Cooperative Cities extend and validate mobility services

WP3

D3.2 – ITS system interfaces to end users and service delivery chain of the service providers

Version

1.0

Dissemination level

Public

Co-Cities is a Pilot Type B Project funded by the European Commission, DG Information Society and Media in the CIP-ICT-PSP-2010-4 Programme

European Commission
Information Society and Media

Contract Number:

270926

Acronym:

Co-Cities

Title:

Cooperative Cities extend and validate mobility services

Contractual date of delivery:

December 2011

Actual date of delivery:

2012-10-04

Main author(s) or editor(s):

Said Rahma (ATOS)

Other author(s):

Carlos Maestre Terol (ATOS), Marco Garré (SOF), Michele Masnata (SOF), Axel Burkert (PTV), Dieter Meinhard (BRI), Tvrzský Tomáš (TMX), Lena Reiser (ATE), Alexander Froetscher (ATE), Roman Pickl (FLU)

Version History:

Version	Date	Main author(s)	Summary of changes
0.1	19/12/2011	Said Rahma	Proposal for the ToC
0.2	15/02/2012	Said Rahma	ToC restructured
0.3	05/03/2012	Said Rahma, Axel Burkert, Marco Garré, Michele Masnata, Dieter Meinhard, Tvrzský Tomáš, Lena Reiser, Alexander Froetscher	Update contents, add first partner contribution
0.4	08/03/2012	Said Rahma, Carlos Maestre Terol	First draft version for review cycle
0.5	30/04/2012	Said Rahma, Axel Burkert, Marco Garré, Michele Masnata, Dieter Meinhard, Tvrzský Tomáš, Lena Reiser, Alexander Froetscher	Consolidate and update contents based on partners pre-review and comments
0.6	07/05/2012	Said Rahma,	Final Document ready for peer review.

			Updated contents based on internal review, feedback and comments
0.7	14/05/2012	Said Rahma	Final document ready for submission of approved version
0.8	01/06/2012	Said Rahma	Added comments that reference reviewer comments from EU
0.9	04/09/2012	Said Rahma	Edit and update contents according to the comments of the EU reviewer and deliverable improvement (See below the relevant list of changes).
		Said Rahma	The cover front has been corrected to follow the project guideline information.
		Said Rahma	Section 2.2 improved, adding the nature of dependencies.
		Said Rahma	Section 3.3 "General remark" improved, adding the definition of the terminology "reference platform for validation" in the context of Co-Cities project.
		Said Rahma, Michele Masnata, Marco Garré, Axel Burkert, Dieter Meinhard	Section 6.2 improved, adding services domains description to provide information about the inclusion of cycling and pedestrian aspects into the Multimodal journey planning service domain.
		Michele Masnata, Marco Garré	Section 6.3 improved adding contents.
		Said Rahma, Marco Garré	Section 7.1 improved, removing the irrelevant characteristics, identifying and adding information on the table 6 "Generic mobile device features" regarding the mobile device operating system and platform that will be used in the Co-Cities demonstrators.
		Said Rahma, Marco Garré	Section 7.2 improved, removing the irrelevant pre-requisites and features and focused on the characteristics that will be used in the Co-Cities pilots.
		Said Rahma	Section 8.1 improved, adding a reference to the D2.3 chapters where the KPI is described and will be used in the Co-Cities validation.
		Said Rahma	Section 8.2 updated content about monitoring performance issues.
		Said Rahma	Chapter 10 section 10.1 improved, adding the means of the missing abbreviations.
0.95	28/09/2012	Said Rahma, Roman Pickl, Tvrzský Tomáš	Updated from internal review process. Final version ready for peer review.
1.0	04/10/2012	Said Rahma, Axel Burkert	Updated from peer review process. Final version ready for submission of approved document.

List of the Co-Cities Project Partners

Partner no.	Partner name	Partner short name	Country
1	AustriaTech Gesellschaft des Bundes für technologiepolitische Maßnahmen GmbH	ATE	AT
2	Softeco Sismat S.P.A	SOF	IT
3	Telematix Software, a.s.	TMX	CZ
4	Fluidtime Data Services GmbH	FLU	AT
5	Brimatech Services GmbH	BRI	AT
	Left intentionally Blank		
7	The Regional Organiser of Prague Integrated Transport	PID	CZ
8	POLIS-Promotion of Operational Links with Integrated Services	POL	BE
9	Atos Origin Sociedad Anonima Espanola	ATO	ES
10	PTV Planung Transport Verkehr AG.	PTV	DE
11	Asociacion Cluster Del Transporte Y La Logistica De EUSKADI	MLC	ES
12	Regione Toscana	FIR	IT
13	Reading Borough Council	RED	UK
14	MemEx S.R.L.	MEM	IT

Table of Contents

List of Tables

List of Figures

1 Executive summary

The currently existing bottleneck for the dynamic adaptation of traffic management measures according to policy goals is the information distribution to end users in urban areas and the adaptation of the information provided to the needs of the single user group. Further, current services are not able to deliver any feedback from the end-user to the Traffic Information Service Provider. These aspects are addressed by Co-Cities by providing one standard interface between city traffic information and the Traffic Information Service Providers, the In-Time common interface, secondly, the availability of the full "feedback loop" enables an end to end testing and validation process for the single traffic information service in the cities and elaborates the future expansion steps for cities and service providers.

In the following figure the main extensions of the approach in relation to cooperative services are shown in an overview.

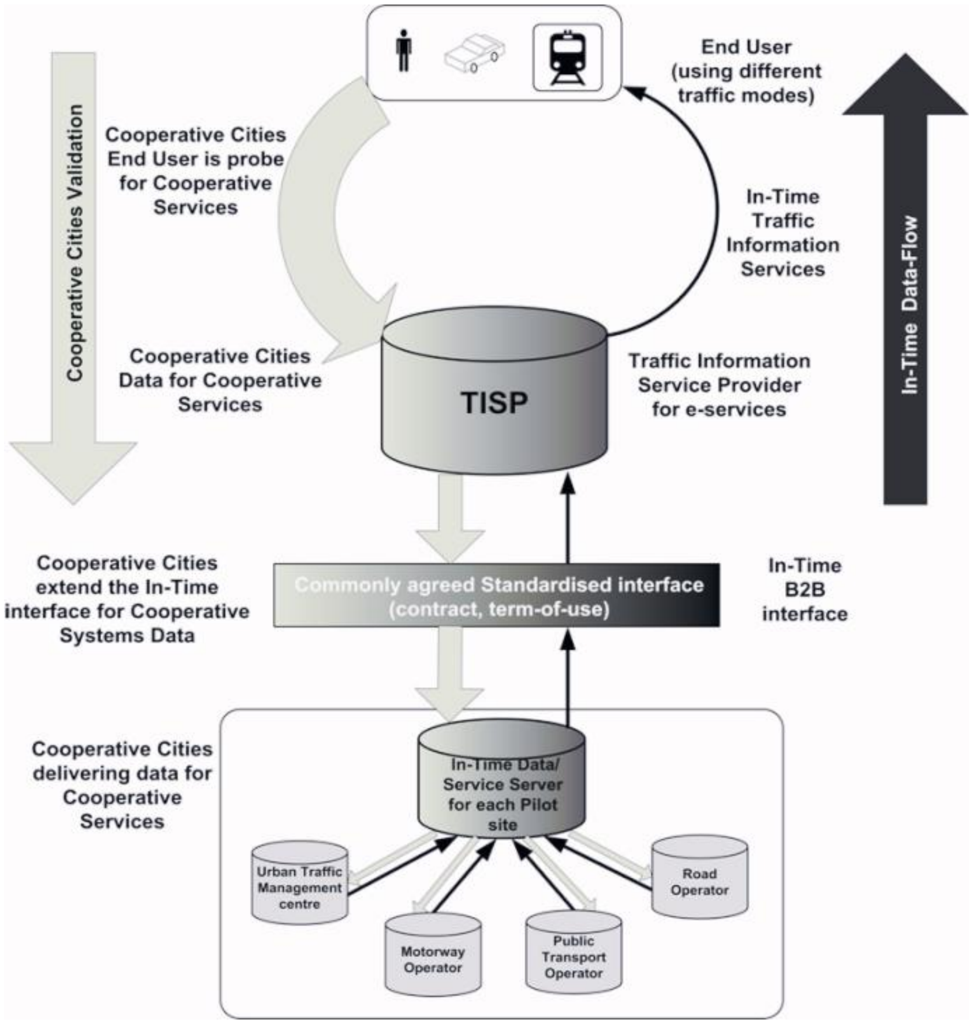

Figure 1: Cooperative cities contribution to cooperative mobility services

The Cooperative Cities proposal is using two main results of In-Time project:

- In-Time Commonly Agreed Standard Data Interface (CAI)
- In-Time Service Definition

The document D3.2 "ITS system interfaces to end users and service delivery chain of the service providers" is one of the deliverables included in the WP3 – SWP3200.

This document aims at describing the main elements involved in the Co-Cities service delivery chain in term of service flow and relationship between these elements as well as the interfaces and data that can be exchanged allowing mechanisms for end user involvement.

One of the main aspects of Co-Cities project is to extend the In-Time CAI (Commonly Agreed Interface) with a 'feedback-loop' to extend mobility services with cooperative elements where the services provision to End-User is based on the In-Time results and the services for feedback loop data provision is the concept developed in the context of Co-Cities project.

Figure 2 presents an overview of the flow of the Co-Cities service delivery chain for services provision to end users which are related to specific service domains, e.g. Road traffic, Point of interest, Public transport, Multimodal journey planning and parking. It should be noted that the implementation of the RDSS (Local systems)/TISPs server blocks depend on the RDSS (Local systems)/TISPs structure and their associated systems.

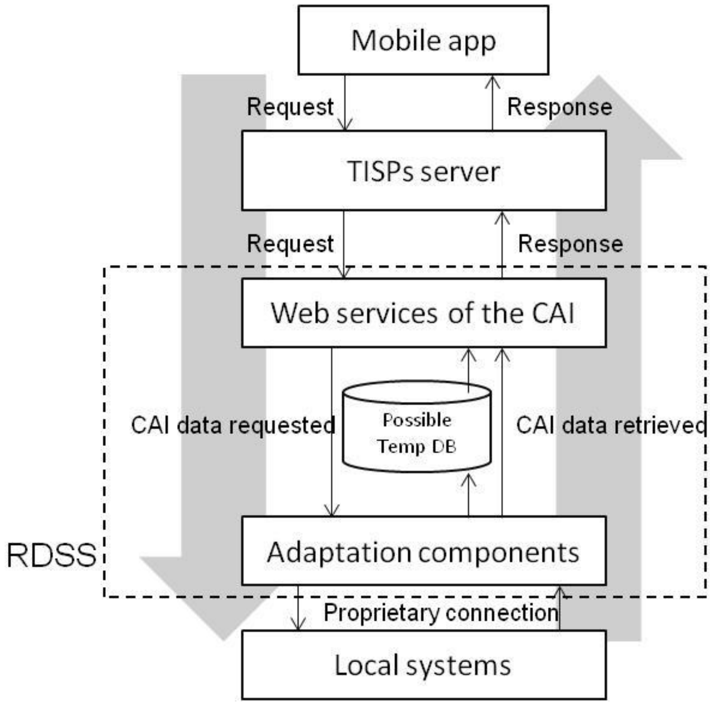

Figure 2: Service chain delivery for provision to the End-User

Figure 3 presents an overview of the flow of the Co-Cities service delivery chain for Feedback data provision to Local system (i.e. Cities system - RDSS) which is the feedback service and data related to the corresponding service domains.

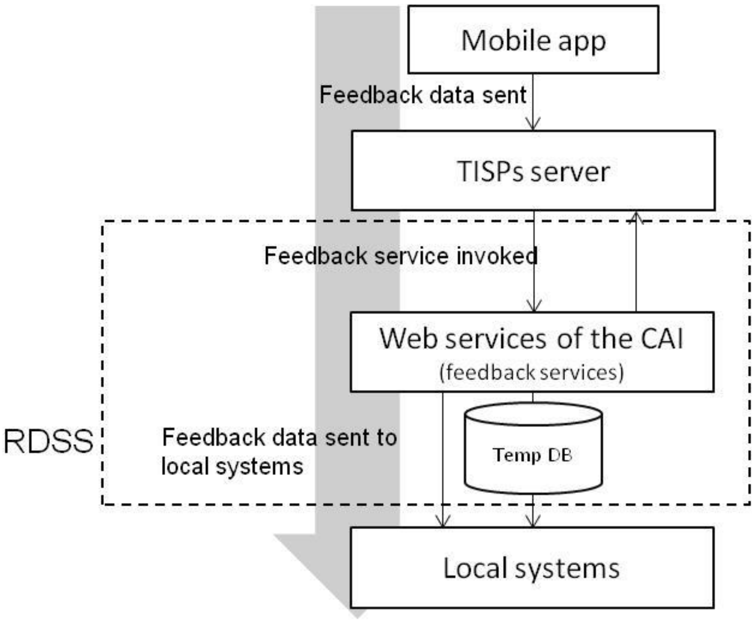

Figure 3: Service chain delivery for Feedback data provision

From the end users' perspective, the TISP Feedback service and the mobile device application will be implemented to allow feedback for each of the defined service domains. While the device should satisfy a set of basic requirements regarding the mobile device capabilities, the TISP mobile application should have improved HMI (Human-Machine Interaction) features allowing an acceptable user experience and usability. On the other hand, the underlying TISP system should integrate the feedback loop engine based on sub-system or components to interface and to manage the feedback service and data. At the TISP server side, it should be possible to do an assessment of the TISP performance using for example a kind of performance monitoring tool in order to detect and avoid some bottleneck issues related to the global TISP system performances.

Figure 4 presents an example of a TISP implementation based on the e-miXer environment [15]. The e-miXer solution provides a service-oriented middleware infrastructure enabling the integration of data/services supplied by different operators in the domain of Traffic and Travel Information.

