**Some Comments on the TfN Fare Project**

2010.12.14. NJSK

# Introduction`

These brief notes provide some comments on various aspects of using NeTEx for the TfN fare project based on my initial visit to Infinity Works in Leeds on 11th Feb 2020.

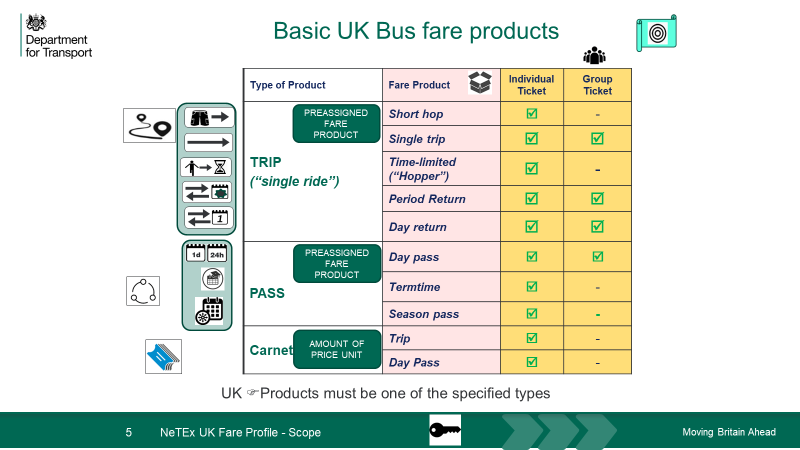
# Comments on Representing UK bus fare products

## Basic UK bus Fare Products

In considering the scoping of features for the TfN Alpha and Beta offerings, the summary tables of *Basic UK bus fare Tariff types* that appear in UK NeTEx profile documents and presentations on Farexchange.org.uk are useful for summarising the different aspects of UK bus fares - and hence the main choices a user needs to make to define a product.

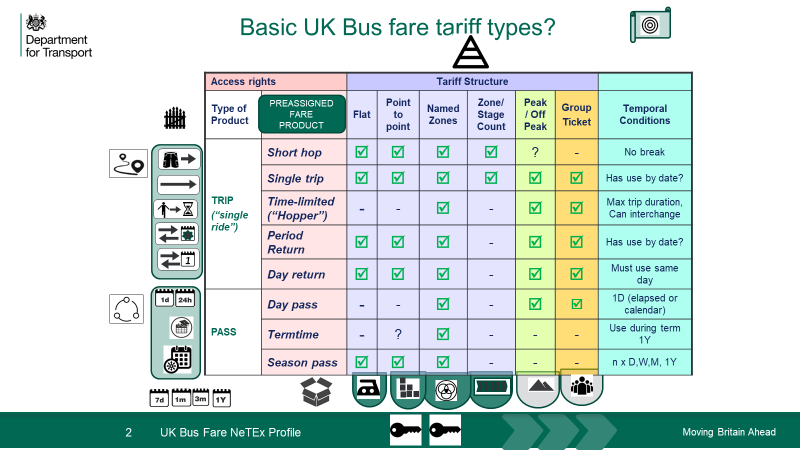
### UK Bus Product Types

* The following table makes a fundament classification of FARE PRODUCTs as being either for a “Trip” (i.e. rights to make a single journey) or a ”Pass” (i.e. rights to make repeated journeys over a period).
* A third type of product the Carnet, provides bundles of trips or day passes at a discount.



### *Basic UK bus fare Tariff types*

The list of products can be further elaborated with columns showing the possible tariff structures upon which the pricing of the product is based, as well as other key features used to characterise products for marketing purposes – in particular when it may be used (Peak/Off peak), and whether it is an individual or group ticket.



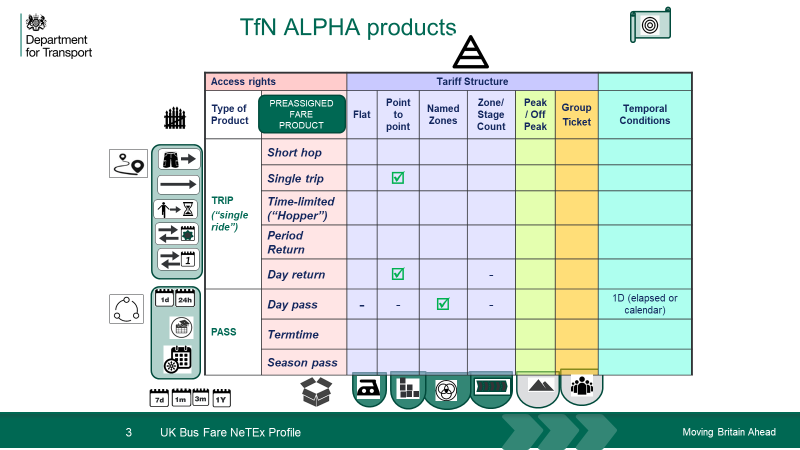
Notes:

