CEN/TC 278
ITS standardization

PUBLIC TRANSPORT
NETWORK TIMETABLE EXCHANGE
(NeTEx)

GETTING STARTED

CEN TC278/WG3/SG9 NeTEx PT001

Nicholas JS Knowles

OCTOBER 2015

TABLE OF CONTENTS

INTRODUCTION

The NeTEx (Network Timetable Exchange) is a CEN standard for exchanging public transport data, split into three parts [N1], [N2], [N3]. This white paper provides some guidance for getting started with using the NeTEx format, giving both some guidance on practical steps to start implementing a NeTEx interface and also a short review of available resources.

AUDIENCE

The paper is intended to convey a high level view sufficient for a technical manager to appreciate some of the issues involved in implementing NeTEx, omitting many 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.

THREE COMMON USE CASES

In practice, any new implementation of NeTEx is likely to be following one of three different use cases:

1 / Adding support for **export** in NeTEx format from an existing PT data system that already has similar data content (but does not use the exact NeTEx model).
- In this case the implementer is constrained by her existing repository model, and the task will be to map the data into the equivalent NeTEx elements in the output documents. It is likely that only certain elements and attributes will be populated, and the level of detail on the versioning of the exported will depend on what is supported in the originating system. A globally unique namespace or namespaces will need to be chosen for the identifiers and added to the exported content.
- The NeTEx specification includes appendices with mappings into a number of European National standards such as VDV452 (de), NEPTUNE (fr) and BISON (nl) and also GTFS see [G1].

2 / Adding support for **import** in NeTEx format into an existing PT data system that already has similar data content (but does not use the exact NeTEx model.
- The implementer is also constrained by the existing repository model, and the task will be to map the data from the equivalent NeTEx elements into the internal format. It may be necessary to extend the existing database model to handle additional elements or attributes present in NeTEx, especially if a lossless round trip exchange is envisaged (otherwise, additional information provided by NeTEx but not required may just be ignored). The importing system may use NeTEx version data present in the document to identify differences in data sets. The import may be limited to the specific namespaces and identifier sets supported by the existing system.

3 / Creating a **new system** based on parts of the Transmodel / NeTEx functional model, with **import and export** capabilities for the NeTEx XML schema. The implementer will be free to choose an internal repository model with a direct or straightforward mapping to the NeTEx model as shown in the Physical UML mode. Only those features that will be in the functional scope of the system need to be implemented – unsupported elements can be ignored.

In all three cases the implementer should be able to use automated tools to create an initial binding and NeTEx object model from the NeTEx XML schema (see below).

BASIC STEPS

This section sets out some basic considerations for using NeTEx.

AGREEING A PROFILE

NeTEx covers many types of transport data, and can be used in a many of different workflows and to support different levels of detail in the data exchanged. However a given group of stakeholders using NeTEx are likely to be using only a subset of the schema for an agreed purpose. Furthermore they may have more explicit requirements as to which namespaces and identifier sets are used and which elements and attributes are mandatory in the exchanged data. For example, users might just be exchanging stop data for the different modes of a country, or rail timetables for a region.

Using NeTEx therefore requires number of choices to be made. The set of choices may be described more formally as a **profile** – a set of decisions as to how NeTEx is to be used by a set of stakeholders who have agreed to exchange data in NeTEx format for a particular purpose.

PROFILES

A profile will typically cover:

- Which data elements are to be exchanged? For example stops, routes, journey patters, timetables, fare structures, etc. This depends on the functional scope of the systems and the use cases for exchanging data.

- Which types of version frame are to be used to exchange the data? In many cases, a single type of frame (e.g. TIMETABLE FRAME) has all the elements needed for the function (since the frames is chosen to group related elements). However sometimes elements from several different frames are needed; in this case a COMPOSITE FRAME is also used to group the frames into a coherent set with common validity conditions and compatible versions.