Figure 4: Overview example of TISP implementation

2 Document structure and links

The content of this document is structured into the following chapters:

- An introduction to the document, including the scope and purpose of the document, and the intended audience.
- An overview of feedback treatment on TISP and RDSS side, describes the feedback model and processing at a high level.
- A concept and guideline of TISP feedback service issues, providing some recommendation related to HMI (Human-Machine Interaction) features, a brief UI (User Interface) guideline and key requirements for UI elements.
- A description of the approach about the data collection and specification related to the categorization of data and services by quality of service and new user data.
- A general description of pre-requisites and recommendation related to the client system, i.e. the mobile device.
- A general description of performance evaluation issues and references on the key performance indicators related to the Co-Cities CAI and the reference platform. A general description is presented for possible monitoring system tools in the context of RDSS/TISP and End-user side.
- Finally, a chapter that summarizes the tasks that have been carried out in this deliverable, describing the work performed and the conclusions reached.

In the following subsections, this task is linked to other tasks in the previous work packages, and also within the overall project.

2.1 Link to the Co-Cities project structure

The work package WP3 is structured in four sub work packages (SWP) which goals are specified as follows in the FPP/DoW (Full Project Proposal/Description of Work):

- SWP 3100 - “Specification of ITS system, interfaces, extensions and modules, reference system”
- SWP 3200 - “Interfaces and mechanisms for end user involvement”
- SWP 3300 - “Specification of ITS system validation from SP point of view”
- SWP 3400 - “Specification of ITS system validation from cities point of view”

The SWP 3200 “Interfaces and mechanisms for end user involvement” of the work package 3 also includes this document D3.2 – “ITS system interfaces to end users and service delivery chain of the service providers” which describes, defines and specifies interfaces to end user device, define service generation process and specifies key performance indicators per service and user group monitoring. The approach is to define interfaces and methods to establish feedback channels from the end user to the RDSS, via the TISP.

The sources of requirements for the Co-Cities ITS system interfaces to end users and service delivery chain of the service providers description are:

- Mainly the results from the Co-Cities SWP 3100 - "*Specification of ITS system, interfaces, extensions and modules, reference system*" [4], the SWP 2100 - "*Service definition and use cases*" [1], the SWP 2200 - "*User group definition and selection*" [2], the SWP 2300 - "*Validation strategy for existing systems, including extensions and reference system test cases*" [3]. But also other results from the In-Time Project [5].
- Information, work done and results of existing platforms and systems implemented by the cities involved in the Co-Cities pilots.
- The FPP/DoW (Full Project Proposal/ Description of Work) as a basic legal reference to be fulfilled.
- The extensive experience of the Co-Cities consortium partners in Traffic and Travel services and management systems, and communication infrastructure.

2.2 Link to the Co-Cities annexes and Co-Cities deliverables

The Co-Cities "ITS system interfaces to end users and service delivery chain of the service providers" encompasses the results of the Co-Cities tasks and the related deliverables as annexes.

Document	Name
D2.1	*"Report of cooperative cities services and set use cases"* where the nature of the dependency is related to the definition and description of the use cases or scenarios and the corresponding services domains and feedback services, as well as the feedback dataset derived from use cases definition.
D2.2	*"List of user groups and interaction process"* where the nature of the dependency is related to the definition and description of the stakeholders, user groups and segment per services, user involvement process, as well as the definition of user interaction channels.
D2.3	*"Validation strategy for existing systems, including extensions and reference system test cases"* where the nature of the dependency is related to the definition and description of the testing and validation strategy overview, as well as the general approach and generic requirements for validation activities.
D3.1	*"ITS system specification description and reference platform for validation"* where the nature of the dependency is related to the detailed description of data and service model of Co-Cities CAI (Commonly Agreed

	Interface), system boundary and general agreements used during the modelling process, specification of the feedback services, as well as a description and specification of the reference platform for validation.

Table 1: Overview about the document links

3 Introduction

3.1 Scope

The deliverable D3.2 was originally scheduled, on the one hand, to provide an interface structure, based on standards, if possible, for the link between the TISP and the End-User devices much alike the detailed specification of the Co-Cities CAI described in the deliverable document D3.1 - *"ITS system specification description and reference platform for validation"* [4]. The following Figure 5 indicates this part of the communication chain in In-Time (as well as the other parts which are described within this document).

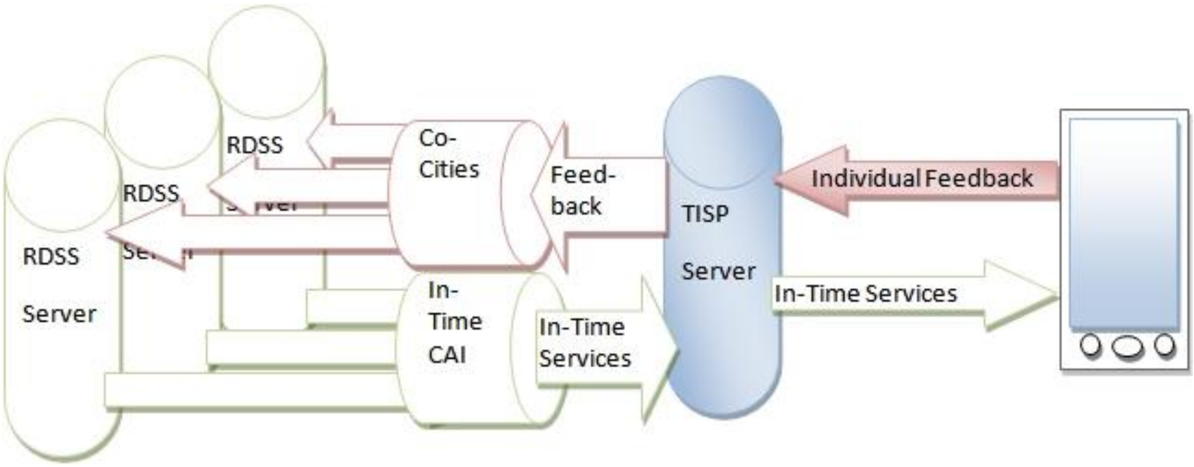

Figure 5: D3.2 Document boundary

As In-Time did not standardize this connection (in contrary to the CAI structure), only the individual feedback was to be considered in the project.

During numerous and extensive discussions in the course of WP3000 it was agreed – or better it had to be accepted by the partners – that this approach is not viable concerning the real market situation. In reality, service providers (TISPs; Traffic Information Service Providers in Co-Cities terms) will utilize their specific service delivery mechanisms which optimizes speed, data volume to be transmitted and other factors to provide a high quality (as perceived by the end user) service to its clients, the end users. These technologies, as the end user applications themselves, are proprietary and also serve as Unique Selling Proposition (USP) factor within the service delivery which is why In-Time did not thrive to come up with any standardized or harmonized approach. The same, of course, is the case for the collection and transmission of feedback as it needs to be deeply embedded in the end user application, which, in turn, is proprietary as mentioned above. Hence, it was decided that the communication link cannot be technically specified within Co-Cities as any common technical specification would not stand any chance of acceptance with TISPs under the prevailing market conditions. What can be done, however, is to define the data and information sets which are to be collected from the end user based on the use cases and data/information sets transmitted via the CAI as described in D3.1.

The very same concern, namely the utilization of proprietary technology which also serves as distinguishing aspect between a given TISP´s offers and those of his competitors applies to the service generation processes of the services on TISP side which cannot be specified in (technical) detail. However, the principle methods of feedback treatment need to be described in chapter 4.

Finally, to guarantee the demonstration and validation activities are comparable between different tests sites present in the project, key performance indicators are provided within this document in chapter 8.

3.2 Intended audience

System architects, information systems designers, system developers and applications, software engineers, and other audiences when designing services and applications taking into account relevant standards and recommendations of standards bodies like IETF, ITU, ISO, W3C, ...

Partners involved in the development and integration tasks, system integrator, peoples who will test, validate and evaluate the Co-Cities pilots and system related with, and all users and stakeholders that will participate in the implementation and the execution of the cities pilots and associated scenarios.

3.3 General remark

This document follows the ISO/IEC Directives, Part 2: Rules for the structure and drafting of International Standards w.r.t. the usage of the word “shall”. The word “shall” (not “must”) is the verb form used to indicate a requirement to be strictly followed to conform to this specification.

In whole of this document, the definition of the terminology “reference platform for validation” or shortly “reference platform” is stated as:

A reference or core system which provide the components, functionalities and applications for:

- Collecting and monitoring the relevant data, characteristics or attributes related to the performance and quality of services and data.
- Validating the elements, services and data related to the service domains and associated scenarios or group of use cases.
- Evaluating and reporting the results of monitored and validated services and data.

In this context, the reference platform should be understood as a “Collecting, Monitoring, Validation and Evaluation System Reference”, as it is defined in this general remark section.

3.4 Background

3.4.1 In-Time project

Co-Cities is closely related to the In-Time project [5], which, based on the eMOTION project [7], defined the so called “Commonly Agreed standardized Interface”, the CAI. This CAI provides a detailed technical description on how well defined services, e.g. for the provision of traffic information, points of interests and (intermodal) routes, are to be presented by a regional provider.

The idea is, that complex local/regional organizational structures which, in reality, deny service providers the use of these data due to extremely high efforts in establishing the contact with each regional provider, setting up separate contracts and following up every change or update in any of the regional system. In-Time provides, via the CAI, a standardized, centralized access to these data and information sets which mean a given TISP would only have to access one common access point for all regional data available. Within Co-Cities, the CAI operator must, much alike in In-Time where he collects the required data from different sources and technical systems, distribute the feedback generated by the end user and forwarded to the RDSS to the correct addressees in the region.

3.4.2 VIAJEO

Another project In-Time liaised with is the VIAJEO [8] project, which is set to establish an open platform concept for service generation utilizing existing, well accepted standards wherever possible. The demonstration of this platform was executed in Athens, Beijing (China) and Sao Paolo (Brazil) and integrated the In-Time service 17 as no common standard was/is available for the provision of intermodal routes. In turn, Co-Cities might take up VIAJEO´s approach for the transmission of FCD (Floating Car Data) data for the same reason, namely that currently no standardized solution exists for these kinds of data (VIAJEO utilized a data model which is used by the largest operating FCD system in Italy involving more than 1M vehicles).

3.4.3 I-TOUR

I-TOUR [9] is focusing on the provision of intermodal traffic and transport services within social network structures to improve service quality and usability. User feedback is essential in the frame of interconnected users sharing of information and experiences and offering rewarding mechanisms for public transport use. Similar services and applications reside in the domains of both projects, as I-Tour offers e.g. an intermodal service. I-Tour is focusing on a more integrated approach than In-Time and Co-Cities, which allows understanding the preferences of the users and support usability by taking these aspects into, account. I-Tour also is expecting advanced services realized introducing public transport means load data and weather conditions to provide a comprehensive traveler information service range.

4 Overview of Feedback treatment on TISP/RDSS side

4.1 Introduction

Within WP2000, a series of use cases was defined which ranged from quality feedback (e.g. rating scheme utilizing 5 stars for quality assessment) to submitting entirely new data sets via Co-Cities´ feedback channel. In principle, a given Co-Cities feedback is either linked to a service delivery, e.g. a route, in which case the feedback information also carries the identifier of the related service delivery, or it is independent from a previously delivered service item which means that a new data set which is relevant for the service generation on RDSS side (may be on TISP side as well) is submitted by the end user, e.g. a new congestion event on the motorway.

For the first class, the content delivered via a service can (as mentioned above) be classified in general or updated, modified or ranked in detail. The latter class comprises all kinds of content ranging from new POIs to new traffic information or schedule data.

According to the In-Time principle, which is the sound basis for Co-Cities, the RDSS is predominantly responsible for the service delivery on regional level and the redistribution of the feedback to the relevant entities and systems. The TISP, on the other side, is directly linked to the end user, his client, to which he delivers services on basis of the In-Time services provided by the RDSS.

The principle schema for a feedback generated on basis of an In-Time service item delivery is depicted in the following Figure 6 (numbers in brackets indicate the succeeding steps):

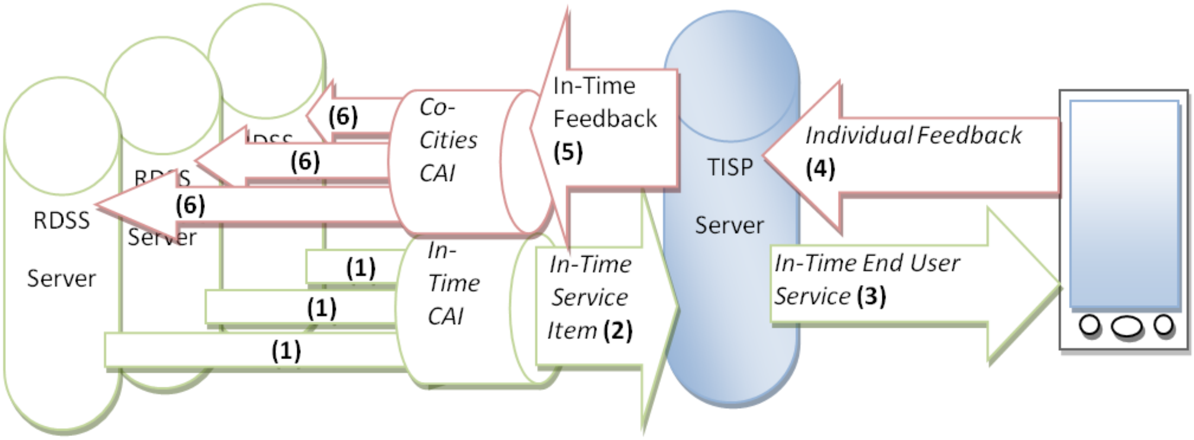

Figure 6: Feedback treatment

Note: computations in the CAI to assemble the RDSS server deliveries into an In-Time service as well as the computations on TISP server and handheld side are not depicted although executed.

The following steps are depicted in the Figure 6 above:

(1): Upon request of the TISP (not depicted) the CAI requests (not depicted) data and information sets from applicable local systems, which deliver those to the CAI server.

(2): The CAI system assembles/computes the data and information sets into an In-Time service and delivers the resulting service item to the TISP.

(3): The TISP delivers the In-Time service or a TISP service built upon the In-Time service item content to the end user.

The steps 1 to 3 are contained in the In-Time service delivery chain while the steps 4 through 6 are residing within the Co-Cities domain.

(4): Upon submission of a feedback by the end user the feedback is transferred to the TISP system.

(5): The TISP system assembles/computes the feedback into the Co-Cities feedback service and provides it to the Extended CAI.