* The concept of a FARE PRODUCT is distinct from that of a TARIFF structure, which characterises which elements of the transport network may be accessed using the product, and how they are priced, (i.e. by zone, by O/D pair, flat, etc.) Several distinct tariff structures are found in practice, each represented by very different abstractions (e.g. DISTANCE MATRIX ELEMENTS, vs ZONEs, GEOGRAPHICAL INTERVALs, etc.). The same TARIFF structure may be used in different products.
* Not all combinations of FARE PRODUCT and possible TARIFF structure are found in practice (e.g. Day Passes are never point-to-point) – especially in basic bus fares. Certain TARIFF structures can be used for both Trip and Pass products (for example, it is possible to have a single trip over one or more zones, as well as a pass for one or more zones).
* Pass products additionally have one or more TIME INTERVAL associated with them in order specify the allowed durations that may be purchased (1 day, week, month, etc).
* Stage fare, zonal and flat fare structures can also alternatively be expressed as point-to-point fares (i.e. be statically precomputed to populate a p2p table) – doing so is usually considerably more verbose and repetitious than using the true tariff structure, but is simpler for the user to understand when pricing a trip since they can be presented with the completed prices for the relevant O/D pairs.
* Various additional parameters can be attached to products to characterize them further. – e.g. USER PROFILEs for eligibility of user type, GROUP TICKETs for multiple users on the same ticket, FARE DEMAND FACTORs to specify the allowed periods of travel as peak/off-peak, etc
* A FARE PRODUCT is thus an arbitrary set of rights that are bundled together and marketed as a named product that the user can select. One cannot be overly prescriptive about how this bundling is done as it depends on the specific business practices of a given operator; for example one operator might choose to market separate named offerings for students and for adults as different “products”, while another might have a single product with just different prices for different types of users. That said, in practice only a few combinations are commonly found for small bus operators, and by choosing and naming only certain combinations and only certain additional parameters we can greatly simplify what is presented to the user when inputting data – as is being done by the TfN NeTEx fares project
* The data that is hardest to capture in the UI is the TARIFF structure itself; since the underlying network elements (e.g. stops, zones etc) must be referenced and grouped. Typically to collect the inputs for each type of Tariff structure, one will need a different path through the UI, so it will be helpful

## TfN Alpha product types

The TfN Alpha is in effect supporting just the following three choices of product and tariff structures out of the wider set given above, representing a judicious choice of fundamental product and tariff structures for an initial prototype. (i.e. a simple Trip, a composite Trip and a Pass)

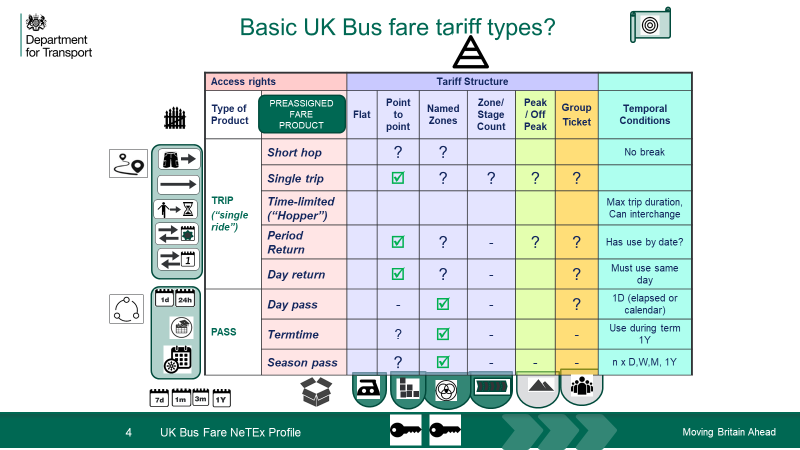
A UK NeTEx Fare Profile document may contain the fares at different levels of granularity - (e.g. fares just for a single LINE, or for a whole (or part) NETWORK (thus it would be possible to have the fares for many LINES in a single document). Although it wasn’t explicitly stated, my understanding is that the TfN project will be limiting point-to-point fares to a single LINE per document. Zonal fares will be for the whole operator network covered by a given product.



