Differences between standards

Webinar

12 April 2021

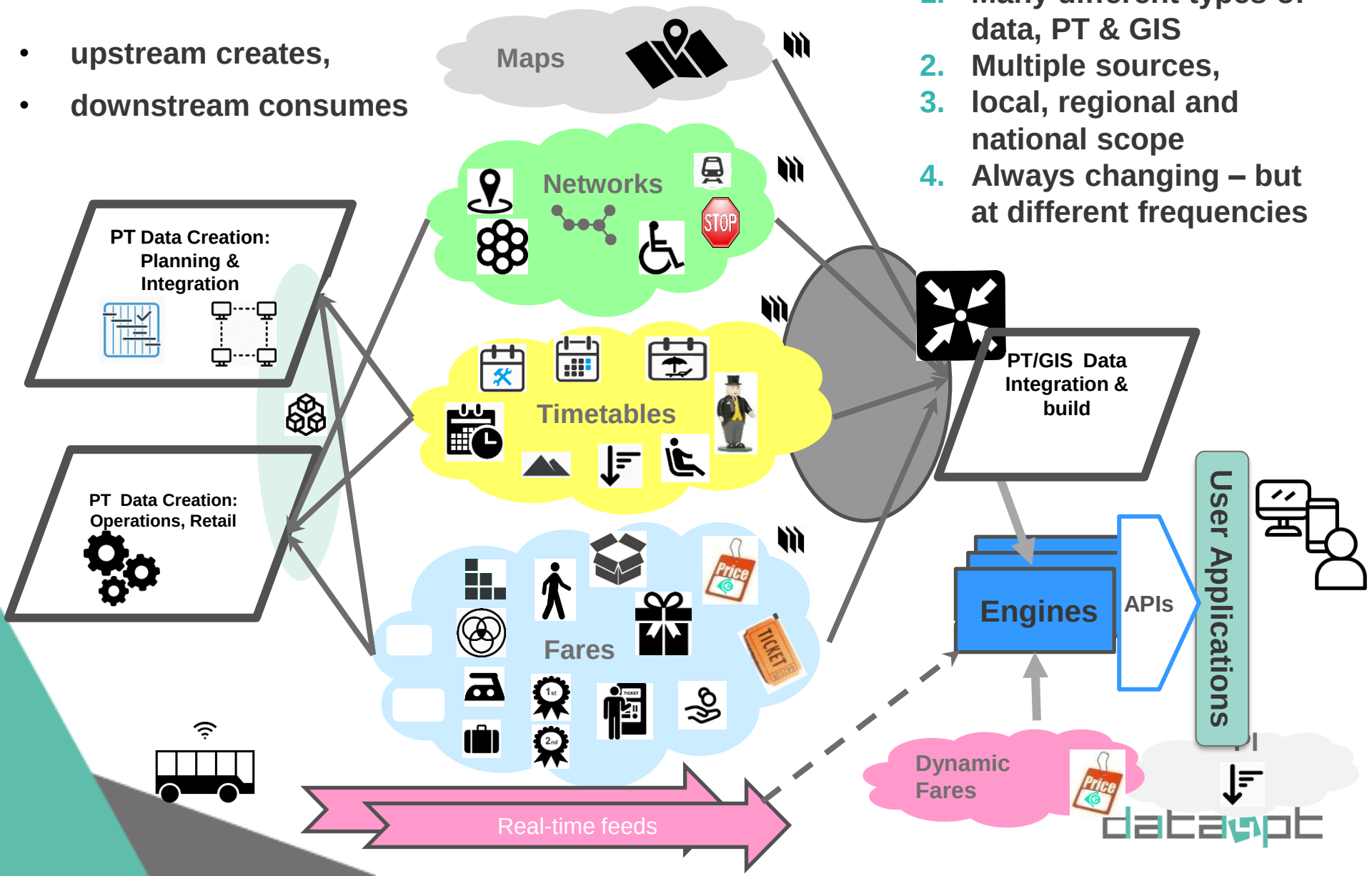
Nicholas Knowles, DATA4PT expert

Data4PT has received funding from the European Union's DG for Mobility and Transport under grant agreement No MOVE/B4/SUB/2019-104/CEF/PSA/SI2.821136

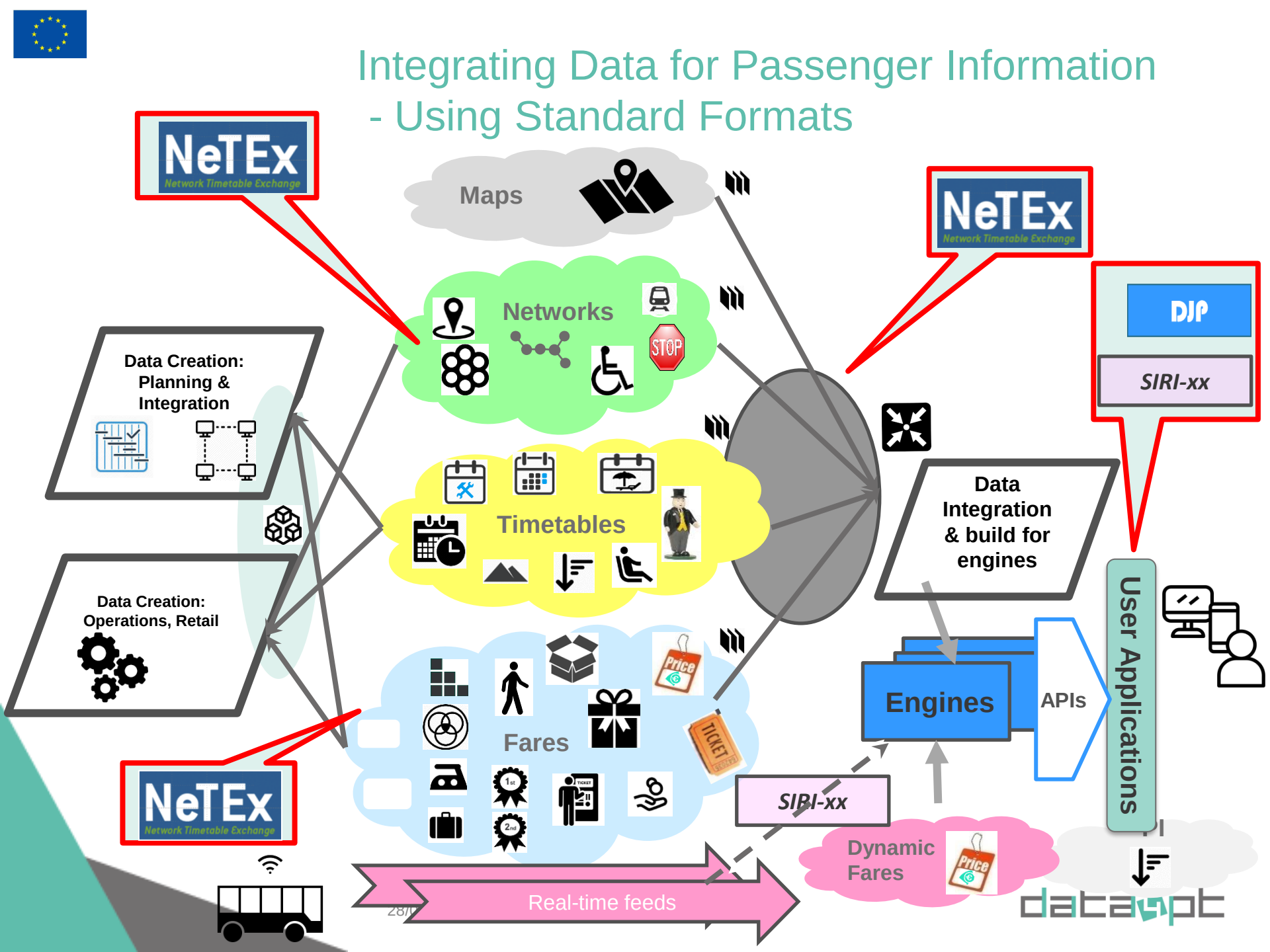
Integrating Data for Passenger Information - The data supply chain

