OPEN Project

STREP Project FP7-ICT-2007-1 N.216552

Title of Document:	D6.7 Testing & Validation results (final deliverable)
Editor(s):	Agnese Grasselli, Stefano Bolli
Affiliation(s):	VF-IT
Contributor(s):	All Open Partners
Affiliation(s):	All Open Partners
Date of Document:	30 July 2010
OPEN Document:	D6.7
Distribution:	EU
Keyword List:	
Version:	1.0

OPEN Partners:

CNR-ISTI (Italy)
Aalborg University (Denmark)
Arcadia Design (Italy)
NEC (United Kingdom)
SAP AG (Germany)
Vodafone Omnitel NV (Italy)
Clausthal University (Germany)

ABSTRACT

Scope of this document is to describe the second (and final) testing and validation phase that has been carried out at the end of the OPEN Migration Service Platform and prototypes development phase. The evaluation will be described and deeply analyzed in the deliverable introducing both the methodological approach and the obtained results.
Moreover, an analysis of collected data is performed and some future development suggestions are provided for the OPEN Migration Service Platform.

CONTENTS

1. INTRODUCTION

The purpose of the final testing iteration is to evaluate the final version of the OPEN Migration Service Platform (MSP) and to provide suggestions for possible further developments of the migration technology implemented inside the OPEN project.
The evaluation has been carried out taking into account all the available prototypes representing different applications, utilities and services developed on top of the platform. This approach can help in:

- Understanding platform's performances in each scenario. In fact, each different scenario has different characteristics that could lead to different platform's responses in specific tests, e.g. an application using high size state (large amount of data) could experiment a larger migration delay respect to another one using a thin state.

- Providing hints for the correct understanding of the results, contextualizing them depending on the scenario and leading to an extended and more proper evaluation.

The document is organized as follows:

- Chapter 2 is focused on the second iteration usability evaluation. It starts describing the usability principles used to evaluate the platform. For each application class defined in Chapter 2 the results of representative application usability evaluation are reported. The chapter ends with a comparative evaluation among the proposed User Interface (UI) adaptation solutions and other products already commercially available.
- Chapter 3 describes the second iteration programmability evaluation. After the methodology definition all the executed tests are reported analyzing the results both from a module and from an overall platform point of view. The conclusions highlight how the improvements suggested in the first test iterations are now correctly integrated.
- Chapter 4 is focused on the second iteration technological validation. In the first part of the chapter a functional evaluation of the OPEN Migration Service Platform is carried out in order to verify that the OPEN initial requirements are implemented in the final platform version. Then a set of performance indicators are proposed and evaluated to verify how well the platform works in a laboratory scenario. At the end, a module by module analysis is reported focusing on the performance of the single modules.
- The deliverable ends with Chapter 5 where obtained results are discussed. Some ideas for future development are also reported.

2. USABILITY

In this chapter the usability approach to the second testing iteration is described and the obtained results are discussed. In the first section the usability principles that drive the evaluation are described, specifying also the qualitative and quantitative indicators considered. Then in the second section the general methodology applied is explained. The last section reports in a very detailed way the test execution phase and the obtained results for the considered prototypes.

2.1. TESTING STRUCTURE

According to the international standard ISO-9241-11, completely available in [ISO-00], usability is defined as: “The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.” So usability is about:

1. Effectiveness - can users complete tasks, achieve goals with the product, i.e. do what they want to do?
2. Efficiency - how much effort is required from users to do this?
3. Satisfaction - what do users think about the products ease of use?

Note that usability should not be confused with correctness, which is purely concerned with the functionality and features of the product and has no bearing on whether users are able to use them or not.

The methodology proposed in the following paragraphs needs to cover all the three aspects explained before: different kinds of quantitative data have to be collected during the tests execution. In particular for each usability aspect a set of different measures is considered in order to provide useful and comparable results among different prototypes.

In this case the “product” to be evaluated is the OPEN Migration Service Platform. The OPEN MSP is a middleware exporting to on top applications a set of migratory features; consequently a usability evaluation requires the use of some prototypes that can exploit OPEN MSP functionalities.

2.1.1. TASK LIST

The key point for the usability evaluation is the task list. The task list drives the user through the test describing the different actions that need to be performed. For each prototype a predefined task list is created. All the usability evaluations are based on the information collected during the task list execution.

2.1.2. EFFECTIVENESS

Effectiveness evaluation answers the question: can users complete tasks, achieve goals with the product, i.e. do what they want to do?
The effectiveness evaluation is based on:

- Number of tasks failed
- Number of users abandoning the tasks

In particular a task is defined *failed* if during the task execution occurs one of the following case:

- The user is not able to complete the task;
- User executes a different action respect to the one required by the task.

An *application failure* (something goes wrong in the application and there is error) is not considered a failure for the usability evaluation because it does not depend on the user testing the application.
Considering the number of tasks failed in respect to the overall number of executable tasks it is possible to obtain a plot as the one reported in Figure 1. It clearly shows the percentage of successful and unsuccessful tasks and also the number of tasks that, eventually, has been abandoned. In red the tasks that are not correctly executed due to application error are reported.

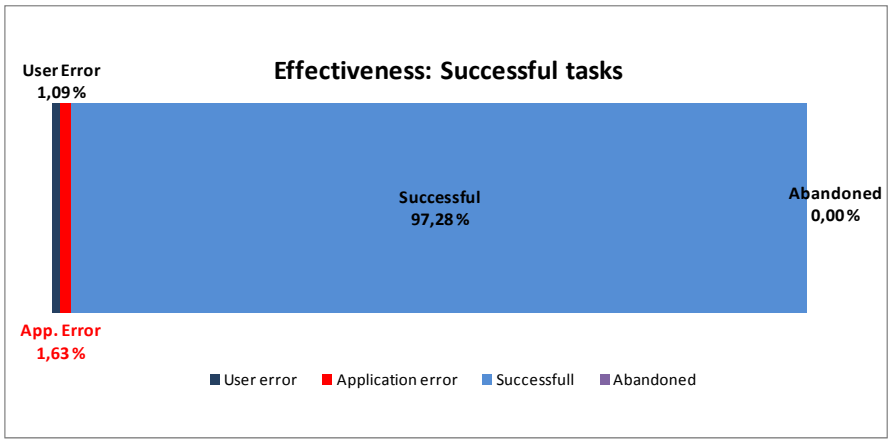


Figure 1 Effectiveness analysis, data representation

2.1.3. EFFICIENCY

Efficiency has to give insights on the effort required by the user to complete the task. The simplest and standard way to collect quantitative data it is to record the time required by the user to complete each task of the task-list. The collected data allows building a time distribution like the one showed in Figure 2 for the whole task-list. Each bar shows the number of participants that completed the task within the related time interval.

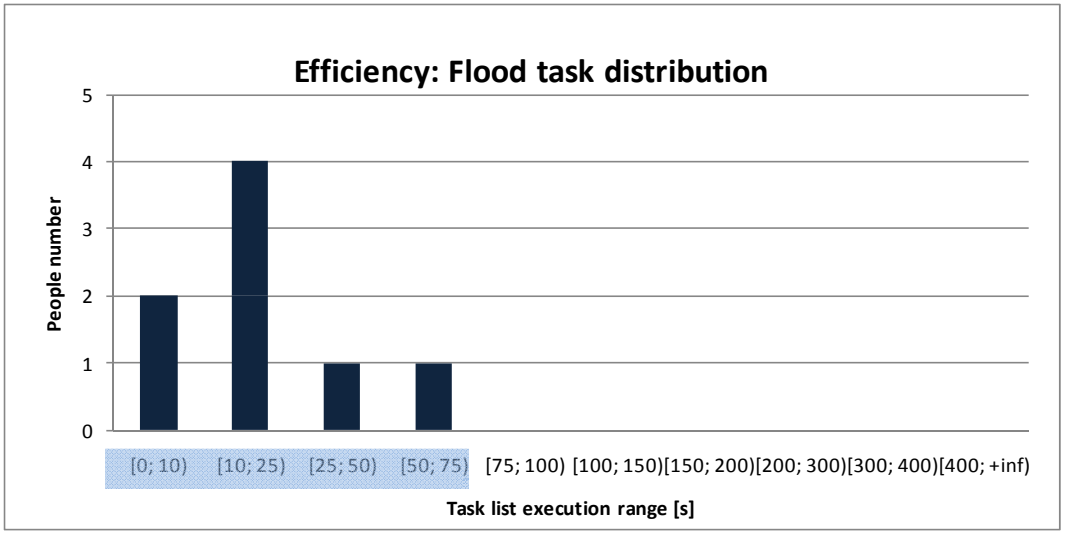


Figure 2 Efficiency analysis, user distribution representation

During the pilot test a reference time to execute each task is estimated, in order to evaluate if the recorded times during the test execution are compatible with the expected duration. In Figure 2 the estimation time for task execution is showed by the light blue bar overlaid on x-axis. Some statistical indicators like mean and standard deviation, lower bound and upper bound can also be easily computed to give a general overview of the system. Another useful representation for tasks time distribution is the one reported in Figure 3 which uses the box plot paradigms [CHA-83], [NIS-03] and [MON-03]. Box plots are an excellent tool for conveying location and variation information in data sets, particularly for detecting and illustrating location and variation changes between different groups of data.

Box plots are formed by vertical axis which contains the response variable measured while horizontal axis contains the factors of interest. For each factor the median and both the lower (25^{th} percentile) and upper quartiles (75^{th} percentile) are computed. In the plot the median is represented with a black line, while the light blue rectangle extends itself from the lower to the upper quartile. This box represents the middle 50% of the data, often call the "body" of the data.

The lines external to the body go from the lower quartile to the minimum point and from the upper quartile to the maximum point.. Typically a symbol is drawn at these minimum and maximum points, although this is optional. Thus the box plot identifies the middle 50% of the data, the median, and the extreme points.

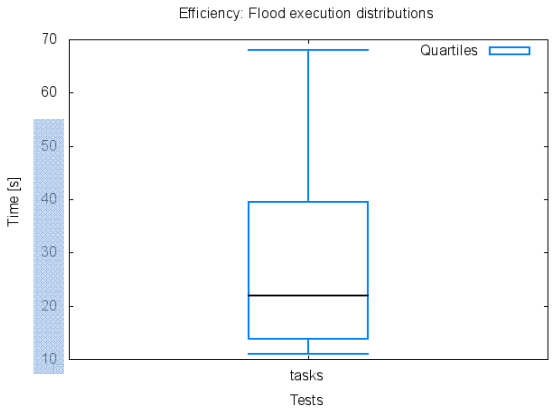


Figure 3 Efficiency analysis, user distribution representation with box plot

2.1.4. SATISFACTION

Satisfaction refers to what users think about the products ease of use. This is a very critical measure because there are a lot of different techniques, usually based on questionnaires, to evaluate it. For example at [QUE-06] an exhaustive lists of possible questionnaires to evaluate the user satisfaction are reported. Here are listed the most used with a brief description:

- SUMI [SUM-94] it is a commercially available questionnaire for the assessment of the usability of software which has been developed, validated, and standardized on an international basis. SUMI is mentioned in the ISO 9241 standard as a recognized method of testing *user satisfaction*. SUMI consists of 50 statements to which the user has to reply that they Agree, Don't Know, or Disagree.
- WAMMI [WAM-08] it is a commercial methodology designed to evaluate the quality of use of web sites. It is purchased on a per report basis.
- SUS [SUS-86] it is a mature questionnaire, very robust and extensively used and adapted. Above all the public domain questionnaires, this is the most strongly recommended.
- QUIS [QUI-90] it is commercially available and it is championed by Ben Shneiderman in his book “Designing the User Interface” [SHN-04].
- USE [USE-01] it is still in development by Arnie Lund however no reliability or validation data are presented.
- CSQU[CSU-95] it is well designed but there is not any standardization base.

Briefly excluding the commercial and the not jet consolidated questionnaires, only SUS, USE, CSUQ are suitable, but the wider used and suggested is SUS. So, in order to apply a standard and easily repeatable approach to evaluate satisfaction the System Usability Scale (SUS) is chosen. Together with SUS, a second methodology is applied: Product Reaction Cards (PRC). It is an innovative evaluation approach that requires the user to select the words inside a big set that

better describe the experience during the test and ask the user to add a synthetic comment. The following paragraphs explain in detail the two considered approaches.

2.1.4.1. SYSTEM USABILITY SCALE APPROACH

The System Usability Scale (SUS) approach is a simple, ten-item scale giving a global view of the experience of the user during the testing execution. The ten questions are standard and are reported in Appendix A. They are in a predefined order that alternates positive and negative question so that it is possible to mitigate the effect of undecided testers. A score is assigned to each answer, and a predefined SUS scoring algorithm is applied in order to obtain the overall value of System Usability (SU) within the range of 0 % to 100 %. In the considered scale 0 % means “hard to use” while 100 % means “easy to use”. Further details about this approach and the scoring algorithm applied are available in [SUS-86].

SU Score range in %	Usability Class (U)	Class Weight (w)
[0;20]	*Hard*	0
(20;40]	*Fairly Hard*	1
(40;60]	*Neutral*	2
(60;80]	*Fairly Easy*	3
(80;100]	*Easy*	4

Table 1 Classifier based on System Usability Score

In order to give readable results about the SUS, the following five classes of the same size have been identified starting from the SUS score.

$$U = \{\text{'hard'}, \text{'fairly hard'}, \text{'neutral'}, \text{'fairly easy'}, \text{'easy'}\}$$

In Table 1 the classification proposal is reported in details. It is so possible to plot the distribution of System Usability results as showed in Figure 4. Note that for each class a specific weight is associated, as reported in the last column of the table. Together with the classification distribution another significant satisfaction indicator is the average SUS score obtained from the filled questionnaires. A score of 100% means that every participant rated the task as easy while a score of 0 % means that every participant rated the task as hard.

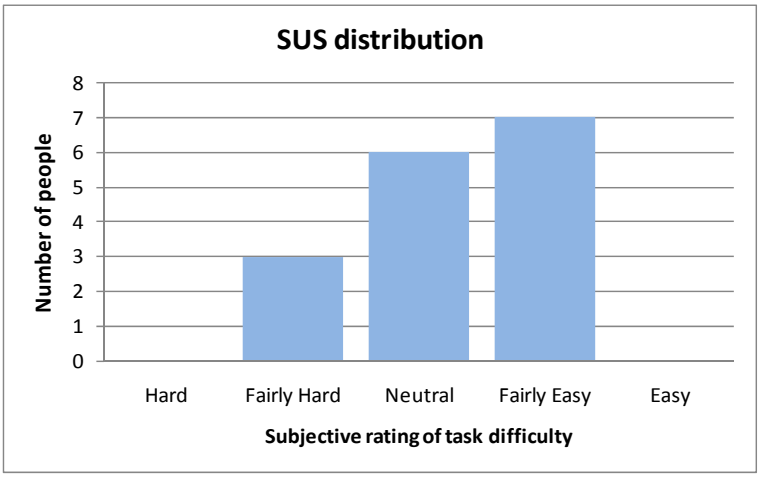


Figure 4 Evaluation of user Satisfaction using SUS approach, distribution.

2.1.4.2. PRODUCT REACTION CARDS APPROACH

The Product Reaction Cards (PRC) approach was developed by Microsoft Corporation (Developed by and © 2002 Microsoft Corporation. All rights reserved.), a detailed description is available in [EDU-02].
It consists of 118 cue cards, each one containing one of the words listed in Appendix B. The word set is developed so that a correct mix of positive and negative concepts is available. At the end of the usability evaluation the complete card set, ordered by ID, is given to the participant. The participant is asked to perform the following actions:

1. The participant is asked to pick the words that best describe the product or how using the product made them feel.
2. Once the user picks the words the moderator returns to the room and records the selected words.
3. Then the users have to narrow their set down to the 5 best words.
4. Finally the users are asked for details about why they picked each of the top 5 words.

Analyzing the user comments and the selected words it is possible to classify the satisfaction inside 2 different classes, positive and negative. A simple, but very readable representation of these results can be showed with a plot like that showed in Figure 5. This test enables the evaluation team to have a better qualitative understanding of the main product characteristics perceived by the user.

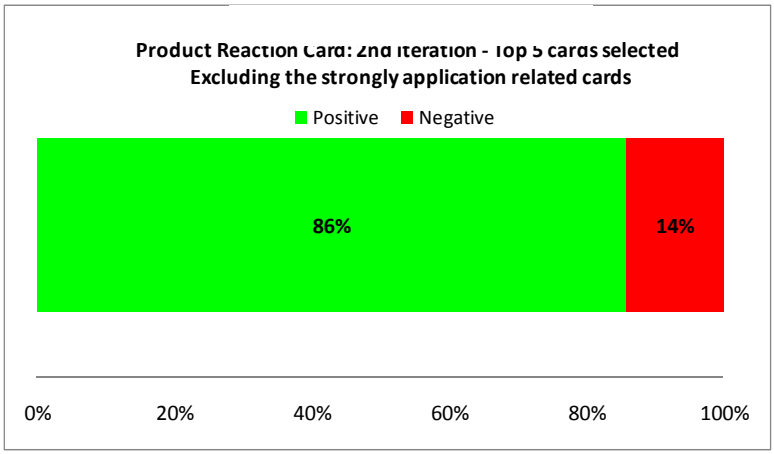


Figure 5 Evaluation of user Satisfaction using Product Reaction Cards approach

2.1.5. COMPETITOR ANALYSIS

It can happen that the OPEN MSP implements features already available also in other products. The competitor analysis consists of executing the same test collecting the same data both for the considered prototypes and for the products that implement similar features. Comparing the obtained results it is possible to evaluate how the feature is implemented in the OPEN MSP in respect to the available product. A possible comparison chart, based on the three ISO usability aspects, is reported in Figure 6. The percentage of successful tasks is reported for the effectiveness, while the time consumed on task is showed for efficiency. In order to compare user satisfaction, specific questionnaires aimed to stress some usability aspects of the considered application are developed and used. Note that in Figure 6 the last column puts clearly in evidence the comparison between the results obtained using the prototype under test in respect to the results collected with the main competitor product.

	MAIN COMPETITOR	THIS PROTOTYPE	STATUS
EFFECTIVENESS	47 %	60 %	◐
EFFICIENCY	260 [s]	114 [s]	○
SATISFACTION	19 %	60 %	○

KEY

● Significantly worse than competitor ◐ No difference ○ Significantly better than competitor

Figure 6 Competitors analysis results

In the OPEN project this kind of analysis is applied focusing on the web page adaptation. In fact there are also other solutions that provide web page adaptation for mobile phones, such as the Opera Mini browser.

2.2. USABILITY METHODOLOGY

Starting from the usability principles described above and reanalyzing the methodology and the results obtained with the first testing iteration the second testing iteration has been carried out. First of all it is important to keep in mind the aspects learned from the previous phase, which are described in D6.4 [D6.4] and whose results are published in D6.5 [D6.5].
The methodology already applied in D6.4 proves its validity, especially for the prototypes that consider migration among different devices. It splits test users into two groups (group A and B) that execute the task list in different order. In this way it is possible to avoid the "learning problem": if a user has already used an application on a device (e.g. laptop) she/he is facilitated to use the same application also on a different device (e.g. mobile phone). Moreover in order to have an expressive data set it is important to involve a sufficient number of people in the testing activities. According to the suggestion available in [HUT-08] it is necessary to consider at least 8 users (4 people in each group).
A different approach from the one applied during the first test iteration is considered with respect to the questionnaire. While in the first test iteration for each prototype there was a different questionnaire in order to better highlight prototype specific possible improvements, for the second iteration a standard questionnaire is used for all prototypes, in order to have a more consistent evaluation.
In order to set up a test plan that can be repeatable and comparable, independently which the prototype under test is, the following tasks need to be executed:

1. Before test execution

    a. The OPEN MSP Platform has to be correctly installed on a server that can be reached by network.

    b. If the application prototype requires some software it has to be correctly installed and configured on the client devices that are involved in the test.

    c. If an on line help is not available in the prototype, the developer of the prototype writes a "how to" that explains to the end user how the application works. It is necessary in order to give autonomy to the user so during the test she/he does not interact with the moderator. Remember that this is a requirement for the usability validation test phase.

    d. The moderator in collaboration with Vodafone Team and the prototype developers prepares the task list for group A and B indicating the expected time to execute each task.

    e. The System Usability Scale (SUS) questionnaire and the Product Reaction Cards (PRC) are printed for the satisfaction evaluation.

f. A pilot test needs to be done before the first test execution. It is necessary in order to check the test environment and also to record the reference time for the task included in the task list.

2. During test execution

   a. The moderator records the number of mistakes made by the user, and eventually if the user abandoned the test.

   b. The moderator records the time that the user spends to execute each task.

3. After test execution

   a. The moderator submits the SUS questionnaire to the user.

   b. The moderator submits the PRC according to the following steps:

      i. The participant is asked to pick the words that best describe the prototype or how using the product made them feel.

      ii. Once the user picks the words the moderator returns to the room and records the selected words.

      iii. Then the users have to narrow their set down to the 5 best words.

      iv. Finally the users are asked for details about why they picked each of the top 5 words and the comments are recorded.

   c. The moderator can eventually add some additional comments about the test execution.

4. Vodafone Team elaborated the data in order to create the usability reports for the analyzed prototype.

The procedure described above is general. At this stage of work, since it is the validation phase, it is correct to have a general methodology with standard measures and questionnaire that can be easily replicated on different prototypes. This means that the obtained results are really comparable starting from the same evaluation concept. Following in next sections a detailed description of the necessary steps to set up the test environment is reported for each considered prototype. Moreover information about the participant characteristic, equipment necessary for the test, task lists, pilot test and test results are described and analyzed in depth.

2.3. EMERGENCY

Considering the real scenario in which this kind of application will be used the Co-Discovery technique is applied. This technique, described in [HUT-08] uses two participants simultaneously during a usability test instead of a single participant. The participants are encouraged to communicate with each other during the test session, and it is this communication that is the key difference during a session. Their dialogue becomes the focal point for understanding how users address problems using the Emergency prototype. It is useful when a product is normally used in real scenario by more than one person who might through perform tasks together, like in the Emergency management case.

2.3.1. PARTICIPANT CHARACTERISTICS

Evaluation tests were performed by Arcadia Design in the first week of May 2010 involving 10 people that never used the Emergency prototype before. All people involved in the tests are familiar with the use of the computer because they are involved in the IT area of Arcadia Design.

2.3.2. TASK LIST

According to the Co-Discovery technique described above and in order to avoid the learning problem (users can give different evaluations if they have already learned how to use the system under test, for further details refer to D6.4 [D6.4]) the test execution has been organized in the following way.

The idea is that a user can give a correct evaluation if he/she tries to use all the features available from the prototype; it means that each user has to use both the Flood Manager and the Traffic Manager. To do this it is necessary to split the test into 2 steps as represented in **Errore. L'origine riferimento non è stata trovata.**. During the first step the generic user A works as the Flood Manager and the user B as Traffic one, while in the second step the roles are swapped. The executed task list is exactly the same, for both the steps, the only thing that changes and it needs to be taken into account is that at the second step the user has already used a similar application, so it is expected that they are little faster in the task list execution.

The task list was proposed by Vodafone team and validated both by SAP and Arcadia Design teams. The underlying idea is to force the user interaction exploiting also the information sharing on the Emergency Wall. The detailed approach is reported in Figure 7 where the steps methodology is clearly showed and is highlighted in yellow.

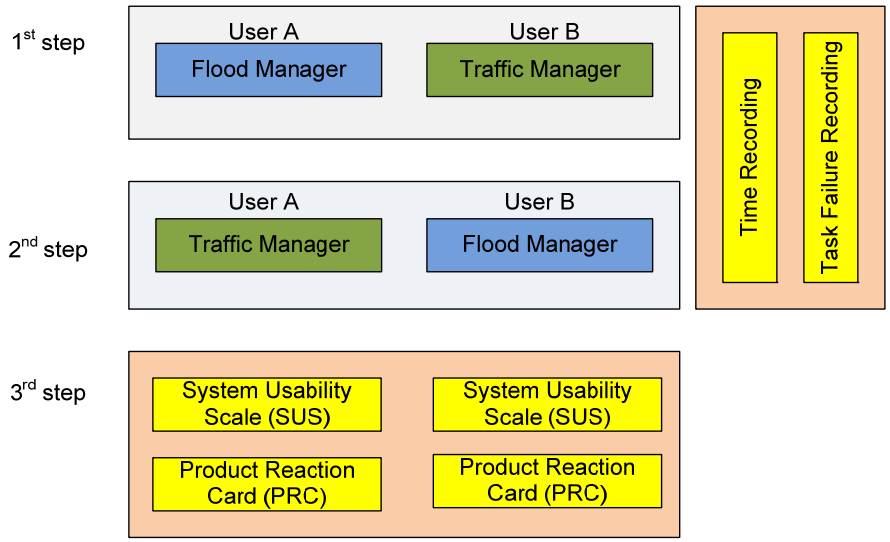


Figure 7 Methodology approach. The record tasks are reported in yellow.

The user is asked to perform some flooding and traffic simulation specific tasks using his/her laptop, and then he/she triggers partial or total migration (depending on the case) and precede the interaction on the destination device (wall or partner PC). The complete tasks list used and the detailed data collected during the test execution are reported in Appendix 1.C.1.

2.3.3. TEST ENVIRONMENT, EQUIPMENT AND LOGISTICS

In order to correctly execute the usability test the following equipment were necessary:

- 2 laptop devices, one for the flooding expert and another for the traffic one;
- 1 computer, either desktop or laptop, connected to a multimedia projector or a big screen for the shared wall;
- a network infrastructure available, better if wireless;
- 2 “Emergency prototype tutorial” printed, one for each person involved in the test. They can consult the document every time that they consider necessary.

2.3.4. FIRST TEST RESULTS

Once the task lists were validated by SAP and Arcadia Design teams a first test was executed by Vodafone team in order to understand which is the expected amount of time to perform the test. This activity is also important to understand if the test usability methodology applied to the Emergency prototype is correct or not.
The estimation of required time is done using a wireless network infrastructure shared with other devices, so reproducing a real scenario where not always a dedicated network is available. In the same way the task list execution was performed in a very relaxed way keeping into accounts that the tester who was involved in testing activity was a newbie user for the application. These data were exported neither to the test moderator nor to the tester, so that they were not influenced by them during the execution and data collection. In Table 2 the estimations are reported in

seconds. Please note that the estimated times are different depending from the application part considered. It is so also because the Traffic Expert has more complex tasks to execute according to the task list.

Application part	Upper bound estimated time [s]
Flood Manager	60
Traffic Manager	400
Wall	30

Table 2 Emergency upper bound execution time estimation

2.3.5. EFFECTIVENESS RESULTS

According to the principles described in Section 2.1 here are presented the results obtained from the Emergency usability test considering the effectiveness of the OPEN platform together with the Emergency prototype. For the effectiveness results analyses it is important to remind that there is a *task failure* if during the task execution occurs one of the following case:

- the user is not able to complete the task;
- user executes actions different from to the required by the task list.

While, as reported in Figure 8 there is an *application failure* when something goes wrong in the application and there is an application error. These cases are reported in red in the plot. In this way, considering as 100 the number of all tasks executed by the entire user sample it is possible to obtain the plot reported in Figure 8. Analyzing the plot the following consideration can be carried out:

- A small percentage of tasks have not been passed (2.71%), but with a deeper analysis this number results even smaller if the 1.63 % of application error is excluded. Note that it makes sense to distinguish the task failure reason because from the usability point of view the deterioration of the usability is devoted to prototype problems about technological aspects and not to problem in interface or presentation layer.
- No task has caused the test abandonment.
- Almost all tasks (97.28%) have a successful execution.

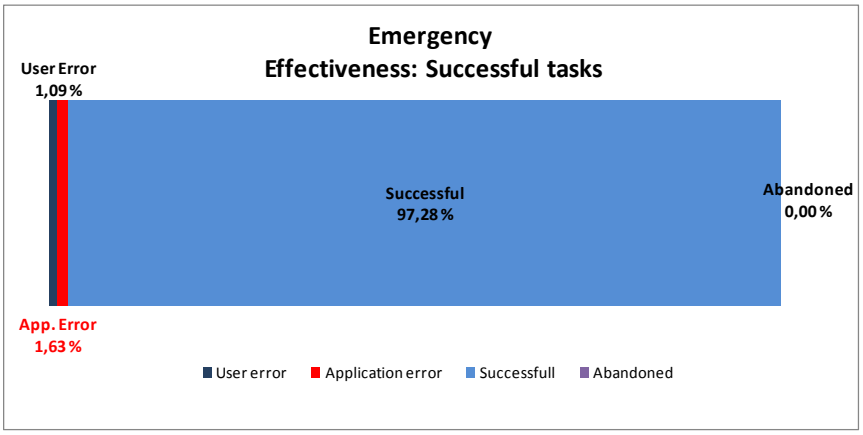


Figure 8 Emergency effectiveness results

Analyzing in a deeper way the collected data it is possible to understand how failures are distributed, in order to formulate some hypothesis about why they occur. In Figure 9 the failure distribution are reported in detail. Figure 9 shows the failures directly related to the user: these happen only during the 1st step execution. These results are expressive about the so called "learning problem", in fact all the errors are concentrated in the 1st step execution. It means that the user fails a task with a higher probability when he does not know the application. This addresses two other considerations:

- The tutorial guide written on paper is not so effective for the user: it can be more interesting to provide a video demo as example.
- The application, even if it is still a prototype, is simple to use because a single run is enough to perfectly learn its capabilities and obtain no user error in subsequent tests.

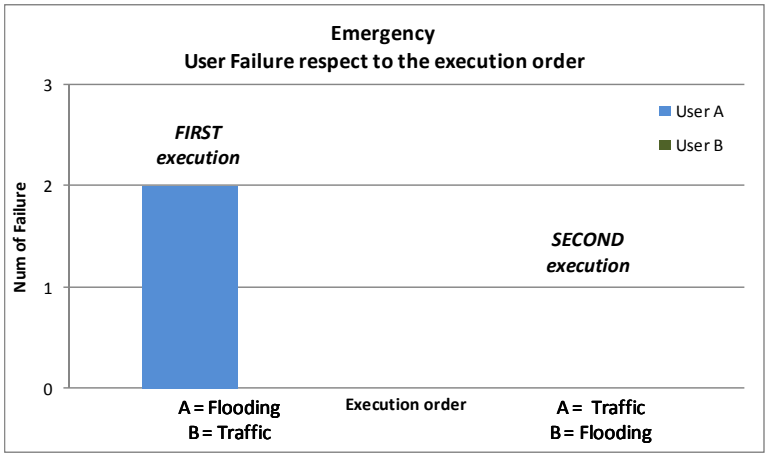


Figure 9 Emergency user failures respect to the execution order

2.3.6. EFFICIENCY RESULTS

Efficiency is the second quantitative indicator about usability and it is related to the task list execution time. Once the task list has been fixed and before the test execution Vodafone team has estimated the time necessary to complete each task list as described before.
In Figure 10, Figure 11 and Figure 12 the execution time distribution for flood, traffic and wall tasks are reported. In all the plots there are the task list execution classes on x-axis and the number of people that fall in each range on y-axis. The color bar in overlay to x-axis ticks shows the estimated time to execute the task list (see Table 2). Each plot represents with a light color the data collected when the user executes the specific task list (flooding, traffic or wall) as first step, while with a dark color when the data refers to the second step (e.g.: in Figure 10, the bar representing users executing flood task list in a time between 10 and 25 seconds is composed by a light blue segment indicating that 2 users execute flood task list as first step and a dark blue segment indicating that 2 users execute flood task list as second step).

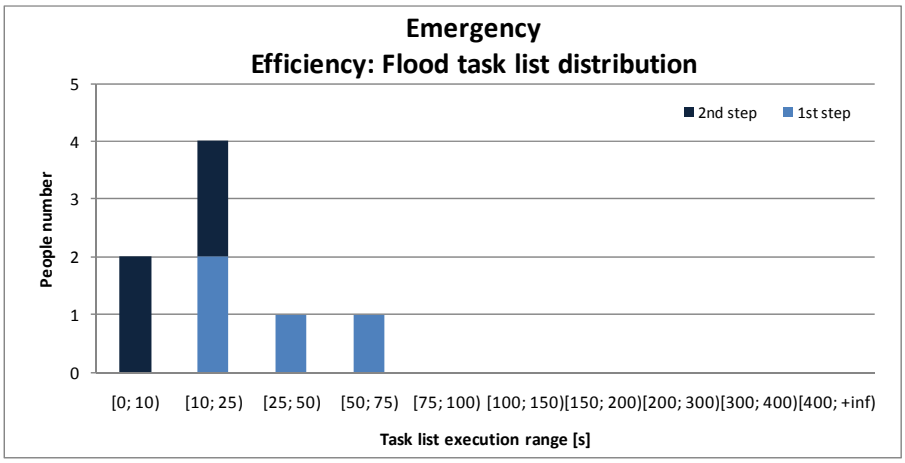


Figure 10 Emergency efficiency results about Flood Manager

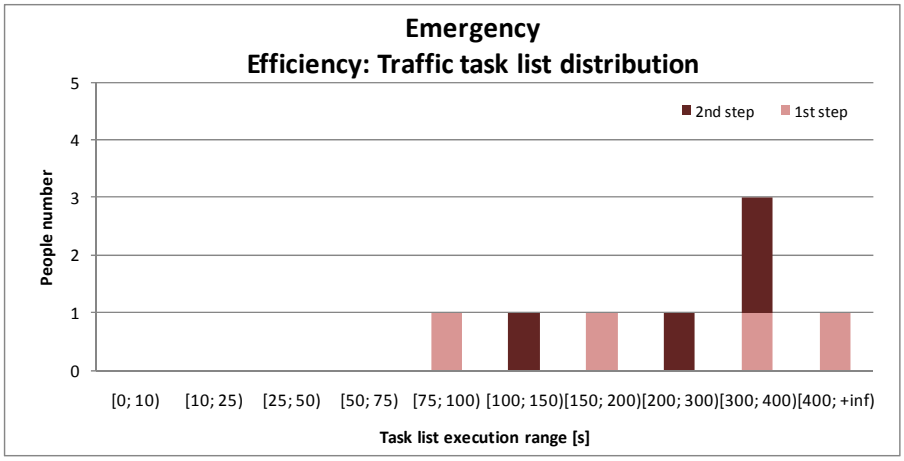


Figure 11 Emergency efficiency results about Traffic Manager

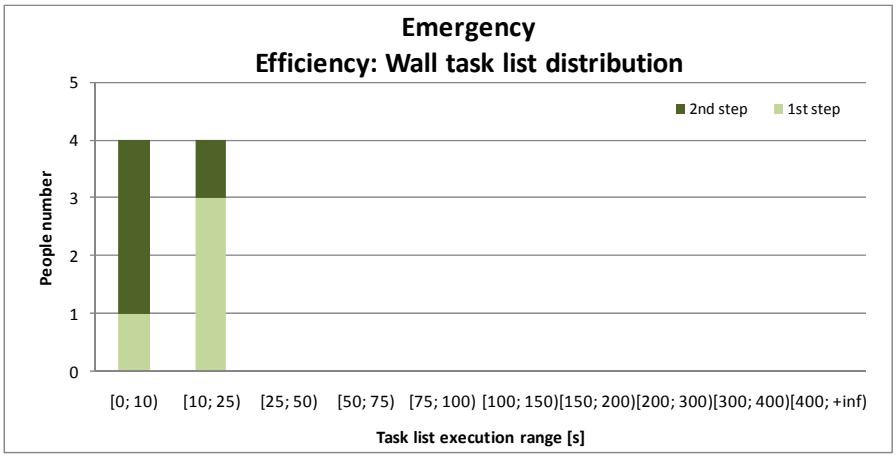


Figure 12 Emergency efficiency results about Wall

Observing the distribution histogram the following considerations can be shared:

- Estimation times are in general respected by the test execution.
- All the distributions are quite focused around a central class. It is positive because different users answer to the test more or less in a close time and it can be a positive indicator of how the platform is usable. The only exception is in Traffic task distribution where the people are more distribute. Note that this is caused by the application errors that have been already cited in the effectiveness analysis.
- The 2nd step tends to shift the distribution towards small classes in all cases. In the plot it is showed by dark colors which are more focused on the left in respect to the light ones.

The same consideration can be carried out also analyzing the distribution box plot. Figure 13, Figure 14 and Figure 15 show the distribution box plots for the Flood and Traffic Manager as well as for the wall and in respect to the other plots they provide the distribution shapes with median and quartile information. A detailed description of the box plot structure is reported in Section 2.1.3. The box plots representation highlights better how the 2nd step tends to shift and compress the distribution towards small classes.

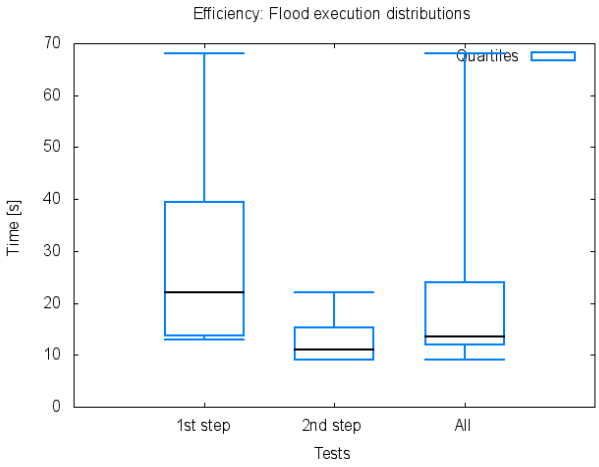


Figure 13 Emergency box plot efficiency results about Flood Manager

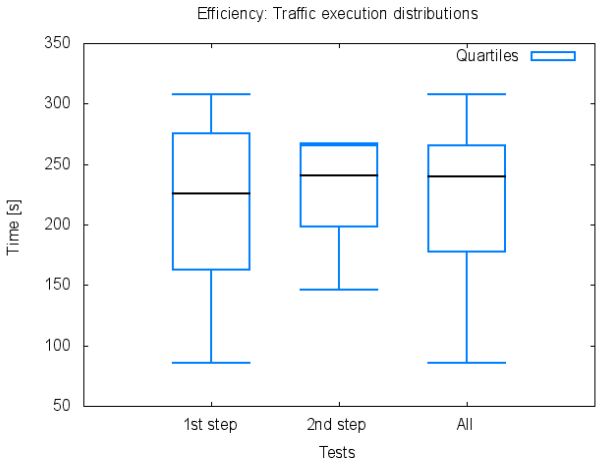


Figure 14 Emergency box plot efficiency results about Traffic Manager

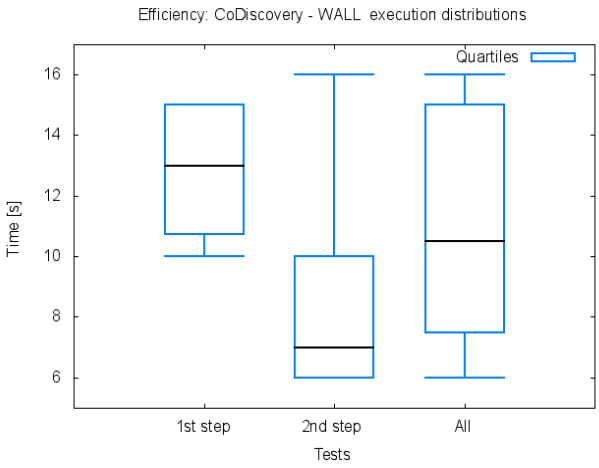


Figure 15 Emergency box plot efficiency results about Wall

2.3.7. SATISFACTION RESULTS – SYSTEM USABILITY SCALE

Here are showed the results carried out using the SUS questionnaire to evaluate the user satisfaction. The evaluation methodology follows the classification described in detail in Section 2.1.4.1. In Figure 16 the classification obtained during the performed tests is reported. During this test the average SUS score obtained is 45 % with a standard deviation of 21 %. It is clear also from the chart that it is a good result..The same SUS score methodology and classification is applied to another group of users considering a shorter tasks list that involves only two personal devices (Flood Manager and Traffic Manager) without the shared Wall interaction. This test was executed by SAP team and, as reported in Figure 17, it clearly shows a better distribution considering the user satisfaction point of view. The SAP task list is completely available in Table 41 for a deeper comparison.

Observing the user comments and behavior it is interesting to note that Emergency management is an inherently complex activity and often not trained people struggle to understand why a complex application is required to manage this kind of events. So a shorter and less complex task list, like the SAP one, allows having better results because the application is perceived as simpler by users not exploiting all application interaction possibilities.

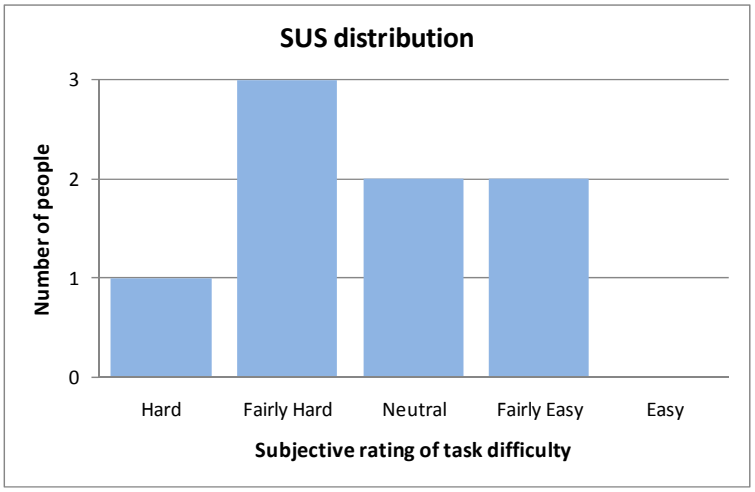


Figure 16 Emergency satisfaction results - SUS results

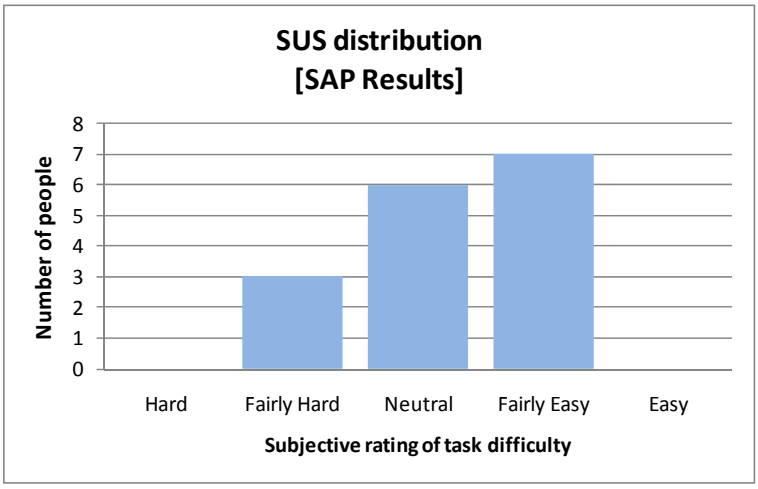


Figure 17 Emergency satisfaction results using SUS from SAP, using a different task list

2.3.8. EVALUATION CONCLUSIONS

Even if Emergency is still a prototype, in this second testing iteration a strong positive result has been obtained as highlighted in the effectiveness (successful tasks: 97.28%) and efficiency (task list completed within the estimated time: 100%) data.
About user satisfaction the SUS score shows a medium level of usability, but the development is on good track.
The modalities in which task lists are proposed to the user influence the satisfaction score given by a user, e.g. considering the different results obtained using the simpler and with less devices SAP task list rather than the Vodafone one. lists However the Product Reaction Cards approach highlights how the user already perceive a positive experience using the migration that can be surely improved with a more stable release of the Emergency application (see 0).
The overall results are good in term of usability for the Emergency and can be extended to the entire class of Emergency/Major Events applications.

2.4. WEB PAGE MIGRATION EVALUATION

In this section it is reported the usability evaluation of the migration support for Web applications designed and developed at CNR. Since also the Social Game developed by Arcadia is a Web application, it has been conducted a usability evaluation of its migration obtained through the general support for Web migration. The task list was proposed by Vodafone team and validated by CNR team. The underlying idea was that each user has to test two experiences related to web migration. The first experience analyzed here refers to the partial migration of the Social Game components which are rendered through a web browser. As second step, two different web sites were involved in the total migration test in order to cover more or less all the most used HTML components. In particular the advanced search page of eBay (http://shop.ebay.com/ebayadvsearch/) was chosen for the large number of forms present. Then also the W3C public account request page (http://www.w3.org/Help/Account/Request/Public) was considered. In order to have the comparison between partial and total migration experience

the same 10 users, properly divided into 2 subgroups, were involved in both evaluations. The detailed approach is reported in Figure 18 where the data logging steps are clearly showed because highlighted in yellow. To put in practice the above described approach a 4 steps test plan was considered.

1. Users belonging to group A tested the Social Game with partial migration on multi devices while group B users tested the web migration using well known web pages taken from the Internet. During this phase all the task execution time and eventually application or task errors have been recorded.
2. After the first execution step a first SUS questionnaire was submitted to both users group. In this way the SUS filled by user A gives feedbacks about how the Social Game with partial migration experience is perceived while the user B gives information about the total migration one.
3. Then another test was proposed to both users swapping the application under test between the users. Also in this case all the execution times and possibly failures were recorded.
4. Finally a second SUS questionnaire is submitted to both users, so that it is possible to have feedbacks about the new migration experiences lived by users. Following a unique Product Reaction Card (PRC) evaluation was submitted to both users to evaluate the overall migration experience tested.

In order to test a representative number of migrations: PC to mobile, PC to PC and mobile to PC cases was considered. The complete tasks list used and the detailed data collected during the test execution are reported in Appendix C.2.

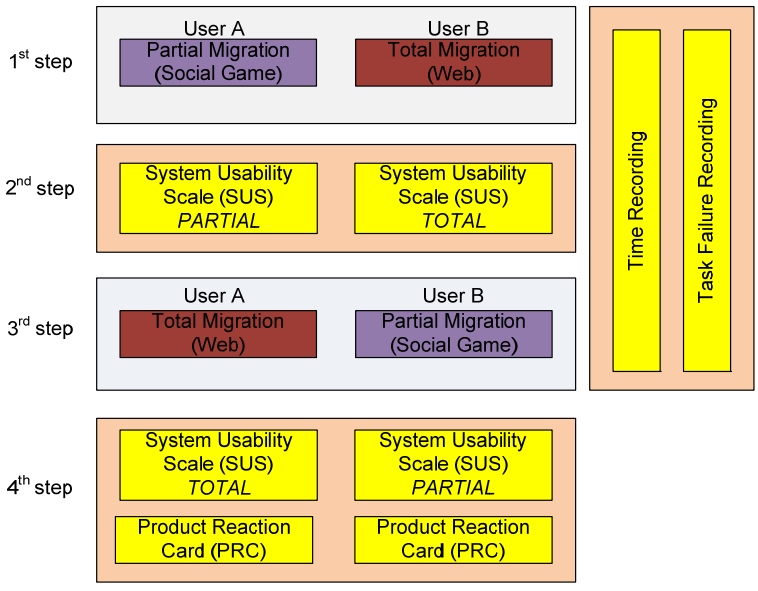


Figure 18 Methodology approach to test with the same users both Social Game and Web migration. Remind that Social Game uses partial migration while Web page migration exploits total migration. The record tasks are reported in yellow.

2.5. PARTIAL MIGRATION WITH SOCIAL GAME

Using the Social Game application for the OPEN Migration Service Platform evaluation means to test an example of entertainment application whose experience is enriched by the migration features enabled through the OPEN platform. In particular partial Web migration has been considered in this test.

2.5.1. PARTICIPANT CHARACTERISTICS

Evaluation tests are performed by CNR in the third week of June 2010 involving 10 people, equally divided into two subgroups A and B. All people involved in the tests are familiar with the use of the computer because they are university students or people involved in the CNR activities. In the sample both men and women were considered in order to have a representative situation of the possible application end user.

2.5.2. TASK LIST

The task list was proposed by the Vodafone team and validated by the CNR team. It was used inside the testing scenario described in Section 2.4.

2.5.3. TEST ENVIRONMENT, EQUIPMENT AND LOGISTICS

In order to correctly execute the Social Game usability test the following equipments are necessary:

- 2 laptop devices;
- 1 mobile device, in this case iPhone is used;
- a wireless network infrastructure available;
- 1 “Social Game prototype tutorial” printed. The tester can consult the document every time that they consider necessary.

2.5.4. FIRST TEST RESULTS

Once the task lists were validated by the CNR team a first test was executed in order to understand the required amount of time to perform the task list. This activity was also important to understand if the test usability methodology applied to the Social Game prototype is correct or not.
The estimation of time required is obtained using a wireless network infrastructure shared with other devices, so reproducing a real scenario where not always available a dedicated network is available. In the same way the task list execution is performed in a very relaxed way keeping into accounts that the tester which will be involved in testing activity is a newbie user for the application. Using this approach the estimated times are real and not too fast. These data have been exported neither to the test moderator nor to the tester, so that they are not influenced by them during the execution and data collection. In Table 3 the estimation is reported in seconds.

Application part	Upper bound estimated time [s]
Social Game with partial migration	500

Table 3 Social Game with partial migration upper bound execution time estimation

2.5.5. EFFECTIVENESS RESULTS

According to the principles described in Section 2.1 here are presented the results carried out from the usability test of the migration of the Social Game obtained through the OPEN platform. For the effectiveness results analyses it is important to remind that there is a *task failure* if during the task execution occurs one of the following case:

- the user is not able to complete the task; users execute actions different from the required by the task list.

As reported in Figure 19 there is an *application failure* when something goes wrong in the application and there is an application error. These cases are reported in red in the plot.
In this way, considering as 100 the number of all tasks executed by the entire user sample it is possible to obtain the plot reported in Figure 19. Analyzing the plot the following consideration can be carried out:

- A very small percentage of tasks have not been passed (1.11%). As showed in the plot they are all due to application error. Note that it has sense to distinguish the task failure reason because from the usability point of view the deterioration of user experience is devoted to prototype problems about technological aspects and not to problem in interface or presentation layer.
- No task has caused the test abandonment.
- Almost all tasks (98.89%) have a successful execution.

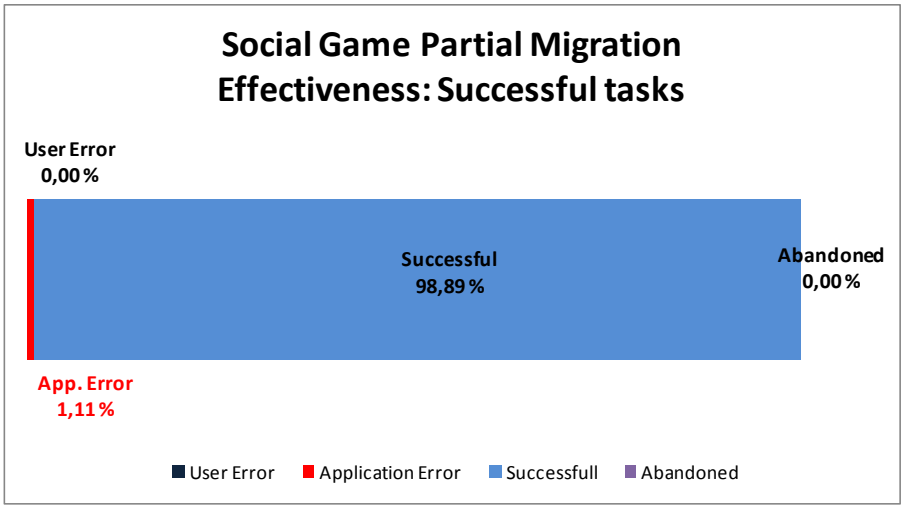


Figure 19 Social Game partial migration effectiveness results

2.5.6. EFFICIENCY RESULTS

Efficiency is the second quantitative indicator about usability related to the task execution time. Once the task list has been fixed and before the test execution Vodafone team has estimated the time necessary to complete each task list as described before.
In Figure 20 the execution time distribution for Social Game is reported. The plot considers the task list execution classes on x-axis and the number of people that fall in each range on y-axis. The color bar in overlay to x-axis ticks shows the estimated time to execute the task list.