(6): The RDSS, responsible for the CAI and fully aware of the local systems and the organizational ancillary conditions, allocates the feedback to the correct local recipients.

The data and exchange models necessary for the communication between RDSS and TISP for delivery of the In-Time services (step 2) are described within the In-Time documentation. The step 1, however, was not described in In-Time as the technology is “proprietary” meaning, that not only the data exchange methods in a given region may be numerous but also that the organizational background, so the question of which institution to approach for which data, is potentially unique. Of course, this also counts for step 6 which is the local distribution of user feedback received via the CAI in the frame of Co-Cities and which must be distributed locally to the correct local recipients. Step 3 was not described by In-Time either, for the reasoning on proprietary data delivery mentioned above.

Step 4 and it´s ancillary processes are described within this document. The Co-Cities document D3.1 [4] holds the description of step 5, the feedback delivery through the Co-Cities CAI.

The following two examples shall illustrate the chain of events within a typical scenario.

The first one describes a use case, where a user requests a route which is calculated by the RDSS. As the destination is false and a subsequent user feedback is generated, the TISP forwards the user feedback to the RDSS. The RDSS must now allocate the correct recipient in his local network, with the challenge to first identify the problem – the example below assumes that it was a clearly identifiable Geo-coding problem, maybe a wrong coding of the clear name address – and allocates the response to the respective recipient. This step, however, may pose a major obstacle in the processing chain as understanding of the clear reason for suboptimal information or service item is sometimes very difficult. It will thus not always be clear which institutions are concerned with some feedbacks, if services provided by the RDSS involve several institutions within the process chain.

The second example describes a user generated congestion event.

First example:

The End-User requests a route within a region. This request if forwarded (by the TISP) to the CAI and thus the RDSS collects the required information and data sets from the local systems **(1)** and provides it to the TISP via the CAI **(2)**. The TISP sends the route to the end user **(3)**. The end user finds that the destination he is arriving at is not the desired one and provides his feedback to the TISP **(4)**. The TISP formats the feedback according to the Co-Cities specifications and sends it to the CAI **(5)**. The RDSS distributes the feedback to the relevant local system, which is responsible for the Geo-coding of the destination **(6)**.

Second example:

An end user drives on the motorway and runs into a congestion event, which is not yet mentioned by the TISP´s service delivered to him. Hence he creates a new feedback message containing the location and extent of the congestion event and sends it to the TISP **(4)**. The TISP reformats the new data set from his proprietary format into the Co-Cities format and provides it to the CAI **(5)**. Again, the RDSS operating the CAI forwards the new item to the relevant local server/service **(6)**.

4.2 Feedback Model

The information flow along the Co-Cities value chain in a functional perspective involves the following basic parts of the architecture:

A. **TISPs**: mobile apps and other TISP nodes depending on TISP's specific service provision schema (i.e. TISP servers in a client-server infrastructure). TISPs provide B2C services and are responsible for enabling end users to provide feedback data.

B. **RDSSs**: includes all elements and nodes, which enable the provision of data and services from local-specific formats to Co-Cities TISPs via the Co-Cities Commonly Agreed Interface. They also enable feedback data to be received from TISPs and sent to local systems.

Figure 7 depicts the basic parts of the architecture:

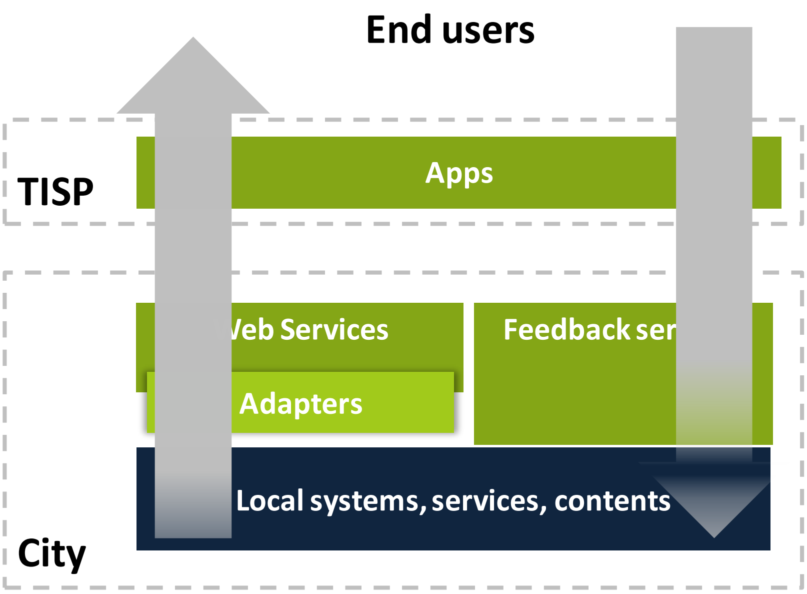

Figure 7: Basic Co-Cities elements

End user services can be mobile or desktop applications, available on a number of different clients, channels and platforms and providing a variety of different services, including In-Time and Co-Cities services. They provide "In-Time-like" services (travel and traffic information) and Co-Cities services feature: a) feedback functionalities and b) travel and traffic information enhanced in a Cooperative way.

The feedback services are web services implemented from a well-defined WSDL produced as part of the Co-Cities specification. They are responsible for accepting service calls from TISPs and make feedback data available to local systems.

The two "directions" of the information flow depicted by the previous Figure 7 can be shortly described as:

A. Service provision to the end user done in "In-Time-like" way (from local systems to TISPs and end users via the CAI).
B. Feedback provision typical of Co-Cities: from end users using TISP apps with specific functionalities to local systems via the CAI (which is enhanced compared to In-Time to handle feedback provision).

Figure 8: Mock-up of a mobile app specific functionality for the provision of feedback data

When feedback data arrives at local systems, it can be used by local systems for several purposes and, in general, re-introduced in the Co-Cities chain to provide enhanced information. In this case the information flow can be brought back to the initial "In-Time-like" service provision.

The information workflow can be described more in detail by referring to the following functional schema (Figure 9), which put in evidence each step of the information provision in all directions.

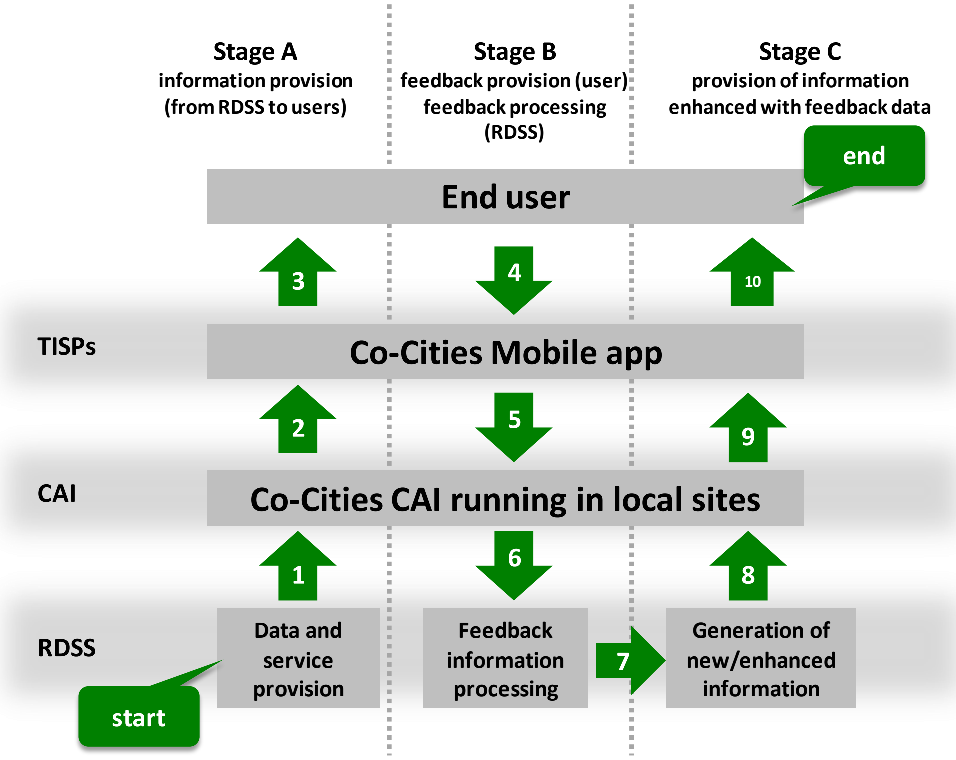

Figure 9: Information flow in Co-Cities

The following steps are related to the above picture.

Stage A: In-Time service provision.

1. Local data, expressed with local-specific data types are converted in the In-Time specific data type by using the In-Time specification. In-Time data become available for service provision along the CAI (see next step).
2. In-Time data is made available to TISPs via the web services compatible with the Commonly Agreed Interface (CAI) specification (WSDL definitions).
3. Data obtained from local sites via the CAI is used for B2C service provision to the end user. The In-Time-like service provision ends at this stage.

Stage B: Co-Cities feedback provision.

4. Via appropriate functionalities available on the mobile app, end users provide feedback for one or more Co-Cities feedback services (Overall quality of service, Feedback on provided data, Provision of new data) and possibly for one or more of the Co-Cities service domains.
5. The mobile app (or the TISP server) invokes the appropriate Feedback Service which is expected to be available as part of RDSS. Their WSDL is part of the Co-Cities technical specification.

6. Each Feedback service should enable feedback data (provided by the TISP) to be made available to the relevant local systems.

At this point the information can be processed in order to produce enhanced information to be re-distributed along the In-Time service chain.

7. The information is processed on the basis of local validation and quality management schemes applying the required policies, algorithms and tools, etc. and a new piece of information is produced.

Stage C: In-Time quality improved service provision.

8. In steps 9 and 10, the newly produced piece of information is re-introduced into the In-Time service provision scheme improving the quality of information.

Note: The exact sequence of operations (requests, responses) and the respective nodes involved may vary depending on the situation. In some cases for example a request from the mobile app can be supplied by a direct (cached) response from the TISP server itself (no request/response is generated to/from RDSS). In other words, besides the above functional view, in practice for each TISP request there may not always be a “direct flow” of information to the local system and vice-versa.

4.3 Feedback processing on TISP server side

In principle, two general methods for the treatment, computation and forwarding of the end user feedback on side of the TISPs are applicable: either the TISP forwards the received end user feedback without further computation through the CAI or he first processes the end user feedback, e.g. to generate aggregated information which might be valuable for the RDSS (in fact, it might be a compensatory element for the RDSS´s service delivery) as the RDSS does not have to execute those computations himself. Thus this step might be a worthy addition to the In-Time business model.

Of course, the aggregation of information must follow specific, sometimes local rules. It might be, that e.g. in Germany a ghost driver warning needs to be processed within a given time frame which is different from the one applicable in the United Kingdom.

Within the two examples mentioned in chapter 4.1 above, the TISP would likely compare the input to the service delivered and/or to the information and data sets he has stored in his databases and the protocols for the respective service delivery to rule out, that the flaw is caused by his systems. He then provides the feedback to the RDSS.

4.4 Feedback processing on RDSS server side

The RDSS, as mentioned above, would have to allocate a given feedback to the correct local addressee, which also contains the identification of the reason for feedback in given situations. This can be a difficult step if several local institutions are involved in the generation of the In-Time service.

It is the responsibility of the local Co-Cities RDSS to retrieve feedback data (by means of the Feedback Services) and dispatch it to the appropriate local stakeholder. Since each piece of information sent to the end user is well identified, an association between the provided data and its related feedback is always possible. An accurate TISP and Local implementation will ensure such association to be correctly in place.

On the basis of the previous descriptions the feedback information workflow can be structured indicatively like in the following diagram:

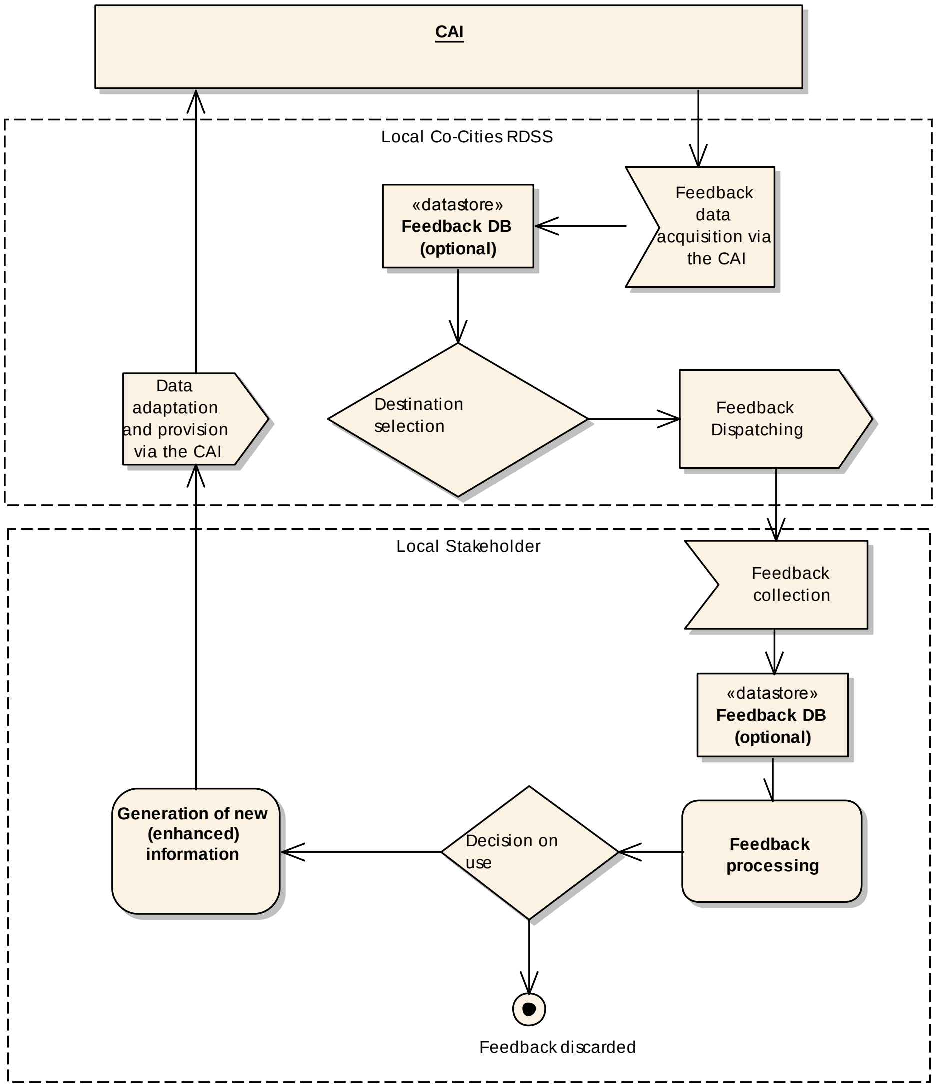

Figure 10: Indicative schema for feedback dispatching and processing

The feedback provision from RDSS to local systems could be seen as a "mirror image" of the conversion from local data to the Co-Cities data types for provision via the CAI to the TISPs. Details on Feedback processing made by local systems depend very much on local policies and system implementation and therefore this part is out of the scope of the Co-Cities technical description.