- upstream creates,
- downstream consumes

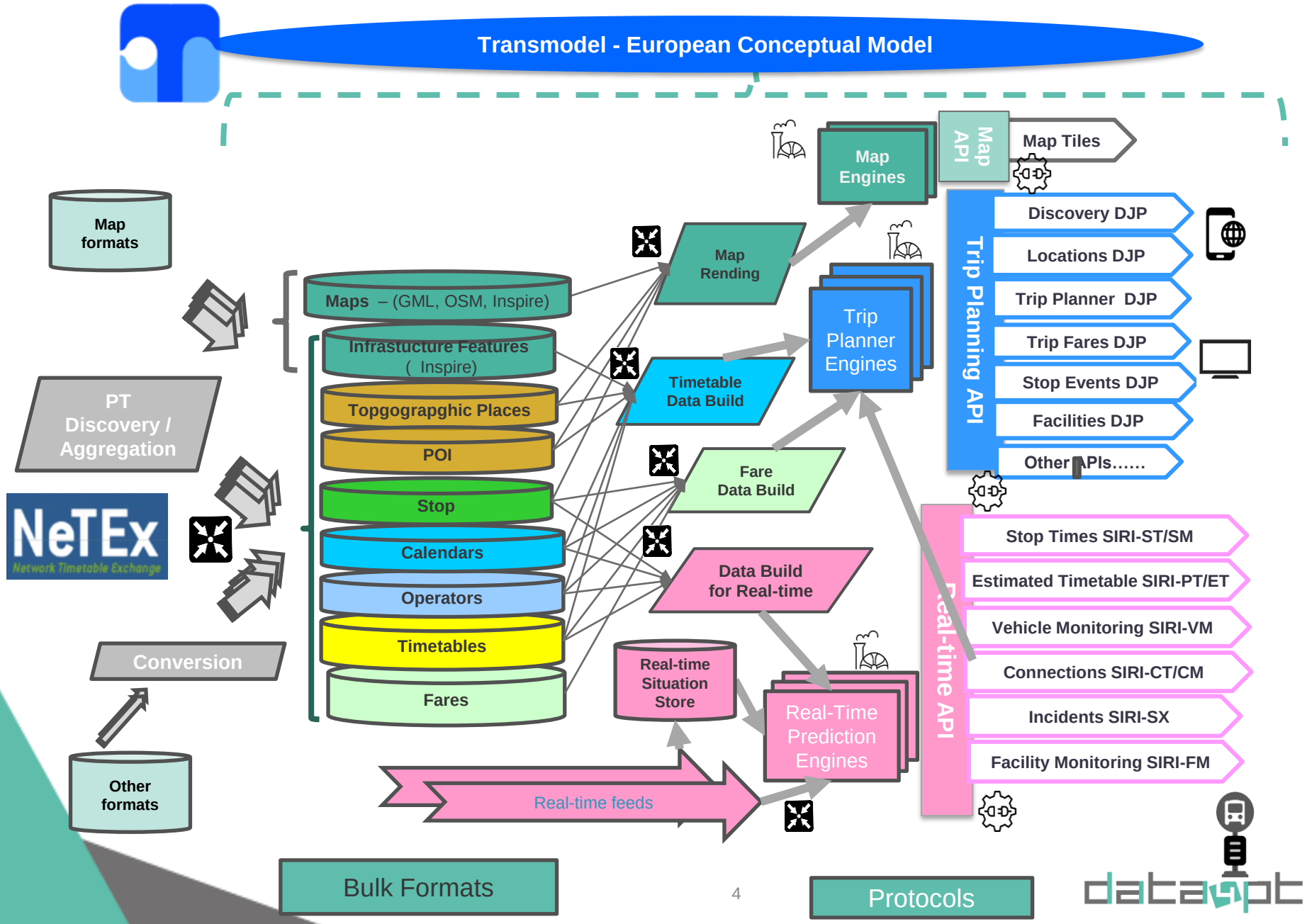
1. Many different types of data, PT & GIS
2. Multiple sources,
3. local, regional and national scope
4. Always changing – but at different frequencies



Integrating Data for Passenger Information - Using Standard Formats



Integrating data from different sources to create PI services



Using model driven design to engineer exchange formats

The *Conceptual Model* is implementation independent

"Joined up" architecture
Addresses multiple use cases

A *Physical Model* maps to each target implementations

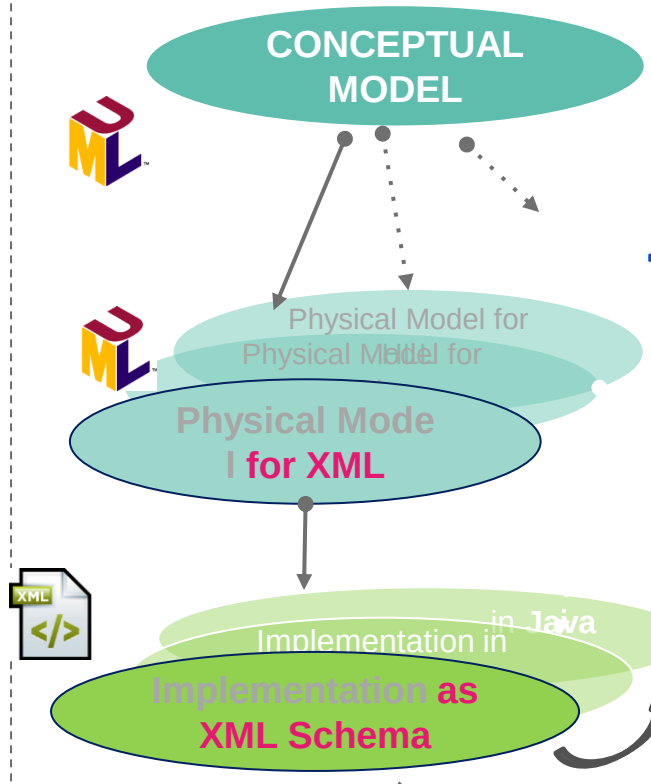
Selective scope: e.g. NeTEx bulk exchange; SIRI RT

A *Format* implements in a specific technology

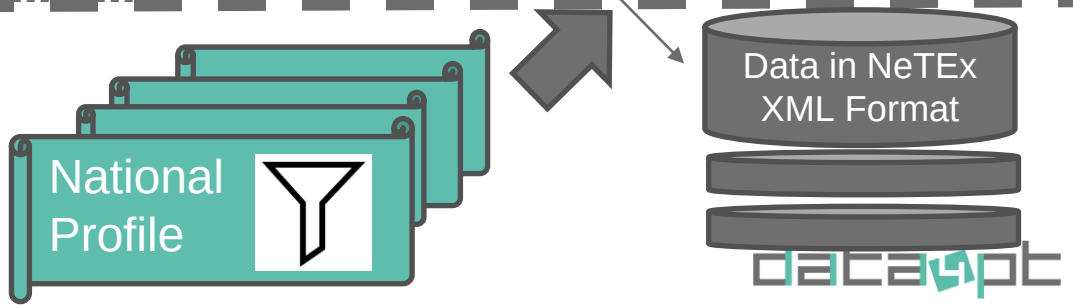
Derived from physical model, e.g. NeTEx XML Schema

A *Profile* specifies how to use the XML in a specif9c context

Functional Scope, use cases, workflow and data identifiers

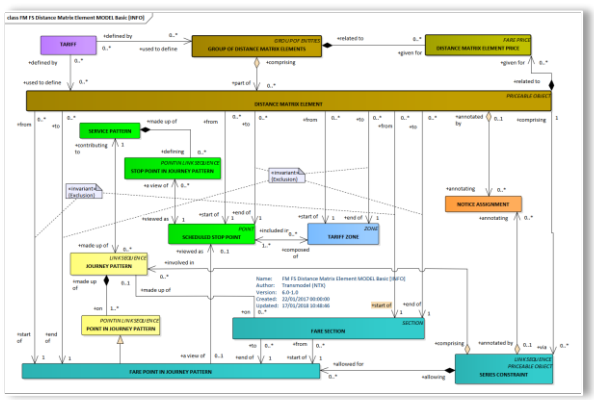


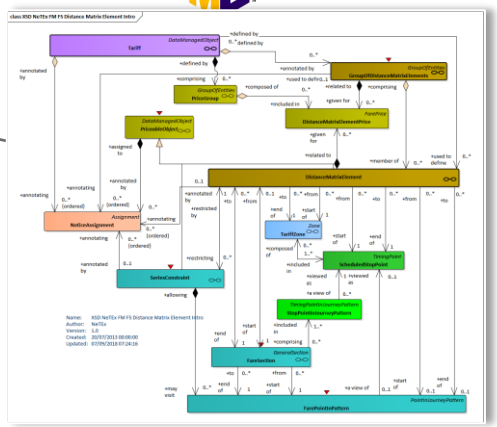
- **Allows impact analysis**
- **Avoid ad hoc reworking!**

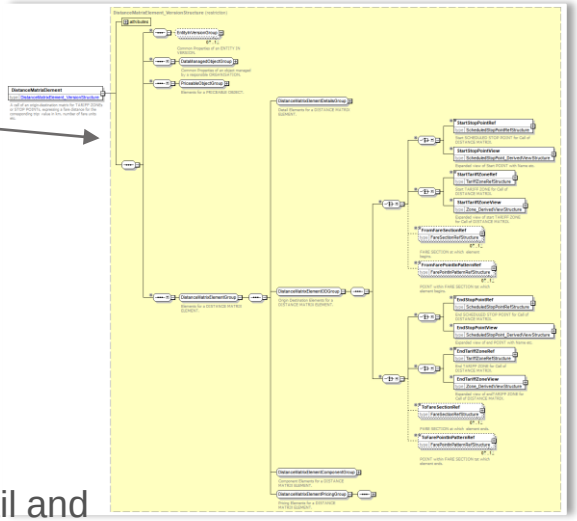


Package & Element level traceability
Modular design for large scale, long term extensibility

Conceptual → Physical → Schema







Traceability

Equivalent elements can be found at each level

Physical design and Implementation each add further detail and restrictions.