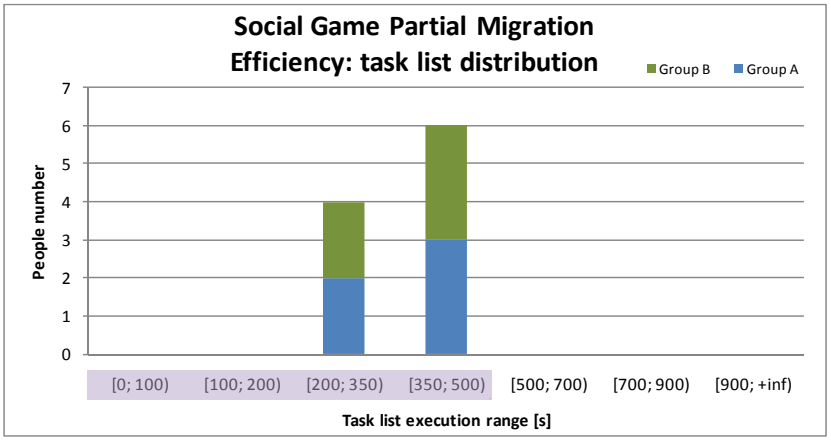


Figure 20 Social Game partial migration efficiency results

Observing the distribution histogram the following considerations can be shared:

- Recorded times are coherent with the proposed estimation.
- All data fill into two classes. It is positive because different users answer to the test more or less in a close time and it can be a positive indicator of how the platform is usable because perceived in the same way by different users.
- There is not any appreciable difference between the people that execute first the Social Game (group A) respect to the ones that executes Social Game after have tried web page migration.

The same consideration can be carried out also analyzing the distribution box plot. Figure 21 shows the distribution box plot that respect to the other plot it provides the distribution shapes with median and quartile information. A detailed description of the box plot structure is reported in Section 2.1.3. The box plots representation highlights better how the 2nd step tends to shift and compress the distribution towards small classes.

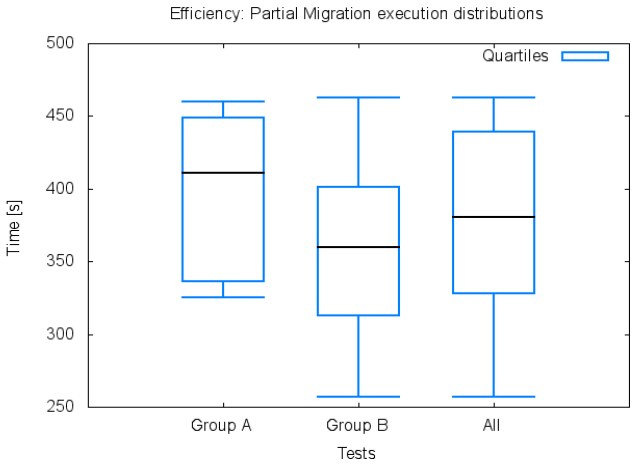


Figure 21 Social Game box plot partial migration efficiency results

2.5.7. SATISFACTION RESULTS – SYSTEM USABILITY SCALE

Here are showed the results obtained using the SUS questionnaire to evaluate the user satisfaction about the partial migration experience. The evaluation methodology follows the classification described in detail in Section 2.1.4.1. In Figure 22 the classification obtained during the performed tests is reported. During this test the average SUS score obtained is 75 % with a standard deviation of 11 %. Please note that the obtained score is a very good one especially considering the standard deviation associated to it.

Observing the user comments and behavior it is interesting to note that the support of migration applied to the Social Game was well perceived by the end user.

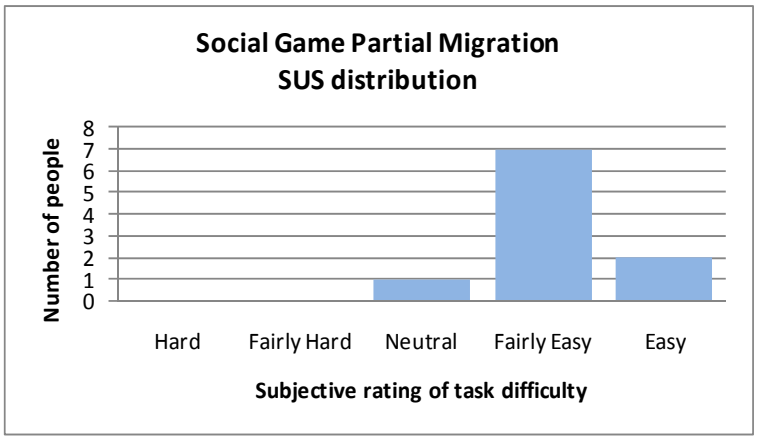


Figure 22 Social Game partial migration SUS results

2.5.8. EVALUATION CONCLUSIONS

The partial migration of web applications is evaluated by the user in a very positive way as interesting and very easy to use features available from the system. Moreover also the effectiveness (successful tasks: 98.89%) and efficiency (task list completed within the estimated time: 100%) results highlight positive performances.

2.6. WEB TOTAL MIGRATION

The web page migration enabled by OPEN Migration Service Platform using a standard internet browser has been considered in this test.

2.6.1. PARTICIPANT CHARACTERISTICS

The participants selected for this test are the same already described for Social Game usability test in Section2.5.1.

2.6.2. TASK LIST

The task list was proposed by Vodafone team and validated by CNR team. It is used inside the testing scenario described in Section 2.4.
The user starts surfing a web page using his/her desktop PC and then migrates to a mobile phone and vice versa. During the task list he/she compiles different forms and selects different page hyperlinks.

2.6.3. TEST ENVIRONMENT, EQUIPMENT AND LOGISTICS

In order to execute usability tests a mobile device is required. It is so necessary to select a single representative or a subset of significant representative mobile devices to be used. Note that more than the mobile device itself, the thing that really characterizes the user experience considering the internet surfing is the browser used.
So it is interesting to consider the most used mobile platforms/browsers in Europe up to April 2010. A market research is done using Global Stat/Stat Counter. It is a web analytics service. As of 1 August 2009, their tracking code is installed on approximately 3 million sites globally which cover various activities and geographic locations. Every month, Global Stat records billions of hits to these sites. For each hit, it analyzes the browser/operating system used and it summarizes these data. This is how Global Stat provides information. It does not manipulate the data in any way and it does not collate it with any other information sources. No artificial weightings are used. Global Stat simply publishes the data as they record it. Regarding the mobile browsers used in Europe from December 2008 to April 2010, Figure 23 shows the plot containing all the collected data.

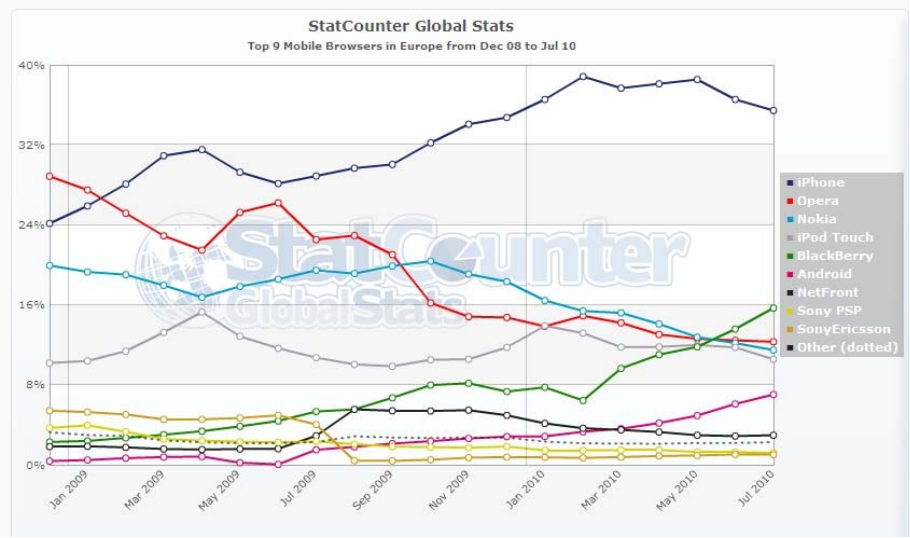


Figure 23 Top 9 Mobile Platforms/Browser in Europe from Dec 2008 to April 2010

From the plot it is clear how the iPhone, more precisely its Safari browser, is the most widely used browser to surf the Internet. The reported data have a European point of view and it is immediate the positive trend that this device has maintained during the last months. According to the statistics all the tests about User Interface adaptation that involve mobile devices will be executed using iPhone devices. In order to correctly execute the web page migration usability test it the following equipments are necessary:

- 2 laptop devices;
- 1 mobile device, in this case iPhone is used;
- a wireless network infrastructure available;
- 1 “User Interface migration prototype tutorial” printed. The tester can consult the document every time that they consider necessary.

2.6.4. FIRST TEST RESULTS

Once the task lists were validated by CNR team a first test was executed in order to understand the required amount of time to perform the task list. This activity is also important to understand if the test usability methodology applied to web page migration is correct or not.
The estimation of required time is done using a wireless network infrastructure shared with other devices, so reproducing a real scenario where it is not always available a dedicated network. In the same way the task list execution it is performed in a very relaxed way keeping into accounts that the tester which will be involved in testing activity is a newbie user for the application. Using this approach the estimated times are real and not too fast. This data have been exported neither to the test moderator nor to the tester, so that they are not influenced by them during the execution and data collection. In Table 4 the estimation is reported in seconds.

Application part	Upper bound estimated time [s]
Web page migration	1000

Table 4 Web page migration with partial migration upper bound execution time estimation

2.6.5. EFFECTIVENESS RESULTS

This section presents the results carried out from the web total migration usability test considering the effectiveness of the OPEN platform used to enable migration of Internet web pages with a standard browser, Mozilla in the test case. For the effectiveness results analyses it is important to remind that there is a *task failure* if during the task execution occurs one of the following case:

- the user is not able to complete the task; users execute actions different from the required by the task list.

While there is an *application failure* when something goes wrong in the application and there is an application error. These cases are reported in red in the plot.
In this way, considering as 100 the number of all tasks executed by the entire user sample it is possible to obtain the plot reported in Figure 24. Analyzing the plot the following consideration can be carried out:

- A very small percentage of tasks have not been passed (0.91%). As showed in the plot they are all due to user error, and in particular the problem is that a form was not identified by the user in the adapted web page.
- No task has caused the test abandonment.
- Almost all tasks (99.09%) have a successful execution.

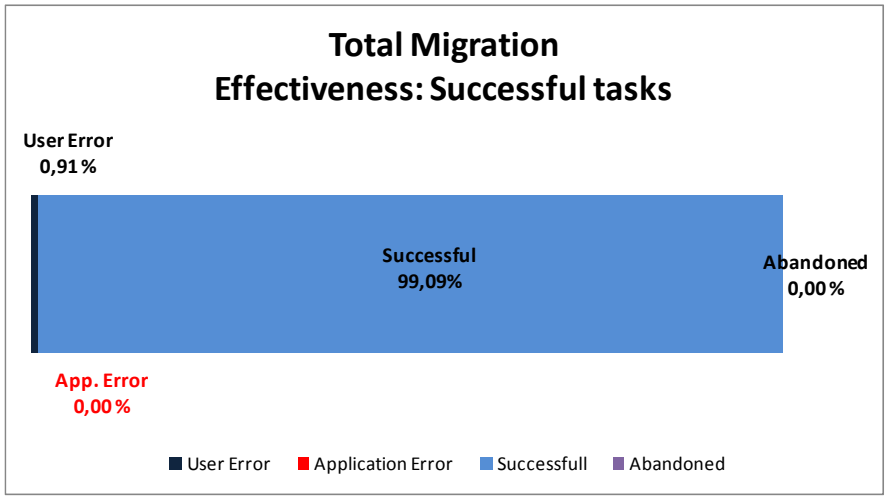


Figure 24 Web Page Total migration effectiveness results

2.6.6. EFFICIENCY RESULTS

Efficiency is the second quantitative indicator about usability related to the task execution time. Once the task list has been fixed and before the test execution Vodafone team has estimated the time necessary to complete each task list as described before.
In Figure 25 the execution time distribution for Web total migration is reported. The plot considers the task list execution classes on x-axis and the number of people that fall in each range on y-axis. The color bar in overlay to x-axis ticks shows the estimated time to execute the task list.

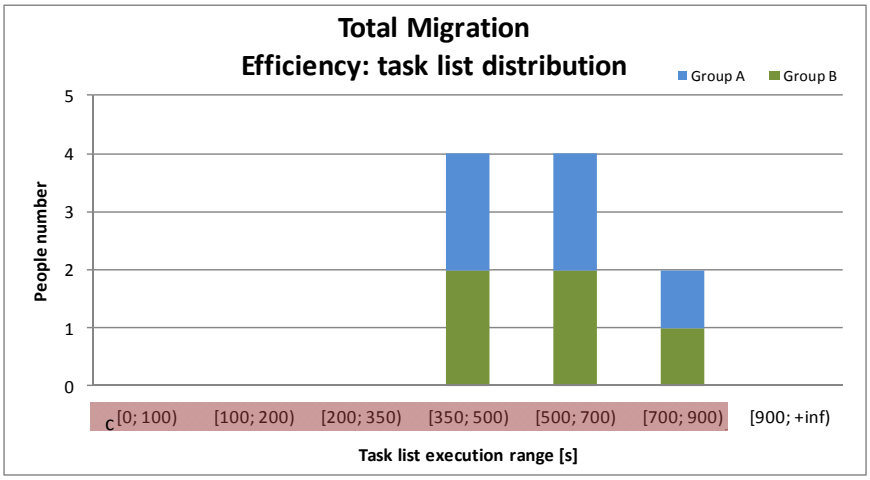


Figure 25 Web Total migration efficiency results

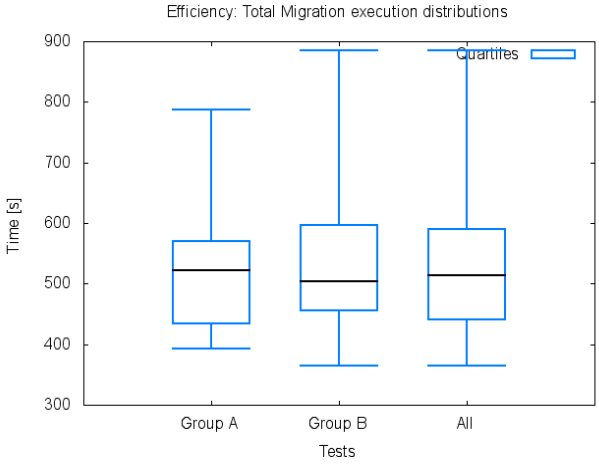


Figure 26 Web Page Total migration box efficiency results

Observing the distribution histogram the following considerations can be shared:

- recorded times are coherent with the proposed estimation.
- All data fill into three classes. It is positive because different users performed the test more or less in similar time and it can be a positive indicator of how the platform is usable because perceived in the same way by different users.
- There is not any appreciable difference between the people that execute first the Web Page migration (group B) respect to the ones that executes first partial migration and after the total migration (group A).

The same consideration can be carried out also analyzing the distribution box plot. Figure 26 shows the distribution box plot that provides the distribution shapes with median and quartile information. A detailed description of the box plot structure is reported in Section 2.1.3. The box plots representation highlights better any difference among the data recorded from user group A and user group B.

Comparing the web total migration efficiency results with the Social Game partial migration one it is clear how some users tend to have difficulties with total migration. The reason is that if the user has already filled some forms the migration tool automatically selects the filled forms for a partial migration. On the contrary in order to have a correct total migration it is necessary to manually deselect all the page components because no selected components for the platform means migrate the entire page.

Just to clarify this aspect in Figure 27, the user has just clicked with the mouse on the part of the UI visualizing the form. The migration platform interpreted this as a potential willingness of the user to migrate this portion of the UI and, as a feedback; a different background color (green in this case) is used to visualize that part of the page. If a migration is triggered in this state of the page, the migration platform will perform a *partial* migration of the page (only the form will be migrated). In Figure 28, the user has just clicked again with the mouse on the same portion of the UI visualized in Figure 27. As you can see from Figure 28, this action performs a "de-selection" of the concerned area (the different background color disappears). If a migration is triggered on the page of Figure 28, the migration platform will perform a *total* migration, which is the *default* migration type (no specific UI portion is selected).

During the tests, before triggering a *total* migration, if any part of the considered web page was highlighted using a different background color, the users had just to deselect such part, if any.

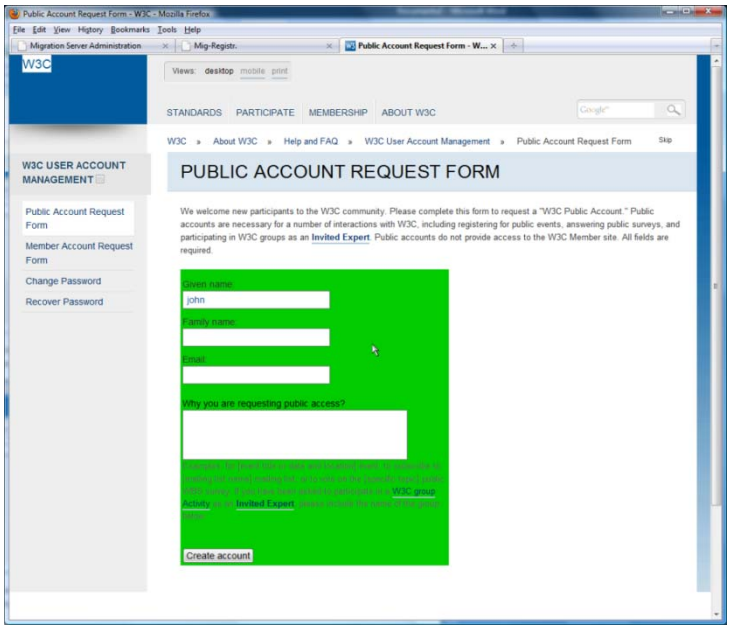


Figure 27 Only a part of the page has been selected for migration (see the green background color)

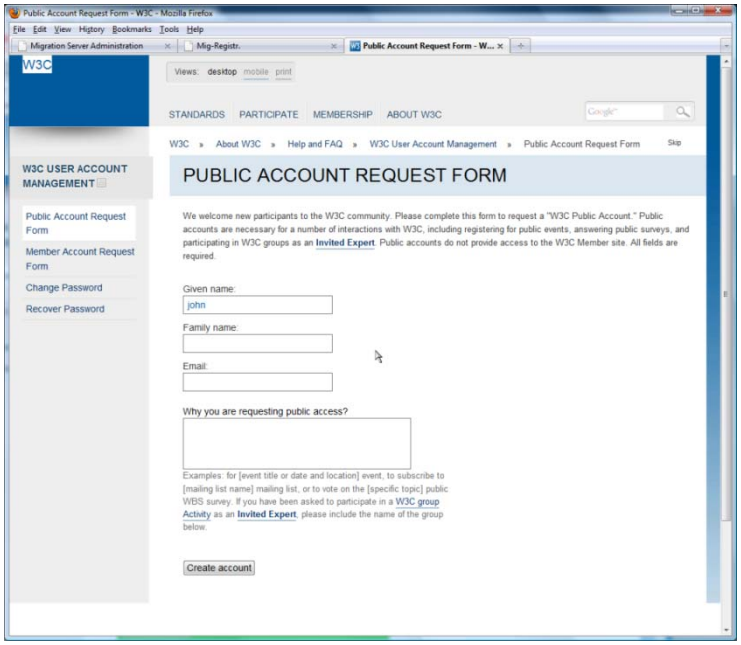


Figure 28 Clicking again on the same part of the UI visualized in Figure 27, the different background color disappears ("un-selection" of the concerned UI portion)

2.6.7. SATISFACTION RESULTS – SYSTEM USABILITY SCALE

Here are the results carried out using the SUS questionnaire to evaluate the user satisfaction about the total migration experience. The evaluation methodology follows the classification described in detail in Section 2.1.4.1. In Figure 29 the classification obtained during the performed tests is reported. During this test the average SUS score obtained is 65 % with a standard deviation of 22 %. Please note that the obtained score is good even if not as good as the one obtained with partial migration tests. This can be established also observing the standard deviation which is not as small as in the partial migration executed tests.

Observing the user comments and behavior it is interesting to note that Web migration is a sort of utility able to enrich the standard Internet surfing experience enabled by browsers. This is well perceived by the end user. Considering also the Internet context in which the solution works also not trained people need very little time to learn the application migration mechanism and appreciate its features. Observing the plot reported in Figure 29 there is a single outlier in the hard section. This outlier user evaluates not so good the experience but however does not export any particular comment about any specific platform problem.

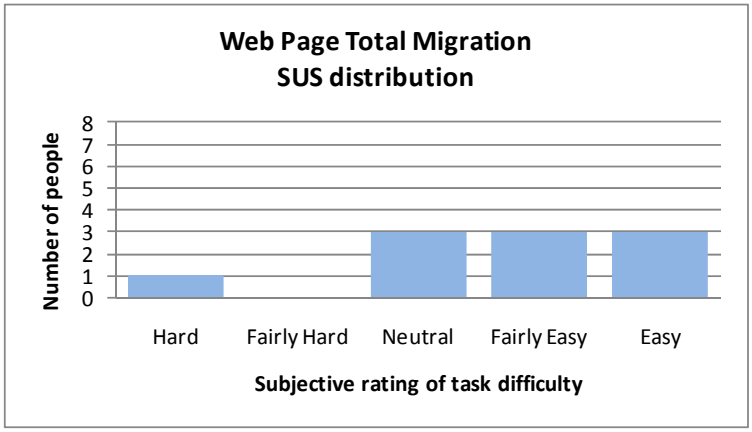


Figure 29 Web Page Total migration SUS results

2.6.8. EVALUATION CONCLUSION

This class of test uses the web migration prototype which enables migration features using a Firefox Mozilla browser, but it works also with other commonly used browsers. As it is detailed in the above data very good results about effectiveness (successful tasks: 99.09%) and efficiency (task list completed within estimated time: 100%) are obtained. Another interesting thing is about the total migration of web applications which is evaluated by the user in positive way even if not so good as the partial migration one.

2.7. USER INTERFACE ADAPTATION COMPARATIVE ANALYSIS

The goal of this Section is to perform a competitor analysis related to the adaptation features available in the OPEN MSP. The idea here is to evaluate only adaptation features and not the complete migration process.

2.7.1. PARTICIPANT CHARACTERISTIC

Evaluation tests were performed by CNR in the last week of May 2010 involving 18 people. All the participants have never used the OPEN User Interface Adaptation features before. All people involved in the tests were familiar with the use of the computer for the Internet surfing activities. On the other hand not all of them were used to Internet navigation through mobile device.

2.7.2. TASK LIST

The aim of this test is to compare 3 different rendering solutions on mobile devices. In order to avoid results bias all the 6 permutations without repetition of 3 elements are considered. In this way 6 different groups of users were created, each one with a different execution order as reported in Table 5. The OPEN MSP provides the adaptation but it requires a browser for adapted pages visualization. For the testing activity the Safari browser was used.

Group	First	Second	Third
Group1	Safari	OperaMini	OPEN MSP
Group2	OperaMini	OPEN MSP	Safari
Group3	OperaMini	Safari	OPEN MSP
Group4	Safari	OPEN MSP	OperaMini
Group5	OPEN MSP	Safari	OperaMini
Group6	OPEN MSP	OperaMini	Safari

Table 5 Group structure for user repartition

Each one of the 6 groups involves 3 users, so overall there are 18 involved users. The number of users in each group is fixed at 3 according to the recommendations presented in [HUT-08] in order to have not too many people involved in the test. In fact always in [HUT-08] it has been demonstrated that no appreciable improvements about the collected data can be recorded involving a greater number of users.
The task list is always the same and was executed by each user with each one of the 3 browsers considered. The browser execution order is fixed by the user group membership. The task list is composed by 3 parts, each one that uses a different web-site with different characteristic to be adapted. In this way all the adaptation features declared by the OPEN MSP were stressed and compared with available commercial products. In the task list were considered a travel web site (British Airways – www.britishairways.com), an e-commerce well known portal (eBay – www.ebay.com) and finally the W3 consortium web pages (www.w3.org).
The task list was developed by Vodafone team specifically to stress the main elements that compose a web page (texts, pictures, links, and tables). After a first draft it was shared with CNR that validated and fixed it for the test execution.

2.7.3. TEST ENVIRONMENT, EQUIPMENT AND LOGISTICS

According to the reasoning presented above, also in this case the device selected for the test is the iPhone (firmware 3.1.2). The first commercially available browser chosen is the iPhone native one, which is Safari. The other commercial product considered for the test is OperaMini in its version 5.0. In order to correctly execute the comparison usability test the following equipments are necessary:

- 1 iPhone mobile device connected to the Internet with installed Safari and OperaMini 5.0, it has been able to access the OPEN server;
- a wireless network infrastructure available for the mobile device connectivity;

2.7.4. EFFECTIVENESS RESULTS

According to the principles described in Section 2.1 here are presented the results obtained from the adaptation comparison usability test considering the effectiveness of each one of the compared solutions. For the effectiveness results analyses it is important to remind that there is a *task failure* if during the task execution occurs one of the following case:

- the user is not able to complete the task; users execute actions different from the required by the task list.

While there is an *application failure* when something goes wrong in the application and there is an application error. These last cases are reported in red in the plot.
In this way, considering as 100 the number of all tasks executed by the entire user sample it is possible to obtain the plots reported in Figure 30, Figure 31 and Figure 32. Analyzing the plots the following consideration can be carried out:

- There is no task failure for OPEN MSP caused by the user. While they are present respectively with 6.35% and 2.77% for Safari and OperaMini. Analyzing the user comments the task failures are mainly caused by the selection of a specific link on British Airways web site that results not sufficiently highlighted by Safari and OperaMini, while it is clearly shown by OPEN MSP.
- There are few application failures for OPEN MSP (red part in Figure 31 3.97 %) with respect to no errors in Safari and very small number in OperaMini (0.40%). It has to be highlighted that OPEN MSP is still a prototype while Safari and OperaMini are already commercial products available on the market. It is interesting that also a commercial product like OperaMini has exposed a bug (button without label in such cases) using the proposed task list. It means that they effectively stress the page adaptation features considering different HTML components.

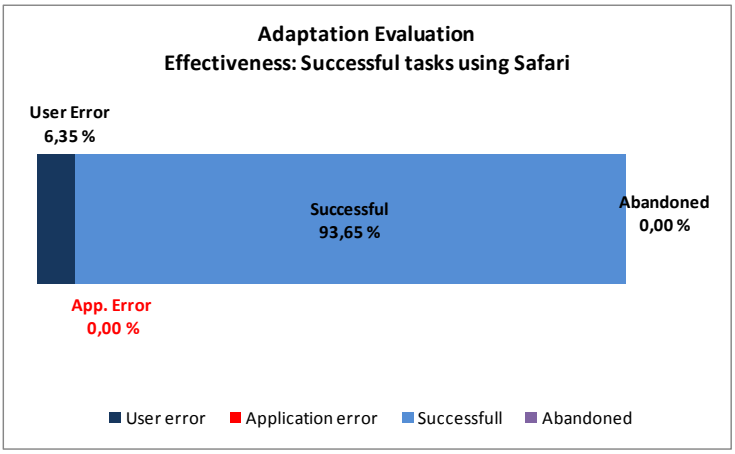


Figure 30 Adaptation evaluation effectiveness results using Safari standard browser

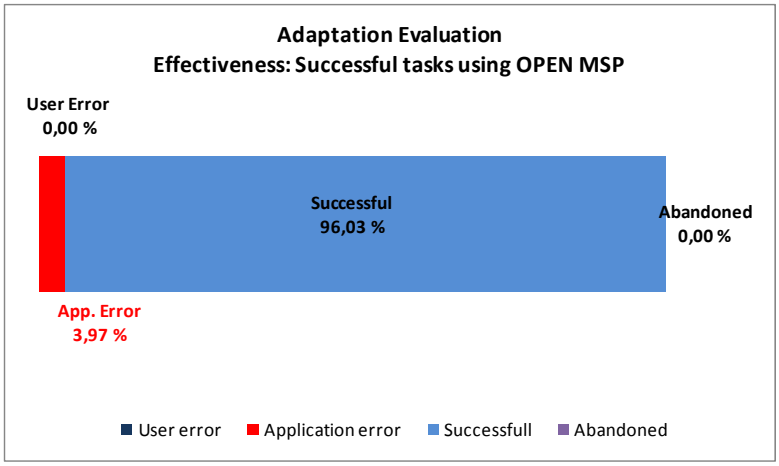


Figure 31 Adaptation evaluation effectiveness results using OPEN MSP WebUIAdaptation tool

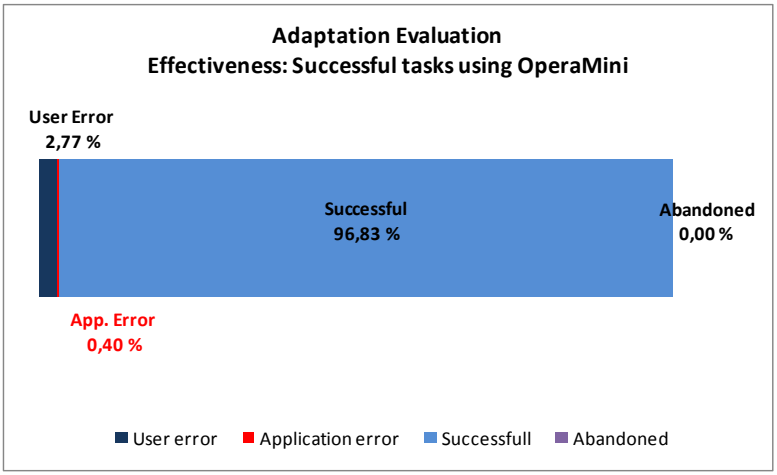


Figure 32 Adaptation evaluation effectiveness results using OperaMini browser

2.7.5. EFFICIENCY RESULTS

Efficiency is the second quantitative indicator about usability related to the task execution time. Unlike in other usability tests here a usability comparison test is in execution, so none time estimation for task list completion is provided.
The recorded time does not take into account the amount of time required to download the first pages of the considered web site. Considering the proposed task list it means that for the efficiency analysis it is not taken into account the time to download the first page of British Airways, eBay and W3c. Another important aspect to consider is that both for Safari without adaptation and when it is coupled with OPEN MSP the browser cache and history are cleaned. On the contrary using OperaMini it is possible only to delete the browser history while the cache is stored on the OperaMini server which is provided by Opera to the users and it is out of our control. Please note also that the 6 user group composition is balanced so the same numbers of users execute as first run each one of the considered solutions.
Considering all the above described shrewdness in Figure 33, Figure 34 and Figure 35 the time distribution with 3 considered solutions is reported in detail. On abscissa the task list execution ranges are reported and for each class the colored bar represents how many tester fall into the considered time class. Starting from the efficiency plots the following points can be carried out:

- OPEN MSP execution times are greater than others perhaps due to the page splitting features.
- OPEN MSP page splitting assures semantic grouping of information also for pages that contain a lot of information. This thing is a strong point particularly appreciated by the user as reported in the satisfaction analysis reported in following section.

In Figure 36 a box plot representation of the execution time required by all the considered solution is reported. This information clearly shows how the distributions of all the three solutions are quite compact considering the 1st and 3rd quartiles boundary (the light blue rectangle). It means that more or less all the users require a similar amount of time. This confirms the fact that adaptation is good because even if tested by different kind of users and in different order the results in term of execution time are not so different.

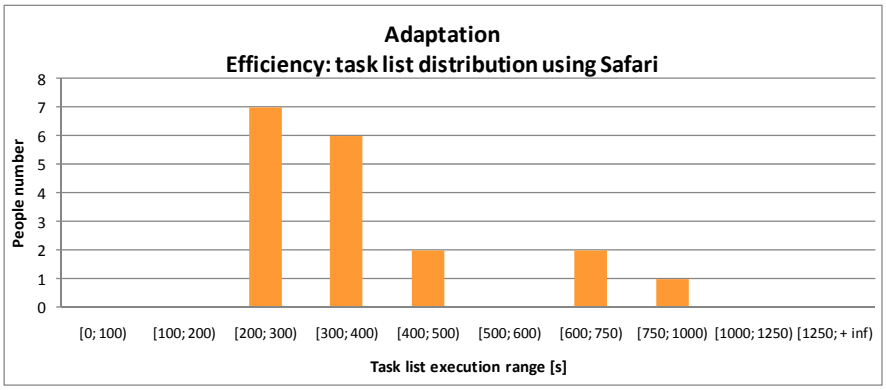


Figure 33 Adaptation evaluation efficiency results using Safari standard browser

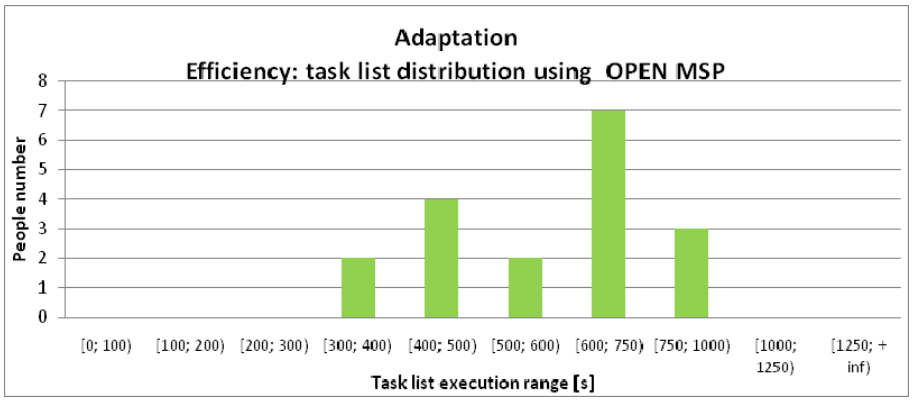


Figure 34 Adaptation evaluation efficiency results using OPEN MSP WebUIAdaptation tool

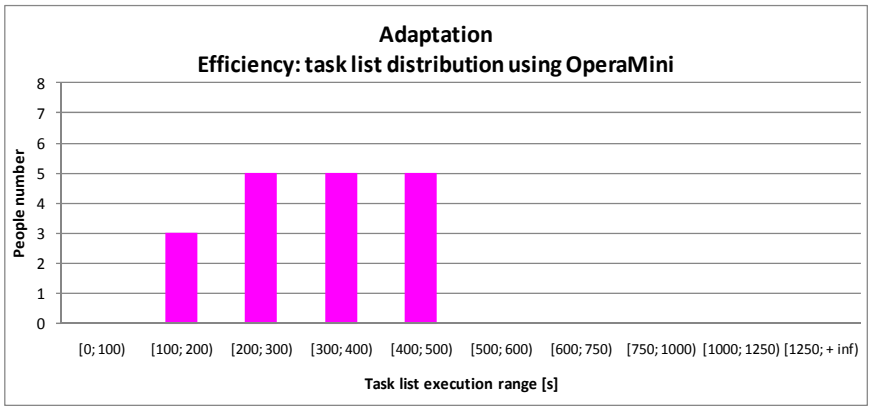


Figure 35 Adaptation evaluation efficiency results using OperaMini browser

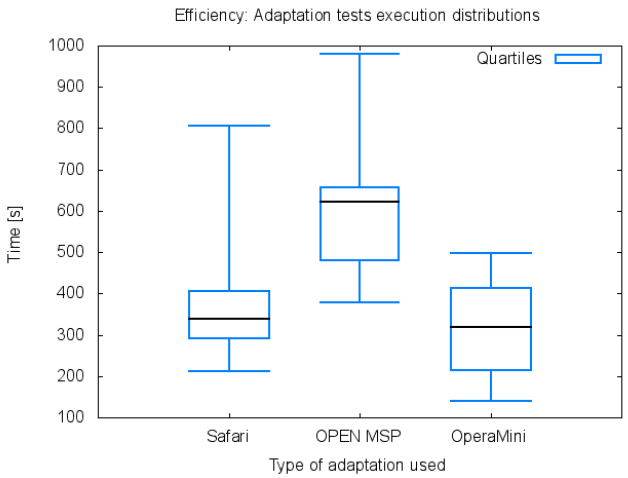


Figure 36 Adaptation evaluation box plot efficiency results. Tested solution comparison.

2.7.6. SATISFACTION RESULTS

In this paragraph the user satisfaction analysis is reported. Please note that the applied methodology here is completely different with respect to the previously proposed ones. It is important to keep into account that here the objective is a comparative analysis among the OPEN MSP prototype tool and two other commercially available web page adaptation solutions (Safari and OperaMini). Considering this scenario the SUS and PRC approaches are not used because they are specific to give a satisfaction evaluation from the user point of view considering the experience obtained with a single user interface version. On the contrary the main objective here is to compare how different solutions are perceived by the user focusing attention on how specific elements present in web pages are adapted.

Considering this scenario an ad-hoc questionnaire was developed with the objective to understand how the user perceives the adaptation. The questionnaire is made up by 4 sections and it is completely available in Appendix C.5.3. Here after the sections are described one by one and then obtained results are presented.

- *Section I: General Information*. This part collects some general information about the date and time in which the test has been executed. Moreover it requires to the user to indicate his/her gender, age and qualification and how often he/she surfs over the Internet both using desktop PC rather than mobile device.
- *Section II: Usability Evaluation*. This second section considers more quantitative aspects in respect to the first one. It contains 10 questions where a mark between 1 (poor) to 5 (very good) is required, eventually adding some free comments. The complete questions list is reported in Figure 37. For each of these questions a mark for each tested solutions is asked to the user. In the considered case these are Safari standard, OPEN MSP and OperaMini. Please note that there is the question number 2 which is quite general, but it is followed by 5 specific sub-questions that aiming to verify the satisfaction level for each one of the main elements that usually characterizes a web page.

LEGENDA

1) Is it clear how to the considered web site is organized?
2) Is it easy to identify the different elements into the web pages?
   - 2.a) What about menus? Are they comprehensible?
   - 2.b) What about links? Are they easily identifiable in the page?
   - 2.c) What about forms? Is it simple to understand and correclty compile the form fields?
   - 2.d) What about images? Are tehy comprehensible?
   - 2.e) What about data tables? Are they clearly readable?
3) Do you think that is correcly balanced the quantitiy of information available in a single web pages?
4) What do you think about the time required to obtain the adapted version of the web page?
5) What do you think about the overall usability of the adapted web site?

Figure 37 Satisfaction questionnaire Section II questions

Here after in Figure 38 and Figure 39 the scores obtained in this second section are collected and reported. In Figure 38 the score reported by each question is summarized among all users while Figure 39 shows the overall results considering the average of the results obtained by each single question. Reading the plot in Figure 38 the following consideration can be done:

- OPEN MSP solution is usually in mid position.
- OPEN MSP implements a very good strategy (better than competitors) to mitigate “information overload” for web pages on mobile devices. This is highlighted by question 3 result which shows how the user appreciates the page splitting feature.
- Some other OPEN MSP strong points are highlighted by question 2.e and 2.d identifying how the tables and images adaptation are respectively better and comparable with Safari which is the main competitor.
- Please note that Safari “plays at home” being the test executed on iPhone so that could be a high loyalty by the Apple users.
- Data are quite compact being the browser market is already a mature market; however there are niches in which OPEN MSP is better than competitors.

The overall analysis reported in Figure 39 provides a browser classification considering user satisfaction, starting from the most to the less appreciated the result is:

1. Safari
2. OPEN MSP
3. OperaMini

Observing both the average value reported and the corresponding standard deviation can be pointed out the following points:

- OPEN MSP which is a prototype is averagely better than OperaMini which is a commercial product.
- Standard deviations are comparable for all the competitors and they are quite small, so it means that there are homogenous evaluations by different users.

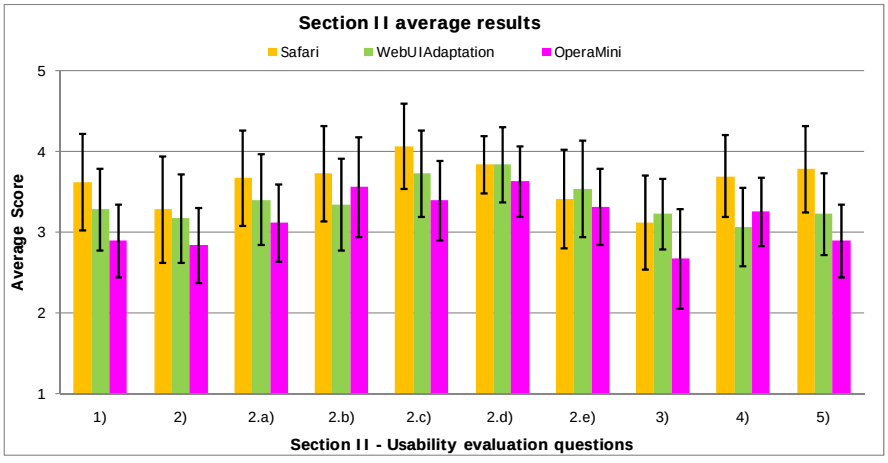


Figure 38 Adaptation evaluation satisfaction results considering Section II of the questionnaire

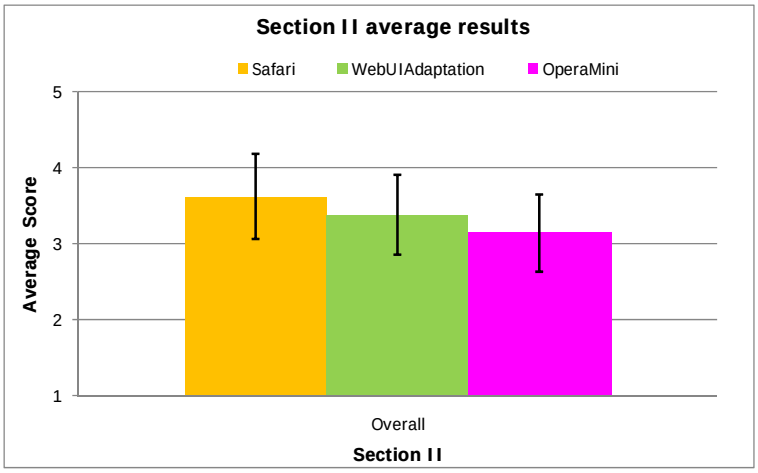


Figure 39 Adaptation evaluation satisfaction results considering Section II of the questionnaire, average results

- *Section III: Page Splitting support evaluation (*OPEN MSP *only)*. This third section asks to the user how the navigability of the split version of a web page is perceived. Considering that this feature is only implemented in OPEN MSP prototype the user is asked to answer only for this solution eventually writing down also some additional comments.
  In Figure 40 the average score obtained in Section III is reported in detail for each group of user and in yellow in the last bar the overall results is showed, too. Observing the plot it can be highlighted:
  - The average score for each group highlights how the page splitting is perceived as an excellent solution by some users while at the same time it is not accepted as a user experience improvement by some others. This is supported also by the standard deviation reported overlap to the average bar. Reading the user comments the aspects that not completely persuade the users are both technical (problem in visualization of some web site areas) and semantic because not always there is coherence among the page contents. Some other users consider not correct to split in more than one piece a single web site.
  - Typically, qualitative analyzing the data it is showed that users which are familiar with iPhone or mobile devices in which touch-based interaction is required perceive better the page splitting innovation giving a positive mark to the implemented solution.

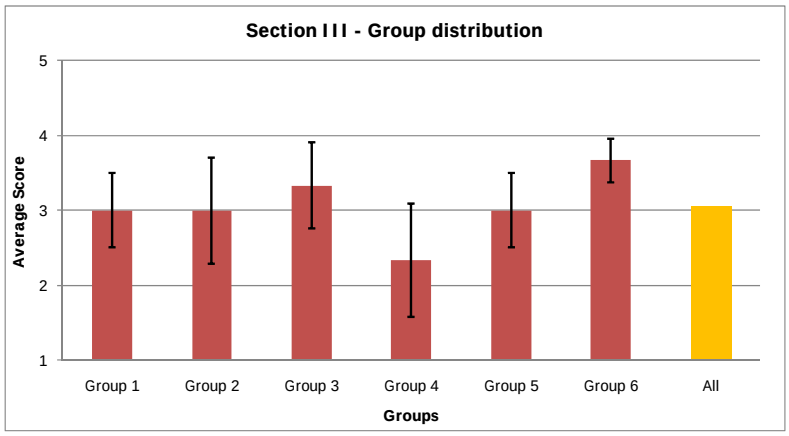


Figure 40 Adaptation evaluation satisfaction results considering Section II of the questionnaire, average satisfaction distribution among user groups

- *Section IV: Overall Strong and Weak points.* In this last section for each considered solution the user is asked to freely write down up to 3 strong and weak points. Respect to the other sections this is more qualitative and it is useful to understand what are the user free impressions without have to answer to a specific set of questions.
  Reading the comments (both for strong and weak points) reported by the user the following qualitative considerations can be done:
  - Users appreciate the page splitting and in general page reorganization like the one proposed by OperaMini even if they are a bit nostalgic of the classic web page which is presented on desktop PC. However they recognize that the visualization of the complete web page on mobile devices without adaptation produces a not so readable output.
  - The page splitting feature is useful but it is not so easy to understand when you are a newbie to it.
  - The form adaptation and changing elements from standard version to mobile is appreciated both in OPEN MSP and OperaMini even if sometimes it is not so good such that the form results dispersive and there are some problems with buttons labels, too.

2.7.7. USER INTERFACE ADAPTATION COMPARATIVE RESULTS ANALYSIS

This class of test focuses on the OPEN MSP module which implements interface adaptation features. Referring to the application classification described in Chapter 2 this module is representative of the Utility class.
It is important to remember that here it is performed a comparison test among the prototype developed inside the OPEN project and two commercial available products which are Safari and OperaMini. As it is detailed in the above data very good results about effectiveness and satisfaction are obtained. This is important because the OPEN MSP is still a prototype and it is very good that in this second testing iteration a strong positive result is obtained.

About the efficiency the test shows a not so high level about performances, but it is comprehensible that both the commercial products are preferred to the prototype from the performance point of view. Obviously they are optimized and delivered to be robust and competitive while the OPEN MSP module aim is, at current stage, to demonstrate the validity of its usability features. The overall results are good in term of usability for the OPEN MSP considering the good placement in respect to the other commercial competitors considered.

2.8. SATISFACTION – PRODUCT REACTION CARD RESULTS

In this section the results about the migration experience satisfaction evaluated using the Product Reaction Card (PRC) approach are reported. The PRC approach is more qualitative respect to the other evaluations, in fact it asks to the user to select a set of words that more represents the total and partial migration usability.

2.8.1. EMERGENCY

Results presented in this section have the main objective to demonstrate how the migration experience is perceived by the user. It is done according to the Product Reaction Card methodology previously described in Section 2.1.4.
Figure 41 shows all the cards picked at least one time during the first iteration when each user can select a free number of words. Then in Figure 42 the second iteration is considered where each user must pick a subset of exactly 5 cards from the ones chosen for the first iteration. In second iteration for each picked card it is provided a detailed explanation about the reasons that pushed the user to select the card. It is so possible to identify if such cards are picked because closely to Emergency application features instead of to the perceived migration experience. In this way the histogram in Figure 42 shows in a precise way how the migration experience is perceived showing exactly the words chosen by users. Please note that the most chosen card is “collaborative” which is a feature enabled to the Emergency application thanks to the OPEN Migration Service Platform. It is clear that without the platform the single application cannot provide a so high level of collaboration, considering migration aspects to the end user. Starting from the results reported in Figure 42 it is also possible to obtain a more synthetic result in term of positive and negative picked cards. This last proposed classification is possible also exploiting the comments that are requested to the user each time he picks a card. In this way the plot available in Figure 43 is obtained. From the plot it is clear how the overall evaluation about migration experience is positive (75%) even if there are some improvements to consider in order shifting to 0% the actual 25% of negative comments.

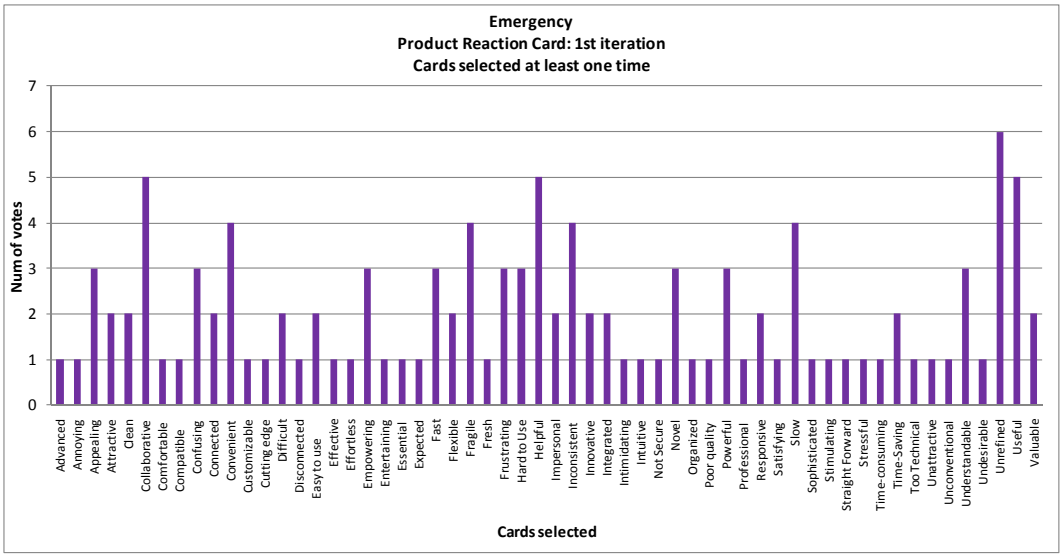


Figure 41 Emergency PRC first iteration results

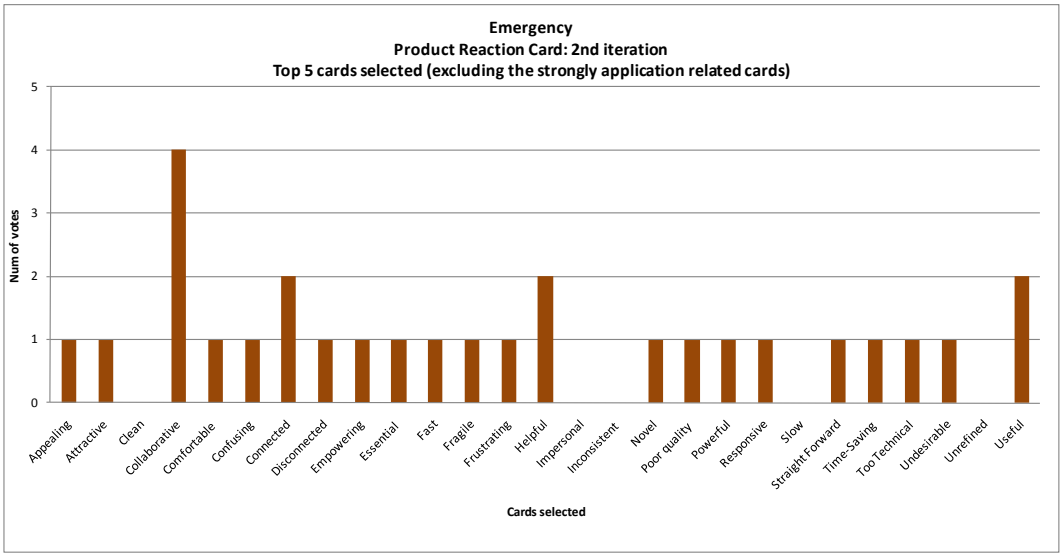


Figure 42 Emergency PRC second iteration results excluding strongly application related card

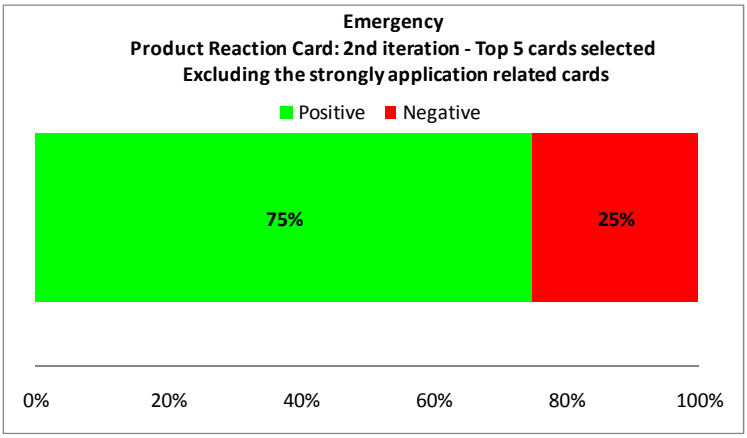


Figure 43 Emergency PRC second iteration results excluding strongly application related card. High level classification of picked cards according to the user comments

2.8.2. PARTIAL AND TOTAL MIGRATION

Results presented in this section have the main objective to demonstrate how the migration experience is perceived by the user. It is done according to the Product Reaction Card methodology previously described in Section 2.1.4.

Figure 44 reports all the cards picked at least one time during the first iteration when each user can select a free number of words. Then in Figure 45 the second iteration is considered where each user must pick a subset of exactly 5 cards from the ones chosen for the first iteration. In the second iteration for each picked card a detailed explanation is provided about the reasons that lead user to select the card. The reason helps understanding the specific part of the experience the user is referring to. The histogram in Figure 45 shows in a precise way how the migration experience is perceived showing exactly the words chosen by users. Please note that the most chosen card is “easy to use” and “flexible” which are features very important for the success of an innovative paradigm, as the migration one is. It is clear that without the platform the migration cannot be enabled to the end user from a simple Internet web page. Starting from the results reported in Figure 45 it is also possible to obtain a more synthetic result in term of positive and negative picked cards. This last proposed classification is possible also exploiting the comments that are requested to the user each time he picks a card. In this way the plot available in Figure 46 is obtained. From the plot it is clear how the overall evaluation about migration experience is positive (86%) even if there are some improvements to consider in order decreasing to 0% the actual 14% of negative comments. At the same time it is important to remind that can happen that some users do not accept the migration as a positive experience.

