Deliverable D2.3
Validation tools release 2

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

Deliverable 2.3 – Validation Tools Release 2

Due date of deliverable: 30 June 2023

Actual submission date: 15 January 2024

Dissemination Level		
PU	Public	X
CO	Confidential, restricted under conditions set out in Model Grant Agreement	
CI	Classified, information as referred to in Commission Decision 2001/844/EC	

Start date of project: 02/01/2020 Duration: 54 months

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

Document Control Sheet

Deliverable number:	**D2.3**
Deliverable responsible:	ITxPT
Work package:	WP2
Main editor:	Anastasia Founta

Editor name	Organisation
Theobald Nutte	ITxPT
Eliot Terrier	ITxPT
Petter Kvarnfors	ITxPT
Jesper Johansson Törnros	ITxPT

Document Revision History			
Modifications Introduced			
Version	**Date**	**Reason**	**Editor**
1.0	20/6/2023	Structure – main input	Anastasia Founta
1.1	7/2023	Technical input	Petter Kvarnfors Jesper Johansson Törnros
1.2	9/2023	Consolidation of input and editing	Theobald Nutte Eliot Terrier
1.3	10/2023	Consolidation of input and editing	Anastasia Founta Theobald Nutte Eliot Terrier
1.4	01/2024	Review	Efe Usanmaz

Legal Disclaimer

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

EXECUTIVE SUMMARY

A key activity of DATA4PT project is the development of validation tool for NeTEx and SIRI datasets. As NeTEx and SIRI are the EU standardised formats for public transport data in National Access Points (NAPs), the purpose of validation is to ensure the published data is of certain level of quality. Ensuring data quality is important for the overall objective of the project, which is to enable the implementation of ITS Directive Delegated Regulation (DR) EU 2017/1926 and therefore the interoperable exchange of travel and traffic data across Europe.

The first step for the development of validation tools was described in the deliverable D.2.1. Requirements Report and includes the definition of functional requirements and a benchmarking survey of the currently available tools. In the deliverable D.2.4 "Testing Procedure Report", it was defined the architecture of the tool and the testing procedure. Considering the priority in static data according to MMTIS DR EU 2017/1926, the architecture focused on the validation of NeTEx datasets, as first step. Following this methodology, the development and first release of the validation tool addressing NeTEx datasets was made available in June 2022.

After one year of pilot implementation of the tool, where several users shared their feedback and many updates have been applied, the second, and the final release of the tool in the framework of DATA4PT was delivered in September 2023.

This report constitutes a short summary of the work done during the pilot implementation of the tool but also a manual for the current and future users, accompanying the tool technical artefacts[1].

[1] All technical artefacts and source codes are available in https://github.com/ITxPT/DATA4PTTools. The manual is also incorporated in relevant link.

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

List of partners

Partner's name	Acronym	Country
Union internationale des transports publics	UITP	Belgium
Information technology for Public Transport,	ITXPT	Belgium
Bundesministerium für Klimaschutz, Umwelt, Energie, Mobilität, Innovation und Technologie	BMK	Austria
Ministry of the sea, transport and Infrastructure	MMPÏ	Croatia
Centrum dopravního výzkumu, v. v. i.,	CDV	Czech Republic
Trafikstyrelsen (Danish Civil Aviation and Railway Authority),	TS	Denmark
Direction générale des infrastructures, des transports et de la mer	DGITM	France
Ministero delle Infrastructture e dei Transporti	MIT	Italy
INSTITUTO DA MOBILIDADE E DOS TRANSPORTES, I.P.	IMT	Portugal
Republika Slovenija, Ministrstvo za Infrastrukturo	MZI	Slovenia
Trafikverket (Swedish Transport Administration	STA	Sweden

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

Abbreviations and Acronyms

API	Application Programming Interface
AVMS	Automatic Vehicle Monitoring Systems
MMTIS	Multimodal Travel Information Services
EPIP	European Passenger Information Profile
GUI	Graphical user interface
MS	Members States
NAP	National Access Point
NeTEx	Network Timetable Exchange
PTO	Public transport Operators
PTA	Public Transport Authorities
SaaS	Software as a Service
SIRI	Service interface for real-time information
TRANSMODEL	Public Transport Reference Data Model

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

TABLE OF CONTENTS

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

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

List of Figures

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

List of Tables

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

INTRODUCTION

Data quality is one of the key factors for the provision of successful services in all domains, including mobility. Therefore, it is also an important parameter in MMTIS Delegated Regulation EU 2017/1926 in order to ensure that the openly available data through the National Access Points are suitable to be re-used, to feed multimodal mobility applications, and to improve mobility services across Europe addressing the changing passenger behaviours after the global pandemic[2].

In the framework of DATA4PT a validation tool (**Greenlight**) has been developed, dedicated to check and validate datasets according to the EU CEN technical standards NeTEx CEN/TS 16614.

The tool has been developed following three main steps:

1) The definition of requirements regarding functionalities and use cases the tool aims to support. The requirements have been collected by Member States and their NAP operators which participated to the research. The requirements have been also designed after investigating the existing tools, evaluating gaps and important functionalities. Based on the findings of this study, an architecture has been designed, giving priority to static data, aligning also with DR priorities. This step is described in *D.2.1. "Requirements"* and *D.2.4 "Test Platform".*
2) The development of version **V.0.** of the tool. This version has been tested by a sample of users, which consists of the NAP operators of the 9 Member States, partners of the project, leading to a first public release for wider utilisation in June 2022 (**V0.3.4**)**.**
3) The collection of feedback from the users and the final public release (**V.1.7**) which contains the technical artefacts, the source codes, the web interface and the manual ("readme"). **This report** summarises the main characteristics of the tool and provide guidelines of how to use the tool. Therefore, it **stands also as manual**.

[2] Future of Mobility post-COVID, UITP & Arthur D. Little, July 2020 https://cms.uitp.org/wp/wp-content/uploads/2020/10/2020-ADL-FoM-Lab-UITP_Future-of-Mobility-post-COVID-study.pdf

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

1 GREENLIGHT - THE DATA4PT VALIDATION TOOL

The tool consists of several components, each with a different responsibility. This will ensure that the tool is modular and that each component is easy to understand and maintain.

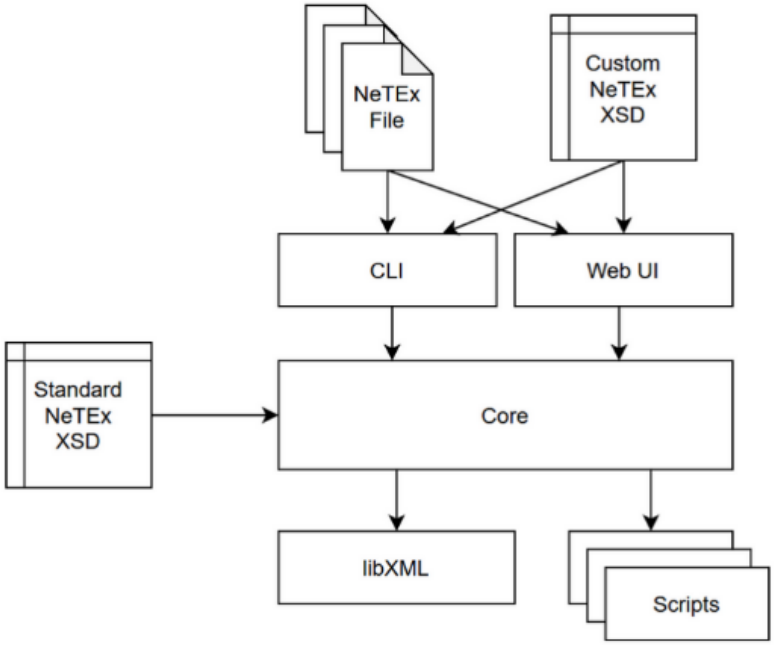

Figure 1: Main components of Greenlight Validator.

Its components are:

- The **core** part
- The standard XML library (libXML)
- The Command Line Interface (CLI)
- The Individual validation rules (scripts)

To facilitate the users, information around system requirements and installation process of the command line Interface is also provided.

The full package of the tool is available in GitHub/ ITxPT/DATA4PTTools. As complementary documentation and material, the users are recommended to consult NeTEx-CEN/NeTEx.

1.1 Core part

The ***core*** is the main component of the tool, it reads the configuration, handles file imports, calls the validation scripts, and summarizes the result. The ***core*** provides an API that other components use to control the validation or to get access to shared functions, e.g, in libXML. **The API also makes it possible to extend the tool with different front ends, as the CLI and Web Interface.**

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

1.1.1 WEB INTERFACE

The web interface provides an easy-to-use interface via the web browser. The web interface makes the tool easier to use for the occasional user or for just testing small files. After loading the web page, you can select the NeTEx profile to use, select one or more validation rules and then run the validation. After completion you get the result on the web page but can also download it to a file. More details will be shared in 4 Manual for Web Interface

1.2 STANDARD XML LIBRARY - LIBXML

An open source, standard library integrated into the tool. It is libXML that does all the XSD and XML validation. It is called from the scripts via the API in the Core component.

1.3 COMMAND LINE INTERFACE - CLI

The Command Line Interface is used in a terminal or integrated in an import/export pipeline. Parameters are used to configure the tool and to specify the files to be validated. The result can be read in the terminal or saved as a file.

1.4 INDIVIDUAL VALIDATION RULES - SCRIPTS

Individual validation rules implemented as scripts. The scripts are written in JavaScript that is easy to start with and JavaScript is also well documented. The validation scripts are small programs that each implements one or more validation rules. The scrips provided with the tool implements one rule per script to make it easy to follow and understand how they work. To gain a better performance several rules can be implemented in the same script. Each script uses the API in Core to load the files to validate and to call functions in libXML. XPath provided via libXML is used by most of the scripts to search for and compare different elements in the NeTEx-files.

1.5 SYSTEM REQUIREMENTS

To run the tool locally you must ensure that the machine used has the capability to handle the files to be validated. The validation times can be long, and the tool can stop if the processing power or memory is too low. Below is a recommendation for the configuration of a machine. Be aware that very large or many files affect the performance and can result in longer validation times, even on a machine with the recommended hardware.

Minimum	Recommended	Best performance
4 cores	6 cores	6 cores
8 GB memory	16 GB memory	32 GB memory

1.6 LOCAL INSTALLATION

To use the tool locally, you need to install Docker on the computer that you will use. You can use Windows, Mac or Linux as your base operating system, and you will find Docker and instructions on how to install in the Docker Getting Started guide.