- Choose the namespaces and identifiers to use to identify instances of elements as unique. NeTEx allows you to declare a globally unique context for each identifier (see Whitepaper on framework [W4]), but it is up to the implementer to allocate specific domains for this purpose and to decide the semantics of the codes within the namespace. In some cases it may also be useful to include legacy identifiers as aliases on elements when they are exported, allowing two way exchange with legacy systems.

- Select the specific attributes of the elements which must, may, or must not be present (other than the attributes which are mandatory in the NeTEx schema and must always be present).

- Determine the granularity of elements within the frame. When outputting data elements within a given frame in some cases the implementer has a degree of freedom to the exact organization of elements within the frame (since they are walking a complex object model in order to serialize it, expressing some relationships by containment and others by explicit reference). For example, in some cases it is more convenient to nest related elements in-line within another containing element in other cases it is more convenient to declare the related elements separately and add a reference to them from the 'containing' element. This choice should make little difference to an importing system, 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, but in some cases it may materially affect the size of the resulting document.

- Determine the level of versioning to be exposed in the exchanged data. All NeTEx objects can hold detailed versioning attributes, and the data attributes may be exposed in the exchange format so that importing and exporting systems can use them to process changes efficiently. However many legacy systems do not support fine grained versioning and the fine grained version data may be omitted or ignored if it is not to be used.

- Select the protocol to use to exchange NeTEx XML documents. The content of a NeTEx document is independent of the method used to transfer it – and in fact the latter may vary according to the application. (e.g. SIRI [S1] http exchange of XML documents, asynchronous FTP of documents, SOAP, etc). A messaging protocol using SIRI allows arbitrary dynamic queries to be formulated – but typically only some query capabilities will actually be implemented. A group of stakeholders using NeTEx will need to decide which method of transmission i.e. protocol they wish to use, and also ensure that the bandwidth and processing capacity are appropriate to the data content and frequency of exchange that is envisaged.

DESCRIBING A PROFILE WITH A TYPE OF FRAME

A FRAME is a concept used to group related and consistent elements. NeTEx includes a mechanism, the TYPE OF FRAME, which allows version frames to be formally marked as conforming to a given profile.
A TYPE OF FRAME can list the elements and attributes that must be included in an instance of a version frame as well as various descriptive attributes and properties. SITE, SERVICE, TIMETABLE, FARE and other VERSION FRAMES can then reference the TYPE OF FRAME to indicate they are constrained to the profiles requirements. Additional validation programs may use the information from the TYPE OF FRAME to apply automatic validation tests over and above those enforced by the XML validators.

CHOOSING A VERSION FRAME

Data must be exchange in one or more version frames; the version frame groups together compatible data meeting a given set of selection criteria and validity conditions (e.g. "Berlin bus stops", "Todays timetable", "Paris Metro Fares Winter 2015"). There is a special version frame for each the of functional data, as well as several general purpose frames. A COMPOSITE frame is used to group other frames.
See table.

Table 1 NeTEX version Frame type

PART	NAME	DESCRIPTION
Part1 Framework	RESOURCE FRAME	Used to exchange common reference data such as operators, modes, facilities, day types, calendars, equipment, vehicle types, etc.
	GENERAL FRAME	Can be used to exchange any arbitrary user defined set of coherent elements.
	COMPOSITE FRAME	Used to group other frames for exchange as a single unit.
Part1 Functional	INFRASTRUCTURE FRAME	Used to exchange details of the road and rail elements making up the underlying network, along with restrictions on using them with specific vehicles,. Also locates different points dedicated to the vehicle and crew changeover.
	SERVICE FRAME	Used to exchange the basic description of a transport network; stops, lines and routes of a transport including stops and connection, along with the timing.
	SITE FRAME	Used to exchange information detailed places and sites such as stations, points of interest parking, including navigation paths and access restrictions.
Part 2 Functional	TIMETABLE FRAME	Used to exchange timetables, including journeys, linked journeys, planned interchanges, service facilities etc.
Part3 Functional	FARE FRAME	Used to exchange fare data, including fare structures, fare products, fare restrictions, sales packages, pricing parameters, prices.
	SALES TRANSACTION FRAME	Used to exchange descriptions of customers and their purchases.