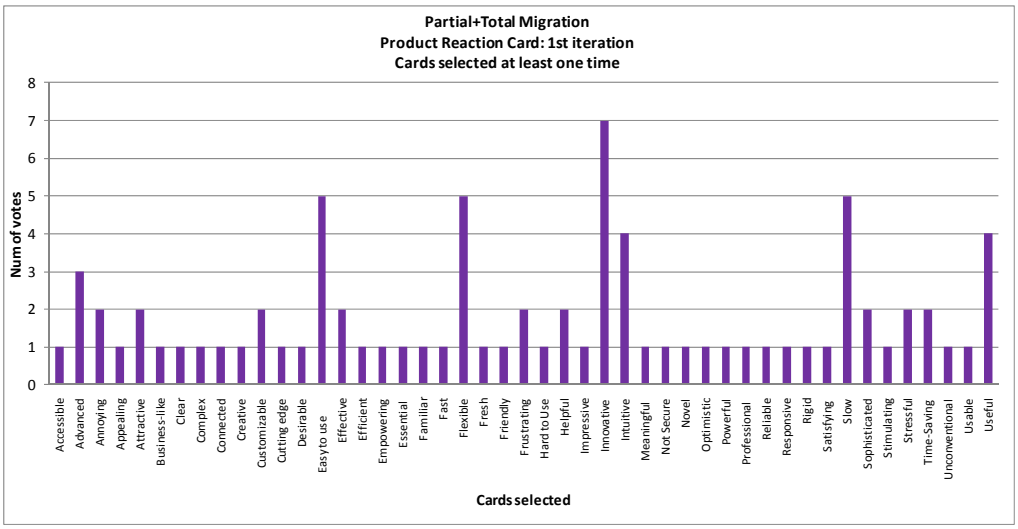


Figure 44 Social Game Partial and Web Pages Total PRC first iteration results

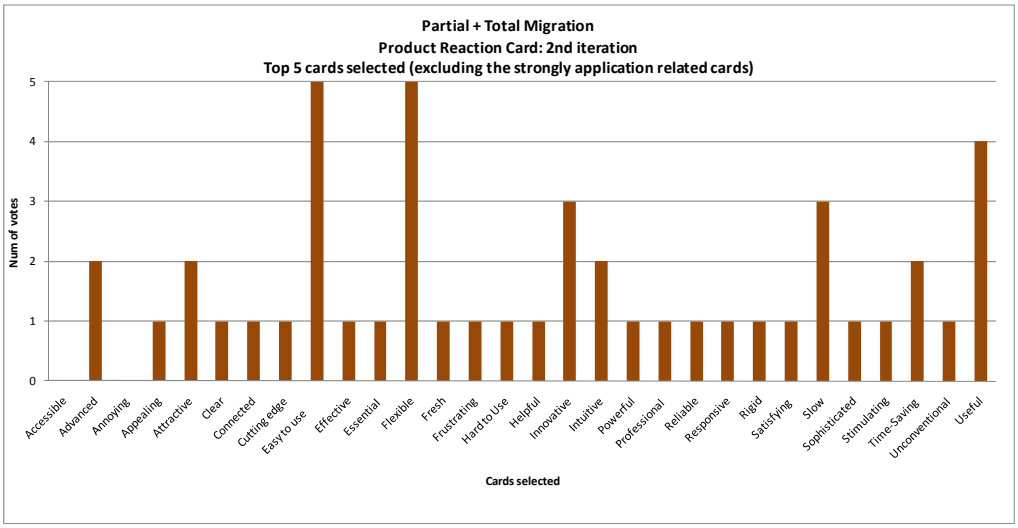


Figure 45 Partial and Total Web Migration PRC second iteration results excluding strongly application related card

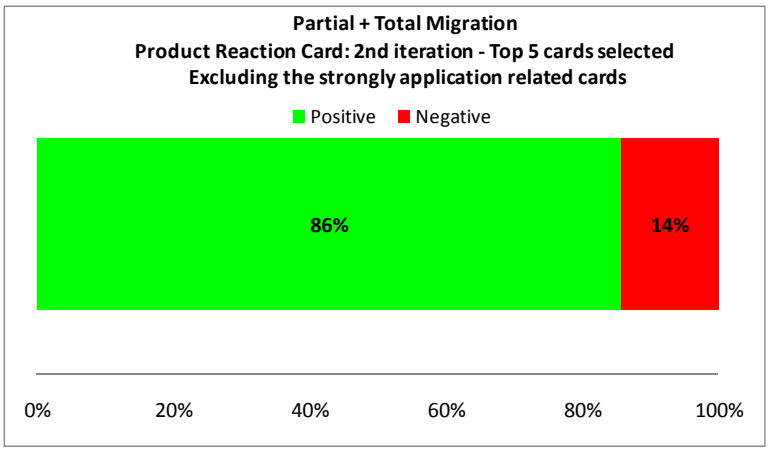


Figure 46 Partial and Total Web Migration PRC second iteration results excluding strongly application related cards. High level classification of picked cards according to the user comments

2.8.3. EVALUATION CONCLUSION

Observing the PRC results reported above it is clear how the overall results are good. It is highlighted by the fact that the migration experience is well perceived (positive comments greater than 75 % in both cases) by the majority of the involved users.

Analyzing the user comments and behaviors it is noted that Emergency management is an inherently complex activity so not trained people struggle to understand why it is necessary a so complex application to manage maps with emergency data. Consequently the users involved in these tests tend to give a worse evaluation to the migration experience with respect to the one that uses migration in an entertainment context (Social Game and Internet surfing). This last set of people are very excited and empowered by the migration experience which is perceived as a very innovative product.

3. PROGRAMMABILITY

In order to evaluate the migratory middleware programmability, it is necessary to focus on the platform implementation. The aim of this class of tests is to evaluate if the platform migratory features can be easily extended or configured without code implementation and recompilation. Unlike in the usability evaluation here the assessment is done using a developer/technical perspective rather than an end user one.

3.1. DEFINITION

Programmability is the capability within hardware and software to accept a new set of variables and instructions that alter its behavior [ENC-00]. In a middleware environment like the OPEN one the previous definition refers to variables and instructions:

- *Variables* refer to context information
- *Instructions* are rules used to describe the migration process behavior depending on the context information.

Both variables and instructions are the building blocks that can be used to implement specific behavior of the migration process, such as migration triggering (when to migrate). Moreover the Application Logic Reconfiguration and User Interface Adaptation can be managed configuring how to execute the adaptation.

The high level idea described above can be better illustrated using the following example. Suppose that a mobile phone that provides battery and signal strength information is available. In the platform, migration triggering rules depending on these variables are defined. Suppose that afterwards another mobile phone that can provide also the “user_location” (latitude and longitude) is available. The idea is now to make possible to the platform to perform the following actions:

- instruct the middleware in order to acquire the new “user_location” variable;
- define a new rule depending on this new variable. E.g.: if "user_location"="home" then trigger the migration towards the TV.

Please note that the aim of the programmability analysis is to verify if variables and rules can be correctly managed by the OPEN Migration Service Platform. This is done first considering single modules, each one with its programmability features and second having an overall point of view of the whole platform. In the current implementation a homogeneous tool for the whole platform configuration is not yet available but each module has its own procedure. It is important to consider that if the single module configuration works well, it is very simple to provide a unique tool for platform programmability, e.g. building a unique graphical user interface able to easily create and modify platform configurations and translate them into proper modules configuration files. The strategy followed by the consortium addressed all the effort on programmability features making modules more customizable without needs of code changing.

3.2. PROGRAMMABILITY METHODOLOGY

The final goal of this second programmability evaluation phase is to obtain a classification of the programmability features considering two dimensions. They are:

- *Extensibility*. A middleware module is *extensible* if and only if it allows adding and removing rules and/or variables able to control its behavior without changing the module source code.
- *Configurability* refers to the capability of a module to allow changing already defined rules and variables in order to modify how the system answers to different situations. Also in this case all the configurations should be possible without changing the module source code.

According to the dimensions defined above it is possible to have a module and consequently a platform classification, which can be easily summarized using the plot reported in Figure 47.

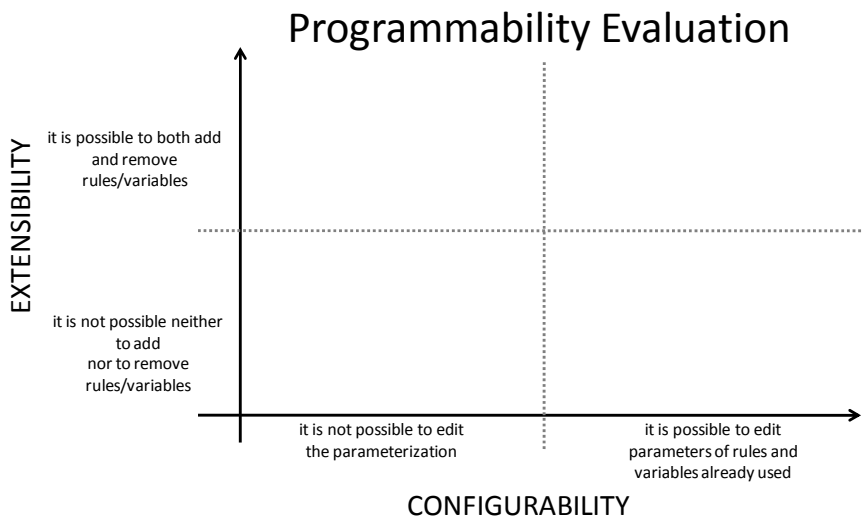


Figure 47 Evaluation chart for programmability test

Fixing the expected results of the process it is important to understand how these evaluations can be performed. The first step consists of selecting which module to analyze and execute a programmability assessment [D6.4]. In this way it is possible to understand if a further evaluation does make sense or if the module under test does not implement any programmability features. If some programmability aspects are supported the next step consists of verifying how well these features work according to the two dimensions described above.
The assessment starts from the module documentation identifying if configuration facilities are available and what can be configured and eventually extended. The second step considers how well the extensions and configurations that are available according to the OPEN MSP implementation (this is the validation phase). In this case two pass/fail dimensions are considered which are:

- *Robustness.* It means that if the configuration files contain some errors these are managed by the platform. For example new behaviors are not enabled, but the entire platform does not crash, but continues to be available.
- *Consistency.* It checks that the edited configurations effectively change the platform behavior adding the features that are specified by the varied files.

For the robustness and consistency validation both available prototypes and ad-hoc testing application have been used.
In Figure 48 a graphical representation of the proposed classification is reported. It is clear that the yes/no approach proposed here is a bit coarse grained, but it is important to remember that this is the final evaluation phase.

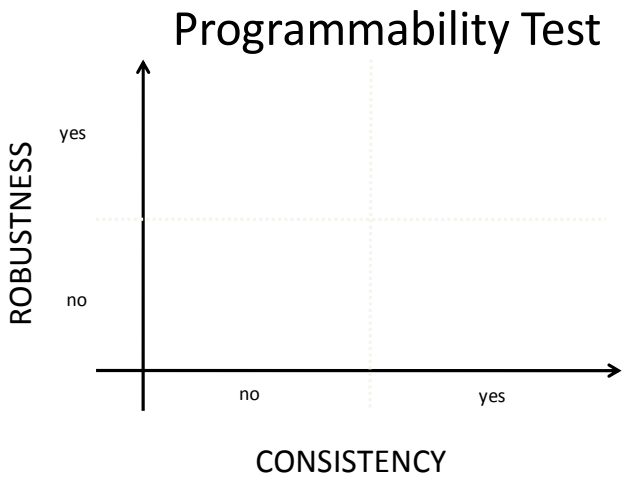


Figure 48 Evaluation chart for programmability test, second level evaluation

The proposed dimension evaluation was structured as follows:

- Robustness is evaluated modifying with no valid parameters or deleting the configuration and check if the platform notifies the configuration error continuing to work without denial of service.
- Consistency is evaluated checking that the configured behavior is the one exploited by the platform. This can be done using prototypes to build scenarios in which the behaviors that stimulate edited configurations are recreated.

3.3. MODULE ANALYSIS

3.3.1. CONTEXT MANAGEMENT FRAMEWORK (CMF)

The Context Management Framework (CMF) is the OPEN MSP module that manages the available data in the environment in which the OPEN Migration Service Platform is running. It is mainly composed by a central Context Management Network (CMN) and a set of Context Agents (CA) that can be distributed and communicate with the CMN through the network.
It is useful because it allows collecting environment information in a very simple way. It is important to point out that CMF provides the infrastructure to collect information independently from data types. The underlying idea is that environment data need to be communicated from where they are produced to where they are required. How data are produced, measured and modeled inside the platform depends on the information themselves so a programmability test

for the CMF needs to evaluate the infrastructure independently from the types of information used.

During the first test iteration the programmability assessment identifies the CMF as a programmable module considering the possibility to add variables and to configure how the CMF itself can obtain information from the environment. So, the improvement proposals were focused on the use of the OSGi technology [OSG-10] that makes possible to add/remove a retriever without the restart of the complete CMF module enhancing the module flexibility. The last OPEN MSP implementation follows the previously exported recommendations in fact the CMF is implemented through OSGi methodology using Equinox which is a certified implementation of the OSGi R4 core framework. Thanks to the OSGi architecture model it is possible to add, remove, start, stop and update single retrievers without restarting the entire CMF module.

It is important to remember that the CMF is composed by *retrievers* which are the modules which collect information from sensors and transducers in order to make them available to the applications which are interfaced to the CMF. Each *retriever* manages a single piece of information like for example battery level, environment temperature, GPS positions, etc. Each of these pieces of information is so considered a variable which is managed by CMF. So, adding a variable means in practical to add a retriever able to manage a new piece of information which was not previously considered.

On the other hand the CMF can be configured on the basis of which method is used to obtain the information from the retriever. Fundamentally three possibilities are available as they are described in detail in Section 4.7.1:

1. Reactive: it is based on a request/response mechanism.
2. Proactive event driven update: retriever sends the data to context agent as soon as the data is available.
3. Proactive periodic update: retriever sends the data to context agent periodically with fixed amount of time window.

About configurability can be configured which one of the available strategy has to be used for the information retrieval.

According to the features described thanks to the OSGi architecture the CMF module can be classified as extensible and configurable. Here after in details:

- **Extensible** because it can be enabled to mange new variables simply creating the retriever module and once it is packed in a jar file it can be added and directly started to a running CMF instance without any module restarting. The module restart is not necessary also when some changes are implemented in the retriever, because it can simply update to the new version through the update functionality of the OSGi framework.
- **Configurable** because it is always possible to edit the method in which the information are retrieved through the XML file that contains the queries subscription specifications. At the same time the CMF module through another XML file available in Figure 51 can be configured to be used on different network nodes with custom ports.

The testing activity has been carried out thanks to an ad-hoc retriever able to generate data (the data generation is configurable). The configuration can be done with the testPUConfig.xml file, for the following test it is set to be an incremental counter that once started increments its value of one unit each 5 seconds. Since the retriever is developed following the OSGi standard it can be

installed and started in the CMF once the CMF is already running. So once the CMF is running writing the command reported in Figure 49 in the CMF OSGi console it is possible to extend the CMF capabilities starting a new retriever. In Figure 50 some other OSGi command to manage bundles are reported.

```
# first install the new bundle in the OSGi framework passing the jar bundle URL
install URL

# ss returns a table with the a pair <#identifier, bundleName>
ss

# to start the new installed bundle
start #identifier
```

Figure 49 List of command to install and start a new retriever into the CMF

```
# to start the new installed bundle
start #identifier

# to stop an installed and running bundle
stop #identifier

# to update an installed bundle
update #identifier
```

Figure 50 Other useful commands to work with CMF bundle

```
<test>
    <contextAgent>
        <hostName>localhost</hostName>
        <port>8035</port>
        <registryName>ContextAgent</registryName>
    </contextAgent>
    <testProgram>
        <hostName>localhost</hostName>
        <port>8036</port>
        <registryName>ContextAgent</registryName>
    </testProgram>
    <testConfiguration>
        <testName>result_QoSv2</testName>
        <noOfDataElements>3</noOfDataElements>
        <timeBetweenRequests>1000</timeBetweenRequests>
        <outputdir>/home/labsmart/OPEN_validation_test/testOutput</outputdir>
    </testConfiguration>
</test>
```

Figure 51 CMF configuration file

In order to execute some experiments with the new CMF a simple program that interface itself with CMF is implemented and applied. It is referenced as testProgramme. It is quite simple but useful because it sends to the CMF the configuration about how the information are being available by the CMF to program (reactive vs proactive methodologies) and eventually send the CALA query waiting for the corresponding CMF results.

About **robustness** some malformed XML configuration files are considered and the objective is to evaluate how the CMF reacts, if throwing correct exceptions signaling to the user the problem or not. In order to verify these aspects three different possible scenarios are considered. First of all a robustness test is done trying to use malformed XML files verifying if it is noticed to the user.

Then a wrong configuration of the CMF parameters is considered in order to verify how it is managed by the module. Finally a CALA query that requires not existent data is requested to the CMF and the corresponding output is analyzed. All the described scenarios results passed and they are reported in detail following in Table 6, Table 7 and Table 8.

Robustness test 1	
Status	Passed
Description	Malformed XML with wrong tags.
Expected output	Error notification.
Configuration File Edited testconfig.xml	<test> <contextAgent> <hostName1234>localhost</hostName> <port>8035</port> <registryName>ContextAgent</registryName> </contextAgent> <testProgram> <hostName1234>localhost</hostName> <port>8036</port> <registryName>ContextAgent</registryName> </testProgram> <testConfiguration> <testName>result_QoSv2</testName> <noOfDataElements>3</noOfDataElements> <timeBetweenRequests>1000</timeBetweenRequests> <outputdir>/home/labsmart/OPEN_validation_test/testOutput</outputdir> </testConfiguration> </test>
Output	Starting request method test org.xmlpull.v1.XmlPullParserException: expected: /hostName1234 read: hostName (position:END_TAG </hostname>)

Table 6 Robustness test 1 technical description

Robustness test 2	
Status	Passed
Description	Malformed CMF configuration XML file with fake configuration data. The CMF parameterization is wrong because the CMF hostname is not existent.
Expected output	Error notification.
Configuration File Edited testconfig.xml	<test> <contextAgent> <hostName>fakehost</hostName> <port>8035</port> <registryName>ContextAgent</registryName> </contextAgent> <testProgram> <hostName>fakehost</hostName> <port>8036</port> <registryName>ContextAgent</registryName> </testProgram> <testConfiguration> <testName>result_QoSv2</testName> <noOfDataElements>3</noOfDataElements> <timeBetweenRequests>1000</timeBetweenRequests>

	<outputdir>/home/labsmart/OPEN_validation_test/testOutput</outputdir> </testConfiguration> </test>
Output	Test program initialized correctly Result is null. IOException

Table 7 Robustness test 2 technical description

Robustness test 3	
Status	Passed
Description	Malformed CMF CALA query XML file. In this case a variable not considered by the CMF is requested with a CALA query.
Expected output	An empty result or an error message.
Configuration File Edited testconfig.xml	<?xml version="1.0" encoding="UTF-8"?> <cala xmlns="http://cmf.neclab.eu/CMF" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://cmf.neclab.eu/CMF ../schemas/CMF.xsd"> <query> <entityIdentifierSelector> <hasIdentifier>**testVarIdFake**</hasIdentifier> <entityType>**testVarTypeFake**</entityType> <attributeName>**testVarFake**</attributeName> </entityIdentifierSelector> <scope> <networkScope>NODE</networkScope> </scope> </query> </cala>
Output	<?xml version='1.0' encoding='UTF-8' ?> <cala xmlns="http://cmf.neclab.eu/CMF" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"> <queryResponse> <entities /> </queryResponse> </cala>

Table 8 Robustness test 3 technical description

About **consistency** three different configurations about information retrieval are verified. The aim of all tests is to verify two things:

- the obtained information (the counter value) is effectively the last one generated by the context generator;
- the information are obtained using the reactive or proactive mechanism according to the one chosen by the CALA XML configuration file send to the CMF through the testProgramme.

Following in Table 9, Table 10 and Table 11 the three considered CALA queries are completely reported. Please note that simply changing the CALA query set up it is possible to interact with the CMF indicating which information retrieval method is used. All the tests performed give successful results, because the expected value of the testVar is always correctly returned by the CMF independently from which is the retrieval methodology applied.

Consistency test 1 [Reactive]	
Status	Passed
Description	Execute a request to the CMF in order to retrieve the counter value, using a reactive approach (reactive CALA query).
Expected Output	testVar=5 which is the actual counter value
Configuration File Edited	<?xml version="1.0" encoding="UTF-8"?> <cala xmlns="http://cmf.neclab.eu/CMF" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://cmf.neclab.eu/CMF ../schemas/CMF.xsd"> <query> <entityIdentifierSelector> <hasIdentifier>testVarId</hasIdentifier> <entityType>testVarType</entityType> <attributeName>testVar</attributeName> </entityIdentifierSelector> <scope> <networkScope>NODE</networkScope> </scope> </query> </cala>
Output	<?xml version='1.0' encoding='UTF-8' ?> <cala xmlns="http://cmf.neclab.eu/CMF" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"> <queryResponse> <entities> <entity> <hasIdentifier>testVarId</hasIdentifier> <entityType>testVarType</entityType> <attribute> <name>testVar</name> <type>int</type> <value> <integer>5</integer> </value> <metadata> <name>attributeValueIdentifier</name> <type>string</type> <value> <string>1276762427260</string> </value> </metadata> </attribute> </entity> </entities> </queryResponse> </cala>

Table 9 CALA query for reactive information retrieval, file requestTestVar.xml

Consistency test 2 [Proactive Event Driven]	
Status	Passed
Description	Execute a request to the CMF in order to retrieve the counter value, using a proactive event driven approach (subscription CALA query).
Expected Output	testVar=157 which is the actual counter value
	<?xml version="1.0" encoding="UTF-8"?>

Configuration File Edited	<cala xmlns="http://cmf.neclab.eu/CMF" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://cmf.neclab.eu/CMF ../schemas/CMF.xsd"> <subscription> <entityIdentifierSelector> <hasIdentifier>testVarId</hasIdentifier> <entityType>testVarType</entityType> <attributeName>testVar</attributeName> </entityIdentifierSelector> <onChangeSubscriptionCondition> <changeType>all</changeType> <attributeName>testVar</attributeName> </onChangeSubscriptionCondition> <scope> <networkScope>NODE</networkScope> </scope> </subscription> </cala>
Output	<?xml version='1.0' encoding='UTF-8' ?> <cala xmlns="http://cmf.neclab.eu/CMF" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"> <notification> <globalSubscriptionId> <subscriptionId>2</subscriptionId> <nodeIdentifier>http://127.0.0.1:8002</nodeIdentifier> </globalSubscriptionId> <entities> <entity> <hasIdentifier>testVarId</hasIdentifier> <entityType>testVarType</entityType> <attribute> <name>testVar</name> <type>int</type> <value> <integer>157</integer> </value> <metadata> <name>attributeValueIdentifier</name> <type>string</type> <value> <string>1276767581873</string> </value> </metadata> </attribute> </entity> </entities> <options /> <sourceNodeIdentifier>http://127.0.0.1:8002</sourceNodeIdentifier> <destinationNodeIdentifier>http://127.0.0.1:8002</destinationNodeIdentifier> </notification> </cala>

Table 10 CALA query for reactive information retrieval, file subscribeTestVarEvent.xml

Consistency test 3 [Proactive Time Driven]	
Status	Passed
Description	Execute a request to the CMF in order to retrieve the counter value, using a proactive time driven approach (subscription CALA query).
Expected Output	testVar=206 which is the actual counter value

Configuration File Edited	<?xml version="1.0" encoding="UTF-8"?> <cala xmlns="http://cmf.neclab.eu/CMF" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://cmf.neclab.eu/CMF ../schemas/CMF.xsd"> <subscription> <entityIdentifierSelector> <hasIdentifier>testVarId</hasIdentifier> <entityType>testVarType</entityType> <attributeName>testVar</attributeName> </entityIdentifierSelector> <periodicalSubscriptionCondition> <period>1</period> </periodicalSubscriptionCondition> <scope> <networkScope>NODE</networkScope> </scope> </subscription> </cala>
Output	<?xml version='1.0' encoding='UTF-8' ?> <cala xmlns="http://cmf.neclab.eu/CMF" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"> <notification> <globalSubscriptionId> <subscriptionId>6</subscriptionId> <nodeIdentifier>http://127.0.0.1:8002</nodeIdentifier> </globalSubscriptionId> <entities> <entity> <hasIdentifier>testVarId</hasIdentifier> <entityType>testVarType</entityType> <attribute> <name>testVar</name> <type>int</type> <value> <integer>206</integer> </value> <metadata> <name>attributeValueIdentifier</name> <type>string</type> <value> <string>1276762427260</string> </value> </metadata> </attribute> </entity> </entities> <options /> <sourceNodeIdentifier>http://127.0.0.1:8002</sourceNodeIdentifier> <destinationNodeIdentifier>http://127.0.0.1:8002</destinationNodeIdentifier> </notification> </cala>

Table 11 CALA query for reactive information retrieval, file subscribeTestVarPeriodic.xml

Coherently with the assessment results the CMF is extensible and configurable. Another strong point is about the OSGi implementation that makes the module very scalable and easy to use and configure. The improvements have not impacted both the strong consistency and well robustness of the module that are confirmed like showed in the first test iteration by the current executed tests.

3.3.2. TRIGGER MANAGER (TM)

Trigger Management is the OPEN MSP module devoted to the migration triggering. As written in D6.4 [D6.4] this is a module that is interesting to make configurable and some suggestions have been already given in that deliverable. Actually the TM is developed in a well structured way but configurability is not yet supported. The fact is that the actual TM configurations are hard-coded and do not consider any external configuration file. The reason is that all the development efforts about this module were addressed on the module integration inside the OPEN Migration Service Platform rather than on its configurability features. Considering the above motivations the module does not result testable from the programmability point of view and it is not considered in this programmability evaluation step.

3.3.3. APPLICATION LOGIC RECONFIGURATION (ALR)

Application Logic Reconfiguration is the OPEN MSP module devoted to the change of the application logic: it modifies the application behavior changing how the application takes decisions and actuates them.

During the first test iteration the programmability assessment identified the ALR as a programmable module considering both application configuration and rewiring rules setting. The main problem in the previous version was that the only way to change the application configuration was to edit the Java code of the application itself. So, the improvement proposals were focused on the possibility to change the application configuration in an easier way for example through an external file.

The last OPEN MSP implementation provides an XML configuration file to set up how the application module can be configured and rewired. This is perfectly compliant with the suggestions proposed by the first iteration tests.

For each application component the corresponding XML component descriptor file is produced in order to describe the input and output interfaces required and provided by the considered application module. Just to give an example in Figure 52 and Figure 53 two configuration files are showed. Note that they describe in detail which is the component name and its interfaces, specifying the corresponding type as showed in Figure 54 using the standard UML component representation. Another file for the application configuration is required in order to specify to the reconfiguration module how the available application component can be wired. A configuration file example for an example application is reported in Figure 55. Please note that the application modules declarations with their parameters, constraints and wiring rules are contained in the file.

```
<?xml version="1.0" encoding="UTF-8"?>

<component name="ShootingLine">
    <provides type="de.cadaisi.repository.service.ShootingLineIf" />
</component>
```

Figure 52 XML component descriptor for the Shooting Line component

```
<?xml version="1.0" encoding="UTF-8"?>

<component name="BiathlonAthlete">
    <provides type="de.cadaisi.repository.service.PositionIf" />
```

```
    <provides type="de.cadaisi.repository.service.AthleteIf" />

    <requires type="de.cadaisi.repository.service.ShootingLineIf" />
</component>
```

Figure 53 XML component descriptor for the BiathlonAthlete component

Figure 54 Application component UML standard representation

```
<?xml version="1.0" encoding="UTF-8"?>

<application name="BiathlonShootingAssistent">
    <rootTemplate name="Athlete" minCardinality="1">

        <provides type="de.cadaisi.repository.service.AthleteIf" />
        <provides type="de.cadaisi.repository.service.PositionIf">
            <constraint method="de.cadaisi.repository.constraint.SkiingConstraints.isAtShootingPlace">
                <param type="java.lang.Boolean" value="$Trainer.PositionIf.isAtShootingPlace" />
            </constraint>
        </provides>

        <requires type="de.cadaisi.repository.service.ShootingLineIf" name="b" cardinality="1" target="ShootingLine">
            <constraint method="de.cadaisi.repository.constraint.SkiingConstraints.sameShootingLine">
                <param type="java.lang.String" value="$target:getShootingLine" />
                <param type="java.lang.String" value="$source:getShootingLine" />
            </constraint>
        </requires>
    </rootTemplate>

    <template name="ShootingLine">
        <provides type="de.cadaisi.repository.service.ShootingLineIf" />
    </template>
</application>
```

Figure 55 XML configuration file for the Biathlon Shooting Assistent example

According to the features provided, using the XML configuration file the ALR module can be classified as extensible and configurable. Here after in details:

- **Extensible** because once the new application modules to consider are available they can be plugged to the logic simply adding them with their wiring constraints to the XML configuration files.
- **Configurable** because you can always change the constraints configuration in the XML files on the basis of requirements.

At this point terminated the assessment phase it is possible to evaluate the robustness and consistency of the module programmability. To perform the test the ALR within the biathlon application prototype are used. The biathlon prototype simulates a biathlon race course with its shooting lines and biathletes as depicted in Figure 56. The biathletes are moving around the race course and are carrying a device which measures its current position and for instance the pulse rate of the biathlete. In the device an application component is running to perform these tasks.

The shooting lines have also a device which receives the pulse rate of the biathlete. For this also an application component is installed at every shooting line device. The component of a shooting line is wired with the component of the biathlete, if the biathlete is at the shooting line. The wiring is done by the ALR.

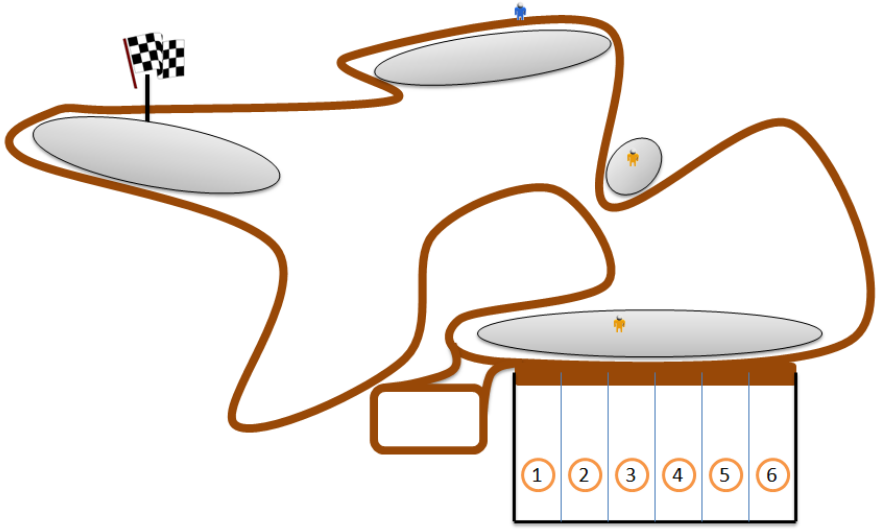


Figure 56 Biathlete racing course simulated by Siafu

About **robustness** some malformed XML configuration files are considered and the objective is to evaluate how the ALR reacts, if throwing exceptions or signaling to the user the problem or not. In order to verify robustness two cases are considered. The first, technically described in Table 12 consider a malformed configuration file from the XML tag point of view. Starting the biathlon application it cannot parse correctly the file, so an exception is thrown and displayed to the user since it is not caught. The problem is that the output is very generic being "NullPointerException" so it is not so simple to identify which is the root of the problem.

Robustness test 1	
Status	Passed
Description	Malformed XML with wrong tags.
Expected output	Error notification.
Configuration File Edited	... <provides**12345** type="de.cadaisi.repository.service.PositionIf"> <constraint method="de.cadaisi.repository.constraint.SkiingConstraints.isAtShootingPlace"> <param type="java.lang.Boolean" value="$Trainer.PositionIf.isAtShootingPlace" /> </constraint> </provides> ...
Output	NullPointerException

Table 12 Robustness test 1 technical description

The second test performed considers the insertion of a fake component considering also fake constraint and parameters. The underlying idea is to check if the ALR module verifies the correctness of the XML configuration file checking that the declared components are effectively

available for a possible instantiation during the re-configuration process. As reported in Table 13 this check is not performed. Even if the programmability is not strongly affected by this behavior as a suggestion this feature could be implemented in a long term view.

Robustness test 2	
Status	Not Passed
Description	XML with “provides” and “constrains” mapped on fake application modules.
Expected output	Error notification.
Configuration File Edited	... <provides type="**fakeInterface1**"> <constraint method="**fakeConstraint1**"> <param type="**fakeType**" value="$**1234**" /> </constraint> </provides> ...
Output	None notification message.

Table 13 Robustness test 2 technical description

About **consistency** two different configurations considering different constraints are verified. Then once they are applied to the biathlon simulator it is verified that constraints are effectively considered during the simulation according to the declaration in the XML configuration file. The complete and detailed descriptions reporting the entire XML files used are reported in Table 14 and Table 15.

Consistency test 1 [no constraints]	
Status	Passed
Description	Application XML configuration files without shooting line constraints.
Expected Output	The user can shoot in every point of the path without the constraint to shoot if and only if he/she is in a shooting line.
Configuration File Edited	<?xml version="1.0" encoding="UTF-8"?> <application name="BiathlonShootingAssistent"> <rootTemplate name="Athlete" minCardinality="1"> <provides type="de.cadaisi.repository.service.AthleteIf" /> <provides type="de.cadaisi.repository.service.PositionIf" /> <requires type="de.cadaisi.repository.service.ShootingLineIf" name="b" cardinality="1" target="ShootingLine" /> </rootTemplate> <template name="ShootingLine"> <provides type="de.cadaisi.repository.service.ShootingLineIf" /> </template> </application>

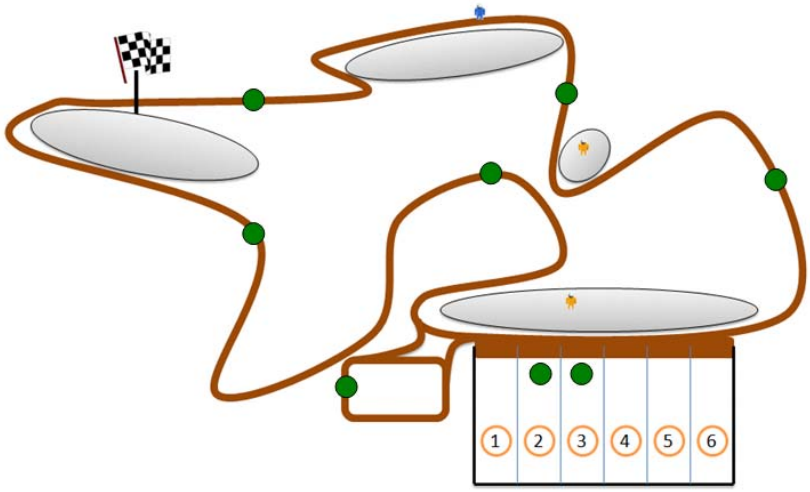
Output	

Table 14 ALR programmability first consistency test description

Consistency test 2 [constraints]	
Status	Passed
Description	Application XML configuration files with shooting line constraints.
Expected Output	The user can shoot if and only if he/she is in a shooting line.
Configuration File Edited	<?xml version="1.0" encoding="UTF-8"?> <application name="BiathlonShootingAssistent"> <rootTemplate name="Athlete" minCardinality="1"> <provides type="de.cadaisi.repository.service.AthleteIf" /> <provides type="de.cadaisi.repository.service.PositionIf"> <constraint method="de.cadaisi.repository.constraint.SkiingConstraints.isAtShootingPlace"> <param type="java.lang.Boolean" value="$Trainer.PositionIf.isAtShootingPlace" /> </constraint> </provides> <requires type="de.cadaisi.repository.service.ShootingLineIf" name="b" cardinality="1" target="ShootingLine"> <constraint method="de.cadaisi.repository.constraint.SkiingConstraints.sameShootingLine"> <param type="java.lang.String" value="$target:getShootingLine" /> <param type="java.lang.String" value="$source:getShootingLine" /> </constraint> </requires> </rootTemplate> <template name="ShootingLine"> <provides type="de.cadaisi.repository.service.ShootingLineIf" /> </template>

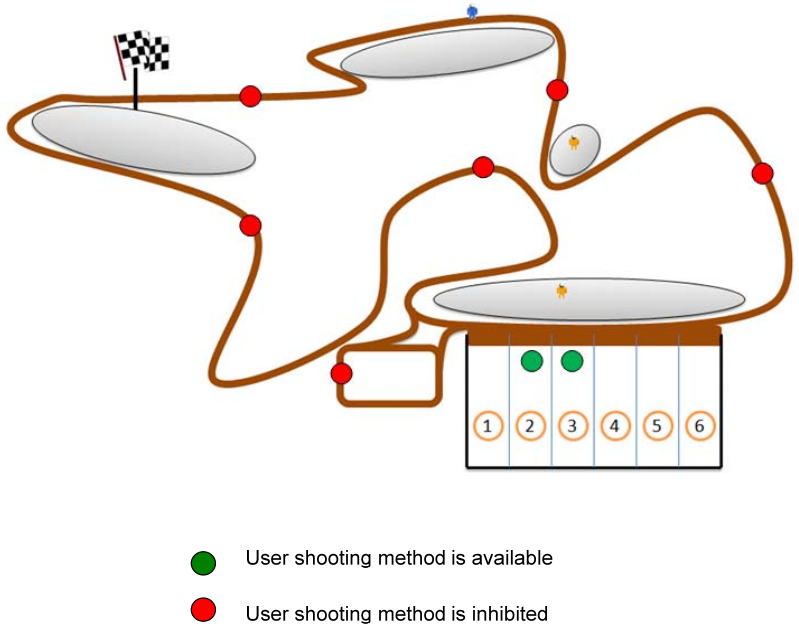
	</application>
Output	

Table 15 ALR programmability second consistency test description

At the end considering the ALR module it is confirmed how it results both configurable and extensible as declared in the documentation and reported during the assessment of the module at the beginning of the programmability evaluation. About robustness the module results are quite good even if it is suggested to improve the exception explanation in order to make them better comprehensible by the user. Another thing to remember is to restart the ALR once the XML configuration file has to be edited. From the consistency point of view the module results are good because the behaviors declared during the configuration phase results effectively reproduced during the application execution. This is a good result because the suggestions exported by the first test iteration are now implemented and working.

3.3.4. WEBUIADAPTATION

Analyzing the available documentation for the WebUIAdaptation module it is possible to classify this module as not extensible but highly configurable. Here after the considered classification:

- **Not Extensible** because the parameters are internally fixed by the module and if you want to consider new parameters you have to necessarily add them hard coding their management rules and recompiling the entire module code, so that the module is not extensible.
- **Configurable** because it is possible to configure a large number of parameters for the desktop to mobile adaptation, selecting which value they have to assume in a predefined range. The underlying concept to the configuration capabilities for this user

interface adapter is to provide to users more control on the adaptation process in order to improve its usability.

In this context more control can mean various things. One important aspect is control on the rules that drive adaptation to the various platforms (the most common case is desktop-to-mobile adaptation). For example, the adaptation engine is able to split the desktop pages when they require considerable amount of interaction resources but some users may like to have more control on the splitting algorithm.

In Figure 57 the web based user interface that allows end users to configure the adaptation process is reported. From the figure it is clear how the various parameters are grouped according to the user interface aspect considered. For the fonts, it is possible to specify the minimum and maximum font in the target device and the associated measure unit, which is the default font with its family. For the radio buttons it is possible to indicate whether they should be transformed into an interaction that supports the same semantics but with using less space screen. In this case, it is possible to specify the threshold, in terms of number of choice options, which should trigger the transformation and the type of interaction to use for its replacement. Similar parameters are available for the list boxes. Other parameters concern the maximum number of characters for a text, maximum and minimum dimensions for images. Finally the page splitting options are reported giving the possibility to the user to enable or disable the page splitting features.

All these parameters determine the cost of rendering a presentation. This cost is compared with the overall sustainable cost in the target device, which is given by the screen resolution multiplied by horizontal and vertical tolerance. The higher the tolerance coefficient values are, the more scrollable the generated user interface will be. This means that end users have the possibility to specify to what extent the adapted content will be scrollable in the target device. The table tolerance provides an additional factor to consider when calculating the sustainable cost. In practice, this means that when there are tables more scrolling will be acceptable before deciding to split the presentation. The customization interface also allows the user to indicate what type of scrolling (horizontal or vertical) to avoid has the priority and the algorithm splitting version to apply. Through the considered interface it is possible to configure and store the specification of user preferences regarding web user interface adaptation.

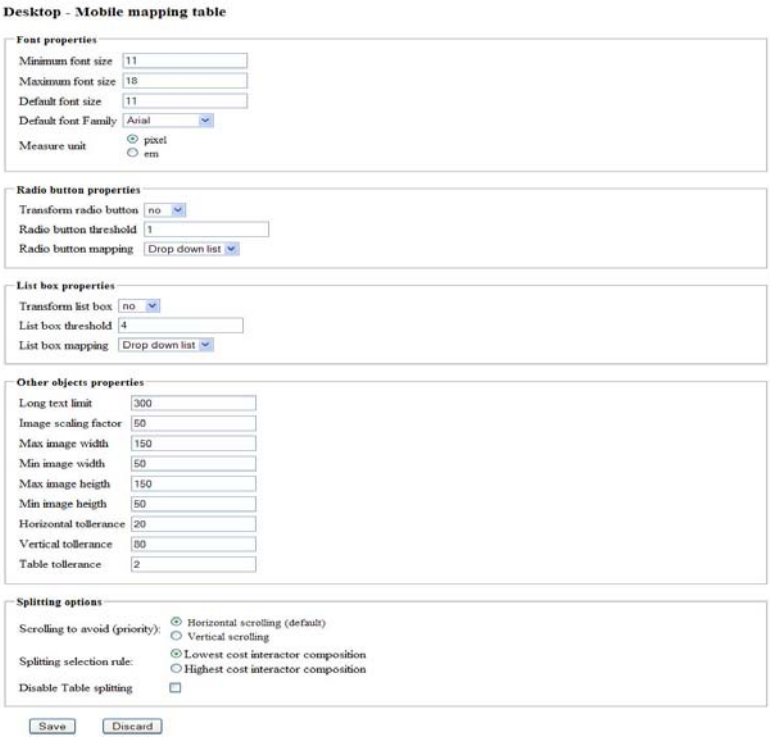


Figure 57 WebUIAdaptation programmability: Desktop to Mobile configuration table

As previously described in the programmability methodology after the theoretical evaluation carried out on the basis of the available documentation it is now possible to test how much the interface configurable features are robust and consistent.

About the **robustness** a GUI is available to the user, so data validation is intrinsically checked by the user before saving the configuration. Note that using a GUI for some parameters means that the configuration simply consists of choosing what value to assign among the available values. Thanks to the web interface the user has no visibility and access to any configuration files and this is good to ensure that WebUIAdaptation module does not export files which are potentially vulnerable.

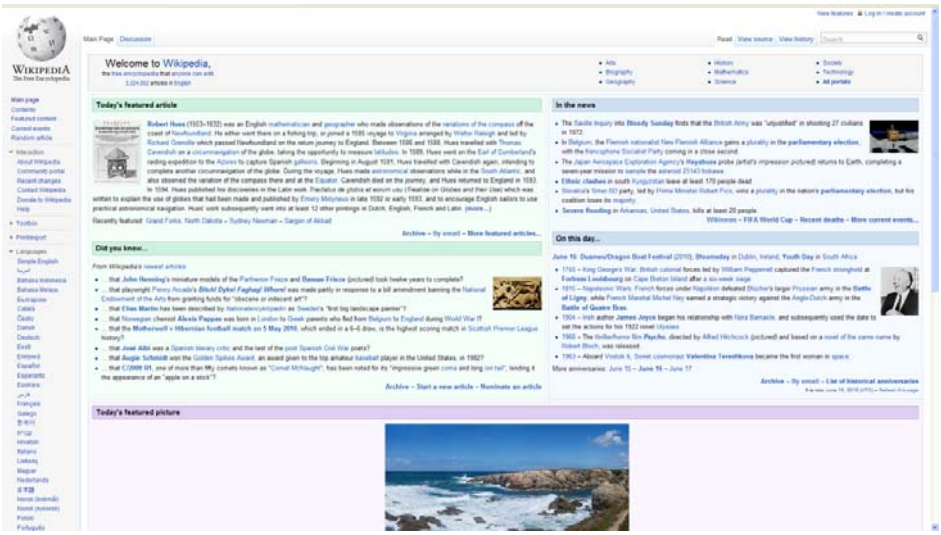


Figure 58 Desktop view of the considered Wikipedia web page

About the **consistency** here after an example is reported to proof how the behavior declared in the configuration GUI is effectively realized by the module. An example desktop web application is considered for the test. Deliberately it is used a well known example so that it is possible to understand the consistency of the setting in the page provided as output from the adaptation module. The desktop look and feel of the considered Wikipedia home page is reported in Figure 58. Now through the web configuration tool two different adaptation configurations are considered and the adapted version of the Wikipedia home page is evaluated. All the screenshots related to the test results are reported in Figure 59 and Figure 60. In each of these two Figures the adaptation configuration is showed on the left in the (a) part, while on the right there are the adapted pages rendered in the Safari browser of an iPod connected to the network via Wi-Fi.
Comparing the first and the second configuration it is clear that the scrolling preferences in the page splitting section changes. In particular:

- In Figure 59 (a) it is requested to avoid the Horizontal Scrolling, and the vertical tolerance in the Other Object properties is fixed to 80. Coherently as you can see in Figure 59 (b, c, d) the Wikipedia home page is split in three different web pages, the main one (b) that links to the bodies which are two: an upper one (c) and a lower one (d). This splitting is mainly caused by the small vertical tolerance fixed during the configuration phase, so the original page results spread on three adapted pages which are navigable using the horizontal scroll.
- In Figure 60 (a) it is requested to avoid the Vertical Scrolling, and the vertical tolerance in the Other Object properties is fixed to 500. In Figure 60 (b, c) it is showed the Wikipedia home page split in only two different web pages, the main one (b) that links to the body (c). In this case the adapted version has fewer pages which are longer considering the vertical tolerance in the other object properties with respect to the one reported in Figure 59. This behavior is obtained coherently with the WebUIAdaptation module configurations that explicitly prefer a vertical scrolling respect to a horizontal one.

As described together with the technical explanation above can be highlighted the high level of coherence of the module considering that all the configuration are effectively confirmed observing the resulting adapted web pages.

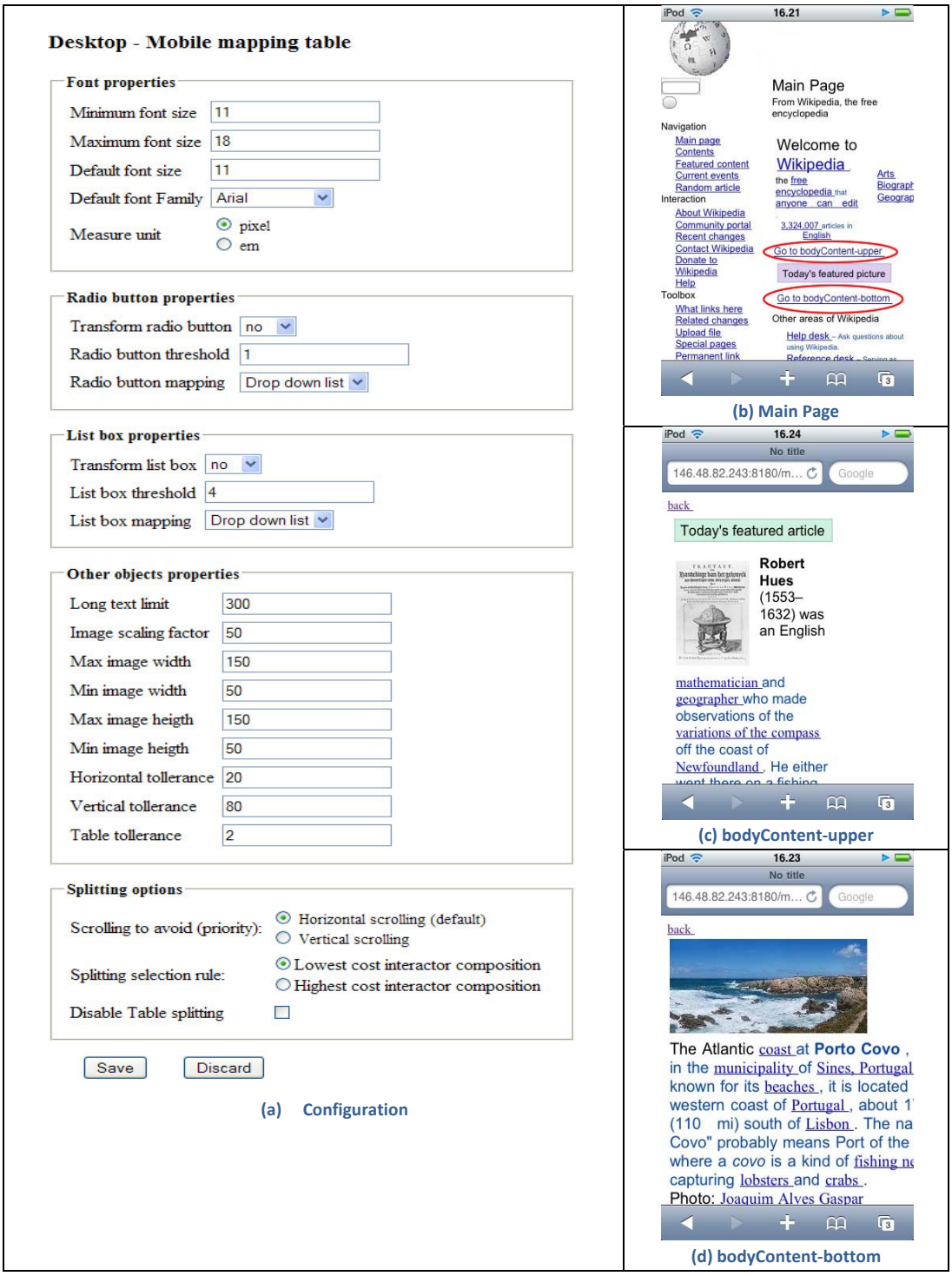


Figure 59 First configuration and corresponding output

Desktop - Mobile mapping table

Font properties

Minimum font size	11
Maximum font size	18
Default font size	11
Default font Family	Arial
Measure unit	(•) pixel ( ) em

Radio button properties

Transform radio button	no
Radio button threshold	1
Radio button mapping	Drop down list

List box properties

Transform list box	no
List box threshold	4
List box mapping	Drop down list

Other objects properties

Long text limit	300
Image scaling factor	50
Max image width	150
Min image width	50
Max image heigth	150
Min image heigth	50
Horizontal tollerance	20
Vertical tollerance	500
Table tollerance	2

Splitting options

Scrolling to avoid (priority):	( ) Horizontal scrolling (default) (•) Vertical scrolling
Splitting selection rule:	(•) Lowest cost interactor composition ( ) Highest cost interactor composition
Disable Table splitting	[ ]

Save Discard

(a) Configuration

(b) Main Page

(c) bodyContent

Figure 60 Second configuration and corresponding output

At the end considering the WebUIAdaptation module it is confirmed how it results highly configurable. Considering the extensibility it is not kept into account because it is clear that once new configurable parameter is considered it is necessary to edit the parser and adapted web page builder features. This can be obviously done changing the module code implementation. A possible future solution that could be investigated could consider implementing the WebUIAdaptation module using an approach like the ALR one. So that it may be extensible adding the necessary code component to manage the new adaptation behavior. Then how the module interacts with the rest of the system could be done through an XML file.
From the consistency point of view the module results are good because the configurations declared in the desktop-mobile configuration results effectively reproduced by the WebUIAdaptation tool behaviors. Another important thing to remember is that it is not necessary to restart the WebUIAdaptation module once the adaptation configuration is changed because it is hot loaded by the module and it becomes immediately effective.
Concluding, this is a good result because in respect to the previous version tested during the first test iteration now the module can be configured considering many parameters and at the same time it results more robust. At the same time it is necessary to keep in consideration that this is one of the more complex modules that compose the OPEN platform and its actual state is a milestone about User Interface adaptation functionalities. Sure next improvements efforts can be focused on its extensibility features.