## TfN Beta product types

The TfN Beta will add further FARE PRODUCT types as well as further parameters to extend their features. The exact scope of these has still to be decided, but might encompass the following product types and fare structures. (In addition, Carnet products might be considered.)

Each FARE PRODUCT TYPE and TARIFF structure requires specific additional elements to be populated. While for the Alpha, the TARIFF structure is unique for the product type (Trips are P2P, Passes are zonal) so the Tariff structure does not need to be further specified by the user, in the Beta it may need to be selected in some cases (e.g. is a trip point to point or for a named zone), as shown below.



## Stage Fares

One point of discussion is the nature of support for Stage Count Fares, that is, a fare structure where the price merely depends *on the number of stages consumed, and not on which specific stage stops are passed*. Traditionally this tariff structure has been widely used to represent bus fares as it provides a compact encoding that requires just a few prices to describe a large number of potential fares.

### P2P representation of Stage Fares

The approach being taken in the TfN Alpha is in effect to resolve these Stage Count tariff structures into a specific point-to-point (or rather zone-to-zone) fare representation, with a filled out “fare triangle” as is classically presented to the user when looking up the price of a fare.

That is, in a zone-to-zone representation such that;

1. A FARE ZONE is defined for each fare stage, with all the relevant stops in the zone being added to it in order.
2. A DISTANCE MATRIX ELEMENT is defined for each permitted zone-to-zone transition.
3. A FARE PRICE is specified for each DISTANCE MATRIX ELEMENT.

While this is a viable approach for small fare tables and is relatively simple to understand, it is verbose (and data redundant) for representing the fares of large networks -- and also fails to capture the conceptual justification for the pricing as a count of stages. The number of matrix element and prices increases N2 on the number of the zones (but is usually symmetric so actually (N2 -N)/2), further multiplied by the number of different user types, sales packages, etc. that have different prices. To make the approach manageable, the TfN project is proposing limiting the number of zones (to less than say, 10-20?)

A journey planner or fares engine using the fare data must hold the whole matrix of P2P price data for each type of user and fare product in order to determine the price for a given trip (since it does not know the cost per stage).

### Interval representation of Stage Fares

In contrast, a true fare stage representation requires only a few prices and its size is linear on the number of fare stages. It would be represented in NeTEx as follows:

1. A GEOGRAPHICAL INTERVAL is specified for each multiple of unit stages that has a price (e.g. 1 stage = £1.00, 2 stages = £1.80, 3 stages = £2.50 etc.).
2. A FARE PRICE is specified for each GEOGRAPHICAL INTERVAL.
3. The stages of the route are marked by either
   1. By flagging the POINTs IN JOURNEY PATTERN of the route as fare stages. (using

FARE POINTs in PATTERN – they can be grouped with a SERIES CONSTRAINT) or

* 1. By using a FARE ZONE for each stage, as in the Zone-to-zone representation above.

A journey planner or fares engine using the fare data need hold the pricing intervals – and note the stages – it can then compute the number of stages passed and look up the price.

### Comment

It might actually be easier to support a true fare stage representation in a GDS style UI, as it would potentially be simpler for users to populate.

It would involve two steps

1. What are the price bands? the user could be offered a list of intervals (1,2,3,4 stages etc) and fill in the prices against each band that is used
2. What are the fare stages? The user could provide a list of stops or be presented with a list of stops taken from a TXC document and then flag the stages (or upload as a very simple spreadsheet with both stops and stages in order).

It would be simple to generate a fare triangle from the above

For example: a) GEOGRAPHICAL INTERVAL & PRICE input.

|  |  |
| --- | --- |
| Unit Stage | Fare Price |
| 1 | £1.00 |
| 2 | £1.80 |
| 3 | £2.50 |
| 4 | £3.00 |
| 5 |  |
| 6 |  |

For example: b) Stop & stage input.