IMPLEMENTING A NeTEx SYSTEM

IMPLEMENTATION TECHNOLOGY

NeTEx systems can be implemented using any technology (JAVA, C#, C++, Visual Basic, Python, PHP, Ruby, etc.) that can process XML documents; pragmatically it should be one that has an object orientated programming mode and tools to automatically create a binding to XML. NeTEx makes use of inheritance.

IMPLEMENTATION OF NeTEx INTERFACES

- To create an import interface, typically an implementer will first create a binding for the chosen language from the NeTEx XML Schema using a tool, this will create an object model that an XML parser will populate when it reads a NeTEx XML document. The implementer will then write a mapping of the elements of this model to his own internal representation, including any logic to resolve references to link to the appropriate version of object.
- To create an export interface, an implementer will write a program to process a query request (as specified by an XML document) to assemble a collection of elements that meet the selection criteria and validity conditions, and then walk the results to output the elements as XML elements wrapped in a specific version frame. Conditions as to the validity of the data may be attached to individual elements and to the whole frame. The output XML document should be validated against the NeTEx schema to check that it is correct.
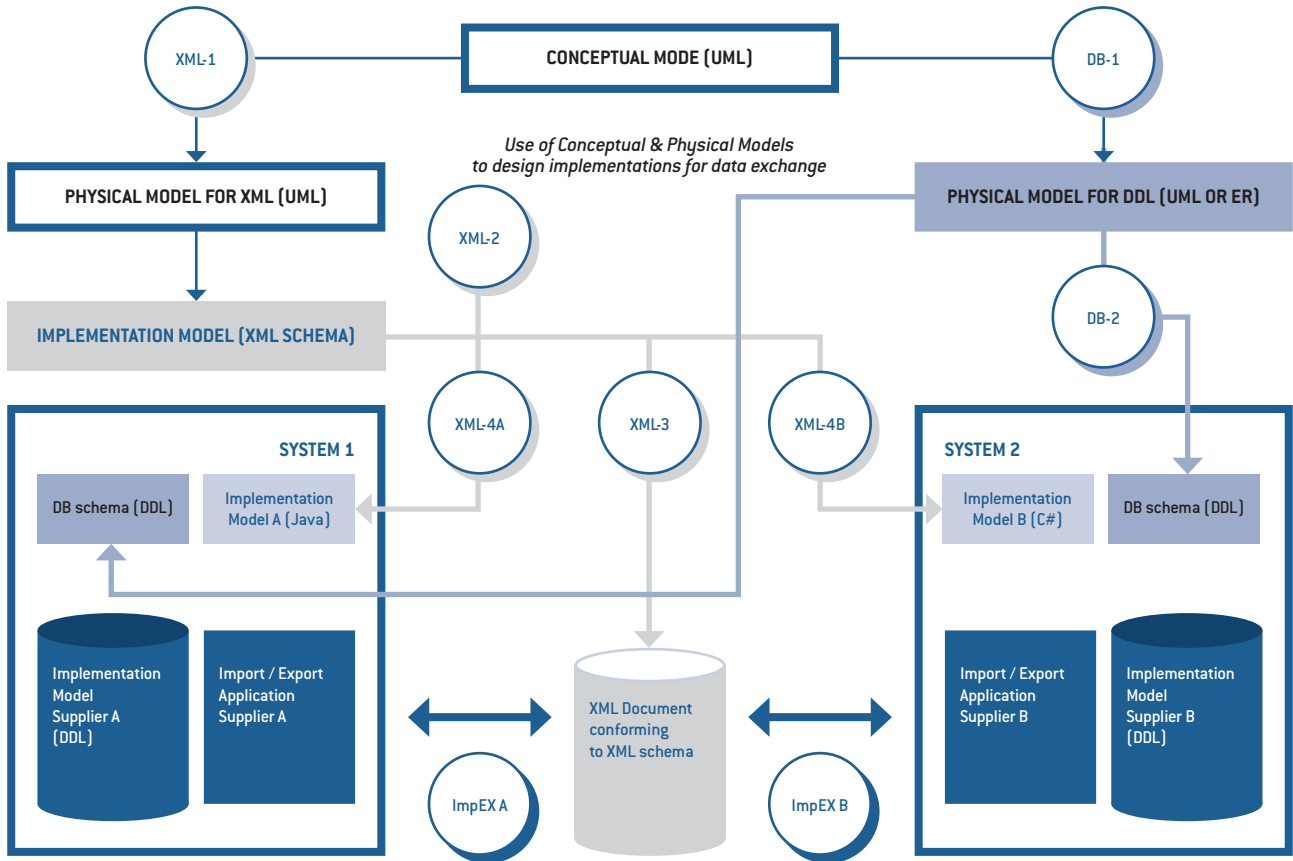
Figure 1 –Use of conceptual and physical models