3.4. OVERALL VIEW

A summary of the single module programmability evaluation is schematically reported in Figure 61 and Figure 62. The situation results very positive considering also that all the suggestions proposed during the first testing iteration results are now implemented and working.

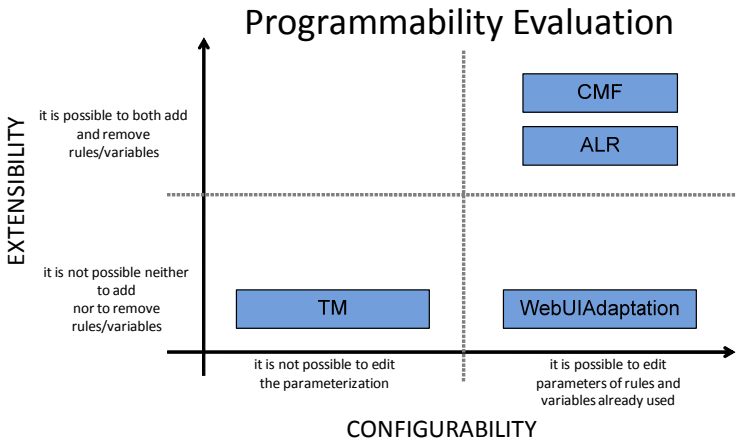


Figure 61 Module programmability evaluation final view

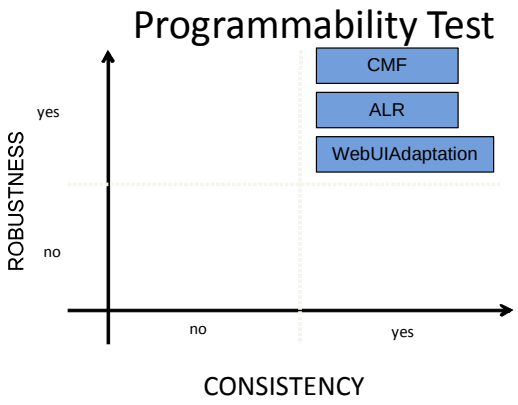


Figure 62 Module programmability test final results

4. TECHNOLOGICAL EVALUATION

In this chapter the technological approach to the second testing iteration is described and the obtained results are discussed. The first section explains the overall methodology performed to evaluate the OPEN Migration Service Platform from a technological point of view. It is important to keep into account that the methodology consists of two parallel tasks: the first focused on initial requirement satisfaction while the second more related to a performance evaluation. The second section shows the results obtained and provides a detailed analysis.

4.1. TECHNOLOGICAL METHODOLOGY

The methodology proposed consists of two different aspects. They together allow giving a complete perspective of how the OPEN Migration Service Platform works from a technological point of view. Conceptually the two questions which it is possible to answer when executing the technological tests are the following:

- Does the OPEN MSP work?
- How well does the OPEN MSP work?

From a more formal point of view to answer the above questions means to consider two different kinds of analysis, respectively:

- Functional evaluation analysis: for each considered requirement it provides a yes/no answer (similar to a standard acceptance test).
- Performance evaluation analysis which carries out, using a quantitative approach, a set of numerical data and measures that are useful to give feedback about how well the OPEN MSP works.

In Figure 63 an overview of the proposed methodology is showed. Note that the two tasks are quite independent and eventually can be executed in parallel.

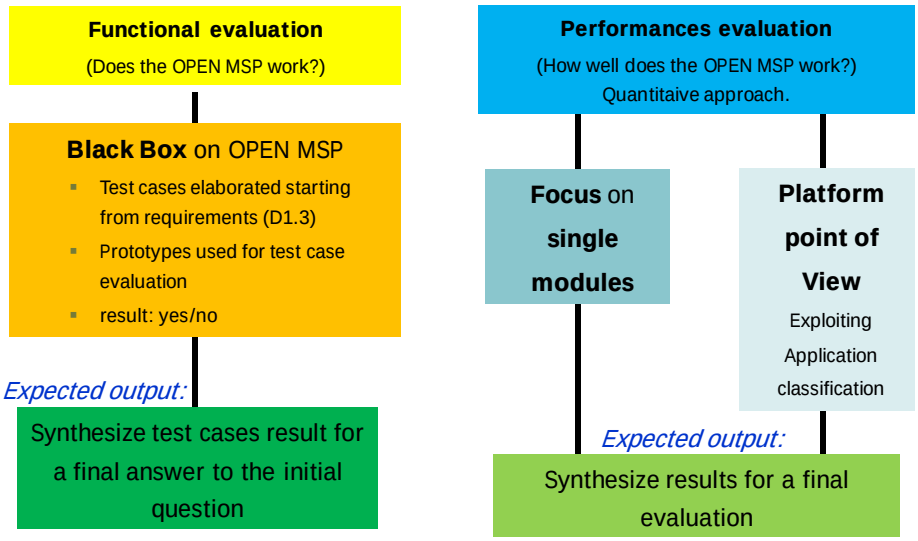


Figure 63 Technology evaluation overall methodology

4.2. FUNCTIONAL EVALUATION

This aspect of the technological evaluation is conceptually equal to an acceptance test. The idea is that now the OPEN Migration Service Platform is completed in its first version, and it is time to check if the requirements elicited in D1.1 [D1.1] and then fixed in D1.3 [D1.3] are satisfied in the final implementation of the platform. The functional evaluation process considers all the requirements elicited in D1.3 [D1.3], possibly adding some or contextualizing the ones expressed only at high level. Since the OPEN MSP is a middleware it is necessary to use some example applications in order to verify that it correctly implements the requirements proposed in elicitation phases. This can be easily done considering as applications the prototypes developed on top of the platform. They are in fact an example of how the OPEN middleware can be used to obtain migratory applications on top of it.

4.2.1. TESTABILITY REQUIREMENTS CLASSIFICATION

Not all the requirements initially elicited in D1.3 [D1.3], when the platform was still just a concept, are actually significant and testable. In order to have a real and pragmatic approach, given a generic requirement, the following "testability" classification is proposed. Each requirement can belong to only one of the following categories:

- **SUPPORTED** means that the requirement is available in the current implementation of the OPEN MSP platform.
    - **VERIFIABLE** means that there exists at least a prototype or an ad-hoc application which is able to verify, through a test case execution, if the requirement is satisfied or not.
        - **PASSED** The requirement is testable and it successfully passed the test case.
        - **NOT PASSED** The requirement is testable but it doesn't pass the test case.
    - **NOT VERIFIABLE** refers to requirements for which no prototype exists able to verify whether or not they work correctly. However, they are implemented in the current version of the middleware.
- **NOT SUPPORTED** The OPEN MSP doesn't implement the requirement in the current version, but it is implementation can be considered in future if the requirement still represents a strategic point.
- **NO ADDED VALUE** Requirements that do not add any value considering the objective of the OPEN MSP. It happens that some aspects that in the early stage of the development are considered as platform requirements result more precisely as already implemented in other software/middleware that can be efficiently used together with the OPEN MSP. It makes in fact more sense to delegate this features to infrastructures and systems specialized to manage this issues (e.g. network aspect, database management). Typically, these requirements refer to an issue that works at lower level respect to the OPEN MSP.

Figure 64 shows in detail the classification described.

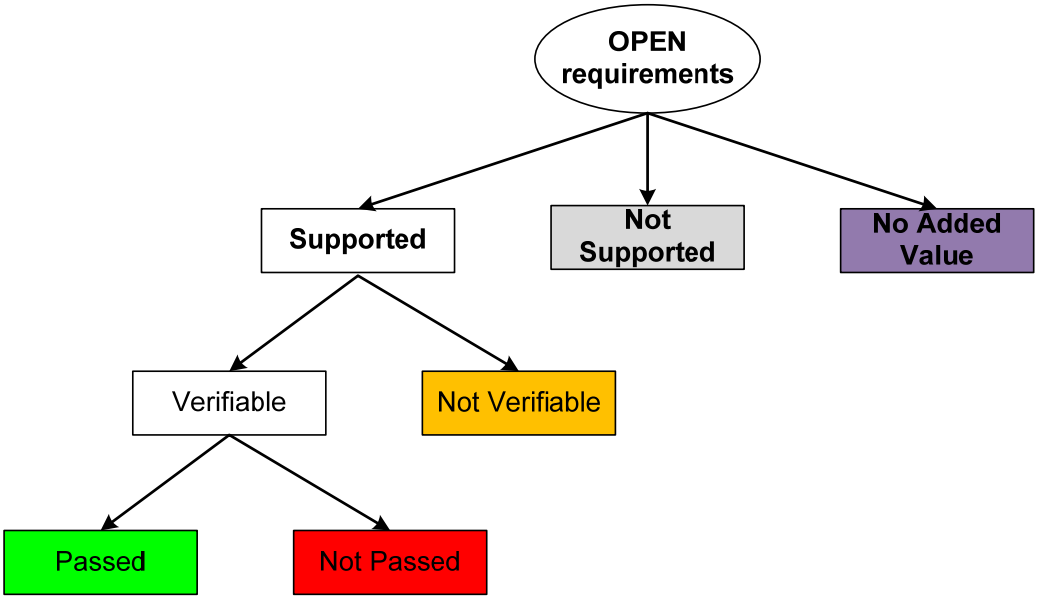


Figure 64 Testability requirements classification

4.2.2. TECHNOLOGICAL REQUIREMENTS CLASSIFICATION

In order to have a simple and clear approach, the requirements are classified i a small number of categories on the basis of which technological features of the platform they refer to. The complete classification is reported in Figure 65, using a set representation. Note that the classification proposed here is slightly different with respect to the one initially proposed in D1.3 [D1.3]. The reason is that in this final validation phase it is important to understand how the whole platform works, so instead of maintaining the requirements classification fragmented as in D1.3, a higher level classification is proposed. The new classification aims to put in evidence the fundamental key points which characterize the platform.

Note that they are very restricted in number, but focus exactly on the main features that characterize the platform itself. The Context, Discovery and Migration class mainly focuses on context management considering also device changes. The Multimodal and Multiuser category refers to the capabilities for more than a single user to use the platform considering also different interaction modalities while the Platform Facilities category collects considerations about the possibility to develop migratory applications on top of the platform. A detailed description of the contents of each category is reported in the following, giving also a quantitative measure of the class dimension. Performing the classification proposed makes also it possible not to directly consider some requirements expressed in D1.3 [D1.3] because some of them are already and better stressed and tested during usability rather than programmability

evaluation. For example, all the requirements related to the user experience which is enabled by the platform to the end user fall in this case.

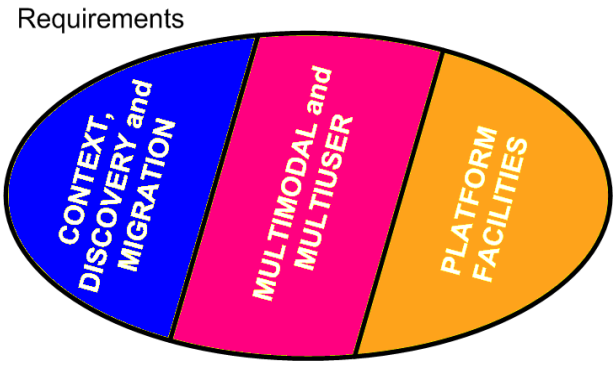


Figure 65 Technological requirements classification

Table 16 reports the technological requirements classification, referring to each requirement through its id number reported in D1.3 [D1.3]. A pie chart to put in evidence the classes' sizes is showed in Figure 66.
The detailed description of each technological class is reported in the following paragraphs, where the test case execution results for functional tests are also analyzed.

Technological Requirements Class	Requirements ID in D1.3
Context, Discovery and Migration	20, 22, 41, 44, 47, 54, 59, 74, 78, 82a, 82b, 82c, 82d, 91, 93, 98, 126, 128, 129, 135, 136, 157, 162, 163
Multimodal and Multiuser	4, 36, 40, 48, 52, 58, 66, 88
Platform Facilities	46, 100, 123, 161
No Added Value	15, 24, 26, 27, 57, 63, 69, 71, 84, 90, 94, 101, 108, 114, 118, 127, 132, 140, 141, 143, 159,

Table 16 Technological requirements classification of requirements elicited in D1.3

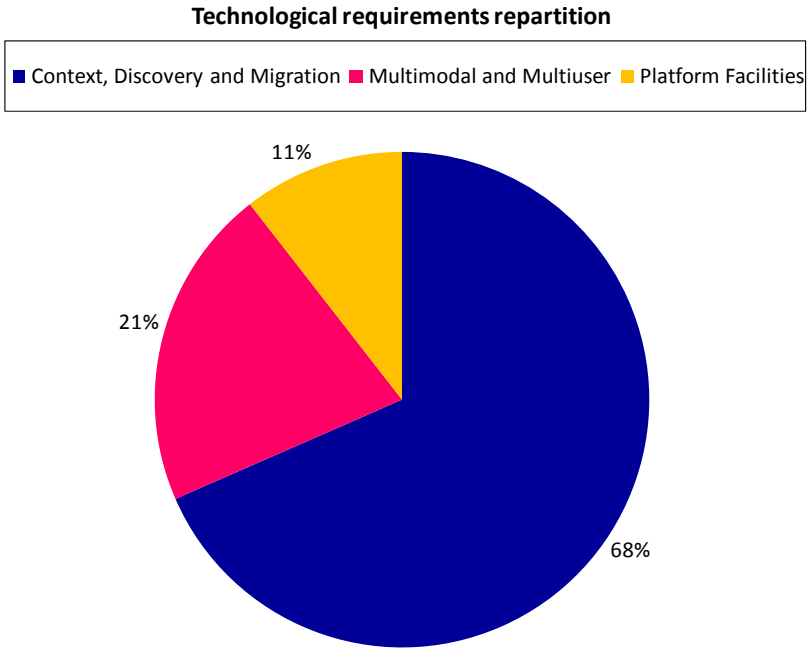


Figure 66 Technological requirements classification in technical classes

The following consideration about the technological testing activity from a methodological point of view applies:

- More than half (67%) of the requirements refer to Context, Discovery and Migration topics. It is consistent because these are the issues that more characterize the OPEN project from a research point of view. It means that a very big effort was addressed on the migration aspects, device discovery and context migration, consistently with the main goals of the project.
- A good part (22%) of requirements refers to platform aspects that involve more than a single user, considering also different interaction modality. It is important to underline that this class is a sort of extension of the primary project requirements considered in the context, discovery and migration one. This class provides further innovative features to the entire platform. It can also represent the starting point for possible exploitation or next steps.
- A small amount (11%) of requirements is devoted to the Platform Facilities available for the development of new applications and services on top of the OPEN MSP. A more detailed documentation for the developer can be found in WP 4 deliverables.

4.2.3. FUNCTIONAL EVALUATION METHODOLOGY

Considering the technological classification presented above, for each technological class each requirement is singularly analyzed, verifying if it is testable or not according to the testability classification reported in paragraph 4.2.1.

For all the requirements that are “Verifiable”, a test case is built describing the expected output, too. Note that all the test cases are in a form such that, executing them using the implemented platform, it is possible to give a pass/fail answer. As said before, a lot of the requirements need some applications on top of the OPEN platform to proof their validity, so a mapping between requirements and prototypes that can be used to test them is done, as graphically shown in Figure 67. Please note that the objective of this validation phase is to evaluate the OPEN Migration Service Platform and not how a single prototype works. By executing the produced test cases, it is possible to evaluate the fulfillment of the OPEN requirements in order to have a complete evaluation of the OPEN platform.

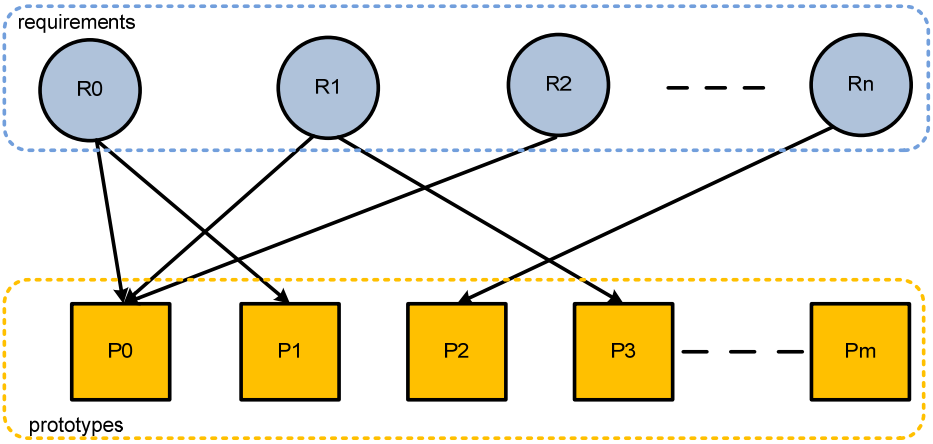


Figure 67 Mapping between verifiable requirements and prototypes

In order to have a unique approach, each test case is specified using the fixed structure reported in Table 17. Once the test is executed the result is clearly readable in the ”Status” field, which can be only “Passed” or “Not passed” without any intermediate evaluation: this is the final validation test phase and it is necessary to point out if the OPEN implementation works or not.

ID	*Test case ID format FExx* *FE stand for Functional Evaluation followed by two progressive numbers*	**Status**	*Passed/Not passed*
Executor	*Test executor*	**Execution date**	*Execution test date*
Item	*What the test case evaluates*	**Prototype**	*Considered prototype(s)*
Description	*Description of the considered requirement(s) taken from D1.3*		
Input	*Input for test execution*		
Expected output	*Expected output of test in case of success execution*		
Actual output	*Output of the executed test*		
General consideration	*Additional notes*		

Table 17 Test case structure description

For all the other requirements that are not “Verifiable” a motivation is reported to specify why the requirement is considered not verifiable. All the test cases considered and the complete explanation about how each requirement is classified is reported in Appendix C. The objective of each use case is to evaluate a single requirement but it can happen that two different requirements are mapped on different use cases, thus requesting the tester to execute a closer task list.
Just to give a synthetic view of the functional evaluation results observing Figure 68 it is possible to point out the following considerations:

- The majority of the requirements are testable; it means that a test case can be produced to evaluate them.
- A successful result is always obtained (0% test cases fail).
- Only 13% of the initial requirements are not supported and not implemented in the final version of the platform.
- A very small amount of requirements (5%) is “Not Testable”. It means that the prototypes developed on top of the OPEN platform guarantee an optimal coverage of the use case in which the OPEN MSP can be used.

In the following Sections the synthetic analysis of the results obtained is presented. In order to have a more detailed analysis, results are described considering one class of technological requirement at a time.

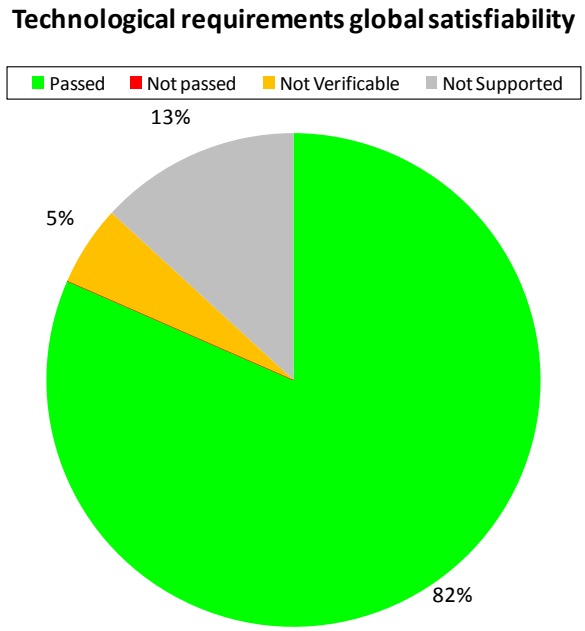


Figure 68 Requirements satisfaction considering all the technological requirements classes

4.2.3.1. CONTEXT, DISCOVERY AND MIGRATION REQUIREMENTS

In this section the functional requirements about Context, Discovery and Migration are considered. Note that this is the technology requirements class that more characterizes the OPEN Migration Service Platform. Basically the attributes evaluated here refer to the context management and migration on different devices. Referring to the migration also the interface adaptation features is kept into account. Moreover the device discovery capability is considered in order to correctly recognize the devices in the vicinity of which can be a source and/or target for the migration process. It is clear that this class of requirements is fundamental for the platform, independently from which is the prototype used to perform the test.
The complete test case descriptions with all collected results are available in Appendix 1.D.1. A synthetic description of test results for this requirements class is reported in Figure 69. Analyzing the pie chart it is possible to formulate the following consideration:

- All test cases that can be executed give a successful output; in fact there isn't any "Not Passed" test case.
- The 77% of requirements give successful results as reported by the green area of the chart. The complete description is available in Appendix D.1.1.
- The "Not Verifiable" requirements are only 8% so it is clear that about context, discovery and migration topics a very good work was done implementing on top of the OPEN platform applications and utilities are able to stimulate the middleware almost in all its features. See description available in Appendix D.1.2.
- A small percentage (15%) of requirements in this class is not supported. It means that the OPEN Migration Service Platform implementation is well consistent with the initial research idea elicited using very high level requirements in D1.3 [D1.3]. The detailed test case description is available in Appendix D.1.3.

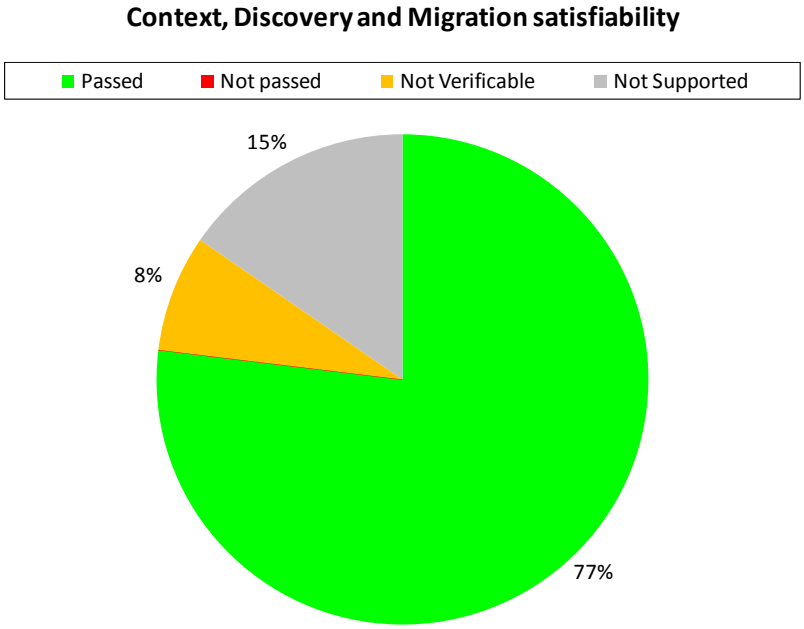


Figure 69 Context, Discovery and Migration requirements satisfiability

4.2.3.2. MULTIMODAL AND MULTIUSER REQUIREMENTS

In this section the requirements related to all the features considering multimodality and multi-user aspects are collected and analyzed. With multimodal requirements the focus is on the evaluation of the different interaction modality available to the devices on which the application is running. An interesting evaluation is about the change of interaction modality after a migration occurs. This is a very important issue considering that the migratory application can completely change the device typology and consequently the interaction paradigm with users. The multi-user requirements help understanding how more than one user can use the OPEN MSP at the same time maintaining high the engagement and the level of the migration experience for each user.

The complete test case descriptions with all collected results are available in Appendix 1.D.2. A synthetic view about test results for this requirements class is reported in Figure 70. Analyzing the pie chart it is possible to formulate the following consideration:

- A lot of requirements (87%) in this category result in “Passed”. However all tests that can be executed give a success output; in fact there isn’t any “Not Passed” test case. It means that a reduced subset of the planned features is implemented, but what is implemented is developed in a very good way. The complete description is available in Appendix D.2.1.
- Only 13% of the requirements results in “Not Supported”, that is a smaller parcentage than in Context, Discovery and Migration class. The complete description is available in Appendix D.2.2.

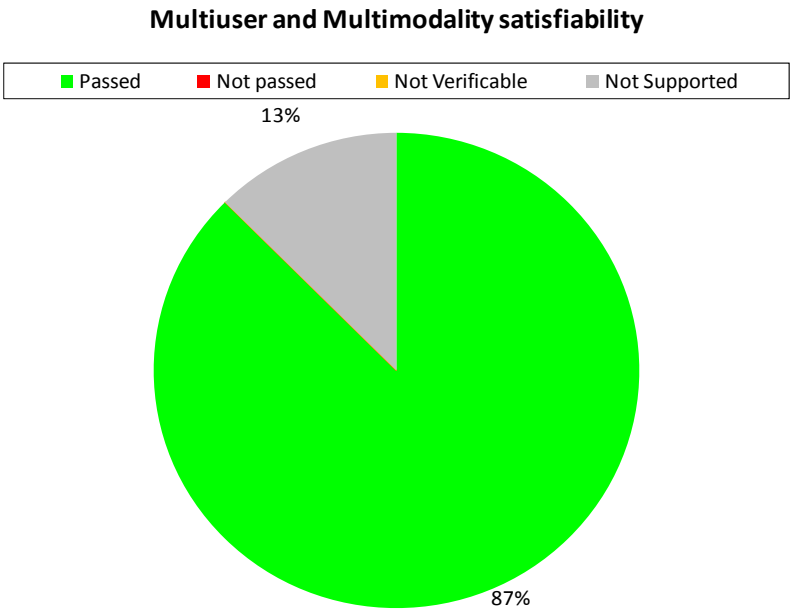


Figure 70 Multiuser and Multimodality requirements satisfiability

4.2.3.3. PLATFORM FACILITIES REQUIREMENTS

This section collects all the requirements that are related to the platform facilities available for the development of new applications on top of the OPEN Migration Service Platform. For this class of requirements it is not possible to develop specific test cases because they refer to the development of something for which the platform is at a pre-requirement stage. Note that these characteristics have been already addressed in the previous chapter considering programmability, from the developer point of view. As reported in Figure 71 it is clear that a point of strength of the platform is the possibility to be expanded, by adding in easy way new features. This is underlined by the modular structure of the entire middleware which facilitates both bug fix and new functionality development.

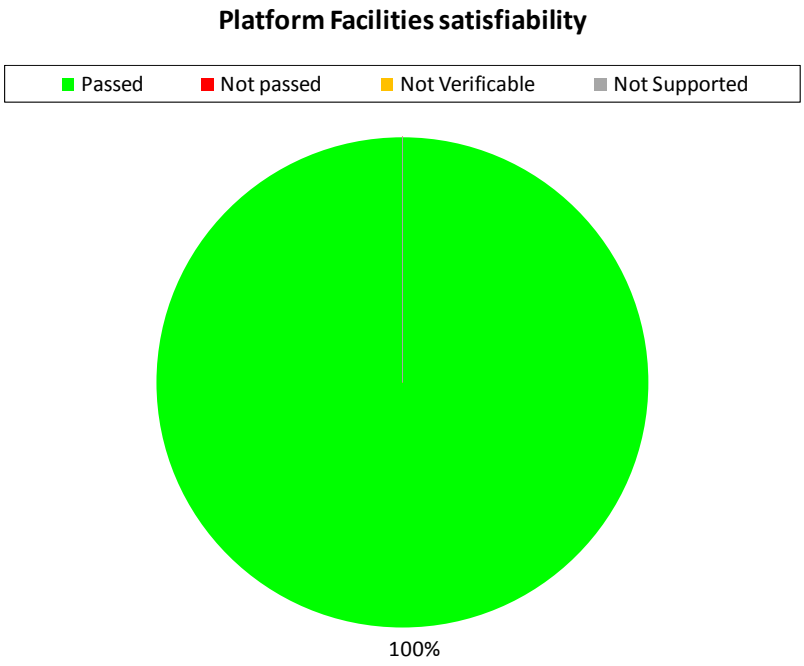


Figure 71 Platform facilities requirements satisfiability

4.2.3.4. MULTICORE

With multicore, the features that enable the migratory application to adapt its behavior in term of concurrent threads are considered. In this category some requirements are assumed in the initial planning, but they are not completely developed at the end of the project. It is necessary to remember that due to the economical crisis the partner that works on Multicore features (NEC) closed its research laboratory involved in the development since Dec 31st 2009. So the features that fall in the multicore category are not implemented in the final version of the OPEN platform. In Appendix 1.D.4 a detailed description of the multicore requirements not implemented is reported.

4.2.3.5. NO ADDED VALUE REQUIREMENTS

This section collects and analyzes all that features which were initially considered as requirements of the OPEN MSP, but after a deeper analysis resulted better classifiable as no-added-value requirements. These changes in classification are mainly related to the fact that the features described by these initial requirements are not peculiarity of the OPEN platform, but are typically well-know technical problems whose solutions are widely implemented in optimized ways, too. Just to give an example, let's consider the network configuration necessary to use the OPEN Migration Service Platform, which is classified as requirement in D1.3 [D1.3]. It is clearly

not possible to develop specific test cases, because the requirement refers to a configuration which is out of the OPEN MSP control, being managed by the Operative System and Network Services of the device on which the OPEN platform is running. Another case is about the security of the network communication: it can be provided as pre-requirement by the network infrastructure in a very efficient way rather than implemented using an application level protocol that causes overhead and loss of performances for the OPEN migration purposes.
A detailed analysis specifying why each initial requirement is put in no added value requirements class is given in Appendix 1.D.5.

4.3. INDICATOR EVALUATION

In this section the OPEN Migration Service Platform is evaluated from a technical point of view, considering all the 7 indicators already identified and proposed in D6.4 [D6.4]. While some requirements are related to performance measures, others refer to adherence to the standards aspects, considering both network protocol level and higher levels, like the HTML code validation.
In order to execute all the tests and measures the last version of the OPEN Migration Service Platform is set up on a dedicated server (OPEN Server) connected to an ad-hoc local network called OPEN NET. The OPEN NET provides both cable connection and WiFi so that both old devices and WiFi enabled ones can be used for the tests execution. The complete test scenario is always the same for all the indicators, and a detailed topology schema is shown in Figure 72.
On the OPEN Server consists of an Intel Core 2 Duo processor @ 2.4 GHz with 4 GB of RAM. It is connected to the network through a 10/100 Mbit Ethernet adapter. The router enables the Internet connection, necessary for some on top applications that collect update information from the web, such as traffic data. The server is equipped with Microsoft Windows 7 Ultimate 64 bit as operative system.
The next paragraphs describe in a synthetic way how the quantitative data necessary for the indicator evaluations are recorded.

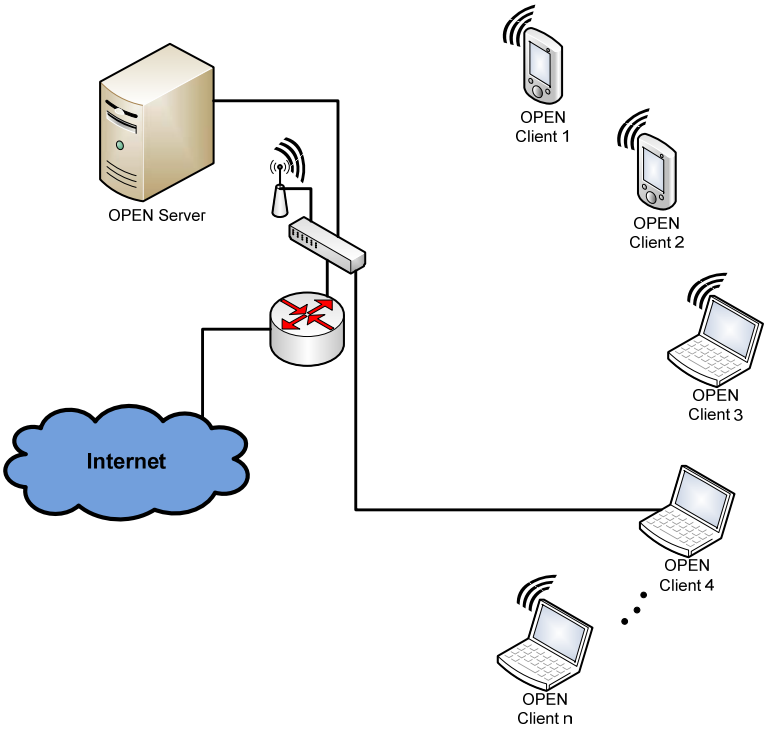


Figure 72 Indicator evaluation general scenario

4.3.1. AVAILABILITY

The availability of a system is the percentage of time when the system is operating. According to this standard definition, the test described in Table 18 is performed, and the obtained results are reported here.

Availability Test	
Description	Availability is monitored recording possible failures, and their duration while executing the OPEN MSP.
Test	The OPEN MSP is actively used during the whole working day by different users and left up also during the night. All the recorded data containing every kind of issue are available in logs files. The test duration is considered of 3 days (from 22 June 2010 at 1:00 PM to 25 June 2010 at 1:00 PM); 20 migration sessions are evaluated.
Involved devices	1 OPEN Server. 5 different OPEN clients.
Migratory applications involved	Social Game Emergency
Obtained Output	The OPEN server correctly manages all the devices and applications registration without any error for all 20/20 considered cases. It is also always reachable from all devices, both the wired connected and the WiFi ones. The only loss of service is due to the Operative System that on 23 June night, at

	3:00 AM restarted the machine due to the scheduled update. Since the OPEN MSP server was not set to be automatically startup, it remained down until the tester identified the problem, reading the log the next morning.

Table 18 Availability test description

Please note that in this second iteration phase, the availability evaluation was possible, while in the first iteration it was meaningless, due to the instability of the OPEN platform. This is the most important fact to demonstrate how the platform has now improved and, as the results show, that it is now stable, the only loss of service during the test having been caused by the operating system updates and not by any platform error.

4.3.2. RELIABILITY

The reliability of a system is the ability of the system to perform its required functions under stated conditions for specified period of time. According to this standard definition the test described in Table 19 is performed obtaining the results reported here.

Reliability Test	
Description	Reliability is monitored recording possible failures during a complete E2E execution of the prototype. It is the probability that the system does not fall in the interval 0…t, knowing that at t=0 it was correctly working. This test is executed for 20 times.
Test	The OPEN MSP is actively used during the whole working day by different users and left up also during the night. All the recorded data containing every kind of issue are available in logs files. The test duration is considered of 3 days (from 22 June 2010 at 1:00 PM to 25 June 2010 at 1:00 PM).
Involved devices	1 OPEN Server. 5 different OPEN clients.
Migratory applications involved	Social Game Emergency
Obtained Output	The execution of the 20 migrations was requested when the OPEN platform was correctly working and they all were correctly managed. So a 20/20 successful rate is obtained and this proves how the OPEN platform is reliable in the current implementation.

Table 19 Reliability test description

Like for the availability, note that in this second iteration phase the reliability evaluation was now possible, while in the first iteration it made no sense, due to the instability of the OPEN platform.

4.3.3. PERFORMANCE

The main indicator about performances, considering the OPEN Migration Service Platform, is certainly the migration time. It measures the time the platform requires for migrating the on top application, or a part of it, from a device to another.

Migration time is recorded starting from the migration orchestration log with a sensibility of 1 millisecond.

- T1 = start of the *triggerMigration* method on Orchestration Server
- T2 = end of the *migrationCompleted* method on OrchestrationServer

Always T2 > T1, so that Migration Time = T2 - T1.
The migration time has been recorded considering the following application:

- Social Game;
- Emergency;
- Web page applications.

Note that the *migrationCompleted* method is called when the application starts to run on the new device, so the 1st step of execution always considers also the time necessary to start the browser in which the migrated application has to run.

<table>
<tr><th colspan="2">Performances - Migration Time [Emergency]</th></tr>
<tr><td>Description</td><td>Migration time recording</td></tr>
<tr><td>Test</td><td>Migration time is recorded starting from the migration orchestration log with a sensibility of 1 millisecond.</td></tr>
<tr><td>Involved devices</td><td>1 OPEN Server.
2 different OPEN clients.</td></tr>
<tr><td>Migratory applications involved</td><td>Emergency</td></tr>
<tr><td>Test details</td><td>All this migration is from PC to PC, so no adaptation should be required.
<table>
<tr><th></th><th colspan="2">1st migration</th><th></th></tr>
<tr><th>Test ID</th><th>from</th><th>to</th><th>type</th></tr>
<tr><td>1</td><td>Flood</td><td>Traffic</td><td>total</td></tr>
<tr><td>2</td><td>Flood</td><td>Traffic</td><td>partial</td></tr>
<tr><td>3</td><td>Traffic</td><td>Flood</td><td>total</td></tr>
<tr><td>4</td><td>Traffic</td><td>Flood</td><td>partial</td></tr>
<tr><td>5</td><td>Flood</td><td>Wall</td><td>total</td></tr>
<tr><td>6</td><td>Flood</td><td>Wall</td><td>partial</td></tr>
<tr><td>7</td><td>Traffic</td><td>Wall</td><td>total</td></tr>
<tr><td>8</td><td>Traffic</td><td>Wall</td><td>partial</td></tr>
</table></td></tr>
</table>

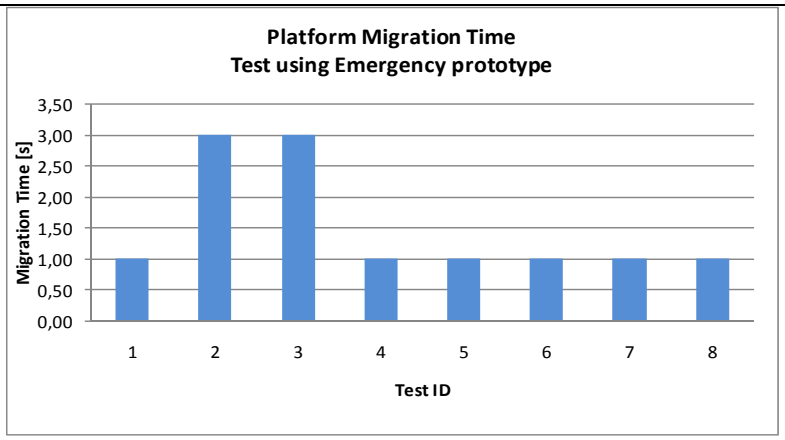
Results	
Observations	Even if different migration typologies are considered (partial/total/flood/traffic/...) the required time for migration process is almost the same. This aspect is clear from the plot above. The mode is 1 second; this is a good result because it means that the most migrations require this amount of seconds.

Table 20 Performance, migration time evaluation using Emergency

Performances - Migration Time [Web migration]	
Description	Migration time recording
Test	Migration time is recorded starting from the migration orchestration log with a sensibility of 1 millisecond.
Involved devices	1 OPEN Server. 2 different OPEN clients.
Migratory applications involved	Web migration
Test details	All tests performed here consider a migration form PC to mobile device that requires UI adaptation computation.

Web page	Component(s)	From	To
Social Game	Betting + Chat	PC	Mob
W3C	Total	PC	Mob
eBay	Total	PC	Mob

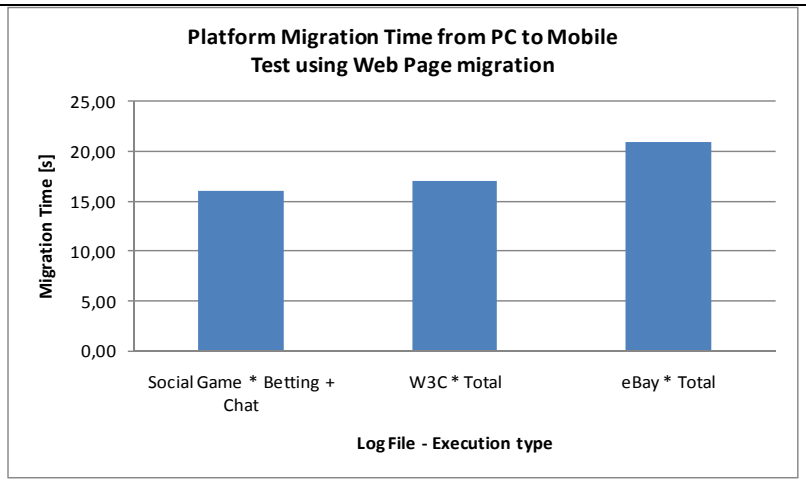
Results	
Observations	The required time for migration process depends on the number and the kind of information that is requested to migrate, considering that it is also necessary to adapt each one of the migrating components. This aspect is clear from the plot above, where different web pages (or pieces of them) are migrated obtaining different times, on the basis of the content and structural complexity of the page considered. Please note that the times recorded here are greater than the ones recorded in other tests because here the device change requires the UI adaptation step, not considered in previous cases.

Table 21 Performance, migration time evaluation using Web page migration

4.3.4. ACCESSIBILITY

The accessibility of the platform here is relative to web pages. It evaluates potential losses of accessibility adapting a web page from the desktop version to the mobile one. The evaluation is carried out with the WAVE (Web Accessibility Evaluation) tool [WAV-10] comparing the number of non conformities of the same web page first considering a desktop visualization and then evaluating the corresponding adapted version on mobile device.
WAVE is a free online tool through which page it is possible to submit a link, upload a file or directly check for accessibility the HTML code of a page. As a result of submitting a page under the WAVE tool a report is produced, in which a number of icons are superimposed on the page in correspondence with the areas where the issue occurs. An example of WAVE report is visualized in Figure 73. The WAVE evaluation is performed on a number of pages, which are listed in Table 22.

Site	URL
British Airways	http://www.britishairways.com/travel/flight-deals/public/en_gb?link=main_nav
Fifa News	http://www.fifa.com/worldcup/news/news.html
Google Advanced Search	http://www.google.com/advanced_search?hl=en

NY Times	http://www.nytimes.com
ESA	http://www.esa.int/esaCP/index.html
Weather.com	http://www.weather.com

Table 22 List of pages considered for evaluation of accessibility and validation

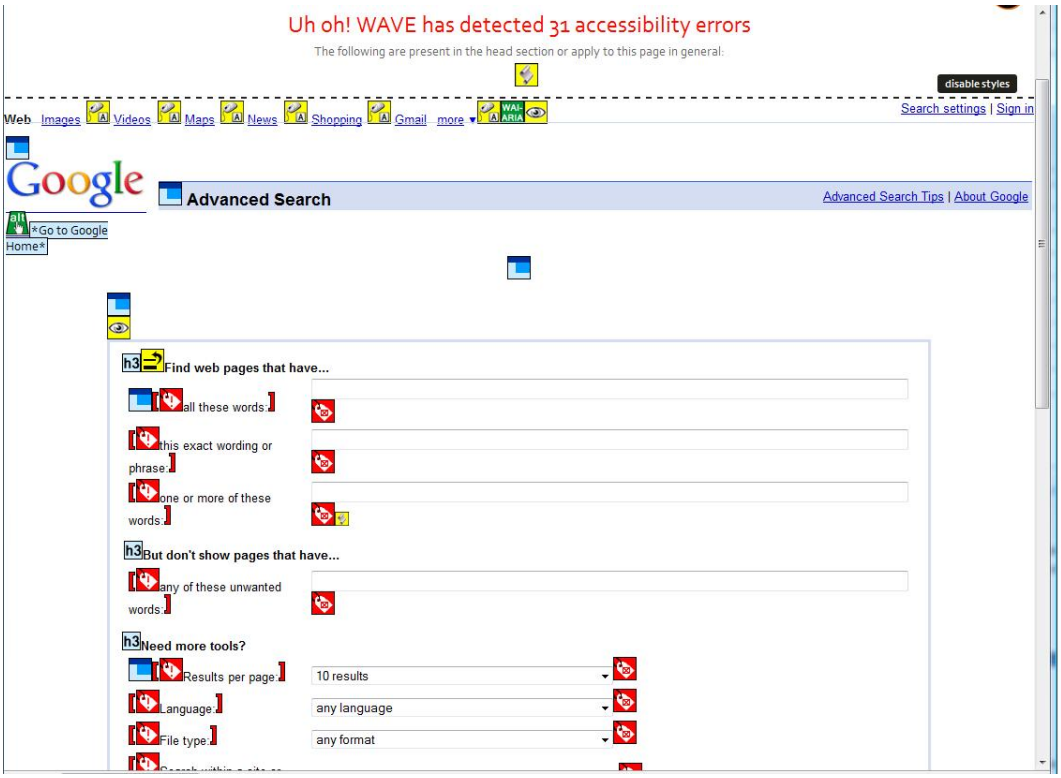


Figure 73 The WAVE report of the Google Advanced Search page

WAVE reports include a number of icons on where problems of accessibility occur. In particular, there are:

- red icons to indicate accessibility ERRORS, which almost certainly cause accessibility issues;
- yellow icons indicate ALERTS: they may or may not be accessibility issues, and typically indicate an area where accessibility might be an issue or where it may be made better.
- green icons which means ACCESSIBILITY FEATURES;
- light blue icons (SEMANTIC, or NAVIGATIONAL ELEMENTS)
- trapezoid shaped icons relate to images.

The reported analysis focuses only on errors and alerts/warnings. In addition, it is in depth analyzed further the nature of errors reported. The list of WAVE icons related to ERRORS, together with a description of their meaning is provided in Table 23.

Errors

Icon	Title	Description
	ERROR: Missing alternative text	Alternative text is not present for an image.
	ERROR: Spacer image missing alternative text	Alternative text is not present for an image used as a layout spacer.
	ERROR: Linked image missing alternative text	Alternative text is not provided for an image that is the only thing within a link.
	ERROR: Image button missing alternative text	Alternative text is not present in a form image button.
	ERROR: Image map missing alternative text	Alternative text is not present for an image that has hotspots.
	ERROR: Image map area missing alternative text	Alternative text is not present for an image map area (hotspot).
	ERROR: Server-side image map	A server-side image map is present.
	ERROR: Invalid longdesc	The longdesc attribute does not contain a URL.
	ERROR: Form label missing	A form <input>, <select>, or <textarea> does not have a corresponding label. (Note: Labels are not required for image, submit, reset, button, or hidden form element types.)
	ERROR: Empty form label	A form label is present, but it does not contain any content.
	ERROR: Multiple form labels	A form element has two labels associated with it.
	ERROR: Orphaned form label	A form label is present, but it is not associated with any form <input>, <select>, or <textarea>.
	ERROR: Frame missing title	A frame does not have a "title" attribute or value.
	ERROR: Broken skip navigation link	A skip navigation link exists, but the anchor for the link does not exist.
	ERROR: Empty heading	A heading contains no content.
	ERROR: Marquee	A <marquee> element is present.
	ERROR: Blinking content	The <blink> element is present.
	ERROR: <title> is missing or not informative	The page title is missing or not descriptive.
	ERROR: Empty link	A link contains no text.
	ERROR: Empty table header	A table header contains no text.

Table 23 The list of ERROR Icons in WAVE

The procedure used for conducting the WAVE analysis was the following:

- First it is executed a WAVE analysis on the original (desktop) version of each page listed in Table 22, and recorded the number (and type) of errors listed in the related WAVE report. In addition, the number of alerts/warnings signaled by the tool is recorded too. Then, a desktop-to-mobile migration of this page is performed.
- Then the migrated resulting page(s) on the mobile device (and then, adapted by the semantic redesign module in order to take into account the characteristics of the new device) were submitted under WAVE evaluation.
- Finally, the results produced by the tool on the resulting page(s) were compared with the results produced by the tool on the original desktop page, in order to evaluate whether the pages obtained after a migration process were improved or not from the point of view of accessibility issues.

It is worth pointing that, as a consequence of the step carries out by the Semantic Redesign module, the original desktop page might have been split in two or more pages. In this case, the errors/warnings obtained on the split pages are summed up and compared with the corresponding errors/alerts issued by the WAVE tool on the original desktop page.
The following images, schematically describe the results obtained by WAVE tool on the various pages considered for evaluation. It is worth pointing out that, for the WAVE evaluation, for each page evaluated two images are provided. The first one shows the total amount of errors and warning for both the original (desktop) page and the migrated (mobile) page; the second one shows the details for the errors, grouping such errors with respect to their type.

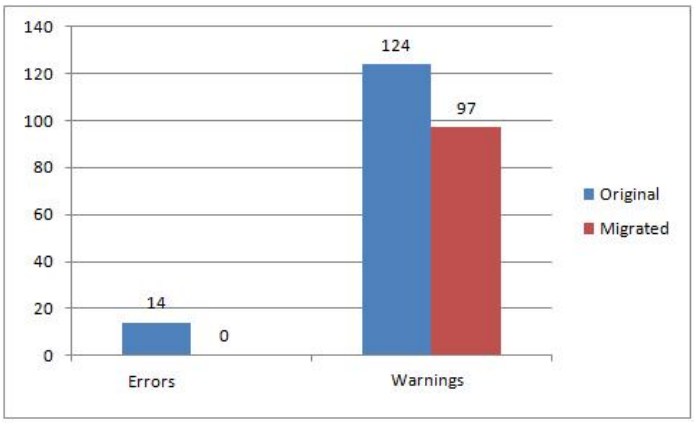


Figure 74 Results of WAVE evaluation on British Airways page

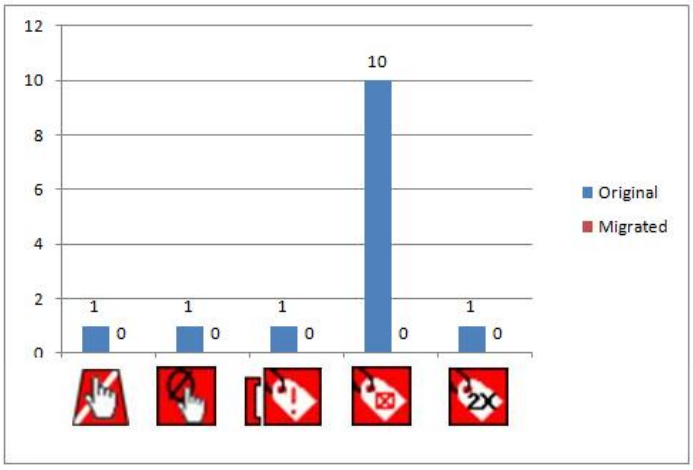


Figure 75 BA page - Wave ERRORS Details

Regarding the BA page, the original page had 14 errors. In particular, 10 out of these 14 errors where occurrences of the same error related to form label missing (see Figure 75): a typical case for this error is when a form <input> or <select> element does not have a corresponding label. As showed in Figure 74, in the resulting (two split) pages obtained after migration, there was no error anymore. Then, the errors from the missing label were fixed by properly associating a form

label with the related input element. In addition, as it is possible to see from Figure 74, also the alerts/warnings were reduced on the migrated page.

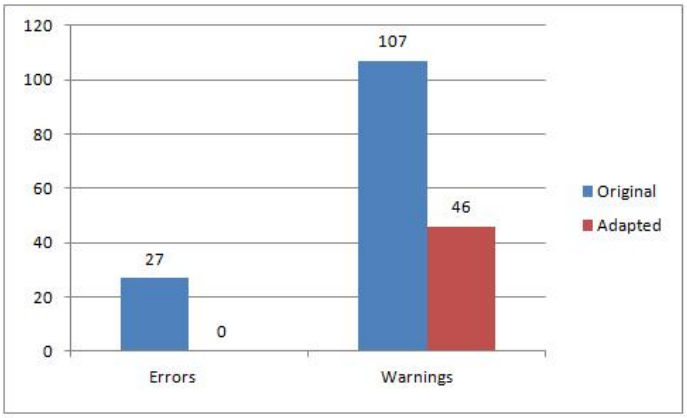


Figure 76 Results of WAVE evaluation on Fifa news page

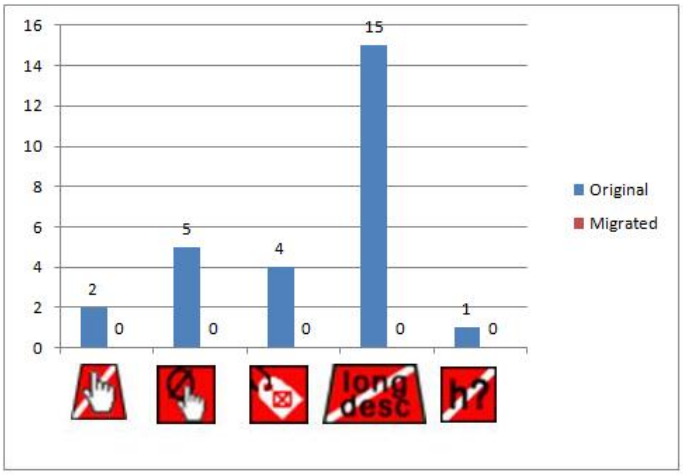


Figure 77 FIFA - Wave Error Details

Regarding the FIFA news page, the original page had 27 errors (see Figure 76). In particular (see Figure 77), 15 out of these 27 errors where occurrences of the same error related to invalid “longdesc” (a typical case for this error is when the “longdesc” attribute does not contain a URL). As you can see from Figure 76, in the resulting (two split) pages obtained after migration, there was no error anymore. In addition, as it is possible to see from Figure 76, also the alerts/warnings were reduced on the migrated page(s).