|  |  |  |  |
| --- | --- | --- | --- |
| Seq | NaPTAN Code | Stop Name | Fare Stage? |
| 1 | **blahblah** | **Alpha** | Y |
| 2 | **blahblah** | **Bravo** |  |
| 3 | **blahblah** | **Charlie** |  |
| 4 | **blahblah** | **Delta** | Y |
| 5 | **blahblah** | **Echo** |  |
| 6 | **blahblah** | **Foxtrot** |  |
| 7 | **blahblah** | **Golf** |  |
| 8 | **blahblah** | **Hotel** | y |
| 9 | **blahblah** | **India** |  |
| 10 | **blahblah** | **Juliet** | y |
| 11 | **blahblah** | **Kilo** |  |
| 12 | **blahblah** | **Lima** |  |
| 13 | **blahblah** | **Mike** |  |
| 14 | **blahblah** | **November** |  |
| 15 | **blahblah** | **Oscar** | y |
| 16 | **blahblah** | **Papa** |  |
| 17 | **blahblah** | **Quebec** |  |
| 18 | **blahblah** | **Romeo** |  |
| 19 | **blahblah** | **Sierra** |  |
| 20 | **blahblah** | **Tango** | y |
| 21 | **blahblah** | **Uniform** |  |
| 22 | **blahblah** | **Victor** |  |
| 23 | **blahblah** | **Whiskey** |  |
| 24 | **blahblah** | **X-ray** | y |
| 25 | **blahblah** | **Yankee** |  |
| 26 | **blahblah** | **Zulu** | Y |

For example: c) Fare Triangle created automatically from above data.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Stage Name | **Alpha**? | **Delta** | **Hotel** | **Juliet** | **Oscar** | **Tango** | **X-ray** |
| **Delta** | £1.00 |  |  |  |  |  |  |
| **Hotel** | £1.80 | £1.00 |  |  |  |  |  |
| **Juliet** | £2.50 | £1.80 | £1.00` |  |  |  |  |
| **Oscar** | £3.00 | £2.50 | £1.80 | £1.00` |  |  |  |
| **Tango** | £3.00 | £3.00 | £2.50 | £1.80 | £1.00 |  |  |
| **X-ray** | £3.00 | £3.00 | £3.00 | £2.50 | £1.80 | £1.00 |  |
| **Zulu** | £3.00 | £3.00 | £3.00 | £3.00 | £2.50 | £1.80 | £1.00 |

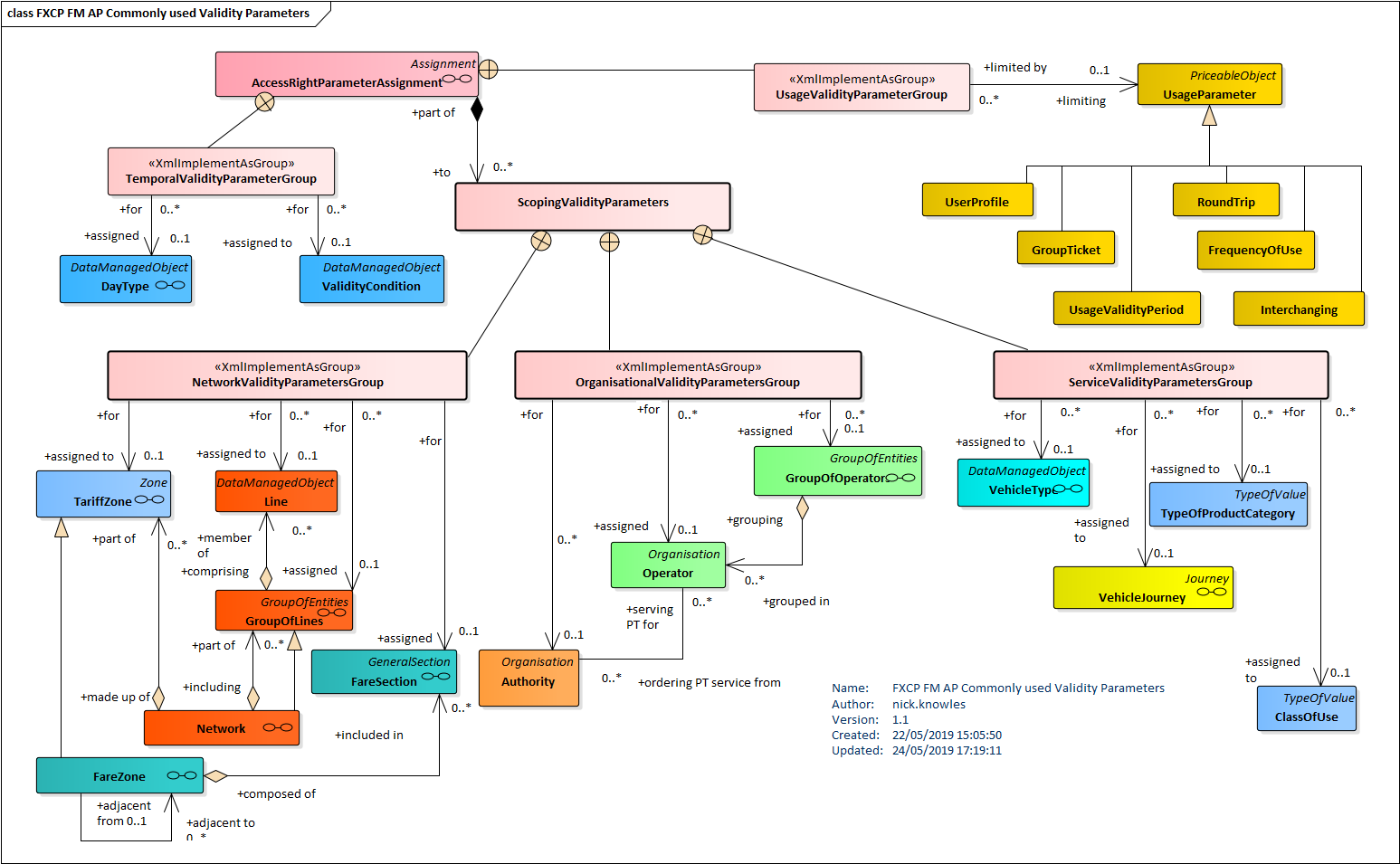
Another way of looking at this is that in either representation one must gather a list of fare stages, but one could then ask an additional question *“Are the fare prices based on the number of fare stages used?”.* If they are, one could ask for the GEOGRAPHIC Intervals and prices and populate the table automatically. If they are not then the fare triangle prices need to be populated by hand.