Local stakeholders receive feedbacks dispatched by the local RDSS, which decides the correct destinations on the basis of policies, and objectives decided in strict coordination between the local RDSS and the local stakeholders.

Feedback data can be collected and used for different purposes by local stakeholders. These purposes may include decisions and actions, which can be possibly carried out in the short, medium or even long term, like:

- Improvement of currently provided information. This is one of the main objectives of Co-Cities. This includes error corrections, re-planning of Journey, provision of real time data instead of planned data (e.g. in Public Transport), etc.
- Creation of new services (e.g. a traffic service is not currently provided because no local system can be used for such type of data but thanks to Co-Cities it becomes possible to give traffic information based on end users' feedback).

Examples of possible associations of feedback data with target stakeholders are depicted in the Figure 11:

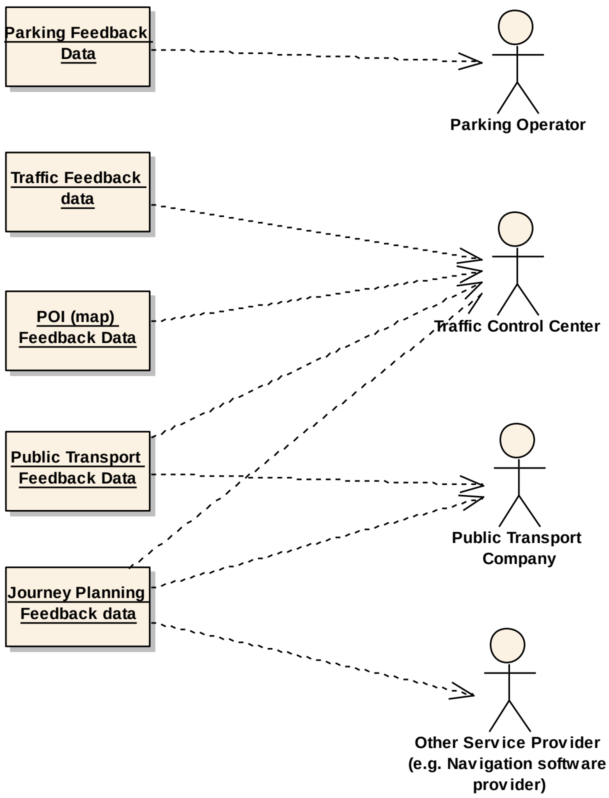

Figure 11: Examples of associations between feedback data and target stakeholders

It should be noted that feedback data could be applicable (and therefore sent) to more than a single local stakeholder possibly depending also on some details of the feedback information itself. The local (RDSS + Stakeholder) policies should take these elements into account. An example of such situation is depicted by the following extract of Use Case: Multimodal Adaptive Journey planning (Table 2):

Cases	Type of feedback information:	Receiver	Possible Purposes
Case A	Misalignment between the planned journey data and the real trip data due to different causes	Traffic Information Service Provider (TISP)	Re-calculate the journey - provide a feedback to the TISP about the quality of mobile service, etc.
Case B	Misalignment between the planned journey and the real trip situation caused by traffic related situations	Traffic Control Center (TCC)	Provide location based information about traffic conditions
Case C	Misalignment between the planned journey and the real trip situation caused by PT service delays, cancellations, etc.	Public Transport operator	Provide information about the quality of the PT service

Table 2: Example situation for Multimodal Adaptive Journey planning

4.4.1 Notes on feedback validation

The validation of feedback data is a process done by local stakeholders. Technically it could be also performed at RDSS level based on policies agreed with the stakeholders. Depending on several factors each piece of feedback information can be classified along a validation schema and rating decided typically by the local stakeholder.

Possible factors to be considered for a validation can be:

- Trust level of the source.
- Type of feedback.
- Number of feedbacks of the same type.
- Others factors, etc.

For example in case of a feedback for traffic domain either:

- New information is generated (e.g. a new traffic event is generated) if a certain number of feedbacks on a traffic event are received in relation to the same location and within a specific timeframe.

- Existing information is corrected when a given number of feedbacks indicate that a provided info is wrong.

In this process of validation it is useful to be supported by appropriate tools, which should be implemented locally.

Basic functionalities of such tools may include for example:

- Possibility to display and manage (e.g. delete) feedbacks by means of lists, map etc.
- Functionalities to aggregate feedbacks using different criteria
- Functionalities to validate feedbacks and to create automatically new information from the validated feedback
- Other tools functionalities.

The following pictures illustrate an example of a supporting tool where user feedbacks about traffic alerts are shown on the screen and possibly on the map. Based on some parameters (ideally under the control of the operator) feedbacks are aggregated and suggested traffic alerts generated.

A special “SEND” function may be available to generate a new traffic alert with pre-compiled data (retrieved by the feedback data). This can be sent to the RDSS and then to the TISP via the CAI.

User Feedbacks

ID	Time	Feedback	Location	Count
00342	15:01	Traffic alert	Praterstern	01
00343	15:01	Traffic alert	Rosenbursenstraße	01
00344	15:03	Traffic alert	Rosenbursenstraße	02
00345	15:03	Traffic alert	Kärntner Ring	01
00346	15:04	Traffic alert	Rosenbursenstraße	03
00347	15:10	Traffic alert	Rosenbursenstraße	04
00348	15:11	Traffic alert	Praterstern	02
00349	15:14	Traffic alert	Praterstern	03
00350	15:14	Traffic alert	Donau City Strasse	01
00351	15:16	Traffic alert	Praterstern	04
00352	15:20	Traffic alert	Praterstern	05
00353	15:21	Traffic alert	Rosenbursenstraße	05

Current Status

Suggested alerts

Traffic info	Location	Last	Count	Operations
Traffic alert	Rosenbursenstraße	15:21	05	SEND DELETE
Traffic alert	Praterstern	15:20	05	SEND DELETE

Settings

VIEW EDIT

Figure 12: Mock-up of a supporting tool for feedback processing

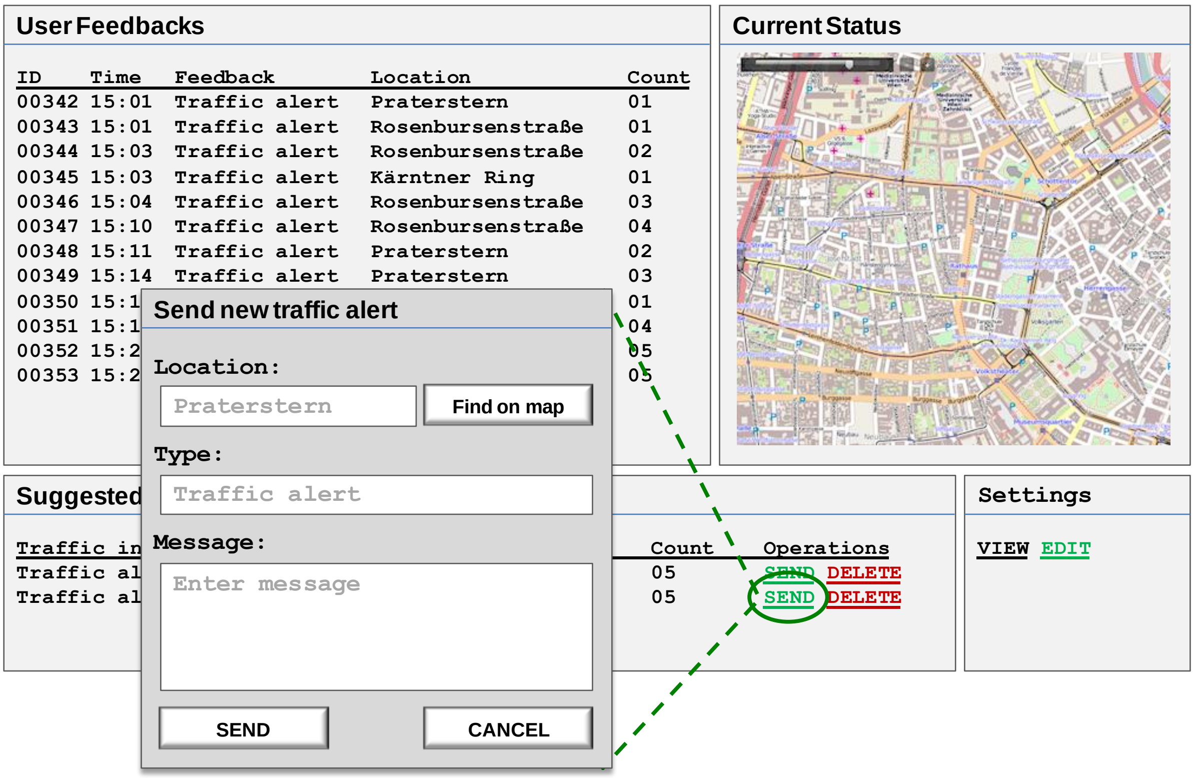

Figure 13: Example of function for enhanced information generation

5 Overview of TISP Feedback service

The TISP feedback services are responsible for collecting feedbacks from end users. The data collection process can be implemented proprietarily as long as data is sent to RDSS through CAI feedback extensions. For example, TISP can implement a custom web service for collecting feedback information from end user mobile phones. Once a TISP feedback service receives a feedback it can be preprocessed and stored in a local database and then send to RDSS in predefined time intervals. Alternatively, the TISP can provide an end user´s feedback upon submission by the end user directly to the RDSS. This flexibility also opens up new business opportunities as feedback data can be “pre-processed” by the TISP according to the needs of the RDSS as reimbursement for the In-Time services provided by the RDSS.

The following chapters describe data collection from the user point of view.

5.1 Recommendation for HMI features

As the Co-Cities project is focused on the feedback from end users it is necessary to consider basic recommendations for human machine interaction within the project.

The main goal of human machine interaction is to improve interaction between humans and computers. HMI can be divided in two parts - a research part and practical part. Researchers are more concerned with developing new design methodologies and new hardware devices. On the other hand designers are more interested in practical part that includes designing graphical user interfaces.

A target group of this project consists of mobile phone users and web users. Furthermore mobile phone users are divided into several other subgroups based on a mobile platform, e.g. iOS, Android, Windows Phone, etc. Therefore HMI for this project should be concerned with a practical application and it is not desired to experiment with a new hardware devices or new design methodologies.

For mobile phones there are official user interface guidelines that should be followed. Like any other design process user interface design can be divided into iterative steps following an iterative design.

1. Design the user interface
2. Test
3. Analyze results
4. Repeat

This process is repeated until a user-friendly interface is created.

Since use cases have already been identified they can serve as a starting point in designing the user interface. TISP will usually demonstrate a limited set of use cases. For example Telematix will demonstrate functionality in Reading using these three use cases.

- Feedback on occupancy in public transport vehicle
- Feedback on public transport location and schedules
- Feedback on overall quality of service.

In designing a user interface these feedbacks should be separated into logical tasks. In other words, user interface screens should be kept simple and never focus on more than one feedback at a time.

If the feedback is requiring more than one screen to be collected, those screens should be logically interconnected following the user interface guidelines for a specific platform.

The goal must be that the user is always aware and in control of what he is doing, confusing the end-user with complex screen designs might result in unwanted feedback submitted not depicting the real impression/opinion of the user.

5.2 User interface guideline

During a process of designing a user interface it is recommended to prepare "wireframes" or so called mockup screens.

The purpose of the wireframe is to provide a blueprint for design and development tasks. The point of creating a wireframe is to reduce costs in software development. It is based on the idea that creating a mockup screen should be faster than actual implementation and any misunderstandings can therefore be identified in early stages of development with minimal costs. This idea is critical and assumes that developers are familiarized with mockup tools and are able to mockup user interface faster than using alternative methodologies.

There are many tools for creating mockups and wireframes. Several factors should be considered while selecting a correct tool for mockups. These factors include price and usability of a program and whether developers already know such tool. Also some tools can be specialized to mockup screens only for one platform, e.g. iOS.

The following non-exhaustive list of tools can be used for a development in this project. However one should not be limited only to this list.

- Balsamiq
- Mockup Builder
- Mockup Tiger
- iPlotz
- Adobe Photoshop CS5
- Interface Builder
- Microsoft Expression Blend

While designing graphical user interfaces one must take existing official user interface guidelines into account. Best practices also recommend using native controls of a selected platform. For example Figure 14 demonstrates usage of native controls in iOS.

Figure 14: Demonstration of native UI elements in iOS

Many UI (User Interface) elements are specific for one platform and therefore the user interfaces need to be designed for each platform separately.

As already stated above, the user interface design for TISP mobile End-User applications is affected by a target platform and set of use cases that will be demonstrated.

In Reading the first developed TISP end user application will run on iOS. A feedback option allowing to comment on the overall quality of service is provided in the settings part of the application as it is illustrated in the Figure 15.

