Contract No. H2020 – 826172

SEMANTICS FOR PERFORMANT AND SCALABLE INTEROPERABILITY OF MULTIMODAL TRANSPORT

D2.4 Recommendation to IP4 – intermediate version

Due date of deliverable: 30/06/2020

Actual submission date: 16/12/2020

Leader/Responsible of this Deliverable: UITP

Reviewed: Y

Document status		
Revision	**Date**	**Description**
0.1	09/03/2020	Table of Content
1.0	02/06/2020	First version
1.1	08/07/2020	Second version
1.2	05/11/2020	Revision draft
1.3	11/12/2020	Final version
1.4	16/12/2020	Revised version following JU comments and TMC approval

Project funded from the European Union's Horizon 2020 research and innovation programme		
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: 01/12/2018 Duration: 27 months

EXECUTIVE SUMMARY

This report is the first output of SPRINT Task 2.4 concerning the recommendation to IP4 after the validation of the C-REL proofs-of-concept in WP5. The information in this deliverable is based on the results of SPRINT Task 2.1, Task 2.2, Task 2.3, WP3 and WP4 and provide recommendations on future developments and deployment of the IF to S2R IP4.

Section 2 delivers recommendations based on analysis of requirements of S2R IP4 projects and related EU initiatives. Section 3 is focused on recommendations for the IF architectural design. Recommendations for the IF architecture as a component to NAP are presented in Section 4. Section 5 provides recommendations for performance and scalability of the IF, including lessons learned from C-REL implementation and validations. Recommendations for the IP4 IF semantic automation are delivered in Section 6. Finally, recommendations related to the market uptake can be found in Section 7.

This report will be updated at the end of the project with the results of validation of the F-REL proofs-of-concept in WP5 and the final recommendations to IP4 will be provided.

ABBREVIATIONS AND ACRONYMS

Abbreviation	Description
BPMN	Business Process Model and Notation
CMMP	Contractual Management Market Place
EIF	European Interoperability Framework
FSM	Full Service Model
GOF4R	Governance of the IF for Rail and Intermodal Mobility
IF	Interoperability Framework
IP4	Innovation Programme 4
MaaS	Mobility As a Service
NAP	National Access Point
S2R	Shift2Rail
SHACL	Shapes Constraint Language
ST4RT	Semantic Transformations for Rail Transportation
TSP	Transport Service Provider
UML	Unified Modeling Language
XML	Extensible Markup Language
W3C	World Wide Web Consortium
WP	Work Package

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

This report is the first output of SPRINT Task 2.4 concerning the recommendation to IP4 after the validation of the C-REL proofs-of-concept in WP5. The information in this deliverable is based on the results of SPRINT Task 2.1, Task 2.2, Task 2.3, WP3 and WP4 and provide recommendations on future developments and deployment of the IF to S2R IP4.

The report defines recommendations for IP4 to support the market uptake by

(i) Providing solutions satisfying requirements of both IP4 members and external EU initiatives;
(ii) Simplifying/automating all the necessary steps which are needed to integrate new services and sub-systems in the multi-modal transport ecosystem;
(iii) Emphasizing the potential role of the IF for NAPs;
(iv) Taking care of improving performance and scalability of the IF following its development and deployment.

This report will be updated at the end of the project with the results of validation of the F-REL proofs-of-concept in WP5 and the final recommendations to IP4 will be provided.

2. RECOMMENDATIONS BASED ON ANALYSIS OF REQUIREMENTS OF S2R IP4 PROJECTS AND RELATED EU INITIATIVES

Based on the analysis of requirements of S2R IP4 projects (CONNECTIVE and ATTRACkTIVE) and different EU initiatives in SPRINT D2.1, the following recommendations to the IF can be proposed:

- **Leveraging automation**

In the future, services developed in the IP4 projects or by the TSPs involved in the IF should be not annotated with meta-data that facilitates their discovery. Since the ecosystem is now rather small, there is not a specific need for automated service discovery. However, as the ecosystem grows, service discovery is expected to become an issue.

Furthermore, creation of services is done manually for now. In future, it is important to support for techniques that would allow service providers – for example, TSPs – to specify the configuration of their services, which would then be traduced into services (or skeletons thereof).

As the discovery of external services is done manually, the process of its automation should be also considered in the following versions of the IF.

- **Monitoring and Governing**

While the IF ecosystem is rather small and limited to the members of the IF4, there is no issue of monitoring and governing the assets. However, with the deployment and scalability, the governing rules should be very well defined and followed by the ecosystem members. More specifically:

- Levels of access to services and assets must be defined (for example, to limit the types of users that could access certain data, for example, the so-called meta-network of a TSP). If any restrictions are necessary, in the future they should be regulated through legal contracts (for instance, integration with CMMP).

- Workflow management and version handling. Artefacts handled in the IF ecosystem (e.g., provided services and data) are not managed now through a codified lifecycle for their creation/update/destruction. So, with the scalability of the IF ecosystem, the workflow has to be set up.

Analysis of initiatives for different modes of transport, including the new ones (e.g., combined mobility operators, MaaS operators) have different requirements to the interoperability, so to consider these requirements in following versions of the IF ecosystem, representatives of these modes should be included in trials and testing. To put all types of stakeholders together require a specific governance body (e.g., ITxPT, MASAI), described in GOF4R D5.1 – Deployment Roadmap.

- **Leveraging the technological neutrality**

The IF development and deployment should follow the principle of the technological neutrality it will help the technology to be pushed to the market and find the right structure interoperable for different stakeholders, even outside the rail sector. In

The EIF provides a range of recommendations to the IF which can be considered during the IP4 IF deployment and scalability process.

Collaboration with ERTICO to set up common interoperability rules can be launched. ERTICO supports DATEX II model, which is a standard model at European level for the exchange of data related to traffic. This standard does not support semantic interoperability but a possibility could be to study if the transformation of DATEX II to an ontology would improve the management and interoperability among the used datasets, as it has identified in the public transport where efforts in the transformation of NeTEx to a corresponding ontology have started.