In any case, if a zone-to-zone representation of what is an underlying stage fares is used, it would be advisable to choose terminology in the UI such that it is not confused with a true fare stage representation

## Usage Parameters in the TfN Beta

NeTEx has a considerable number of USAGE PARAMETERs for specifying the many various conditions as to purchase conditions, travel conditions and after sales rights associated with a given FARE PRODUCT and/ or SALES OFFER PACKAGE. These can largely be ignored for an Alpha but a small subset of them will be relevant for the TfN Beta. They can be presented to a data entry user as further simple questions in a simple GDS UI (in most cases with default to use if no value is input). For example

* Travel Conditions
  + Does the ticket allow passengers to make a transfer between vehicle journeys? (INTERCHANGING)
  + For a pass, when does validity start: time of purchase or start of day (USAGE VALIDITY PERIOD Usage Trigger)?
  + For a pass, when does validity end: 24 hours etc or end of travel day (USAGE VALIDITY PERIOD – UsageEnd)?
* Purchase Conditions
  + What type of user may buy the ticket? (adult, child, senior, student, concession, etc). (USER PROFILE)
  + Specific characteristics of a user type - What age is a child, senior, etc? (USER PROFILE MinimumAge, MaximumAge)
  + How many people may travel on the ticket – and of what type? (GROUP TICKET)
* After Sales condition
  + Is the ticket transferrable to another person? (TRANSFERABILITY)
  + Is the ticket refundable? (REFUNDABLE)
  + For how long does the ticket remain valid to use after purchase? (USAGE VALIDITY PERIOD: StandardDuration)
  + For a pass, what is the earliest / latest by when the ticket can be purchased? (PURCHASE WINDOW)



For a Pass

## Sales Offer Packages

The NeTEx SALES OFFER PACKAGE specifies adds further information to a product concerning the materialisation and distribution of a FARE PRODUCT as a purchasable offering. In particular, the SALES OFFER PACKAGE states the TYPE OF TRAVEL DOCUMENT (i.e. media such as paper, mobile app etc.) used. Further package elements specify the distribution channels, payment methods etc that are important to know if a user wants to purchase a fare.

SALES OFFER PACKAGEs can also be used to describe more complex offerings, for example multiple products can be bundled as a single package. However, the use of packages for the TfN bus fare project will generally only need to be simple – one SALE OFFER PACKAGE per product and media type.

As suggested by the UK Profile examples, it will probably be possible to define a fixed set of TYPES of TRAVEL DOCUMENT, DISTRIBUTION CHANNELs, and FULFILMENT METHODs to use in all the TfN offerings. For example

* TYPES of TRAVEL DOCUMENT: *Paper, smartcard, mobile app.*
* DISTRIBUTION CHANNEL, *At Stop machine, Onboard Conductor, Mobile internet, Ticket Office, Online Order* (For passes and carnets).
* METHOD OF PAYMENT: *cash, card, mobile app*.
* FULFILMENT METHOD: *From Machine, from driver/conductor, online, post*

Definitions for these fixed instances can be included as boiler plate VERSION FRAMEs included in all TfN NeTEx documents generated and be referenced by the operator specific contents

## Difference between a single and a return fare

The NeTEx fare model is intended to include concepts that allow fare products to be related to the validation and control processes that check the user has a valid ticket.