Simplifty implementation by Uniform “plumbing”; ids, references, versioning, integrity constraints

Tool support (EA, XML SPY, OXYGEN, etc)

Supporting multiple use cases – Different considerations for using NeTEx

Functional scope, (From Profile) Which Elements?

- Strict profile or allow extensions
- Completeness

Granularity of XML documents - Choose for efficiency/workflow

- One document per timetable, operator, network, region, country, etc ?
- One document per product, set of products, operator, etc

Organisation of data elements - Choose for easy human verification

- Version Frames, by function, operator , line
- Nested in-line or flat.

Identifier scopes - Choose to be unique in integration context

- Single local codespace, per object type
- Shared codespaces W3C domains e,g,national +Local:

Versioning – Choose for workflow

- Whole Dataset, Whole Frame, Individual element
- Full data set vs Deltas

Additional Validation rules – Choose for data quality requirements

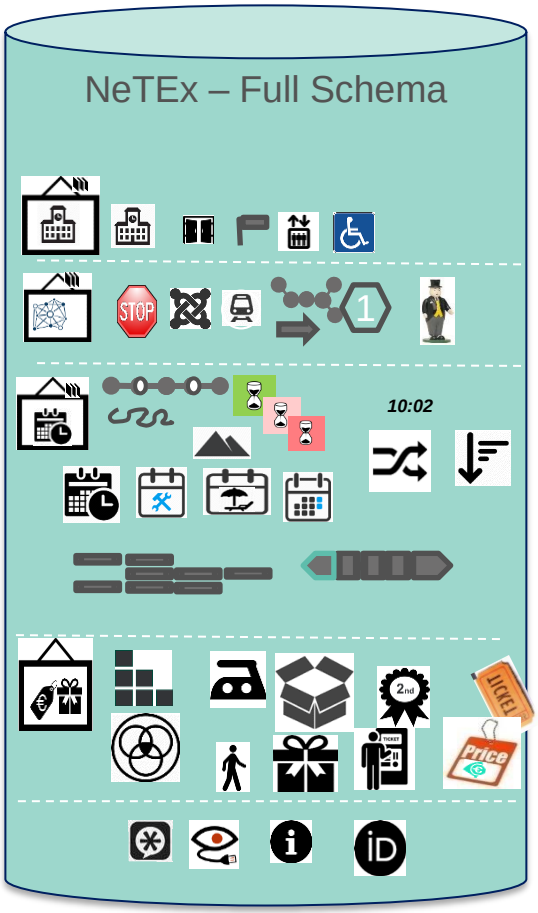
- Completeness, semantics

Protocols to exchange documents

- Periodic FTP output for preset parameters
- SIRI Request for dynamic parameters

Function scope Example of simplifying: **European** PI **Timetable Profile** -

Rich scope



EPIP Timetable Profile

Specific Scope: timetables for trip planners

Simpler to implement!

Organizing data sets : Different granularities are possible: …

Timetable Examples

Separate timetable per line?

E.g. Each XML document contains STOPs and JOURNEYS for one LINE of a given OPERATOR

Shared Network, e.g.

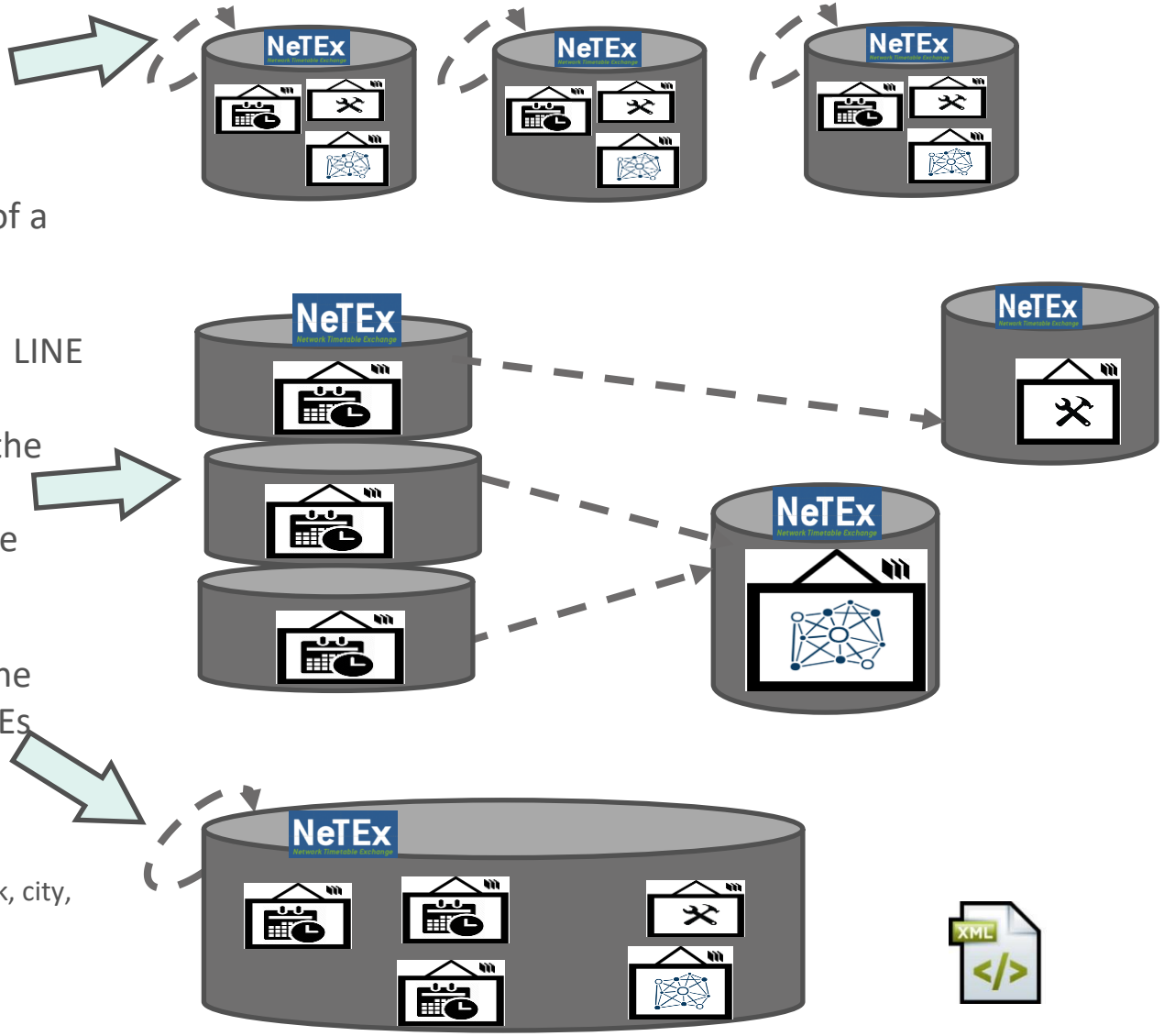
One document for each LINE with all its JOURNEYs

One document with all the STOPs

One document for all the OPERATORs

Whole Network, e.g.