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

After you have installed Docker, you can get the latest version of the Greenlight image by typing the following command in a terminal window:

```
docker pull itxpt/greenlight
```

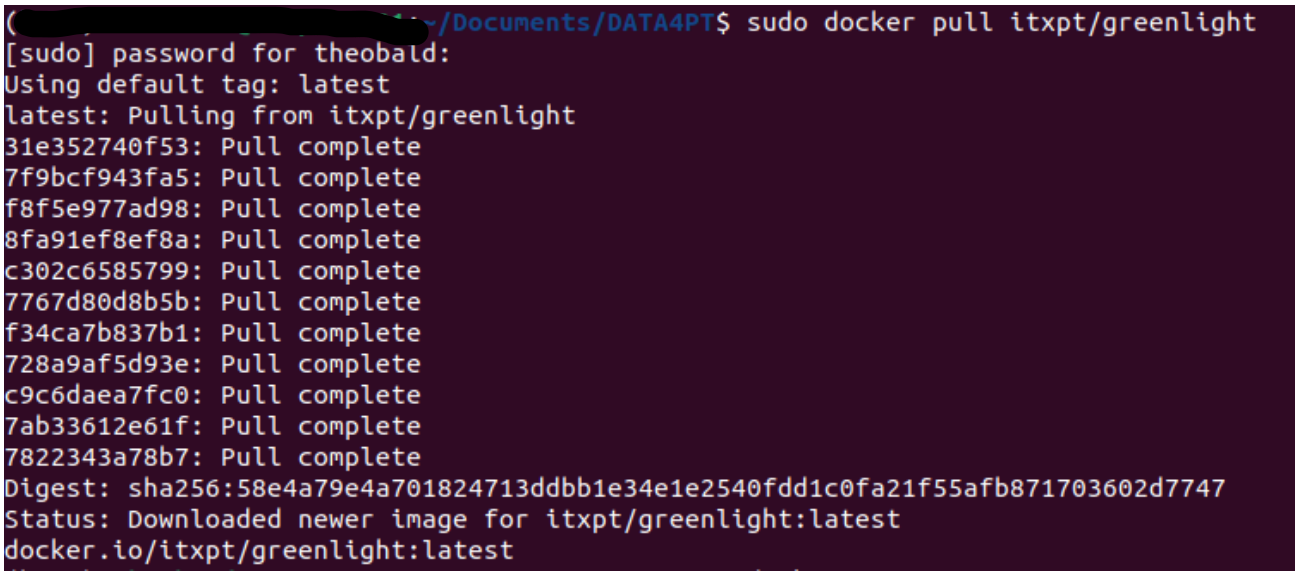

Figure 2: Command of terminal window to call the latest version of the tool.

After installing Docker, we suggest that you first start to use the web interface to verify that the installation works and to learn more about the functionality.

To start the web interface, use the following command:

```
docker run -it -p 8080:8080 itxpt/greenlight server
```

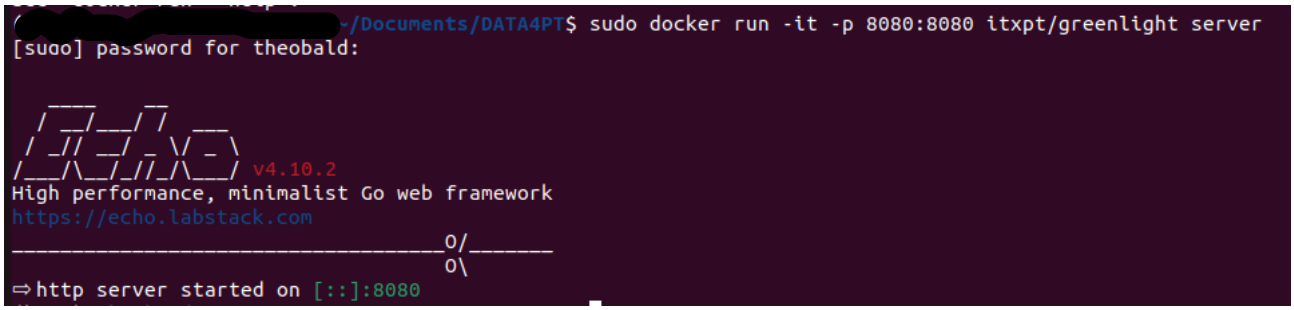

Figure 3 Command to use the web interface to verify that the docker installation works properly.

If you have used the image before, you can also start the web interface via Docker Desktop. Go to the Containers tab and press Start on the Greenlight container.

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

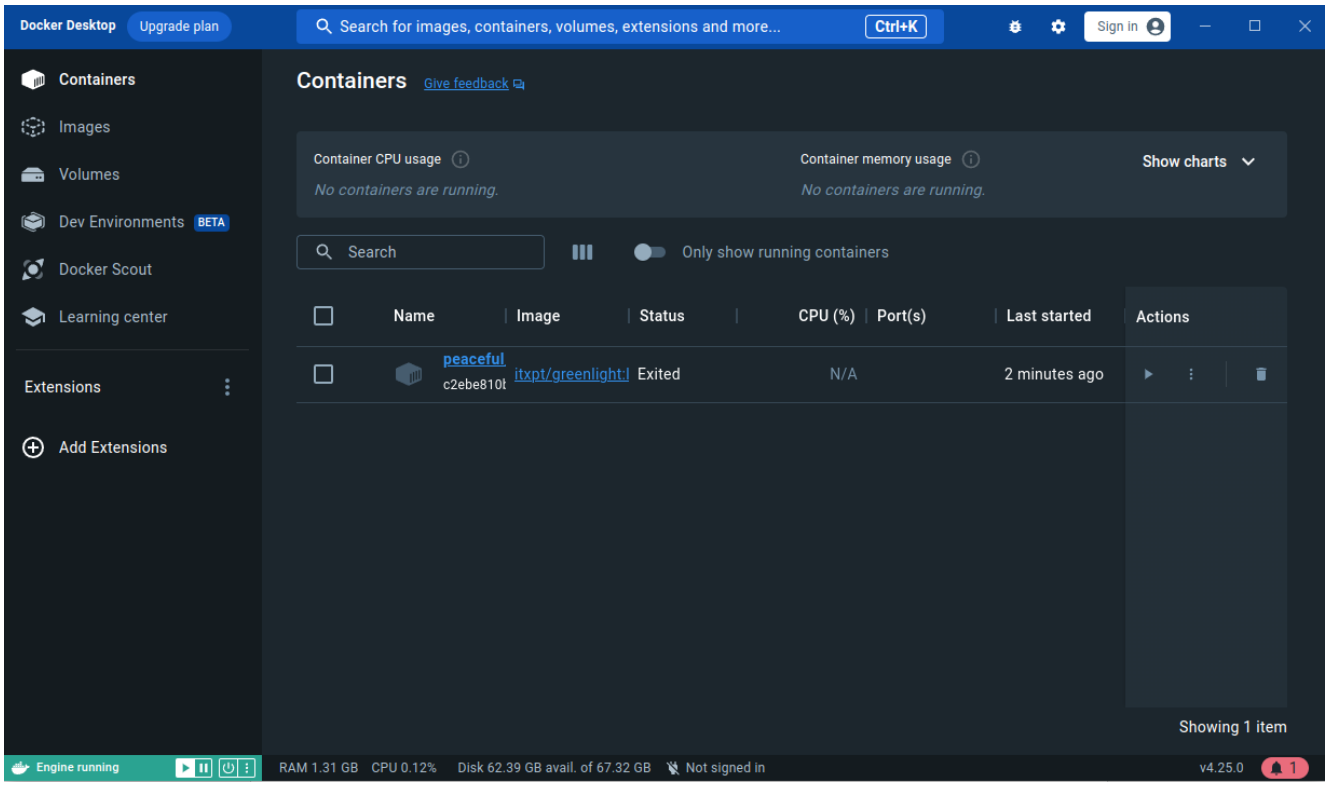

Figure 4 Using the Docker Desktop to run the web interface.

1.7 RELEASES HISTORY

The tool has been developed in three (3) main steps:

- First version (V.0.0) for testing purposes.
- First public release (V.0.3.4) for pilot implementation from different stakeholders and systems.
- Second public release (V.1.0.7)

However, updates have been made throughout the testing and pilot implementation with the aim to support the users, replying to their needs based on the collected feedback. The collected feedback is summarised in D.4.3 Feedback on validation tools 2.

The following table summarises the main features that are incorporated in the first and second public release of Greenlight NeTEx validator, summarising also the main additions, updates and fixes that have been performed. The detailed Release history is available in GitHub/ITxPT/DATA4PTools/releases.

Table 1 Main features incorporated in the first public release V.0.3.4 in June 2022.

Relevant releases	v0.3.4 [2022-05-30], v0.3.3 [2022-05-09], v0.3.1 [2022-04-10], v0.3 [2022-04-10]
Chore (actions no related to the code)	- Add dependencies - Add EPIP schema - Create README

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

	- Add NeTEx related XSD links in web interface - Improve version reference - Add License and texts in web interface - Update validation rule texts in README and web interface - Add note about limitations on web interface
Refactor (changes regarding structure)	- Remove GFX (graphics) terminal output of reporting - Add JSON output of reporting to terminal - Add rules configuration - Add subsequent validation callback - Add docker callback in result - Add a text section about rules in configuration window - Replace logger, update configuration tab and implement required model - Move file upload in preparation of custom XSD
Features	- Basic frontend to recover MQTT - Add MQTT broker fork - Add MQTT broker fork - Publish progress over MQTT - Remove old design - Remove redundant terminal GUI (web interface) - Add cap support and timings to concurrency - Proxy MQTT and add report download - Re-enable a couple of rules - Copy, styling, report download and fixed - Add web app build stage - Implement new rules - Add additional rules to web interface
Bug fixes	- Handle ws ssl (WebSockets Secure Sockets Layer) - Handle nextjs (a framework used for web application) parameterized paths - Fix readme link and dirname - Make sure error count match with returned messages - Disable rules until they have been resolved - Add a stateless approach to file loading
Notes	- Update readme
Performance	- Only render when needed

Table 2 Main features incorporated in the second public release V.1.0.7 in August 2023.

Relevant releases	**v1.0.7 [2023-08-07], v1.0.6 [2023-07-28], v1.0.5 [2023-07-12], v1.0.4 [2023-06-26], v0.5.5 [2023-03-21], v0.5.3 [2023-02-08], v0.5.2 [2023-02-06], v0.5.1 [2023-01-25], v0.5.0 [2022-12-02], v0.4.3 [2022-11-09], v0.4.2 [2022-09-12]**	
Chore	- Update dependencies - Add diff NeTEx versions - Version XSD schemas - Clean up dependencies - Fix build warnings - Update readme - Update ignore files	- Move profiles and scripts to app - Stricter linter - Simplify building by adding Makefile - Add legal documents - Add built commands
Refactor	- Move CLI-only relevant code to cmd and cleanup from js API - Implement new js API - Minor refactor from previous API changes	- Flatten script compilation - Fix configuration after dependencies upd - Optimize schema cache - Add links to NeTEx, SIRI and TRANSMODEL

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

	- Add more types and update response structure - Update CLI to new API - Update configuration and output - Remove unused code - Gofmt (a tool that automatically formats go code)	- Link to exact GitHub folder - Add navigation buttons and config to steps
Features	- Remove telemetry collection - Complete rewrite of js API - Add internal API - Add favicon - Add more event types - Add cli only docker build - Add profiles to validation - Add example profiles -	- Add profiles to web server - Add profile endpoints - Updated design and app icon - Added optional Firebase authentication - Add custom XSD - Update EPIP version 1.1.2 - Accent input label
Bug Fixes	- Fix memory issue using setcontextnode - Fix slow queries and type of response objects - Disable next telemetry collection - Fixed load state on validation - Fixed js api typing - Add missing return result - Add missing line no to XSD errors - Handle nested next js resources - Handle configuration state change - Update privacy policy - Fix environment key	- Minor script optimizations and fixes - Fix unresponsive validation - Add better control of tmp files - Modify key ref constraint evaluation - Ignore externally referenced keys - Correctly handle attribute lookups - Sort validation results by line number - Fix typo - Backdrop cover entire view - Handle limbo state when moving back
Notes	- Add XSD benchmark - Add typography style	
Performance	- Optimize docker build size - Add build arm platform	
Notes	- Modify text sizes and change font - Rewritten JS API Extended standard library with ability to query multiple files, added error types and predefined xpath paths for ease of use Extended node API with shorthand methods for properties, attributes and values as well as added feature for method chaining - Added support for script configuration (e.g. setting max distance between two stops) - Added support for different NeTEx schema versions (1.01, 1.02, 1.03 and 1.2) - Added support for different NeTEx schema versions (1.01, 1.02, 1.03 and 1.2) - **known issue:** Legacy NeTEx versions is seemingly incompatible with _libxml2_ (or in general?)F - Replaced large part of the validation lifecycle with a event emitter, giving the user control of which information is consumed - Squashed a bunch of bugs related to performance, validation result, memory security & errors	

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

2 VALIDATION RULES

The Validation Rules relevant to Greenlight NeTEx validator are mainly divided into two major categories:

- The rules based on XML schema
- The rules beyond XML schema

2.1 RULES BASED ON NETEX XML SCHEMA

This kind of rules is relevant to data quality dimensions such as uniqueness, consistency and completeness.