Thus, a return fare product isn’t simply a single ticket flagged as being usable in both directions, rather it should have a separate ACCESS RIGHT IN PRODUCT for travel in each direction, each referring a VALIDABLE ELEMENT, since the order of use is a material part of the product definition. Usually an outward leg must be used before a return leg, and each leg must be separately validated.

# Comments on Encoding data in NeTEx

Out goal will be to encode the data so that it can (a) be validated easily and (b) is relatively easy to import.

## Some guidelines for grouping and coding data

The NeTEx XML schema allows considerable flexibility as to how data elements are grouped within an XML document. The order of specific VERSION FRAMEs within a document is arbitrary. Within a frame, certain elements may either be nested in context, or simply included as direct children of the frame. For example, VALIDABLE ELEMENTs may either be nested within the FARE PRODUCT they apply to, or (if they apply to multiple products) be directly embedded in a FARE FRAME and referenced by each product that uses them. Similarly, FARE STRUCTURE ELEMENTs and DISTANCE MATRIX ELEMENTS may be nested in a TARIFF, or declared directly.

🡺 In general, it makes for more modular, saleable and easier to follow XML if elements are nested in context, and this is typically done in most of the NeTEx fare examples that are provided, and some rules are set out in the UK Fare Profile specification. Such an approach is recommended for TFN.

In particular a TARIFF and FARE STRUCTURE ELEMENTS should be used to group the elements defining the access rights and USAGE PARAMETERS. In principle, complex conditions can be built up for FARE STRUCTURE ELEMENTs that logically AND and OR the various data elements needed to define the rules governing the fares. In practice this can (and should) be kept as simple as possible by segregating different concerns into different FARE STRUCTURE ELEMENTs. For example,

1. Place all the elements describing the access rights to the network of the Tariff structure (either DISTANCE MATRIX ELEMENTS, GEOGRAPHICAL INTERVALS, FARE ZONES, etc) in a separate “access rights” FARE STRUCTURE ELEMENT
2. Place the USAGE PARAMETERs governing the eligibility (i.e. alternative USER PROFILEs or GROUP TICKETs specifying who may buy the product) in a separate “eligibility” FARE STRUCTURE ELEMENT.
3. Place the TIME INTERVALS governing the alternative durations for which a pass may be purchased in a separate “pass duration” FARE STRUCTURE ELEMENT (e.g. 1D. 1W, 1M etc).
4. Place the USAGE PARAMETERS governing the travel conditions (e.g. INTERCHANGING, FREQUENCY OF USE, TRANSFERABILITY, USAGE VALIDITY PERIOD) in a separate “travel\_condition” FARE STRUCTURE ELEMENT – for a basic product usually these are fixed and logically ANDed together
5. Place the USAGE PARAMETERS governing the commercial conditions (e.g. PURCHASE WINDOW, REFUNDING, REPLACEING, etc) in a separate “sales\_condition” FARE STRUCTURE ELEMENT – for a basic product usually these are fixed and anded together

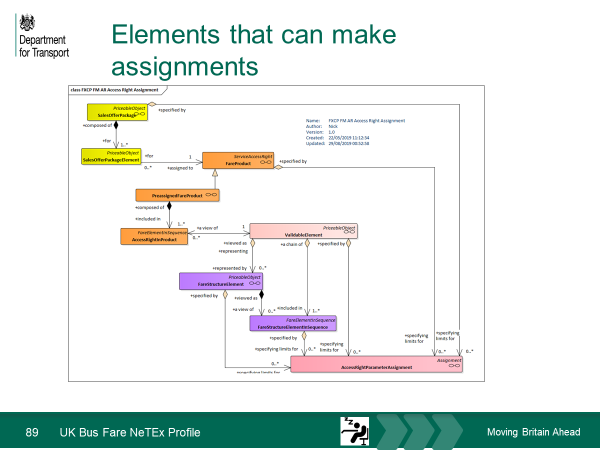
Thus, each FARE STRUCTURE ELEMENT contains just a list of the alternatives (if any) that may be selected for a given “dimension” of the fare.

This discussed further in the UK Fare NeTEx profile part 3. in section 12.2.

Parameter assignment precedence

ACCESS RIGHT PARAMETER ASSIGNMENTs are used to specify the access rights and rules of a fare product by associating network elements TARIFF STRUCTURE and USAGE PARAMETERs with different levels of fare component: (a) FARE STRUCTURE ELEMENTS (b) VALIDABLE ELEMENTs (c) FARE PRODUCTS (d) SALES OFFER PACKAGES.