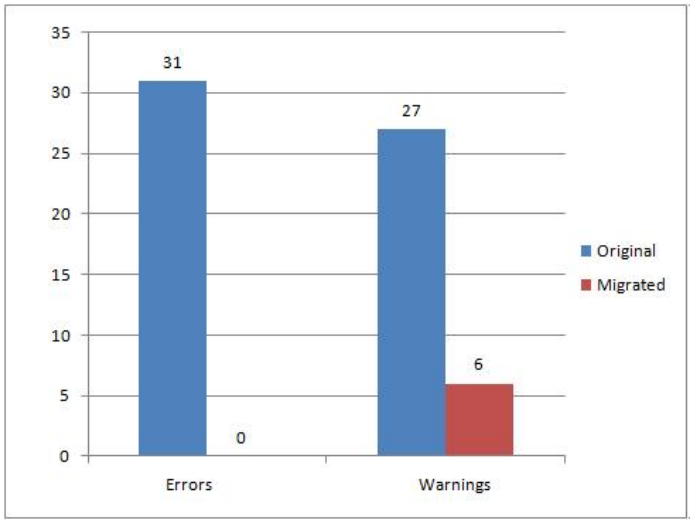


Figure 78 Results of WAVE evaluation on Google Advanced Search page

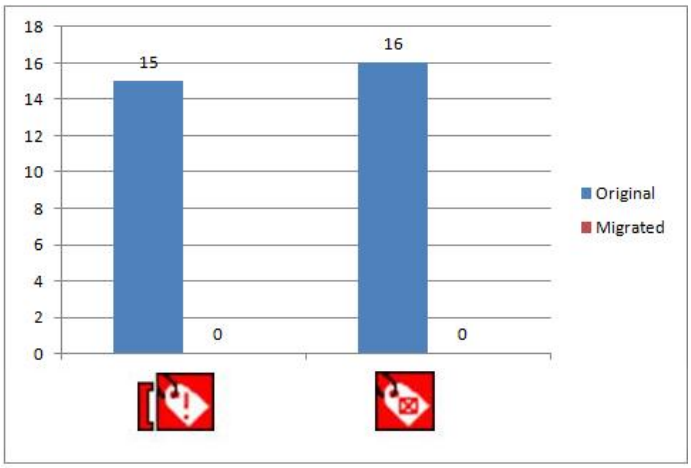


Figure 79 Google Advanced Search – Wave Error Details

Regarding the Google Advanced Search page (see

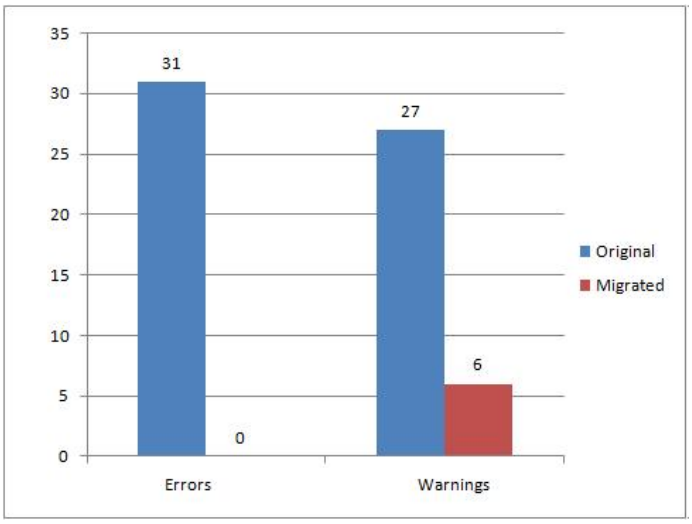


Figure 78), the original page had 31 errors. Such 31 errors were just occurrences of only two types of errors (see Figure 79): orphaned form label (which is when a form label is present but not associated with any form <input>, <select>, etc.) and form label missing (which is when a form or a text area does not have the corresponding label). As you can see from

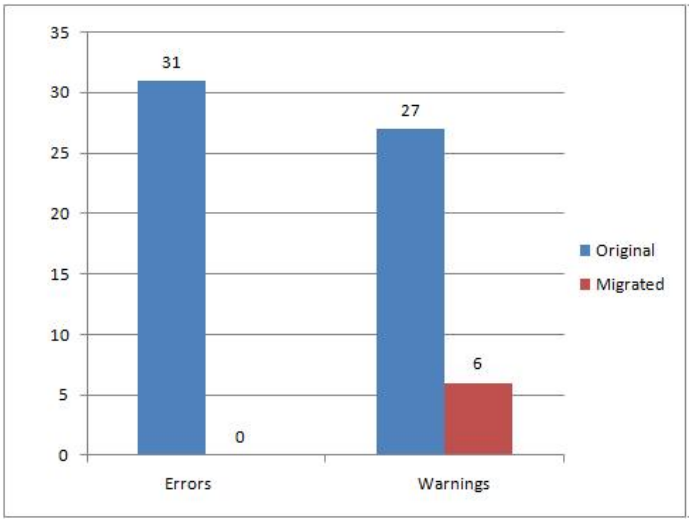


Figure 78, in the resulting page obtained after migration, there was no error anymore. In addition, as it is possible to see from

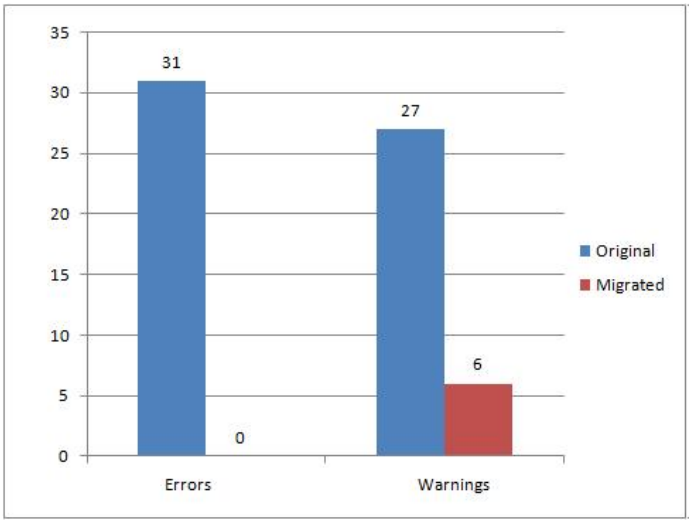


Figure 78, also the alerts/warnings were reduced on the migrated page (from 27 to 6).

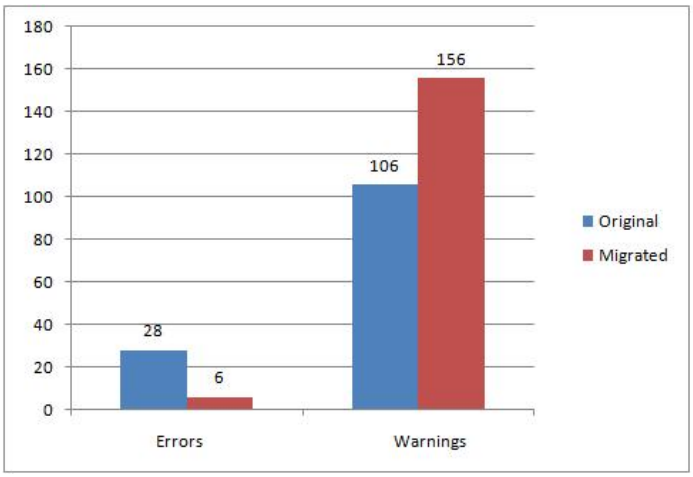


Figure 80 Results of WAVE evaluation on New York Times page

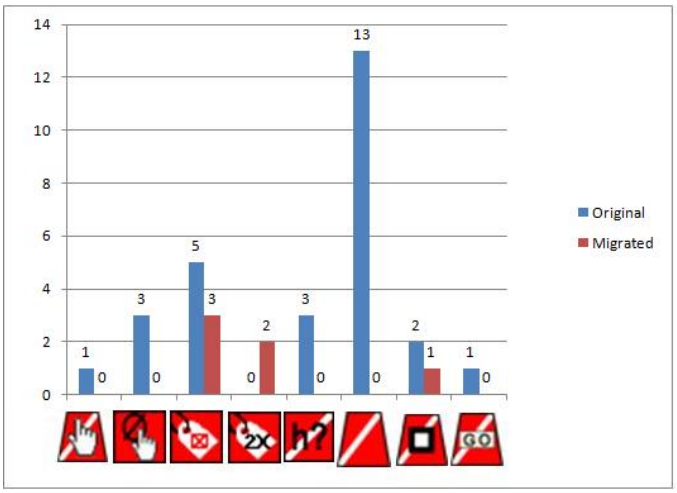


Figure 81 New York Times – Wave Error Details

Regarding the NYTimes page (see

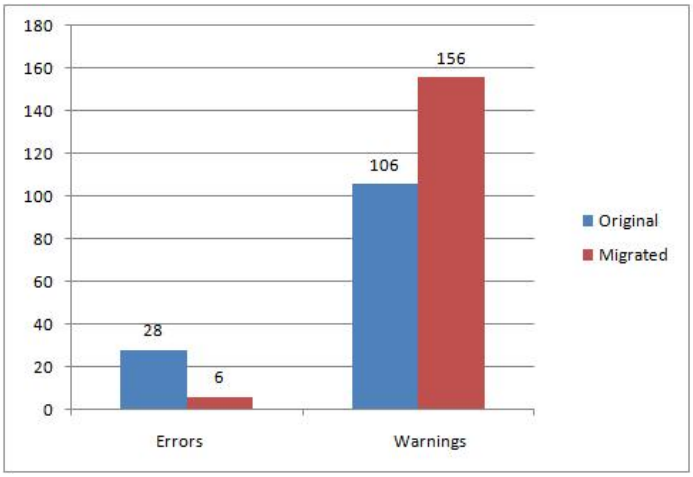


Figure 80), the original page had 28 errors. Such 28 errors were occurrences of a number of errors, which are shown in

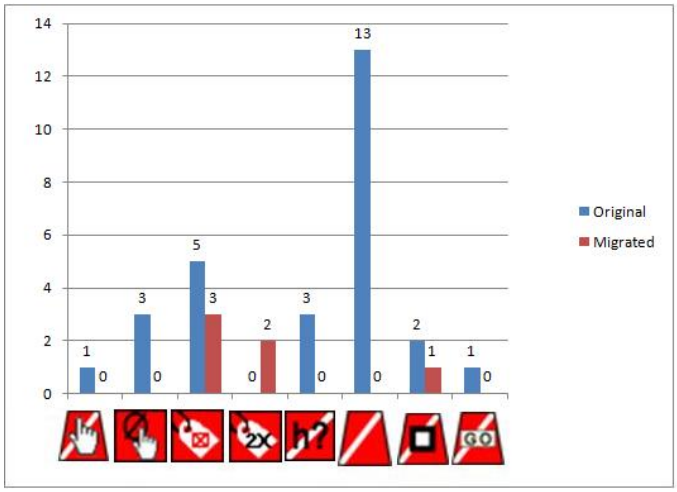


Figure 81. As a result of migration, there was a substantial reduction in errors (from 28 to 6), while there was an increasing in the number of alerts/warnings (see

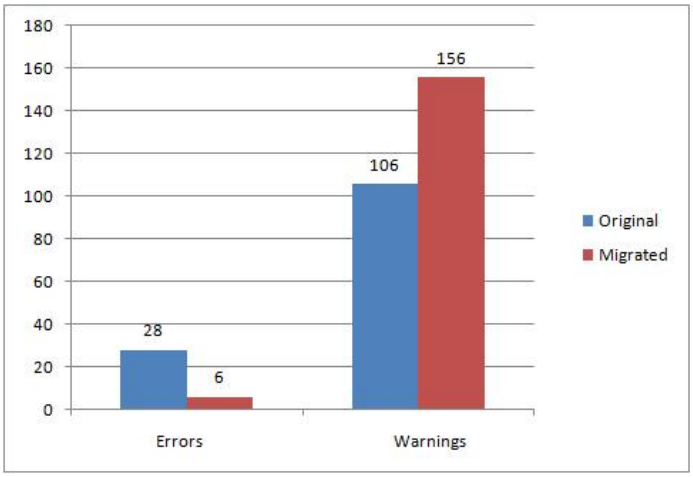


Figure 80). This can be explained by the fact that some errors were just ameliorated in their impact on raising accessibility issues, and then they might have contributed to increase the number of warnings. In

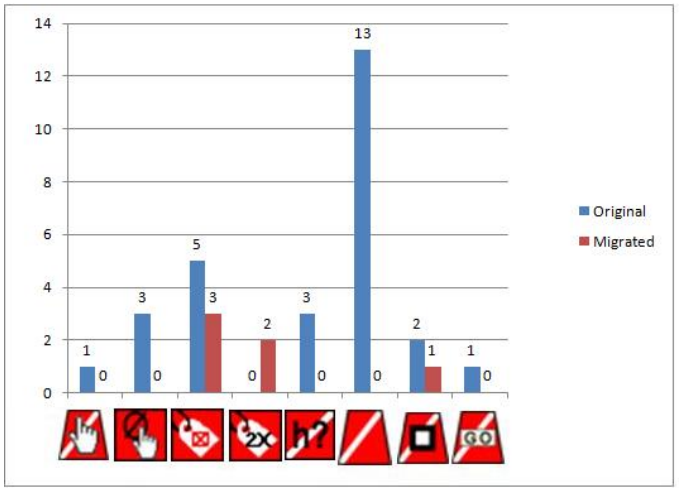


Figure 81 you can see the type of errors recorded before and after migration on the New York Times page: they range from errors like linked image missing alternative text, to missing alternative text, to multiple form labels, to empty heading, etc.

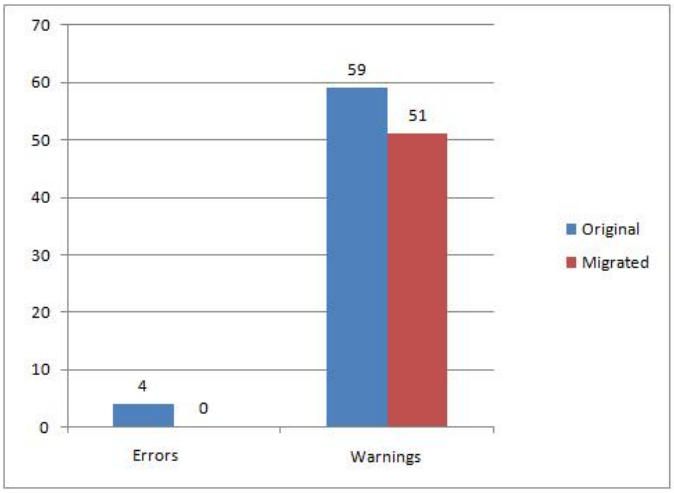


Figure 82 Results of WAVE evaluation on ESA page

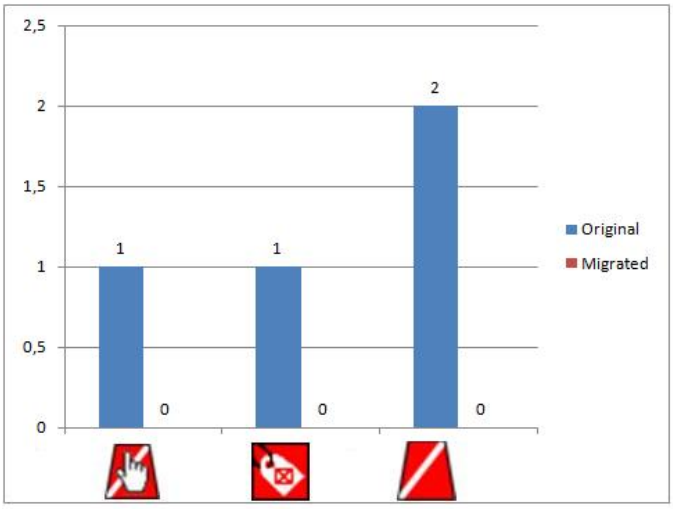


Figure 83 ESA - WAVE Error Details

Regarding the ESA page (see

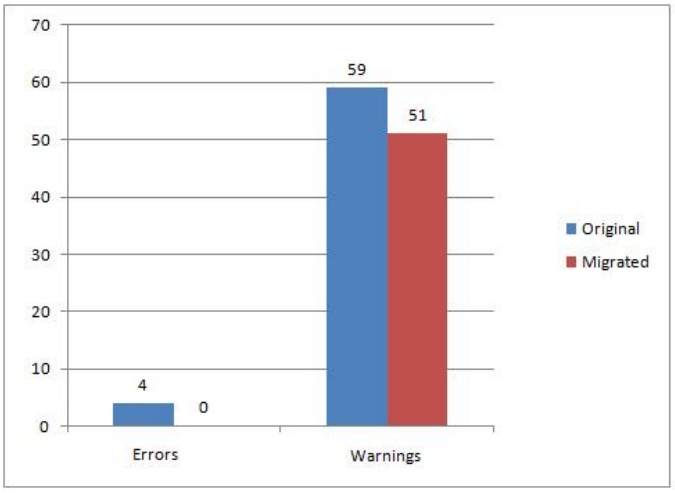


Figure 82), the original page had only 4 errors. The 4 errors of the original ESA page disappeared after migration (see

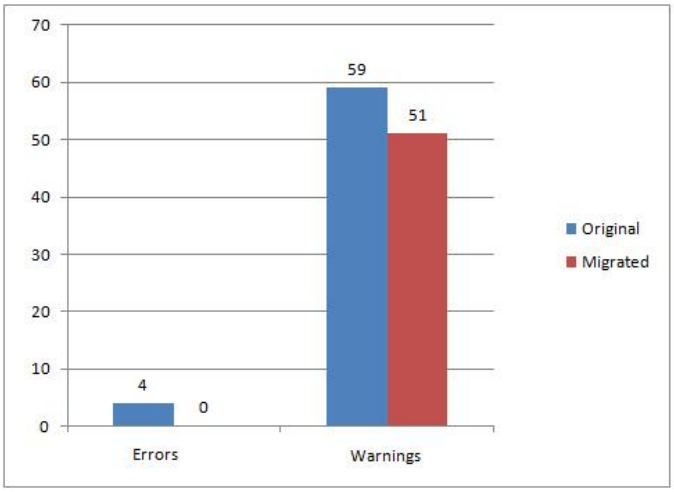


Figure 82 and

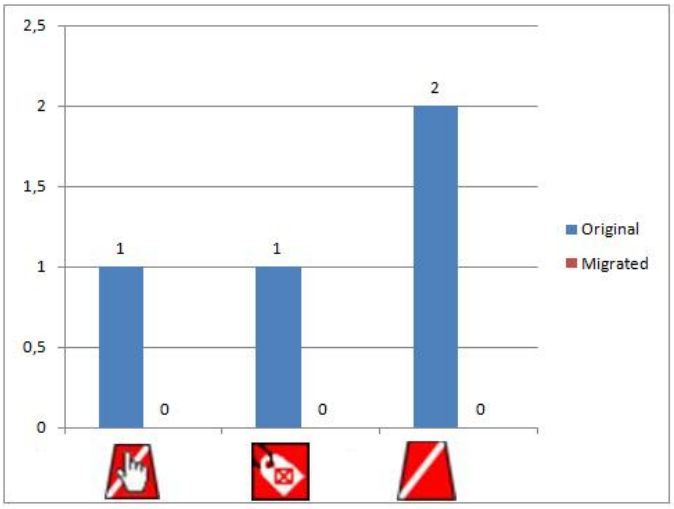


Figure 83 for details). In addition, after migration, there was also a reduction in the number of warnings/alerts (see

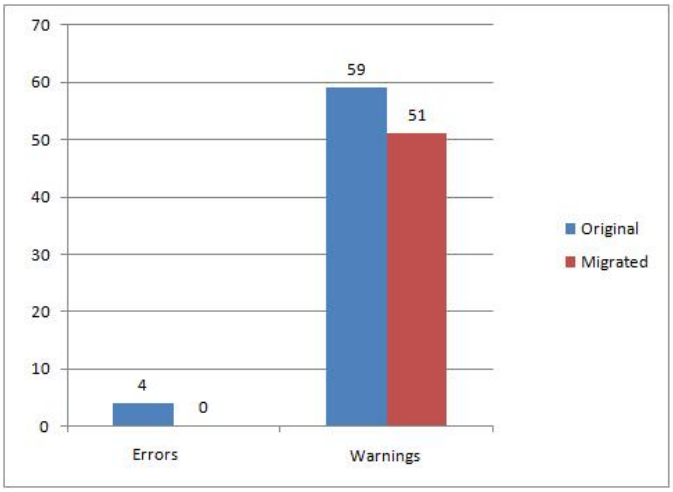


Figure 82).

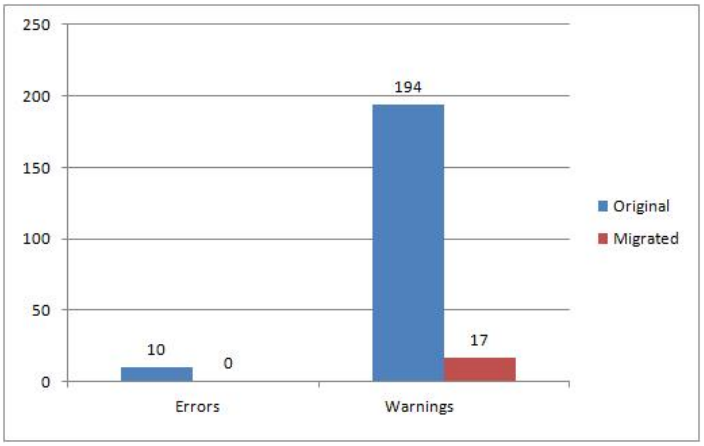


Figure 84 Results of WAVE evaluation on Weather.com page

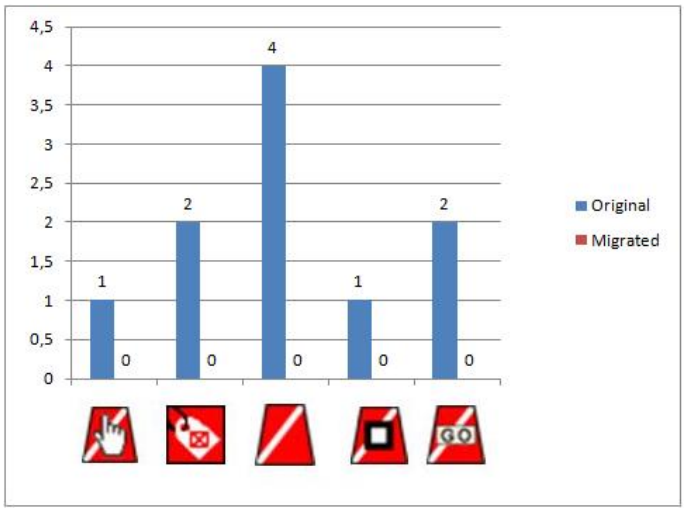


Figure 85 Weather.com – WAVE Error Details

Regarding the Weather.com page (see

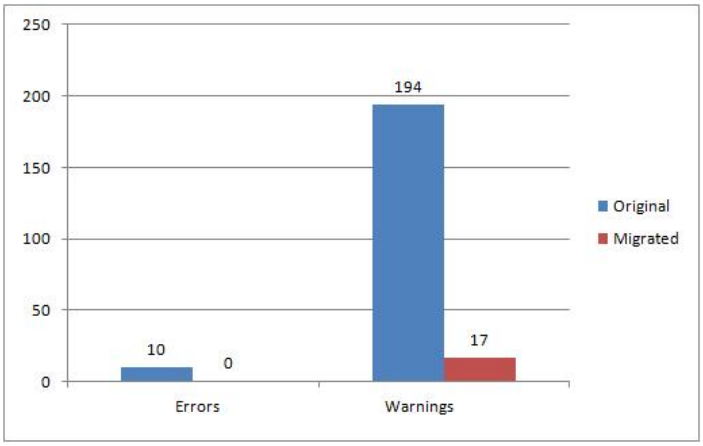


Figure 84), the original page had 10 errors. Such 10 errors were occurrences of the types of errors shown in

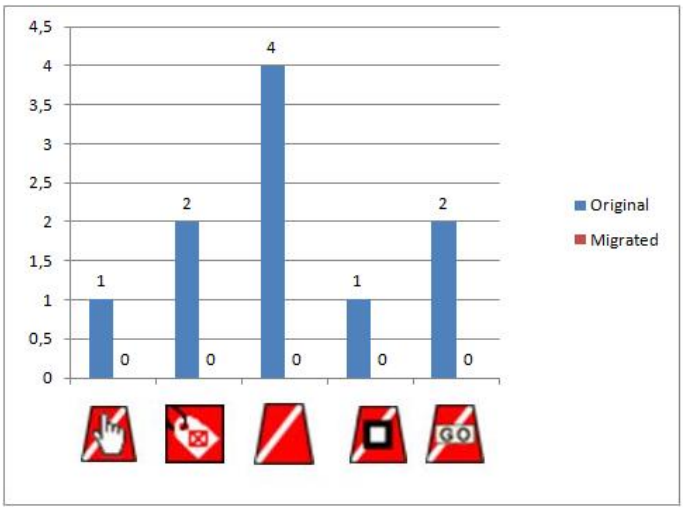


Figure 85. As a result of migration, there also was a substantial reduction in warnings (from 104 to 17, see

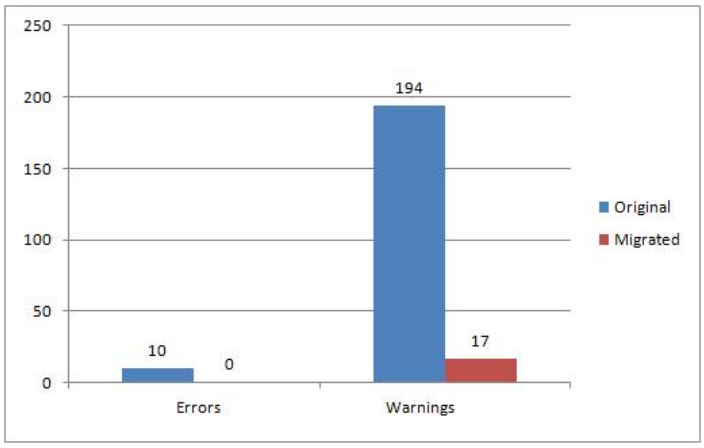


Figure 84). In

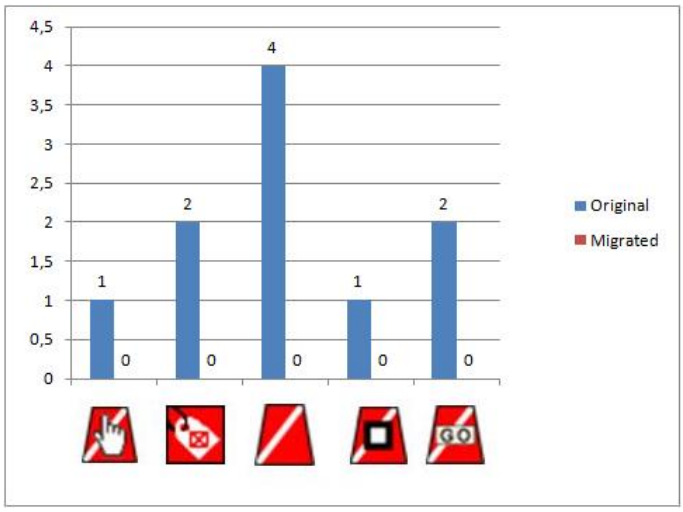


Figure 85 you can see a detail of the errors (before and after migration).

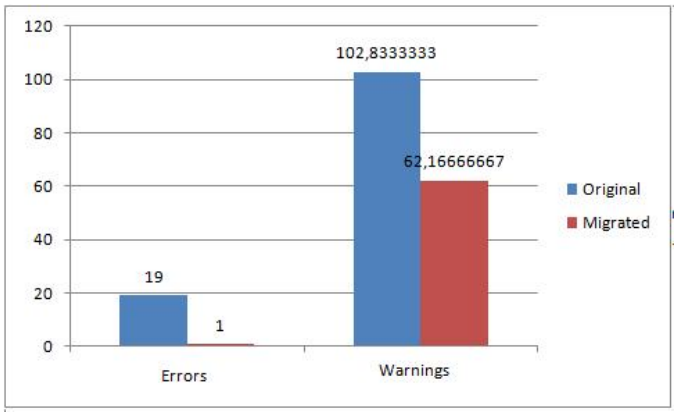


Figure 86 Mean values for WAVE on considered pages

In order to provide a summary view for WAVE tool on the considered pages, the mean values for both errors and warnings for original and migrated pages were computed. The result is shown in

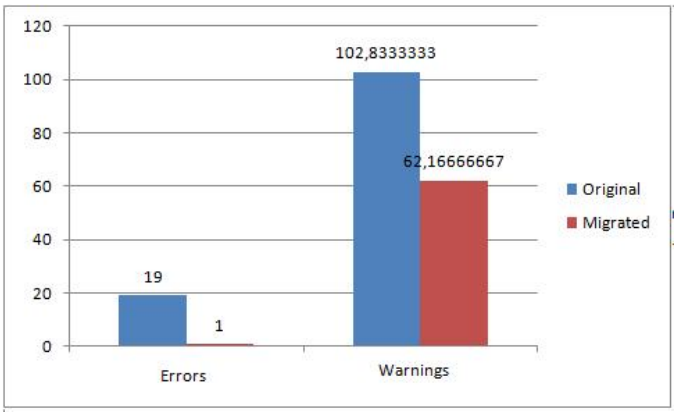


Figure 86. There was a reduction in errors (from 19 to 1 on average) and also in warnings (from 102.83 to 62.16, on average), which is a result that shows that the migration process improves the pages from the point of view of accessibility.

4.3.5. SCALABILITY

The scalability of a system indicates its ability to either handle growing amounts of work in a graceful manner or to be readily enlarged.

It makes sense to measure this indicator when a product is ready to be commercially distributed and used. The last version of the OPEN MSP is a stable prototype, but no effort has been addressed to the performance in term of number of clients simultaneously supported. For this reason, neither in this second testing iteration, it makes sense to evaluate the scalability indicator. A basic idea about performances is reported above, presenting the time requested by the OPEN Migration Service Platform to perform a migration considering different kind of on-top applications.

4.3.6. SECURITY

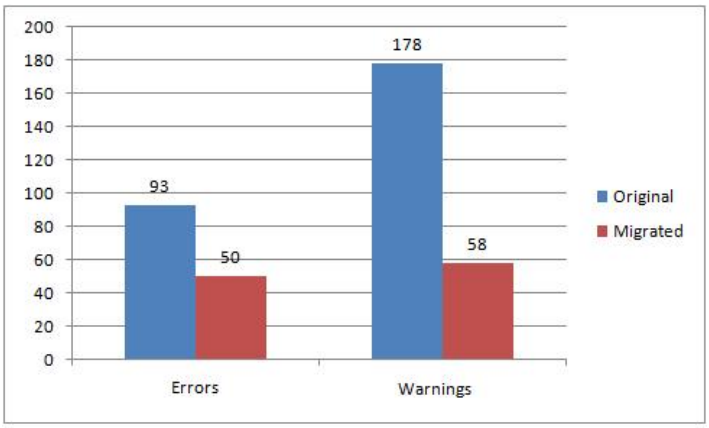
The security of a system refers to the capability of the system to operate in a secure manner. In the scenarios in which OPEN could be used, this means that the communication among different application parts and the OPEN MSP should be encrypted. This is necessary in order to guarantee to the users that their personal and private data do not travel over the network that not necessarily is private.
The final prototype of the OPEN Migration Service Platform does not implement any security mechanism at application level, but uses security features of lower level. For this reason, the security features are not properties of the platform, but more correctly they constitute pre-requirements for the networks on which the platform runs. Thus, at this final testing stage the security aspect is not considered during the indicator evaluation.

4.3.7. ADHERENCE TO THE STANDARDS

In the following paragraph the adherence to the standard is evaluated considering both the W3C validation for HTML pages and also the XML-RPC protocol implemented by the platform for inter-modules communications.

4.3.7.1. W3C VALIDATION

The adherence to the standards refers to the property of the platform to use standard protocols and languages for the information representation, where possible. As already done in D6.5 [D6.5] the HTML validation process for the adapted web pages is considered and replicated. In this way the adherence to the HTML standard language is verified. In order to do this, a set of pages with different HTML component such as images, tables, links, etc. is considered.
The pages considered for the W3C validation are the same considered in accessibility evaluation, reported in Table 22. W3C Markup Validation Service Validator is a free online tool available at [W3C-10] which checks the markup validity of a page. Through this free service is possible to check the validity of a web page by either submitting a URI, uploading a file or directly inputting the HTML code of a page.
Among the options provided by the W3C Validator there is the possibility to select the DOCTYPE according to which is requested to check the validity of the syntax of the considered page. Regarding this point, in the test it is considered, both for the desktop page and the migrated page, the DOCTYPE existing in the original desktop page and automatically recognized by the tool.
As a result of submitting a page under the W3C Markup Validation Service tool, a report is produced, in which both errors and warnings are shown and it is also possible to group together the errors by category.



Regarding the BA page (see Figure 87), there was a substantial reduction in both errors and warnings in the pages produced after migration. Both versions of the pages (before and after migration) were validated with respect to XHTML 1.0 Transitional.

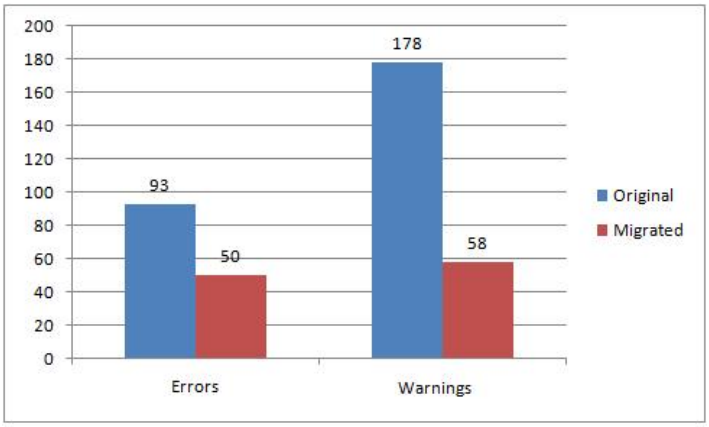
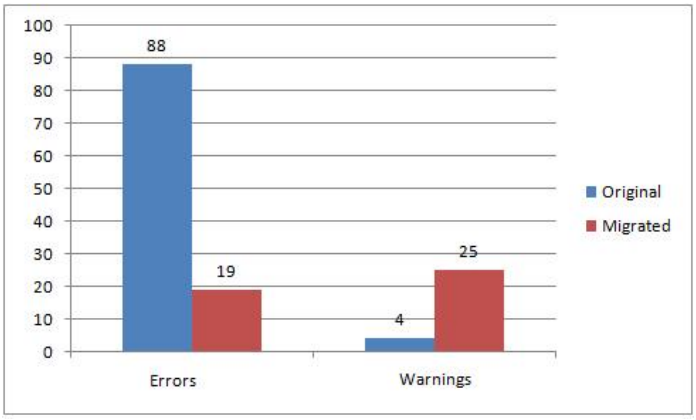


Figure 87 British Airways

Regarding the FIFA News page (see Figure 88), there was a neat reduction in errors, while the number of warnings was a bit higher in the migrated page with respect to the original page. This can be explained with the fact that some errors present in the original pages were not completely removed, but contributed to



increase the number of warnings. Both versions of the pages (before and after migration) were validated with respect to XHTML 1.0 Strict.

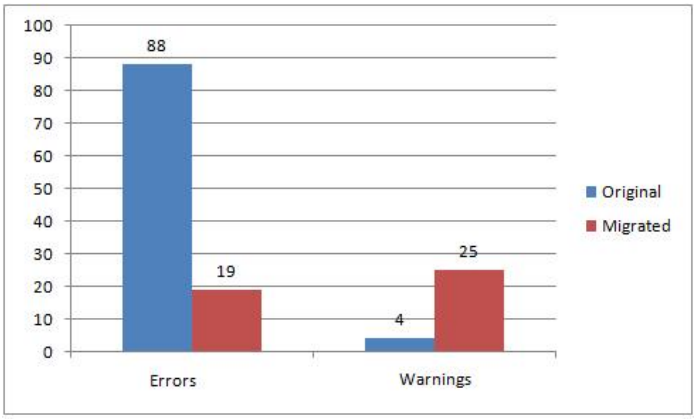


Figure 88 Fifa News

Regarding the Google Advanced Search page (see

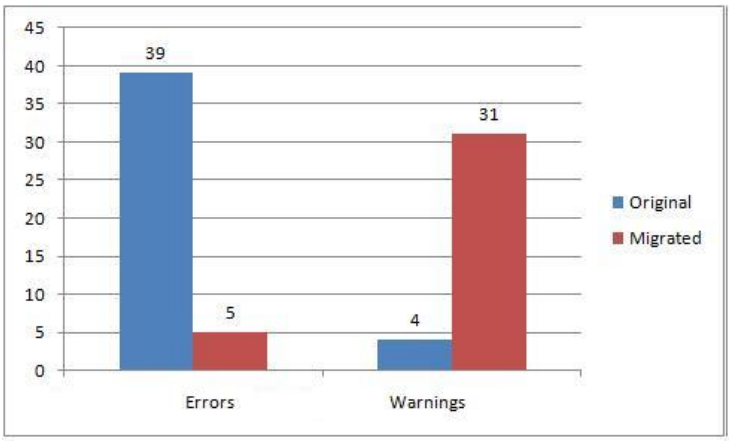


Figure 89), there was a neat reduction in errors, while the number of warnings resulted higher in the migrated page with respect to the original page. Both versions of the pages (before and after migration) were validated with respect to HTML 4.01 Transitional.

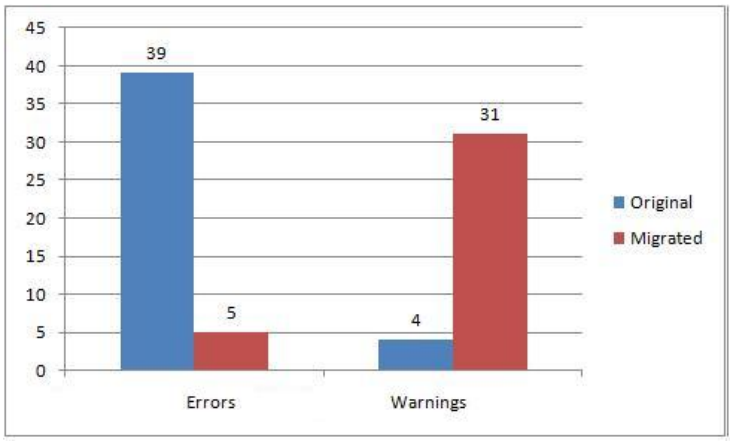


Figure 89 Google Advanced Search

Regarding the New York Times page, (see

Figure 90), it is reported again a substantial reduction in errors (more than a half of the original number of errors was cancelled in the migrated page, while the number of warnings resulted higher in the migrated page with respect to the original page. Both versions of the pages (before and after migration) were validated with respect to HTML 4.01 Transitional.

Figure 90 New York Times

Regarding the ESA page, (see Figure 91), it was reported again a radical reduction in errors (from 92 to 20) in the migrated page, while the number of warnings resulted higher in the migrated page with respect to the original page. Both versions of the pages (before and after migration) were validated with respect to HTML 4.01 Transitional.

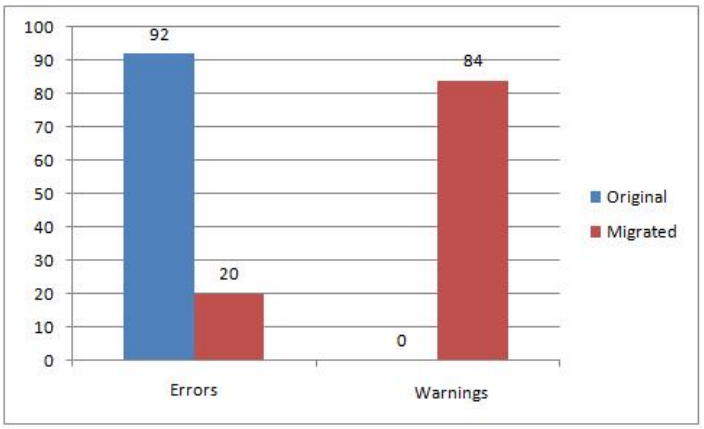


Figure 91 ESA

Regarding the Weather.com page, (see

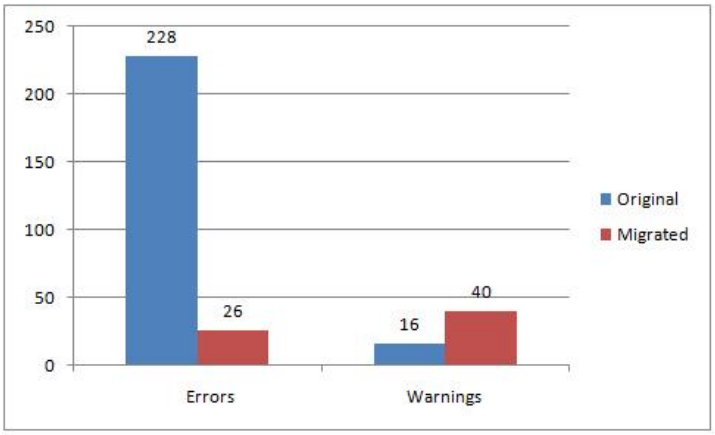


Figure 92), it is reported a clear reduction in errors (from 228 to 26) in the migrated page, while the number of warnings resulted a bit higher in the migrated page with respect to the original page. Both versions of the pages (before and after migration) were validated with respect to HTML 4.01 Strict.

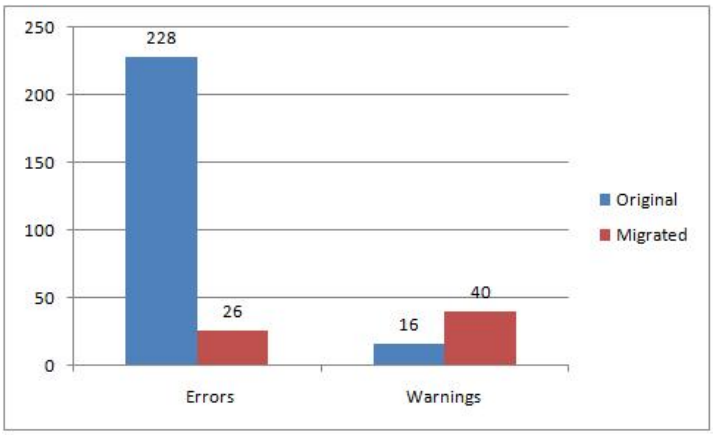


Figure 92 Weather.com

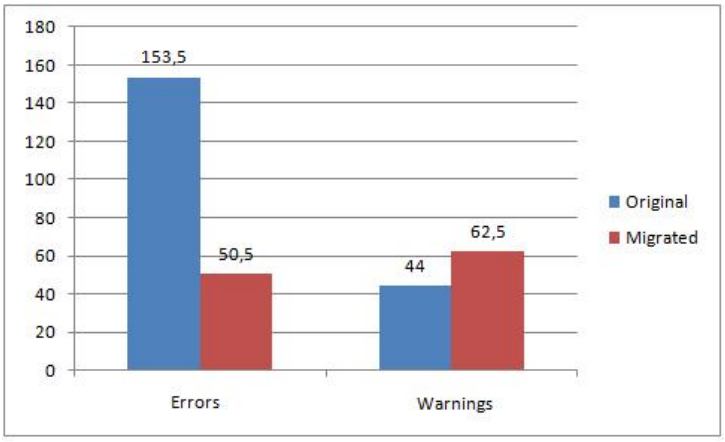


Figure 93 Mean Values for W3C Validator for considered pages

Again, in order to provide a summary view for W3C Validator tool on the considered pages, the mean values for both errors and warnings for original and migrated pages were computed. The result is shown in

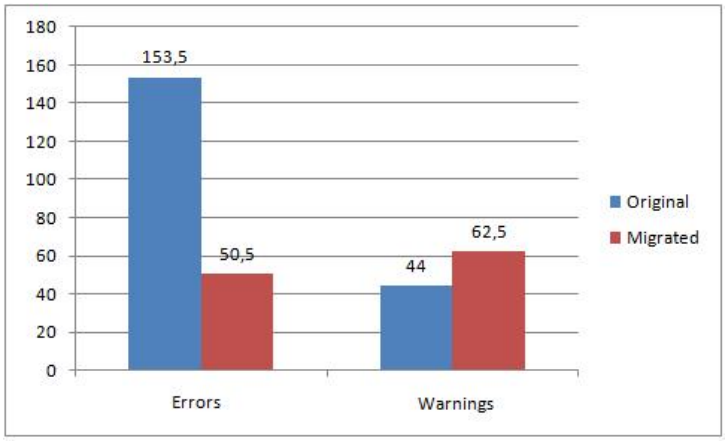


Figure 93. There was a reduction in errors (from 153.5 to 50.5 on average) while warnings a bit increased (from 44 to 62.5, on average), which in overall is still a result that shows that the migration process improves the pages from the point of view of the validity of the markup HTML code. In overall, the figures show good results and demonstrate an improvement in the resulting migrated pages considering validity aspects.

4.3.7.2. XML-RPC STANDARD

In this final testing iteration also the communication protocol among the OPEN modules and between the OPEN Migration Service Platform and the on top running applications is validated. The OPEN documentation declares the XML-RPC as communication protocol. Since XML-RPC is a standard protocol that runs over HTTP protocol without any encryption, the idea is to verify through a network packet inspection that the declared XML-RPC protocol is actually used. The packet inspections are carried out with Wireshark randomly sniffing the network during the availability and reliability test execution. The detailed results are reported in Table 24 and Table 25. The adherence to the standard application is tested using the platform and two different on top applications: Social Game and Emergency. While the Social Game communicates with the

Orchestration directly through the XML-RPC, the Emergency application contacts the Orchestration through the browser and a specific servlet that redirect the XML-RPC listener.

Adherence to the standard Test * XML-RPC protocol [Social Game]	
Description	Adherence to the XML-RPC standard is monitored recording TCP/IP packet and analyzing its corresponding payload.
Test	Follow TCP stream on the network on which are plugged the OPEN server and clients.
Involved devices	1 OPEN Server. 2 different OPEN clients.
Migratory applications involved	Social Game
Obtained Output	POST /xmlrpc HTTP/1.1 Content-Type: text/xml User-Agent: Apache XML RPC 3.0 (Sun HTTP Transport) Content-Length: 229 Cache-Control: no-cache Pragma: no-cache Host: 192.168.1.5:8989 Accept: text/html, image/gif, image/jpeg, *; q=.2, */*; q=.2 Connection: keep-alive <?xml version="1.0" encoding="UTF-8"?> <methodCall xmlns:ex="http://ws.apache.org/xmlrpc/namespaces/extensions"> <methodName>OrchestratorServer.unregisterApp</methodName> <params> <param> <value>1</value> </param> </params> </methodCall> HTTP/1.0 200 OK Content-Type: text/xml; charset=ISO-8859-1 content-length: 138 <?xml version="1.0" encoding="UTF-8"?> <methodResponse> <params> <param> <value> <boolean>1</boolean> </value> </param> </params> </methodResponse>
Observations	Here are reported the results of a TCP stream followed inspecting the network packets. Please note that the messages use the XML-RPC protocol over HTTP as declared in the documentation. Moreover it is possible to identify from the obtained XML that this communication sketch refers to the invocation of the

	unregisterApp method on the OrchestrationServer from the Social Game application.

Table 24 Adherence to the standard for XML-RPC protocol test description using Social Game application

Adherence to the standard Test * XML-RPC protocol [Emergency]	
Description	Adherence to the XML-RPC standard is monitored recording TCP/IP packet and analyzing its corresponding payload.
Test	Follow TCP stream on the network on which are plugged the OPEN server and clients.
Involved devices	1 OPEN Server. 2 different OPEN clients.
Migratory applications involved	Emergency
Obtained Output	POST /OpenServerServlet/xmlrpc HTTP/1.1 Accept: */* Content-Length: 415 Content-Type: text/xml Accept-Encoding: gzip, deflate User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322; .NET CLR 2.0.50727) Host: 192.168.1.5:8080 Connection: Keep-Alive Cache-Control: no-cache <?xml version="1.0" encoding="utf-8"?> <methodCall> <methodName>OrchestratorServer.registerDevice</methodName> <params> <param> <value> <struct> <member> <name>hasName</name> <value> <string>FloodManager</string> </value> </member> </struct> </value> </param> </params> </methodCall>
Observations	Here are reported the results of a TCP stream followed inspecting the network packets. Please note that the messages use the XML-RPC protocol over HTTP as declared in the documentation. Moreover it is possible to identify from the obtained XML that this communication sketch refers to the invocation of the registerDevice method on the OrchestrationServer from the Emergency application.

Table 25 Adherence to the standard for XML-RPC protocol test description using Emergency application

4.4. MODULES PERFORMANCES EVALUATION

For the module performances evaluation the focus is on modules that mostly influence the platform performances, e.g.: modules which behaviors are most impacted by platform operative conditions. The selected modules are:

- CMF
- ALR
- Web Migration Support which consists of WebUIAdaptation module that manages UI adaptation features together with other OPEN components that provides the state continuity.

Following in Section 4.8 a multi-network analysis is carried out considering the OPEN MSP coupled with Mobility Support module necessary to enable the multi-network application migration.

4.5. APPLICATION LOGIC RECONFIGURATION (ALR)

The Application Logic Reconfiguration (ALR) module is the core module that executes the logic application adaptation according to the context in which the application runs. Here after a detailed description of the ALR its evaluation is reported. First the description of key performance indicators for this module evaluation is shown, and then some results recorded in a real scenario are commented and analyzed.

4.5.1. PARAMETERS

The ALR module performs the reconfiguration when it is triggered for instance by the Context Management or by the registration of a new component. The reconfiguration time depends on the application instances which are registered at the ALR. To evaluate the logic duration two instances of time are measured:

- t_0 = the trigger of Application Logic Reconfiguration
- t_1= the Application Logic Reconfiguration done

The adaptation duration is defined as:

$$adaptation_{duration} = t_1 - t_0$$

The measurement is done in several steps. The number of application logic instances is increased at every step. A complete possible output of the described methodology is reported in Figure 94.

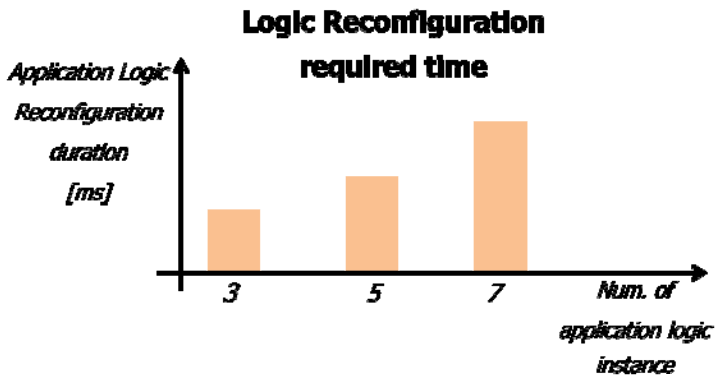


Figure 94 Logic Reconfiguration required time

According to the modality that can start the re-configuration it is important to remember that the ALR module has the task to (re-)configure the application based on two events:

1. A new component instance registers or unregisters at the ALR module during application runtime.
2. The context in which a component is working changes.

In both cases ALR has to reconfigure the application based on the application descriptor deposited by the application developer at the ALR module. The result analysis distinguishes between these kinds of reconfiguration.

To perform the test the ALR within the biathlon prototype previously described in Section 3.3 is used.

4.5.2. RESULTS ANALYSIS

In order to test the ALR module, a scenario which is able to evaluate the performance of the ALR module for typical reconfiguration scenarios during migration was developed. The setting consists of two components, one that provides an interface, and one that requires that interface. The task of the ALR module is to bind instances of these components together according to their availability and context.

The example application called Training Monitoring consists of two components, one representing a trainer, and one representing an athlete. The *AthleteComponent* is responsible for collecting training-relevant data like heart beat rate, current speed, training time and so on. It provides the interface *AthleteIf*. The *TrainerComponent* retrieves the information from all available *AthleteComponents* which are in the same team than the trainer. This component requires *AthleteIf* and provides *TrainerIf*.