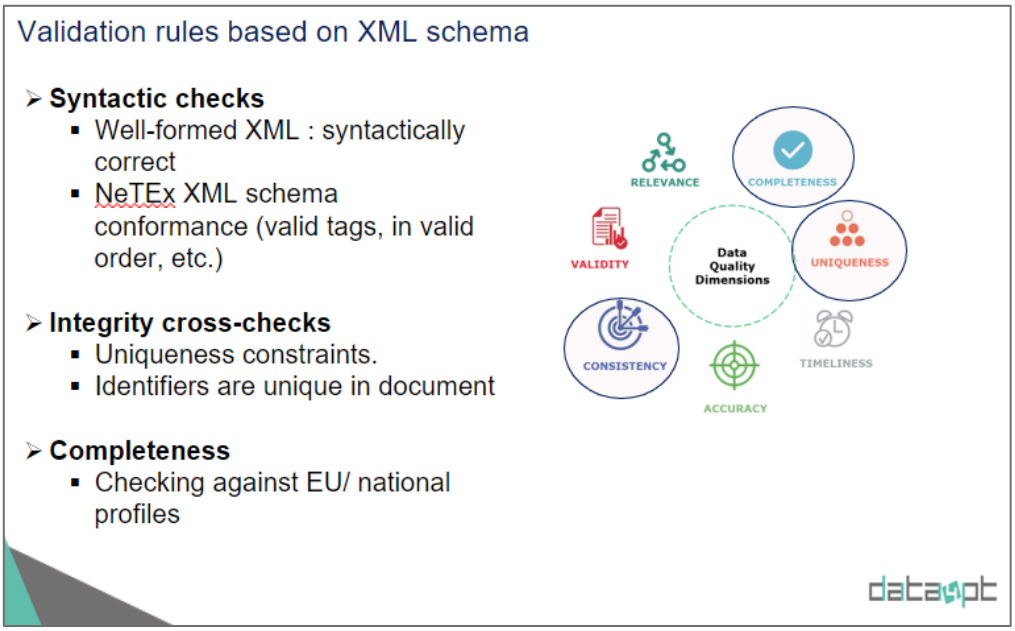

Table 3 Data Quality dimensions and correspondance of XML schema based validation rules.

The NeTEx XML schema rules can be applied automatically by any XML Validator. They concern, *Syntactic checks*, *XML schema conformance checks, Integrity cross-checks.* Examples of this kind of rules are shown in the following Table.

Table 4 Examples of rules based on XML schema.

Rules category	Examples of rules
Syntactic checks	• Well-formed XML : syntactically correct . i.e. <tag attribute="xx">data value</tag>
XML schema conformance checks	• Valid tags, in valid order. No empty tags • Valid cardinality: required, optional, 0,1,n • Encoding of Data Types: - *Date, Time, text, number, currency value, etc., etc.* • Enumerated values are valid. E.g. Mode bus, rail, tram…

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

Integrity cross-checks	• Uniqueness constraints. - Identifiers are unique in document • Referential integrity constraints. - Any referenced entity must also be present in same file.

2.1.1 COMPLETENESS CHECKS

The existence of the NeTEx profiles allows to check the completeness of a dataset against a particular profile. As full NeTEx schema is quite complete, including all elements that concern public transport, profiles are often used to limit the scope and address national and local specificities and needs. The Greenlight validator offers the possibility to upload your custom profile (e.g. a national or local profile), and choose from the predefined list the European Minimum Profile (EPIP).

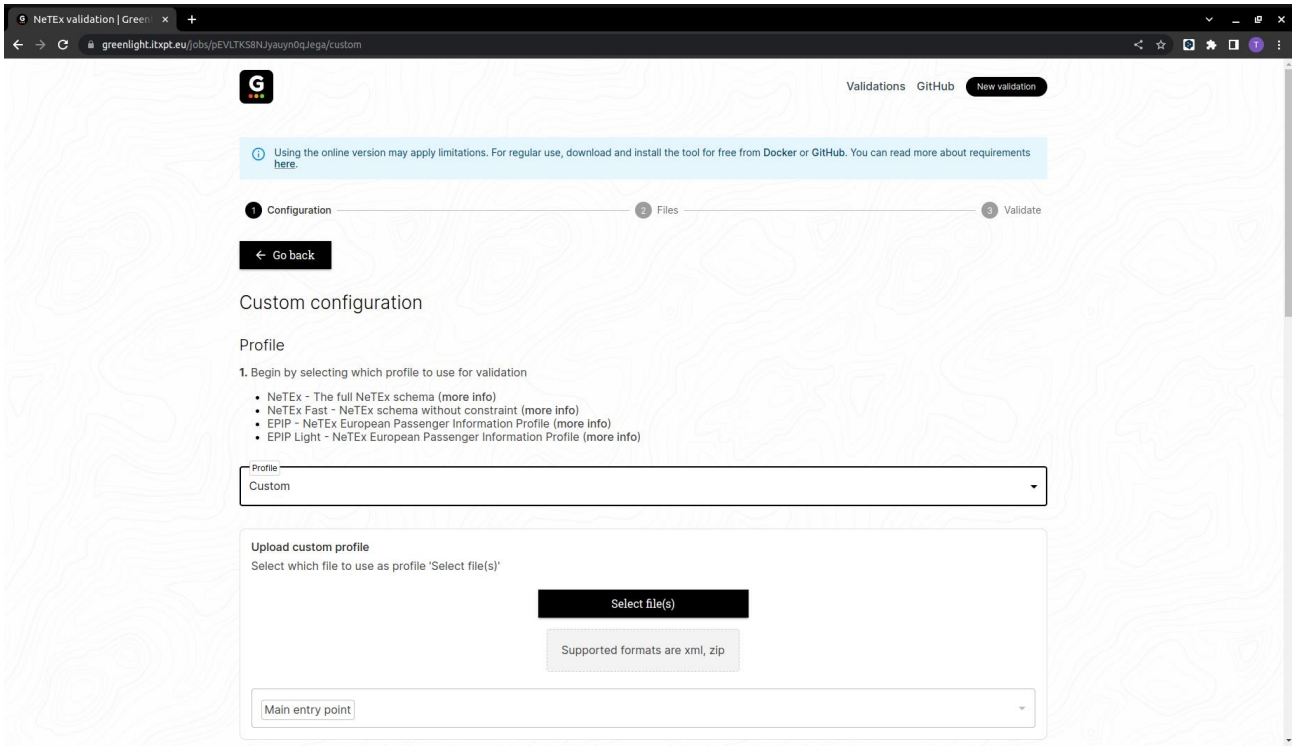

Figure 5 Where to upload a custom profile

2.1.2 IMPROVING PERFORMANCE BY SEPARATING INTEGRITY CROSS-CHECKS FROM XML SCHEMA VALIDATION

Integrity cross-checks require a lot of memory which must create issues in performance when checking multiple files simultaneously or big files (of many GB). Therefore, in the tool the option to perform such checks "outside" XML schema validation has been added by using script coded rules.
In particular the relevant rules are:

- *Validate NeTEx element uniqueness*

- *Make sure NeTEx references have matching keys*

The scripts of these rules are available in builtin folder. Check also *2.3.Build your own* rules
You can change or add your own rules by cloning the Greenlight repository from GitHub and modify one of the scripts in the directory builtin. Save it with a new name and then map the *builtin* folder to the docker container with the Docker parameter -v.

-v c:\code\greenlight\builtin:/usr/local/greenlight/builtin

Use the script in the same way as the ones of the standard scripts with the flag -r and name of the script.

Example -r mymodifiedrule

Check also *0*

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

Manual for command line interface.

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

Source codes inventory.
To apply these rules, choose *NeTEx schemas without constraints (so called **NeTEx Fast/EPIP light** on the web interface, and **NeTEx@1.2-nc, epip@1.1.2-nc.***

How to perform such checks using the Web Interface

In ***Configuration*** page

- Select ***Packages*** → ***NeTEx Fast (v.1.2), all rules*** package,

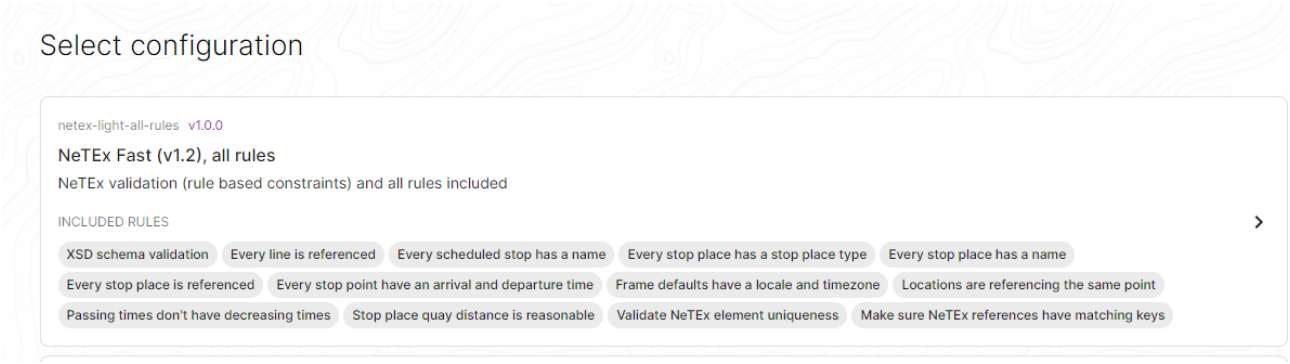

Figure 6 Package selection for *fast* integrity checks using separated scripts.

OR

- Select ***Custom*** → ***Profile*** → ***NeTEx Fast*** OR ***EPIP light***
- Select ***Rules*** →

- Validate NeTEx element uniqueness
- Make sure NeTEx references have matching keys

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

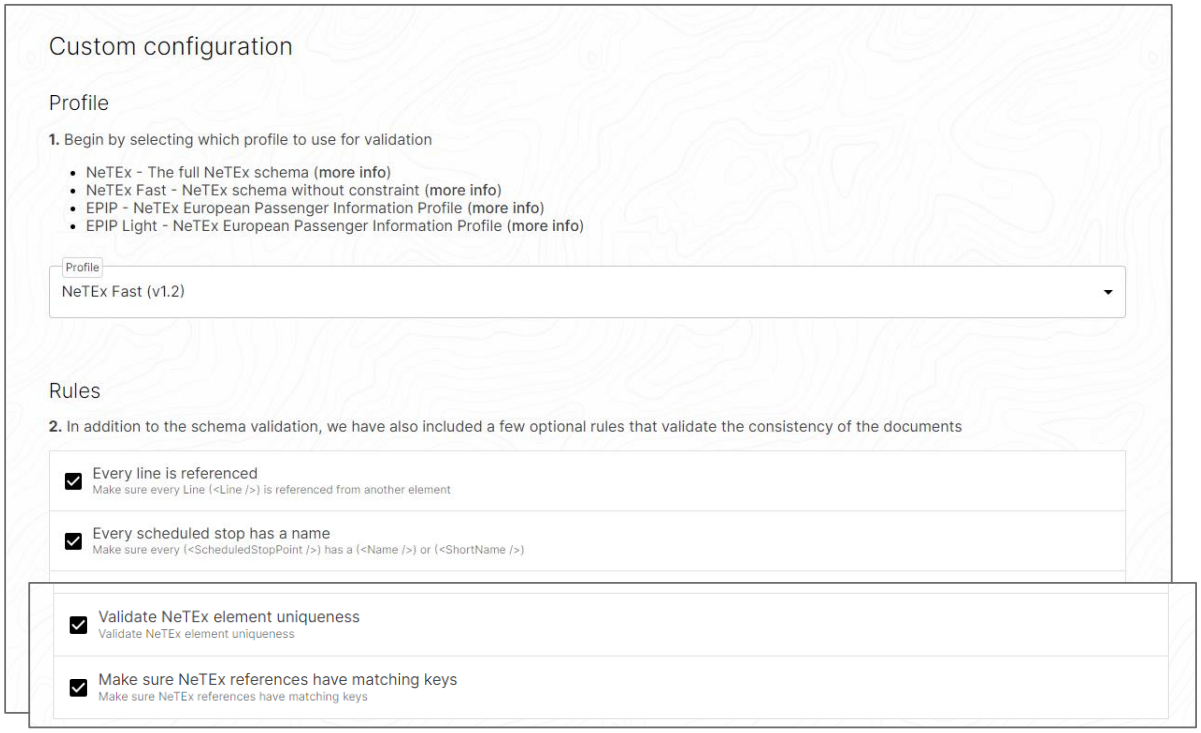

Figure 7 Custom configuration for *fast* integrity checks using separated scripts.