Deployment of the IF for NAPs is another challenge to consider by S2R. An overview of the NAPs across Europe shows that the NAPs vary in system architecture, organisation, monitoring of data users, accessibility, etc. Thus, there is a need for a more coordinated approach and exchange of ideas and best practices. Theoretically, the IF ecosystem can become such a solution, so the involvement of different Member States is required.

3. RECOMMENDATIONS FOR THE IF ARCHITECTURAL DESIGN

In this section, we first overview the most important lessons learned during the IF architecture requirement analysis as well as during the development of the IF, along with general discussions and recommendations. Then, the concrete and specific recommendations for further development of the IF architecture are summarized at the end of the section.

Lessons learned and related discussions and recommendations

The most prominent lesson we have learned within this project *was* ***to avoid the centralized deployment of the IF***, since it may jeopardize its overall scalability and robustness. Having one IF node (a software unit running in one server) as the single (physical) point of interaction that is responsible for the coordination and running of the whole aspects of IF leads to the single-point-of-failure problem. More importantly, it would become a performance bottleneck and it would prevent the scaling up as the eco-system grows.

The SPRINT project, in particular, has stressed all along that the concept of the IF as a monolithic software with centralized deployment, offering a fixed set of services/components which are over-tailored to work with certain transportation services is not suitable.

Firstly, the IF should not be perceived as a middleware that itself mediates the non-interoperable interactions between different parties. Rather, it is an infrastructure that offers base services, components and utilities (such as Converter, Automated Mapping, Semantic-Based Discovery. etc.) as the enablers of interoperability to the interested transportation actors. Hence, there is a need for having a federated IF, that is multiple IF Nodes distributed across Europe. While IF nodes could be in communication with each other, their operations are stand-alone and tailored to a particular region.

In this direction, our recommendations are as follows.

- The IF should be implemented in a distributed manner, without having a single IF Node responsible for the whole European Union. In particular, we recommend one IF node per National Access Points, but the distribution and granularity of regions could be possibly any of the following:
  - *one IF Node per EU country* (which occurs if each IF node acts as National Access Point), or
  - *one IF Node per district*, or
  - *one IF Node per major transport operator* (e.g., SNCF, Trenitalia).
- Yet, the complexity of the distributed system must be hidden from users. Hence, we recommend providing users with a single point of interaction to have access to the various IF functions. In particular, this could be achieved through the utilization of design patterns such as API Gateway [1] that leads all users to a single (logical) access point where the gateway redirects the requests to the desired IF instance/server.

Secondly, in the SPRINT project, we believed that interoperability should be established not only in the components and services offered by the IF, but also in its architecture and the way it has been

built and deployed. Accordingly, the IF in the SPRINT project has been designed and developed as a **modular set of self-contained components** that can be offered in an IF Node.

In this direction, to maximize the extendibility of IF, and to enhance engagement with the IF, it should offer its functions/services in a manner that is as fine-grained as possible. This is achieved by following a decomposition strategy based on the business purpose, requirements, and function, but hiding such complexity from users. In particular, we recommend the following:

- A microservice-based architecture, rather than a monolithic architecture.
- The IF should let users to selectively opt the various functions based on their needs and use only those components of the IF in which they are interested.
- The IF should allow interested parties to extend any desired services and components independent from the rest of the components and services.

The prototype IF developed in the SPRINT project is already in line with the above recommendations, both for what concerns the architecture of the IF itself, as well as the architecture of its internal components. As demonstrated in Deliverable D5.3, the components of the IF are self-contained, they can be deployed in a stand-alone manner, and they can be registered and later discovered to/via Asset Managers. Furthermore, where modularization is applicable, even single components of the IF can be created by the composition of multiple modules. For example, in scenario 8 in D5.3, users can customize the inner modules of a Converter and build a customized Converter based on their needs.

Another important observation we made was a subtle mistake that may occur for the development of IF, that is to consider it as a giant shared data centre for all the partners to store and share their data. The objectives of the IF, however, go **beyond the provision of a data-sharing framework**. The IF, instead, aims to facilitate technological and data interoperability that lets organizations interactively cooperate in order to use each other's data and services as seamlessly as possible and to build new services and utilities. Our strong recommendation in this regard is

- To avoid storing data in the IF and reducing its function to a data storage/sharing infrastructure.

Regarding data sharing, another essential consideration is that **data ownership always matters.** The willingness of business parties and organizations to expose their data usually comes with a strong desire to keep the ownership and full control of their data. The IF architecture and associated technologies, then, must comply with such demand by design and let the owners fully control the access rights to their data. In the current implementation of the IF, the owner of data and assets can control who can access the published assets.

- In line with the distributed nature of the IF architecture, as well as the importance of preserving the full control of owners over the access policy of their data, we recommend that the accessibility to the system should not be governed by a centralized authentication and authorization system. A distributed access control mechanism is recommended that lets individual business parties and organizations have full control over their authorization policies.

Moving to another dimension, an important lesson we learned during the SPRINT project, and, in particular, in our collaboration with the CONNECTIVE project, was that **the integration with systems currently in use matters.** The goal of the IF should not be to replace existing functions, and it should not create any process duplications; rather, it should re-use the current infrastructure and integrate with existing systems. For example, the Operator Portal is an existing and well-established framework currently to facilitate user management in the transportation domain. The IF hence can integrate with such system to let users seamlessly log-in to the IF instead of creating yet another registration portal and mandating users to re-do the registration process over and over again. Furthermore, such integration greatly decreases the administrative burden on the users. In particular, we recommend the following:

- Facilitate registering to and joining the ecosystem by providing a single-sign-on solution. This also makes it simpler for various transportation-related organizations and operators to (logically) enable their users to use the IF easily and seamlessly.
- Leverage the distributed and cross-organization collaboration.