In this test scenario the ALR module has the task to (re-)configure the application based on two events:

1. A new *AthleteComponent* instance registers or unregisters at the ALR module during application runtime (registration-triggered)
2. The team membership of an *AthleteComponent* instance changes its position (context-triggered).

In the following, the performance tests results are introduced.

4.5.3. PERFORMANCE OF REGISTRATION-TRIGGERED RECONFIGURATION

Each time a new component instance is registered at ALR the module has to check whether it has to integrate that component into the running application based on the specification given by the application descriptor. That could result in a replacement of an already running component, by adding the component to the running application, or by ignoring the new component in case its current context is not suitable for the application. If ALR decides to integrate the new component, the bindings between components have to be updated.

In the first setting it is assumed that an instance of *TrainerComponent* is registered at ALR. Afterwards one *AthleteComponent* instance after another is started, up to 20 instances. The time between registering the new component and the start of the new component is measured. This is the time that ALR needs to determine that the new component should be integrated into the application and to actually start that component. This procedure is repeated 20 times in order to get reasonable data.

The test is performed with two settings. In the first setting, all components together with the ALR module run on one device like depicted on the left hand side in Figure 95. In the second setup, the *AthleteComponents* are running on a remote machine, like shown on the right hand side of Figure 95.

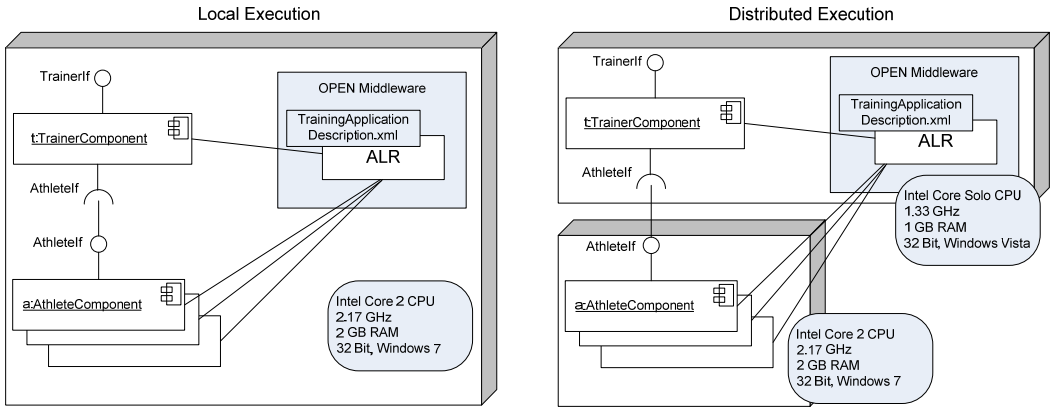


Figure 95 The two deployment settings for testing the registration-triggered reconfiguration performance.

The *TrainingApplicationDescription.xml* file specifies that an *AthleteComponent* instance should be connected to the *TrainerComponent*, if it represents an athlete which is in the same team. Both interfaces (*TrainerIf* and *AthleteIf*) define a method called *getTeamName() : String* which returns the team name of the according component instance. If a new component is registered at the ALR module, the module first has to check whether it is a Trainer or Athlete component. Then it has to check, whether the athlete is in the same team as the trainer by calling the method *getTeamName()* on both instances. If both are in the same team, the athlete has to be connected to the *TrainerComponent* and finally the new *AthleteComponent* has to be started.

The results for the local and remote case are shown in Figure 96.

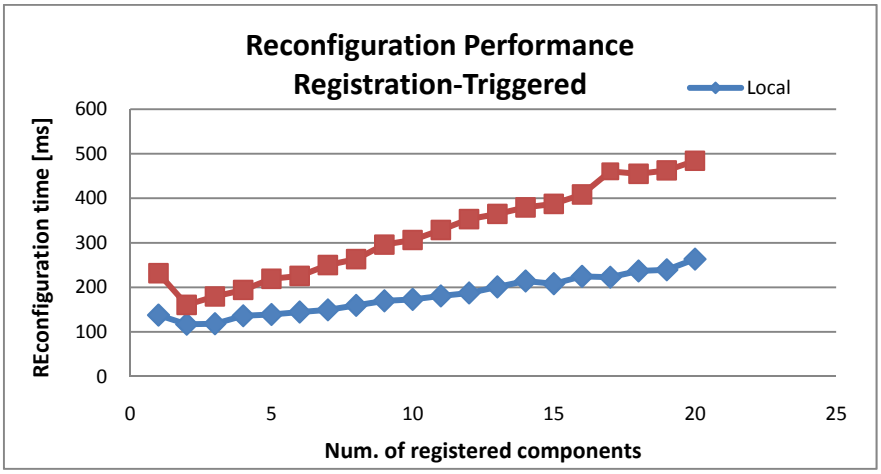


Figure 96 Registration-triggered reconfiguration performance.

It shows that the reconfiguration time depends on the number of components already registered. The registering of the first component takes in both setups (local and remote) much more time than for integrating the following instances. The reason for this is that the first *AthleteComponent* instance not only triggers the integration of that component, but also the execution of the *TrainerComponent*. This is because the application descriptor specifies that at least one athlete has to be connected to a *TrainerComponent* instance in order to start that component. The results show also that the reconfiguration time is strictly proportional in the remote and local case, while the gradient in the remote case is higher.

4.5.4. PERFORMANCE OF CONTEXT-TRIGGERED RECONFIGURATION

Another task of the ALR module is to perform reconfiguration triggered by the change of the context of single components. In this case, the context change by changing the team membership of an athlete during application runtime is simulated. To do this, first a *TrainerComponent* instance is registered and afterwards an *AthleteComponent* which represents an athlete which is not in the same team as the trainer. That means the ALR module already knows the instance, but decides not to integrate it into the application. Then the team membership during runtime is changed and the time is measured that the ALR needs to recognize that the component now should be integrated and started. The result is shown in Figure 97.

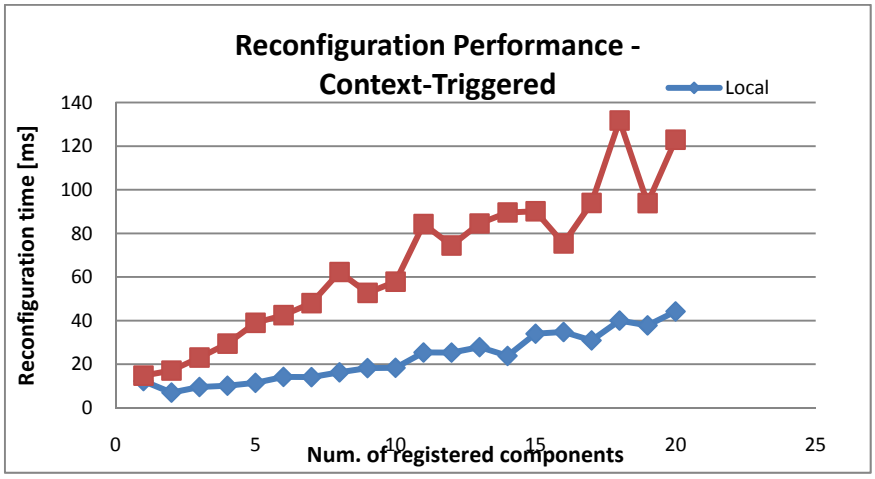


Figure 97 Context-triggered reconfiguration performance.

For the local and remote case (see Figure 95) the same deployment setup is used as introduced in the previous section. The absolute time required for reconfiguration is less than the time required during registration-triggered reconfiguration. The reason for this is that ALR already has all structural information about the component. Thus it knows that the component should be inserted if its context changes. What ALR does is to check the context of all registered components for their compliance with the current application iteratively. If it finds a registered component that became compliant, it integrates that one into the application. In both setups (remote and local) there is a quasi-linear proportional growth of the reconfiguration time based on the components already managed by the ALR.

4.6. WEB MIGRATION SUPPORT

This section describes a test bench used to evaluate the performance of the Web Migration support. It consists of WebUIAdaptation module that manages UI adaptation features together with other OPEN components that provides the state continuity.
The aim of the test is to get quantitative indications on the efficiency of the Web Migration Support, in order to understand whether there are weaknesses and to indicate the parts that can be improved. The Web Migration is performed through many steps and involves several sub-modules.

4.6.1. OVERVIEW OF THE DESKTOP TO MOBILE MIGRATION

- The user navigates a web site via the **migration proxy**, that for each page of the site:
  - gets the original page from the application server and parses it to obtain the document
  - modifies the document references (e.g., links to other files)
  - annotates the document by adding the scripts for managing the migration process
- The user, from the **migration client**, selects the target device for migration
  - the migration client sends a migration request to the Web Migration Server
  - the Migration Server unlocks the Ajax call done by the annotated page

  - the annotated page sends to the Migration Server the current source document together with the state of the forms and of the JavaScript
- In the Migration Server, the **State Mapper** maps the state of the forms in the source document and passes the document with state to the Reverse
- The **Reverse** creates a logical description of the source page by:
  - parsing the html to get the document
  - analyzing the style sheets
  - extracting the JavaScript and saving it in a separate repository
- The **Semantic Redesign**, adapts the logical description of the source page for the target device by, eventually:
  - splitting the original presentation into two or more ones
  - scaling the images which do not fit into the target display
  - converting the interaction components (e.g., radio buttons into drop-down list)
- The **User Interface Generator** creates a web page for every presentation defined in the adapted logical description

4.6.2. TEST DESCRIPTION AND RESULTS

Ten pages have been involved in total migration from PC to mobile (see Table 26 for the actual addresses). The elaboration time for each module has been registered, as well as the main features of the page.

Site	URL
English Wikipedia	http://en.wikipedia.org
Amazon home page	http://www.amazon.com
Ebay Advanced Search	http://shop.ebay.com/ebayadvsearch/
BBC	http://www.bbc.co.uk
FIFA news	http://www.fifa.com/worldcup/news/index.html
Google advanced search	http://www.google.com/advanced_search?hl=en
United Nations documents index	http://www.un.org/en/documents/
W3C	http://www.w3.org
British Airways special offers	http://www.britishairways.com/travel/flight-deals/public/en_gb?link=main_nav
Vodafone	http://www.vodafone.com

Table 26 Web pages involved in the total migration from PC to mobile and their address

Table 27 summarizes the results of the migration process for the above listed web pages. The table also reports the main features of the pages (such as HTML structure) and some details on the adaptation process (such as the number of images scaled and of presentations generated).
In the following, a description of the parameters reported in the table is provided.

RELEVANT DOM NODES INDICATED THROUGH THE TAG NAME

- *src, url, href*: link to resources such as images or other pages
- *script*: node containing script code or referring an external script file
- *style*: node containing a style sheet or referring an external style sheet
- *input*: fillable / editable component
- *iframe*: an area containing another web page
- *ul / li / ol*: containers for formatting lists
- *div*: general purpose HTML container
- *form*: area to let the user send data
- *table*: container for two-dimensions formatting
- *img*: image container

STATE COMPONENTS

Number of components (e.g., elements of a form) for which the state is encoded and passed to the migration server.

HTML DEPTH

It is the maximum node nesting within the HTML document.

JAVASCRIPT SIZE (BYTES)

It is the total number of bytes of JavaScript code. This includes both internal and external (referred) code.

CUI DEPTH

It is the maximum node nesting within the logical description of the user interface.

SCALED IMAGES

It is the number of images for which the sizes have been reduced to meet the requirement of the target device.

PRESENTATIONS

It is the number of presentations that have been generated starting from the source page.

PROXY TIME

- *HTML parsing*: construction of the page document object, starting from the content of the HTML original page

- links modifying: manipulation of the links (src, href, url, …) contained in the document and in the referred resources (external style sheets, script files, …)
- *total (estimated)*: estimated time for the proxy. It is not an exact value since the proxy can be accessed in a non-deterministic way (after the first loading) if the page has dynamic content.

STATE MAPPER TIME

Time needed to map the state (content of the forms) of the source web page to the document to be reversed.

REVERSE TIME

- *HTML parsing*: construction of the page document object, starting from the content of the HTML page with state
- *CSS reading*: analysis of style sheets (including the download of the external ones)
- *DOM analysis*: examination of the page document to create the logical description (CUI)
- *Total*: the total time of the reverse module (it is bigger than the sum of the sub-modules, because it includes the file reading and writing)

SEMANTIC REDESIGN TIME

Adaptation of the logical description (CUI) of the page: text/presentation splitting and images scaling.

UI GENERATOR TIME

Generation of as many web pages as the number of presentations defined in the adapted logical description.

MIGRATION TIME

It is the total time taken by the entire migration process. It is bigger than the sum of the previously mentioned modules, because it includes the network latency related to the exchange of messages between client device and migration server regarding the migration.

			Wikipedia	Amazon	Ebay	BBC	FIFA	Google	United Nations	W3C	British Airways	Vodafone	(AVERAGE)
PAGE	RELEVANT DOM NODES	src	25	42	13	48	103	1	3	21	49	25	33
		url	0	8	0	0	0	0	0	0	0	0	0,8
		href	333	207	67	284	167	27	106	118	191	81	158,1
		script	14	36	22	43	16	6	3	1	26	14	18,1
		style	0*	9	1	0*	1	3	1	1	0*	2	2,6
		input	2	2	48	44	7	16	12	2	61	3	19,7
		iframe	0	2	0	0	0	0	0	0	0	0	0,2
		ul/li/ol	135	135	16	246	502	0	106	58	90	71	135,9
		div	61	327	198	190	250	21	29	62	222	134	149,4
		form	1	1	2	4	2	3	2	2	4	3	2,4
		table	9	17	8	2	1	14	0	0	0	1	5,2
		img	16	46	6	42	96	1	2	20	32	19	28
	state components		2	1	45	40	5	16	10	1	58	1	17,9
	HTML depth		22	24	25	19	16	23	11	18	23	26	20,7
	JavaScript size (Bytes)		6244	36855	21675	18200	40964	10936	1724	1084	13316	7298	15830
RESULTS	CUI depth		22	23	25	17	17	24	12	19	18	25	20,2
	scaled images		2	7	0	1	1	0	0	0	10	5	2,6
	presentations		3	2	2	2	2	1	1	3	2	2	2
TIME (milliseconds)	PROXY	HTML parsing	0,031	0,047	0,015	0,031	0,015	0,016	0,001	0,016	0,031	0,031	0,023
		links modifying	0,218	0,250	0,234	0,250	0,125	0,093	0,078	0,093	0,218	0,140	0,170
		total estimated	0,249	0,297	0,249	0,281	0,140	0,109	0,079	0,109	0,249	0,171	0,193
	State Mapper		0,250	0,375	1,578	1,875	0,328	0,156	0,219	0,141	2,110	0,203	0,724
	REVERSE	HTML parsing	0,140	0,250	0,094	0,344	0,125	0,032	0,062	0,078	0,156	0,187	0,147
		CSS reading	0,922	0,937	0,547	1,454	1,437	0,031	0,516	1,969	1,953	0,390	1,016
		DOM analysis	2,297	3,094	2,313	7,406	6,656	0,079	0,125	0,656	3,015	0,438	2,608
		total	3,812	4,324	3,609	9,891	8,844	0,719	1,281	3,297	5,718	1,672	4,317
	Semantic Redesign		5,688	6,796	8,594	6,078	5,031	0,891	1,906	7,219	11,969	8,093	6,227
	UI Generator		1,297	1,094	0,765	0,360	1,234	0,797	0,594	0,672	1,156	0,672	0,864
	migration time		14,782	17,234	17,640	26,422	21,562	10,219	5,578	22,204	22,640	13,422	17,170

Table 27 Features of the pages involved in the total migration and time needed. * Even without "style" elements, a page can refer one or more style sheets through the attribute "href" within "link" nodes

Table shows that the results of the Web Migration are already satisfying because the total migration time for the pages considered varies between 5 and 26 seconds, which is a reasonable time considering also that in the meantime the user has to change device (and position with the new device).

By analyzing the overall migration time with respect to the characteristics of the source page, it is rather difficult to find a direct correlation: in general there is no proportionality between the page complexity (such as number of links/images/forms, document depth, HTML containers, etc.) and the migration time.

The only remarkable value is related to the BBC case, which is the most time consuming (over 26 s). In particular, the large amount of scripts (43) seems to affect the Reverse total time.

In general, the average time of the main modules (Proxy, State Mapper, Reverse, Semantic Redesign and UI Generator), indicates that the most time consuming modules are the Semantic Redesign and the Reverse (Figure 98).

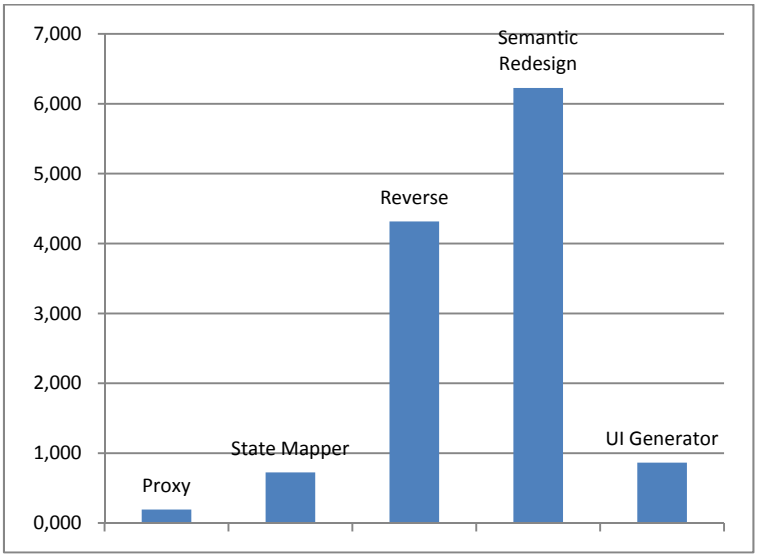


Figure 98 Average time in seconds for the different modules involved in the web migration

By considering the time of the composing (sub)-modules, it is easier to find which page parameter impacts more each sub-module. Some examples are provided in the following. The State Mapper time is correlated to the *number of components with state*. Such components are the HTML elements had the state modified on the source device, state that has to be associated with the new version of the page Ebay, BBC and British Airways, that have the highest number of state components, take also the highest time to be state-mapped (see Figure 99).

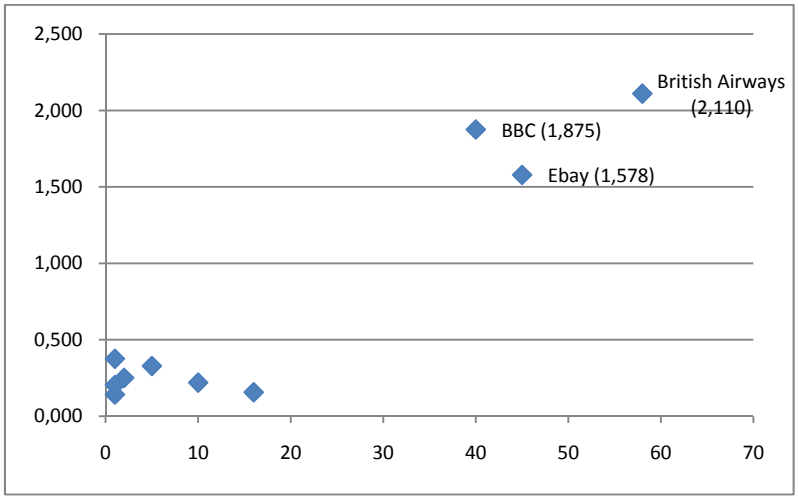


Figure 99 State Mapper time, in seconds, in function of the number of state components

The British Airways case, which is the most time consuming for the Semantic Redesign, is also the one with the largest number of images scaled (10). However, by looking at Figure 100, the correlation seems not so evident, since there are pages (such as Ebay and W3C) for which the Semantic Redesign takes a considerable time even without scaling any image. This could be explained by the relatively deep structure of the Ebay page (25 levels) and by the amount of splitting for the W3C page (3 presentations generated), which require some additional work to the Semantic Redesign, with respect to the other cases.

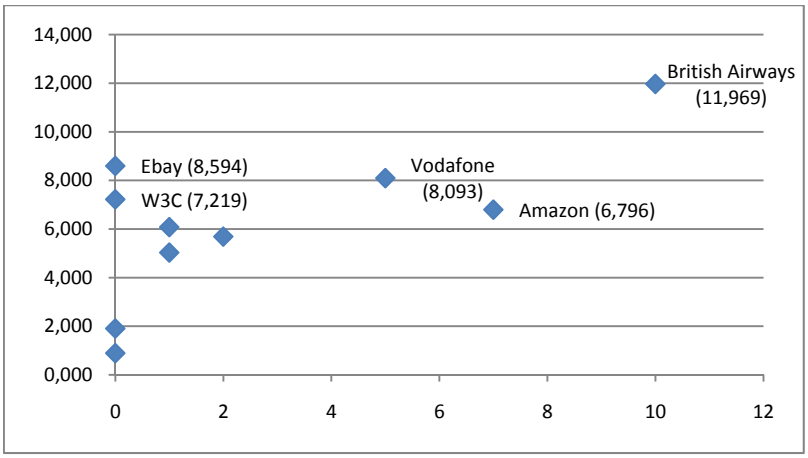


Figure 100 Semantic Redesign time, in seconds, in function of the images scaled

The Semantic Redesign time can also be observed according to the number of presentations generated in the mobile version dynamically created. Even if there is not any proportionality between the presentations and the time, it can be noticed that the simplest cases (Google and United Nations) took much less (see Figure 101).

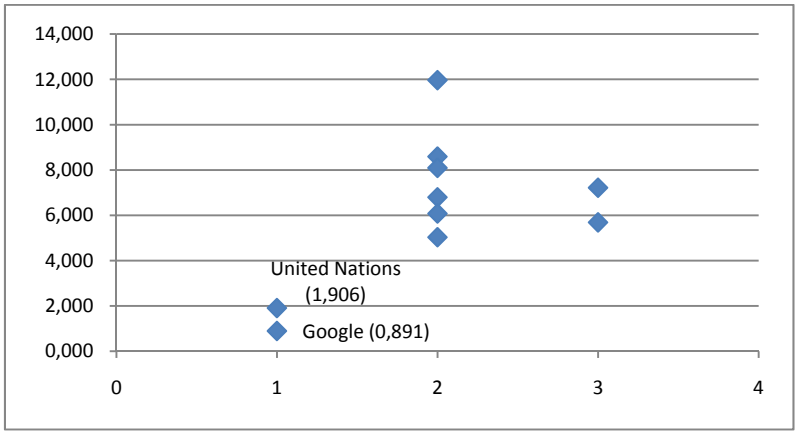


Figure 101 Semantic Redesign time, in seconds, in function of the presentations generated.

The *number of script nodes* is somehow related to the Reverse total time. The relation is more evident if the DOM analysis time is considered (which is the main phase of the Reverse). Figure 102 shows a comparison between total time and DOM analysis time.

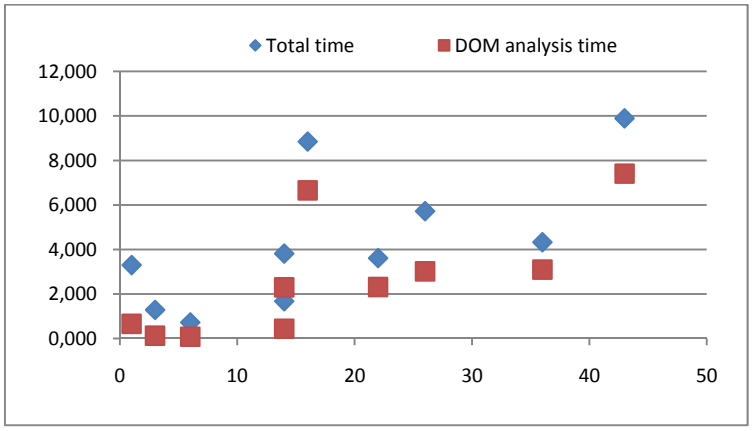


Figure 102 Reverse total time and DOM analysis time, in seconds, in function of the number of script nodes in the page

4.7. CONTEXT MANAGEMENT EVALUATION (CMF)

The main purpose of the Context Management Framework is to allow easy access to distributed, dynamic context information for applications and system components.

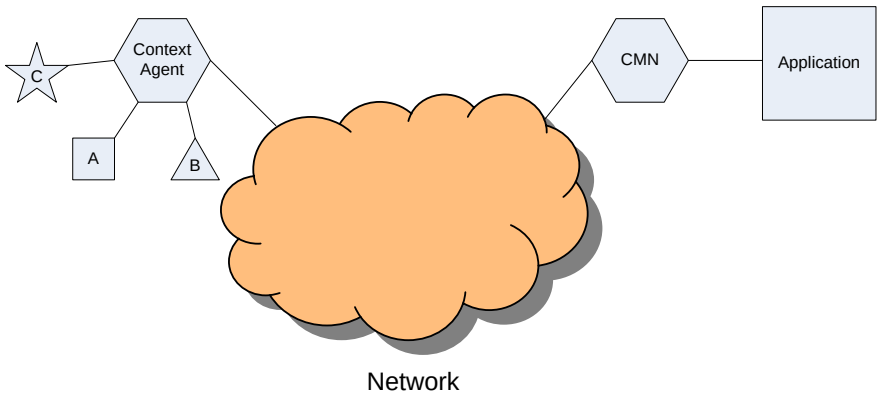


Figure 103 Conceptual setup for performance evaluation of the Context Management Framework

Figure 103 shows two Context Agents, which are connected via a network, one configured as the Context Management Node (CMN), and one which provides the context information used for the test, referred to as the Context Agent in the following. Thus, as seen in the Figure 103, the Context Agent has access to a set of information elements called: A, B and C, which are required by the application accessed via the Context Management Node.

4.7.1. RELEVANT PERFORMANCE METRICS

The relevant performance metrics differs from the various access strategies taken. Thus, their definitions are divided accordingly to the chosen strategy;

a) reactive;
b) proactive event driven;
c) proactive, periodic updates.

4.7.1.1. REACTIVE STRATEGY

Figure 104 shows how the reactive strategy ensures the information availability at the Content management Node (CMN), by sending a request to the Context Agent, and then uses the received response.

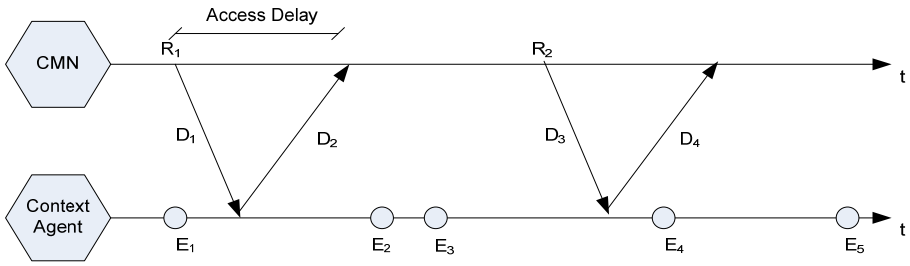


Figure 104 Reactive access strategy with some test event process at the Context Agent

ACCESS DELAY

For the reactive strategy the access delay is defined as the time at which a request is sent until the corresponding response is received:

$$t_{access} = t_{response} - t_{request}$$

Access delay (t_{access})is always measured locally at the node where the access is being made, i.e. the Context Management Node by marking sending and receiving times and logging these for later evaluation.

NETWORK OVERHEAD

The amount of bytes/second generated depends on the request process, the size of the request and response messages. Focus is to measure the sizes of the request and response messages only, whereas the actual network overhead can easily be calculated.

MISMATCH PROBABILITY

The mismatch probability is defined as the probability of getting information at the local node, e.g. R_2 in Figure 104 leads to a mismatch, while R_1 does not. As showed in [BOE-10] this parameter depends on statistical properties of the event and delay processes. As the event process can be anything, the following analysis focus on results based on the delay distribution measured in conjunction with the event process described in D3.4 [D3.4] on the location (which was analytically derived). Notice, the results obtained in D3.4 [D3.4] are valid only for the reactive strategy, hence will only be used for this strategy.

The expressions for the various strategies are given in [BOE-10], with E as the event process (hereby E[E] denotes the first moment of the event inter time arrival distribution), F_d indicates the delay distribution, F_E the event time distribution and the over line indicates that it is the reliability distribution, i.e. 1-F(t) that is being used.

$$\text{mmPr}_{\text{Reactive}} = \text{mmPr}_{\text{Event}} = \frac{1}{\text{E}[E]} \int_0^\infty \overline{F}_D(t)\overline{F}_E(t)dt$$

4.7.1.2. PROACTIVE, EVENT DRIVEN UPDATE

The proactive access strategy depends on a subscription sent to the Context Agent, which then sends an update to the Context Management Node as events occur. When an event occurs at the Context Agent, it first sends a notification to the CMN, which then triggers a request/response pair in order to fetch the actual value of the information. The application will only get access to the updated information after the request/response message has been resolved. From an application point of view, the mechanism appears as a single message update as shown in the dashed lines in Figure 105. In the following, the term update is used as the single message concept, while it actually covers both a notification and a request/response interaction.

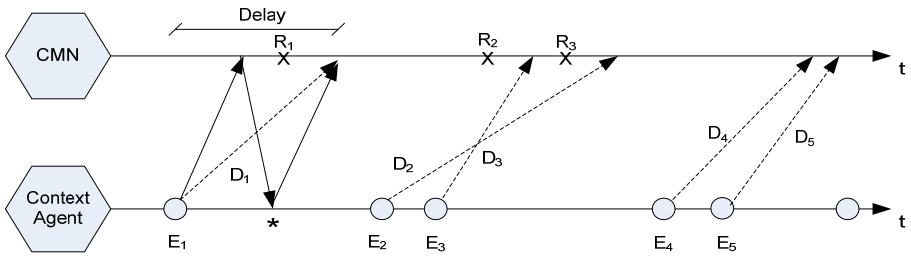


Figure 105 Proactive event driven update strategy, where each event triggers an update of the information.

ACCESS DELAY

The communication between application and CMN software entity is assumed to be zero, or at least insignificant compared to the network communication time. This delay is therefore characterized as the time difference between the time event E_x happens and the application has received the information (see Figure 105**Errore. L'origine riferimento non è stata trovata.**).

NETWORK OVERHEAD

Network overhead in terms of bytes/second, depends on the event process and the size of messages. The focus of the measurement here is the overhead size of single updates.

MISMATCH PROBABILITY

For the proactive, event driven case, a mismatch can happen in several ways.

i. if an application receives an update from the CMN while a new update from the latest event is still in transit, then this would lead to a mismatch, e.g. R_2 in Figure 105 leads to an error because of this.

ii. if the updates are sent incrementally, i.e. only differences since the last event are transmitted, then a mismatch happens if just one update is in transit, e.g. R_3 would be an example leading to such a mismatch, while it would be matching if the full information was sent at E_3.

Due to these definitions and the underlying notification/request/response approach, it is important to assume that once an event happens it is kept the same during the update process, i.e. if another event happens between the first event (e.g. E_1 in Figure 80) and the reception of

the fetch request (in **Errore. L'origine riferimento non è stata trovata.** marked by a '*'), then this does not alter the value of the initial event, e.g. E_1. This could be the scenario with E_4 and E_5. For this metric, statistical properties of the network delay and event process are the key impacting factors, see [BOE-10].
[BOE-10] concludes that the expression for the reactive and proactive, event driven update? are the same, however, as it will be shown later, due to the implementation nature the delay distributions are not the same, hence the resulting mismatch probability will become different.

4.7.1.3. PROACTIVE, PERIODIC UPDATE

The proactive, periodic update strategy is characterized by a subscription that established between the Context Agent and the CMN, in which the Context Agent sends updates to the CMN with specified time intervals. Similar to the proactive, event driven case, an update covers in fact a notification message followed by a request/response part that fetches the value of the updated information. This is done, again, transparently to the application.

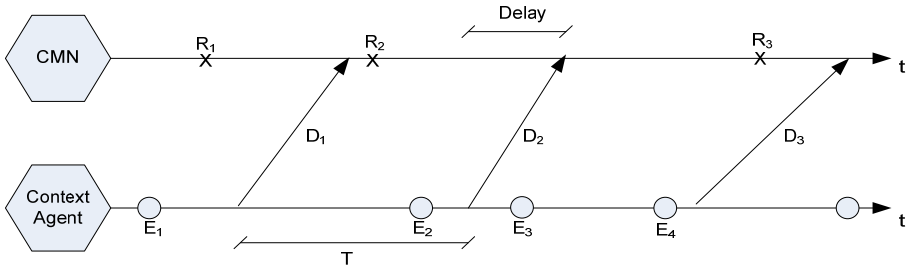


Figure 106 Periodic update strategy with a given time interval between updates send to the CMN from the Context Agent.

ACCESS DELAY

The same interaction patterns with the application as with the proactive, event driven case is also true for the periodic case, and the only difference is that updates are received periodically instead of per event. Therefore, the definition of access delay for this case is the same as for the proactive, event driven case.

NETWORK OVERHEAD

The network overhead in bytes/second is determined by the update time interval. Thus focus is on measuring a single update message, to evaluate the size of the update in bytes.

MISMATCH PROBABILITY

In this case, mismatches happen if:

1. the application gets the latest value received by the CMN, while an event has happened and no updates are currently in transit, i.e. an in-between-update situation. R_1 is an example of this scenario.

2. the application gets the latest value received by the CMN, while an event has happened and the latest updates are in transit. R_3 is an example of this scenario.

Beside the statistical properties of the event and delay distribution, the metric for this method also depends on the update time interval. Similar as with the proactive, event driven case, this metric is evaluated using measured delay distribution and then with various event process types (deterministic to stochastic) with various update intervals. See [BOE-10] for information of the actual calculation of this metric.

The expression for the mismatch probability is derived in [BOE-10] and is given as:

$$\text{mmPr}_{\text{Periodic}} = \int_0^{\infty} \exp(-\int_0^{t} \tau F_D(s)ds) F_E(dt)$$

It is important to mention here that the update time interval is required by the models in [BOE-10] to be random with the time intervals exponentially distributed, with the rate τ. A model for deterministic time periods have not yet been found, but from simulation studies, it can be shown that deterministic time intervals always produce a lower mmPr, hence the results obtained using the models in [BOE-10] lead to what can be said to be worst case results.

4.7.1.4. IMPACT FACTORS AND PARAMETERS

Beside the different access strategies and their respective parameters (e.g. the time interval for proactive updates or caching period for a reactive strategy with cache), the two main influencing factors on these performance metrics are delay and information dynamics.

DELAY

Whatever causing the delay between the response/update messages between the CMN and Context Agent, e.g. communication delay or processing delay, this has an impact on the mismatch probability and the access delay. Thus, it is interesting to evaluate what influences the delay, whereas only some are of interest to the overall performance study considered here:

- **Cross traffic in the network**: More cross traffic means longer waiting times (or even packet drops) in buffers along the network.
- **Size of information elements**: (the larger context information size, the more transport time is required)
- **Information dependencies**: (some context information is inferred from other elements that require additional information gathering that prolongs the access delay. The access delay thus depends on also potential dependencies).
- **Network type**: Wireless links are more often prone to errors at lower layers of the communication protocol stack, which means retransmissions are needed. This leads to longer delays at the higher level.

Since later will be proposed to investigate on the influence of Quality of Service on the context management, it makes sense to consider cross traffic as one parameter, first without QoS to create a comparison base.

As the exchange of context information is a part of the migration process, then the information volume of the context information can be considered as a part of the system state that needs to be exchanged. Therefore, this parameter is also chosen as input parameter.

Finally, the scalability in terms of accessible information types is of interest, due to its intended functionality relates to, in principle, an unbounded amount of information types.
All of the above are evaluated via a practical test bed that implements the network scenario as shown in Figure 107.

INFORMATION DYNAMICS

Depending on the strategy, the dynamics (i.e. how and with what time intervals the information changes its value) impacts the mismatch probability and potentially the network overhead. The dynamics is described by an underlying stochastic process, which structure and parameter setting depends highly on the actual information element. For example, in D3.4, this is analyzed for a mobility model describing it by a Markov Chain. In some cases there are links between the delay, the information dynamics and other performance parameters. E.g. in [D3.4] it is described the case of location information with respect to a mobile target that finds and evaluates the link between achievable accuracy of positioning estimate with respect to the mobility of the target and the processing delay required to achieve certain accuracies.
For the remaining part it is assumed at all times, an information type which assumes

- its value cannot attain the same value twice;
- the time interval value changes following a random exponentially distributed model.

4.7.1.5. EVALUATION SETUP AND PROCEDURES

For the evaluation of the context management framework, it is used the test bed setup shown inFigure 107. This allows controlling all the aspects of the network. In the following they are described this.

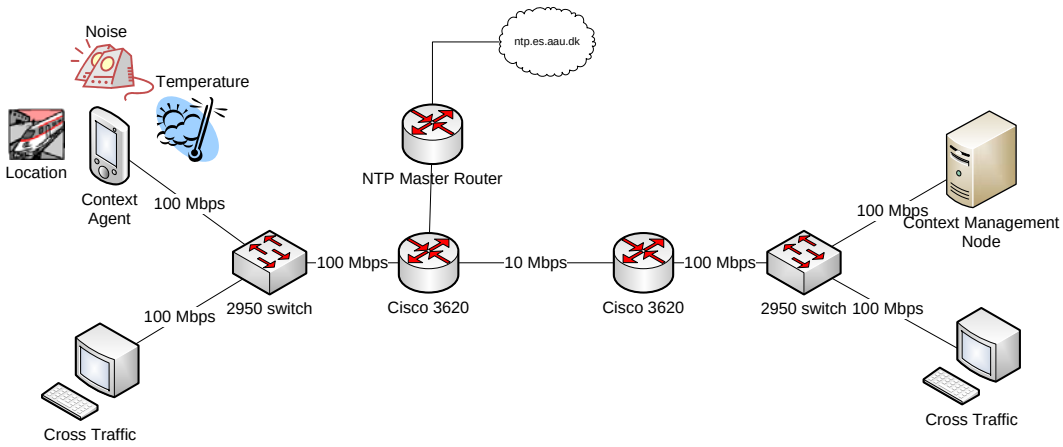


Figure 107 Overview of test bed for performance evaluation of the Context Management Framework.

To the left the Context Agent which measures some context information is located, e.g. the position, noise level or local temperature, and to the right there is the Context Management Node which needs to access context information measured at the Context Agent. The purpose why it needs the information, it is not considered further, but it could be a part of the migration evaluation process.

In the middle, there is a network consisting of a set of switches and Cisco routers. In between the two Cisco routers, it is established a resource constrained link at 10MB/s to obtain a bottleneck. Side by side at the CMF entities, two sources for traffic generation are also put. They generate constant bit rate data flows in order to get the network constantly occupied. Within the network there is the possibility to enable QoS traffic. With this setting, a series of tests has been carried out with and without QoS for comparison, and evaluate the resulting performance of the context management framework.
For each test 50 sequences are carried out. For the reactive strategy this means 50 request/responses are done, and for the proactive approaches 50 updates are received. The test scenarios vary the external traffic load from 80% - 97% throughout the test to evaluate the impact of cross traffic on the context management frameworks performance. From the 50 sample set, three metrics of interest are evaluated, namely, network traffic overhead generated; access/time delay and resulting mismatches probability. In the following the steps done for each metric are reported.

BASE EVALUATION STRATEGY

The following describes the methodology to measure the three performance metrics.

NETWORK TRAFFIC

Using Wireshark statistics of the data exchange are evaluated. The TCP streams between the Context Agent and the CMN are filtered out for each direction of communication, and statistics extracted related to 1) data exchange and 2) the overhead information, including TCP overhead, required for the transmissions. The statistics are normalized to the 50 samples, so the results are mean values for a single request, response or update, while in fact, for each sample interaction retransmission and multiple segments may occur. To evaluate the impact of context state sizes, a reconfigurable retriever is used which allows setting the state of the accessed information. The impact of state size on the traffic is evaluated for the range 10 - 10KB state information.

DELAY WITHOUT QOS (PRACTICAL)

For accessing dynamic context information remotely, time is essential which is the rational for investigating the delay and access times. For each of the sample, time stamps are logged and used to examine the delay for the respective access strategy. Two principle methods are used:

- For the reactive strategy, the system time is logged just before making the RPC call to the Context Agent, and when this returns with a value, the time is logged again. The time difference is used as an indication of the access delay time for the reactive strategy.
- For the proactive strategies, the context information being updated is equipped with a time stamp, which upon receiving at the CMN is used to calculate the difference in sending time and notification time. NTP is used to clock synchronize the CMN and Context Agent, and is done on a separate network than the one used to carry the context information updates.

The impact of the state size on the delay is evaluated for the range 10 - 10KB state.

EVALUATE MISMATCH PROBABILITY (THEORETICAL)

The delay distribution impacts the mismatch probability, which is the reason for which the delay distribution is measured to evaluate the effective mismatch probability from previous measurements with the theoretical models derived in [BOE-10]. Due to the empirical distribution function, numerical integration is used for evaluation of the mmPr.

APPLYING DIFFERENT QOS CAPABILITY TO THE NETWORK

The same methodology is reused in this investigation, now, only with QoS enabled on the Context Management information exchange.

4.7.1.6. EVALUATION RESULTS

This paragraph describes the obtained results for two different scenarios, the basic one and the QoS one.

BASE EVALUATION RESULTS

The following paragraphs describes the obtained results in term of network traffic, access time and mismatch probability in the base evaluation case, where QoS aspects are not considered.

NETWORK TRAFFIC

The following tables show the results from the network traffic analysed. For the following tables, the notation (1) and (2) are used such that:

(1) indicates traffic in the direction from the Context Agent to the CMN.
(2) indicates traffic in the direction from the CMN to the Context Agent.

Reactive strategy [Byte]					
State size	Data (1)	Data (2)	Overhead(1)	Overhead(2)	Total traffic
10	1009	1238	304	442	2993
1000	2066	1238	297	487	4088
2000	3132	1238	315	530	5215
5000	6264	1238	247	602	8351
10000	11590	1238	257	844	13929
Avg.	4812	1238	284	581	6915,2
Std.Dev.	233*	0	30	158	

Table 28 Reactive strategy results. All results are in bytes.

Proactive strategy (80% congested) [Byte]					
State size	Data (1)	Data (2)	Overhead(1)	Overhead(2)	Total traffic
10	3841	2983	1027	960	8811

1000	5055	3064	1039	1057	10215
2000	5862	2896	984	1054	10796
5000	9412	3008	959	1174	14553
10000	14472	2902	924	1399	19697
Avg.	7728	2970	986	1128	12814
Std.Dev.	301*	48	72	169	

Table 29 Proactive strategy (80% congested) results. All results are in bytes.

*** The standard deviation has been cleaned from the state size, i.e. based on sqrt(Var[$Data_x$(1)-$stateSize_x$]).**

Proactive strategy (97% congested)					
State size	Data (1)	Data (2)	Overhead(1)	Overhead(2)	Total traffic
10	3697	2896	965	1020	8578
1000	4775	2902	980	1075	9732
2000	5880	2920	965	1141	10906
5000	9089	2927	909	1222	14147
10000	14770	2964	921	1511	20166
Avg.	7642	2922	948	1194	12706
Std.Dev.	435*	27	31	193	

Table 30 Proactive strategy (97% congested) results. All results are in bytes.

*** The standard deviation has been cleaned from the state size, i.e. based on sqrt(Var[$Data_x$(1)-$stateSize_x$]).**

As it can be seen, the overhead of transporting the state information for the proactive approaches is significantly larger than for the reactive approach, in the order of 3 KB for the data to be updated (comparing the Data(1) column, state size by state size, between the proactive and reactive strategies), but also a 1.5KB (comparing the Data(2) column, state size by state size, between the proactive and reactive strategies) overhead of data to the source is added compared with the reactive approach. The reason for this is that the implementation only sends updates regarding which information has changed at the given moment, and not the value of the information. The Context Management Node needs subsequently to send a query to the Context Agent for the values of the information, creating the additional traffic per updates in both directions.

Further, for the proactive strategies a subscription is required to be set up and made which is included in the results of the data shown here. However, the traffic for setting up the subscription is roughly 2500 bytes of the total 50 sample update traffic, which leads to roughly a 0.5%-1% extra traffic included in traffic per update compared with the reactive.

In the following two figures, the distributions of the average traffic type are shown for the reactive and proactive strategies.

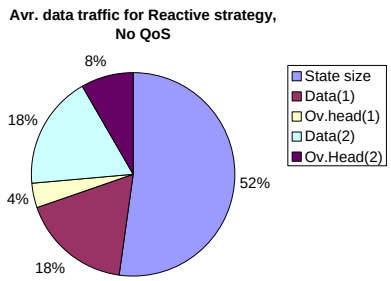


Figure 108 Data distribution for reactive strategies

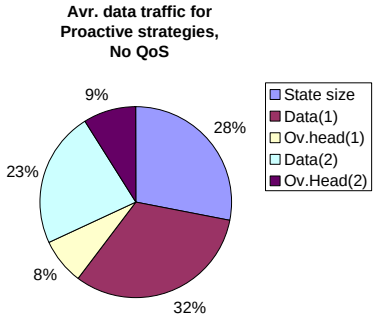


Figure 109 Data distribution for proactive strategies

Due to the additional query included in the proactive approaches, the relative state size takes up less of the created traffic, than for the pure reactive approach. This renders the proactive approaches much less effective compared to the reactive approach with the given settings of same request, event and update time intervals. Obviously, if the event time interval was much larger, the proactive, event driven strategy would be best, but on the other hand, so it could also be said with respect to the request rate or the update time interval.

For scenarios with large amounts of cross traffic, additional traffic is generated due to dropped packages etc. Figure 110 shows an overview of the causes to increased traffic overhead as a function of increased state size due to 1) additional TCP retransmissions, 2) Lost TCP segments and 3) Duplicated TCP acknowledgements. The numbers in the graph are relative to the amount of request-response/notifications received, i.e. 50. Hence, the plot relates to the number of events per request-response or notification observed.

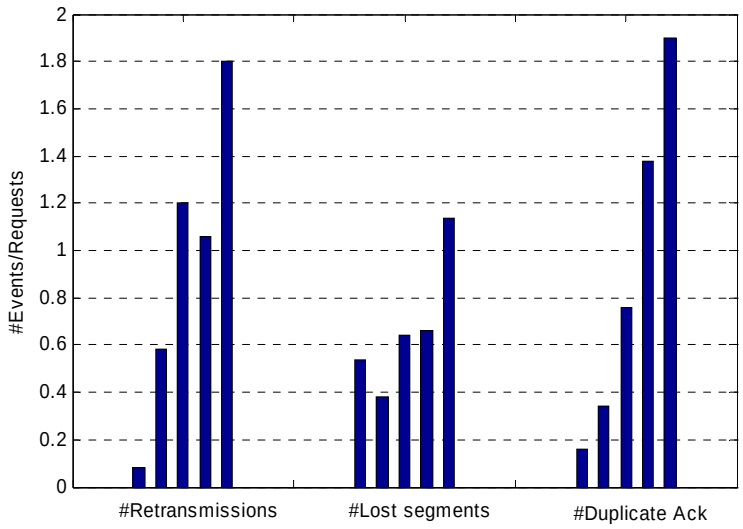


Figure 110 Overview of three major causes to the increased network overhead in the scenario of 97% network utilization. Numbers are normalized to a single request/response pair/update/notification. Columns are from left to right 10, 1000, 2000, 5000 and 10000 byte state sizes. For the cases with 80% utilizations all of the above are zero.

The amount of lost segments appears less influenced by the state size, but also in the non-QoS scenario all packets have roughly the same probability of being dropped. Lost segments lead to retransmission, which for an increased state size then leads to the increased number of retransmission observed. The duplicated ACKs increases as the existing ones do not arrive in due time because of the congested state in the network. All these effects are also a part of the increased communication time which is reflected in the following sections analysing the timing aspects of the context management framework.

CDF FOR DELAY AND ACCESS TIMES

Figure 111 and Figure 112 show the Cumulative Distribution Function (CDF) for the reactive strategy under different network congestion values, using a state size of 1 KB.

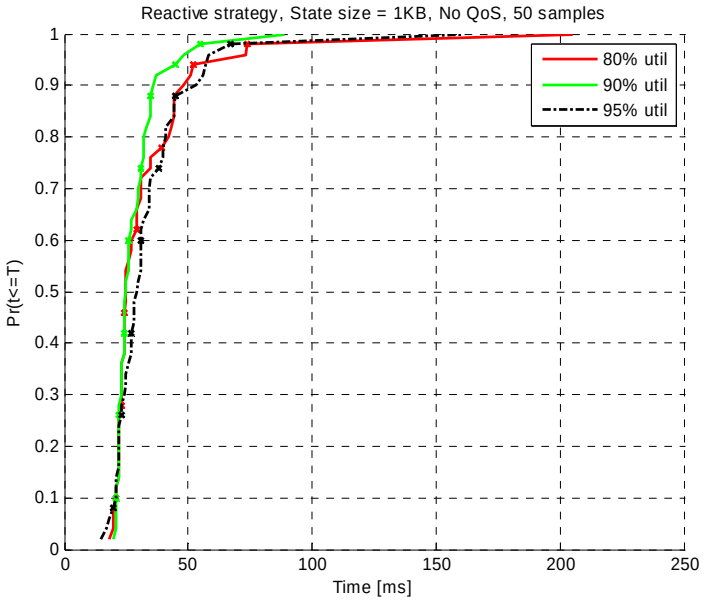


Figure 111 CDF of times for the reactive access strategy for network utilization values up to 95%.

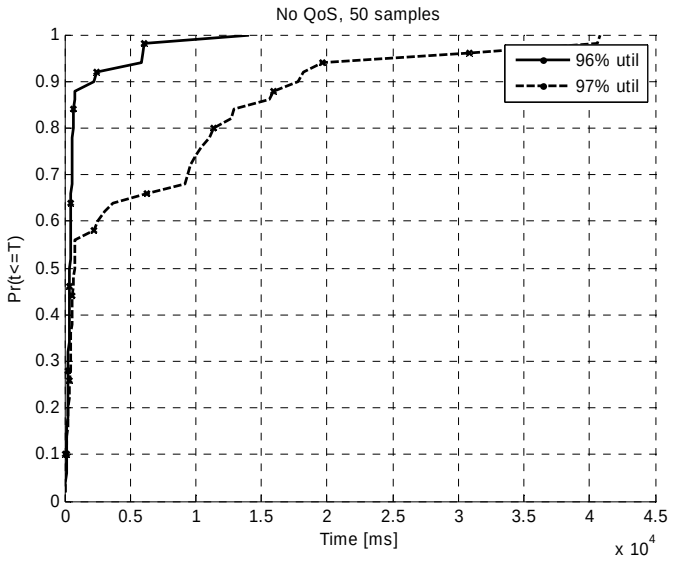


Figure 112 CDF of times for the reactive access strategy for network utilization values from 96% to 97%.

As it can be seen from these results (please note that the x-axis scales are quite different in the two plots), the access delays become much longer as the traffic saturates the network. This is not a big surprise, however, a highly undesirable situation for the adaptation of an application based on context needs to react quickly and reliably which are requirements hard to fulfil in such situation as saturated networks. Therefore, the aim is to provide Quality of Service on the context management, and later the effect of applied QoS on the context management is evaluated.

In Figure 113 and Figure 114 the CDF's for the proactive strategies are shown.

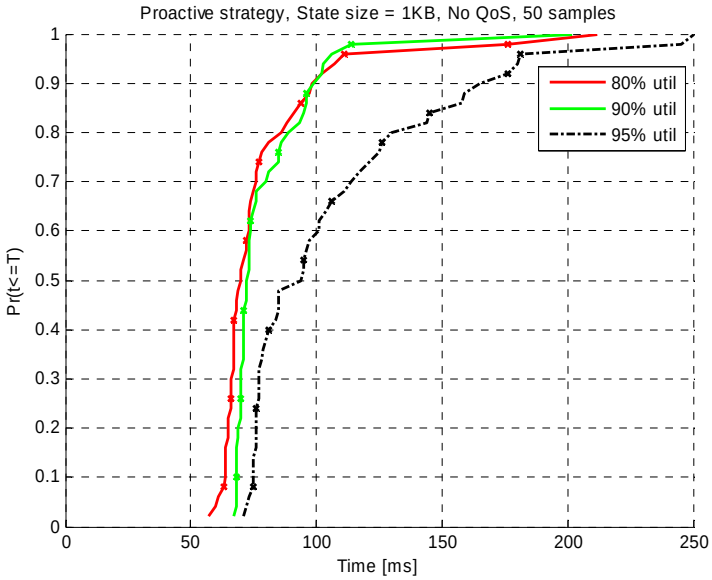


Figure 113 Proactive strategies (both periodic and event, E[E] =5 sec, T_{period} = 5 sec)

Figure 114 shows that the time delay is generally longer than the reactive strategy for the same network utilization factor, and that already from 95% starts to experience significantly higher delays. This is caused by the implementation in terms of the approach of sending a query upon the reception of a notification in order to get the value of the information.