How to perform such checks using the Command Line Interface

When running the docker setup you will have to add the –schema parameter with ***NeTEx@1.2-nc or epip@1.1.2-nc***, this will by default use all the rules available.

```
docker run -it itxpt/greenlight validate -schema epip@1.1.2-nc -i testdata
```

OR setting only the rules with this parameter :

```
-r netexUniqueConstraints,netexKeyRefConstraints
```

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

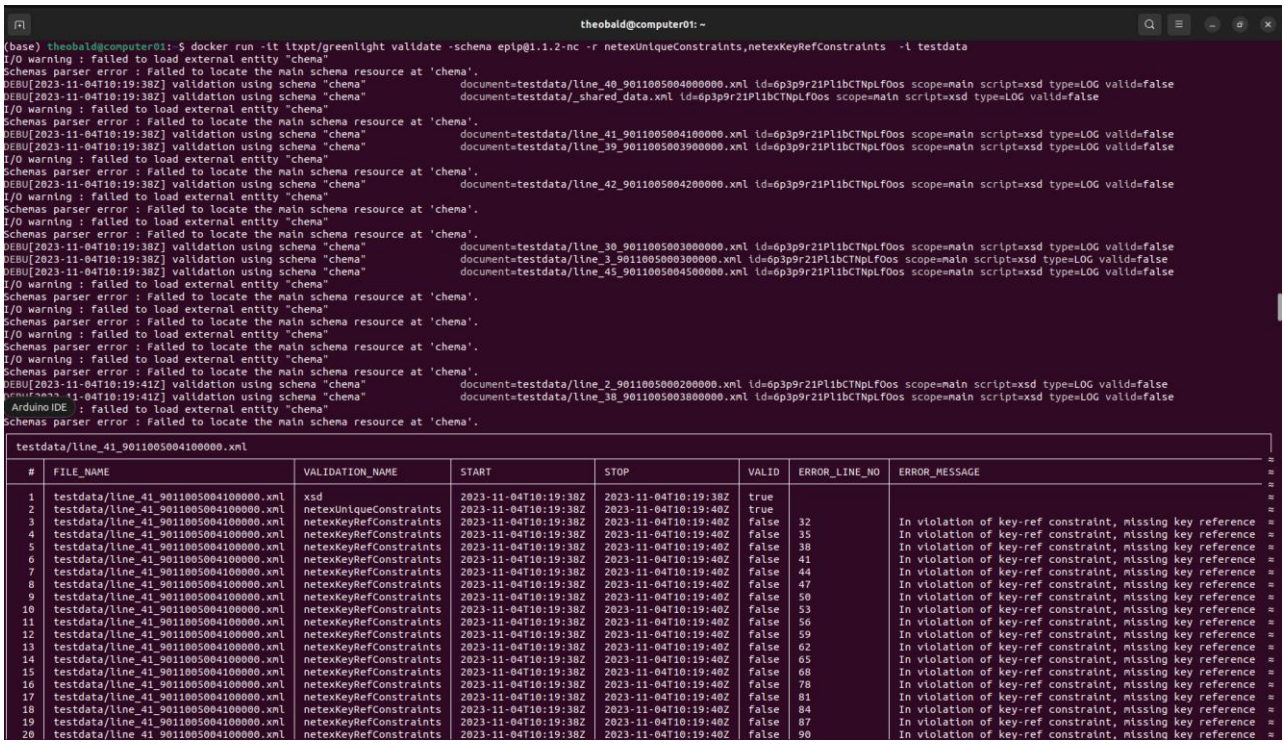

Figure 8 Custom configuration for *fast* integrity with customs rules on CLI

2.2 RULES BEYOND NETEX XML SCHEMA

The rules that cannot express in XML and have been developed and incorporated in the current tool correspond mostly to validity, consistency, and accuracy.

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

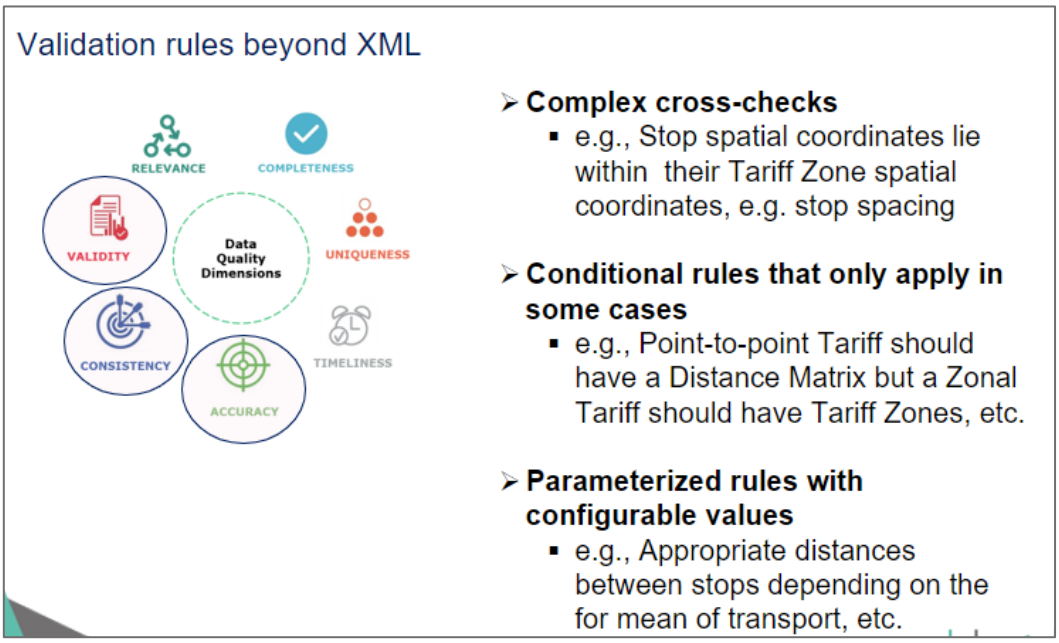

Figure 9 Data Quality dimensions and correspondance of XML schema based validation rules.

Table 5 Examples of rules that cannot express in XML.

Rules category	Examples of rules
Complex cross-checks	E.g. Validity dates of elements fall within validity dates of frame. E.g. Stop spatial coordinates lie within their Tariff Zone spatial coordinates
Conditional rules that only apply in some cases	E.g. Point-to-point Tariff should have a Distance Matrix but a Zonal Tariff should have Tariff Zones, etc., etc., etc
Parameterised rules with configurable values	E.g. Appropriate distances between stops for transport mode. E.g. Appropriate transfer distances to interchange.
Checks against external data sets/ databases.	E.g. Operator codes, spatial coordinates.
Data modularized into multiple XML documents with cross- references.	E.g. Large National data sets.

In total, we have been identified 138 rules potential rules beyond XML schema basics. These rules have been evaluated in terms of priority and concern both general rules and profile specific rules. It is envisaged that this list will the basis for further developments.

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

Table 6 Summary of all identified rules – scripted or not.

Priority level	Number of rules	Topics	Specific to
High	5	Header – versions; Identifiers – Codespaces; Date ranges – Frame; Journey Parts	General schemasou
Mid- high	31	Journey Parts; Frequencies; Timings...	General schema
Medium	52	Stop point; Stop place; Line...	EPIP and General schema
Mid – low	47	Hierarchy; Topographic; Display...	Profile Specific and general schema
Low	3	Unused data; Stop Point; Journey	Profile Specific and EPIP

We currently handle 15 rules that are listed in the following Table 7 Currently integrated rules. In this table you may find the details of each rule together with the link to the corresponding script.

These rules have been chosen considering several criteria (such as importance, effort to be implemented, relevance to the full NeTEx profile and more).

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

Table 7 Currently integrated rules

	Script name	Description	Functional Area	Aspect	Severity (10=high 50=low)	Development details
1	passingTimesIsNotDecreasing	On a VERSION FRAME. ToDate must not be later than FromDate on any date range.	Common Content	Date ranges - Frame	10	Check that *from* date is before *to* date on the VERSION FRAME
2	everyStopPointHaveArrivalAndDepartureTime	Every POINT IN JOURNEY POINT In a JOURNEY PATTERN used by a JOURNEY must have a PASSING TIME with arrival and departure time (except for the first and last stop)	Timetable	Timings	20	Check that an appropriate ArrivalTime and DepartureTime exists in for each PASSING TIME TimetabledPassingTime in a SERVICE JOURNEY.
3	everyStopPlaceHasAName	Every STOP PLACE has a Name or ShortName attribute	Stop	Stop Place	20	Name attribute should be filled in for all STOP PLACEs
4	passingTimesHaveIncreasingTimes	Successive DayOffset+PassingTimes for the POINTs IN JOURNEY Pattern or CALLS of a Journey must not decrease.	Timetable	Timings	20	Successive DayOffset+PassingTimes for the POINTs IN JOURNEY Pattern or CALLS of a Journey must not decrease
5	frameDefaultsHaveALocaleAndTimeZone	The FrameDefaults of a VERSION FRAME should have values appropriate to the content	Common	Frame	30	Depends on Frame type. - For all frames check and DefaultLanguage exists in DefaultLocale in FrameDefaults. - For frames that contain spatial coordinates, check that default LOcationSystem is specified (usually WGS84) - For Frames that contain elements with Timezones. (e.g. STOP PLACEs etc. in . SITE FRAME. SCHEDULED STOP POINT in SERVICE FRAME, Check that Time Zone is specified. - For frames that hold monetary values, e.g. FARE FRAMES or if amount specified. NB can be specified on outmost COMPOSITE FRAME if common to all.