In line with these considerations, the current Asset Manager is integrated with an Identity Provider to provide maximum interoperability with other systems such as the Operator Portal. Hence, users can register to either the Asset Manager or the Operator Portal and have access to both systems.

Another critical insight we gained starting with the requirement analysis, but also later during the implementation and the development of the IF was that the **IF should offer options**. Different users with different business goals and demands and different level of technical expertise need to engage with the IF differently. Hence, the IF should avoid constraining users to a single option for engaging with the IF and its component, but it should foresee multiple possibilities and solutions. In particular, we recommend the following.

- The IF should provide different deployment options, from the direct download of components to service-based model, for different categories of users and business partners.
- The IF should try to minimize any technical requirements and reduce the barriers of entering the ecosystem and facilitate engagement with the IF for different categories of users and business partners.

This consideration has been at the centre of our design goals and the current IF implementation enables various engagement and deployment possibilities. Such options have been successfully developed and validated in Deliverable D5.3, and specifically through scenarios S4, S5 and S6.

Another important message we would like to highlight here is that **Automation Matters**. Indeed, automation plays a pivotal role to increase interoperability. It breaks a complex process into intermediate steps and provides a formal description for each step to be processable by machine. Accordingly, the existence of such a formal and machine-understandable description of procedures further leads to the possibility of formalizing the interoperability and integration without any human intervention. In this direction the IF architecture should promptly foster automation, in particular, we recommend the following.

- The IF should support automatic software building, in particular through the realization of continuous integration/delivery tools.
- The IF should support automatic/semi-automatic deployment via deployment scripts.

The current implementation of the IF has already taken such considerations at heart and it contributes to this vision in many ways. For example, the Mapping Tool is one of the components of the IF that particularly offers a semi-automated mapping generation between heterogeneous standards. The early validation of this tool in Deliverable D5.3 Scenario S7 showed promising results. Furthermore, the IF enables many automated procedures for the creation as well as the deployment of Converters, as shown scenarios S8 and S9 of D5.3. Finally, the IF offers various contributions on Semantic Automation and the corresponding recommendations are reported in this document in Section 6.

Last but not least, it is evident that semantic web technologies win. Semantic technologies greatly help to overcome existing heterogeneity and lead to “a common framework that allows data to be shared and reused across application, enterprise, and community boundaries” [1]. In this direction the IF architecture must natively support semantic technologies, so we recommend the following.

- The incorporation of semantic meta-data and descriptions into asset descriptions.
- To favour semantic-based searching and discovery, in particular through distributed SPARQL endpoints.

Semantic technologies are one of the pillars of IF. In a nutshell, the IF is designed to both use semantic technologies to enhance the semantic interoperability – for example through distributed SPARQL endpoints, as well as meta-data annotations of the assets – and to provide enablers to foster semantic technologies and approaches – for example through ontology management utilities. The C-REL implementation and validation of all the above tools have shown promising results for achieving the above-mentioned goals.

Summary of recommendations:

- Avoid implementing the IF as a centralized framework as well as a centralized data centre.
- Distribute the instances of IF nodes within European Union regions, possibly having one per National Access Points.
- Avoid a monolithic architecture.
- Favour the use of structured formats, such as ontologies and vocabularies, to describe data, to provide structured and machine-readable service descriptions, etc.
- Favour the use of semantic discovery, and in particular of distributed SPARQL endpoints, to provide unified access to a complementary set of (sometimes overlapping) knowledge graphs.
- Architect the IF as a modular software and in compliance with the Service-Oriented Architecture paradigm. In particular, we recommend implementing the IF and its components as microservices.
- Favour the API Gateway pattern for the microservice-based architecture.
- The IF should store only meta-data and not concrete data.
- The IF should provide a registry that lets organizations advertise data by sharing the “meta-data” only.

- Use of any data (if they are not open) should be completely controlled by the owners of data.
- Integrate with Operator Portal to facilitate the single-sign-on process.
- Minimize the set of services/component for instantiating IF.
- Provide the possibility of extending IF features, functions and components upon needs of participating actors.
- Ensure deployability of IF services, in particular by the deployment of IF components/services through container technology that packages such components as portable, self-contained and ready-to-run software units.
- Emphasize a plug-and-play approach and provide means for the automated generation of software units.

4. RECOMMENDATIONS FOR THE IF ARCHITECTURE AS A COMPONENT TO NAP

D2.3 defines different usage scenarios about how the Interoperability Framework can relate to National Access Points. As we described in the document, the IF can be used as a way to implement a NAP, or as a companion to existing NAPs. The features which are deemed as mandatory and nice to have are all covered by the current version of the SPRINT tools, which provide features far more advanced than what the current National Access Points implementations provide. The integration between the Asset Manager and the Converters allows automatic dataset conversion, and therefore an IF-based NAP could hide the complexity of Transmodel-based standards to all TSPs currently providing their data in different formats (like GTFS). The possibility to define complex lifecycle management processes moreover allows for finer-grained and automated control of the process and its implied roles and responsibilities.

Though the IF (as described in the various SPRINT deliverables) can play the role of a National Access Point, each member state is already designing or providing a solution which is not based on the IF. Therefore, a realistic role of the IF is as a companion to NAPs, to ease obtaining metadata from multiple sources and to contribute to a specific NAP according to the regulations.

Being a companion to a NAP implies several constraints which must be addressed and solved. The first constraint imposed by the NAP is the usage of Transmodel-based specifications. NeTEx, DATEX, SIRI, are all standards and specifications derived from Transmodel, which acts as a common conceptual model. The first recommendation is, therefore, to align the IT2Rail/IP4 ontology to Transmodel (which should be turned into an ontology itself), and to provide a mapping to be able to import data from NAPs into the IP4 ecosystem and to export data complying to the regulations. An important requirement of this alignment is that it must be “lossless”, i.e. it must be possible to import all the information contained in a NeTEx dataset, and to export all the information to that format.