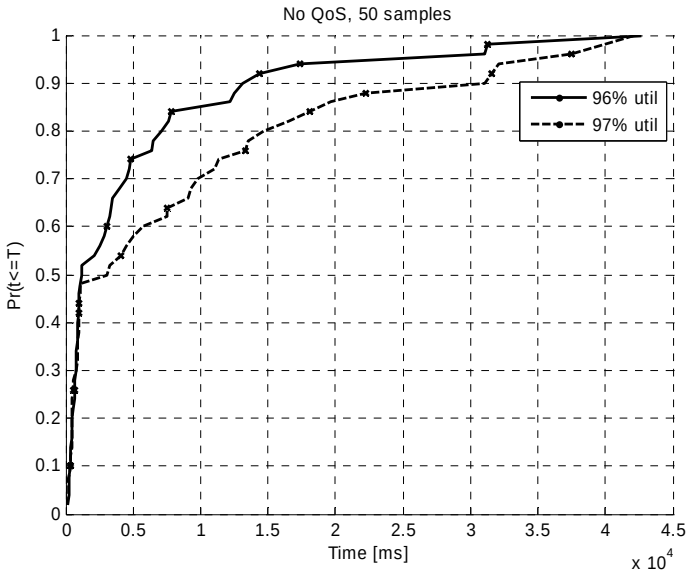


Figure 114 Proactive strategies (both periodic and event, E[E] =5 sec, T_{period} = 5 sec)

For the cases with higher utilization values of 96% and 97%, the proactive approaches suffer rather badly and worse than the reactive one due to the extra request/response caused by the implementation.

MEAN ACCESS TIME WITH VARYING NETWORK UTILIZATION

In Figure 115 the mean access times of the reactive strategy with a 1KB state, is shown for 80%, 90%, 95%, 96% and 97%, and as it can be seen, only up to 95% network utilization not much impact from the traffic can be seen.

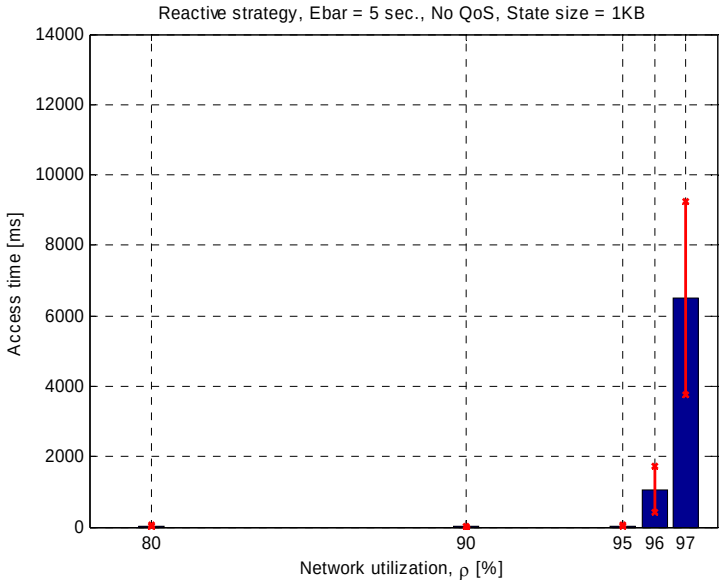


Figure 115 Mean access delay for the reactive access strategy with a 1 KB state size, with 95% confidence intervals of the mean values indicated by the red lines.

Same tendency can be observed for the proactive cases, however, with an increased mean delay due to the added request/response pattern to receive the value of the information.

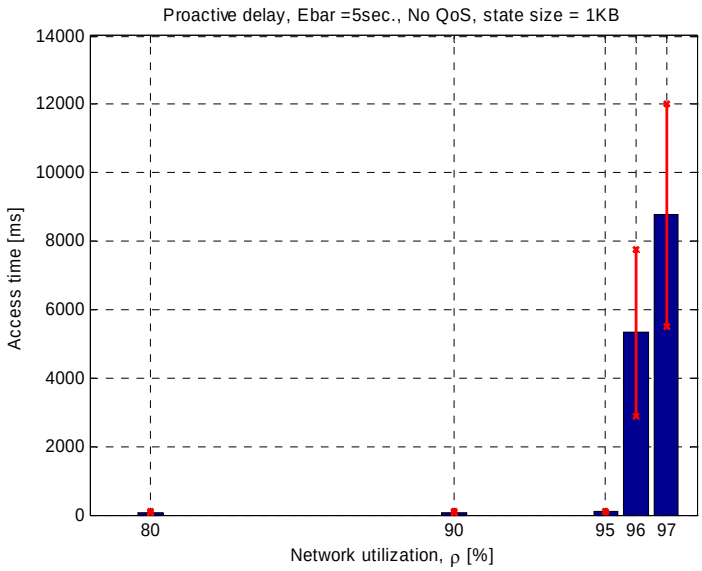


Figure 116 Mean delay times for the proactive access strategy with a 1 KB state size, with 95% confidence intervals of the mean values indicated by the red lines.

DELAY DUE TO NETWORK UTILIZATION FOR VARIOUS STATE SIZES

The state size is suspected also to have a large impact to the delay, since all the data needs to be transferred through a TCP connection, and may suffer from a set of retransmission due to lost packages, leading to retransmissions and thus longer delay. Figure 117 shows the mean access delay times for varying network utilization and state size of the desired context information for the reactive approach, and Figure 118 the same for the proactive approaches.

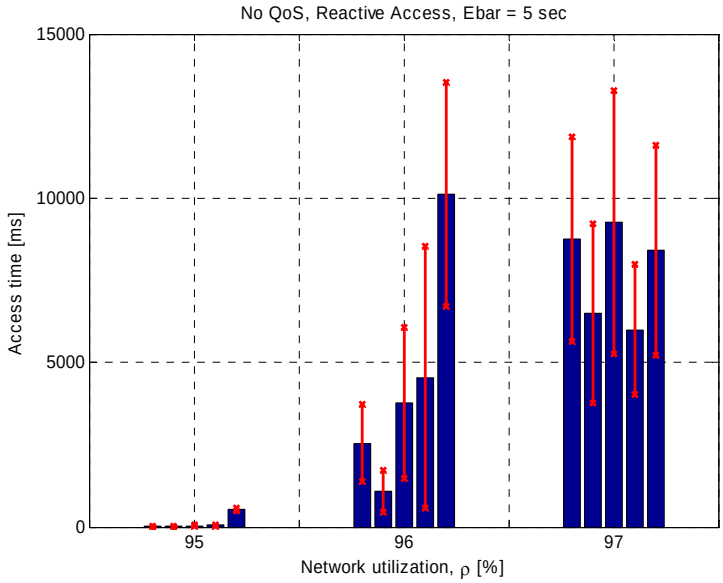


Figure 117 Mean access delay times for the reactive strategy with varying network utilizations and state sizes. Notice, that for graphical comparison some bars are located slightly to the left/right of their respective values. The bars are organized such first bar in a set is 10 byte state size, second bar is 1000 bytes, third bar is 2000 bytes, fourth is 5KB and fifth is 10KB.

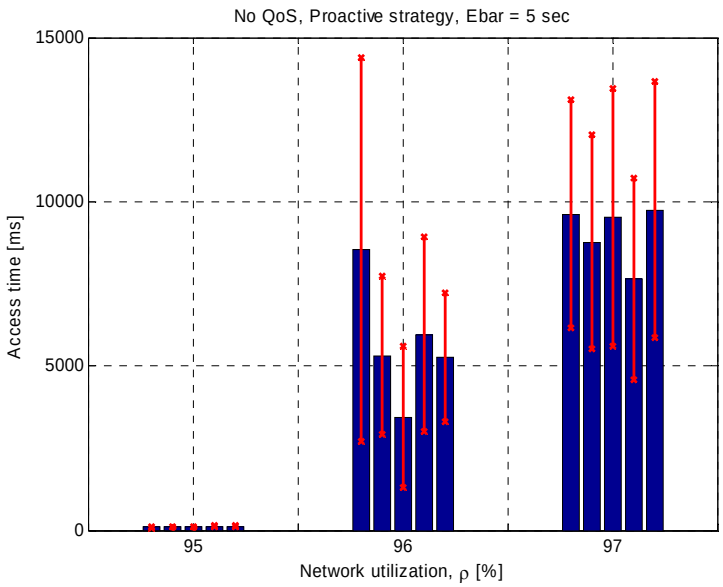


Figure 118 Mean access delay times for the proactive strategy with varying network utilizations and state sizes. Notice, that for graphical comparison some bars are located slightly to the left/right of their respective values. The bars are organized such first bar in a set is 10 byte state size, second bar is 1000 bytes, third bar is 2000 bytes, fourth is 5KB and fifth is 10KB.

Clearly, the impact for all the mean delay increases with the network utilization as expected. The impact of the state size is not as clear, as it varies too much to say anything meaningful. This is, however, not so surprisingly as packets in the saturated networks are dropped at random, which may lead to a high spread of access delays observed.

RESULTING MISMATCH PROBABILITY

In the following, the resulting mismatch probability of the different strategies is shown using the various access time delay distributions obtained through the measurement campaigns done.
Notice, for the reactive strategy the access time was measured, while for the mmPr it is the delay of the response that matters. Therefore, it is assumed *a delay symmetric link for the reactive strategy and use half the access delay measured, shown in* Figure 117.
The expressions shown for each of the strategies are shown 4.7.1 and used with the respective delay distributions as well as time intervals. All events are assumed to be a Poisson process with rate 1/5, that is a mean event time of 5 sec (this is noted Ebar in figures). The mean request rate is 1/5 sec, i.e. a request is send per 5 seconds. This is noted Rbar.

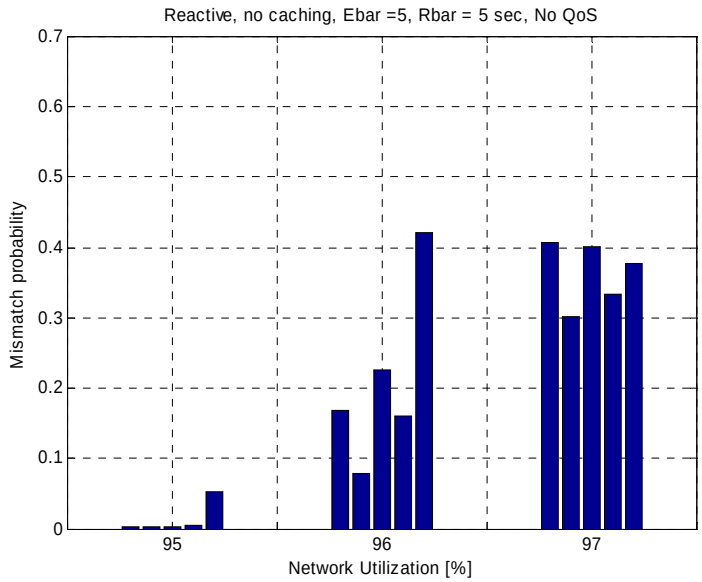


Figure 119 Mismatch probability for the reactive strategy (bars organized in 10B, 1KB, 2KB, 5KB, 10KB per utilization point)

In Figure 119 which shows the resulting mmPr for the reactive strategy, there is definitely an increase of mmPr due to the network utilization, however, the state size is again as with the access delays a bit varying, except for the low utilization case, where the mmPr for the 10KB state size is significantly larger than the cases with lesser state sizes. Also, worth noticing, is the mmPr for the 5KB case at 96% utilization, which has less mmPr than the 2 KB state, while the mean access delay shown in Figure 117 is higher than the mean access time of the 2KB case. The reason for this is that the variance of the delay observed, is much higher for the 5KB case than the 2KB case, which leads to a reduction of mmPr as foreseen in [BOE-10].

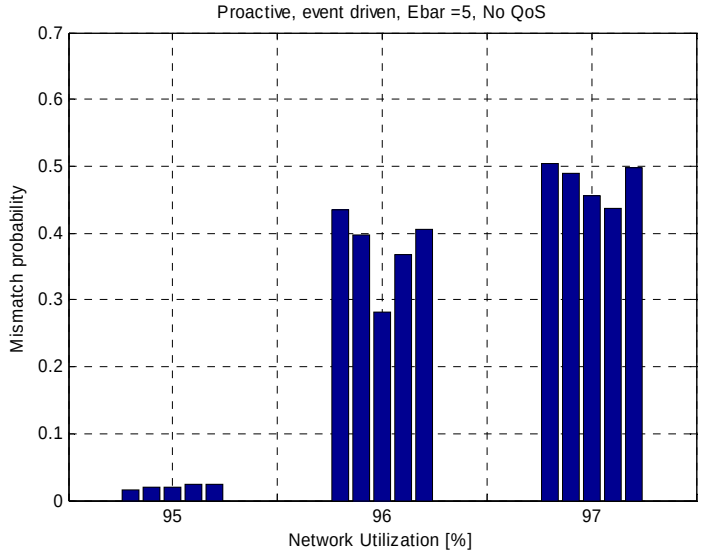


Figure 120 Mismatch probability for the proactive, event driven strategy (bars organized in 10B, 1KB, 2KB, 5KB, 10KB per utilization point)

Notice, for the proactive strategies the CMN sends a request for the context value after it has received the context update notification from the Context Agent. For the mmPr calculation a single update message is assumed, whereas in the implementation more communication is done via a reactive fetch mechanism. Because of this, in the following it is assumed that:
Upon an event, E_1, that changes the value, the new value is stored in a local cache and used as response for the subsequent request, R_1, from the CMN. If an event E_2 happens before R_1 is resolved, E_2 is stored in a separate cache and hereby not altering the response value of R_1. This could be the case if the communication is done by a thread, spawned by the event and destructed upon successful transfer.

With this assumption, then for the proactive cases, as shown in **Errore. L'origine riferimento non è stata trovata.** and Figure 121, the resulting mmPr's are much less influenced by the state size. Hence, the main influencing factor is here caused by the network utilization. Both the event driven as well as the periodic update case lead to higher mmPr's than the reactive, but part of this is caused by this additional request/response pattern required to get the actual information, which means that additional delays are introduced. For the periodic strategy, the resulting mmPr is also influenced highly by the update period which may be set to a higher frequency to reduce mmPr at the cost of increased network traffic and therefore network utilization.

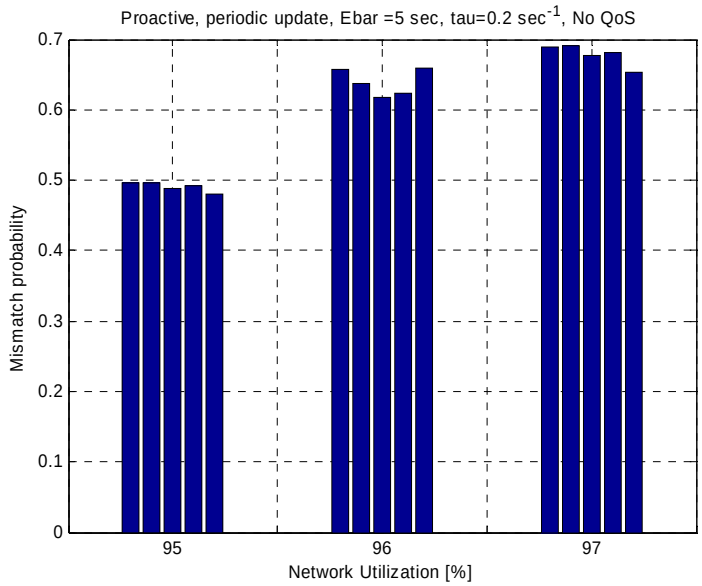


Figure 121 Mismatch probability for the proactive, periodic update strategy using a low update frequency (T_{per} = 5 seconds)

One interesting question remains unanswered here, whether it really pays off to increase update rate to reduce the mmPr or not? In Figure 122 the same situation is shown, however, now with a 10 times higher update frequency. In this situation, there is a bit more variation in the mmPr due to the characteristics of the various delay distribution, however the mmPr is much lower. This, of course leads to a traffic increase of a factor 10, but if that means a situation of using T_{per} = 5 sec and 95% utilization (Figure 121 leftmost column), goes to a situation with T_{per} =0.2 sec and 97% utilization (Figure 122 rightmost column), then this might be a valid option in order to gain

necessary information reliability. The exact relation between Figure 121 and Figure 122 is left for future study.

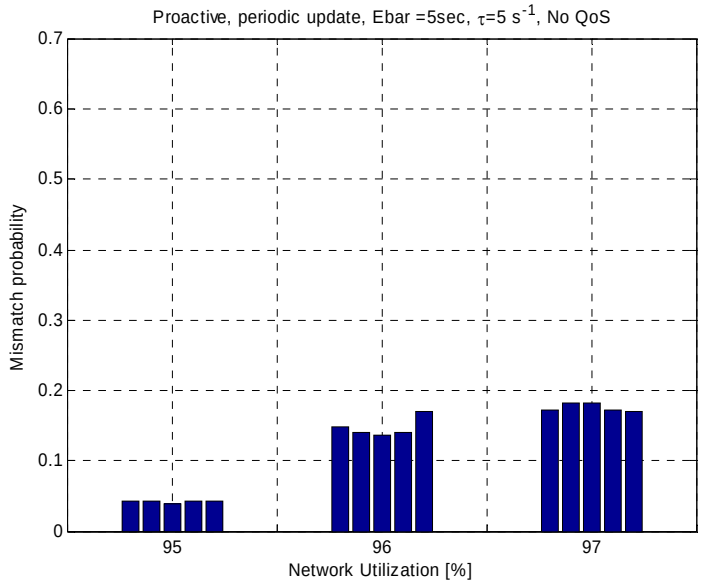


Figure 122 Mismatch probability for the proactive, periodic update strategy using a high update frequency (T_{per} = 0.2 seconds)

APPLYING QOS CAPABILITY TO THE NETWORK

In this part the same measurements are redone, but with the difference that all traffic that is going between the Context Agent and CMN is now equipped with a QoS flag, and is done in a QoS enabled network environment. In this way, it is expected that the delays will be heavily reduced due to the packet prioritization, and thereby improving reliability of the information.

NETWORK TRAFFIC

The following tables show the traffic statistics for the case when QoS is being applied. The main potential difference would be in the high load cases with respect to reduces retransmissions due to lost packages. For the following tables, the notation (1) and (2) are used such that:

(1) indicates traffic in the direction from the Context Agent to the CMN.
(2) indicates traffic in the direction from the CMN to the Context Agent.

Reactive strategy [Byte]					
	Data (1)	Data (2)	Overhead(1)	Overhead(2)	Total traffic
10	1009	1238	337	292	2876
1000	2066	1238	337	351	3992
2000	3132	1238	338	411	5119
5000	6264	1238	272	470	8244
10000	11529	1238	272	734	13773

Avg.	4800	1238	311	452	6800,8
Std.Dev.*	209	0	36	171	

Table 31 Reactive strategy. All values in bytes

Proactive strategy (80% congested) [Byte]					
	Data (1)	Data (2)	Overhead(1)	Overhead(2)	Total traffic
10	3769	2952	1003	945	8669
1000	4775	2896	987	986	9644
2000	5977	2952	1006	1081	11016
5000	9057	2896	934	1107	13994
10000	14710	2952	940	1410	20012
Avg.	7658	2930	974	1106	12667
Std.Dev.*	388	31	35	183	

Table 32 Proactive strategy (80% congested). All values in bytes

Proactive strategy (97% congested) [Byte]					
	Data (1)	Data (2)	Overhead(1)	Overhead(2)	Total traffic
10	3769	2952	1016	925	8662
1000	4868	2952	1011	993	9824
2000	5977	2952	1014	1068	11011
5000	9234	2952	946	1135	14267
10000	14710	2952	939	1401	20002
Avg.	7712	2952	985	1104	12753
Std.Dev.*	379	0	39	184	

Table 33 Proactive strategy (97% congested). All values in bytes.

* The standard deviation has been cleaned from the state size, i.e. based on sqrt(Var[$Data_x$(1)-$stateSize_x$])

As it can be seen from the data in the tables, there is relatively little variance in all the involved messaging, as compared to the no-QoS case. The difference between the proactive strategies from 80% to 97% congested also shows no significant increase in the data traffic, except that which is caused by the state size itself. The distribution of the data traffic types related to a request and notifications are shown in Figure 123 and Figure 124 for the reactive and proactive strategies, respectively.

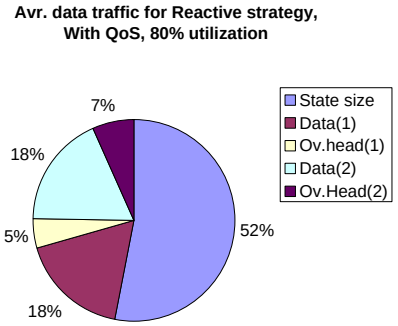


Figure 123 Data distribution for reactive strategy

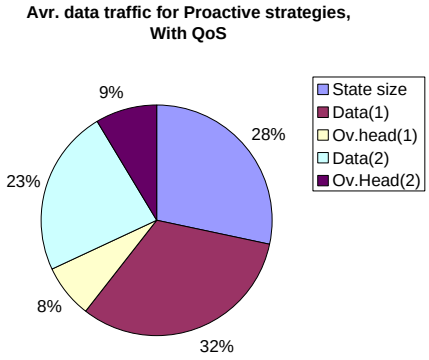


Figure 124 Data distribution for proactive strategies

The Figure 123 and Figure 124 show the same distribution of traffic between the same nodes as for the non-QoS case.

CDF FOR ACCESS TIMES

Figure 125 shows the empirical Cumulative Distribution Function for the reactive access time measured with QoS in a congested network situation.

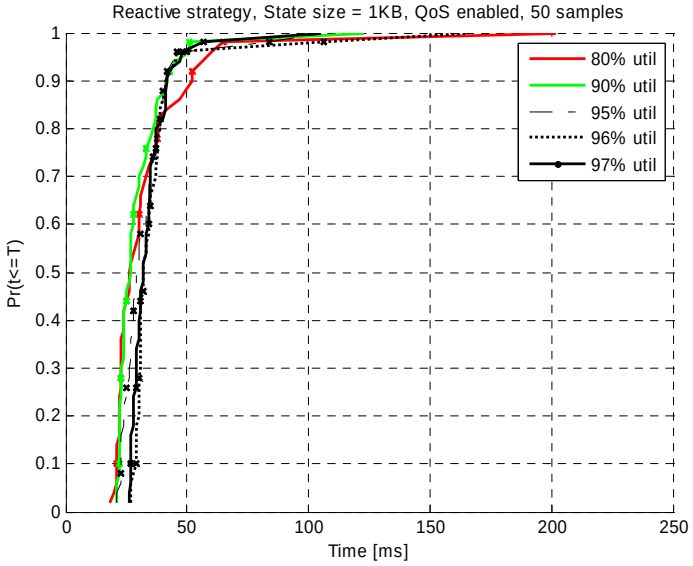


Figure 125 Empirical cumulative distribution for the reactive access time

As it is seen from **Errore. L'origine riferimento non è stata trovata.**, there is very little difference in the access times considering the network utilization interval of 80%-97% congestion. This result, however, is only valid as long as the active QoS class channel is not saturated. In that case, the results would become as the no-QoS class situation shown in Figure 111 and Figure 112. Thus, in case of the context manager should need to exchange a significant amount of

information, e.g. if it contains a high number of context information which is frequently accessed from remote nodes, it may need to classify information differently and in more than one QoS class to distribute the load.

For the proactive strategies, for which the results are shown in Figure 126 the spread in delay time still exists and a slight impact of the increased traffic can be observed, but at a much lower scale than for the non-QoS case. It is worth mentioning here again that the time delays observed are again much higher than the reactive, due to the implementation specific value access, i.e. first the CMN is notified, and thereafter it sends a reactive access to the Context Agent. The application does not "see" this process and only experiences that with some delay as shown in Figure 126**Errore. L'origine riferimento non è stata trovata.**, it gets a notification message.

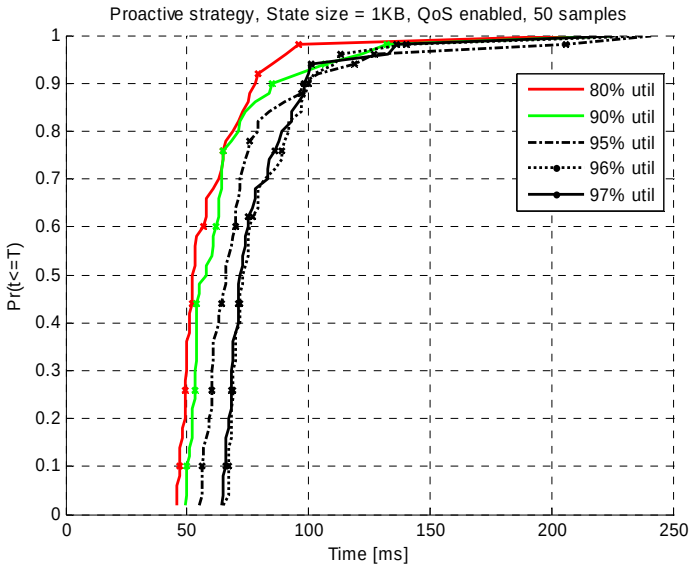


Figure 126 Empirical cumulative distribution for the proactive delay time

MEAN ACCESS TIME WITH VARYING NETWORK UTILIZATION

From Figure 127, which shows the mean access delay times, it can be seen that applying QoS makes the access delay time insensitive towards the existing cross traffic. Again, if the context manager starts to receive significant amounts of requests, the situation for that particular QoS class changes to what was shown in the non-QoS class. Ebar denoting the mean time interval between events is set to 5 seconds, and Rbar the mean time between requests that are being sent is also set to 5 seconds.

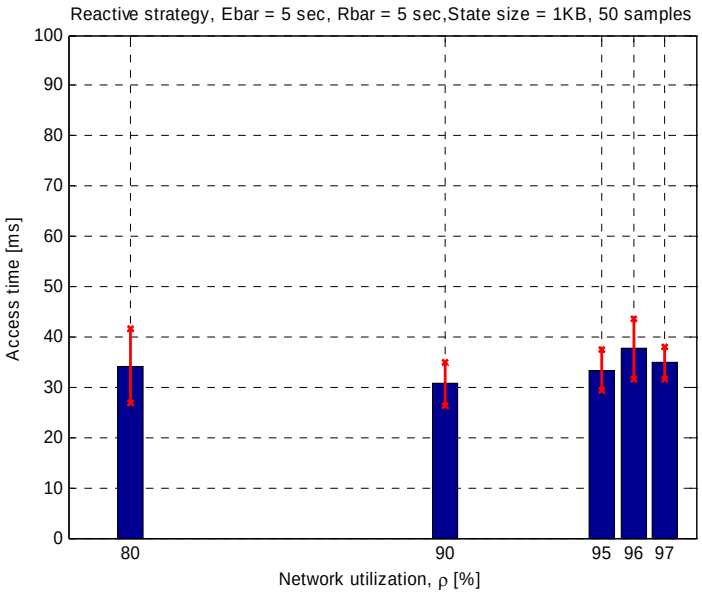


Figure 127 Mean access times for the reactive access strategy with varying network utilization with 95% confidence intervals

Figure 128 which shows the mean delay for the proactive strategies, shows not surprisingly the similar characteristics, however with an increase mean delay as to the reactive which is due to the underlying reactive access that the context manager does in order to fetch the actual value of the information.

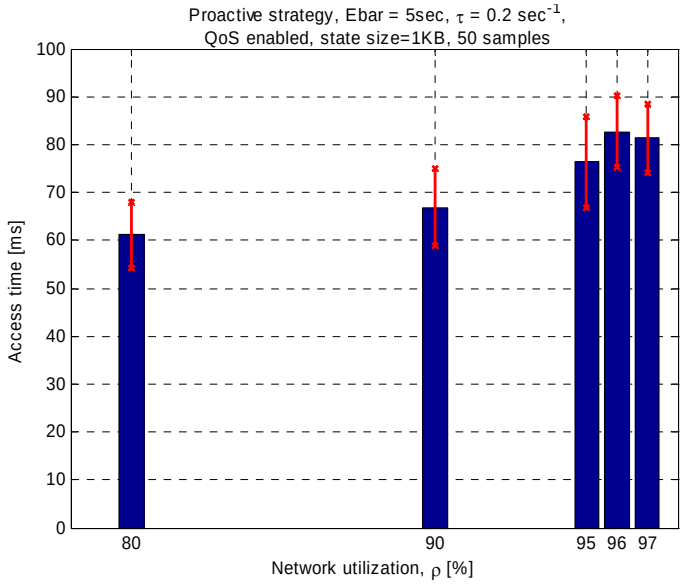


Figure 128 Mean delay times for the proactive access strategy with varying network utilization with 95% confidence intervals

DELAY DUE TO NETWORK UTILIZATION FOR VARIOUS STATE SIZES

Figure 129 shows similar results, but for different state sizes and only for the top three highest network utilization values investigated.

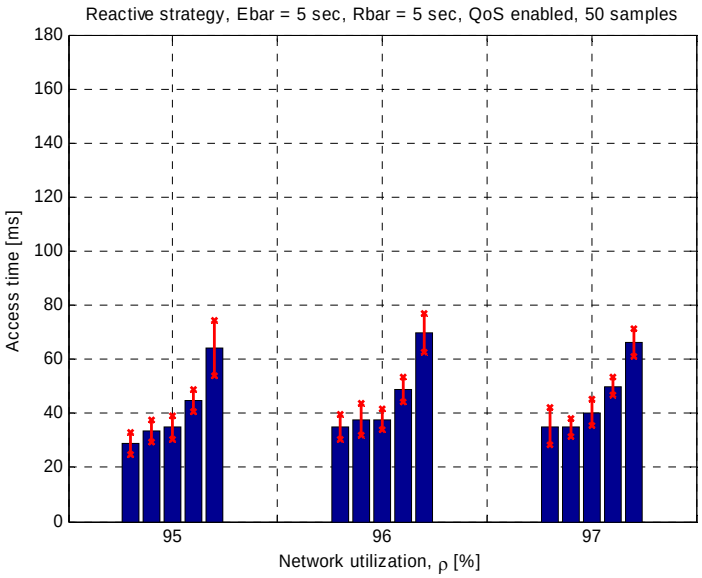


Figure 129 Mean access times for the reactive strategy with first column as 10 byte state size, second column as 1000 bytes, third column as 2000 bytes, fourth column as 5KB and fifth as 10KB

The results seen from this figure show that the impact of the access time is as expected not influenced by the network utilization, but by the state size itself. The impact significantly increases the access delay when the state size becomes larger than 2KB. In this case an increase of more than 100% in access delay can be observed when increasing from 2KB to 5KB. The same increase is not seen from the proactive strategy shown in Figure 130, however, the delay times are also at all time much larger due to the underlying access technique as described. Beside this, the same behaviour of an increase in delay time only caused by the increased state size is observed as shown in Figure 130.

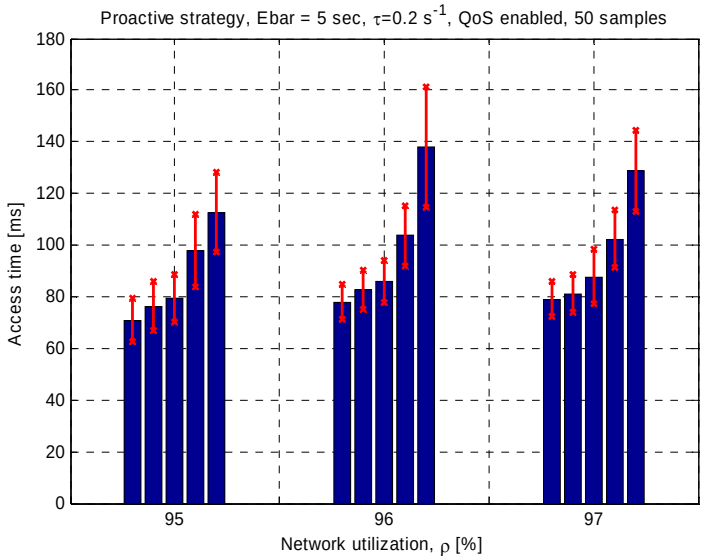


Figure 130 Mean access times for the proactive strategies with first column as 10 byte state size, second column as 1000 bytes, third column as 2000 bytes, fourth column as 5KB and fifth as 10KB

RESULTING MISMATCH PROBABILITY

Resulting mmPr for the reactive strategy of the respective cases are shown in Figure 131 with the same assumption on symmetric delays.

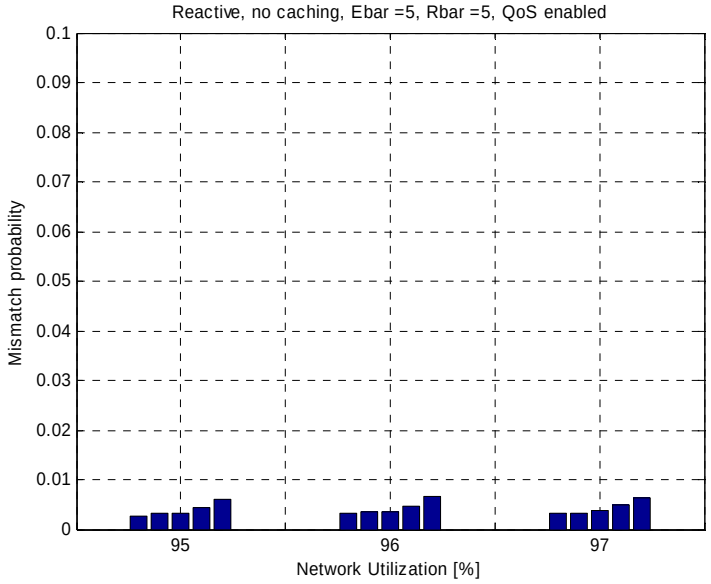


Figure 131 Resulting mismatch probability for the reactive strategy of with varying state sizes (10, 1000, 2000, 5000 and 10000 bytes) and network utilization

The access delays are directly observed in the resulting mmPr for this case and attain the same increasing characteristics as shown in Figure 129. For the proactive, periodic update strategy shown in Figure 133, the results show clearly that the impact of the mmPr is neither the state size

nor the network delay. The point here is that using QoS for the periodic, leads to such small delays that it is rather the update rate that determines the resulting mismatch probability which in this case leads to a mmPr value slightly lower than 0.5, until that particular QoS class traffic starts to hit saturation level.

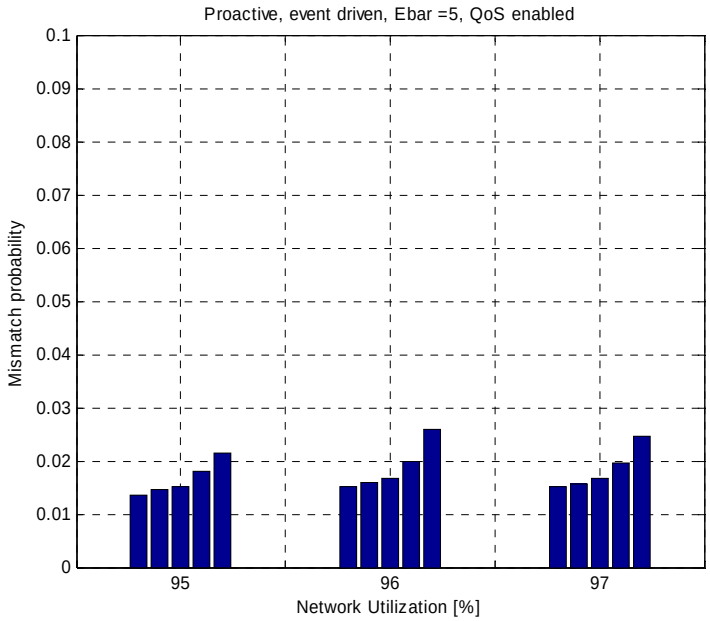


Figure 132 Resulting mismatch probability for proactive, event driven update strategy.

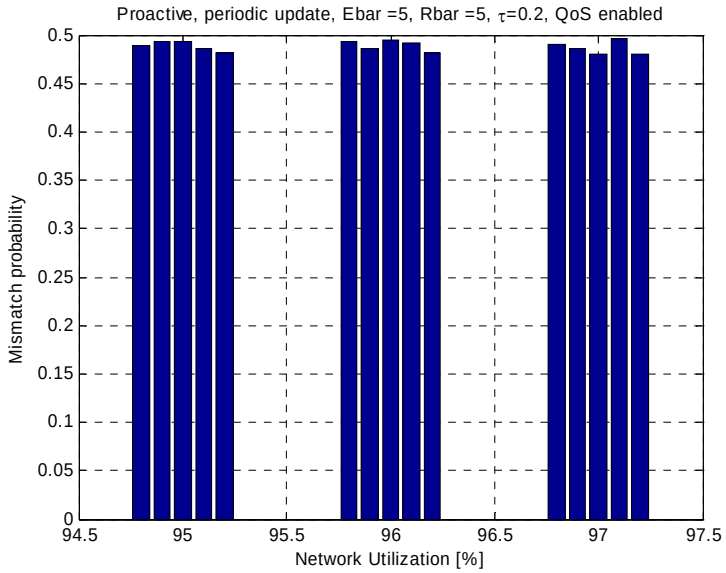


Figure 133 Resulting mismatch probability for the proactive, periodic update with varying state sizes (10, 1000, 2000, 5000 and 10000 bytes) and network utilization for a low update frequency (T_{per} = 5sec)

Figure 132 shows the resulting mismatch probability using the event driven strategy with varying state sizes and network utilization but with QoS enabled for the context management framework. Although, the mmPr in this case is influenced mainly by the event process (and the delay, but due to QoS it is low), there is a factor relating to the state size of the information in the same way as the reactive strategy.

IMPACT OF QOS

In this section the QoS and non-QoS scenarios are compared, summarize, discuss and conclude the evaluation of network-related aspects involved in migration.
A summary of the total amount of traffic overhead (i.e. excluding the data itself) for the reactive and proactive strategies under a 97% network congestion situation and normalized to traffic per request-response/update is shown in Table 34 and Table 35.

Reactive strategy			
	Total overhead (No QoS) [Byte]	**Total overhead (QoS) [Byte]**	**Relative diff.**
10	746	629	0.84
1000	784	688	0.88
2000	845	749	0.89
5000	849	742	0.87
10000	1101	1006	0.91

Table 34 Total traffic overhead generated for reactive strategy per request/response in bytes

Proactive strategies			
	Total overhead (No QoS) [Byte]	**Total overhead (QoS) [Byte]**	**Relative diff.**
10	1985	1941	0.98
1000	2055	2004	0.98
2000	2106	2082	0.99
5000	2131	2081	0.98
10000	2432	2340	0.96

Table 35 Total traffic generated for proactive strategies per update

From these results it is clearly seen that using QoS benefits the reactive strategy the most, as the amount of overhead required to transport the data, which includes retransmissions, duplicate acknowledgements etc., are reduced by 10-15%, whereas the overhead is only reduced 2-4% for the proactive strategies. This, again, reflects that the proactive strategies generate much more traffic due to the implementation strategy of executing a reactive access upon the proactive notification. Even for the QoS case, the proactive strategies produces 2-3 times more network overhead due to this.
The following tables contain a summary of the results on the timing and reliability metrics from the case of 97% network congestion. In the following tables, E[R] indicates the mean request

rate, E[E] the mean event time interval, E[D] the mean delay, and V[D] the variance of the delay observed.

Reactive strategy (E[R] = 5 sec)									
State size	No QoS			QoS			Rel. diff. (QoS/NoQoS)		
	E[D]	V[D]	mmPr	E[D]	V[D]	mmPr	E[D]	V[D]	mmPr
10	8759	125E6	0.4076	35	639	0.003	0.004	5E-6	0.007
1000	6494	98E6	0.3027	35	143	0.003	0.005	1E-6	0.010
2000	9289	209E6	0.4011	40	311	0.004	0.004	1E-6	0.010
5000	6002	52E6	0.3329	80	268	0.008	0.013	5E-6	0.024
10000	8416	134E6	0.3772	143	497	0.014	0.017	4E-6	0.037

Table 36 Mean delay, delay variance and resulting mismatch probability for the no-QoS and QoS scenarios with different state sizes for the reactive strategy.

Proactive, event driven (E[E] = 5 sec)									
State size	No QoS			QoS			Rel. diff. (QoS/NoQoS)		
	E[D]	V[D]	mmPr	E[D]	V[D]	mmPr	E[D]	V[D]	mmPr
10	9621	157E6	0.5032	79	575	0.015	0.008	4E-6	0.030
1000	8775	137E6	0.4892	81	682	0.016	0.009	5E-6	0.033
2000	9517	201E6	0.4568	88	1527	0.017	0.009	8E-6	0.037
5000	7634	123E6	0.4369	102	1560	0.020	0.013	13E-6	0.046
10000	9758	200E6	0.4970	129	3216	0.025	0.013	16E-6	0.026

Table 37 Mean delay, delay variance and resulting mismatch probability for the no-QoS and QoS scenarios with different state sizes for the proactive, event driven strategy.

Proactive, periodic update (τ = 1/5 sec, E[E] = 5 sec)									
State size	No QoS			QoS			Rel. diff. (QoS/NoQoS)		
	E[D]	V[D]	mmPr	E[D]	V[D]	mmPr	E[D]	V[D]	mmPr
10	9621	157E6	0.6893	79	575	0.4898	0.008	4E-6	0.7106
1000	8775	137E6	0.6913	81	682	0.4866	0.009	5E-6	0.7039
2000	9517	201E6	0.6775	88	1527	0.4806	0.009	8E-6	0.7094
5000	7634	123E6	0.6804	102	1560	0.4916	0.013	13E-6	0.7225
10000	9758	200E6	0.6542	129	3216	0.4808	0.013	16E-6	0.7350

Table 38 Mean delay, delay variance and resulting mismatch probability for the no-QoS and QoS scenarios with different state sizes for the proactive, periodic update strategy.

From these results it is seen that although the traffic generation for the proactive strategies are not really reduced using QoS, the fact that more bandwidth is available to the given traffic class helps reducing the delay significantly. The delay becomes only slightly worse and with a higher variance due to the added traffic required, however, at least for the proactive event driven strategy, this factor only leads to a slight increased mmPr. This can be remedied either by changing the implementation so that it sends the state information along the notification message, or by using an incremental update variant so that only the actual differences since last update is being send when updating the context information. For the proactive, periodic case,

the low delay in this case does not help due to the update time interval is too high, hence the resulting mmPr becomes roughly the same value and for this case around a value of 0.48. Any reduction in mmPr for the proactive periodic case needs to be done via an increase of update frequency but at the cost of equally raising the network traffic by the same frequency, and thereby also the network utilization.

4.7.1.7. CONTEX MANAGEMENT FRAMEWORK EVALUATION CONCLUSIONS

From the evaluation of the context management framework some insight has been gained both in terms of functionality and in terms of its performance. The following bullets summarize the lessons learned and suggestions for future work:

- Proactive strategies (event as well as periodic) should include the state information inside the initial notification message, instead of having the notification triggering a request/response message pair to get that information. This will save a lot of network overhead and reduce timing and potentially also increase reliability of the context information accessed.
- The overhead of sending a few bytes state information is really huge, so in order to improve efficiency, it can be considered to piggy-bag additional information when sending updates/responses if relevant. This would obviously complicate a CMF as it would need to know information demands and schedule and pack/unpack updates/responses much more carefully.
- Applying QoS obviously helps a lot; however, this shall not be done blindly as in cases where some information is very popular or many information elements are required, a particular QoS classification of context management traffic may only lead to bottlenecks for that particular QoS class. Therefore, the CMF should be able to intelligently distribute the network traffic over various classes. Knowledge about the information dynamics (the event process) should be used in this process to classify updates/response messages containing values of slow changing information which are more delay tolerant, than fast changing information.
- The proactive, periodic update strategy appears to be relatively robust against state size and network delays, meaning that the relative change between the non-saturated and saturated network states tested for, are much lower than for any of the other strategies. The main dominating factor is the update time interval, which influences positively to a lower mmPr but negatively to an increased network traffic generation. Next steps in this direction, is to look into this trade-off and loop of reducing mmPr while increasing network traffic. Combinations of periodic updates and event driven or reactive strategies may also be of interest in order to optimize network overhead while keeping reliability of the information high.

4.8. MULTINETWORK SCENARIO EVALUATION

As discussed in D3.4 [D3.4], the main purpose of the Mobility Support module in the OPEN Migration Service Platform is to ensure that device and network changes, which might occur during the migration process, do not affect the communication between the client and the server. A prototype to demonstrate the capability of Mobility Support module was described in D3.3 [D3.3], and from now on it is referred to as the Mobility Support Prototype.

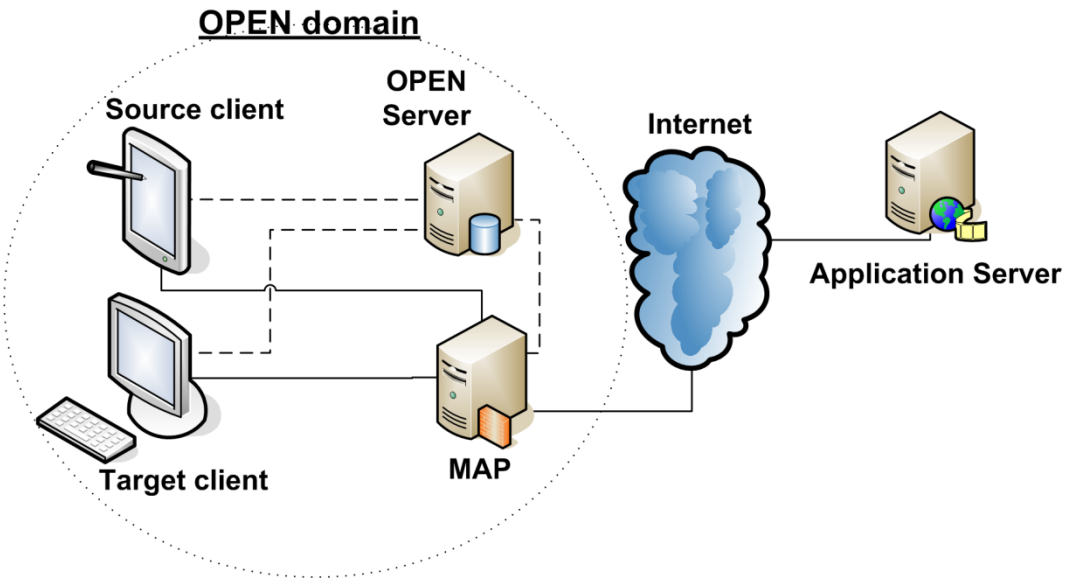


Figure 134 Session mobility is supported by introducing Mobility Anchor Point (MAP)

4.8.1. PERFORMANCE METRICS

A general concern regarding to the Mobility Support module is how much handover delay it introduces into the migration process. The delay affects user's experience during migration, which should be minimized. Such delay is affected by a number of factors, such as:

- Network conditions:
  - Network congestion
  - Links' delay
  - Links' throughput
  - Links' packet loss
- OPEN-related delay:
  - Migration overhead, such as the transferred state size etc.

The considered performance metric will be the handover delay of session mobility, which includes the effect of Mobility Support module and also its interplay with other OPEN modules. As such it is not possible to isolate the Mobility Support module as a separate factor contributing to the performance, since it cannot work without e.g. the orchestrator functionality to extract and insert state.

The Mobility Support module is realized partly through the use of a modified SOCKSv5 proxy as described in [D3.4]. The performance of SOCKS proxies has been investigated in detail in [MAL-99] where it is concluded that a normal application layer proxy has a high performance impact on the delay of the application traffic and in general does not scale very well. A technique to overcome these limitations is proposed and investigated always in [MAL-99] and it is concluded that the improvement is significant.

4.8.2. EVALUATION TESTBED

To measure the handover delay occurred during an application's migration, it was setup a test bed, which allows recording the timing of migratory events. Since those events take place in a network consisting of multiple components, either a clock synchronization scheme or a common central point to timestamp events is required. The latter can be achieved using a network emulator called "Air-in-a-box" [SCH-00], which was developed in the HIDNETS project [HID-10]. From now on, this network emulator is referred to as Topology Emulator.

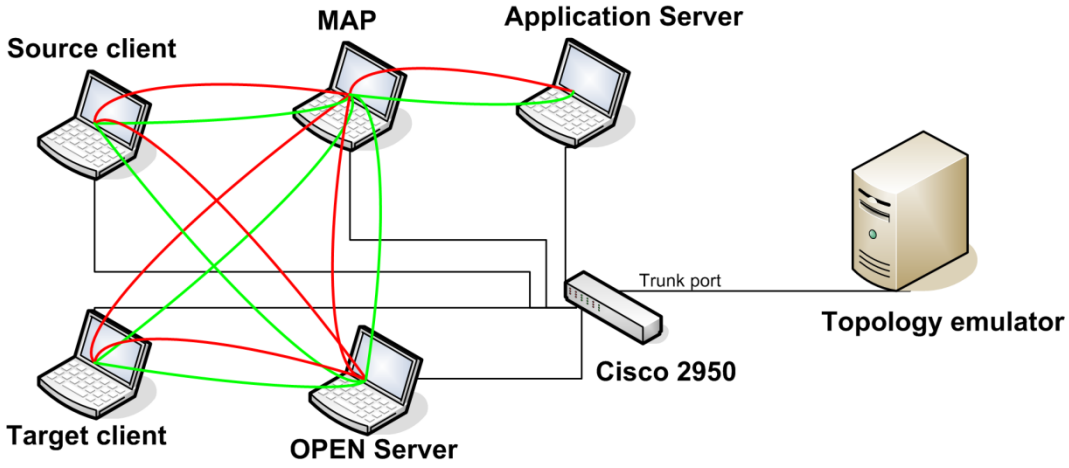


Figure 135 Topology Emulator and its setup. The black lines indicate the physical wire connections while the red and green lines indicate the emulated TX and RX links

The Topology Emulator is a powerful PC equipped with a Gigabit Ethernet card, which is connected to a Cisco 2950SX-24 switch trunk port (see Figure 135). The switch is configured such that each port gets its own Virtual Local Area Network (VLAN), so that there is a complete separation of network traffic from different nodes. Since all traffic is sent through the trunk port, the Topology Emulator can bridge different VLANs together, if two or more nodes in the network should be able to communication. The emulator is also able to delay and/or drop traffic at the data-link layer using EBtables in the Linux kernel. A user-space process can read information about the topology and the link properties and feed this information into the kernel thereby emulating a real network where nodes e.g. move in and out of range.

The Topology Emulator becomes a central point in the network, where all network traffic between all nodes can be recorded. The network traffic has been recorded using Wireshark and processed afterwards for the relevant migration events, since each event has a one-to-one

relation to packets in the network. The delays between the events can now be calculated since all timestamps are made with the same clock.

Node	CPU	RAM	Network Card
Application Server (AS)	Pentium M 1.73GHz	1GB	Gigabit Ethernet
Mobility Anchor Point (MAP)	Pentium M 1.70GHz	1GB	Gigabit Ethernet
Clients	Core2duo T7300 2GHz	2GB	Gigabit Ethernet
OPEN Migration Service Platform (OPEN MSP)	Core2duo T5500 1.66GHz	1,5GB	Gigabit Ethernet
Topology Emulator	Core2 6300 1,8GHz	3GB	Gigabit Ethernet

Table 39 Hardware used in the evaluation

The actual topology used to evaluate the session mobility prototype is shown as red and green lines in Figure 135. There are bi-directional links between the five entities, namely the OPEN server, MAP, source device, target device and the Application Server. The capacity of these links is 100Mbps, and in some of the presented evaluations, network delay and packet loss are introduced into the links marked in red. The purpose is to see how delay and packet loss will affect the performance of OPEN platform. All OPEN entities are running on separate laptops, as well as the Application Server. All laptops running the prototype software had Ubuntu 9.10, sun-java6-jdk and Eclipse installed, while the Application Server laptop had Ubuntu 8.10 LTS and Lighttpd installed and the Topology Emulator had Ubuntu 7.10 with a custom build 2.6.20.7 kernel and additional emulation user-space software installed. The hardware specifications are given in Table 39.

4.8.3. MIGRATION PROCEDURE WITH MOBILITY SUPPORT

Figure 136 illustrates the generic migration procedure, which is going to be evaluated in this section. This procedure consists of 7 events; each of them is marked by a unique ID, as shown in Table 40. As above-mentioned, the Topology Emulator enables to record the timing of all events occurring in migration since the packets are time-stamped in a central location. The total migration delay is measured as the duration from the SYN packet of TCP stream for the first event, to the last packet of TCP stream for the 5th event. Delay due to a particular event is calculated as the duration from its SYN packet to the last packet of TCP stream belonging to the event.

