HORIZON2020 FRAMEWORK PROGRAMME

TOPIC EUK-03-2016

"Federated Cloud resource brokerage for mobile cloud services"

D2.2
Use Cases Analysis

Project acronym: BASMATI

Project full title: Cloud Brokerage Across Borders for Mobile Users and Applications

Contract no.: 723131

Workpackage:	2	Complete Analysis, Architecture, Integration	
Editor:	Young-Woo Jung		ETRI
Author(s):	All Partners		
Authorized by	K. Tserpes		ICCS
Doc Ref:	D2.2		
Reviewer	K. Tserpes		ICCS
Dissemination Level	Public		

Document History

Version	Date	Changes	Author/Affiliation
v.0.1	21-11-2016	Created ToC	Young-Woo Jung / ETRI
v.0.2	15-12-2016	Added Mobile Virtual Desktop Use Case, Das Fest Use Case, Trip Builder Use Case	Young-Woo Jung, Ganis Z Santoso / ETRI Frank Sembowski, Antonia Schwichtenberg, Richard Wecker / CAS Franco Maria Nardini, Patrizio Dazzi, Emanuele Carlini / CNR
v.0.3	20-01-2017	Revised Das Fest Use Case	Frank Sembowski, Antonia Schwichtenberg, Richard Wecker / CAS
v.0.4	24-01-2017	Revised Trip Builder Use Case	Franco Maria Nardini, Patrizio Dazzi, Emanuele Carlini / CNR
v.0.5	25-01-2017	Revised Mobile Virtual Desktop Case	Young-Woo Jung, Ganis Z Santoso / ETRI
v.0.6	26-01-2017	Revised Mobile Virtual Desktop Use Case and added Introduction and Executive Summary	Young-Woo Jung, Ganis Z Santoso / ETRI
v.0.7	10-02-2017	Revised Trip Builder Use Case	Franco Maria Nardini, Patrizio Dazzi, Emanuele Carlini / CNR
v.0.8	21-02-2017	Revised All Chapters	Young-Woo Jung, Ganis Z Santoso / ETRI
v.1.0	29-11-2017	Document Finalization	Young-Woo Jung, Ganis Z Santoso / ETRI

BASMATI Glossary

Term/Acronym	Definition
Mobile cloud services	Online services offered by cloud resources to support mobile apps. The backend of the mobile apps.
CP	Cloud Provider. The actor that provides the cloud infrastructure/resources, such as VMs
CSP	Cloud Service Provider. The actor that provides cloud services on top of a rent infrastructure from a CP
Cloudlet	Limited capacity infrastructures with virtualization capabilities, often used to support a limited amount of users or perform a limited set of operations on behalf of the central cloud infrastructure that hosts the complete application
Edge resources	Resources aimed to operate specialized functionality, located at the "edge" of the network infrastructure, thus, closer to the end users. Examples are (clusters of) RaspberryPis or cloudlets
BUDaMaF	BASMATI Unified Data Management Framework
KE	Knowledge Extractor
DM	Decision Maker
RB	Resource Broker
MVD	Mobile Virtual Desktop
DASFEST	An 3-day long music festival taking place in Karlsruhe, Germany every July
ACE	Amenesik Cloud Engine. The cloud service deployment tool through which actual federation is achieved
BEAM	BASMATI Enhanced Application Model. An extension of the TOSCA specification
ASP	Application Service Provider. A Federation user that rents resource services in order to provide an Application services to End-users
Brokering	The matchmaking support provided by BASMATI platform to decide about the best cloud resources to exploit for the execution of the back-end of BASMATI applications. This activity regards the placement of the services or data on computational resources and storages belonging to the cloud data centre and the cloudlets within the federation.
End user	A user who benefits the various application and infrastructure services provided by the Cloud. Within BASMATI, the most typical example is exploiting the Cloud federation via a mobile device (possibly a laptop) using specialized apps or a web browser.
Offloading	The ability of BASMATI platform supporting the runtime placement of the components composing the front-end of BASMATI applications on edge resources available nearby the end user. This activity takes place both when edge and mobiles exchange one each other their own workload or when such devices transfer some workload to the clouds or cloudlets. In BASMATI we often distinguish Front-end offloading, related to the mobile part of application, from Back-end offloading, concerning the server side of applications. The latter roughly translates to the known concept of Cloudbursting.
QoE	Quality of experience. It is a measure of a customer's experiences with a service. It may be related to some aspects of the QoS and QoP, but can also take into account other metrics.

Service handover	Service handover refers to the activity of transferring an active service between two computational resources (e.g. Cloudlets) with minimal or no disruption on the availability of the service. Ideally, service handover is transparent with respect to the user.
Situational Awareness	The ability of the BASMATI platform to recognise the “situation” characterising the actual combined status of users, applications and resources, aimed at achieving an effective and efficient management of applications and resources.

Executive Summary

This deliverable discusses and specifies description, functional, and non-functional requirements for three applications that are the use cases in BASMATI project. The three uses cases are Mobile Virtual Desktop, DAS FEST (Large Events), and TripBuilder applications. The applications used in BASMATI use case are originally created from BASMATI partners therefore the conditions and knowledge for each application can be explained and analyzed without any difficulties.

The benefits of the BASMATI project for each use cases are also explained so that the strong points of BASMATI can be seen clearly and possibly be shared to other applications outside of BASMATI projects. In the end, from this report BASMATI consortium can use this to harmonize and verify the technologies developed in order to realize the BASMATI environment.

Table of Contents

Executive Summary 5
Table of Contents 6
1 Introduction 8
1.1 Relationship to other Deliverables 8
1.2 Outline of Deliverable 9
2 Use Case 1: Mobile Virtual Desktop 10
2.1 General Description 10
2.2 Application Description 12
2.2.1 MVD client 12
2.2.2 Connection manager 12
2.2.3 Resource pool 13
2.2.4 VM infrastructure 13
2.2.5 Virtual desktop delivery 13
2.3 Application Scenarios 13
2.3.1 Automation of Enterprise Virtual Desktop 14
2.3.2 Cross Bordering Business Trip 14
2.3.3 Multiple Cloud Providers 16
2.3.4 Cross Cloud Provider User 16
2.3.5 Heterogeneous MVD users 17
2.4 Benefits by BASMATI Platform 17
3 Use Case 2: Das FEST (Large Event) 18
3.1 General Description 18
3.1.1 Key User and Stakeholder Roles 18
3.2 Application Description 19
3.3 Application Scenarios 20
3.3.1 Have a picnic on the day of the Classic-Breakfast 20
3.3.2 Eat at a food stand 21
3.3.3 Find the nearest toilet. 22
3.3.4 Meet friends on the area of DAS FEST 23
3.3.5 Switch between two concerts 23

3.3.6 Leave the festival grounds as soon as possible 24
3.3.7 A visitor faints and needs first aid 25
3.3.8 Find the quietest place to relax 26
3.4 Benefits by BASMATI Platform 27
4 Use Case 3: TripBuilder 28
4.1 General Description 28
4.2 Application Description 28
4.3 Data characterization 30
4.4 Application Scenarios 30
4.4.1 Visiting Karlsruhe and Das Fest 30
4.4.2 Visiting the city 32
4.4.3 Visiting Das Fest 33
4.4.4 Static Trip Planning 34
4.4.5 Dynamic Trip Planning 35
4.4.6 Detailed information on a PoI 36
4.5 Benefits by BASMATI Platform 37
5 References 38

1 Introduction

Unfortunately, it is quite common for a platform in a development phase to be not resembling the real-world requirements and conditions. The visions of the platform's architects and developers might not be consistent with the tasks and the situation that the platform will face. One of the solutions is to create a platform based on the use cases from the real-world scenarios. Fortunately for BASMATI, there are 3 use cases that can be the basis of the development. Furthermore, the use cases are using the applications that are designed by BASMATI partners. This means that the expertise of each use cases can easily be extracted and BASMATI can easily adapt and accommodate its design and architecture to satisfy each use case's requirements.