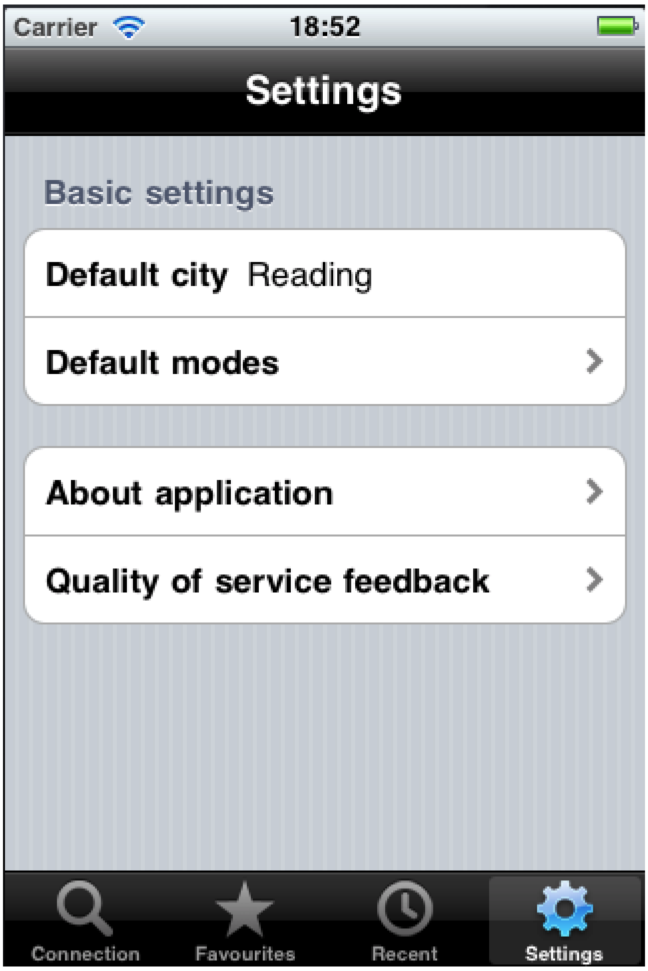

Figure 15: Demonstrates how the user can navigate to quality of service feedback

When the user clicks on a feedback row the application will start a task that will consists of one or more following screens to collect all necessary information. At the end user will be provided with a button to send a feedback to a TISP server.

5.3 Key requirements for UI elements

UI elements are more or less specific to a target platform however there is a subset of UI elements that can be found in all platforms that serve the same or similar function. These elements are textbox, label, button, checkbox and tab. Names of these elements can also vary depending on a platform.

Table 3 summarizes these elements with a description of basic functionality.

UI Element	Usage
Label	Displays read-only text for a user
Textbox	Provides user-editable text field
Button	Starts some predefined action

UI Element	Usage
Checkbox	Provides a boolean on/off switch
Tab	Separates UI into logical units

Table 3: Common UI elements and usage

While designing a User Interface (UI) one should not depend only on this subset of UI elements but should make a full use of the target platform.

6 Data collection and specification

One of the main value propositions of Co-Cities in relation to standard travel and traffic information offerings is the direct involvement of users in the service value chain. In the course of the Co-Cities project, three main groups of end-user driven information are distinguished following the respective service categories identified:

- User feedback on the overall Co-Cities service quality.
- User feedback related to the quality of individual Co-Cities services (e.g. parking, traffic information, routing, etc.)
- Mobility data submitted by end users to the TISP as a basis for the delivery of accurate services (user generated data like availability of parking slots or notification of a traffic jam).

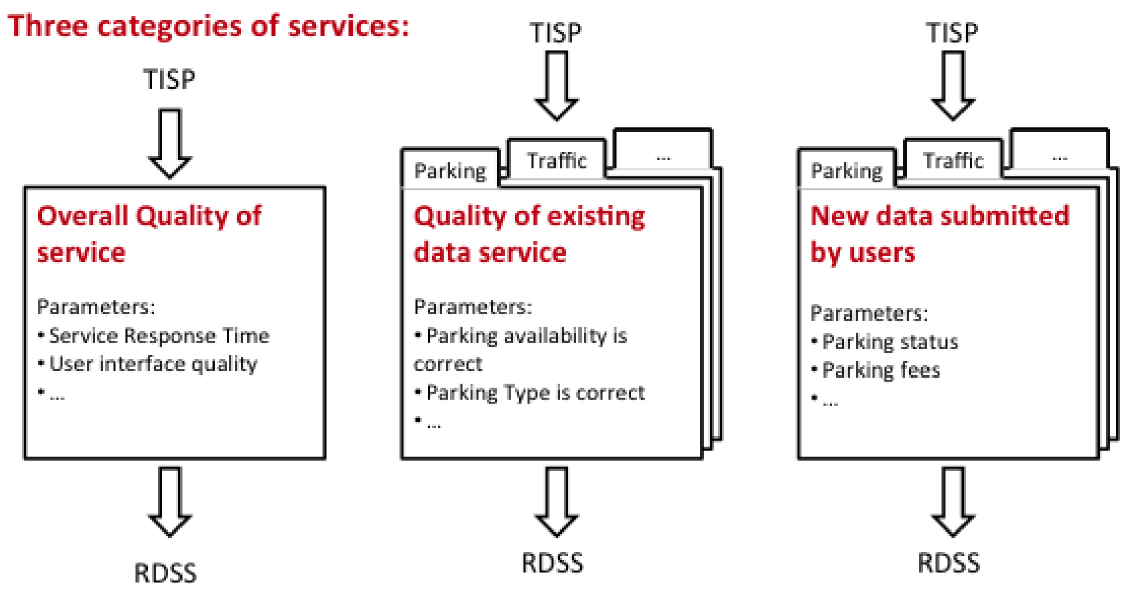

Figure 16: Categories of feedback service

6.1 User feedback on overall service quality

In order to derive a direct user feedback right after the usage of Co-Cities services, a graphical (i.e. icon-based) feedback functionality is embedded in the service interface. Together with a detailed acceptance monitor provided to test users by means of the Web, this approach allows for focused and direct as well as more comprehensive user inputs related to the Co-Cities services and related travel environments.

As a basis for this assessment, the overall quality of Co-Cities services is elaborated by means of a rating right after the usage of services comprising the following items described in the Table 4:

Id	Use case	Quality characteristics/Questions	Feedback type
1-16	All	Data quality Service quality and reliability Service interfaces Possibility to provide user data	

Table 4: Rating right after the usage of services

6.2 Usage-based feedback on individual service quality

Assessing the individual Co-Cities services in more detail, the following data is collected for the user feedback related to the quality of individual services/use cases (based on the Co-Cities deliverable document D2.1 – "*Report of cooperative cities services and set use cases*" [1]).

"Use cases" have been formulated to help identifying the required additional features (in terms of feedback data), which extend the In-Time data model, and at the same time they can be considered as demonstration scenarios in cities.

"Services" on the other hand are the "building blocks" necessary for achieving a demonstration of a use cases. More specifically, each use cases require:

- Some End-user functionalities to be implemented by TISPs (at least one).
- The required feedback services to be implemented locally as part of the CAI.

From the above considerations a distinction between "implementation" of services and "demonstration" of use cases has to be made.

Furthermore, taking into account all use cases (scenarios to demonstrate), not all In-Time/Co-Cities domains have been included in demonstration scenarios (e.g. for instance there is no specific freight or Flight-related use cases).

In Co-cities the technical group decided to group together some In-Time services into the so called "service domains". This decision also was taken on grounds of the observation that in the real-world implementations of In-Time services did e.g. not strictly distinguish between static and dynamic services (e.g. the dynamic parking and static parking was implemented as a "parking" service which in some cases uses static data and in other dynamic data The interface is the same, only the nature of data is different).

In general, to identify the service domains, dynamic and static services are grouped together (dynamic parking + static parking becomes simply "parking", dynamic Public Transport + static Public transport becomes simply "Public Transport" etc.).

Another criteria of identification of service domains of relevance for Co-Cities is related to the multimodal Journey Planning service which "embeds" different kinds of information like walking and cycling routes.

This result in the following viewpoint within In-Time and Co-Cities concerning services:

- Services providing information about a specific domain, “separately” (e.g. the “public transport service” does not provide journey planning but it provides timetables, stop names etc.) these are called “Data Services”.
- The multimodal Journey planning combining the information from single modes of transport including walking and cycling (of course where available).

As the In-Time data model remains 100% valid, which means that the implementation of any In-Time service is feasible (including data services for walking, flight etc.). It should be considered, that Co-Cities does not “cut” anything from In-Time. For the additional part (Feedback), instead, some domains have been considered as more relevant than others: the use cases helped to identify the domains where additional feedback features were necessary and have been added.

Following this philosophy, cycling and pedestrian or walking routes are not considered as separate service and are included in the multimodal journey planning service domain. This issue is described in the deliverable D2.3 [3] – chapter “5.2 Existing service” and in the section “5.3.4 Multimodal journey planning”.

The Table 5 describes in detail, for each clustered use case defined in D2.1, the feedback data that an end user can submit. The table is organized as use cases applying the rule already used in the project (aggregating the delivery of walking and cycling information in the multi-modal journey planner use case).

Id	Use case	Quality characteristics / Questions	Feedback type
1	**Parking information in urban areas**	Correctness of data: • Parking availability • Parking information (vehicle type, opening hours, parking duration, fees) • Info on entry point for parking • Guidance to parking place	
		Quality of service: • Subjective response time • Map representation • Possibility to provide user data • Objective response time	 Measurement
2	**Road Side Parking**	Correctness of data: • Parking availability	

Id	Use case	Quality characteristics / Questions	Feedback type
		• Parking information (vehicle type, opening hours, parking duration, fees) • Info on entry point for parking • Guidance to parking place	
		Quality of service: • Subjective response time • Map representation • Possibility to provide user data • Objective response time	 Measurement
3	**Traffic Info / Traffic jam**	Correctness of data: • Information on traffic jam • Info on the cause of the traffic jam • Info on delay caused	
		Quality of service: • Subjective response time • Map representation • Possibility to provide user data • Objective response time	 Measurement
4	**Speed limit information**	Correctness of data: • Information on speed limit	
		Quality of service: • Subjective response time • Map representation • Possibility to provide user data • Objective response time	 Measurement

Id	Use case	Quality characteristics / Questions	Feedback type
5	**Toll information**	Correctness of data: • Information on road charging • Info on fees • Info on type of payment	
		Quality of service: • Subjective response time • Map representation • Possibility to provide user data • Objective response time	Measurement
6	**Road works**	Correctness of data: • Road works information • Info on the status of the road works • Info on part of the road affected	
		Quality of service: • Subjective response time • Map representation • Possibility to provide user data • Objective response time	Measurement
7	**Provision of traffic information**	Possibility to provide info on speed limits Possibility to provide info on parking Possibility to provide info on tolling Possibility to provide info on road works Possibility to provide info on traffic jams Map representation	

Id	Use case	Quality characteristics / Questions	Feedback type
8	**Map-related interaction**	Correctness of data: • POI information • Address information • Parking information • Journey planning	
		Quality of service: • Subjective response time • Map representation • Possibility to provide user data • Objective response time	 Measurement
9	**Estimated time of arrival**	ETA (Estimated Time of Arrival) information	Automatic feedback (time difference between ETA and actual time of arrival)
10	**Visual destination information**	Correctness of data: • Visuals on destination location	
		Quality of service: • Subjective response time • Map representation • Possibility to provide user data • Objective response time	 Measurement
11	**Destination information**	Correctness of data: • Info on destination	

Id	Use case	Quality characteristics / Questions	Feedback type
		Quality of service: • Subjective response time • Map representation • Possibility to provide user data • Objective response time	Measurement
12	**Last mile map on mobile device**	Correctness of service functionality and data: • Enquiry on last mile map demand • Display of last mile map • Info on current location and destination	
		Quality of service: • Subjective response time • Map representation • Possibility to provide user data • Objective response time	Measurement
13	**Public transport location and schedules**	Correctness of data: • Locations of stop points • Info on public transport schedule for selected stop	
		Quality of service: • Subjective response time • Map representation • Possibility to provide user data • Objective response time	Measurement

Id	Use case	Quality characteristics / Questions	Feedback type
14	**Dynamic road traffic info, Dynamic traffic event info**	Correctness of data: • Info on road network traffic conditions • Location for traffic events	
		Quality of service: • Subjective response time • Map representation • Possibility to provide user data • Objective response time	Measurement
15	**Adaptive real-time multimodal journey planning**	Correctness of data: • Multimodal journey planning result • Info on transport means • Info on start point • Info on stop point • Info on interchange point(s) • Info on trip segment(s) • Re-calculation of routing result based on situation • Estimated time of arrival	
		Quality of service: • Subjective response time • Map representation • Possibility to provide user data • Objective response time • Matching of routing result and actual route	Measurement Automatic collection of GPS data

Id	Use case	Quality characteristics / Questions	Feedback type
16	**Occupancy of public transport vehicle**	Correctness of data: • Info on occupancy of public transport vehicle	
		Quality of service: • Subjective response time • Possibility to provide user data • Objective response time	 Measurement

Table 5: Data collection for the user feedback (quality of individual services/use cases)

In order to allow the collection of user feedback related to individual Co-Cities services, a matching between the respective use cases consumed and the feedback questions displayed is needed in the End-User service application.

6.3 User generated mobility contents

In project deliverable 3.1 - D3.1 - *"ITS system specification description and reference platform for validation"* [4], a description of the data and service model necessary to extend the in-Time CAI by adding the necessary Co-Cities feedback functionalities is given.

This description gives the detailed indication of all the possible data types sent via the CAI from the end user application (TISPs) to the RDSS. In other words, user generated mobility data types are potentially all data types 'understood' by the CAI which is the only channel Co-Cities TISPs can use to make this data available to Co-Cities RDSSs.

In terms of practical implementation, feedback services may be set up in different ways from TISP to TISP depending on different technologies, user interfaces, etc. Moreover, the completeness of each data-set provided for a given domain and service may differ from TISP to TISP and from Site to Site because of several reasons and local conditions (e.g. data is complete in one case and less complete in others). This can be considered a normal situation and it's technically managed thanks to a specification, which prescribes a minimum number of mandatory attributes for each service and additional non-mandatory attributes to give more details if possible.

Also, the concrete meaning of the feedback in specific aspects can be flexibly applied to different TISP/RDSS pairs. As an example, the RDSS and TISP should agree on the approach for positive confirmation of information provided via the In-Time service chain (e.g. "true" is only fed back if the user actively approved an information set) to avoid misinterpretations on side of the RDSS.

Thanks to this flexible mechanism the implementation of feedback functionalities of the three types is easier for those situations where certain data is unavailable or difficult to retrieve. Having introduced the maximum possible extension of user generated mobility data as the entire data collection defined in the data model and having remarked that some differences may exist in terms of user data provision from TISP to TISP and from site to site, a few additional indications can be formulated to suggest what user data should be concretely generated in order to effectively cover the Co-Cities use cases as defined in WP2 which remain the reference global demonstration scenario for Co-Cities.