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

	Script name	Description	Functional Area	Aspect	Severity (10=high 50=low)	Development details
6	everyStopPlaceHasACorrectStopPlaceType	Every STOP PLACE has a StopPlaceType attribute with correct value	Stop	Stop Place	30	Each STOP PLACE should have a StopPlaceType attribute. This should match any type on the QUAYs.
7	netexKeyRefConstraints.js	All stop identifiers (QUAY. all STOP PLACEs. GROUPs OF STOP PLACEs and ACCESS) must comply with the profile codification	Stop	Stop Place	30	For Stop specifically [COUNTRY code]: [INSEE common code]: [Type of object]: [Specific stop code]: [Issuer code of the technical code or LOC].
8	locationsAreReferencingTheSamePoint	SCHEDULED STOP POINT must have similar spatial coordinates to those of the assigned STOP PLACE	Stop	Stop Point	30	Should be with a certain tolerance of distance, varying by mode. . Will not necessarily be the same centroid.
9	stopPlaceQuayDistanceIsReasonable	Distance Between QUAY and STOP PLACE too long	Stop	Spatial	30	Distance between QUAY and STOP PLACE should not be too far apart. STOP PLACE location is centroid of station. QUAY is centroid of QUAY.
10	locationsAreReferencingTheSamePoint	The location of QUAY and SCHEDULED STOP POINT should be within reasonable distance of the location or surface of STOP PLACE	Stop	Stop place	30	Take the positions from the QUAY and SCHEDULED STOP POINT and calculate the distance in meters. Hard code 500m in first version. later add parameter to set distance

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

	Script name	Description	Functional Area	Aspect	Severity (10=high 50=low)	Development details
11	everyScheduledStopPointHasAName	A SCHEDULED STOP POINT must have an instantiated Name field	Stop	Stop Point	30	The name of a stop should be given. Applies s to Both STOP PLACE and SCHEDULE D STOP POINT
12	everyLineIsReferenced	A LINE must have one or more ROUTE instances	Timetable	Line	40	Check that a LINE is referenced in at least one ROUTE
13	everyStopPointIsReferenced	Any SCHEDULED STOP POINT that is declared should be used. i.e. referenced by an assignment or POINT IN PATTERN etc.	Timetable	Unused data	40	Check that each SCHEDULED STOP POINT is used in one or more JOURNEY PATTERNs.
14	everyStopPlaceIsReferenced	Any STOP PLACE that is declared should be referenced by a STOP ASSIGNMENT	Stop	Unused data	50	Every STOP PLACE should be referenced in at least one STOP ASSIGNMENT. Depends on the profile.
15	locationsAreReferencingTheSamePoint	SCHEDULED STOP POINT must be assigned to a STOP PLACE	Stop	Stop Point	50	Every SCHEDULED STOP POINT should be referenced in at least one STOP ASSIGNMENT. Depends on the profile.

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

2.3 BUILD YOUR OWN RULES

You can change or add your own rules by cloning the Greenlight repository from GitHub and modify one of the scripts in the directory builtin. Save it with a new name and then map the *builtin* folder to the docker container with the Docker parameter -v.

-v c:\code\greenlight\builtin:/usr/local/greenlight/builtin

Use the script in the same way as the ones of the standard scripts with the flag -r and name of the script.

Example -r mymodifiedrule

Check also *0*

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

Manual for command line interface.

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

3 SOURCE CODES INVENTORY

This table provides a comprehensive inventory of the assets and components within GitHub repository. This is not a detailed inventory, but it will help you find your way around in the repository.

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

Table 108 Source codes Inventory.

Folder/File	Definition	Folder/File	Definition	File	Definition
app	Web interface: next.config to use static file	**api**	Define how the client can communicate with the application (main script client.ts and type definition in types.ts)		
		hooks	handle different component	useApiClient	allow the API
				useEmail	enable the use of email
				useFirebase	enable the use of email
				useWebConfig	set up Webconfig using firebase
		components	all the script that are used (example FileUpload handle the uploading, the name are self-explanatory)		
		pages	pages hardcoded	Jobs	validation views -> custom
		public	icon css style for the pages		
		styles	css style for the pages		
builtin	Actual scripts for validation rules coded in javascript (their names are quite self-explanatory). Look at the rules part for more details.				
cmd	Command line tool	**validate.go**	Is the main script		
		profile.go	Handle the profile type and setup		
		root.go	Setup the root github folder needed		
		file.go	Used to handle different file and clean temporary one		
xml-stream-parser	External library to find path in xml				
internal	Add intern feature to improve efficiency the tool				
js	Transform javascript feature into understandable feature in GO programming language				
media	Media used in the readme file				
testdata	Test xml data set				

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

Folder/File	Definition	Folder/File	Definition	File	Definition
xml	Link library libxml2 to .go and create new bindings				
xsd	NeTEx schemas from NeTEx GitHub				
validation.go	Hold the functions to validate a file				
result.go	Hold the functions to generate the result				
Makefile	Details of the command line				
cliff.toml	Generate changelog file				
CHANGELOG .md	Log with all changes performed in the GitHub folder				
README.md	The README file where you can find more info on the tool and how to use it				

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

4 MANUAL FOR WEB INTERFACE

After the container has started you can use the web interface by opening a web browser and type the address http://localhost:8080/ or you can access the online tool at the address https://greenlight.itxpt.eu/. Then you can click on Start validating to start a new validation session. You can also always use the New validation button in the upper right corner to start over with a new validation.

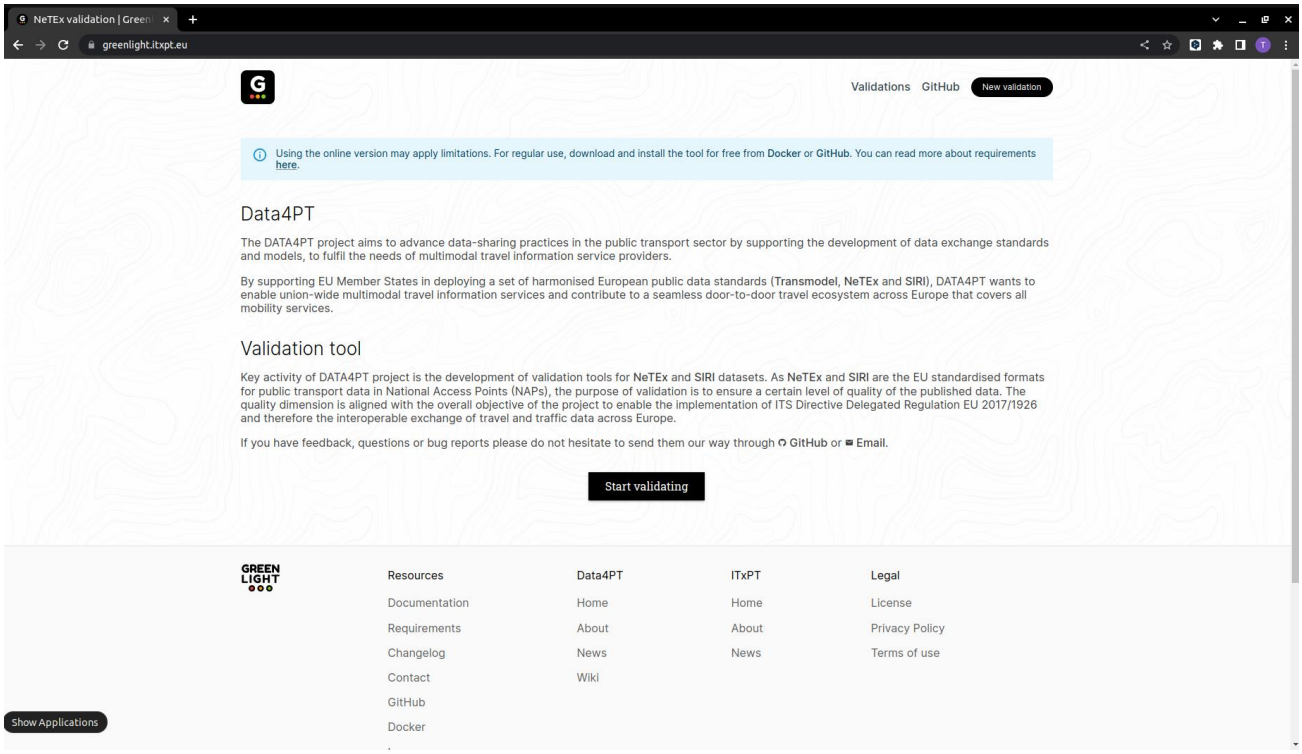

Figure 10 Validation tool Greenlight website

4.1 NAVIGATION

At the top of the web page is a menu bar and a progress indicator. The logo to the left always take you to the start page. Validations in the menu will show recent done validations, GitHub will take you to our page with documentation and the source code and New validation will start over with a new validation. You can also use Go back to navigate to a previous step.

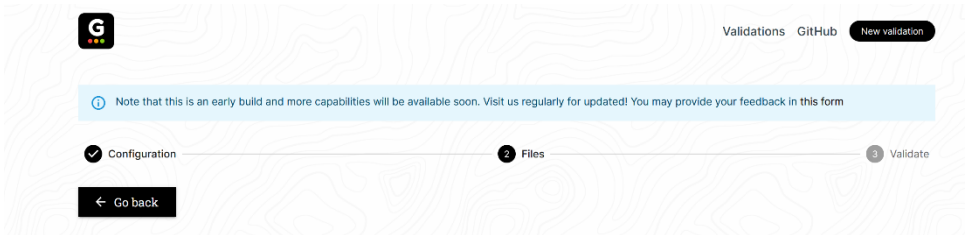

Figure 11 All steps for a validation

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

4.2 CONFIGURATION

To start a validation, you first decide if you want to use a premade configuration package or use a custom configuration. The packages are predefined with schemas and rules that are commonly used together. To select your own combination of schemas and rules you can do a custom configuration.

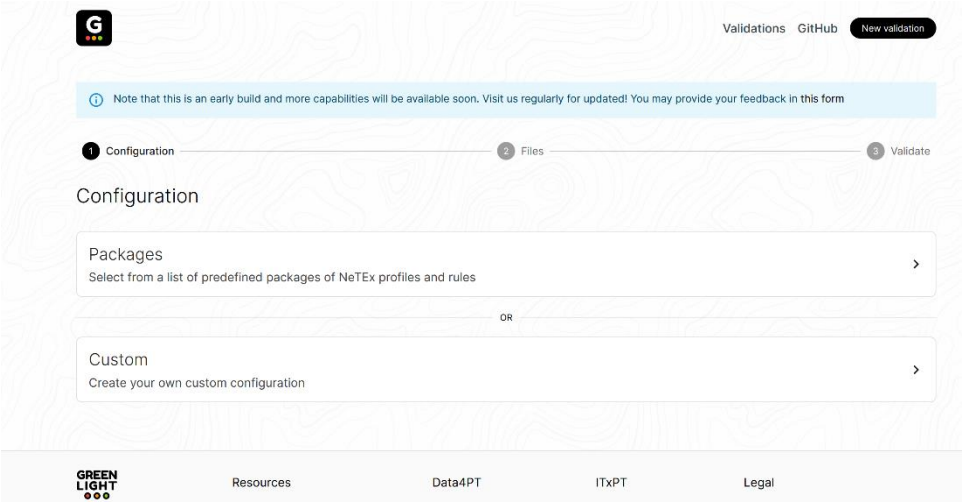

Figure 12 Configuration page

4.3 PACKAGES

If you select to use the premade packages you are presented with a list to select from. Select the one that works best with your validation requirements. When you click on one of the packages you continue to the selection of files.