One document with all the JOURNEYS for all the LINEs



Fare examples (Not shown)

Products sPer operator, network, city, etc

Products Vs Prices

Depends on Workflow, and data volumes– Choose for efficiency

Differences between Standards - Data Interoperability & Mapping with non Transmodel Standards – Use cases

1. To establish boundaries between standards covering different functional areas
   - Equivalent concepts that allow separate data sets to be integrated as one in an engine
     - Eg NeTEX/Inspire - Overlap: GIS FEATURE, ADDRESS, LOCATION
     - Eg NeTEx/Datex - Overlap: PARKING, ROAD ELEMENT
2. To import from legacy & existing data sets with similar scope (Open Data is Good!)
   - Individual specific mappings needed
     - Eg Rail Tap TSI B1, B2, B3 □ NeTEx Fares
     - E.g. GTFS Timetable □ NeTEx Timetable
     - E.g. GBFS □ NeTEx New modes
3. To export to other systems that want our data (Open Data is Good!)
   - Individual specific mapping for a given 'Profile'
     - E.g. NeTEx Timetable □ GTFS
     - E.g. NeTEx New Modes □ GBFS

Using model driven design to systematically compare standards and create mappings

1. Conceptual mapping (Against Transmodel)
   - Functional scope,
   - Conceptual Elements: differences in separation of concerns, views
   - Granularity of exchange
2. Physical mapping (against NeTEx, SIRI, DJP, etc)
   - One-to-one or one-to–several mapping of individual elements
   - One-to one mapping of attributes
   - Translation of data types, values
3. Technology mapping against NeTEx, SIRI, DJP, etc)
   - eg XML, CVS, JSON, WSDL etc
   - Metadata:: data source, versioning etc

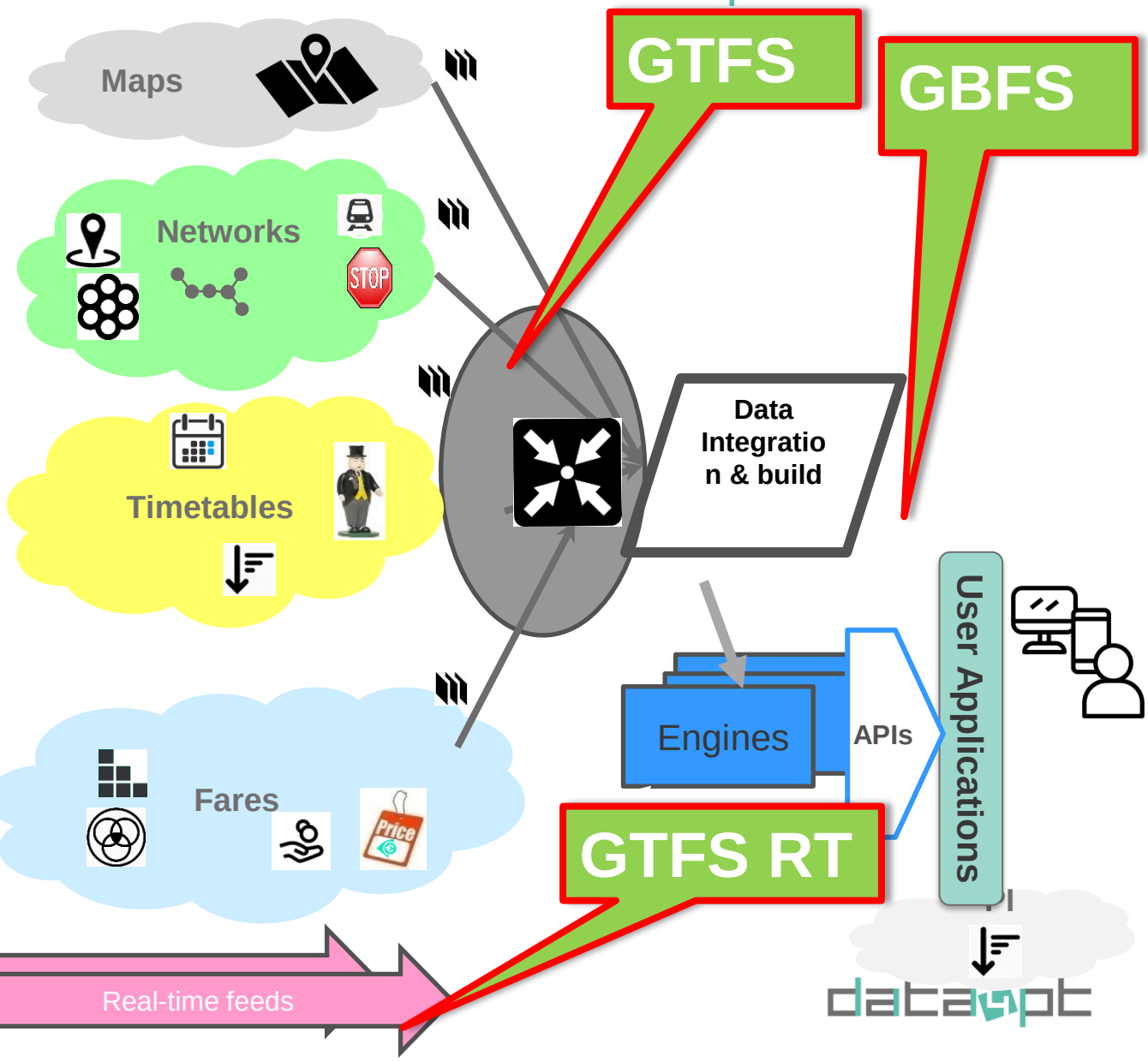
AN example Mapping - GTFS
- Other formats for downstream – An Example GTFS

Simple use case: Downstream, passenger info only

- Leaves out operational data sets, e.g timing
- Limited Fares: prices only

Simplifies 'deep' model using flat views

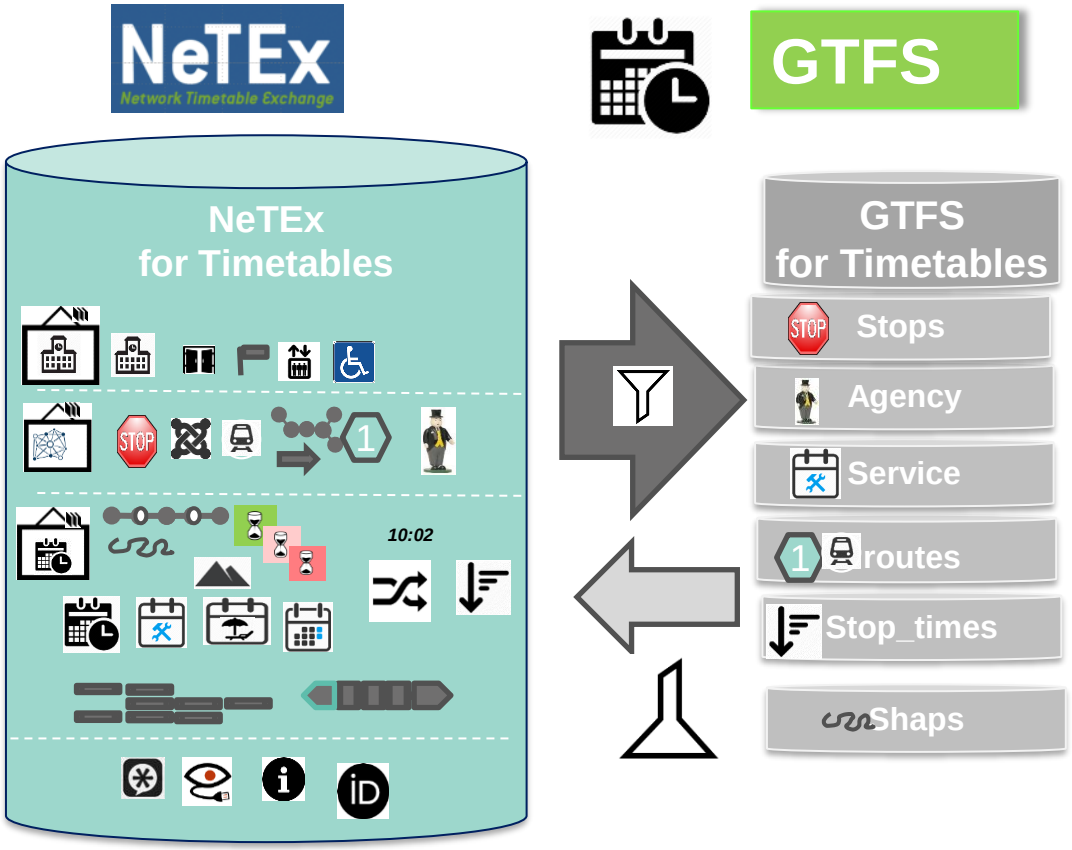
- GBFS different model to GTFS



Open Data is Good!
NeTEx & GTFS Interoperation - **Timetables**

GTFS is useful subset of timetable data for trip planners

- Does not have underlying reusable elements to build data sets
  - e.g. journey patterns, routes.
    - Simple view for consuming system: Layers & times at stop resolved to single sequence
- Does not cover complex aspects
  - e.g. grouping, connections, join/spilt, makeup, etc
- Does not cover some operational data



Round trip is "lossy"

- ❑ NeTEx to GTFS: OK
- ❑ GTFS to NeTEx: limited function

GTFS & NeTEx “Ecosystem” Comparison - Basic

GTFS

- Governance, Terms, Support
  - Google originated, free licence
  - Mobility Data support
  - GTFS community
- Methodology
  - No conceptual model
    (Eg GTFS &GBFS differ)
  - On Line documentation
  - Ad hoc profiles & extensions
- Functional Scope
  - Use case – **data for trip planner**
  - Simple , flattened model
    - Stops & Timetables.
    - Fare prices.