A difference between “Service” and “use case” may be formulated as follows:

- Use Case: scenario, defined in an informal way depicting a possible use or demonstration of a combination of services.
- Service: group of specific functionalities provided by a system built from well-defined technical specifications.

The main criteria for an effective identification of services to be implemented in order to enable the practical realization of the group of use case should ensure that:

- All Co-Cities Service Domains (introduced in Deliverable 2.1) are covered.
- All three types of feedback services are covered.

Once these constraints are fulfilled, the reference set of minimum user data, for each domain, which can be considered in order to reasonably define an object in one of the In-Time/Co-Cities domains (like a new traffic event) can be found in the ***In-Time simplified model*** defined in : **In-Time Data Model (Package In-Time Simplified Model).**

The approach for defining the In-Time simplified model is described in In-Time Deliverable D3.2.1 (section 5) [6].

For example taking into account the **Use Case 7 – Provision of Road Traffic Information** one of the aspect to cover is to permit the user to inform the cities regarding traffic events that he’s experiencing.

To support this use case the “New Data” feedback service (see the description in Deliverable 3.1) for the Traffic Domain must be implemented by the RDSS and correctly supported by the TISPs.

A TISP application must then present a UI which includes, at least, the mandatory field defined by the Co-Cities model for feedback.

The app enables the selection of the domain (traffic) and the feedback function where a user can indicate (generate) new data related to a traffic event.

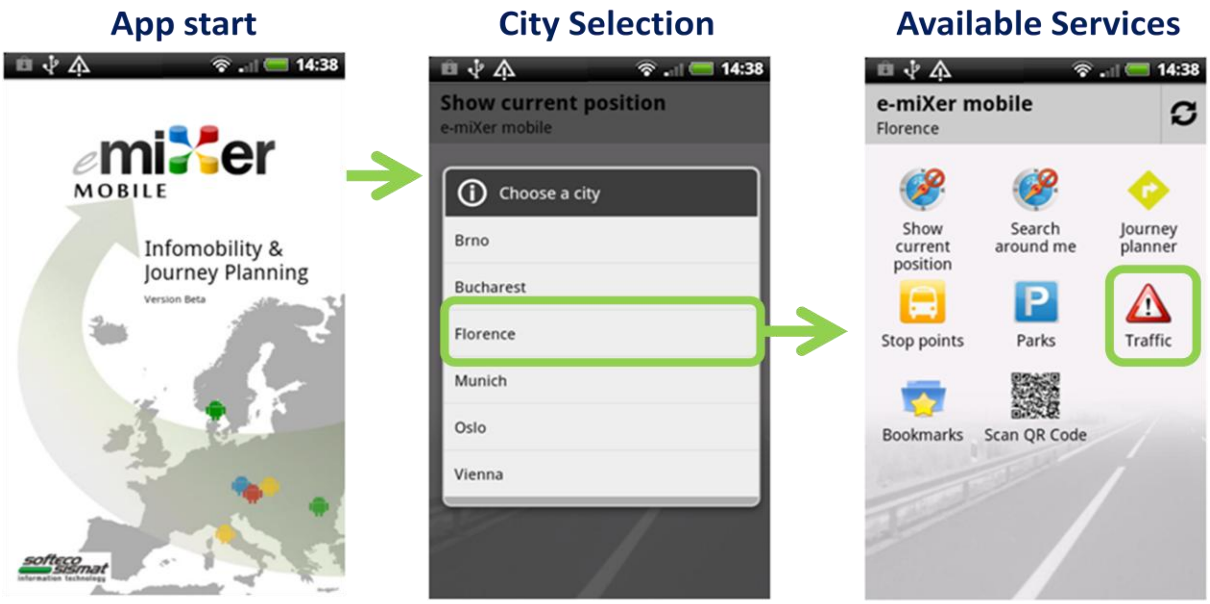

Figure 17: Step a – selection of traffic functions

The supported functionalities by a specific city are then showed to the user.

In case of the use case in example the functionality (send of new data) can be presented to the user using a dedicated icon under the Traffic domain. For other types of feedback the approach should be different: the delivery of quality information is subsequent to the receiving of some data by the user.

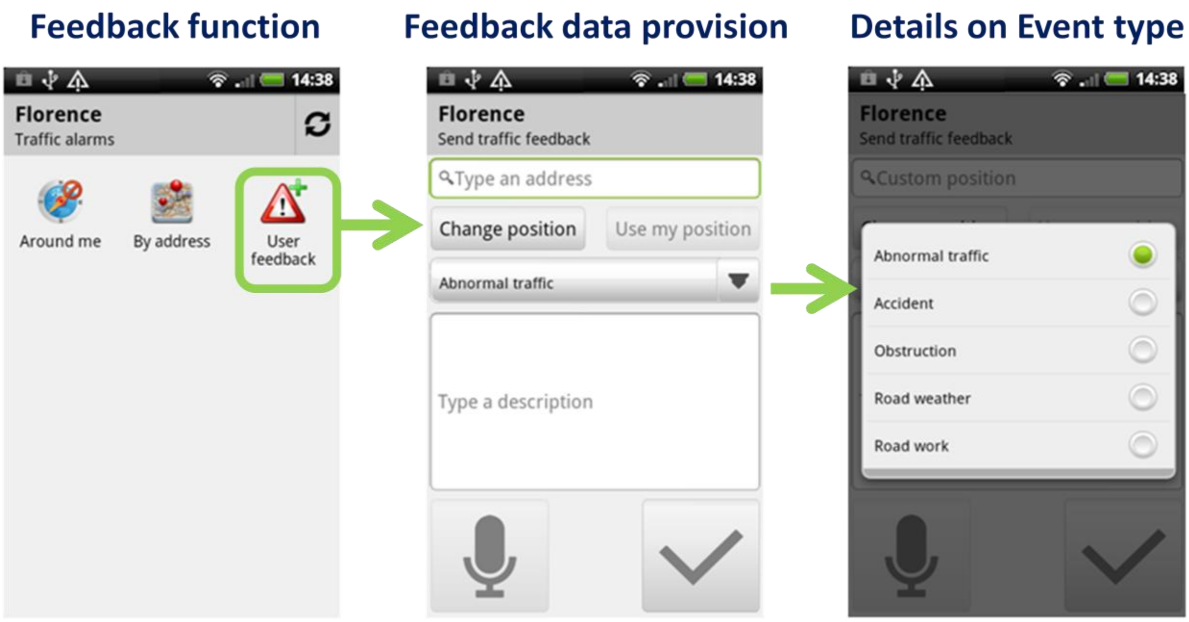

Figure 18: Step b – indication of traffic related data

While developing the user interface for the TISP application, a basic strategy must be applied. The form presented to the user must focus on the very minimum set of information, which is considered mandatory. This is necessary in order to avoid confusion on side of the end user. Hence, asking him for very detailed information which are practically optional, a more complete form can be showed using a “more options” icon or similar technique rather than within the standard workflow scheme.

In the example we are considering the user-generated data composing a new traffic includes:

- The Type of event
- Its Location
- A Date and Time
- A Trust level (this is not explicitly indicated by the user but it’s provided by the TISP)
- Some Free text for additional description, comments etc.

Once the user clicks on a “Send Button”, the information is transmitted from the mobile device to the TISP and then this data is sent to the local system using the Co-Cities Common Interface.

At the RDSS side the feedback is received, processed and an acknowledge message is then sent to the TISP to confirm the reception.

The local treatment of the feedback data, of any type (both quality and new data) is something, which is strictly related to the local procedures and local systems, which are different from city to city.

For this reason in Co-Cities there is no detailed analysis on how the data coming from the end users should be used at local level.

7 Client system and mobile device issues

In the Co-Cities system the mobility aspect is fundamental, since the main concept of Co-Cities is to allow user feedback that enhances the user involvement related to traffic and travel services; to provide their proper point of view and comments about the accuracy and pertinence of the information, data and services that they receive as consumers of these services; and also the consumption of services and content on the move through mobile devices (or terminals) in a system based on the interconnection of mobile users to TISP.

A Co-Cities mobile user can receive constantly updated traffic and travel information, but these same users also will be able to give information about the quality, consistency and usefulness of this information through the feedback loop system, following the concept of always-on connected. Due to this the connectivity and the communication system must be very reliable and offer high performance.

7.1 Mobile device pre-requisites and features

The mobile device will be an important piece inside the Co-Cities system, thus the devices and mobile terminals must have some specific requirements, depending on their capabilities and capacities, in order to support the features and functionalities relating to the fundamental concepts of Co-Cities in terms of the following categories of supplementary requirements as recommendations:

- Usability, ease of use, acceptable response times (responsiveness) for the end-user, ease of the interactivity with the terminal (e.g. good ergonomic, shortcuts, gestures…).
- Visibility of the basic information and necessary help screens or visual assistance (e.g. wide display and screens, high resolution and high colors quality) to obtain an acceptable user experience and to send information feedback to the user about what has to be done (including sound, highlighting, animation and their combinations).
- Intrinsic requirements to the mobile device such as the capacity and duration of the battery (to increase the time of use), quality and strength of materials used for manufacturing and construction, the aspect design and the aesthetic, a set of media communication to access networks by different way and maintaining if possible the connection and sessions with the remote service or resources (i.e. related to the concept of ubiquity and always-on connectivity).

In relation to the user and the terminal interactivity (or at the level of the mobile terminals) one of the main requirements is the usability concept that is described in the study of interaction between user and computer usability or machines (HCI: Human-Computer Interaction [10][11]). This is fairly described by some literature, documents, papers and reports, recognized studies available on the Web (Internet).

One of the parameters to be considered is the response time relating to any action of the user on the mobile terminal (by keyboard and pointer interface, haptic interface, vocal interface,...) and the result of the expected functionality or sensitive effect (visual and sound effect).

Responsiveness is therefore considered as an essential usability issue for Human-Computer-interaction (HCI). The rationale behind the responsiveness principle is that the system should deliver results of an operation to end users in a timely and organized manner. Long delays can be a major cause of user frustration, or can lead the user to believe the system is not functioning, or that a command or input gesture has been ignored. The frustration threshold can be quite different, depending on the situation (i.e. the type of expected function or process).

In the context of Co-Cities scenario it is assumed that the mobiles terminal may have these suitable pre-requisites and features:

- Open Operating system (OS) where it is possible to access services and applications that support multi-task system and inter-process communication, user interface framework, standardized platforms for integration of different technologies (Web, XML, HTTP, socket TCP/IP client and server,…), a flexible environment for application running and runtime that support the maximum portability of application, support of common and open developer tools (SDK, library, API, 3rd party component,…).
- Processor capacity of CPU (multi core, high frequency and process unit, high CPU performance).
- High capacity of internal memory and as well as external memory extension or others storage system.
- Ad-Hoc networks (WiFi, WIMAX, Bluetooth,…).
- Set of radio communication technologies (4G - LTE, 3G, UMTS, GPRS, CDMA, and its variants…).
- Wide and comfortable display with a high screen resolution and high color quality.
- Integrated keypad and touch screen with a practical navigation system.
- Integrated location system such as a GPS capability and others movement sensor (angle, acceleration, translation, gyroscope based system, etc.). This allows having LBS (Location Based Service) contextual information provided from the user mobile that can be exploited by the service provider (i.e. the TISP) to accurate the data and services provided to the user.

Co-Cities can make services more attractive and appealing to end users, e.g. to be mobile and to have the opportunity to access traffic and travel services on-the-fly and providing user valued information about the quality of these services, only smart phones are reasonable for the Co-Cities project. For the Co-Cities project it is essential to keep device and user interface restrictions in mind. Mobile device input is often difficult when compared with use of a desktop device equipped with a keyboard ("Mobile Web Best Practices" [14]).

Current existing mobile systems, especially smart phones, which are in focus of Co-Cities service domains, come with a broad range of capabilities through wireless communication technologies. The Co-Cities project will provide several pilots and demonstrators based upon new generation of mobile device. As of the first quarter of 2012, Android & iOS based devices hold the 82% (source IDC) of the market share.

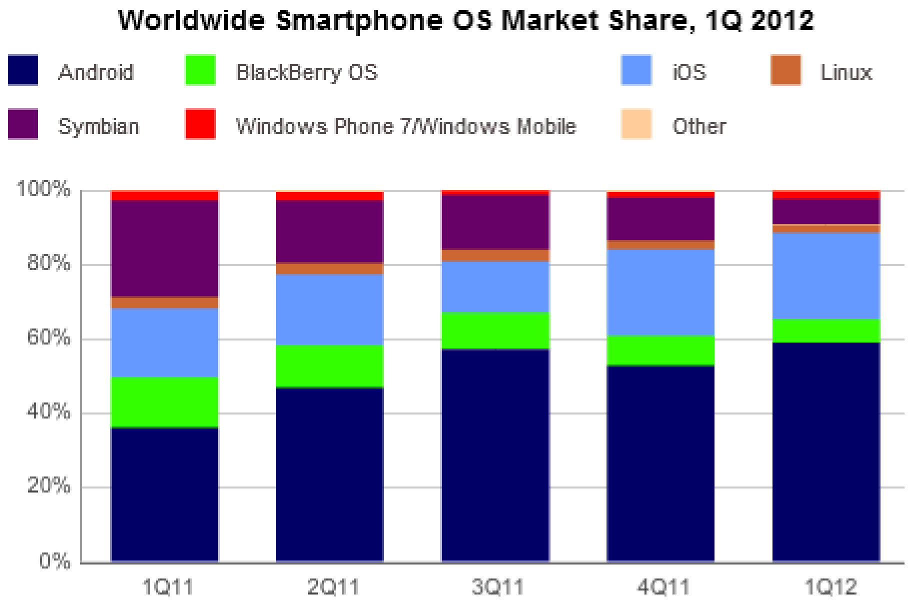

Figure 19: Worldwide Smartphones OS Market Share, 1Q 2012 (source: IDC)

The trend in one year (1Q2011 to 1Q2012) shows an increase of devices sold powered by these two Operating Systems of a 30% total. For this main reason, at least initially, the focus of Co-Cities end-user application will be in these two technologies, which practically represent the most parks of currently used mobile devices.