The first use case is originated from ETRI which is an application called as Mobile Virtual Desktop (Kim et al., 2016). MVD is an application to enable virtual desktop experience to users remotely. It provides many advantages, such as lower maintenance and operation costs and higher utilization.

Another use case is called as Das Fest, an application to handle a large event, which is co-designed by CAS (CAS, 2016). DASFEST is a music festival in Germany that attracts about 300.000 visitors over a course of 3 days. CAS created an application called the same as the festival name to complement the experience in the real world.

The last but not the least use case is an application called as TripBuilder which is co-authored by CNR (Brilhante et al., 2014). It's a user-friendly and interactive platform for scheduling a time-budgeted sightseeing tour of a city on the basis of the points of interest and the patterns of movements of tourists gathered from user-contributed data.

There are many approaches on technologies that can be used in developing BASMATI. This report is to give a clear guidance on what approach which is suitable for BASMATI by looking at the functional and non-functional requirements given by each use case.

1.1 Relationship to other Deliverables

This report is the overview of the use cases and the scenarios for each use cases which are used in BASMATI. In particular, it will be a guidance on what technologies that can be utilized to satisfy the requirements for each use cases. Therefore, the deliverables which are responsible to create a software demo or prototype should refer to this document. In addition, Work Package 6 (User Cases Definition, Experimentation and Validation), also related to this report. In that package, the more detail and more technical discussions are going to be stated so the scenarios can be conducted precisely. Figure 1 shows the relationship of this report to other deliverables.

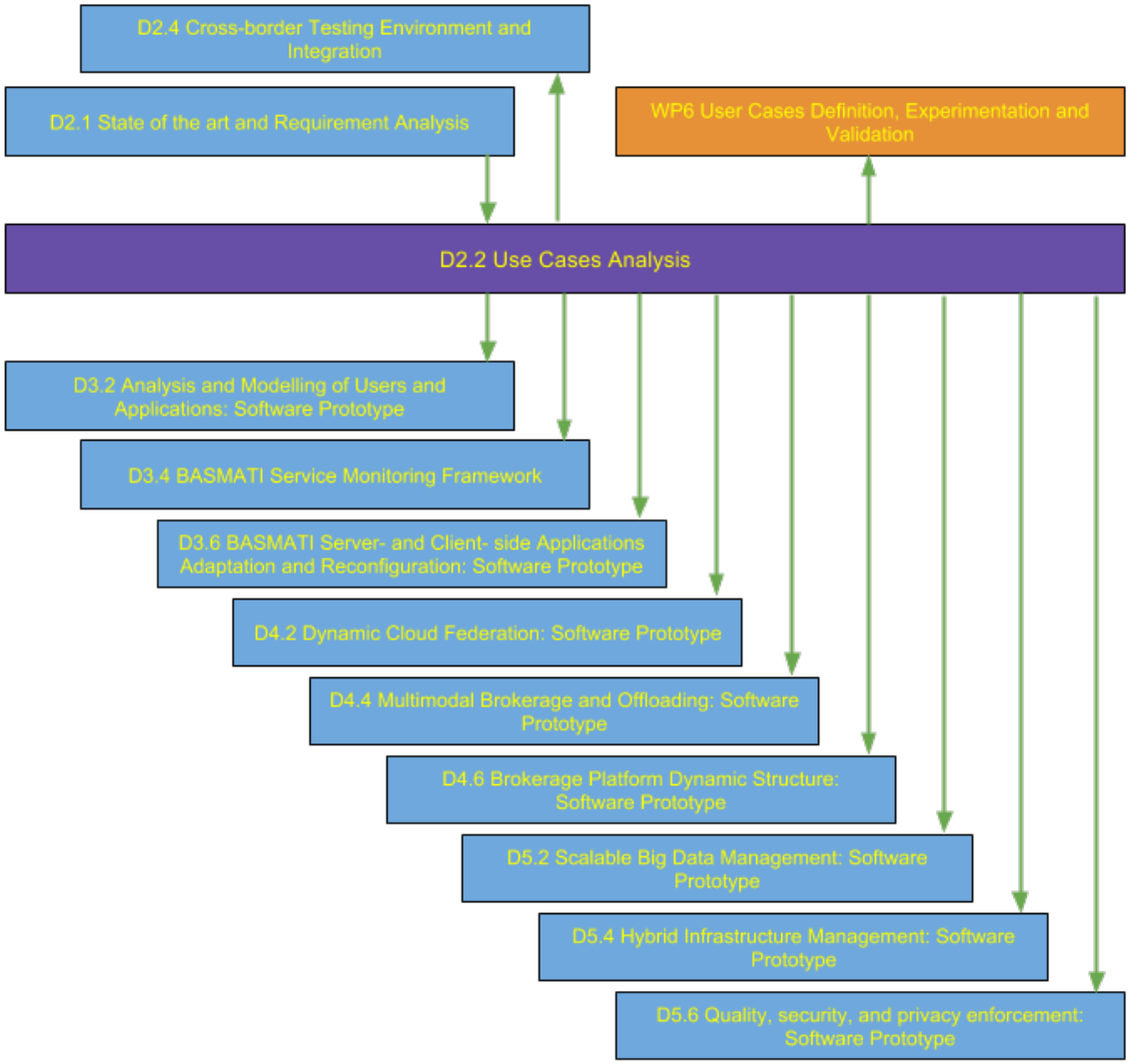


Figure 1. Relationship of D2.2 to other deliverables

During the course of the project, D2.2 report is also influenced by D2.1 State of the Art and Requirement analysis because it is important for the use case to always refer to the most updated state of the art technologies in the academics.

1.2 Outline of Deliverable

Section 2 in this report discusses about Mobile Virtual Desktop. In section 3, Das FEST is the main topic. Meanwhile in section 4, the use case TripBuilder from CNR is examined.

By reading each section of the use case, the readers should understand the main objectives, requirements, and designs of the application. The multiple scenarios to test the validity and effectiveness of BASMATI in running the applications are also described. Furthermore, the benefits of BASMATI are explained clearly.

2 Use Case 1: Mobile Virtual Desktop

2.1 General Description

Mobile Virtual Desktop (MVD) is defined as a cloud service category in which the capabilities provided to the Cloud Service Customer (CSC) are the ability to build, configure, manage, store, execute and deliver users' mobile desktop functions remotely. With MVD, the user experience is achieved through a UI, which is presented on a MVD client over the network. Figure 2 shows the conceptual view of MVD.