NeTEx

Network Timetable Exchange

- Governance, Terms, Support
  - CEN Standard, free XML schema licence
  - Data4PT support
  - SIRI/NetEx communities
  - EC mandated - for NAPs
- Methodology
  - Derived from Transmodel
  - odel driven design UML,
  - Uniform overarching architecture
  - CEN SPEcifcation
  - Profiles to specify usage
- Functional Scope
  - **Multiple use cases** (profiles)
  - Rich model, reusable components
    - Network, Equipment
    - Stops & Timetables,
    - Operational data, Timings, Equipment
    - Fares, Products, Fare Prices
    - Accessibility, Train makeup]
    - Etc ETc

GTFS & NeTEx Comparison - Technical

GTFS

- Technology
  - **CVS flat file format** (like 1st generation European CIF)
  - **One record type per file**
    - **Simple objects only**
    - Not all entities reified
    - Some overloading of attributes to have alternative meanings
  - Packaged as a zip file
  - Versioning at whole feed level
- Tools & Validation
  - **Custom programmed validator**,
    - Must be modified for each new feature
  - Open source editors & Translators available
  - Compact, Efficient

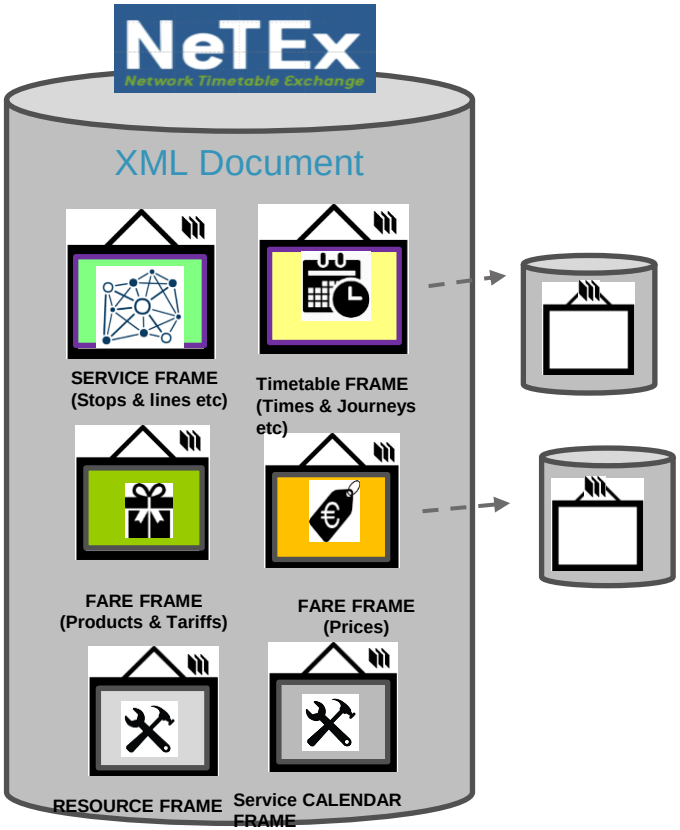
NeTEx

Network Timetable Exchange

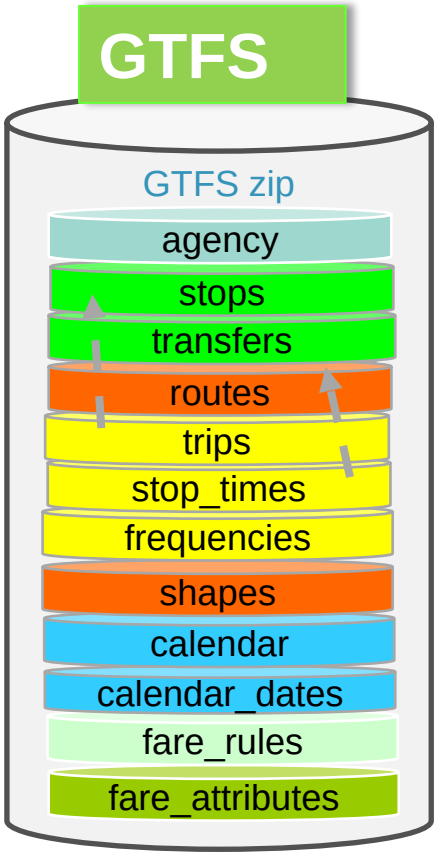
- Technology
  - W3C **XML Schema**
  - Multiple data elements per file
    - **Complex object structures allowed**
    - Explicit (no overloading)
  - Single (or linked)documents
    - Version Frames to organize
  - Uniform versioning (fine grained possible)
  - Responsibility mode
- Tools & Validation
  - **Automatic XML validators**
    - Tags, order, Data types, Referential integrity
  - Additional custom programmed validators possible
  - Various open source tools
  - More Verbose, complex

Packaging: Cvs records vs XML Documents

- NeTEx uses “Version Frames” to organise data within a document
- Rich objects
- Validation of data types & referential integrity
- Documents may have external references
- Global identifiers
- Alternative partitions possible for different workflows / scales
- Coherent Validity at frame level;



- GTFS zips flat files together
- Simple records,
- A separate file is needed for each object type
- Identifers unique within zip
- Fixed partition
- No inherent validation



Example mapping of a GTFS Agency to NeTEx OPERATOR

GTFS

```
agency_id,agency_name,agency_url,agency_timezone,agency_lang,agency_phone,agency_fare_url
10000,Transport For Ireland,http://transportforireland.ie,Irish Standard Time,en,1-800-300-604,http://transportforireland.ie/fares
```

NeTEx
Network Timetable Exchange


```
<Operator version="any" id="10000">
   <keyList>
      <KeyValue typeOfKey="gtfs">
         <Key>gtfs_agency_fare_url</Key>
         <Value>http:// transportforireland.ie/fares</Value>
      </KeyValue>
   </keyList>
   <Name>Demo Transit Authority</Name>
   <Locale>
      <TimeZone>Irish Standard Time</TimeZone>
      <DefaultLanguage>en</DefaultLanguage>
   </Locale>
   <ContactDetails>
      <Phone>1800 300 604</Phone>
      <Url>http://www.transportforireland.ie</Url>
   </ContactDetails>
</Operator>
```

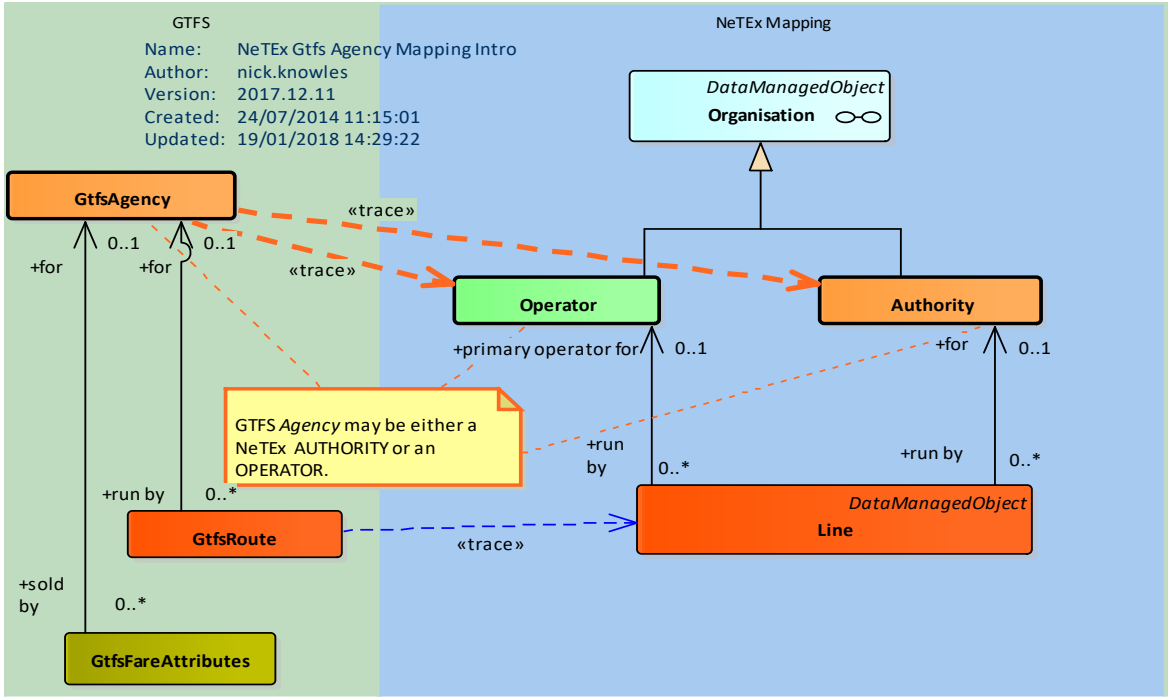
Simple Mapping Example GTFS Agency Mapping Intro