The Table 6 summarizes the main minimum characteristics that a Co-Cities mobile device should have (focusing on Android and iOS devices):

Feature	Description
Operating system (OS)	Mobile device based on Android platform v2.3 or later, iOS v5 or later (iPhone OS) platform
Processor capacity of CPU	• ARM processor and architecture (Cortex-A8 MP core 600Mhz or later) • QualComm Snapdragon processors (600Mhz or better) • NVIDIA Tegra 2 or later • others
Memory capacity	Memory: 256MB or better

Feature	Description
	Storage: 2GB or better
Networks	WiFi, Bluetooth
Radio communication	4G, 3G
Display	Display dimension: 3.2" or better Display resolution: 320x480 or better
Keyboard and touch interface	Integrated keypad and touch screen with a practical navigation system
Integrated location sensor	GPS capability and others movement sensor

Table 6: Generic mobile device features

7.2 Overall system pre-requisites and features

System requirements for Co-Cities was based on the results of use cases scenarios analysis, users' roles and needs analysis, but the whole system of Co-Cities related to the system in general as well as related to the mobile device and its embedded system and the network service infrastructure may have these suitable pre-requisites and features:

- Usability: Accessibility (ease of access to and use of specific functionality), ease of use (ease of learning and using the system), UI consistency (consistency of the user interface, both within the system and with other systems to avoid ambiguity). Usability is a key aspect from a TISP point of view. The end user application is responsible in presenting the data to the user. If data is not correctly displayed user can interpret the not consistent representation as incorrect information coming from the city.
- Performance: Throughput (e.g. the number of transactions per minute), response time (how fast the system responds to events), utilization of resources (utilization of memory, disk space, database storage, and so on), capacity (the number of users that the system can accommodate). Cities that want to deliver their data towards the CAI infrastructure must ensure a certain performance level in order to permit TISPs to deliver information as fast as possible to end users.

Others considerations related to the definition of the System Requirements are listed below:

- Scale and Scope: The problems to be addressed by the Co-Cities Architecture are large in two important respects. On the one hand, they might involve a large number of heterogeneous elements (such as users and stakeholders, data and models, mobile devices and terminals). On the other hand, each such element may itself be large in size. The former is referred to here as "*Scope*", while the second is called "*Scale*". Co-Cities systems can be ready for future growth of need to access data. In that sense both TISPs and RDSS must consider preparing their system to easy scale to support new users interested in information provided by the CAI.

- Long life time: A Co-Cities system is a system which needs to operate over a long period of time. The anticipated period of operation of a service network is longer than typical technological cycles in IT partly due to the evolutionary character of the IT services and architectures, mobile devices and technologies.
- Quality: There is need for a service to support the distribution of quality information. Therefore the Co-Cities Architecture should provide a model, which addresses confidence. A quality situation such as the one on the World Wide Web (in which information quality is not generally known) is not acceptable for a Co-Cities service network. Levels of confidence need to be attached to data, services, providers, etc.
- Access Control: Organizations (regions, administrations, enterprises, services providers…) are reluctant to grant data access to other organizations, even within the same government or country. One technical reason for this is that there are no common strategies and technical solutions for handling access privileges across organizational borders within loosely coupled systems in a practical, transparent and reproducible way.
- Communications: Needs to provide high connectivity and a good infrastructure of interconnection of all the elements involved in the system in an optimal way (e.g. Application server, Web server, Geo-server, routers, content management server (CMS), Database server, Cluster system, Load balanced system…).

Note that these sets of pre-requisites and features can also be applicable and valid for the telecommunication operators and/or network operators (telecom providers, service and web housing) as well as the RDSS and TISP operator.

8 Performance evaluation issues

This section describes the performance evaluation issues related to the TISP and the End-user side, taking into account that the TISP side is mostly proprietary and they can only be understood as black boxes. The same applies for the link between TISP and the end-user, it is proprietary and all technical details are unknown because the complete information is not available (e.g. network settings for data handling by network operators).

An approach before the performance evaluation, selection and deployment of a performance monitoring tools is to identify characteristics, features and data that can be considered as performance indicators and in order to establish a kind of performance monitoring plan.

A performance monitoring plan is a critical tool for planning, managing, and documenting data collection. It contributes to the effectiveness of the performance monitoring system by assuring that comparable data will be collected on a regular and timely basis. These are essential to the operation of a credible and useful performance-based management approach.

Various performance definitions can be found in the literature. For example, in the computer domain, the performance is characterized by the amount of useful work accomplished by a computer system compared to the time and resources used. Another aspect related to networks, is the network performance, which refers to the service quality of a telecommunications product from the point of view of the customer or user.

Often the system performance is related to the reliability of a system and to the quality of service (QoS) of this system. Quality of service awareness where certain services provide similar functionalities but with a different quality (performance, reliability, cost, etc.), e.g. a user can connect to a Traffic Information gateway to access their services, by using his Smartphone through Wi-Fi and 3G Networks. The choice of the network depends on the QoS properties of each network. On the other hand, the elements system, platform and network services infrastructure underlying the TISP system is also involved in the overall performance and quality of service.

The Real-time performance monitoring to identify poorly or under-performing systems, components or services has become an integral part of preventative maintenance. Among others, rising energy costs, the user requirement in terms of usability and response times, and the increasing demand for improved product quality are elements to take into account, where the analysis of the performance and the associated tools help to optimize the system and allow detecting performance problems that can be resolved. Automatic process control solutions that incorporate real-time monitoring and performance analysis are fulfilling these needs.

The main goals of monitoring and control system include the following:

- Observe the “health” of IT services.
- Take remedial actions that minimize the impact of service incidents and system events.
- Understand the infrastructure components responsible for the delivery of services.

- Provide data on component or service trends that can be used to optimize the performance of IT services.

Any system performance evaluation relies on a set of key performance indicators, calculated using some specific software tool. Both, the key performance indicators and the suggested tools in Co-Cities are described in the next chapters.

8.1 Key performance indicators (KPIs)

This section references the key performance indicators (KPIs) introduced in the document D2.3 – *"Validation strategy for existing systems, including extensions and reference system test cases"* [3].

The Chapter 7.5 *"Testing and validation criteria and methodology"* of the deliverable D2.3 describes the testing and validation criteria and associated data and indicators managed by the reference platform for validation where an overview of the validation process is presented and the common validation test is described.

The performance indicators related to the capabilities available in the reference platform are focused on the Co-Cities CAI (Commonly Agreed Interface).

As a summary, the common key parameters to be measured and evaluated in the context of the service domains and the Co-Cities CAI are essentially categorized as the following:

- **Automated CAI tests**: On the server side, the CAI (referred to as "interface") tests that will be conducted are characterized by the following features:
  - **Availability**: This test examines whether the interface is available. This test is passed successfully if the interface responds to a predefined request without errors.
  - **Conformity**: This test examines whether the interfaces conform to the In-Time and Co-Cities specifications. This test is passed successfully if the XML data is validated against the relevant XSD (XML schema definition) without errors and contains the expected data.
  - **Level of Service (LoS)**: This test examines the stability of the evaluated interface. Over the period of several days the interface is queried in regular intervals and responds in 99,5% of cases within set time limits of 2 seconds and does not exceed the maximum time limit of 20 seconds. The level of service is the proportion of successful queries divided by the total number of queries conducted.
  - **Response Time**: This test examines the average response time of the interface. Over the period of several days the interface is queried in regular intervals. The average response time is calculated as the average of all queries conducted and the test is passed if the average time is below 1 second, the maximum does not exceed the time limit of 20 seconds, respecting the minimum number of 50 tests.
- **Client tests including CAI monitoring**: On the client side the following tests will be conducted while monitoring the CAI with the help of the reference platform:

- **Availability**: This test examines whether the information is sent to the CAI or is displayed on the end-user device. This test is passed successfully if the data is sent to the CAI and the information is displayed on the end-user device.
- **Conformity**: This test examines whether the interfaces conform to the In-Time and Co-Cities specifications. This test is passed successfully if the XML data is validated against the relevant XSD (XML schema definition) without errors and contains the expected data, or the information is displayed according to the defined interface specifications.

- **User feedback analysis**
  - **Quality**: The quality of the services will be analyzed and validated with the help of the provided user feedback. Overall more than 90% of positive user feedback is the target value to roll out services in full city areas. The gathered data will be used to calculate performance indicators for relevant parameters.

Also in the deliverable D2.3, chapter 7.3.7 “Strategic objectives and expected results” and in the section 7.3.7.2 “Performance indicators overview”, there is an overview of the performance indicators summarized in the corresponding Table 12 – “Principle Clusters of Performance Indicators”.

8.2 Monitoring performance issues related to RDSS/TISP proprietary system

Due to the fact that implementations at TISP and RDSS side differ and are mostly proprietary, they can only be understood as black boxes. The scope of the validation is limited as the links from the TISP to the final customer (e.g. a traveler) are proprietary and can therefore not be compared in all technical details because the complete information is not available.

As stated and summarized in the section 3.1 of this document, the service providers (RDSS/TISPs in Co-Cities terms) will utilize their specific service delivery mechanisms and specific technologies, as well the end user applications themselves, are proprietary. This makes difficult to establish and access performance indicators, monitoring and evaluating these proprietary system, where the access to the resources (data, internal services), characteristics and specifications, software and hardware, server and network are restricted from external customer, actor or stakeholder.

Nevertheless it can be possible to integrate non-intrusive monitoring systems tools to perform a generic monitoring with the minimum impact in these proprietary systems, if they agree and give the necessary permissions and authorizations.

8.2.1 Generic performance indicators

This section describes a set of generic performance indicators that can be applied in systems based on client/server architecture and focused on the server side.

Each performance indicator needs a detailed definition including the unit of measurement and the associated data source, where the indicator source is the entity from which the data are obtained. Performance measures are an important part of optimizing and maintaining system performance through performance monitoring. These monitoring measures can be based on simple statistics or complicated model-based performance criteria and algorithms.

There are a wide variety of technical performance metrics or indicators that indirectly affect overall system performance.

The Table 7 indicates some of the typical characteristics, features or relevant indicators:

Characteristics or features	Description	Measure
Throughput	Network throughput is the average rate of successful message, data or packet delivery over a communication channel.	bits per seconds (bit/s or bps)
Response time	How fast the system responds to events/requests	Milliseconds
MTBF	Mean time between failures (MTBF) is the predicted elapsed time between inherent failures of a system during operation. MTBF can be calculated as the arithmetic mean (average) time between failures of a system.	MTBF = ∑(start of downtime – start of uptime)/(number of failures)
Server available resources	Mainly related to CPU capacity CPU usage is a measured ratio between the CPU sitting idle and actually doing "work". It can be also associated as a number of instruction units executed by seconds and by processes. Essentially it depends of the processor hardware technology (number of core, cache level and size, frequency, clock rate,…)	% related to the CPU usage

Characteristics or features	Description	Measure
	Memory usage	% related to the total memory or absolute in Megabytes
	Available disk space	Megabytes or Gigabytes
	Database storage	Megabytes or Gigabytes
Current server behavior	Current number of users connected to the server	Integer number
	Current number of open connections in the server	Integer number
	Current number of processes	Integer number
	Current number of threads	Integer number
Maximum server acceptance	Maximum number of users connected to the server	Integer number
	Maximum number of open connections in the server	Integer number
	Maximum number of processes	Integer number
	Maximum number of threads	Integer number

Table 7: Variety of technical performance metrics or indicators

8.2.2 Performance monitoring tools

On the market there are many solutions to measure and to monitor the performance of a system.

Some of the relevant criteria or characteristics related to a monitoring solution/system in the Co-Cities context are:

- The cost of monitoring tools: It is suitable to use an open source or low cost solution.
- The solution should be easy to deploy, to configure and to setup.

- The solution should be scalable, reliable, adaptable and extendable.
- The monitoring system should be based on non intrusive tools on the host to monitoring.
- The monitoring system should allow sending alerts and provide tools for events management.
- The monitoring system should provide reporting tools (graphics, statistics) and should allow to presents the information, data monitor and indicator performances based on web interfaces.

As an example the Table 8 enumerates several exemplary monitoring tools as a non-exhaustive list. These tools use different KPIs included in the previous Table 7 to assess system performance and visualizing and reporting the status and data monitoring performance.

Monitoring Tool	Resource or reference	Description
Ganglia	http://ganglia.sourceforge.net/	Ganglia is a scalable distributed monitoring system for high-performance computing systems such as clusters and Grids. It is based on a hierarchical design targeted at federations of clusters. It leverages widely used technologies such as XML for data representation, XDR (External Data Representation) for compact, portable data transport, and RRD tool (Round-Robin Database tool) for data storage and visualization. It uses carefully engineered data structures and algorithms to achieve very low per-node overheads and high concurrency. The implementation is robust, has been ported to an extensive set of operating systems and processor architectures, and is currently in use on thousands of clusters around the world. It has been used to link clusters across university campuses and around the world and can scale to handle clusters with 2000 nodes. Ganglia is a BSD-licensed open-source project.
Nagios	http://www.nagios.org/	Nagios (XI) is a powerful IT infrastructure monitoring solution designed for scalability and flexibility. Nagios is an enterprise-class monitoring and alerting solution (focused on the enterprise and organization monitoring issues). Nagios provides monitoring of all mission-critical

Monitoring Tool	Resource or reference	Description
		infrastructure components – including applications, services, operating systems, network protocols, systems metrics, and network infrastructure. There is a Free License For Small Environments (Free Trial) but otherwise the product should be purchased.
NimSoft	http://www.nimsoft.com/	Nimsoft Monitor solution improves service quality and reduce the costs of IT service delivery. Nimsoft Monitor is an IT monitoring solution architected for modern infrastructures. Nimsoft Monitor provides an unparalleled array of capabilities that speed deployment, unify management, and maximize performance and uptime. Nimsoft has a variety of flexible licensing and pricing models. A free Trial product can be acquired by download.
Zabbix	http://www.zabbix.com/	Zabbix is an enterprise-class open source distributed monitoring solution for networks and applications. Zabbix is an "All In one" Solution that offer True Open Source software, Performance monitoring, agents for all platforms, agent-less monitoring, availability and SLA reporting, collection of any data and great graphs and network maps. Zabbix software is released under the GNU General Public License (GPL) version 2. If Zabbix is used in a commercial context such as profit by its use, it is requested to further the development of Zabbix by purchasing some level of support.
Zenoss	http://community.zenoss.org/index.jspa	Zenoss Core is an open source IT monitoring product that delivers the functionality to effectively manage the configuration, health, performance of networks, servers and

Monitoring Tool	Resource or reference	Description
		applications through a single, integrated software package. Zenoss is driven by the Zenoss community. Zenoss is released under the GNU General Public License (GPL) version 2.
Argus	http://argus.tcp4me.com/	Argus is a system and network monitoring application, which allows monitoring nearly anything that can be monitored (TCP + UDP applications, IP connectivity, SNMP OIDS, Programs, Databases, etc). Argus has a clean and intuitive web interface, it allows to send alerts by numerous ways (such as via pager). Argus was originally designed to monitor servers and network connections in a mission-critical ISP (Internet Service Provider) environment, and scales well from small-businesses through large enterprises, supporting a wide variety of protocols and network applications, and support redundant multi-server configurations. Argus is open-source available at no charge.