Another option for such alignment, although with a higher cost, is to re-design the IT2Rail/IP4 ontology as an extension to Transmodel. That would require re-writing the Service Implementation inside Brokers to reflect the model changes but would widen the adoption of the IF since Transmodel being pushed as a central element of NAPs means it will become a sort of “official” model for all EU transport operators.

A second constraint imposed by NAPs is related to management processes. Each NAP is implemented with a different publication process in mind (as described in D2.3), and accessing a dataset implies activating a process which is again dependent on the specific NAP implementation. A NAP-aware IF should then split the acquisition of a new data source between asking for permission to access a dataset and the actual downloading of the dataset. When publishing new data, instead, the IF should provide metadata according to the specific NAP constraints, and use NAP-specific authentication means to be able to upload a new dataset.

Summary of recommendations

- Define a Transmodel ontology
- Align the IT2Rail/Shift2Rail ontology to Transmodel or rewrite it as an extension to Transmodel
- (In case the two ontology are simply aligned) define mappings to convert data between Transmodel and IT2Rail/Shift2Rail data models
- Define a common metadata ontology as a superset of the existing NAP metadata schemas
- Develop Converters to import metadata from NAPs according to the to-be-defined metadata ontology
- Develop Converters to export metadata to the specific format adopted by the destination NAPs
- Implement NAP-aware publication processes for selected asset types (like Journey planning)

5. RECOMMENDATIONS FOR PERFORMANCE AND SCALABILITY OF THE IF

First and foremost, a significant point that must be highlighted here is that the scalability and performance of the IF can be considered and analyzed from two different perspectives: first, for the IF as a whole and, second, for its individual components. In the following, we first focus on the scalability and performance considerations for the whole IF, on the related challenges and the corresponding recommendations. Then, we discuss the performance and scalability issues of individual components of the IF, how they can be scaled and achieve performance targets, what are the related challenges and bottlenecks and the corresponding recommendations, mainly based on the results and analysis of the C-REL implementation and of the validation of such components.

Scalability and performance are categorized as two different properties of a software system, but they are highly correlated. For a given environment that consists of properly-sized hardware, properly-configured operating system, and related middleware, if the performance of a software system deteriorates rapidly with an increasing load (number of users or volume of transactions) before reaching the intended load level, then it is not scalable and it will eventually underperform [2].

In this regard, the deployment of strategy the IF – i.e., centralized vs distributed – becomes the key factor in managing the performance and scalability of the IF. Accordingly, we recommend practising the federated deployment of multiple IF instances distributing the load throughout many nodes that are cooperating to create a holistic distributed infrastructure. This approach – in comparison with the centralized model where one single node is responsible to manage every aspect and ever-increasing user's loads – enhances the overall performance of the system and ensures the scalability of the IF to become a framework used by considerably large numbers of transportation operators and actors all over Europe.

Accordingly, enhancement of the scalability and performance of IF is mainly reflected in the architecture of the IF and the architectural design decisions are the key factor to balance the scalability and performance requirements of the IF. Hence, our recommendations concerning the scalability and performance of the IF are in line with those mentioned in Section 3, and they are as follows.

- Avoid a centralized implementation of the IF.
- A federated IF is recommended, in particular as a materialization of NAPs.
- The IF should not be used as a data storage.
- The IF should only store meta-data.
- The initial setup of the IF must be minimized so users can add only those components and uses of the IF according to their needs and performance constraints.

Also, a critical performance issue for the IF is the ability to handle a load of requests for the downloading of artifacts; this is yet another case that highlights the necessity of having multiple federated IF Nodes to scale up as the number of download request increases in such a way that the system is able to sustain its regular functionality without suffering a slowdown in its overall performance.

Furthermore, employing (and anticipating) the suitable *Deployment Approach*[1] for each service/component can practically deal with scalability and performance issues.
For example[2], for the use case of the batch data conversion process using a SPRINT Converter, and the automated learning of similarities among multiple standards through the SPRINT Mapping Tool, since the process is accomplished off-line and not in very frequent cycles, *The Direct Download of Deployable Component* approach seems the best option. Through that, the consumer downloads a deployable converter/mapping tool artefact (JAR, Docker image) to use it locally. Hence, the responsible entity to ensure the scalability of the converter/mapping tool is the service consumer. So, to ensure scalability and performance of the IF our recommendations include the following:

- Outsource the performance management to consumers by favouring the Direct Download Deployment strategy for those components of the IF which are to be used as self-contained modules.

The reason behind this recommendation is that a single instance of a stand-alone IF component, for example, a converter/mapping tool with a reasonable performance profile (e.g., few hours for batch conversion and few seconds for the mapping process) is enough for each service consumer and in the case of higher demand, the consumer could horizontally scale up its system by running multiple independent instances of the converter/mapping tool, which in turn is an external activity concerning the IF Node.

There are however a group of IF components/processes which inherently do not have strict performance requirements, but they may impose scalability challenges. To give an example, the process of joining the IF (registration, role assignment, etc.) must be done only once, and the information provided in this step seldom changes. Subsequently, users can tolerate a slower process without jeopardizing or losing interest in the framework. Similarly, the discovery process typically includes multiple executions of a simple search operation, each time adding filters to the previous attempt. In such cases, the main threat is the ability of the IF to be able to bear with the increasing number of users operating with the system simultaneously. In this regard our recommendation is as follows:

- Outsource the performance and scalability management to the service providers by favouring the Direct Access Deployment strategy (see Deliverable D3.2) for the interoperability services of IF.
- Scale-up IF capacities by favouring the Runtime Environment Deployment by automatic composition, deployment and replication of IF components and services on distributed nodes through cloud orchestrators such as Kubernetes.

Finally, other performance-critical aspects of the IF are mainly related to the functions of the IF that must deal with some sort of real-time data processing. For example, Runtime Message Conversion (see Deliverables D5.1 and D5.2), which aims at converting messages exchanged between two parties – i.e., converting the message represented in the sender standard to the standard

[1] for more details regarding various deployment approaches please refer to SPRINT Deliverable D3.2.