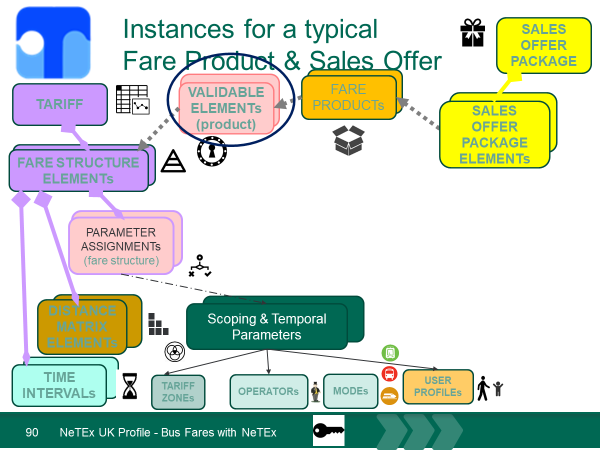
### Hierarchy of assignment

v

### Common assignments

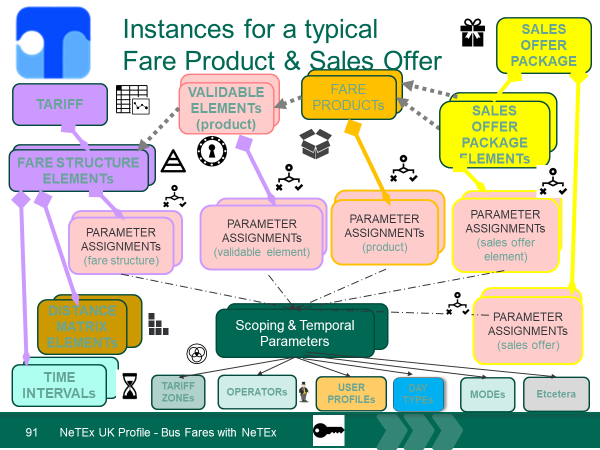
In general, it is preferable (i.e. most general and simple) to make assignments at the lowest level - that is , using FARE STRUCTURE ELEMENTs. The following diagram illustrates the most common assignments needed to define products (all of which are relevant for the TfN project); a fare may apply to specific lines, zones, operators and or modes. Note that DISTANCE MATRIX ELEMENTS and TIME INTERVALS can be referenced directly and do not need VALIDTY PARAMETER ASSIGNMENTs.

Most simple products can be defined using just FARE STRUCTURE ELEMENTs and be associated with a product indirectly through VALIDABLE ELEMENT.



### Product and offer level assignment

Note however that in some cases it is useful to specify additional conditions that apply to a specific FARE PRODUCT or SALES OFFER PACKAGE (For example a mobile app SALES OFFER PACKAGE might be available only to students at a given institution), so in practice assignments can be made at any level. (Probably not needed for the TfN project).



### Fare Prices

The UK Fare profile recommends that FARE PRICES are kept in a separate FARE FRAME from the elements that they price, if necessary nested in FARE TABLEs for efficiency. In the simplest cases this is slightly more verbose than in-lining the prices with the elements that they price, but is conceptually cleaner - and actually more efficient if there are multiple alternative prices, say for different user types, media etc. FARE TABLES can also be used to capture presentation aspects (Column headings, footnotes etc) for rendering fares for display to the user, for example as a fare triangle.

## UK Fare Profile

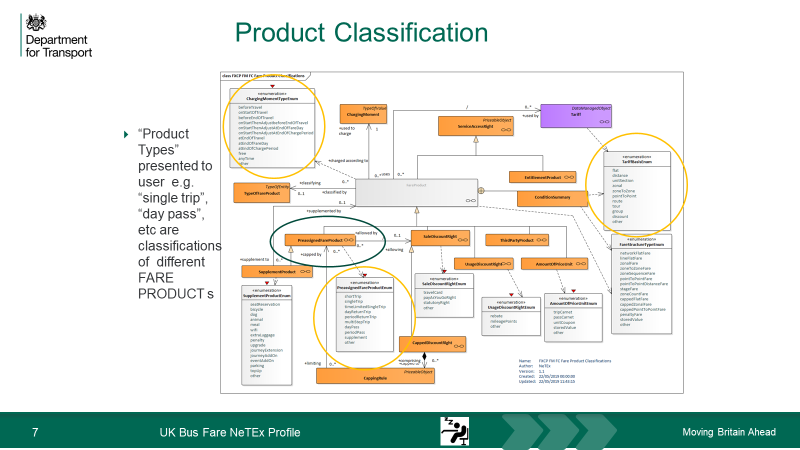
The task of importing NETEX data in downstream applications that want to use the fare data is greatly simplified if it can be assumed by the importing program that the data is conformant to a specified profile, such as the UK Fare Profile. This limits the data contents that are to be processed to just those elements permitted by the profile, and guarantees they are organised and grouped as required by the profile -- and use the CODESPACEs and identifiers mandated by the profile for specific data sets.