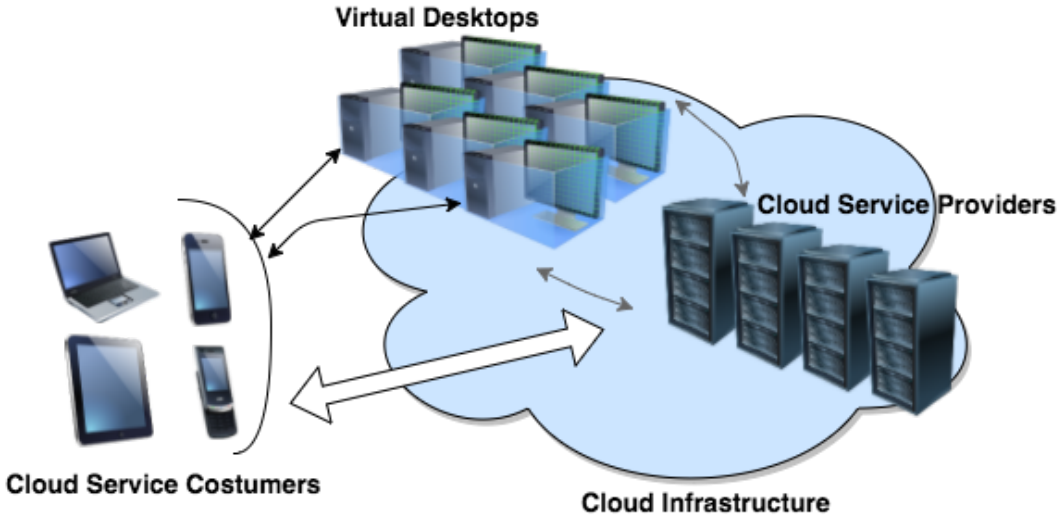


Figure 2. Conceptual view of Mobile Virtual Desktop

Instead of maintaining and running a mobile operating system and applications on CSC's devices, servers of a CSP located in the cloud are used to execute the instances of users' virtual desktops. This allows a party (e.g. an organization) to run end user's operating systems and applications, and keep their data in cloud computing environment.

Based on application streaming and virtualization technologies, CSCs can access the virtual desktop environment through cloud infrastructure. A few technologies can be used for providing services of MVD with various delivery protocols such as virtual desktop delivery protocol and web-based delivery protocol in Figure 3.

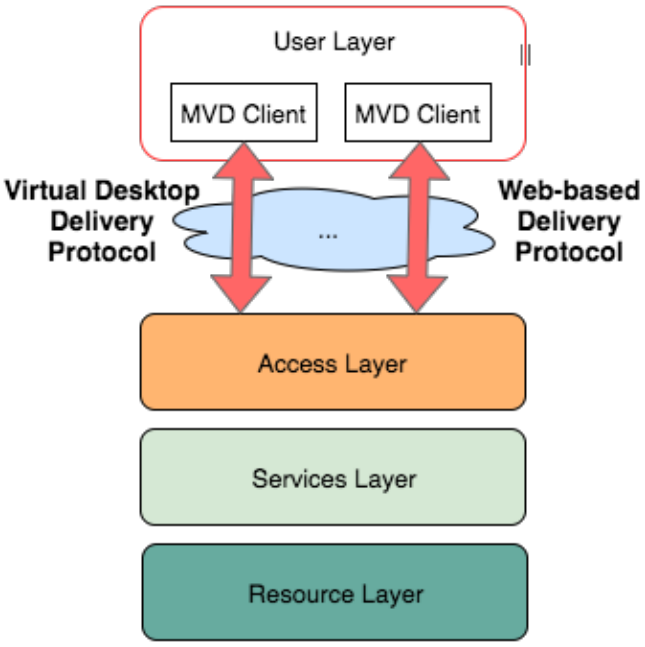


Figure 3. MVD delivery solutions

MVD supports the users' virtual desktop and recreates it in an environment hosted on a remote system. A virtual desktop is executed for each user from the server side and users then access this environment remotely from MVD clients, with all the processing associated with virtual desktop. MVD also uses a virtual desktop delivery protocol to deliver the virtual desktop.

In the web-based MVD solution, a web-based server invokes application services from different servers and aggregates them to build a virtual desktop service. The web-based MVD solution relies on cloud services provided through the use of web oriented technologies, i.e. based on HTTP, HTML and the new features supported by HTML5.

The main advantages of MVD are:

- **Enhanced management and security**: Since all desktop applications actually run in a server, they are more secure than if they were installed on each user's PC because the CSP can focus more on security aspects.

- **Lower Total Cost of Ownership (TCO)**: By placing emphasis on the data centre rather than individual user devices, MVD promotes longer hardware life. Organizations or enterprises seeking to avoid additional costs can switch part of their ICT infrastructure from capital expenditure (CAPEX) to operating expenditure (OPEX), as they now pay for virtual desktops. Also, by decoupling the desktop operating system from the hardware, smaller and cheaper user's devices can be employed, leading to substantial savings.

- **Preservation of the rich user experience**: MVD can provide an uncompromised user experience. This is due to the fact that it leverages a hypervisor which enables the hosting of individual authentic user operating systems. Conversely, shared service

environments offer a user experience that may compromise between application compatibility and user personalization.

- **Separation of CSP and CSC roles**: MVD allows separation between the roles of the CSP and the CSC. The CSP is responsible for everything up to the virtual desktops (i.e. servers, storage, virtualization software, etc.), and the CSC is responsible for everything inside the virtual desktops (e.g. OS, application packaging, user profiles, etc.)

2.2 Application Description

Figure 4 shows the architecture for MVD logical components. The environment is based on a traditional client-server model and mainly consists of a MVD client, a connection manager, a resource pool, a VM infrastructure, and a virtual desktop delivery.

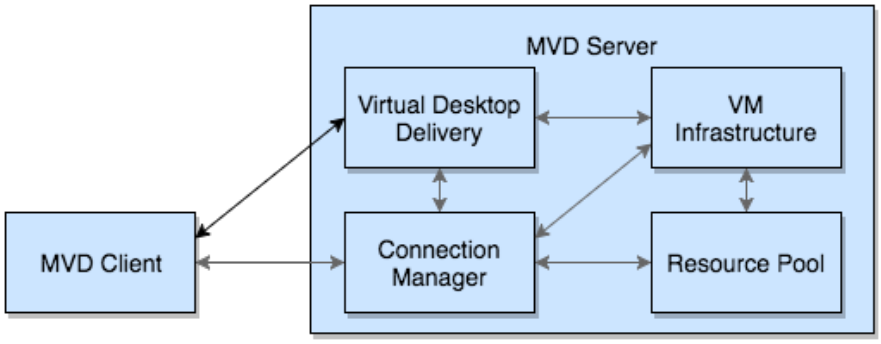


Figure 4. Architecture for Mobile Virtual Desktop

Main logical MVD components shown in Figure 4 are as follows:

2.2.1 MVD client

MVD users can be provided with their virtual desktop remotely through their MVD clients. To access a MVD server, MVD users can employ one of methods among dedicated software, general-purpose web browser, and firmware depending on the type of MVD client.

Depending on the type of MVD client, when the MVD client is booted up, it starts a login procedure with access information to a corresponding virtual desktop such as its identification, password, and IP address. In case of the termination of the corresponding virtual desktop, the MVD client recognizes it and begins on a logoff procedure including turning off the power of VM and the MVD client.