[2] for more details please refer to SPRINT deliverables D5.1, D5.2.

understandable by the receiver – in real-time. For instance, when a shopping application tries to discover various itineraries offered by different and heterogeneous TSPs, a swift conversion process is required to proceed with the shopping procedure. In such cases a reasonable performance of this component of the IF is highly critical, otherwise, it would become the bottleneck for the whole process.

- Avoid monothetic and complex services that might consume huge memory and processing time and favour modular and micro-service-based architecture for each component and sub-system of the IF to distribute the loads.
- Favour horizontal scaling strategy and replication of the services/components.

5.1 LESSON LEARNED FROM C-REL IMPLEMENTATION AND VALIDATIONS

In this section, we discuss the outcomes of D5.3 functional validation, identifying additional requirements for F-Rel.

5.1.1 Asset Manager

The implementation of the scenarios defined in D5.1 for C-Rel showed that the Asset Manager is able to play the role of the ecosystem catalogue inside the Interoperability Framework. The architecture and implementation choices showed the following advantages related to flexibility and scaling:

- multiple authentication mechanisms can be used, therefore the Asset Manager can benefit from a wide array of possible identity providers (Google, Facebook, Github, …);
- leveraging on containerization, there is a clear separation of concerns and each component is devoted to a specific task (CMS, CI/CD, process automation, caching, long-running tasks, …);
- components can scale independently.
- even if the Asset Manager implemented in SPRINT is a complete rewrite with respect to what was delivered in IT2Rail, the overall stability is improved benefiting from higher reuse of production-ready open-source software.

Currently, the only weakness related to the scalability of specific components is represented by the Process automation component. A BPMN process engine relies on a database to keep track of the statuses of different process executions, and such a relational database is a bottleneck. This disadvantage is anyway mitigated by the fact that the process engine manages the lifecycle of an asset and the process by which users can obtain the permission to access a specific asset. Such processes are not heavily stressed, since lifecycle management happens only during publication time, and being the Asset Manager mostly devoted to organizations and developers ensures that the number of requests for gaining access to a specific asset will not rise dramatically over time.
While integrating the Distributed SPARQL endpoint, we realized that the Asset Manager can also play the role of an “aggregator of catalogues”. In the transportation domain, multiple initiatives are offering their catalogue of datasets or services (one of them being the National Access Points), and we believe that the Interoperability Framework can benefit from having a single and controlled source of “truth” showing meaningful assets coming from trusted external catalogues. We plan to study how

to offer such a feature in F-Rel, taking into account the existence of National Access Points. We believe that the current tools provided by SPRINT will be flexible enough to cover such a case. The F-Rel version of the Store could be modified to achieve such result, and distinguish between locally-managed assets which are published and managed via the BPMN lifecycle processes deployed on the Asset Manager and remotely-published assets which come from remote (but trusted) sources. Such feature would enable integrate a single Asset Manager with different National Access Points, and that would enable a pool of companies using the SPRINT IF to use the Asset Manager as a single point for accessing transport data pertaining to different countries, and to re-use such data in a controlled way.

5.1.2 User Manager

The SPRINT IF architecture comprises a User Manager as the component devoted to authentication and authorization. In C-Rel such features are embedded inside the Asset Manager to speed up the development process. The final version of the SPRINT IF should investigate the usage of off-the-shelf, open-source solutions for identity management as a separate component. With an independent Identity Provider, the SPRINT project could leverage on widely accepted standards (like OAuth 2.0) for authentication and authorization, and maybe also leverage on the “federation” features that could enable accessing different IF nodes with the same user credentials. In the context of identity management, it will be important to stress the difference between authentication and authorization. The former aspect must be mandatorily centralized in the Interoperability Framework deployment, while the second aspect (“who is authorized to do what”) can also be delegated to the specific components.

5.1.3 Mapping Tool

The initial functional validation of the Mapping Tool is promising[3]. However, the tool needs to be extended/improved in several ways. In particular, the enhanced (F-REL) version of the Mapping Tool will have some added features. Most importantly, currently, the Mapping Tool is a command-line application, so we have the plan to add a Graphical User Interface to facilitate working with it and increase its user-friendliness. Furthermore, the final release of the tool will offer an automated generation of annotations, which are necessary for the conversion mechanism used by Converter components of the IF. The annotations mechanism will support both Java-based annotations as well as RML-based annotation.

The mapping techniques also need some improvements. The C-REL implementation is mainly based on the semantic similarity of the terms in the source and target standards. However, the C-REL results of the mapping showed us that the linguistic similarity of terms alone may not be a heuristic powerful enough to generate a comprehensive one-to-one mapping between concepts of different standards. Hence, for the F-REL, we plan to further extend the work to also include Structural mapping. The idea behind the structural mapping is to extract the similarity of the terms based on the syntactical structure of the source and target standards. We hope that the inclusion of structural mappings into the mapping algorithm can enhance the overall accuracy of the suggestions.

[3] The accuracy of the Mapping Tool varies depending on the standard: the maximum accuracy is 76%, while the minimum is 67%, which leads to an average accuracy of 72%.

5.1.4 Distributed SPARQL endpoint

In C-REL the implementation was based on having a distributed SPARQL endpoint that runs queries on two data sources and based on the results obtained, we can say that the Distributed SPARQL endpoint still needs more research since in most cases they fail because:

i) They do not fully support SPARQL 1.1 operators.
ii) They produce incorrect or incomplete responses, i.e., the number of results obtained differs with respect to the baseline (RDF materialized graph).
iii) The performance and scalability are very low when the data size increases.

It is important to note there is still a lot of research to be conducted in the current field of construction of virtual knowledge graphs and t there is much work to be done to optimize queries on functions, on the distributed approach, and latency when executing a query on multiple sources. for F-REL, we continue working to include user preferences to the queries as proof of concept to check if some value can be added to the queries in the IF architecture. Skyline queries are a kind of queries based on user preferences that identify the set of rows that are not dominated by any other row, where a row is considered to dominate another one if it is as good or better in all criteria and better in at least one criterium.