VERSION LIFECYCLE FOR DATA ELEMENTS

Systems that allow the creation and maintenance of NeTEx data elements and that provide full support for the versioning of the elements (allowing the concurrent handling of one or more different versions of the same component within the same repository) should follow the uniform "Edit, Version, Release" cycle inherent in NeTEx versioning model.

- In order to edit a component a new "open" version is created; changes are then made to the open version; and then when these are complete, it is versioned.
- In order to version a subcomponent (for example a point within a journey pattern) an new open version of the parent component must also be created, and the new subcomponent version be released into it before the parent can be versioned.
- Once versioned, no further changes can be made; a new version of the component must be made first.
- Only versioned components can be exported.

VALIDATION & CONFORMANCE OF XML DOCUMENTS

One of the goals of NeTEx is to improve data quality and to automate the routine checking of documents for errors using the normal XML validators and parsers, without the need for explicit programming.

- The use of XML data types allows content strings to be checked for conformance to type (e.g. dates must be valid dates, periods must be valid durations, email addresses valid email addresses, etc.).
- The use of XML identity and reference constraints allows identifiers to be checked for uniqueness and references to be checked for completeness (i.e. referenced entities must be declared in a document in which there are referenced).

A NeTEx document must validate against the schema before it is deemed fit for import. Together these measures help catch many types of routine error, saving time and improving data quality. However not all integrity conditions of the model can be checked by the XML validator and an implementer may want to consider which additional tests of imported data should be performed. For example, "every QUAY and other area in a STOP PLACE should have at least one entrance" (otherwise it cannot be reached by a path); "the passing times on a vehicle journey should be monotonically increasing over the course of the journey", etc. In many cases it is possible to make an auto correction – but note that any change to the data should be reflected by creating new versions as well.

SCHEMA VERSIONS AND SCHEMA COMPATIBILITY

NeTEx is intended as a strategic interface; the schema and subschemas are versioned and it is likely there will be future versions from time to time. Successive schema versions will be **forwards compatible**, that is, documents that conform to an earlier version will also conform to a later version. They will also generally to be **backwards compatible**, that is documents that conform to the latest schema that do not use new features will also conform to earlier schema. Schemas include a version number that can be used to detect schema levels automatically.

RESOURCES

The following are resources for finding out more about NeTEx:

NeTEx CEN TECHNICAL SPECIFICATIONS

NeTEx is formally described as set of PDF documents that may be purchased from CEN national bodies such as VDV, AFNOR, BSI, NEN,

NeTEx Part1: Network description [N1]
NeTEx Part2: Timing information [N2]
NeTEx Part3: Fare description [N3]

The CEN Technical specifications are large and somewhat unwieldy documents prepared in accordance with CEN rules.