2.2.2 Connection manager

This logical component is responsible for connecting a MVD user to a virtual desktop available and suitable. Connection manager's tasks include (i) end user authentication and license verification to validate the user and user's application, (ii) assignment of a virtual desktop, (iii) coordination of a delivery protocol to be used between a MVD client and a MVD server, and (iv)the necessary storage allocation. In addition, the connection manager is responsible for load

balancing and managing the number of users per server, reconnecting a user to the virtual desktop.

The connection manager uses the resource pool and the VM infrastructure to allocate the required resources such as computing, network, and storage in VM infrastructure.

2.2.3 Resource pool

Resource pool is an abstraction of software resources such as OS, applications, and user profiles. The software resources can be transferred to a certain VM in streaming form timely, and run on the VM. A resource pool can offer provisioning information regarding the software resources on request by a connection manager.

2.2.4 VM infrastructure

VM infrastructure is an abstraction of VM resources. The main role of the VM infrastructure is to support hardware and software resources and create VMs. In a virtual desktop environment, a virtualization function, called a hypervisor, is highly desirable to employ hardware resources efficiently. A hypervisor can abstract physical hardware resources and assign them dynamically to a virtual desktop on hypervisor. Consequently, the end user's application runs on the virtual desktops provided by the VM infrastructure. The VM infrastructure supports high availability features within which many running VMs are created from the same VM template with pre-defined configuration parameters.

2.2.5 Virtual desktop delivery

This component is responsible for encapsulation and delivery of either access to an entire information system environment, or the environment itself to a remote MVD client through the network. A protocol for the virtual desktop delivery provides the communication channels between the MVD client and the MVD server in order to transfer all the interaction information. The interaction information includes display information, control and configuration information, monitoring information etc.

MVD can be either a personal virtual desktop or part of a shared virtual pool. In the case of a personal virtual desktop, there is a one-to-one mapping of virtual desktop to cloud service users with VM. Each user is assigned a virtual desktop that can be personalized and customized. These changes are available to the user each time that user logs on to his or her personal virtual desktop. For a shared virtual pool, a single image is replicated across many VMs and users can reuse a single VM over time. As users connect to the shared virtual pool, they are dynamically assigned a virtual desktop. Shared virtual pool allows for uniformed experience across all end users, while combined with simplified administration means.

2.3 Application Scenarios

2.3.1 Automation of Enterprise Virtual Desktop

Title	Automation of enterprise virtual desktop	
Use Case ID	MVD-1	
Actor	Enterprise workers	
Use Case Description	MVD users (enterprise workers) access the already installed enterprise applications and data hosted in virtual desktops which are created within a MVD server. Common applications of this type include online word processing, email, communication, co-operating development, and so on.	
Pre-Conditions	A template of virtual desktop with installed enterprise applications is exist.	
Success and Failure	All users are able to work directly without have to transferring initial data and installing enterprise applications in their own desktop.	
Frequency	Very often	
	Crucial	Description
Statistical Data	Yes	Most popular enterprise applications
Issues	A powerful and fast internal network in the company is needed in order for seamless experience of desktop usage. The number of workers to use certain software maybe is limited due to its license.	
Expectations	There are many type of golden images available. Each enterprise workers can select the image that is most suitable for their working environment.	

2.3.2 Cross Bordering Business Trip

Title	Cross Bordering Business Trip
Use Case ID	MVD-2
Actor	A Korean employee
Use Case Description	A Korean employee has a business trip to a European country, but still prefers to use their national VDI (Virtual Desktop Infrastructure) service to continuously perform business work, to access to specific data, or to enjoy favourite personal applications.

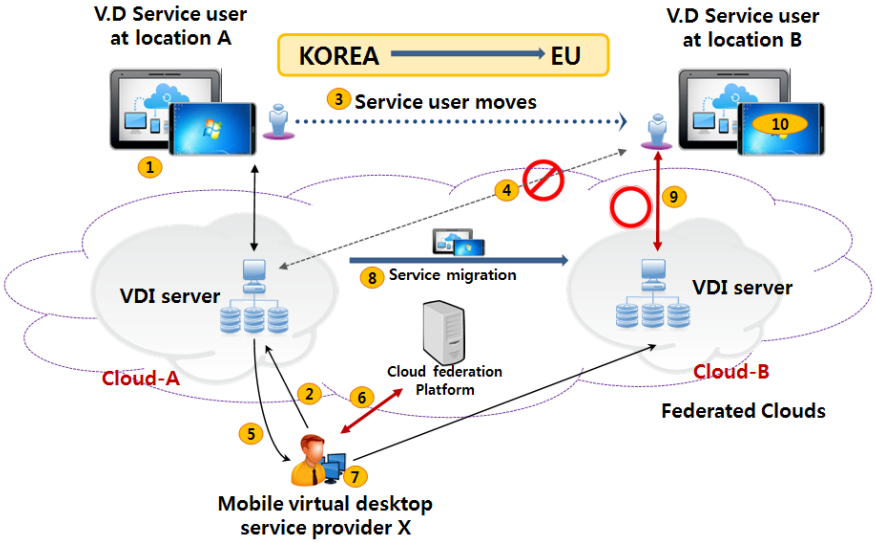
<table>
<tr><td></td><td colspan="2"> 

Figure 5. The flow of Cross Bordering Business Trip Scenario.</td></tr>
<tr><td>Pre-Conditions</td><td colspan="2">Virtual desktop has been activated with several applications running.</td></tr>
<tr><td>Success and Failure</td><td colspan="2">User receiving a stable and effective virtual desktop service seamlessly during the movement.</td></tr>
<tr><td>Frequency</td><td colspan="2">Rare</td></tr>
<tr><td></td><td>Crucial</td><td>Description</td></tr>
<tr><td>Personal Data</td><td>yes</td><td>Expected time and location of user’s destination.</td></tr>
<tr><td>Real-time Trajectory Data</td><td>no</td><td>Current user’s location</td></tr>
<tr><td>Issues</td><td colspan="2">The size of snapshot of a user’s virtual desktop is growing as he/she use it. The big size of the snapshot can slow down the user’s desktop transfer to another region.</td></tr>
<tr><td>Expectations</td><td colspan="2">The transfering of user data is finished before the user is reaching his/her final destination.

Golden image of the user is already existed in the destination server. Therefore BASMATI can just transfer the snapshot.

The transferring is happened automatically without manual authorization from MVD admins or MVD users.</td></tr>
</table>

2.3.3 Multiple Cloud Providers

Title	Multiple Cloud Providers
Use Case ID	MVD-3
Actor	MVD Users
Use Case Description	In this scenario, the MVD users are running on multiple Cloud Providers. Half of MVD users' virtual desktops are running on AWS (Amazon Web Service) and another half are running on Microsoft Azure.
Success and Failure	All users are running virtual desktop without any difference between different cloud provider.
Frequency	Rare
Issues	Different underlying technologies in different cloud providers that are not exposed.
Expectations	MVD admins can see the number of users serviced by each Cloud Service Provider

2.3.4 Cross Cloud Provider User

Title	Cross Cloud Provider User
Use Case ID	MVD-4
Actor	A MVD User
Special Stakeholders	-
Use Case Description	In this scenario, a virtual desktop is transferred from a cloud provider to another cloud provider due to user's request or financial reason.
Pre-Conditions	A virtual desktop is exists in a cloud provider.
Success and Failure	The transfer between cloud provider is done successfully with no differences felt by user.
Frequency	Rare

Issues	Different underlying technologies in different cloud providers may pose incompatibility
Expectations	MVD admins can authorize the user transferring to another cloud provider.