Finally, we expect this study to be a stepping stone in this area where much research and development has been done for decades, but there is a need for more mature applications to be used in real-world environments. Indeed, our experimental study has shown that there are still relevant open issues such as SPARQL conformance, semantic preservation in the translation from SPARQL queries to the query languages used to query raw data, and the application of query evaluation.

5.1.5 Converter

In the context of C-Rel we created a Converter framework (Chimera), to provide a flexible solution to let developers implement conversion processes. We took into account two main conversion cases: datasets conversion and service mediation. Such cases imply very different requirements related to performance and scalability. Conversion of datasets can potentially take hours or days, and the main challenge is related to keeping memory consumption low. In service mediation the size of the messages is small, and the challenge is being able to process a message as fast as possible to be able to convert more messages in parallel. While being able to offer such performances, we also aimed at offering a flexible solution. Flexibility, in this case, means offering to developers the possibility to use different lifting and lowering techniques, and to be able to use the same framework to implement batch conversions, REST services, SOAP services, integrate with message queues, and in general being open to integration in existing production systems.
C-Rel version of Chimera currently features:

- Annotation-based lifting and lowering based on Pinto[4]
- Annotation-based lifting and lowering based on ST4RT technology, which was ported to the new framework
- Declarative lifting based on RML

[4] https://github.com/stardog-union/pinto

- Declarative lowering based on a custom solution embedding SPARQL queries inside Apache Velocity templates

The choice of Apache Camel as the basis for the creation of Chimera allows the integration with hundreds of components[5], further increasing the possibility to integrate a semantic-based conversion solution in a production system.

Considering the *batch data conversion* scenario, we tested performances and scalability of a Chimera pipeline for conversion implemented using the RMLMapper lifting block and the Apache Velocity Template-based lowering block. The presented analysis showcases the potentiality of the implemented solution, managing to generate and handle knowledge graphs with millions of triples within the conversion pipeline. On one hand, we highlighted important aspects to be considered to optimize performances of both lifting and lowering approach. Most importantly, we showcased how the number of *join*s condition in the RML mappings highly affects performances. On the other hand, considering scalability issues we noticed that memory consumption can be a problem and Chimera introduces a not negligible overhead. One of the elements which contributes to the high memory consumption is the fact that all the data is processed in memory. For this reason, for F-Rel we will try to optimize memory usage in Chimera modifying the RMLMapper (e.g. trying to integrate an external repository to store the materialized graph) and studying the possible integration of additional blocks (e.g., a block interacting with the tested SDM-RDFizer or last optimizations techniques that have been proposed in the state of the art [17]). Moreover, since materialization is the main responsible for memory consumption, we plan to test and compare conversion pipelines exploiting a hybrid solution including ideas defined in virtualization for the lowering phase.

Considering the *runtime data/message conversion* scenario, we tested and compared performances of the ST4RT annotation-based method ported in the new converter architecture (Chimera). Results obtained show a great improvement in performances. For F-REL, we will investigate further performances considering scalability in the number of concurrent requests made to the converter in this scenario. Moreover, considering *runtime data/message conversion* requirements, we will investigate performances and scalability of a declarative approach based on RML mappings and lowering templates, like the one used for the *batch data* scenario.

[5] https://camel.apache.org/components/latest/

6. RECOMMENDATIONS FOR THE IP4 IF SEMANTIC AUTOMATION

The Interoperability Framework supports by design a wide array of automation solutions, and it is being used as a starting point to implement an IP4 ecosystem. To that extent, the IP4 ecosystem (as shown in Figure 1) is just one of the possible interoperability solutions which can be implemented using the IF.

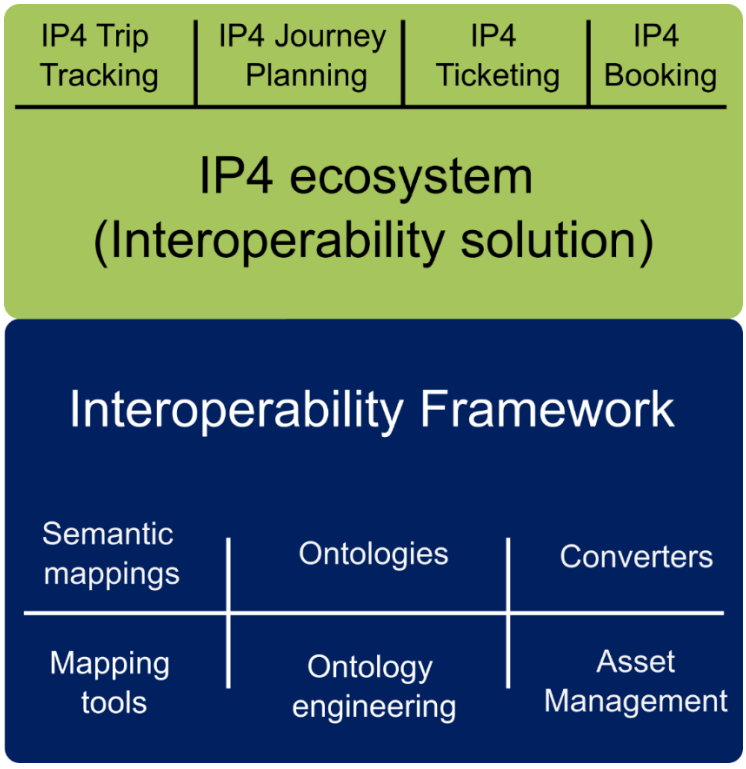

Figure 1 Interoperability Framework and IP4 ecosystem

The establishment of an IP4 ecosystem encompasses several different activities which must be taken care of. Such activities stem from the creation of a common model to the development of the integration solutions to the usage of sharing platforms to raise awareness and ease the adoption. A high-level list of activities is reported here:

1. Develop the ontology
2. Analyse existing ontologies
3. Develop mappings
4. Develop converters
5. Develop resolvers
6. Publish artifacts
7. Gain access to existing artifacts
8. Perform tasks after the successful publication of artifacts

Automating parts of this long list of activities can play a key role in establishing an efficient ecosystem, lowering the maintenance costs and helping govern a complex distributed system ran by different transport operators. Some of those activities can be fully automated, while others require

human intervention. In the following sections, we will describe the possible roles of automation and provide a set of recommendations to the rest of the Shift2Rail IP4.