GTFS

NeTEx
Network Timetable Exchange

Mappping

- **Gtfs Agency record ➔**
- NeTEx OPERATOR (or AUTHORITY)



- NOTES ;

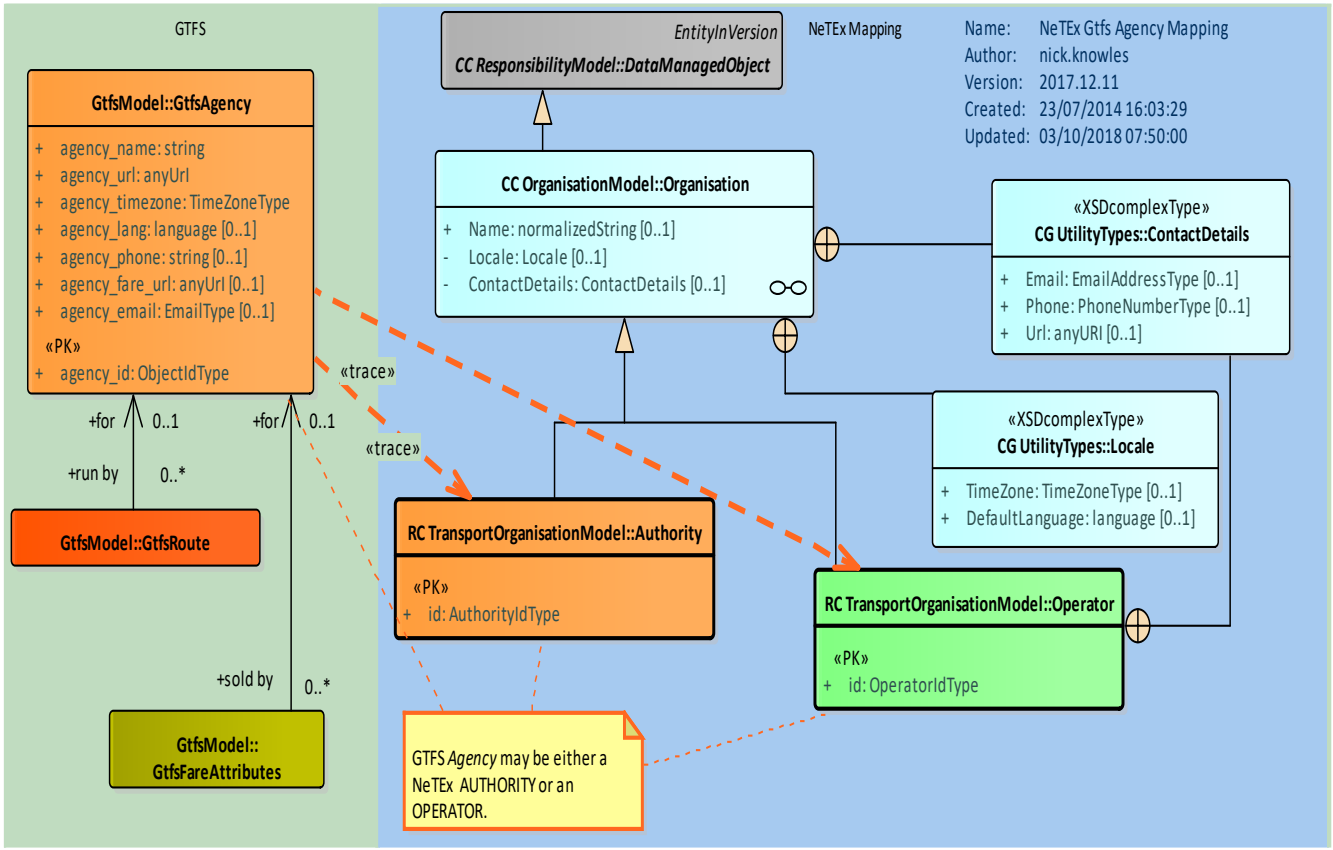
Conceptual mapping of entities is usually one to several

1. GTFS records are simplified views
2. Transmodel/NeTEx
   - Separates concerns unto separate elements
   - Uses inheritance

Mapping Example : GTFS Agency Mapping - Details

Network

- **Gtfs Agency record ➔**
- NeTEx OPERATOR (or AUTHORITY)



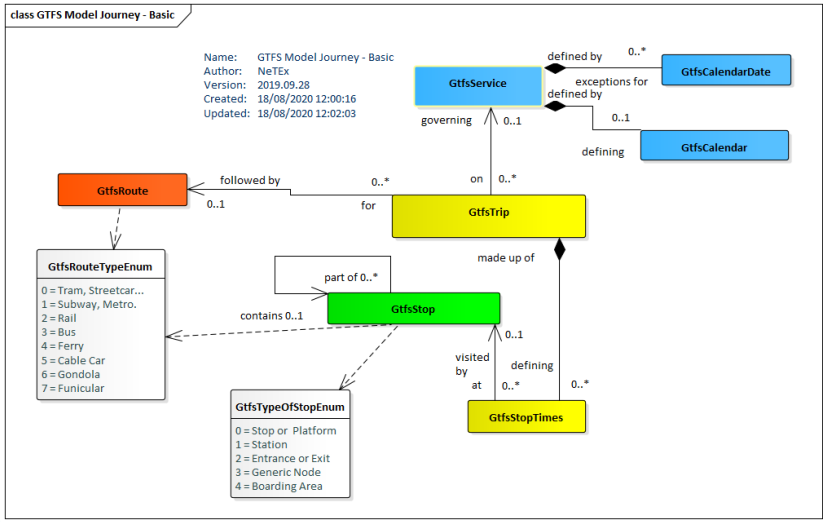
Gtfs Agency / NetEx OPERATOR : Attribute mapping table

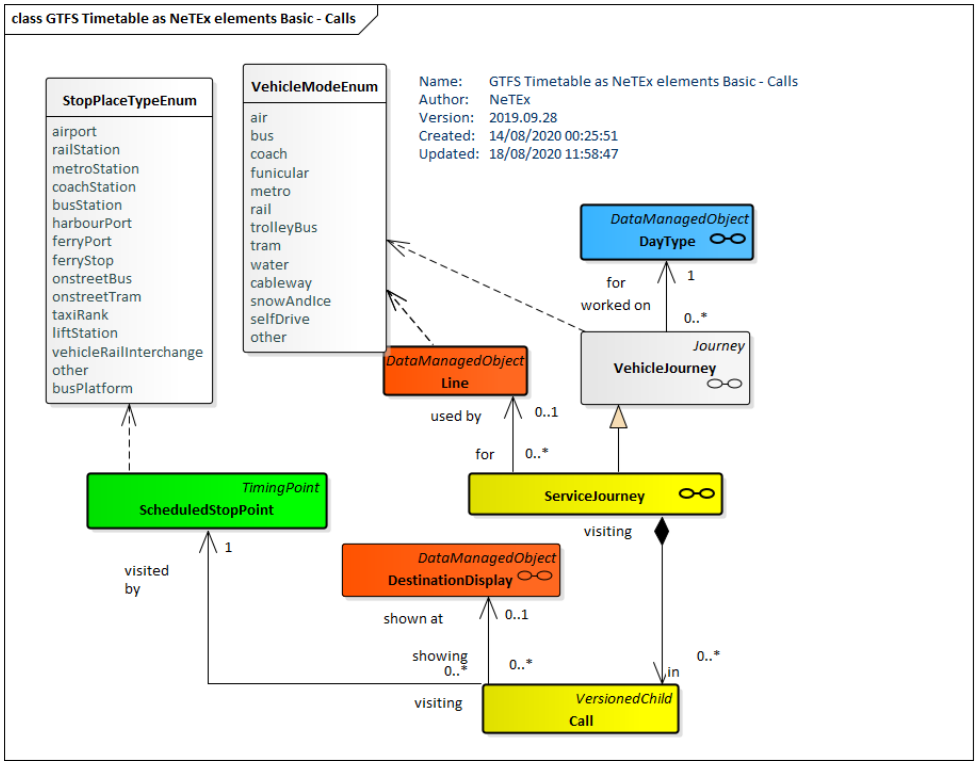
GTFS record	GTFS Attribute	NeTEx element	NeTEx attribute	Type	Alt Text
agency	agency_id	***Operator***	***id***	*OperatorIdType*	
	agency_name		***Name***	*MultilingualString*	Y
	TimeZone		***Timezone***	*xsd:string*	
	agency_language		***DefaultLanguage***	*xsd:lang*	
	agency_phone		***ContactDetails.Phone***	*PhoneNumber*	Y
	email		***ContactDetails. Email***	*Email*	Y
	agency_url		***ContactDetails. Url***	*xsd:anyURI*	Y
	agency_fare_url		Keylist.gtfs_fare_url	*xsd:string*	

Mapping GTFS Trips to NeTEx Journeys – The basics

Easy!