2.3.5 Heterogeneous MVD users

Title	Heterogeneous MVD users	
Use Case ID	MVD-5	
Actor	MVD users	
Use Case Description	In this scenario, MVD users are spread all over the world. Each of them is given service by the nearest server available.	
Pre-Conditions	MVD is already installed and activated.	
Success and Failure	All users are running virtual desktop without any difference between different location.	
Frequency	Often	
	Crucial	Description
Statistical Data	Yes	User's location and server's location
Issues	Different server location may have different specification and price even though from the same cloud provider.	
Expectations	MVD admins can see the number of users serviced by each regions.	

2.4 Benefits by BASMATI Platform

MVD is an application that's very sensitive to response time. Slow response time is resulting in jerky movements of the application and low satisfaction of the users. In order to maintain an acceptable QoE, response time lower than 150 ms is needed. One of the methods to reduce response time is to make MVD servers as near as possible to MVD clients. Since not a single cloud service provider can be available in all areas, the availability of federation cloud in BASMATI is a great advantage in providing a fast response time. Because a federation cloud consisted of multiple cloud service providers can cover more areas. As a result, the traveling MVD users can access their desktops with fast response time anywhere they go.

Backend offloading of BASMATI is aiding MVD servers to balance the work load to different region and different cloud service provider. However, since the data that should be transferred is enormous, it is advisable to schedule the transfer. For example, if a user is traveling from South Korea to Italy using an airplane. During his/her commuting in airplane, MVD servers are transferring his/her data from a server in South Korea to a server in Europe. Unfortunately, the other type of offloading, frontend offloading is not feasible in MVD application without changes made in the source code.

Another advantage is the ability of BASMATI to recognize user's patterns. BASMATI knows if a certain MVD client is using a resource extensively and BASMATI has the ability to optimize the machine of that respective client accordingly.

3 Use Case 2: Das FEST (Large Event)

3.1 General Description

We are focusing on investigating new applications for major events. All case studies refer to the large event DAS FEST in Karlsruhe (Germany). DAS FEST is an open-air event held since 1985, which annually attracts 200 to 400 thousand visitors. The music festival is the main part, but unlike other festivals, an extensive sport, children's and family program is also provided. In addition to sponsoring, DAS FEST is mainly financed by the sale of beverages. Due to its increasing popularity, the entrance to the site had to be stopped for the first time in the season 2006.

The implementation of DAS FEST requires the provision of an extensive infrastructure. The continuously high number of visitors places demands on security, which has an effect, for example, on the division of the terrain. The festival site itself is a closed area. Access to the festival site is through several entrances around the festival grounds. This makes it possible to accurately determine the number of users. The festival is easy to reach by public transport. Also the journey by bike or on foot is possible. Parking spaces for cars and buses are available in limited numbers.

Participation in the event naturally requires physical presence on the ground. Thus the use cases cover mainly visitors who participate in the live concerts and other events of the DAS FEST. As a future vision, one could imagine extending the participation to peoples who receive live streams. In this way one would provide a kind of virtual festival, which ideally complements the festival on site with a further dimension, linking both, the real and the virtual world.

3.1.1 Key User and Stakeholder Roles

Physical Visitor: The main actor is a user who physically participates in the event. He or she uses the application during different phases of a visit, for planning, travelling and guidance on the festival grounds, whereby each of these phases introduces different functional needs and environmental conditions.

Virtual Visitor: The virtual visitor is a person who does not physically take part in the event but shall be enabled to interact with it by the cloud environment and its services.

By using the application some of the physical visitors are turned into hybrid visitors as many features introduce a bidirectional communication and interaction with cloud features that are also open to virtual visitors. Thereby the cloud enables an exchange and interaction between the virtual and the physical community.

Artists: The members of a band; the accompanying staff of a band, such as sound engineers and the management.

The Das FEST staff: The staff covers many different functional aspects like planning, setup, marketing, operational and technical management, ticket collection, and security issues during the festival. It bundles the connection to external parties and subcontractors like the festival security staff, emergency doctors, fire brigades, and police services.

Subcontractors: Operators of food stalls, suppliers for food and beverages.

Medical and emergency services: Emergency doctors, paramedics, fire brigade.

Public relations: Agents, promoters, journalists, politicians, VIPs.

3.2 Application Description

YellowMap has written in cooperation with the organizers of Das FEST the official app. It is available for Android and iOS and displays information about the bands and artists, the calendar of events and other things, such as emergency information. The app also includes an offline-capable terrain plan (map), which serves for orientation on the 180 thousand square meter terrain. It shows, among other things, the positions of the five stages, the event locations, the stands (food and drink), toilets and emergency doctors.

In the future, it is planned to expand the app further and make it more interactive and flexible in order to meet the needs of the users. This is where BASMATI comes into play. In particular, the position data of the festival visitors are to be used for different scenarios, which makes the stay more pleasant and safe. It is planned to supplement the BASMATI multi-cloud with a new cloud-based service, which can manage and evaluate tracking data of visitors.

Picture 6. The appearance of Das Fest application.

The app should provide users with suggestions on how to reach their goals faster or better in different situations. Users should also be helped to bridge waiting times by providing information about the waiting period and the reason for it.

3.3 Application Scenarios

The scenarios are best described in the case studies, for which the most common examples are given. Not all use cases will be implemented at the same time in the future. The use cases, which would be interesting for the organization staff as well as for the rescue workers, are left out. We focus on the physical visitors.

3.3.1 Have a picnic on the day of the Classic-Breakfast

Title	Have a picnic on the day of the Classic-Breakfast	
Actor	Adult female	
Special Stakeholders	Spouse, children	
Use Case Description	A family with two children would like to have a picnic on the day of the Classic-Breakfast. The visit to the music event is of minor importance.	
Pre-Conditions	DAS FAST app has been activated at home. It has been decided to go by car.	
Success and Failure	Family has found a free place near the stage.	
Frequency	On the day of the Classic-Breakfast this scenario is fairly common.	
	Crucial	Description

Personal Data	no	• Number of persons involved • Travel information
Environmental Data	yes	• Weather information
Real-time Trajectory Data	yes	• Occupancy of the meadows near the concert stage • Queue at the entrance
Statistical Data	no	• How many hours before the event is the place occupied?
Issues		
Expectations	System suggests in time when to leave and notifies the app/user. System points out when it is too late to leave or when no free places are available anymore. System takes queue at the entrance and the corresponding time delay into account. Time calculation should include the search for a parking space and the access time.	

3.3.2 Eat at a food stand

Title	Eat at a food stand	
Actor	Any visitor	
Special Stakeholders	Group of visitors	
Use Case Description	A person wants to eat something before the concert.	
Pre-Conditions	Begin of the concert has been specified. It was specified where you want to eat.	
Success and Failure	The person has taken a good seat in front of the stage.	
Frequency	Very often (standard requirement)	
	Crucial	Description
Personal Data	no	Needs to choose a food stand (select a point of interest) and a concert/event.
Environmental Data	no	

Real-time Trajectory Data	yes	Length of queue and calculated waiting time.
Statistical Data	no	How many hours before the event are all the good seats in front of the stage occupied?
Issues		
Expectations	The system indicates when to leave to arrive on time. The following points are considered: time needed to get food, time needed to eat, time needed to walk.	