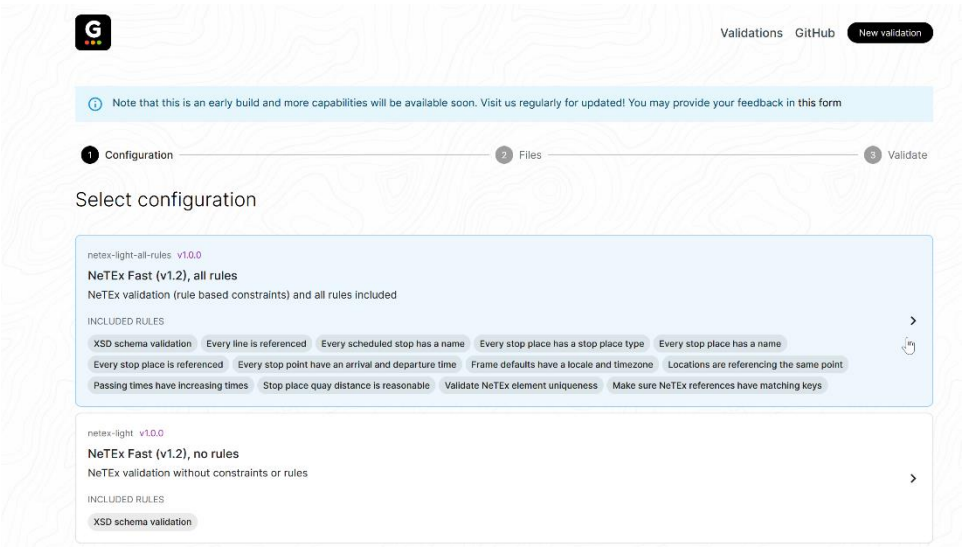

Figure 13 Premade package available on the website

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

4.4 CUSTOM CONFIGURATION

With the custom configuration you can be more detailed in which NeTEx Profile and combination of rules to use. In the list of rules, you get brief description of each rule. Zero or more rules can be selected by clicking the checkbox for each rule.

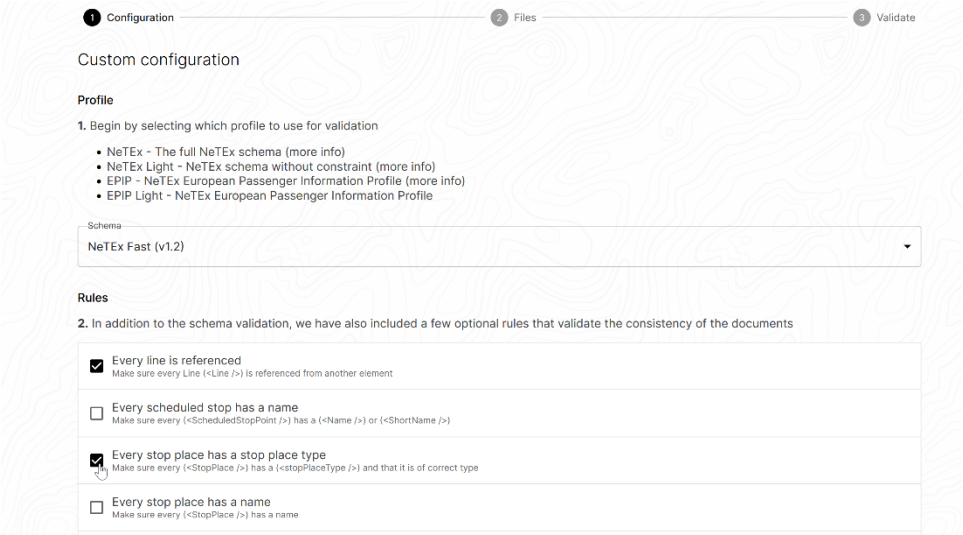

Figure 14 Custom configuration page

Some rules use parameters as input to the validation. Those rules have a default value that can be changed by clicking on the Configure icon to the right.

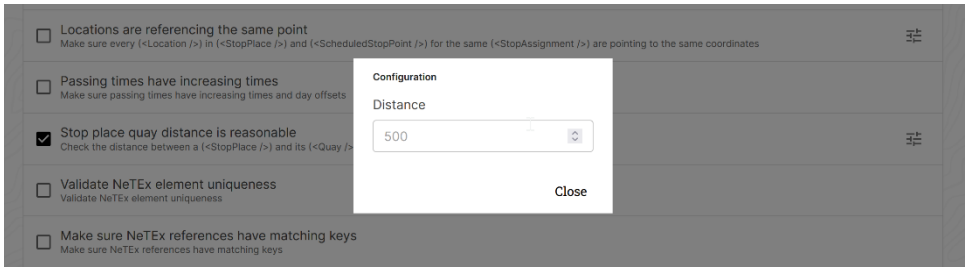

Figure 15 Setting up parameter of Stop place quay distance rule

4.5 SELECT FILES TO VALIDATE

The last step is to upload the files to be validated, it can be single files or multiple files compressed in an archive. Click Select file(s) to select which files to upload and then wait until all files has been uploaded, see the Status indicator in the files list.

If you want to get back to the selection of rules you can use the Go back button

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

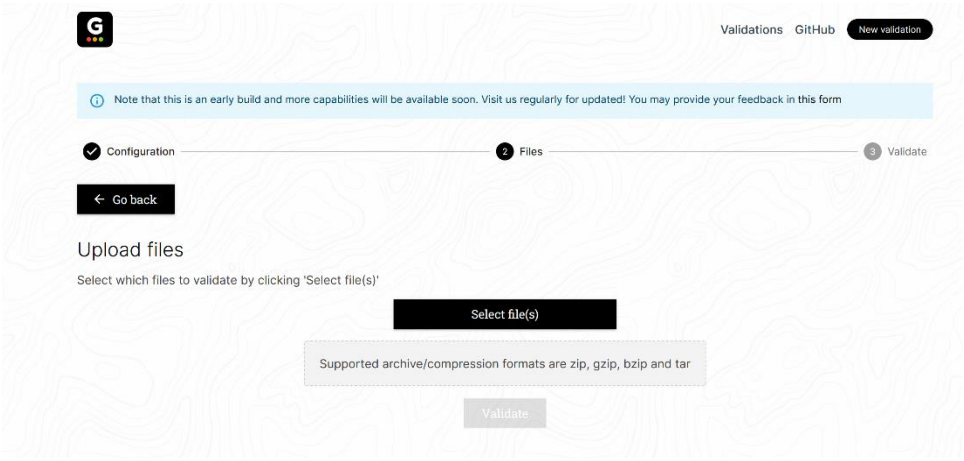

Figure 16 Upload validation file page

When all files are uploaded you start the validation by clicking on Validate.

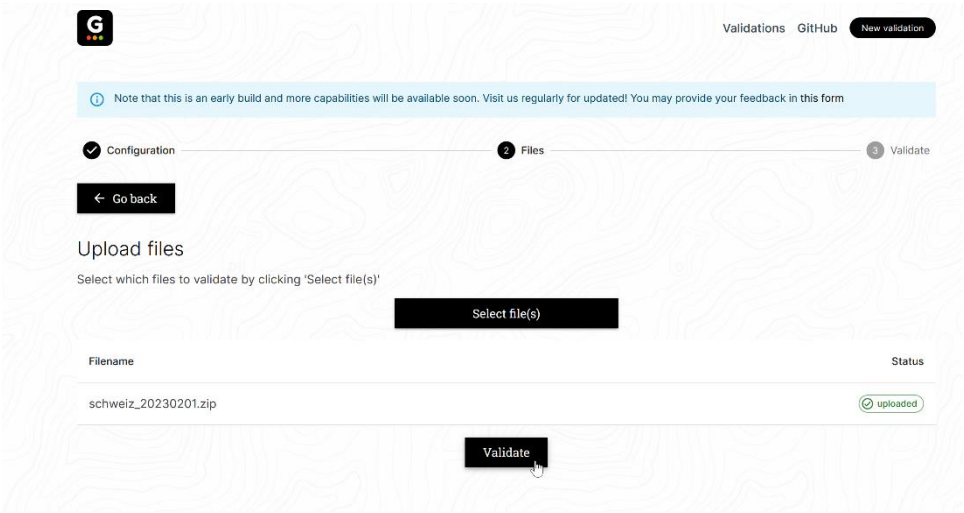

Figure 17 Validation of the upload files

Each file is validated against the selected schema and rules, all validations run in parallel. Depending on the number of files and their sizes the validation can take some time to complete.

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

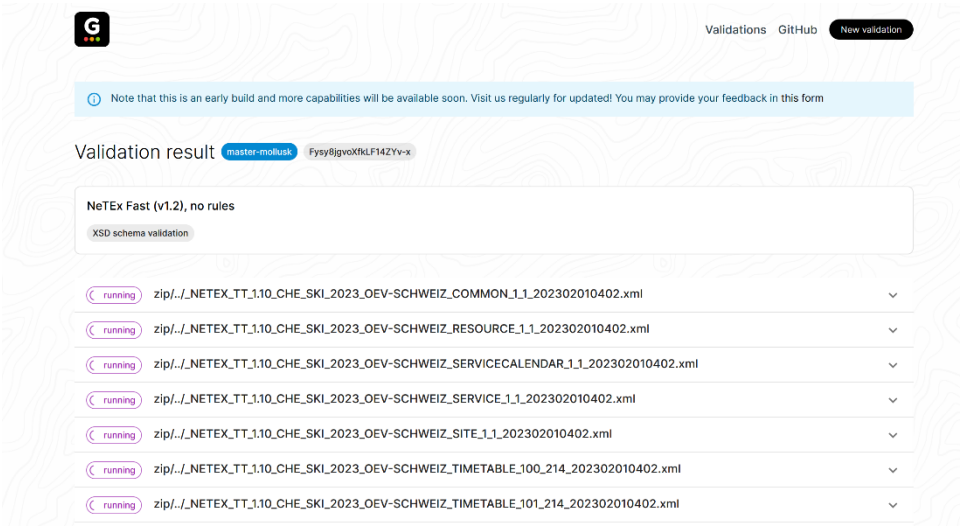

Figure 18 Running validation page

4.6 VALIDATION RESULT

When the validation is done you get an overview of the result. You can see the status of the validation for each file. If there are any errors, you can get all the details by clicking on the down arrow to the right of each file.

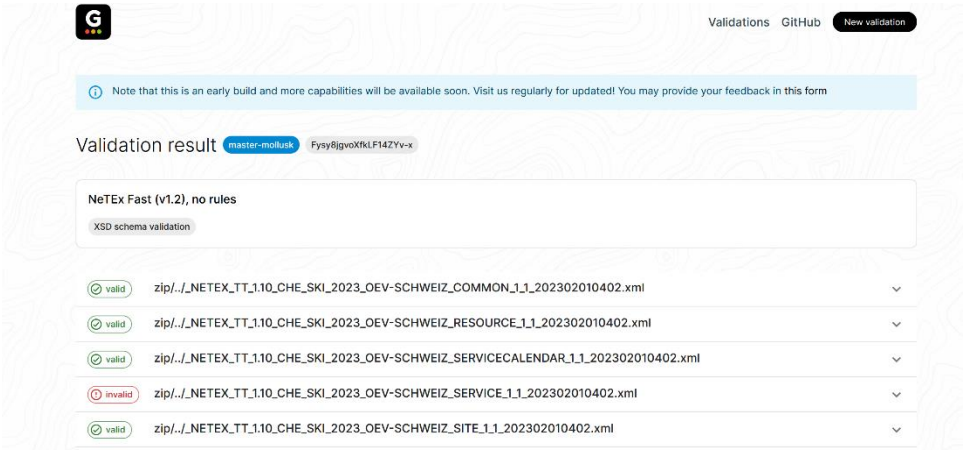

Figure 19 Result page

The details display the number of times that specific error occurs in the file, and you can page between them with the arrows to the right. For each error you get information about the type, line number in the file and a more detailed explanation.