Table 8: Performance monitoring tools

Below, the core features of the Ganglia monitoring system are presented for its relevance as open-source solution related to monitoring system for high-performance computing systems:

- Ganglia is an open source scalable distributed system monitor tool for high-performance computing systems such as clusters and grids. It uses carefully engineered data structures and algorithms to achieve very low per-node overheads and high concurrency. This tool can fit perfectly with the TISP servers because is able to measure distributed services, so it is a good option to be used within the Co-Cities project.
- Ganglia is a non-intrusive monitoring tool due to it is not necessary to install an heavy software or bundle on the server to be measured. It is enough to run an executable agent or daemon on each server node to monitor, and to start for using it.
- Some of the KPIs used in Ganglia are the number of CPUs, the response time of a server or the current situation of the memory of the server in each moment.

- The ganglia system includes a PHP-based web front-end, command line tools, libraries, and a few other small utility programs. It can provide some graphics, data and statistics to show the current situation of the target server behavior, as illustrated in the Figure 20.

Figure 20: Example of Ganglia Web Front-End tools

The tools mentioned in this section are examples from a wide range of solutions, the RDSS and TISP should select applicable tools according to the ancillary conditions at hand.

9 Conclusions

The main objective of this task was to provide the principles schema of the feedback treatment, as well as, an overview of the feedback process on TISP and RDSS side. A feedback model has been depicted describing the information flow along the Co-Cities value chain following the basic parts of the architecture.

This document has also tried to some extent describe the mechanisms responsible for service provision related to feedback issues from the perspective of end users by establishing recommendations for characteristics related to human-machine interaction and recommending a guideline for user interface issues on mobile devices always in the perspective of the use cases that have been identified.

Another issue that has been addressed is the main value propositions of Co-Cities related to the possible data and specifications involved in the service value chain and the categorization according to quality criteria and their usage based feedback on individual service quality and related to the identified use cases.

Viewed from the perspective of end users, these users will consume the services that the TISP offers via the TISP network infrastructure and at the level of the user with his own mobile device. In the same way the user may provide feedback with its comments and ratings of the services received.

This implies that the mobile device used should have minimum characteristics, as pre-requisites and features that allow so simple and fast action without much effort from the user to interact with the mobile to access these services and to give his opinion or assessment of the service.

Finally, performance evaluation aspects and suitable performance indicators were introduced taking into account that the RDSS/TISP systems (network infrastructure, services and applications) are proprietary and should be considered as a black box. The limited access to its structure and internal resources pose a challenge for an in-depth analysis of the performance directly by means of the tools and the Co-Cities system. A list of generic performance indicators has been presented but these indicators need to be refined and will be selected, defined and specified in the scope of the next work package (WP4000) and related tasks. Also a list of non intrusive tool has been proposed to allow capturing some TISP performance information.

10 Glossary

The glossary provides the coherent terminological framework used in this document.

10.1 Abbreviations

API	Application Programming Interface
CAI	Commonly Agreed Interface
CDMA	Code Division Multiple Access
CPU	Central Processing Unit
DoW	Description of Work
ETA	Estimated Time of Arrival
FCD	Floating Car Data
FPP	Full Project Proposal
GPL	GNU – General Public License
GPRS	General Packet Radio Service
GPS	Global Positioning System
GUI	Graphical User Interface
HCI	Human-Computer Interaction
HMI	Human-Machine Interaction
IaaS	Infrastructure as a Service
ICT	Information and Communications Technology
IP	Internet Protocol
IR	Infrared
IT	Information Technology
ITS	Intelligent Transportation System
J2ME	Java 2 Platform, Micro Edition
KPI	Key Performance Indicators
LoS	Level of Service
LTE	Long Term Evolution
MTBF	Mean Time Between Failures

NFC	Near Field Communication
OGC	Open Geospatial Consortium
OIDs	Object Identifiers
OS	Operating System
PaaS	Platform as a Service
PI	Performance Indicators (PIs)
PT	Public Transport
QoS	Quality of Service
RRD (tool)	Round-Robin Database tool
RDS-TMC	Radio-Data-System - Traffic Message Channel
RDS	Regional Data Services
RDSS	Regional Data / Service Server
RFID	Radio Frequency IDentification
RTPI	Real Time Passenger Information
RTTI	Real Time Traffic Information
SaaS	Software as a Service
SIM	Subscriber Identity Module
SNMP	Simple Network Management Protocol
SOA	Service Oriented Architecture
SVVP	Software Verification and Validation Plan
SWP	Sub Work Package
TCP	Transmission Control Protocol
TISP	Traffic Information Service Providers
UDP	User Datagram Protocol
UML	Unified Modeling Language
UMTS	Universal Mobile Telecommunications System
USP	Unique Selling Proposition
W3C	World Wide Web Consortium
WIMAX	Worldwide Interoperability for Microwave Access

WP	Work Package
WSDL	Web Services Description Language
XDR	External Data Representation
XML	Extensible Markup Language
XSD	XML Schema Definition

10.2 Terms and definitions

This section provides definitions of any terms that may be needed in order for the reader to understand the terminology used in the document. The author should define any definition/acronym or technical term used in the document that may be unfamiliar to the reader, and it is best to err on the side of too many rather than too few definitions. This also allows the author to frame a word within a specific context, which provides the reader with a common understanding of the author's definition.

Acceptance and trust

Acceptability indicates the degree of approval of a technology by the users. It depends on whether the technology can satisfy the needs and expectations of its users and potential stakeholders. Within the framework of introducing new technologies, acceptability relates to social and individual aspects as well.

Accounting

Process of gathering information about the usage of resources by subjects.

Application

Use of capabilities, including hardware, software and data, provided by an information system specific to the satisfaction of a set of user requirements in a given application domain.

Application Domain

Integrated set of problems, terms, information and tasks of a specific thematic domain that an application (e.g. an information system or a set of information systems) has to cope with.

Architecture (of a system) [ISO/IEC 10746-2:1996]

Set of rules to define the structure of a system and the interrelationships between its parts.

Availability

Availability refers to the degree to which a system, subsystem, or equipment is in a specified operable and committable state at the start of a mission, when the mission is called for at an unknown, i.e., a random time. So, availability is the proportion of time that a system is in operating condition.

Capability

Capabilities are a set of functionalities, through a combination of software and hardware, used to provide services and data. They can reside in the terminal itself as embedded capabilities or they can be available through the network services and infrastructure and others communication technologies (Bluetooth, USB, serial port,…) as external capabilities.

Data acquisition

Methods of data acquisition in FOTs include methods to collect background data, digitally acquire data from sensors, and subjective data (such as data acquired from questionnaires). In addition, data in the form of manually or automatically transcribed data and reductions of collected data is also considered sensor acquired data (but with a manual sensor – the analyst).

End-User

All users that are involved in an application domain and that use the applications, the services built by the system users according to the service Architecture.

Feature [derived from ISO 19101]

Abstraction of a real world phenomenon [ISO 19101] perceived in the context of an Application. In this general sense, a feature corresponds to an "object" in analysis and design models.

Framework [http://www.opengeospatial.org/resources/?page=glossary]

An information architecture that comprises, in terms of software design, a reusable software template, or skeleton, from which key enabling and supporting services can be selected, configured and integrated with application code.

Open Architecture [based on (Powell 1991)][12]

Architecture whose specifications are published and made freely available to interested vendors and users with a view of widespread adoption of the architecture. An open architecture makes use of existing standards where appropriate and possible and otherwise contributes to the evolution of relevant new standards.

Performance indicators definition (PI)

PIs are quantitative or qualitative measurements, agreed on beforehand, expressed as a percentage, index, rate or other value, which is monitored at regular or irregular intervals and can be compared with one or more criteria.

Platform (Service)

Set of infrastructural means and rules that describe how to specify service interfaces and related information and how to invoke services in a distributed system.

Examples for platforms are Web Services according to the W3C specifications including a GML profile for the representation of information or a CORBA based infrastructure with a UML profile according to the OMG specifications.

Resource

Functions (possibly provided through services) or data objects.

Reliability

Reliability is the ability of a system or component to perform its required functions in routine circumstances, as well as hostile or unexpected circumstances, under stated conditions for a specified period of time.

Reference Platform

In the context of the Co-Cities Project, the term "reference platform for validation" or shortly "reference platform" refers to a core system developed in Co-Cities that is the main testing tool for validation and acts as "Collecting, Monitoring, Validation and Evaluation System Reference".

Service [ISO 19119:2005; ISO/IEC TR 14252; http://www.opengis.org/docs/02-112.pdf]

Distinct part of the functionality that is provided by an entity through interfaces.

Software Component [derived from component definition of http://www.opengeospatial.org/resources/?page=glossary]

Software program unit that performs one or more functions and that communicates and interoperates with other components through common interfaces.

System [ISO/IEC 10746-2:1996]

Something of interest as a whole or as comprised of parts. Therefore a system may be referred to as an entity. A component of a system may itself be a system, in which case it may be called a sub-system.

Note: For modelling purposes, the concept of system is understood in its general, system theoretic sense. The term "system" can refer to an information processing system but can also be applied more generally.

System User

Provider of services that are used for an application domain as well as IT architects, system developers, integrators and administrators that conceive, develop, deploy and run applications for an application domain.

Terminal

Terminals are a mobile device that is capable of running mobile services and/or mobile applications.

Use case

A common definition of use cases is the one described by Jacobson (Jacobson et al (1995) [13]): *"When a user uses the system, she or he will perform a behaviourally related sequence of transactions in a dialogue with the system. We call such a special sequence a use case"*. In Other words, a use case is a textual presentation or a story about the usage of the system told from an end user's perspective.

The use cases provide some tools for people, with different skills (e.g. software developers and non-technology oriented people), to communicate with each other. The use cases are general descriptions of needs or situations that often are related to basic scenarios and that are independent of the technologies and implementations of the underlying system.

User

Human acting in the role of a system user or end user of the service and system.

Web Service

Self-contained, self-describing, modular service that can be published, located, and invoked across the Web. A Web service performs functions, which can be anything from simple requests to complicated business processes. Once a Web service is deployed, other applications (and other Web services) can discover and invoke the deployed service.

11 References

The following references are used as background documents for the preparation of this document. References are categorized standards (i.e. standards and specifications from the consortium working groups or alliances and specifications or drafts standardization bodies) and other documents, publications and technical or scientific books.

[1]	Co-Cities project. Cooperative Cities extend and validate mobility services. Deliverable D2.1 *"Report of cooperative cities services and set use cases" ("Service definition and use cases")*, January 2012.
[2]	Co-Cities project. Cooperative Cities extend and validate mobility services. Deliverable D2.2 *"List of user groups and interaction process" ("User groups and interaction process")*, January 2012.
[3]	Co-Cities project. Cooperative Cities extend and validate mobility services. Deliverable D2.3 *"Validation strategy for existing systems, including extensions and reference system test cases"*, March 2012.
[4]	Co-Cities project. Cooperative Cities extend and validate mobility services. Deliverable D3.1 *"ITS system specification description and reference platform for validation"*, March 2012.
[5]	In-Time project. Intelligent and efficient travel management for European cities. http://www.in-time-project.eu/ http://www.in-time-project.eu/en/library/deliverables/
[6]	In-Time project. Intelligent and efficient travel management for European cities. Work package 3, deliverable - D3.2.1 – "*Specification Document of In-Time server, interfaces and protocols*". Section 5 – "*Use of the In-Time Technical Specifications*".
[7]	eMotion project. European-wide multi-Modal On-trip Traffic InformatION. http://www.emotion-project.eu/
[8]	VIAJEO project. 'VIAJEO' for more efficient travel, transport planning. Project Co-funded by the EC DG Research for Specific International Cooperation Actions (SICA). http://www.viajeo.eu/
[9]	i-Tour project. Intelligent Transport system for Optimized URban Trips. http://www.itourproject.com/web/
[10]	HCI, Human systems Interaction, http://en.wikipedia.org/wiki/Human–computer_interaction
[11]	Human-Computer Interaction by Steve Love, IEEE Distributed Systems Online, vol. 7,

	no. 10, 2006, art. no. 0610-ox006.
[12]	Powell, D. (Ed.) (1991). Delta-4: A Generic Architecture for Dependable Distributed Computing. Re-search Reports ESPRIT. Project 818/2252 Delta-4 Vol.1. ISBN 3-540-54985-4 Springer-Verlag 1991
[13]	Jacobson, I., Bylund, S., Jonsson, P., and Ehneboom, S. (1995), "Modeling with Use Cases: Using contracts and use cases to build plugable architectures". Journal of Object Oriented Programming, Vol. 8, No. 2, pp. 18-24.
[14]	World Wide Web Consortium (W3C), "Mobile Web Best Practices", W3C Recommendation, Version 1.0, July, 2008, http://www.w3.org/TR/2008/REC-mobile-bp-20080729/.
[15]	e-miXer product and solutions. e-miXer is a flexible environment enabling integration, processing and delivery of Multimodal Real Time Traffic and Travel Information services. http://www.softeco.it/emixer/home.aspx

11.1 Normative references

OGC Standards and Specifications	Open Geospatial Consortium specifications http://www.opengeospatial.org/standards
OMG specifications and Standards	The OMG (Object Management Group) consortium. OMG's modeling standards, including the UML (Unified Modeling Language). http://www.omg.org/

11.2 Documents and books

W3C (2004). Web Services Architecture.	W3C Working Group Note 11 February 2004. http://www.w3.org/TR/ws-arch/