3.3.3 Find the nearest toilet.

Title	Find the nearest toilet.	
Actor	Any visitor	
Special Stakeholders	Disabled person	
Use Case Description	A person needs to go to the toilet.	
Pre-Conditions	A person is looking for the nearest toilet.	
Success and Failure	Location has been found.	
Frequency	Very often (standard requirement)	
	Crucial	Description
Personal Data	no	Yes, if handicapped person
Environmental Data	yes	Position of toilets
Real-time Trajectory Data	yes	GPS position every 30 seconds Total number of visitors on the area Occupancy of potential locations
Statistical Data	no	
Issues		
Expectations	The system shows the user where the nearest toilet is located.	

3.3.4Meet friends on the area of DAS FEST

Title	Meet friends on the area of DAS FEST	
Actor	Any visitor	
Special Stakeholders	Other visitors	
Use Case Description	A person wants to meet with a couple of friends on the area of DAS FEST.	
Pre-Conditions	Actors are aware that a meeting spot cannot be identified in advance, because of the amount of people and their movements. All persons have smartphones with Wi-Fi/WLAN and GPS capability.	
Success and Failure	Friends have met.	
Frequency	Very often (standard requirement)	
	Crucial	Description
Personal Data	no	Yes, if handicapped person
Environmental Data	yes	Wi-Fi/WLAN access points.
Real-time Trajectory Data	yes	GPS position every 30 seconds Total number of visitors on the area Occupancy of potential meeting spots
Statistical Data	yes	Particularly important are characteristics of the terrain to avoid crowds of people and confusing places.
Issues	GPS signals are weak or the geographical position is not accurate.	
Expectations	The system suggests a meeting spot between or near the location of the visitors. The system indicates the distance and the route time to the meeting place. The system calculates the optimal meeting time.	

3.3.5Switch between two concerts

Title	Switch between two concerts

Actor	Any visitor	
Special Stakeholders	Other visitors, group of visitors	
Use Case Description	A person wants to leave a concert during the performance in order to visit another concert or event.	
Pre-Conditions	Several concerts and events take place at the same time.	
Success and Failure	Switch of concert was successful. No change was carried out which has led to a bad seat.	
Frequency	Uncertain	
	Crucial	Description
Personal Data	yes	Person must express his wish to switch between two events.
Environmental Data	no	
Real-time Trajectory Data	yes	Density of the crowd in front of and near the concert stage. Any narrow places on the way to the next stage.
Statistical Data	no	
Issues	The speed at which you move through the crowd is not really predictable.	
Expectations	A good and accurate estimate of the travel time. It should be advised not to make a change if the new stage is too crowded.	

3.3.6Leave the festival grounds as soon as possible

Title	Leave the festival grounds as soon as possible
Actor	Any visitor
Special Stakeholders	Group of visitors
Use Case Description	A visitor wants to leave the festival grounds quickly.
Pre-Conditions	
Success and Failure	A visitor has left the festival area without getting stuck in a crowd or

	queue of peoples.	
Frequency	Very often (standard requirement)	
	Crucial	Description
Personal Data	no	
Environmental Data	yes	Traffic reports
Real-time Trajectory Data	yes	General movements of the crowds Queues at the exit points of the site. GPS data of visitor
Statistical Data	no	
Issues		
Expectations	The system tells the visitor how long it takes to leave the festival grounds. In addition, the system tells the visitor when he will arrive at home. Because the shortest path in such a situation is not always the fastest, the distance will always be specified. The actor can choose between different solutions.	

3.3.7A visitor faints and needs first aid

Title	A visitor faints and needs first aid
Actor	Another visitor (who initiated the emergency call) and the emergency doctor
Special Stakeholders	Fainted person, security staff, other visitors, bystanders
Use Case Description	A visitor faints and needs first aid. The emergency doctor is called by another visitor.
Pre-Conditions	The emergency doctor is on the festival grounds. Initially a helicopter is not available.
Success and Failure	The ambulance is arrived on time.

Frequency	Rare	
	Crucial	Description
Personal Data	no	Static health data (allergies, blood group, handicaps etc.) Medical or emergency knowledge Sensory health data
Environmental Data	no	
Real-time Trajectory Data	yes	GPS position of fainted person Distribution of crowds
Statistical Data	no	
Issues	If a confusing situation, people would panic.	
Expectations	First, the system provides a simple button that allows to notify the emergency doctor immediately. Second, the nature of the emergency can be specified (the app guides the user). Tips will be offered to give first aid on the spot. The app asks if physicians are present. The app recommends to form a passage.	

3.3.8 Find the quietest place to relax

Title	Find the quietest place to relax	
Actor	Visitor	
Special Stakeholders	All other visitors	
Use Case Description	A visitor or a group of visitors want to avoid the turmoil.	
Pre-Conditions		
Success and Failure	A peaceful place has been found.	
Frequency	Rare	
	Crucial	Description

Personal Data	no	
Environmental Data	no	Characteristics of the terrain
Real-time Trajectory Data	yes	Geographical positions of all people
Statistical Data	no	
Issues		
Expectations	The system proposes several quiet places. The system calculates the most comfortable and fastest route to the destination.	

3.4 Benefits by BASMATI Platform

The local cloud storage is part of the BASMATI multi-cloud. The local storage will provide a security system that can be queried and accessed by BASMATI.

The user interface will run in the user's execution area. Apart from this, there will be a front-end application manager and a front-end service, which will be used to load the BASMATI services. Resource issues do not apply to memory or CPU of the clients, and therefore not to the algorithms – the computing power of modern mobile devices would be sufficient to calculate them.

The "Application & User Data Collector" provided in BASMATI is only informed about the tracking activity; the user's position data will not be passed on. This prohibits both the operator's policy and the current privacy guidelines.

The backend system can benefit from the BASMATI platform by means of scalability and re-location. The backend consists of different loosely coupled components for gathering user mobility data, processing and mining this data and for searching. Each component can be dynamically load-balanced and scaled by the BASMATI platform during high peaks of activity.

The BASMATI platform enables the application to infer relevant insights where they occur by allocating locally available computing resources. Like in modern data architectures a huge benefit of BASMATI is to bring the computation to the data avoiding unnecessary data movements.

4 Use Case 3: TripBuilder

4.1 General Description

TripBuilder is an unsupervised system helping tourists to build their own personalized sightseeing tour [Brilhante et al., 2015]. Given a target city, the time available for the visit, and the tourist's profile, TripBuilder provides a time-budgeted tour that maximizes tourist's interests and takes into account both the time needed to enjoy the attractions and to move from one Point of Interest (PoI) to the next one.

The knowledge base feeding the sightseeing tour generation algorithm of TripBuilder is entirely mined from publicly available sources, namely, Wikipedia, Flickr and Google Maps.