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

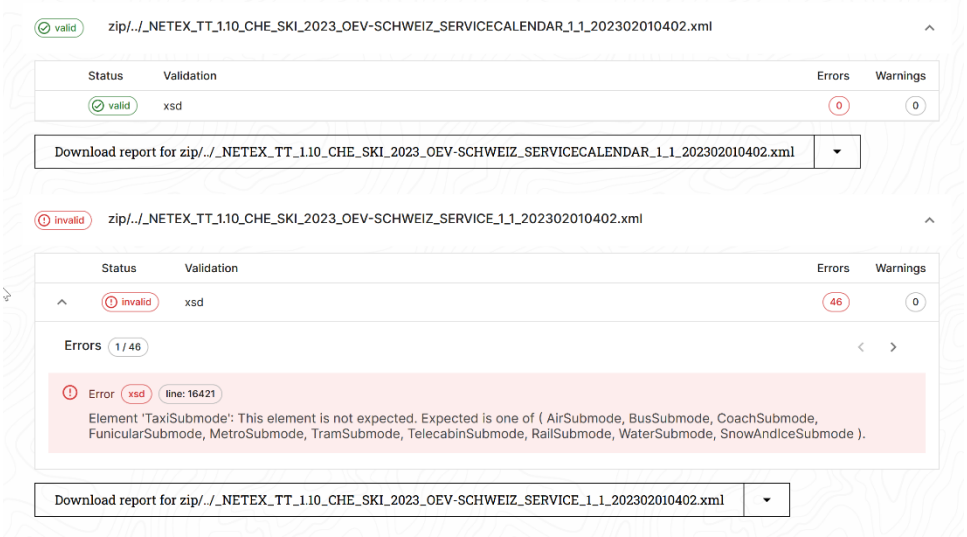

Figure 20 Details of the result on the website

4.7 DOWNLOADING THE RESULT

You can download the result for each error or the complete validation to a file in json or csv format to process it further. For example, to give as documentation to someone who can correct the error

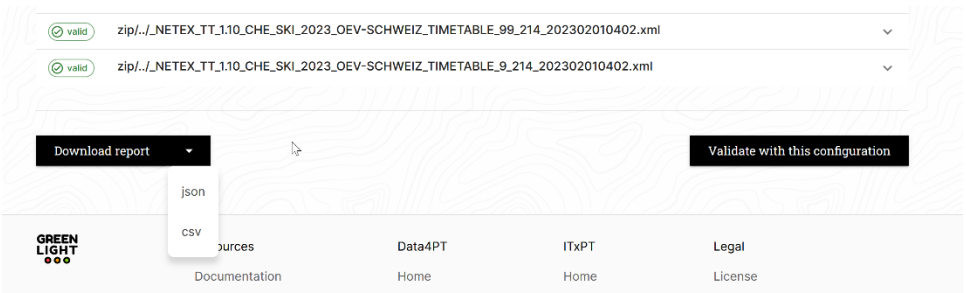

Figure 21 Downloading result as json or csv

Example of the saved data in json format.

Figure 22 Example of data saved as json

4.8 PREVIOUS VALIDATION

At the bottom of the result page, you have an option Validate with this configuration to go back and start a new validation with the same configuration but with new files. You can also see recent validation by using the menu selection Validations at the to of the page. By clicking on the name of a job you will see the result page for that validation again.

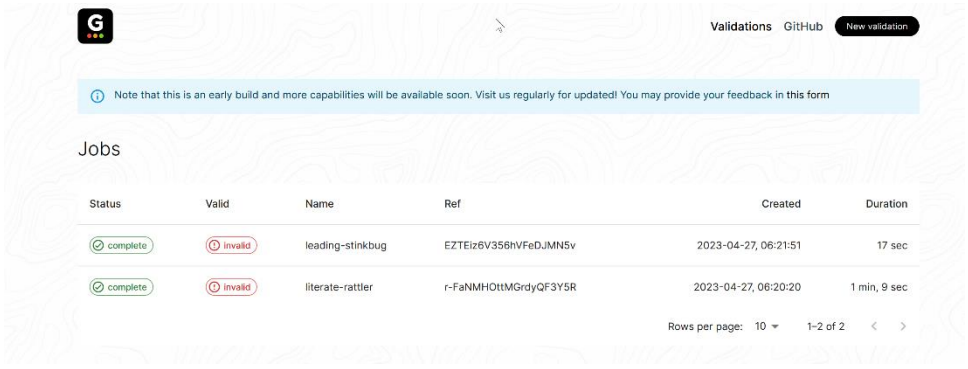

Figure 23 Validation history page

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

4.9 TECHNICAL ERROR MESSAGES

Sometimes the web interface will show error messages if the Greenlight tool stops to execute. Often that occurs when the communication to the web server is lost, or the local Docker version has stopped. Check the status of your connection and that the Docker container is running if using it locally.

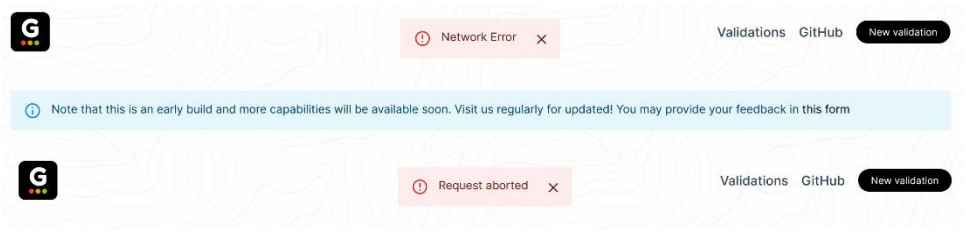

Figure 24 Example of Error messages

5 MANUAL FOR COMMAND LINE INTERFACE

The CLI (command line interface) is for more advanced use cases where you want more control over the validation or if you want to include the validation in your own pipeline. An example could be to recieve a file via an integration, validate the file with GreenLight and if there are any errors inform via email and otherwise save the file for use in another system.

To use the CLI you must first download the Docker image as described in Getting started

When you use the CLI you first give the command docker and the parameters run -it [docker_image] in this case the docker_image is itxpt/greenlight. After that you give the different commands and flags to greenlight, e.g., help. If you want to use other docker parameters, you have to put them before the name of the image to use. See below for more complex examples of how to invoke the greenlight command.

5.1 GETTING HELP

The tool has a built in help system that gives explanations of all commands and parameters in the tool. Use the command below to get an overview of the help you can get.

```
docker run -it itxpt/greenlight help
```

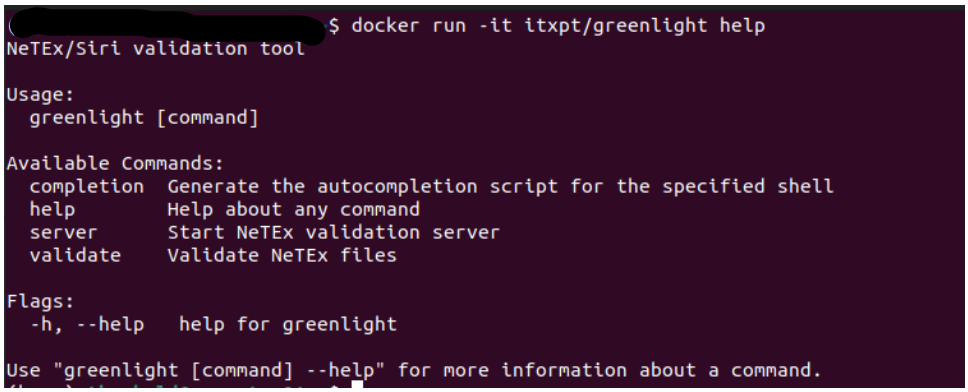

Figure 25 Help command on the Command Line Interface

5.2 SERVER COMMAND

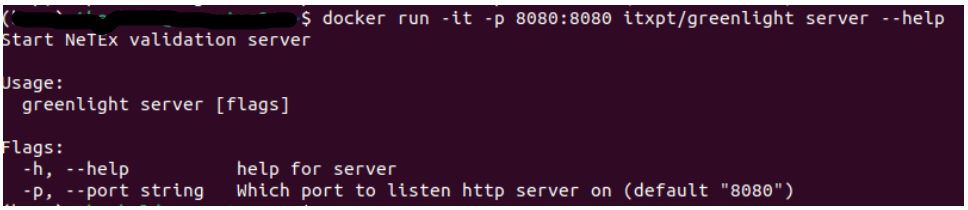

Figure 26 Command to build the web interface

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

```
docker run -it -p 8080:8080 itxpt/greenlight server
```

This will start the built-in web interface and it can be accessed via http://localhost:8080/. Se Web Interface for a guide on how to use it.

5.3 VALIDATE COMMAND

A validation is started with the command Validate, it uses the following flags as input to configure the validation.

```
Flags:
  -h, --help                help for validate
  -i, --input string        XML file, dir or archive to validate
  -l, --log-level string    Set level of log output (one of "trace", "debug", "info",
"warn", "error") (default "debug")
  -o, --output string       Set which output format to use (one of "json", "xml",
"csv", "pretty" (default "pretty")
  -p, --profile string      Set path of validation profile (note: flags 'rules' and
'schema' is ignored)
  -r, --rules strings       Set which validation rules to run (defaults to all inside
the builtin dir)
  -s, --schema string       Which xsd schema to use (supported "NeTEx@1.2",
"NeTEx@1.2-nc", "epip@1.1.2", "epip@1.1.2-nc") (default "NeTEx@1.2-nc")
      --silent              Running in silent will only output the result in a boolean
fashion
```

To verify that the tool works yo can do a validation with a NeTEx file provided with the tool.

```
docker run -it itxpt/greenlight validate -i testdata
```

.

Figure 27 Example of output from a validation done in the CLI

5.4 NETEX PROFILE

To select NeTEx profile use the flag -s or --schema and the name of the profile. Valid names are NeTEx@1.2, NeTEx@1.2-nc, epip@1.1.2, epip@1.1.2-nc. If no schema is selected the NeTEx@1.2-nc is used. -nc at the

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

end means that the validation is with No Constraints. Which is a faster validation but needs that the no-constraints rule is used instead.

Example of how to use the EPIP schema when validating the built in test file

```
docker run -it itxpt/greenlight validate -schema epip@1.1.2 -i testdata
```

5.5 RULES

Select which rules to use with the flag -r or --rules and then give the name of the rules to use. Several rules can be specified by separating them with a comma.