6.1 ONTOLOGY DEVELOPMENT

Ontology development is a complex subject, and we can roughly divide it into two main branches: creating a new ontology from scratch and converting an already existing data model using the W3C Semantic Web stack. The former activity requires a fully human activity since it involves understanding a domain and its rules and representing them as a set of logical axioms. Automation is any way possible to support the human activity, easing collaborative editing and providing up-to-date graphical representations of the ontology. Those two aids are especially useful when the domain which is being modelled is vast, and when the team is actively working at different aspects at the same time.

Full automation is not possible even in the latter branch of the ontology development, namely the conversion of an existing model into an ontology. In this case, the conceptualization of the domain has already been performed by someone else, but it has been serialized into a non-ontological format. XML or RDB schemas, UML diagrams, are all examples of such non-ontological formats. The aim of the ontology development activity, in this case, is to obtain a clean model, removing attributes and relations which are usually introduced by the specific format, while at the same time staying very close to the original model to keep compatibility. The role of automation, in this case, is to provide a first rough draft of the ontology, which can be used by ontology designers to speed up the development process.

While converting messages and connect different systems, the quality of mappings is a critical issue. Such quality is influenced by many factors, like the level of knowledge of the domain, knowledge of the two ends of the communication channel, and also changes in the models during the time. The semantic-based solution can help in identifying missing data while developing mappings, and in identifying incompatibilities generated by changes in the ontologies. SHACL is an RDF-based language useful for data validation while converting messages. This technology can be used both to detect missing data during conversion and to drive the development of new mappings. Its role in the Semantic Web stack is akin to XML Schemas in the XML stack since it allows specifying cardinalities and consistency rules for RDF data. The development of SHACL shapes could be included in the ontology engineering efforts, and releasing SHACL shapes with each ontology release could help mapping developers in providing better Converters. Ontology changes during time is another factor affecting mappings quality. Each new version of an ontology could break existing Converters, and early detection of incompatibilities will be a key factor in keeping soundness of the IP ecosystem. The development of test cases could help to tackle such issue, and such test cases could be automatically executed by the Asset Manager after publishing a new version of the IP4 ontology. Test cases for the IP4 ecosystem should imply the following requirements:

- Each Ontology should define a set of queries (or competency questions) that can be answered
- Each Ontology should publish an example dataset
- Each Converter should declare a sample input and output message, and the ontologies used during the conversion

This set of requirements would allow a test to notify Converters and Mappings owners that changes in the ontology structure are going to break their artifacts, in case they are developed to use the latest version of the reference Ontology.

Summary of recommendations

- Provide up-to-date diagrams of the Shift2Rail IP4 ontology
- Provide documentation about possible alignments with existing ontologies
- Release SHACL shapes together with the Shift2Rail IP4 ontology to show how the ontology is intended to be used
- Use competency questions to create tests
- Provide examples of how the Ontology can be used (data samples)
- Provide input and output examples for Converters
- Link a Converter to Mappings and Ontologies in the Asset Manager

6.2 INTEGRATION ENACTMENT

The current efforts of the SPRINT and CONNECTIVE projects are aiming at providing and testing interoperability solutions based on the concept of the Interoperability Framework. Integrating different systems requires overcoming multiple technical difficulties, and the integration process requires analyzing several aspects, like:

- whether the two systems use the same communication approach (pull vs. push);
- whether the two systems are stateful or stateless;
- whether the two systems use compatible processes;
- how much information from a source system can be sent to the destination system.

The SPRINT project is demonstrating that some aspects of the integration process can be streamlined using a combination of semantic techniques and already existing open source solutions. When integration is a message-to-message conversion, the Asset Manager can automatically generate a running Converter by just stating the relevant ontologies, dataset and mappings. Even if we showed that automation can be fully applied, the case of a message-to-message conversion is any way just a small subset of the cases in the domain of the transportation domain. Past work in the ST4RT project demonstrated that the biggest obstacle in integration is the different information granularity between two systems, and the case of FSM to TAP/TSI 918 showed that message exchanges are usually part of larger processes. In such cases, it is important to store the context information which is attached to the process instance.
In all the cases where a simple message-to-message conversion is not feasible, a possible and recommended solution is to take the simple case as a starting point and to create customized conversion pipelines leveraging on the components supported by Apache Camel, which is at the core of the SPRINT Converter solution. In cases where the processes to be implemented are complex, a viable solution which minimizes manually coding is to embed a Business Process engine inside the Converter. With that solution, a large part of the process mediation could be implemented by drawing BPMN diagrams, while the actual conversion of messages would be delegated to the semantic components already supported by the SPRINT Converter framework (Chimera).