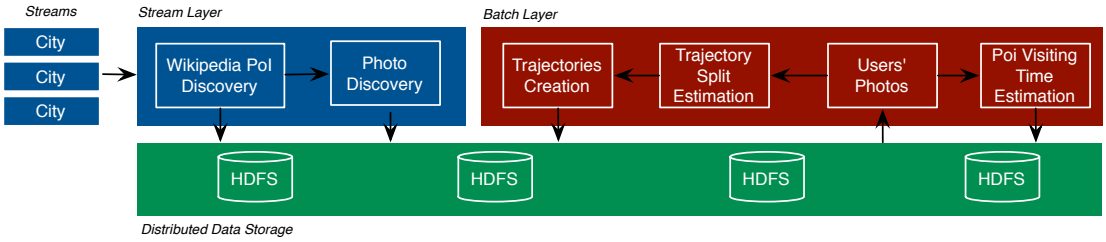
TripBuilder relies on a scalable and robust Cloud architecture (combining both stream and batch processing) to download the data from the heterogeneous sources and build a huge TripBuilder knowledge base covering most popular cities worldwide.

4.2 Application Description

Specific details on the structure of the application, intended as a collection of high-level services. The ideal output would be a high level diagram that represents the main services as first-class entities and the relevant link among them.

The backend architecture of the TripBuilder system involves three different layers:

Stream Layer with Apache Storm. This layer is composed of two different modules that retrieve the relevant information from Flickr and Wikipedia by receiving city bounding boxes as a stream. In particular, each item of the stream is used by Photo Discovery to query Flickr to retrieve the metadata (user id, timestamp, tags, geographic coordinates, etc.) of photo albums, i.e., sequences of photos taken in the given geographic area. An important assumption we are doing is that photo albums implicitly represent sightseeing itineraries within a city. To strengthen the accuracy of our method, this module retrieves only the photos having the highest geo-referencing precision. This process thus collects a large set of geo-tagged photo albums taken by different users in the given geographic area. The second module, Wikipedia PoI Discovery, collects PoIs from Wikipedia. In particular, we assume each geo-referenced Wikipedia named entity, whose geographical coordinates falls into a given area, to be a Point of Interest. For each PoI, we retrieve its descriptive label, its geographic coordinates as reported in the Wikipedia page, and the set of categories the PoI belongs to, which are reported at the bottom of the Wikipedia page. Then, photos from Flickr and PoIs from Wikipedia are matched by spatial proximity according to their coordinates. The Figure 7 below highlights the components on the Stream layer built by means of Apache Storm, a free and open source distributed real-time computation system. The results of the real-time computation are stored on a distributed data storage.



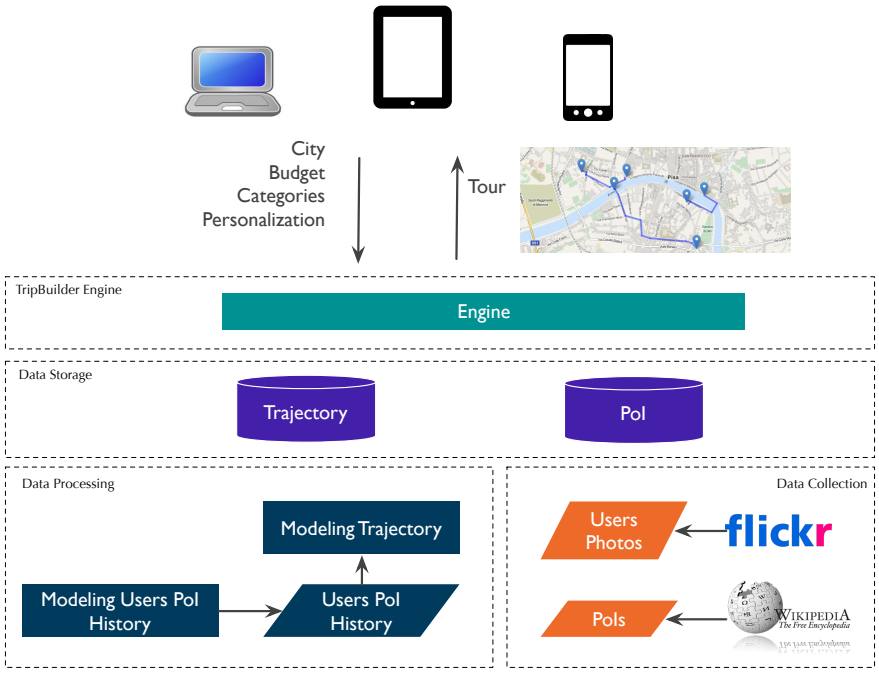
Picture 7. Summary of the Components on the Stream Layer

Batch Layer with Apache Spark. This layer is made up of different components each one manipulating the data previously collected. It is in charge of cleaning and transforming the data by means of distributed computing frameworks like Apache Hadoop and Spark to speed up the data processing step. In particular, the modules here transform sequences of photos from Flickr to sequences of visited Wikipedia PoIs, i.e., trajectories, to be used in the TripBuilder module. Moreover, this step is in charge of computing popularity and other important characteristics of PoIs by considering metadata and information extracted both from Flickr and Wikipedia. We take advantage of the functional capabilities of Spark to distribute and parallelize the computation on the cloud cluster.

Distributed Data Storage. This component is responsible for storing, querying and indexing trajectory and PoI data. It is composed by a database management system and a distributed filesystem that efficiently provides information to the "TripBuilder Engine" component and a distributed data storage to support Stream and Batch layers. The database component contains a well-defined schema to enable flexibility in integrating other data sources. Geo-spatial indexes are used for searching spatial objects, such as PoIs and tourist traces, within a given region (e.g. polygon). The system also takes advantage of indexes over PoI categories and tourist traces, both represented as arrays, to efficiently retrieve relevant PoIs to the user preferences. Moreover, the distributed filesystem is built by using the Apache Hadoop Distributed Filesystem (HDFS).

The first three layers implements the data collection, processing and storage functionalities. Once the knowledge base is built, a second module is then used to build the sightseeing tour.

TripBuilder Engine. It is the core of the architecture. It starts from a set of trajectories crossing a set of PoIs, a time budget, user preferences and a factor used to tune the level of personalization as input, and generates the personalized sightseeing tour.



Picture 8. Architecture of TripBuilder from Functional Point of View

Overall, the architecture of TripBuilder from a functional point of view is depicted in the figure 8 above. We report the four main functionalities of the system, i.e. Data Collection, Data Processing, Data Storage and TripBuilder Engine and the interactions between them and with the final user.

4.3 Data characterization

	Das Fest	Karlsruhe
Static data	Data from previous versions of Das Fest	POIs from "Internet"
Dynamic data	POIs of events Queues at stands	Queues at museums and other interest sites

4.4 Application Scenarios

4.4.1 Visiting Karlsruhe and Das Fest

Title	Visiting Karlsruhe and Das Fest	
Use Case ID	TB-1	
Actor	Single occasional traveller	
Use Case Description	A single traveller wants to maximize her visit of a single day, having the best of both the city and the music events in Das Fest.	
Pre-Conditions	Users have smartphones with Wi-Fi/WLAN and GPS capability. Tripbuilder application must be installed on the phone.	
Success and Failure	User is satisfied with the proposal, and possible follow up changes. A failure consists in a plan that does not meet user preferences or time budget.	
Frequency	A plan is provided to the user then the plan could be updated for taking into account: i) a possible change in the plan or ii) new PoIs becomes attractive within Das Fest so that they are included in the plan if they can be of interest for the user.	
	Type	Place
Data used	Static and Dynamic	Das Fest
	Static	Karlsruhe
Issues	GPS signals are weak or the geographical position is not accurate. The mobile phone lost data connection and no updates are thus performed on the plan for the time it is offline.	
Expectations	TripBuilder suggests a sequence of PoIs to be visited in Karlsruhe and Das Fest. The plan contains the average visiting and connecting time between PoIs. The suggestion takes into account the preferences and the available time budget for the user.	