- You say *route*, we say LINE…
- You say *trip*, we say VEHICLE JOURNEY...
- You say *stop_times*, we say CALL…
- You say *headsign* ,we say DESTINATION DISPLAY



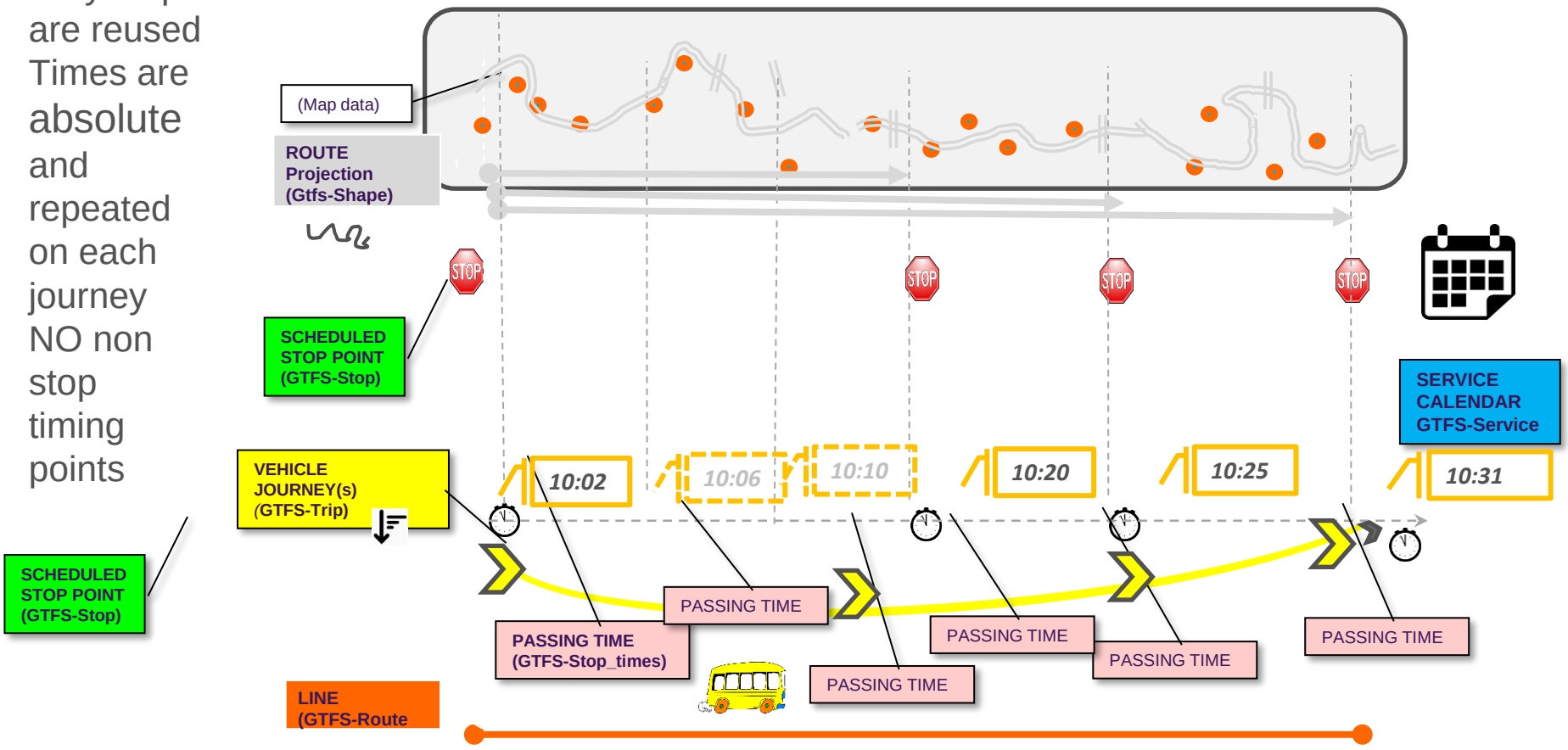


GTFS → NeTEx

GTFS

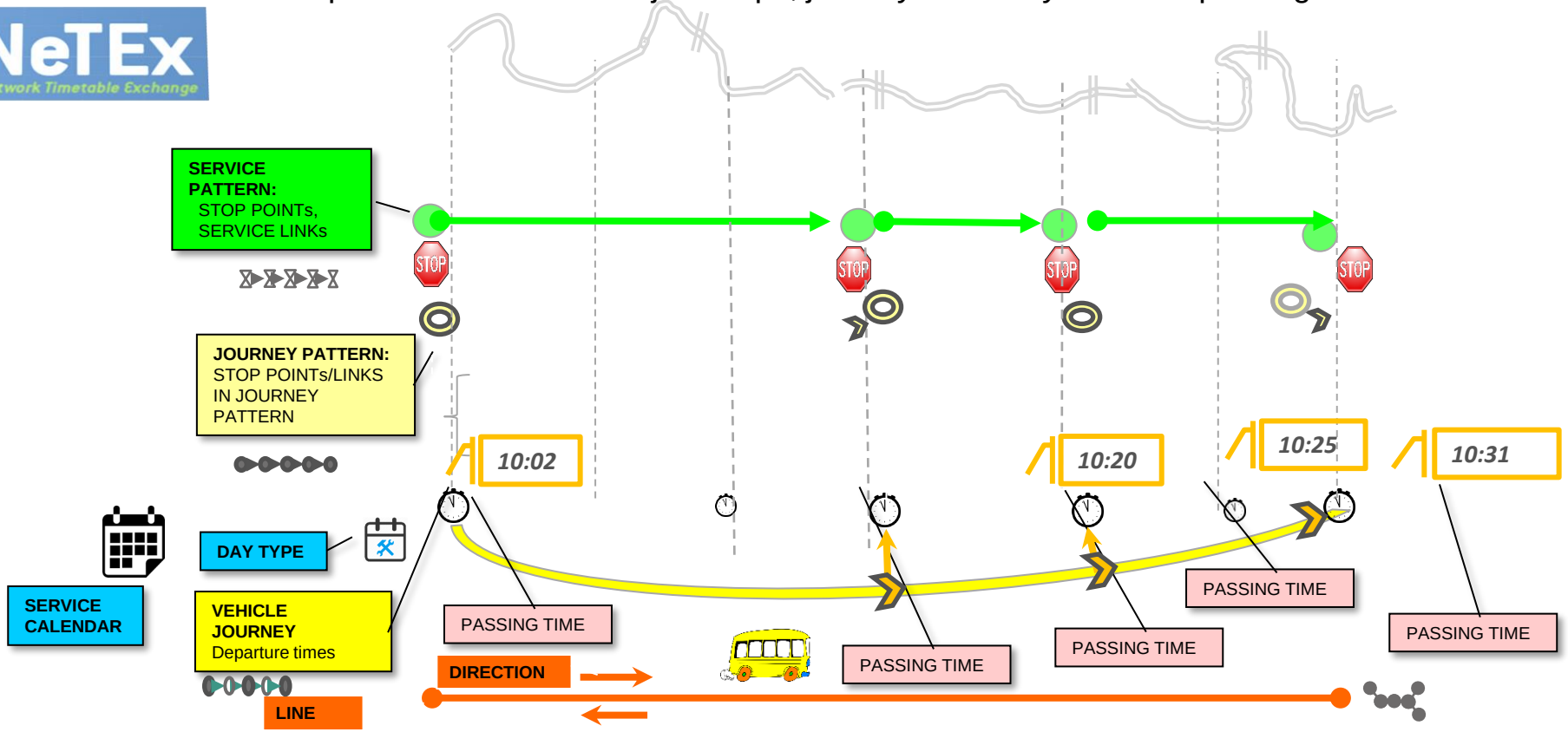
A VEHICLE JOURNEY (Gtfs-Trip) is for a LINE

Only stops are reused
Times are absolute and repeated on each journey
NO non stop timing points