Summary of recommendations

- Document the list of features of each system which is being integrated inside the IP4 ecosystem
    - Pull-based vs. push-based
    - Stateful service vs. Stateless service

    - Processes involved with each message exchange
- Use the basic message-to-message conversion pipelines automatically generated by the Asset Manager as a starting point to create stateful or process-based Converters
    - In case the processes to be mapped are complex, consider embedding a Business Process engine inside the Converter

6.3 ECOSYSTEM MAINTENANCE

As introduced before, the IP4 ecosystem is an interoperability solution which is being built for the transportation domain using the tools of the Interoperability Framework. Some of the tools provided by the SPRINT project can be used to efficiently address ecosystem maintenance. With the term “ecosystem maintenance” we mean all the activities which contribute to keeping a distributed system running with minimal inconsistencies and downtimes. Such activities encompass the high-level governance of the ecosystem, dealing with rules and contracts between companies, and the constant checking that the information contained in the asset descriptions is up to date and does not lead to an inconsistent distributed system. The Asset Manager can play a key role in automating all the technical parts of the governance process since its role is of being the single source of knowledge for all the components of the ecosystem. It can, therefore, ensure that a major release of service is notified in time to all its users, it can be used to automatically execute ontology-based tests to ensure that changes do not break any existing Converters, and it can be used to automatically convert datasets into other formats (like the Transmodel-based standards required by National Access Points).
Exploiting the automation functionalities of the IF as implemented by the SPRINT project requires defining clear governance of the IP4 ecosystem. Such governance should define roles and responsibilities, and should also define lifecycle management processes for the various types of assets which are to be managed by the ecosystem. Once such processes are defined, they can be drawn and developed using BPMN and deployed onto the Asset Manager, which will then be able to enact such governance structure and automate its effects.

Summary of recommendations

- Define a governance structure for the ontology
- In case of a centralized deployment of the IF, define a governance structure and IF ownership
- Define and share the details of lifecycle management processes
- Pay attention to the dependencies between assets to avoid breaking the IF functionalities
- Use the Asset Manager as a command and control centre, linking the lifecycle management to automation tasks to be performed after a successful publication.

7. RECOMMENDATIONS FOR THE MARKET UPTAKE

This section is related to the section on business and market validation in SPRINT D5.3. The results of the validation contribute to the first set of recommendations for the market uptake.

Indicator	Description	Recommendation
Data openness	To what extent the solution supports data openness that influences market uptake of the IF positively, provided that the data opening does not negatively impact the provision of services operated under public service obligations.	The IF should facilitate the exploitation of open data concepts and policies for the provision of advanced digitalized end-to-end multimodal mobility services.
Gaining the critical mass of IF participants	Easiness in joining the IF ecosystem which can lead to the increasing the number of users in the IF ecosystem	The critical mass can be gained through minimizing or eliminating the need for adaptation of legacy systems and participation in centralized governance. The mandatory requirement to join the IF ecosystem has to be limited by the necessity to register. The way how the IF should work: availability to publish resources and discovery of other stakeholder resources through the Asset Manager.
Market diversity and inclusiveness	Removing the distinction between big market players and small TSPs. IF's scope should be widened to MaaS operators. Ability to deal with different stakeholders' policies and regulations.	Create the same rules and requirements for all stakeholders by minimizing ICT development costs and governance overhead through minimizing or eliminating the need for adaptation of legacy systems.
Stakeholder's management	One of the key elements that the IF governance has to address: lack of cooperation among stakeholders, collaboration with other non-transport related entities (organizing authorities, financial institutions, IT services,…)	"lack of collaboration" should be considered from a point of view of making interoperability a digitalized mechanism instead of a 'governance' policy. The focus should be on registering/publishing the resources/assets in the Asset Manager keeping full control of them, and discovering available resources registered/published by other stakeholders. In this scenario, the nature of governance is therefore changed to maintaining the collaboration *tools*, i.e. the IF components themselves.

User-friendliness	To what extent the solution addresses the issue of lack of knowledge in semantic ontology and data-interoperability	The solution has to use standard semantic web specifications and other standard technologies and should be architected to allow the flexible composition of processing chains from common blocks through configuration scripts of open-source build and runtime frameworks., minimizing the need for ICT development.
Reliability and security of the ecosystem	How the solution addresses the issues on cybersecurity.	Reliability and security should be delegated to the runtime environment in which the components execute. In this way, a specific runtime environment can be configured for specific reliability and security requirements, protecting components that do not need to implement their own and can therefore be standardized.

8. CONCLUSIONS

This deliverable is the first version of recommendations to Shift2Rail IP4. Recommendations are extracted based on

- the analysis of high-level and architectural design requirements of S2R IP4 projects (CONNECTIVE and ATTRACkTIVE) and different EU initiatives in SPRINT D2.1;
- the analysis of NAP;
- the work in other SPRINT WPs related to performance and scalability, as well, as IF semantic automation.

Therefore, the development of the IF should be focused on the following principles:

- leveraging automation;
- proper monitoring and governing;
- leveraging the technological neutrality;
- decentralization and distribution;
- modularity;
- meta-data over data sharing;
- minimalism and customization;
- Unified, widely used ontology (e.g., TRANSMODEL);
- Considering NAP as the potential targeted audience of the IF.

9. REFERENCES

[1] Semantic Web Group, "W3C SEMANTIC WEB ACTIVITY," 11 December 2011. [Online]. Available: https://www.w3.org/2001/sw/. [Accessed 14 August 2019].

[2] H. H. Liu, Software performance and scalability: a quantitative approach, John Wiley & Sons, 2011.

[3] M. Glinz, "On non-functional requirements," in *n 15th IEEE International Requirements Engineering Conference (RE 2007)*, 2007.

[4] European Commission, Directorate-General for Informatics, "New European interoperability framework : promoting seamless services and data flows for European public administrations," Publications Office of the European Union, Luxembourg, 2017.

[5] SPRINT project, “D2.1 Initial analysis of requirements of S2R IP4 projects and other EU initiatives,” 2019.

[6] SPRINT project, “D2.2 Requirements for an IF architectural design (C-REL),” 2020.

[7] SPRINT project, "D2.3 Requirements for an IF architectural design (F-REL)," 2020.

[8] SPRINT project, "D3.2 Performance and scalability requirements for the IF (C-REL)," 2020.

[9] SPRINT project, "D4.2 A lightweight solution to automate the generation of ontologies, mappings and annotations (C-REL)," 2020.

[10] SPRINT project, "D5.1 Requirements, scenarios and use cases for the proof-of-concept (C-REL)," 2020.