The profile should be declared by a TYPE OF FRAME element on the VERSION FRAME. See Section 12 in the UK Profile Part3

## Classification of products, tariff structures, etc

NeTEx includes attributes to classify various elements systematically. These classifications can make it easier for an importing program to interpret the data for a product correctly. In some cases, classification is made using fixed enumerations of values, in others using a TYPE OF (ENTITY) element (for example TYPE OF FARE PRODUCT) allowing arbitrary user defined classification values to added. In other cases, both may be used.

The available enumerated classifications of different FARE PRODUCT types and TARIFF TYPES are shown below. A TYPE OF FARE PRODUCT ELEMENT could be used to add further arbitrary values.



The following example, taken from the UK Fare Profile specification shows the use of some FARE PRODUCT and TARIFF structure classification elements.

<**PreassignedFareProduct** version="1.0" id="myb:Trip@single">

<Name> Single Ticket</Name>

**<ChargingMomentRef versionRef="fxc:v1.0" ref="fxc:prepayment"/>**

**<ChargingMomentType>beforeTravel</ChargingMomentType>**

**<TypeOfFareProductRef versionRef="fxc:v1.0" ref="fxc:standard\_product@trip@single"/>**

<OperatorRef version="1.0" ref="noc:MYBUS">137122</OperatorRef>

<ConditionSummary>

<FareStructureType>pointToPointFare</FareStructureType>

<TariffBasis>pointToPoint</TariffBasis>

</ConditionSummary>

<!--==== VALIDABLE ELEMENTs ==== -->

<validableElements>

<**ValidableElement** version="1.0" id="myb:Trip@single@travel">

<Name>Single ride</Name>

<fareStructureElements>

<FareStructureElementRef version="1.0" ref="myb:Tariff@single@access"/>

<FareStructureElementRef version="1.0" ref="myb:Tariff@single@eligibility"/>

<FareStructureElementRef version="1.0"

ref="myb:Tariff@single@conditions\_of\_travel"/>

</fareStructureElements>

</**ValidableElement**>

</validableElements>

<!--==== ACCESS RIGHTs ===== -->

<accessRightsInProduct>

<AccessRightInProduct version="1.0" id="myb:Trip@single" order="1">

<ValidableElementRef version="1.0" ref="myb:Trip@single@travel"/>

</AccessRightInProduct>

</accessRightsInProduct>

<ProductType>singleTrip</ProductType>

</**PreassignedFareProduct**>

## Use of Metadata

### Names of documents and Frames

The UK profile sets out some rules for naming conformant XML documents (and the version frames within them) after the operator and line they describe so that they can be easily identified. It also requires these values to be included in the document header as metadata so that that is easy for an automated process to determine the intended contents of a document.

See section 12.3 and 12.4 in the UK Fare profile.

The NeTEx specification allows the fin egrained specification of data sources

## Validation & Data quality

Data quality is critical for success and it is worth mentioning certain aspect of NeTEx that are relevant for promoting it.

### Guidelines for referential integrity

The NeTEx XML includes an extensive set of integrity constraints that can be used to automatically check the correctness of a document using a standard XML validator. As is explained in the UK Profile document, NeTEx exploits a feature of the XML validation that makes it possible to decide whether validation is or isn’t applied to a particular reference (since different NeTEx profile may require references to be internal or external)

Validation should be used wherever possible on internal references so as to prove the correctness of the generated documents. Thus, all elements should be declared with a ***version*** number (dummy or otherwise) and internal references to the elements within the document should include the ***version*** number so that the reference is checked by a validator. References to external entities may use a ***versionRef*** attribute top record a version number that will not be checked by the XML Validator (But might be checked by an impoort8ing program that is assembling different data sets)

### Further Validation

Schema driven XML Validation by an XML validator is limited to validation of data types, the order and presence of tags and integrity checking of internal references. Some further validation is possible to apply the domain specific rules of a profile. For example

* To check that external codes, e.g. for NaPTAN and NOC codes.
* To check Values for prices are progressive by distance (or at least not regressive)

See UK Fare Profile 13.1 Summary of data quality rules for FXCP for a list of possible rules