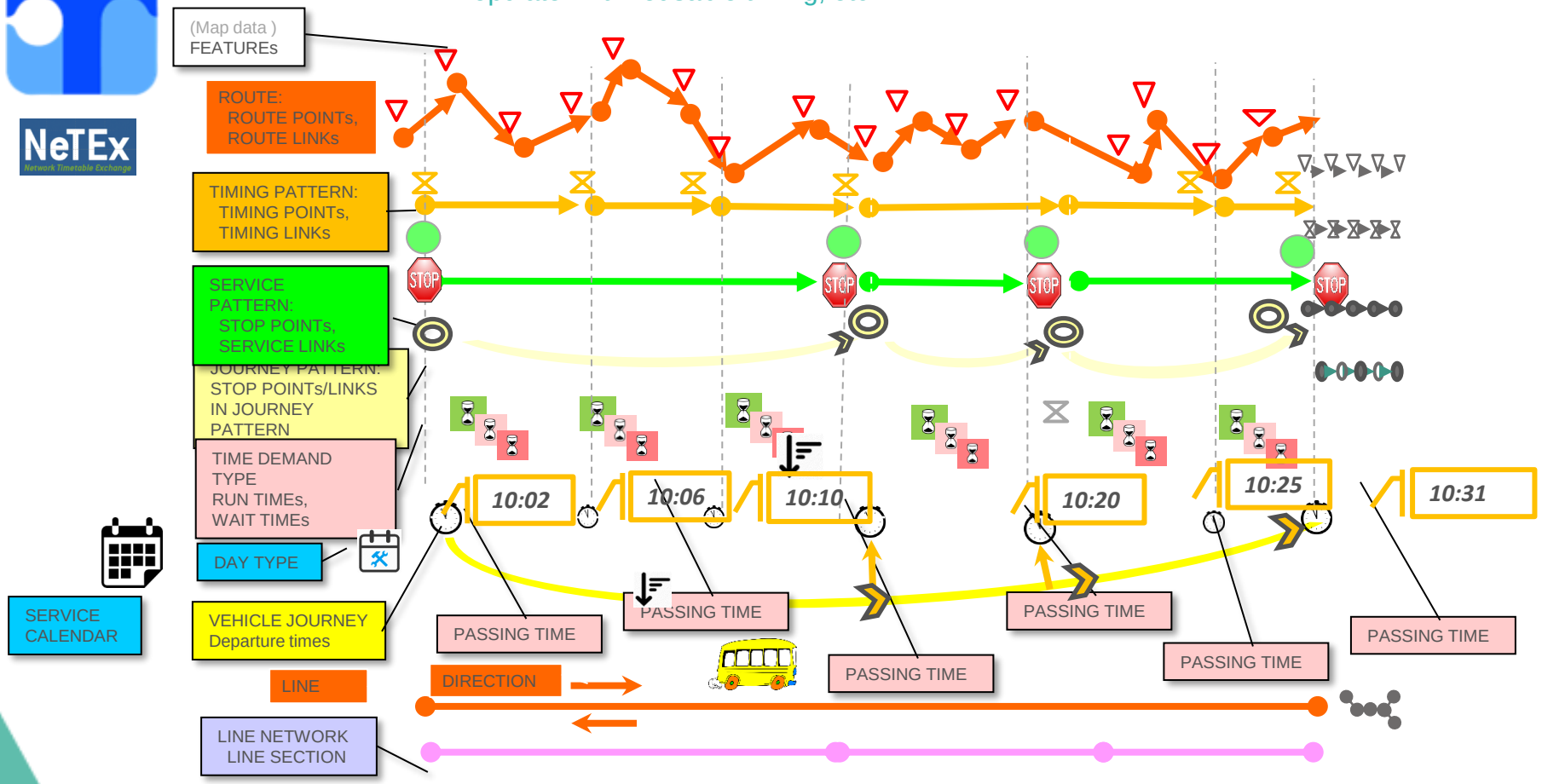
TM Minimal: A VEHICLE JOURNEY follows a JOURNEY PATTERN, for a SERVICE PATTERN, along a ROUTE for specified PASSING TIMES

Populate document with just stops, journeys and fully resolved passing times



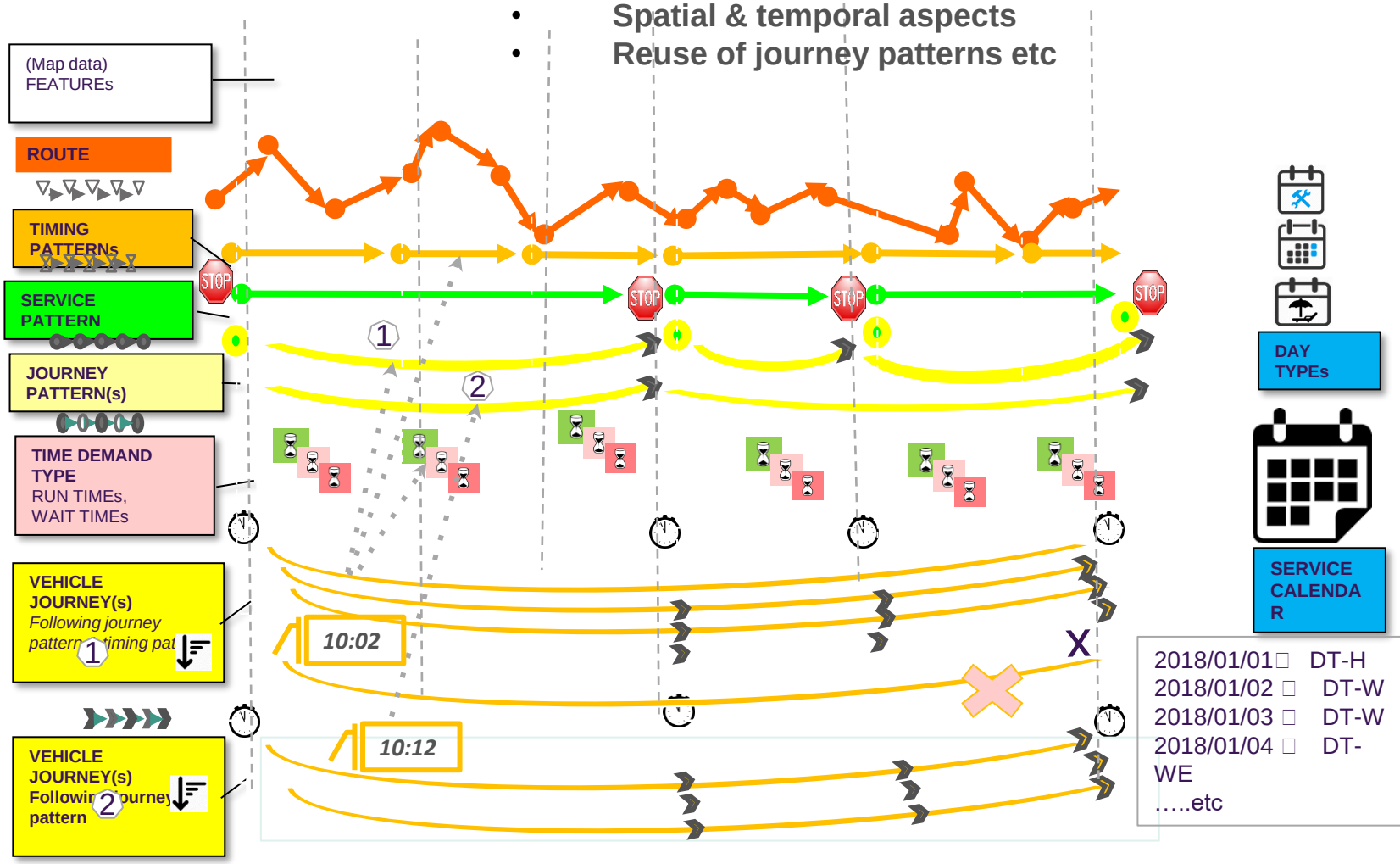
TM Full Model: A VEHICLE JOURNEY follows a JOURNEY PATTERN, to a TIMING PATTERN, over a SERVICE PATTERN, along a ROUTE during a TIME DEMAND TYPE

Populate with reusable timing, etc



JTM reuse: journeys can be specified completely just by indicating a Joueney pattern and a START Time

- Separate levels of abstraction for each concept
- Spatial & temporal aspects
- Reuse of journey patterns etc



GTFS record / NeTEx basic correspondences




GTFS record	Transmodel / NeTEx	Notes
agency	**OPERATOR** or AUTHORITY	
stops	**SCHEDULED STOP POINT**, STOP PLACE + QUAY	Complex mapping
pathways	**PATH LINK,** SIGN EQUIPMENT	
transfers	CONNECTION SERVICEJOURNEY | INTERCHANGE, **INTERCHANGE RULE**	Complex mapping
routes	**LINE**	
calendar	DAY TYPE, **DAY TYPE ASSIGNMENT**	
calendar_dates	**DAY TYPE ASSIGNMENT** and OPERATING DAY	
trips	**SERVICE JOURNEY** + DESTINATION DISPLAY	
stop_times	**STOP POINT IN PATTERN** + PASSING TIMES + DESTINATION DISPLAY &/ or CALL	Complex mapping
frequency	**HEADWAY JOURNEY GROUP,** RYTHMICAL JOURNEY GROUP with TEMPLATE SERVICE JOURNEY	
shapes.txt	ROUTE LINK, **POINT ON LINK,** LINK PROJECTION, LineString,	
levels	**LEVEL**	

Thank you for your attention!

www.data4pt-project.eu/

@Data4PT

DATA4PT Project