smaller submodels in order to be manageable; both to understand it; to implement and test systems based on it; and to allow the separate evolution of unrelated subdomains over time. NeTEx encapsulates model elements into small packages of just a few related elements concerned with a particular function, each with their own self-contained diagrams and documentation. The core framework elements and common components are included in the base packages in Part1 and then referenced by the dependent packages that deliver the actual concrete functional of NeTEx.

Dependencies are linearized as far as possible, so that base packages can be used independently of other packages. To help users navigate the models the documentation includes both high level and low level dependency diagrams, and also uses a consistent set of colours for components from different functional domains.

Pkg XSD NeTEx High Level Dependencies Overview



NAME: XSD NeTEx HIGH LEVEL DEPENDENCIES OVERVIEW
AUTHOR: NeTEx
VERSION: 1.0
CREATED: 21/01/2010 12:24:53
UPDATED: 03/10/2014 18:57:48

Figure 2 – Dependency overview

ACKNOWLEDGEMENT

This White Paper has been produced thanks to the support of Fabrizio Arneodo (5T srl), Kasia Boureé (KBIC), Christophe Duquesne (Aurige), Nick Knowles (Steam Intellect), Andrej Tibaut (University of Maribor), Jan Tijmensen (InTraffic , Connekt).

FURTHER READING

THE NeTEx STANDARD

[N1] NeTEx-Part 1: *Public Transport Network Topology exchange format, CEN/TS 16614-1:2014*
[N2] NeTEx-Part 2: *Public Transport Scheduled Timetables exchange format, CEN/TS 16614-2:2014*
[N3] NeTEx-Part 3: *Fare Information exchange format, CEN/TS 16614-3:2014*

OTHER NeTEx WHITE PAPERS

[W1] NeTEx *Introduction* - White Paper
[W2] NeTEx *Getting Started* - White Paper
[W3] NeTEx *Framework* - White Paper
[W4] NeTEx *Reusable Components* - White Paper
[W5] NeTEx *Networks* - White Paper
[W6] NeTEx *Flexible Networks and Multimodality* - White Paper
[W7] NeTEx *Accessibility* - White Paper
[W8] NeTEx *Timetables* - White Paper
[W9] NeTEx *Fares* - White Paper

OTHER REFERENCES

[T1] *Public Transport Reference Data Model-Part 1: Common Concepts* (Transmodel), EN12896-1
[T2] *Public Transport Reference Data Model-Part 2: Public Transport Network* (Transmodel), EN12896-2
[T3] *Public Transport Reference Data Model-Part 3: Timing Information and Vehicle Scheduling* (Transmodel), EN12896-3
[T4] *Identification of Fixed Objects for Public Transport*, EN28701

FURTHER INFORMATION

NeTEx website: **www. netex-cen.eu**
Enquiries / email: **info@netex-cen.eu**