4.4.2 Visiting the city

Title	Visiting Karlsruhe and Das Fest	
Use Case ID	TB-2	
Actor	A family with two childrens	
Use Case Description	A family wants to visit the historical city of Karlsruhe and some area of Das Fest friendly to children. They provide the application the total budget, in terms of money, they intend to spend for the visit.	
Pre-Conditions	Users have smartphones with Wi-Fi/WLAN and GPS capability. Tripbuilder application must be installed on the phone.	
Success and Failure	Family is satisfied with the proposal, and possible follow up changes. A failure consists in a plan that does not meet user preferences or time budget.	
Frequency	A plan is provided to the user then the plan could be updated for taking into account: i) a possible change in the plan or ii) new PoIs becomes attractive within Das Fest so that they are included in the plan if they can be of interest for the user.	
	Type	Place
Data used	Static	Das Fest
	Static	Karlsruhe
Issues	GPS signals are weak or the geographical position is not accurate. The mobile phone lost data connection and no updates are thus performed on the plan for the time it is offline.	

Expectations	TripBuilder suggests a sequence of PoIs to be visited in Karlsruhe and Das Fest. The plan contains the average visiting and connecting time between PoIs. The suggestion takes into account the preferences and the available budget (money) for the family.

4.4.3 Visiting Das Fest

Title	Visiting Das Fest	
Use Case ID	TB-3	
Actor	Group of friends	
Use Case Description	A group of friends wants to visit Das Fest. The group has few interest of going to Karlsruhe	
Pre-Conditions	Users have smartphones with Wi-Fi/WLAN and GPS capability. Tripbuilder application must be installed on the phone.	
Success and Failure	Friends are satisfied with the proposal, and possible follow up changes. A failure consists in a plan that does not meet user preferences or time budget.	
Frequency	A plan is provided to the user then the plan could be updated for taking into account: i) a possible change in the plan or ii) new PoIs becomes attractive within Das Fest so that they are included in the plan if they can be of interest for the user.	
	Type	Place
Data used	Static and Dynamic	Das Fest
Issues	GPS signals are weak or the geographical position is not accurate. The mobile phone lost data connection and no updates are thus performed	

	on the plan for the time it is offline.
Expectations	TripBuilder suggests a sequence of PoIs to be visited in Karlsruhe and Das Fest. The plan contains the average visiting and connecting time between PoIs. The suggestion takes into account the preferences and the available budget (time) for the group. The preferences of the group are "averaged" starting from the single preferences of each friend.

4.4.4Static Trip Planning

Title	Static Trip Planning	
Use Case ID	TB-4	
Actor	A User	
Use Case Description	A user is requesting a trip plan without anything changed after initial request.	
Pre-Conditions	The user has the TripBuilder app installed on the phone. The phone has the GPS activated and an internet connection.	
Success and Failure	The user is satisfied with the proposal. A failure consists in a plan that does not meet user preferences or time budget.	
Frequency	Often	
	Type	Place
Data used	Static	PoIs

Issues	
Expectations	The proposal is sent back to user phone that visualizes it on a map that allow an easy navigation for the user.

4.4.5Dynamic Trip Planning

Title	Dynamic Trip Planning	
Use Case ID	TB-5	
Actor	A User	
Use Case Description	A user is requesting a trip plan but something changed after initial request. For example, the queue in one of the destinations is too long and doesn't fit the time budget. TripBuilder should propose a new plan to the user.	
Pre-Conditions	The user has the TripBuilder app installed on the phone. The phone has the GPS activated and an internet connection.	
Success and Failure	The user is satisfied with the new proposal. A failure consists in a new plan that does not meet user preferences or time budget	
Frequency	Often	
	Type	Place
Data used	Static	PoIs
Data used	Dynamic	Queues in PoIs
Issues	Difficulty in tracking the queues in PoIs and user current position.	

Expectations	The trip plan is sent back to the app on the user phone that visualizes it on a map to allow an easy navigation for the user. The app receives also the possible edits of the plan computed by the TripBuilder server and it takes care of visualizing it by notifying the user if she wants to do the edit.

4.4.6 Detailed information on a PoI

Title	Detailed information on a PoI	
Use Case ID	TB-6	
Actor	A User	
Use Case Description	A user is asking a detailed information of one of the PoIs from the plan given by TripBuilder.	
Pre-Conditions	The user has the TripBuilder app installed on the phone. The phone has the GPS activated and an internet connection.	
Success and Failure	It is considered success if the detailed information of the PoIs can be fetched by user. It is a failure if the information is not delivered to user.	
Frequency	Very often	
	Type	Place
Data used	Static	PoIs
Issues	-	
Expectations	The user is able to click on a PoI to show some associated metadata, e.g., description, the categories it belongs to, some pictures, estimated time to visit it, etc.	

4.5 Benefits by BASMATI Platform

TripBuilder will benefit from BASMATI platform in different ways and at the different levels. As aforementioned, TripBuilder can be seen as composed by three different systems. The first two systems are parts of the application backend: one aimed at gathering and processing data, one providing the path suggestions in which the application is structured. The third system is the one on the mobile device presenting the suggestions to the user.

The systems composing the backend can benefit from BASMATI as it will provide ways and manners supporting the automatic reconfiguration, decomposition and re-location of the application. By means of the re-location support, the application will be able to get closer to the origin of the workload, i.e., the end-users. This is a very important feature when the amount of users increases as well as mass of users moves from their home location and get concentrated in a different area (e.g., festivals, Olympic games, etc.). Furthermore, through the adaptivity support BASMATI will allow to achieve an automated scale-up and scale-down of the application accordingly with the actual requirements.

TripBuilder can benefit from its integration with BASMATI also on the mobile application side. In fact, TripBuilder could be able to have a more detailed monitoring of the application behavior and drive, consequently, the suggestions proposed to the end-users.

5 References

Kim, S., Choi, J., Kim, S., & Kim, H. (2016, January). Cloud-based virtual desktop service using lightweight network display protocol. In *Information Networking (ICOIN), 2016 International Conference on* (pp. 244-248). IEEE.

"Das Fest“ als Testlabor für neue Technologien. Retrieved from: http://www.cas.de/nc/aktuelles/news/details/article/das-fest-als-testlabor.html

Brilhante, I., Macedo, J. A., Nardini, F. M., Perego, R., & Renso, C. (2014, April). Tripbuilder: A tool for recommending sightseeing tours. In *European Conference on Information Retrieval* (pp. 771-774). Springer International Publishing.

Brilhante, I. R., de Macêdo, J. A. F., Nardini, F. M., Perego, R., & Renso, C. (2015). Scaling up the Mining of Semantically-enriched Trajectories: TripBuilder at the World Level. In IIR.