NeTEx WHITE PAPERS

The free NeTEx white papers provide short introductory papers on different aspects of NeTEx.
These fall into two groups:

General papers
NeTEx Introduction [W1]
NeTEx Getting Started [Present document]
NeTEx Design Methodology [W2]
NeTEx Framework [W3]
NeTEx Reusable Components [W4]

Specific functional areas
NeTEx Networks [W5]
NeTEx Flexible Networks [W6]
NeTEx Accessibility [W7]
NeTEx Timetables [W8]
NeTEx Fares [W9]

NeTEx UML MODELS

NeTEx provides models in UML notation of all the NeTEx entities and relationships as electronic artefacts. The models are modularised and provide a convenient way of browsing and studying the NeTEx models. The UML physical model includes many additional diagrams and views that are not present in the CEN Specification that help to understand the model and its dependencies.

The NeTEx UML model includes two distinct models:
NeTEx UML Conceptual Model
NeTEx UML Physical Model

Ancillary models showing a mapping to TAP/TSI B1, B2 and B3 specifications and to GTFS are also available. In order to view the UML models you need an appropriate tool, such as the Enterprise Architect viewer (www.sparxsystems. au) or an XMI enabled UML viewer. The NeTEx models are free and are available from www.netex-cen.eu (from the Download page).

NeTEx XML SCHEMA

The primary software resource from implementing a NeTEx interface is the NeTEx XML schema which is available at the www.netex-cen.eu.
Two main variants of the schema are available each providing a different protocol for embedding the same content model subschemas.

1 / Simple NeTEx documents exchange: (NeTEx _publication.xsd). A schema to use with NeTEx documents input or output by a system that are exchanged as files using FTP, email etc.

2 / NeTEx document exchange using SIRI http requests: (NeTEx_siri_SG.xsd). A schema that embeds the NeTEx elements in a sequence of http messages that define request/response and publish/subscribe interchanges for exchanging NeTEx data. Requests used NeTEx elements to specify the desired data. Responses are wrapped in version frames. The messages are specialisations of the SIRI framework.

The schema is modularized into smaller subschema, corresponding to the NeTEx modules and includes complex integrity constraints. The schemas are versioned to allow for concurrent support of successive versions. Use of a tool such as XML SPY or Oxygen to view the schemas is strongly recommended.

NeTEx WSDL BINDINGS

Three different SIRI WSDL bindings are available for NeTEx, with separate consumer (the system requesting data) and producer (the system supplying the data) messages.

	PRODUCER	*CONSUMER*
WSDL2	NeTEx_wsProducer-WSDL2.wsdl	NeTEx_wsConsumer-WSDL2.wsdl
Document	NeTEx_wsProducer-Document.wsdl	NeTEx_wsConsumer-Document.wsdl
RPC	NeTEx_wsProducer-Rpc.wsdl	NeTEx_wsConsumer-Rpc.wsdl

The bindings are also versioned to allow for concurrent support of successive versions.

TOOLS & TECHNOLOGY

General purpose XML tools such as parsers, validators and binding tools can be used with the NeTEx schema.

A number of different open source implementations and tools are under development including:
CHOUETTE available on https://github.com/afimb
(see http://www.chouette.mobi/developpeurs/formats-supportes/)

ACKNOWLEDGEMENT

FURTHER READING

THE NeTEx STANDARD

[N1]   NeTEx-Part1: *Public Transport Network Topology exchange format, CEN/TS 16614-1:2014*
[N2]   NeTEx-Part2: *Public Transport Scheduled Timetables exchange format, CEN/TS 16614-2:2014*
[N3]   NeTEx-Part3: *Fare Information exchange format, CEN/TS 16614-3:2014*

OTHER NeTEx WHITE PAPERS

[W1]   NeTEx *Introduction* - White Paper
[W2]   NeTEx *Design Methodology* - 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 *Timetable* - White Paper
[W9]   NeTEx *Fares* - White Paper

OTHER REFERENCES

[G1]   *General Transport Feed Specification* https://developers.google.com/transit/gtfs/
[S1]   Service Interface for Real Time Information (SIRI) prCEN/EN EN 00278181, prCEN/EN EN 15531

FURTHER INFORMATION

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