Example

```
-r everyLineIsReferenced,everyScheduledStopPointHasAName
```

```
(base) theobald@computer01:~$ docker run -it itxpt/greenlight validate -i testdata -r everyLineIsReferenced,everyScheduledStopPointHasAName
DEBU[2023-11-02T17:22:12Z] validation using schema "netex@1.2-nc"        document=testdata/line_38_9011005003800000.xml id=cjAAQzrwOzOehe2CcMpby scope=main script=xsd type=LOG valid=false
DEBU[2023-11-02T17:22:12Z] validation using schema "netex@1.2-nc"        document=testdata/line_39_9011005003900000.xml id=cjAAQzrwOzOehe2CcMpby scope=main script=xsd type=LOG valid=false
DEBU[2023-11-02T17:22:12Z] validation using schema "netex@1.2-nc"        document=testdata/line_3_9011005003000000.xml id=cjAAQzrwOzOehe2CcMpby scope=main script=xsd type=LOG valid=false
DEBU[2023-11-02T17:22:12Z] validation using schema "netex@1.2-nc"        document=testdata/line_45_9011005004500000.xml id=cjAAQzrwOzOehe2CcMpby scope=main script=xsd type=LOG valid=false
DEBU[2023-11-02T17:22:12Z] validation using schema "netex@1.2-nc"        document=testdata/line_41_9011005004100000.xml id=cjAAQzrwOzOehe2CcMpby scope=main script=xsd type=LOG valid=false
DEBU[2023-11-02T17:22:12Z] validation using schema "netex@1.2-nc"        document=testdata/line_40_9011005004000000.xml id=cjAAQzrwOzOehe2CcMpby scope=main script=xsd type=LOG valid=false
DEBU[2023-11-02T17:22:12Z] validation using schema "netex@1.2-nc"        document=testdata/line_2_9011005000200000.xml id=cjAAQzrwOzOehe2CcMpby scope=main script=xsd type=LOG valid=false
DEBU[2023-11-02T17:22:12Z] validation using schema "netex@1.2-nc"        document=testdata/line_42_9011005004200000.xml id=cjAAQzrwOzOehe2CcMpby scope=main script=xsd type=LOG valid=false
DEBU[2023-11-02T17:22:28Z] validation using schema "netex@1.2-nc"        document=testdata/_shared_data.xml id=cjAAQzrwOzOehe2CcMpby scope=main script=xsd type=LOG valid=false
DEBU[2023-11-02T17:22:28Z] validation using schema "netex@1.2-nc"        document=testdata/line_30_9011005003000000.xml id=cjAAQzrwOzOehe2CcMpby scope=main script=xsd type=LOG valid=false
testdata/line_38_9011005003800000.xml
| # | FILE_NAME | VALIDATION_NAME | START | STOP | VALID | ERROR_LINE_NO | ERROR_MESSAGE |
|---|---|---|---|---|---|---|---|
| 1 | testdata/line_38_9011005003800000.xml | everyLineIsReferenced | 2023-11-02T17:22:12Z | 2023-11-02T17:22:12Z | true | | |
| 2 | testdata/line_38_9011005003800000.xml | everyScheduledStopPointHasAName | 2023-11-02T17:22:12Z | 2023-11-02T17:22:12Z | true | | |
| 3 | testdata/line_38_9011005003800000.xml | xsd | 2023-11-02T17:22:12Z | 2023-11-02T17:22:28Z | true | | |
| | | VALID | TRUE | | | | |
```

Figure 28 Command line example of rule selection

You can change or add your own rules by cloning the greenlight repo from GitHub and modify one of the scripts in the directory builtin. Save it with a new name and then map the builtin folder to the docker container with the Docker parameter -v.

```
-v c:\code\greenlight\builtin:/usr/local/greenlight/builtin
```

Use the script in the same way as one of the standard scripts with the flag -r and name of the script.

Example

```
-r mymodifiedrule
```

5.6 PROVIDING FILES

The files to test can be single files, a folder with files or a compressed archive with files. Put the files to be tested in a local folder and use the docker parameter -v to map it with a folder in the greenlight container.

```
-v C:\code\NeTEx\testdata:/usr/local/greenlight/testdata
```

Then you can use the greenlight flag -i to include the files in the validation

Command to validate a folder with files

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

```
docker run -it -v c:\code\NeTEx\testfiles:/usr/local/greenlight/testdata
itxpt/greenlight validate -s NeTEx@1.2-nc -r everyLineIsReferenced -i testdata
```

Command to validate an archive with several files

```
docker run -it -v c:\code\NeTEx\testfiles:/usr/local/greenlight/testdata
itxpt/greenlight validate -s NeTEx@1.2-nc -r everyLineIsReferenced -i
testdata/xt_2023_04_15.zip
```

Command to validate a single file

```
docker run -it -v c:\code\NeTEx\testfiles:/usr/local/greenlight/testdata
itxpt/greenlight validate -s NeTEx@1.2-nc -r everyLineIsReferenced -i
testdata/line_2_9011005000200000.xml
```

5.7 OUTPUT

The result of the validation can be presented in different formats. For example, the pretty will give an output adopted to be read on the screen. The other formats json, xml and csv can be used to pipe the output to a file for further processing.

```
docker run -it itxpt/greenlight validate -s NeTEx@1.2-nc -r everyLineIsReferenced
-i testdata -o json > greenlight-result.json
```

5.8 COMPLETION COMMAND

Generate an autocompletion script for Greenlight for different shells. The generated script can be added to your shell profile. Scripts can be generated for bash, fish, zsh and powershell.

Note: This command is for power users who uses the CLI a lot and want to make it easier and faster to type commands and parameters.

As an example, will the command below generate a script for bash

```
docker run -it itxpt/greenlight completion bash
```

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

6 BUILDING FROM SOURCE

6.1 PREREQUISITES

Greenlight is using Go and is powered by libxml2, so make sure those are installed first. If you want to work on the web interface, you will also need Node.js

Go

Download and install the latest version with standard settings

libxml2

Install using

- Mac: brew install libxml2
- Linux: sudo apt install libxml2
- Windows: Build from source or download precompiled binaries

nodejs - only required for the web interface

Download and install the latest version with standard settings

6.2 GETTING STARTED

Open a terminal and navigate to the folder where you want to install the source code.

```
cd /home/developer/code
```

Clone repository

git clone https://github.com/ITxPT/DATA4PTTools

Navigate to project

```
cd DATA4PTTools
```

Downloading dependencies

```
go get
```

6.3 BUILDING AND RUNNING THE CLI

With the source code and all dependencies downloaded you can try the tool with build in test files to verify that all is working

Validate with demo files provided in the source

path definition will differ running on windows

```
go run cmd/*.go validate -i testdata
```

You can now start validating your own files by providing the path to your document

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

Validate using your own files

path definition will differ running on windows

```
go run cmd/*.go validate -i /path/to/documents
```

That is all that is needed to start using the tool and to be able to modify the core or work with your own validation scripts (you find the scripts in the folder builtin).

6.4 BUILDING THE WEB GUI

When running the web interface the core tool and the interface are started as two separate servers. The backend server is hosting the core tool and provides the functionallity for the validation itself. The interface is then started in a web server and calls the backend when needed.

1. Open a terminal and navigate to the DATA4PTTools directory

   ```
   cd /home/developer/code/DATA4PTTools
   ```

2. Start the backend server

   ```
   go run cmd/*.go server
   ```

3. Then open a new terminal and navigate to he DATA4PTTools directory again

   ```
   cd /home/developer/code/DATA4PTTools
   ```

4. Set current configuration for backend server

Remember to update with the current path to the backend server.

```
echo "NEXT_PUBLIC_API_URL=http://localhost:8080" > app/.env.local
```

5. Navigate to the web app directory

   ```
   cd app
   ```

6. Install dependencies for the web server

   ```
   npm i
   ```

7. Start the web server

   ```
   npm run dev
   ```

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

8. Open the web interface

Open a browser and navigate to http://localhost:3000 and you will see the web interface of Greenlight.

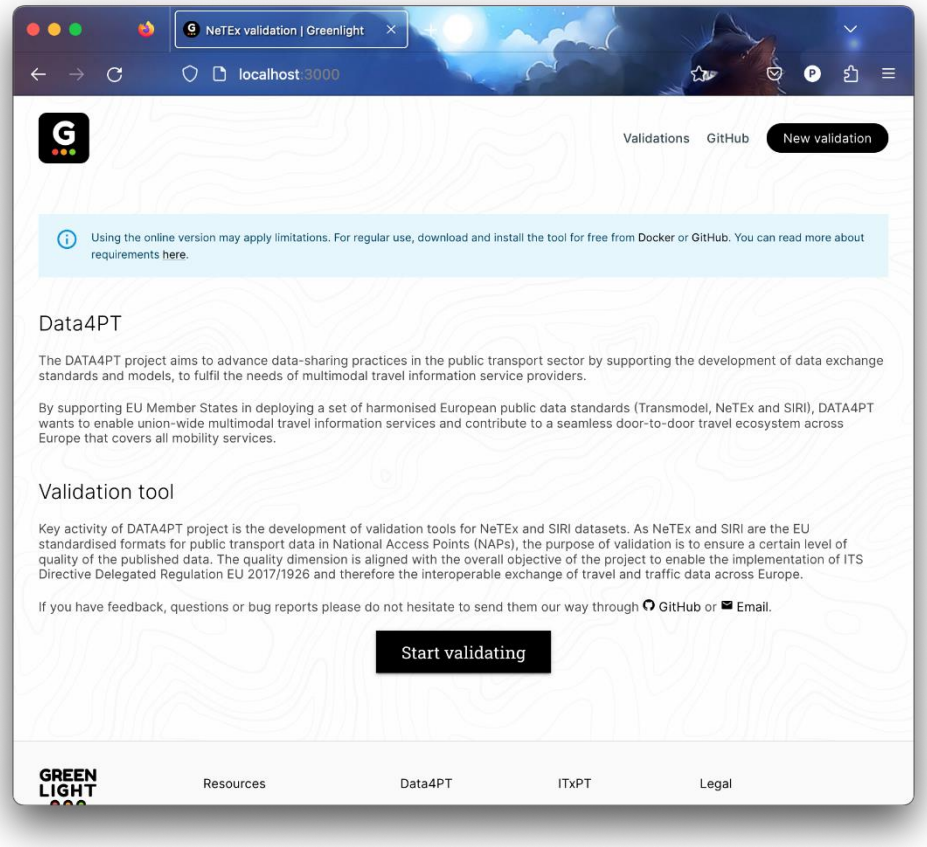

Figure 29 Web interface built from source

To verify that the web interface has contact with the backend server, you may look at the API Status in the lower right corner of the web page.

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

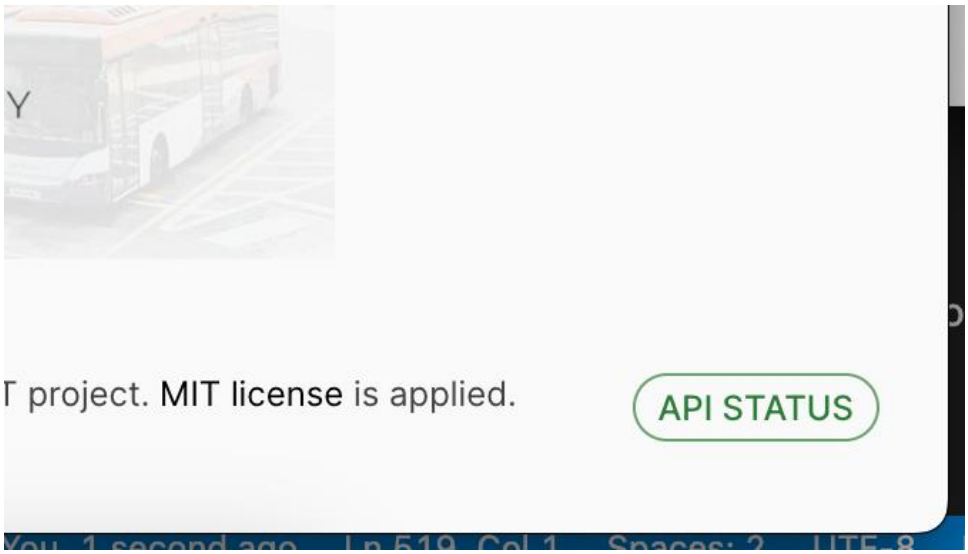

Figure 30 API status on web interface built from source

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

CONCLUSIONS

The Greenlight NeTEx validator aims to support in priority data producers and data consumers that need to check the quality of individual NeTEx files or a set of files to be used as basis for their implementations. It is also addressed to National Access Point (NAP) operators, as an additional asset to ensure quality of the published data. The web user interface does not require technical skills and therefore it is easy to get a quick overview of the status of the datasets. Moreover, the open-source character of the tool offers the possibility to be extended and adapted to specific local and/or national needs. Thus, guidelines on building from source are also available.

Nevertheless, further development is required in the field of additional validation rules to address the specificities of the existing and up-coming European minimum profiles (EPIP, EPIAP, Fares etc.). Defining technical specifications for a collaboratively created library of rules is one of the next steps to increase the added value of the tool. Finally, the improvement of performance is also under consideration aiming at the integration of the tool in platforms where large amount of data is processed (for example in NAP platforms).

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

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