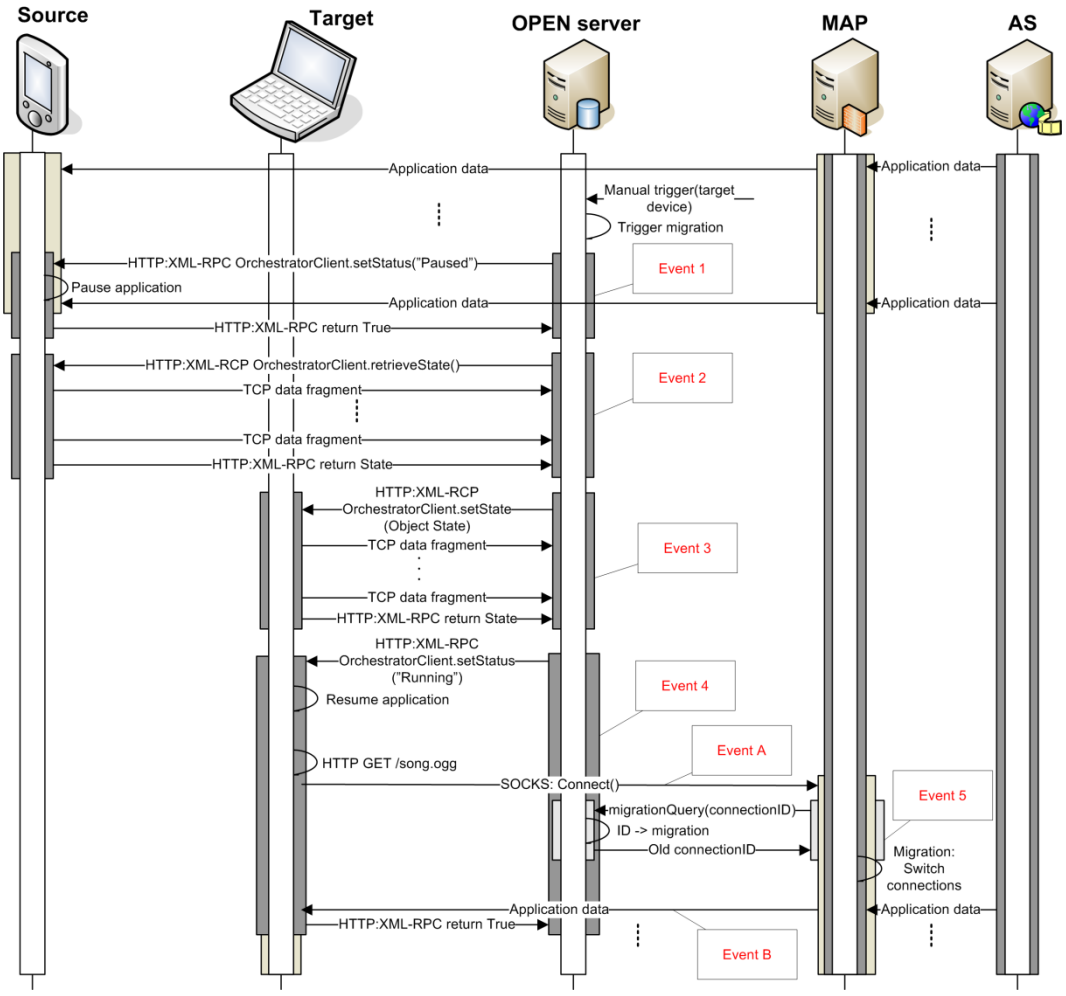


Figure 136 Migration procedure for the mobility prototype. All grey boxes marks the duration of a TCP connection

Event ID	Involved Nodes	Description
1	Source Device – MS	Pausing application
2	Source Device – MS	Retrieving State
3	Target Device – MS	Setting State
4	Target Device – MS	Resume application
A	Target Device – MAP	Connecting to MAP
5	MAP – MS	Querying MS
B	MAP Target – Device	First data packet sent

Table 40 Summary of migration events

Events A and B do not have a numbered event ID. The reason for this is that the TCP connection containing these events will not end until the application completes the streaming, which makes it difficult to use the duration of this TCP connection in the evaluation. This does not influence the results much since the time from the end of the TCP connection with event ID 5 to the event where the first data packet is sent to the client application is negligible.

Three sets of experiments were conducted using the Topology Emulator setup namely a set where the link delay was varied, a set where the link packet loss was varied and finally a set where the state size was varied. Each of these sets is described in the following sections.

4.8.4. EVALUATION RESULTS

In the following sections a detailed test results analysis is reported.

4.8.4.1. LINK DELAY

In the first series of experiments the link delay was adjusted using the Topology Emulator, from 0 ms to 500 ms (additional delay) on each link going to and from the OPEN Migration Server (see Figure 135). For each delay setting ten migrations were performed and a Wireshark trace file from the Topology Emulator was saved.

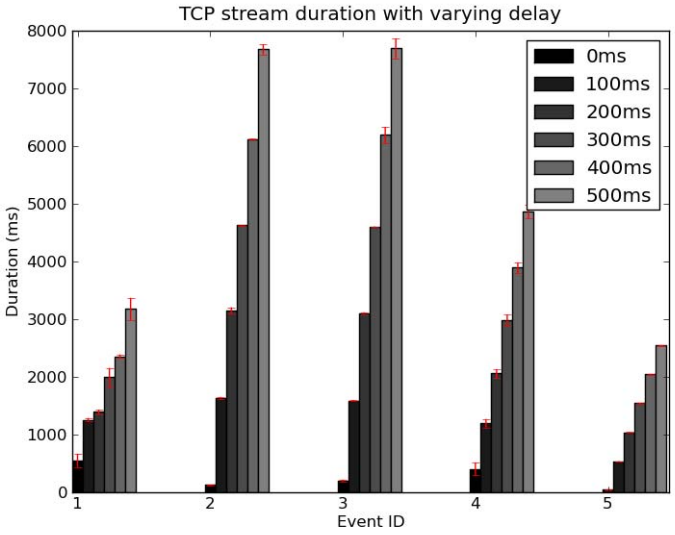


Figure 137 The duration of the TCP connections for each migration event, when the link delays to and from the OPEN Migration Server is adjusted from 0ms to 500ms with a 95% confidence interval

Figure 137 shows a plot of the duration of the TCP streams matching the events in Table 40. The results show a linear increase in the duration of the TCP connections when the link delay is increased. Figure 138 shows a plot of the sum of the duration of all the TCP connections involved in a migration. The sum of the durations is the actual migration delay seen from the network point of view, and it corresponds to the migration delay the client application will experience. The results show that the complete migration delay scales linearly with the link delay to and from the OPEN Migration Server.

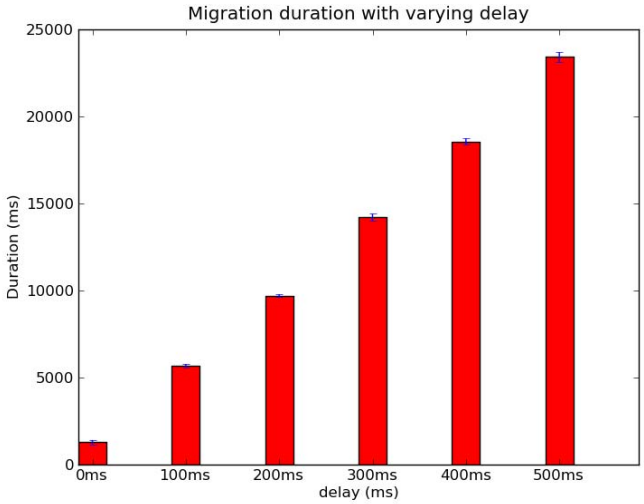


Figure 138 Migration delay when the link delay to and from the OPEN Migration Server is adjusted from 0ms to 500ms plotted with a 95% confidence interval

4.8.4.2. PACKET LOSS

In the second series of experiments the link packet loss was adjusted using the Topology Emulator, from 0% to 10% uniformly distributed random packet loss on each link going to and from the OPEN MSP. For each packet losses setting ten migrations were performed and a Wireshark trace file from the Topology Emulator was saved.

Figure 139shows a plot of the duration of the TCP streams matching the events in Table 40. The variance in the durations increases when the packet loss increases which means it is not possible to tell if there is a specific trend when the packet loss goes from 0% to 10%, since the confidence intervals are overlapping.

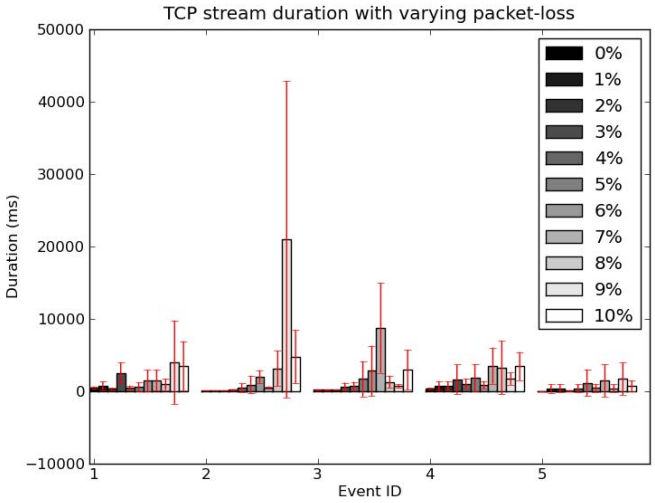


Figure 139 The duration of the TCP connections for each migration event when the packet loss is adjusted from 0% to 10% on the link to and from the OPEN Migration Server

Figure 140 shows a plot of the complete migration delay when the link packet loss is varied from 0% to 10% uniformly distributed random packet loss on the links to and from the OPEN Migration Server. The results show that the migration delay increase to above 10 seconds when the packet loss is high, but the high variance in the results makes it impossible to conclude on any trend when the packet loss is going from 0% to 10%.

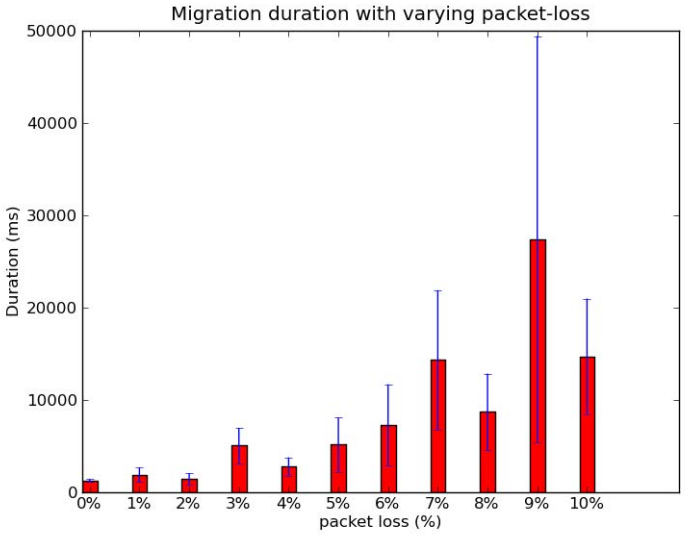


Figure 140 Migration delay when the link packet loss to and from the OPEN Migration Server is adjusted from 0% to 10% plotted with a 95% confidence interval

4.8.4.3. STATE SIZE

In the last series of experiments the state size transferred between the client applications was adjusted from 100bytes to 10.000.000bytes. For comparison the original prototype application for the mobility support prototype uses a state size of approximately 250Kbyte. For each state size setting ten migrations were performed and a Wireshark trace file from the Topology Emulator was saved.

Figure 141 shows a plot of the duration of the TCP streams matching the events in Table 40 when the state size is varied from 1byte to 10.000.000bytes. The results show that when the state size is increased then even 2 and 3 will take longer to complete. Event 2 and 3 are the events where the state is transferred.

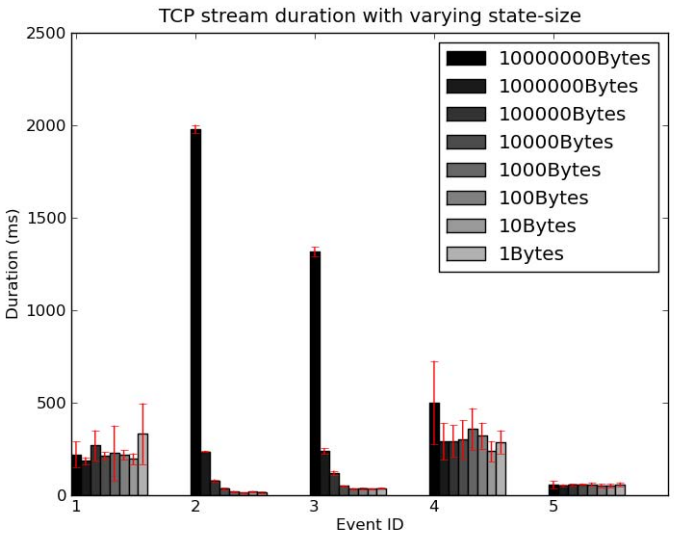


Figure 141 The duration of the TCP connections for each migration event when the packet loss is adjusted from 0% to 10% on the link to and from the OPEN Migration Server

Figure 142 shows a plot of the complete migration delay when the state size transferred between the clients is varied from 1byte to 10.000.000bytes. The results show that for state sizes below 1.000.000bytes the complete migration delay is not affected by the state size, although this number would probably be lower on a lower bandwidth link, since these experiments have been conducted with 100Mbit/s Ethernet links.

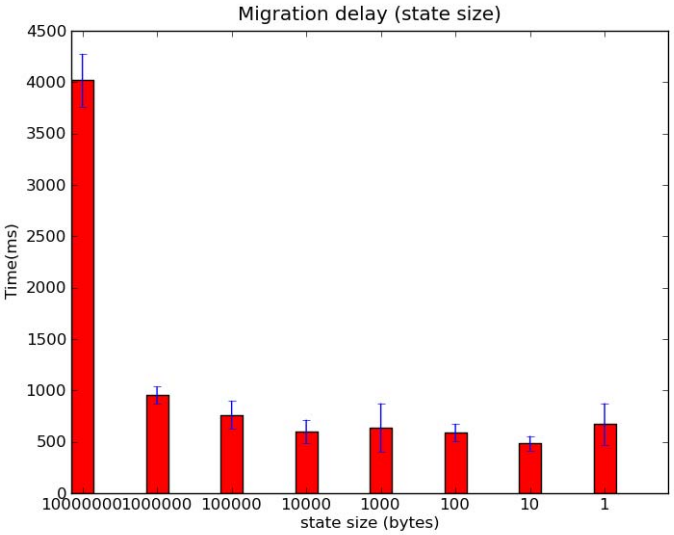


Figure 142 Migration delay when the state size is adjusted from 1Byte to 10.000.000Bytes plotted with a 95% confidence interval

4.8.5. EVALUATION CONCLUSIONS

- The results described in this section concern the evaluation of a multi-network migration scenario considering the Mobility Support module's influence on the overall migration performance. It is not possible to strictly isolate the performance of the Mobility Support module as this only works when interacting with other OPEN modules. Results show a linear increase in migration delay when the link delay is increased. Due to the fact that a migration consists of many TCP connections, the complete migration duration is significantly increased e.g. with a link delay between 100ms and 200ms the migration duration is increased from 1s (optimal link delay) to 5 - 10s.
- When the packet loss increases, the uncertainty of the migration duration increases and the migration delay average is also increased. Again the solution is quite sensitive to packet loss because a lot of independent TCP connections are created and the risk of dropping e.g. SYN packets becomes higher.
- When the state size is increased it does not have an effect until a very large state size, but this limit would be different on a lower bandwidth link.

5. CONCLUSIONS

The innovative aspect targeted by the OPEN European project is a middleware proposal, coupled with on top application prototypes able to support application component migration to different devices. The purpose of this testing iteration was to evaluate the final version of the OPEN Migration Service Platform and to provide directions for possible future developments of the migration technology. For this final evaluation, the three different aspects already evaluated for the first testing iteration have been taken into consideration: usability (end user point of view), programmability and technology (OPEN MSP stakeholders: developers, service providers, etc.).

Considering the usability evaluation, the on top application prototypes were used as probes about how well the migration experience is enabled by the platform in different contexts: the applications are necessary to exploit migration services making them available to users. The evaluation has followed the usability principles defined by the ISO norm: effectiveness, efficiency and satisfaction. Moreover, a comparative analysis about the adaptation features is carried out considering some product already commercially available such as Safari and Opera Mini. At the end of this usability evaluation, the following results have been obtained:

- From the efficiency point of view, the majority of the tasks were autonomously completed by users according to the assigned task lists.
- The time requested by users to execute the task lists is always compliant with the expectations. Moreover, different users require approximately the same time to execute the tasks; this confirms that being able to efficiently use the migration features does not require a specific knowhow.
- The satisfaction evaluation carried out using SUS and PRC approaches highlights how users typically perceive as very innovative the migration idea and its implementation through the OPEN features.
- Regarding competitors' analysis for web page adaptation the prototype results are comparable from the performances point of view with the considered products, but OPEN platform was very appreciated considering the semantic redesign and page splitting features. The users appreciated the balanced amount of information which is put in each split adapted page.

An interesting further improvement suggested by many users in different tests was to simplify the migration mechanism, e.g. the procedure to select which destination device and application components are interested in the migration. This suggestion needs to be considered both from the OPEN MSP point of view and also for on top migratory application user interface development.

The programmability evaluation was executed addressing all the modules which have been considered interesting from programmability point of view in D6.5 [D6.5]. First an assessment phase, based on module documentation provided by owner was carried out, identifying if each module is configurable or extensible through configuration files or interfaces. Then an evaluation about robustness and consistency was performed, checking that effectively the configured features behave as expected. At the end of this programmability evaluation, the following results have been obtained:

- Effectively the new platform release is easy to set up and customize from a system administrator/developer point of view.
- All the considered modules gave a good feedback, resulting effectively configurable in an easy way. The key point is that all the suggestions addressed during the first programmability evaluation result were now implemented.

An interesting further improvement can be to produce a unique configuration tool considering the whole platform. Technically, it could be a sort of OPEN configuration tool that aggregates in a single tool all the required configurations.

The technological evaluation was performed in order to evaluate the final version of the OPEN MSP:

- The most important initial requirements elicited at the beginning of the OPEN project were satisfied. This was checked through test cases that use the on top prototypes developed to verify whether it is possible to satisfy the requirements.
- The indicator evaluations highlight that the OPEN MSP performances are quite good taking in consideration that it is still a prototype. In particular, the migration time, which is a very critical performance indicator confirmed a short platform response time.
- The suggestion about log improvements in order to collect detailed information about the OPEN platform was fully considered: in this last version, logs are carried out with precise timestamps. In this way they track the complete application execution, errors and actions.
- Single modules analysis was carried out in order to understand modules performances in different scenarios. This analysis was also useful for understanding the different components impact in the OPEN Migration Service Platform. Usually a linear behavior was obtained for all the considered modules.

Further improvements could address: migration time, robustness, platform set up and deployment procedure.

6. BIBLIOGRAPHY

[CHA-83] **John Chambers, William Cleveland, Beat Kleiner and Paul Tukey**. Graphical Methods for Data Analysis, Wadsworth. 1983.

[CSQ-95] **CSQU.** http://oldwww.acm.org/perlman/question.cgi?form=CSUQ. 1995.

[D1.1] **Open partners.** Requirements for OPEN Service Platform. 2008.

[D1.3] **Open partners.** Final Requirements for OPEN Service Platform. 2009.

[D6.4] **Open partners.** Testing and Validation Methodology. 2009.

[D6.5] **Open partners.** Evaluation Results. 2009.

[EDU-02] **Joey Benedek and Trish Miner.** Measuring Desirability: New methods for evaluating desirability in a usability lab setting. *Usability Professionals' Association Conference.* Orlando, 2002.

[ENC-00] **Encyclopedia.** Programmability definition. http://encyclopedia2.thefreedictionary.com/programmability

[HID-10] **HIDENETS: Highly Dependable ip-based NETworks and Services project.** http://www.hidenets.aau.dk. 2010.

[HUT-08] **Jeffrey Rubin and Dana Chisnell**. Handbook of Usability Testing. Wiley Publishing, 2008.

[ISO-00] **International Organization for Standardization.** ISO 9241-11: Ergonomic requirements for office work with visual display terminals (VDTs). 2000.

[MAL-99] **David A. Maltz, Pravin Bhagwat.** TCP Splice for application layer proxy performance. Journal of High Speed Networks. Volume 8, Number 3/1999, pp 225-240.

[MON-03] **Douglas C. Montgomery and George C. Runger.** Applied Statistics and Probability for Engineers, 3rd edition. John Wiley, 2002.

[NIS-03] **NIST/SEMATECH.** e-Handbook of Statistical Methods. http://www.itl.nist.gov/div898/handbook/eda/section3/boxplot.htm

[OSG-10] **OSGi Alliance.** http://www.osgi.org. 2010.

[PEA-10] **Wikipedia.** Pearson product-moment correlation coefficient. 2010. http://en.wikipedia.org/wiki/Pearson_product-moment_correlation_coefficient

[QUE-06] **Usability Net.**Questionnaire resources. 2006. http://www.usabilitynet.org/tools/r_questionnaire.htm

[QUI-90] **QUIS.** http://www.lap.umd.edu/QUIS/index.html. 1990.

[SCH-00] **H. Schulzrinne and E. Wedlund.** Application-layer mobility using SIP, SIGMOBILE Mobile Computer Communication Rev., vol. 4, no. 3, pp. 47–57. 2000.

[SHN-04] **Ben Shneiderman and Catherine Plaisant.** Designing the User Interface: Strategies for Effective Human-Computer Interaction (4th Edition). Pearson Addison Wesley, 2004.

[SUM-94] **SUMI.** http://sumi.ucc.ie. 1994.

[SUS-86] **John Brooke**. SUS - A quick and dirty usability scale. 1986.

[USE-01] **USE.** http://www.stcsig.org/usability/newsletter/0110_measuring_with_use.html. 2001.

[W3C-10] **W3C Markup Validation Service.**http://validator.w3.org/. 2010.

[WAM-08] **WAMMI.** http://www.wammi.com. 2008

[WAV-10] **WAVE: Web Accessibility Evaluation tool.** http://wave.webaim.org. 2010.

A. APPENDIX: SYSTEM USABILITY SCALE

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5

B. APPENDIX: PRODUCT REACTION CARDS

The complete set of 118 Product Reaction Cards

Accessible	Creative	Fast	Meaningful	Slow
Advanced	Customizable	Flexible	Motivating	Sophisticated
Annoying	Cutting edge	Fragile	Not Secure	Stable
Appealing	Dated	Fresh	Not Valuable	Sterile
Approachable	Desirable	Friendly	Novel	Stimulating
Attractive	Difficult	Frustrating	Old	Straight Forward
Boring	Disconnected	Fun	Optimistic	Stressful
Business-like	Disruptive	Gets in the way	Ordinary	Time-consuming
Busy	Distracting	Hard to Use	Organized	Time-Saving
Calm	Dull	Helpful	Overbearing	Too Technical
Clean	Easy to use	High quality	Overwhelming	Trustworthy
Clear	Effective	Impersonal	Patronizing	Unapproachable
Collaborative	Efficient	Impressive	Personal	Unattractive
Comfortable	Effortless	Incomprehensible	Poor quality	Uncontrollable
Compatible	Empowering	Inconsistent	Powerful	Unconventional
Compelling	Energetic	Ineffective	Predictable	Understandable
Complex	Engaging	Innovative	Professional	Undesirable
Comprehensive	Entertaining	Inspiring	Relevant	Unpredictable
Confident	Enthusiastic	Integrated	Reliable	Unrefined
Confusing	Essential	Intimidating	Responsive	Usable
Connected	Exceptional	Intuitive	Rigid	Useful
Consistent	Exciting	Inviting	Satisfying	Valuable
Controllable	Expected	Irrelevant	Secure	
Convenient	Familiar	Low Maintenance	Simplistic	

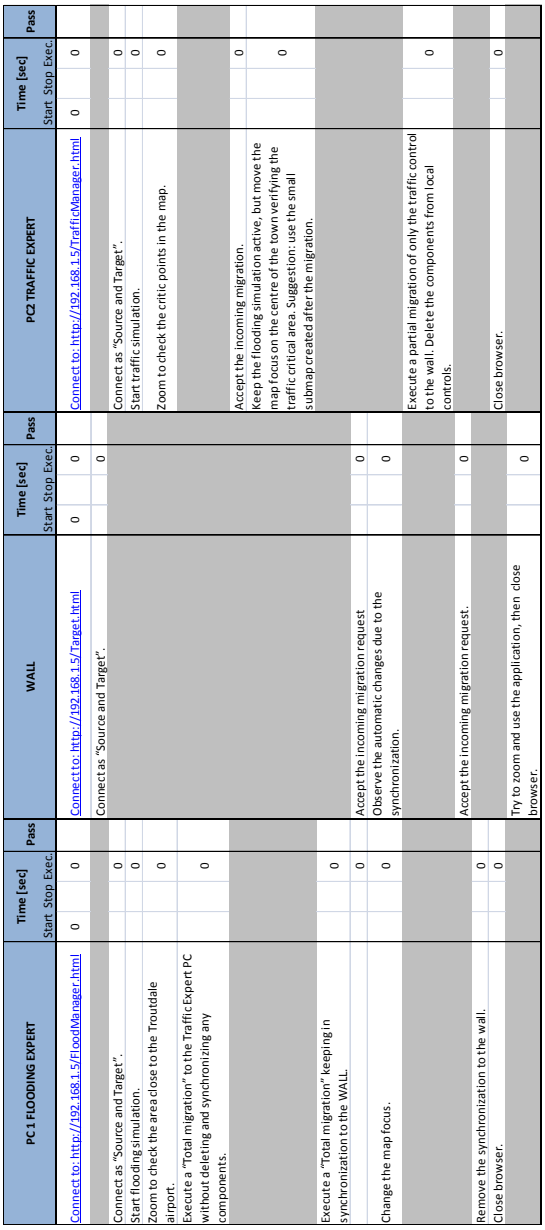
C. APPENDIX: USABILITY TEST DOCUMENTATION

In this Appendix all the detailed data computed for usability tests are reported. They are grouped considering the tested prototype.

C.1. EMERGENCY

Here after the complete data about the usability evaluation of the Emergency prototype are reported.

C.1.1 TASK LIST



PC1 FLOODING EXPERT	Time [sec] Start	Stop	Exec.	Pass	WALL	Time [sec] Start	Stop	Exec.	Pass	PC2 TRAFFIC EXPERT	Time [sec] Start	Stop	Exec.	Pass
Connect to: http://192.168.1.5/FloodManager.html	0		0		Connect to: http://192.168.1.5/Target.html	0		0		Connect to: http://192.168.1.5/TrafficManager.html	0		0	
Connect as "Source and Target".			0		Connect as "Source and Target".			0		Connect as "Source and Target".			0	
Start flooding simulation.			0							Start traffic simulation.			0	
Zoom to check the area close to the Troutdale airport.			0							Zoom to check the critic points in the map.			0	
Execute a "Total migration" to the Traffic Expert PC without deleting and synchronizing any components.			0											
										Accept the incoming migration.			0	
										Keep the flooding simulation active, but move the map focus on the centre of the town verifying the traffic critical area. Suggestion: use the small submap created after the migration.			0	
Execute a "Total migration" keeping in synchronization to the WALL.			0		Accept the incoming migration request			0						
			0		Observe the automatic changes due to the synchronization.			0						
Change the map focus.			0							Execute a partial migration of only the traffic control to the wall. Delete the components from local controls.			0	
					Accept the incoming migration request.			0						
Remove the synchronization to the wall.			0							Close browser.			0	
Close browser.			0		Try to zoom and use the application, then close browser.			0						

Figure 143 Emergency prototype task list used for test execution

A1. You are an emergency expert in Portland. Right now, there is a flood danger in the region, so you are working on a flood simulation in order to prevent damages. Please, start your simulation.	**B1.** You are an emergency expert in Portland. Right now, there is a flood danger in the region, so you are working on a traffic simulation in order to prevent damages. Please, start your simulation.
A2. You are co-working with a colleague of you who is responsible for a traffic simulation and you want to exchange your simulations. Connect to the OPEN platform as target and source.	**B2.** You are co-working with a colleague of you who is responsible for a flood simulation and you want to exchange your simulations. Connect to the OPEN platform as target and source.
A3. Migrate your simulation to your colleague (you will find him as Traffic Manager in your device list). For the migration you don´t need to choose any special options.	
	B3. Your colleague would perhaps want to show you his simulation, so don´t hesitate to accept any requests from him and wait to receive his simulation.
	B4. Migrate your simulation to your colleague (you will find him as Flood Manager in your device list) so that he sees on his PC what you are doing with the simulation on your PC. You can zoom in and out the map in order to show your simulation.
A4. He would perhaps also want to show you his simulation, so don´t hesitate to accept any requests from him.	

Table 41 Task lists used by SAP for a more user empowering test execution

C.1.2 EFFECTIVENESS

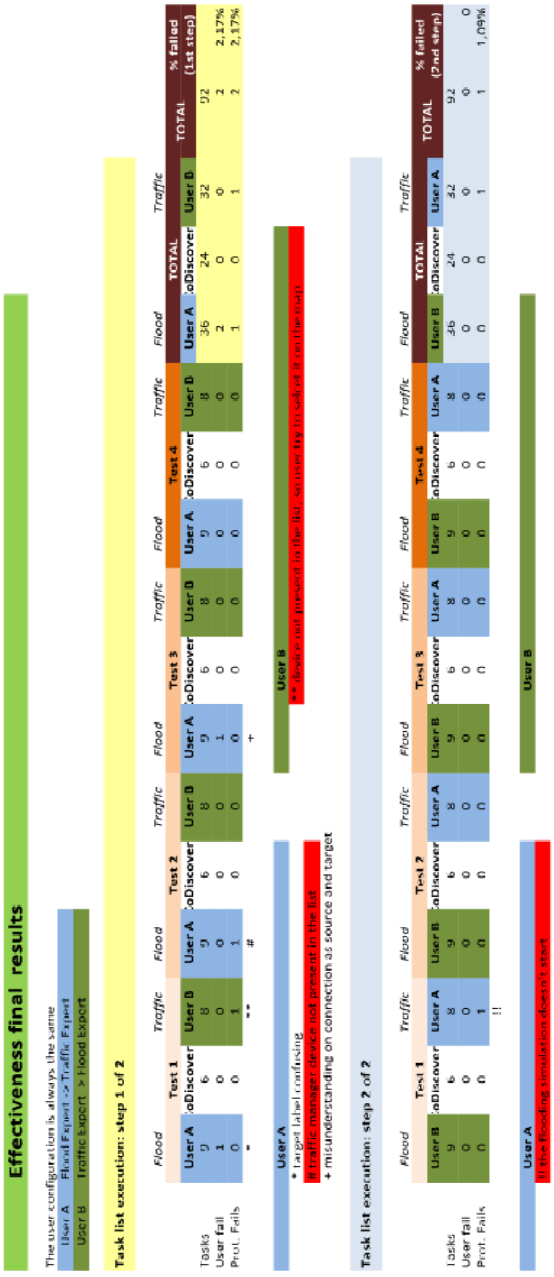


Figure 144 Effectiveness analysis final results

TOTAL	% User failed (both steps)	% Prot. failed	% success (both	% abandone
184	2	3		
	1,09%	1,63%	97,28%	0,00%

Figure 145 Effectiveness analysis final total results

C.1.3 EFFICIENCY

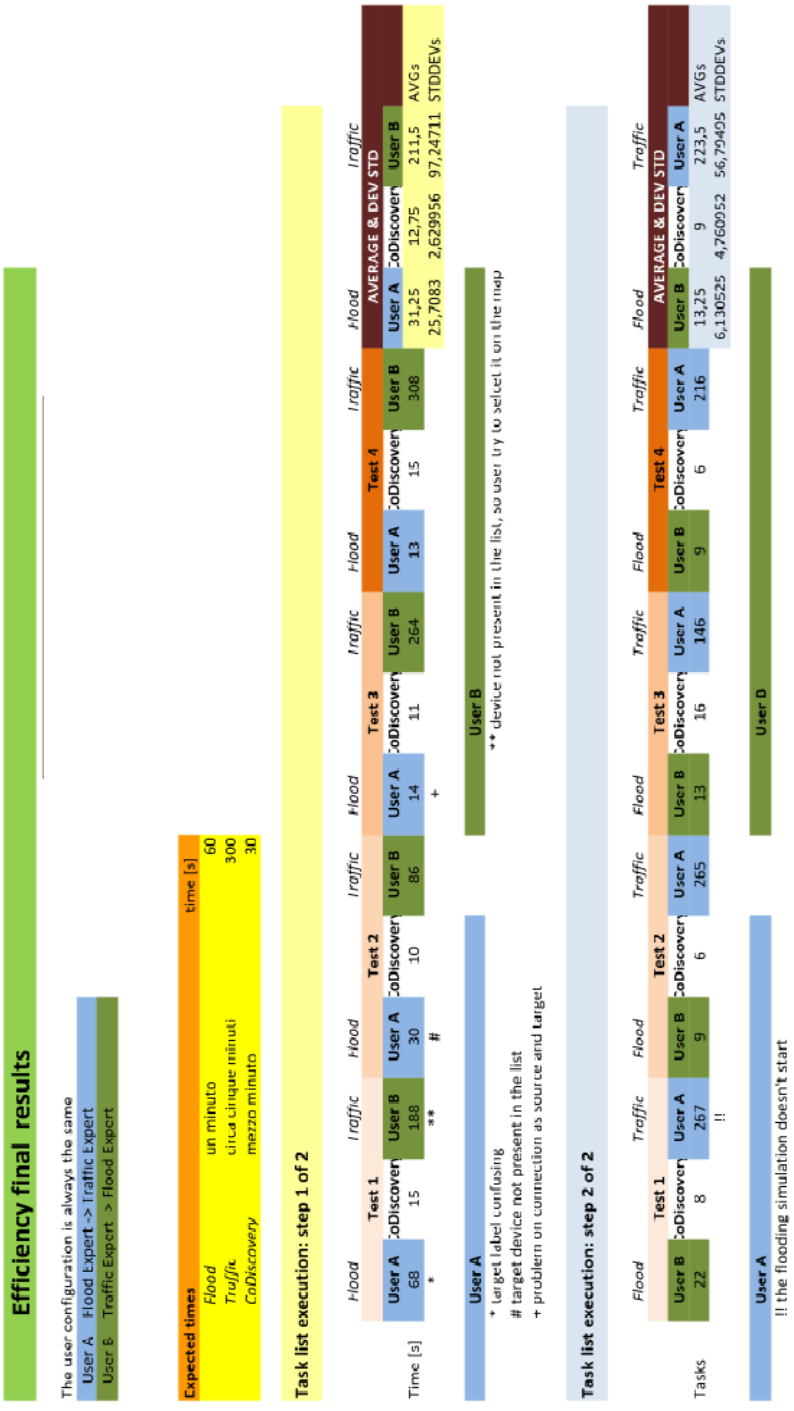


Figure 146 Efficiency roughly collected data

Class [s]	Flood			Traffic			Wall		
	1st	2nd	TOT	1st	2nd	TOT	1st	2nd	TOT
[0; 10)	0	2	2	0	0	0	1	3	4
[10; 25)	2	2	4	0	0	0	3	1	4
[25; 50)	1	0	1	0	0	0	0	0	0
[50; 75)	1	0	1	0	0	0	0	0	0
[75; 100)	0	0	0	1	0	1	0	0	0
[100; 150)	0	0	0	0	1	1	0	0	0
[150; 200)	0	0	0	1	0	1	0	0	0
[200; 300)	0	0	0	0	1	1	0	0	0
[300; 400)	0	0	0	1	2	3	0	0	0
[400; +inf)	0	0	0	1	0	1	0	0	0

Figure 147 Efficiency time class distribution

C.1.4 SATISFACTION - SUS

System Usability Scale (SUS) final results

	Test 1		Test 2		Test 3		Test 4	
User age	34	38	39	36	25	21	38	27
	User A	User B	User A	User B	User A	User B	User A	User B
SUS score	47,5	35	17,5	52,5	75	72,5	22,5	37,5

AVG User A	40,625	DEV STD	22,87022
AVG User B	49,375	DEV STD	14,93475

AVG TOTAL	**45**	**DEV STD**	**21,17107**

The user configuration is always the same

User A	Flood Expert -> Traffic Expert
User B	Traffic Expert -> Flood Expert

Figure 148 Satisfaction using SUS collected data

C.1.5 SATISFACTION - PRC

Product Reaction Card (PRC) final results

	Test 1		Test 2		Test 3		Test 4	
User age	34	38	39	36	25	21	38	27
	User A	User B	User A	User B	User A	User B	User A	User B
1st selection	12	22	19	14	16	10	13	11
2nd selection	5	5	5	5	5	5	5	5

Figure 149 Satisfaction using PRC number of words picked at each step by user

ID	CARD VALUE	Test 1		Test 2		Test 3		Test 4		TOT
		User A	User B	User A	User B	User A	User B	User A	User B	
2	Advanced		x							1
3	Annoying			x						1
4	Appealing		x			x		x		3
6	Attractive			x					x	2
11	Clean						x	x		2
13	Collaborative	x	x	x		x		x		5
14	Comfortable					x				1
15	Compatible								x	1
20	Confusing			x				x	x	3
21	Connected		x	x						2
24	Convenient	x			x	x			x	4
26	Customizable				x					1
27	Cutting edge		x							1
30	Difficult		x	x						2
31	Disconnected	x								1
35	Easy to use					x	x			2
36	Effective					x				1
38	Effortless					x				1
39	Empowering		x	x		x				3
42	Entertaining						x			1
44	Essential						x			1
47	Expected			x						1
49	Fast			x		x	x			3
50	Flexible		x	x						2
51	Fragile		x	x		x		x		4
52	Fresh			x						1
54	Frustrating	x		x	x					3
57	Hard to Use		x					x	x	3
58	Helpful	x	x			x	x	x		5
60	Impersonal				x			x		2
63	Inconsistent			x	x			x	x	4
65	Innovative		x		x					2
67	Integrated		x	x						2
68	Intimidating				x					1
69	Intuitive						x			1
75	Not Secure	x								1
77	Novel	x	x		x					3
81	Organized		x							1
86	Poor quality				x					1
87	Powerful			x				x	x	3
89	Professional							x		1
92	Responsive					x	x			2
94	Satisfying					x				1
97	Slow	x	x		x				x	4
98	Sophisticated		x							1
101	Stimulating	x								1
102	Straight Forward						x			1
103	Stressful				x					1
104	Time-consuming				x					1
105	Time-Saving					x			x	2
106	Too Technical		x							1
109	Unattractive			x						1
111	Unconventional		x							1
112	Understandable			x		x	x			3
113	Undesirable								x	1
115	Unrefined	x	x	x	x			x	x	6
117	Useful	x	x		x	x		x		5
118	Valuable	x	x							2

Figure 150 Satisfaction using PRC number of cards picked at first step by each user

ID	CARD VALUE	Test 1		Test 2		Test 3		Test 4		TOT
		User A	User B	User A	User B	User A	User B	User A	User B	
4	Appealing					It is a type of application that I would like to use also in different contexts and usages				1
6	Attractive								It creates curiosity while using the application	1
11	Clean						The UI is essential and clear, there are not unnecessary commands or controls			1
13	Collaborative	It helps collaboration between experts	It improves the collaboration among people with different expertise	A good idea for fast sharing different kind of data		It can be used with other people and allows working alone and then share the elaborated info				4
14	Comfortable					It allows to share data in an easy way, working in your own environment				1
20	Confusing							The UI is not consistent in terms of menus and symbols	Some interactions with the UI cause unexpected results	2
21	Connected		It must be used through a network connection	The single value of a module doubles when connected to the other						2
31	Disconnected	Sometimes the devices were not correctly connected								1
39	Empowering		It allows doing novel operations with respect to traditional applications							1
44	Essential						There is only what is needed for using the application			1
49	Fast			When it works, browsing the info is fast						1
51	Fragile		Sometimes it stucks, it should be more robust	The system has unexpected behavior				It not very stable and it has unpredictable behavior		3
54	Frustrating	It is not easy to understand how to do things and sometimes the results are unexpected								1
58	Helpful	It is useful for sharing and merging different kind of information					It helps sharing and comparing different kind of data	It is useful because the information can be moved to different devices		3
60	Impersonal							The UI for the two experts should be different because they cover different domains		1

Figure 151 Satisfaction using PRC number of cards picked at second step by each user. It is reported also the motivation given by each user

Cards refers to the Platform
Cards refers to the Applicaiton and not to the paltform

ID	CARD VALUE	Test 1		Test 2		Test 3		Test 4		T PLATFOR
		User A	User B	User A	User B	User A	User B	User A	User B	
4	Appealing					It is a type of application that I would like to use also in different contexts and usages				1
6	Attractive								It creates curiosity while using the application	1
11	Clean						The UI is essential and clear, there are not unnecessary commands or controls			0
13	Collaborative	It helps collaboration between experts	It improves the collaboration among people with different expertise	A good idea for fast sharing different kind of data		It can be used with other people and allows working alone and then share the elaborated info				4
14	Comfortable					It allows to share data in an easy way, working in your own environment				1
20	Confusing							The UI is not consistent in terms of menus and symbols	Some interactions with the UI cause unexpected results	1
21	Connected		It must be used through a network connection	The single value of a module doubles when connected to the other						2
31	Disconnected	Sometimes the devices were not correctly connected								1
39	Empowering		It allows doing novel operations with respect to traditional applications							1
44	Essential						There is only what is needed for using the application			1
49	Fast			When it works, browsing the info is fast						1
51	Fragile		Sometimes it stucks, it should be more robust	The system has unexpected behavior				It not very stable and it has unpredictable behavior		1
54	Frustrating	It is not easy to understand how to do things and sometimes the results are unexpected								1

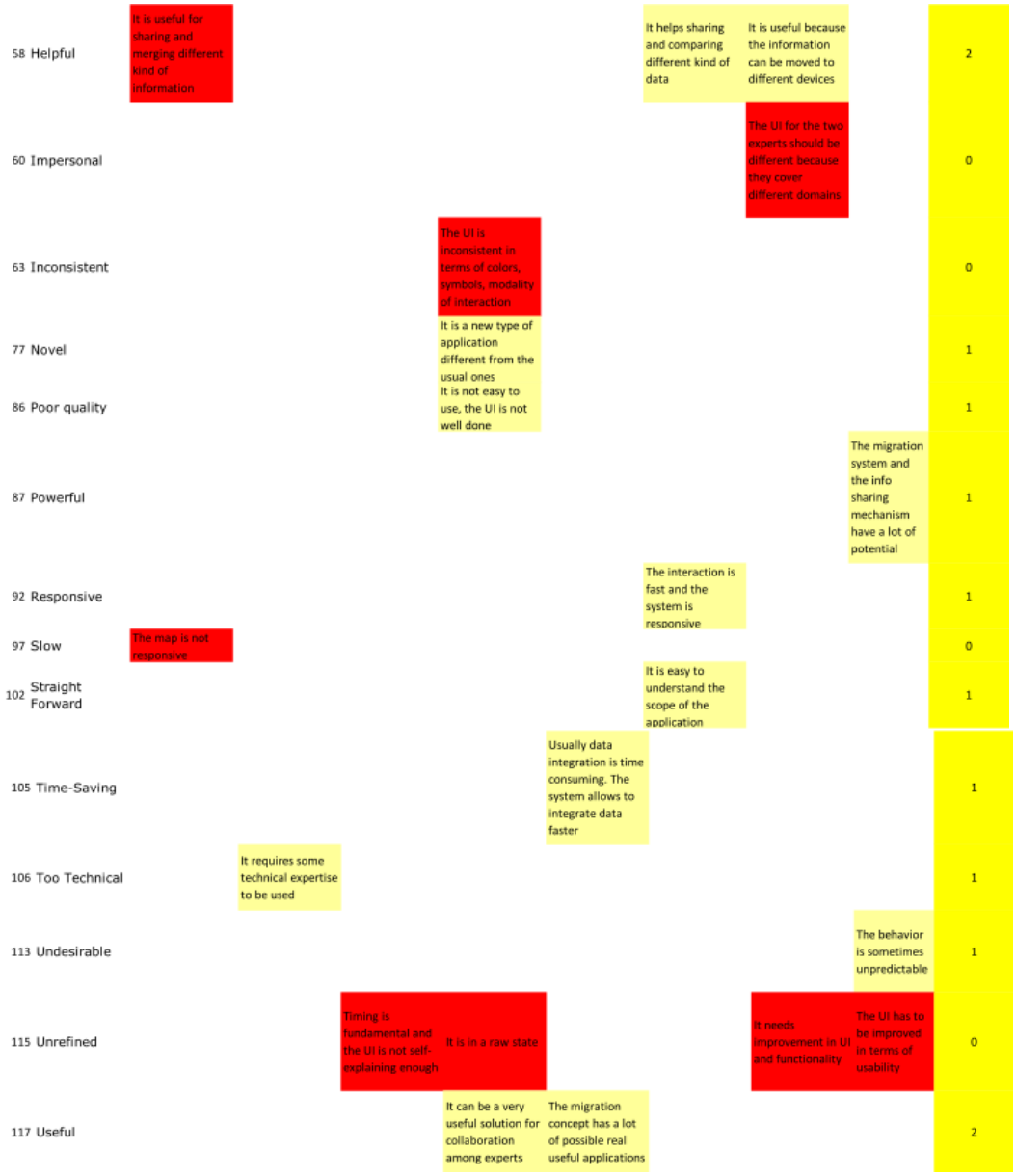


Figure 152 Satisfaction using PRC number of cards picked at second step by each user. It is reported also the motivation given by each user. In RED the motivation strongly related to the application while in YELLOW the ones related to the user migration experience

C.2. SOCIAL GAME

Here after the complete data about the usability evaluation of the Social Game prototype are reported.

C.2.1 TASK LIST

Name of the tester		
	User A	
Mobile	**PC1**	**PC2**
	Start the Social Game interface	
	Login to the Social Game, using as username: "vod1" and password: "vod1"	
	Send the message "Hello from [your name] !" to the chat	
	Migrate partially to another available PC the	
		The IPTV Appears on the other PC
	In betting area select "Best Lap" and "Felipe Massa" as driver	
	Migrate the betting component and the chat component to mobile phone	
Accept the incoming migration and observe that the state is maintained		
Change the first Bet Parameter to "Winner", insert in the text box an import of 50,00 Euro and submit request		
	Exit closing the browser	
Exit closing the browser		Exit closing the browser

Figure 153 Social Game prototype task list used for test execution Group A

Name of the tester		
	User B	
Mobile	**PC1**	**PC2**
	Start the Social Game interface	
	Login to the Social Game, using as username: "vod1" and password: "vod1"	
	Send the message "Hello from [your name] !" to the chat	
	Migrate partially to another available PC the IPTV	
		The IPTV Appears on the other PC
	In betting area select "Best Lap" and "Felipe Massa" as driver	
	Migrate the betting component and the chat component to mobile phone	
Accept the incoming migration and observe that the state is maintained		
Change the first Bet Parameter to "Winner", insert in the text box an import of 50,00 Euro and submit request		
	Exit closing the browser	
Exit closing the browser		Exit closing the browser

Figure 154 Social Game prototype task list used for test execution Group B

C.2.2 EFFECTIVENESS

Effectiveness

Partial Migration (Social Game)

	Group A										Group B									
	u1		u2		u3		u4		u5		u1		u2		u3		u4		u5	
	user	app	user	app	user	app	user	app	user	app	user	app	user	app	user	app	user	app	user	app
9	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0

Figure 155 Effectiveness analysis final results

C.2.3 EFFICIENCY

Efficiency

Partial Migration (Social Game)

Group A					Group B				
u1	u2	u3	u4	u5	u1	u2	u3	u4	u5
user	user	user	user	user	user	user	user	user	user
460	336	449	325	411	257	463	401	360	313

Figure 156 Efficiency roughly collected data

	Partial Mig.(Social Game)		
Class [s]	Group A	Group B	Tot
[0; 100)	0	0	0
[100; 200)	0	0	0
[200; 350)	2	2	4
[350; 500)	3	3	6
[500; 700)	0	0	0
[700; 900)	0	0	0
[900; +inf)	0	0	0
Tot	5	5	

Figure 157 Efficiency time class distribution

C.2.4 SATISFACTION - SUS

System Usability Score (SUS)

	Group A					Group B				
	u1	u2	u3	u4	u5	u1	u2	u3	u4	u5
age	32	27	41	34	27	28	29	28	27	27

Partial Migration (Social Game)

	Group A					Group B					All	
	u1	u2	u3	u4	u5	u1	u2	u3	u4	u5		
score	72,50	80,00	57,50	70,00	77,50	82,50	65,00	67,50	97,50	80,00		
	Avg	DevStd				Avg	DevStd				Avg	DevStd
	71,50	8,767839				78,50	13,06235				75,00	11,11805

Figure 158 Satisfaction using SUS collected data

C.3. WEB PAGE MIGRATION

Here after the complete data about the usability evaluation of the web page migration are reported.

C.3.1 TASK LIST

Name of the tester	
User B	
Mobile	**PC**
Open the browser and go to http://www.w3.org/Help/Account/Request/Public (via the migration proxy)	
In the "Public account Request Form" fill the first name and the last name	
Total migrate to PC	
	Accept migration and then check that the entered input is still available
	Complete the form by filling in the two remaining fields. (Do NOT select "Create
	Total migrate to MOBILE
Exit from the browser.	
	Open the browser and go to the "Advanced Search" page of Ebay (via the migration proxy): http://shop.ebay.com/ebayadvsearch/
	Specify in the search form the parameters for finding "guitar drums" (exact words, exact order), in the category "Musical Instruments", in such a way to have results both with Paypal accepted and best offer.
	Total migrate to MOBILE
Accept migration and then check that the entered input is still available	

Figure 159 Web page total migration prototype task list used for test execution Group A

Name of the tester	
User B	
Mobile	**PC**
Open the browser and go to http://www.w3.org/Help/Account/Request/Public (via the migration proxy)	
In the "Public account Request Form" fill the first name and the last name	
Total migrate to PC	
	Accept migration and then check that the entered input is still available
	Complete the form by filling in the two remaining fields. (Do NOT select "Create
	Total migrate to MOBILE
Exit from the browser.	
	Open the browser and go to the "Advanced Search" page of Ebay (via the migration proxy): http://shop.ebay.com/ebayadvsearch/
	Specify in the search form the parameters for finding "guitar drums" (exact words, exact order), in the category "Musical Instruments", in such a way to have results both with Paypal accepted and best offer.
	Total migrate to MOBILE
Accept migration and then check that the entered input is still available	
Continue the specification of the search in the mobile device by selecting the options to: i) exclude the word "acoustic" from your search; ii) select "Auction" for the Buying formats, and iii) select "euro" for the currency	
Exit from the browser.	

Figure 160 Web page total migration prototype task list used for test execution Group B

C.3.2 EFFECTIVENESS

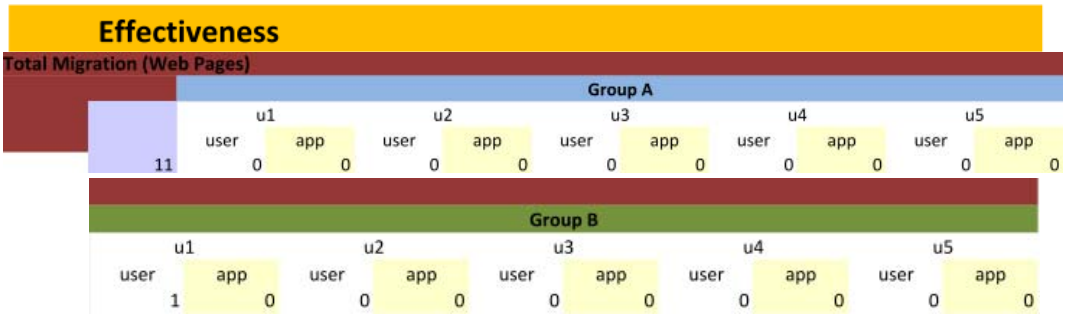


Figure 161 Effectiveness analysis final results

C.3.3 EFFICIENCY

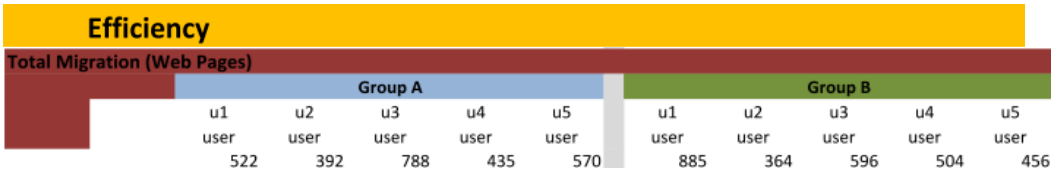


Figure 162 Efficiency roughly collected data

	Total Mig. (Web Pages)		
Class [s]	Group A	Group B	Tot
[0; 100)	0	0	0
[100; 200)	0	0	0
[200; 350)	0	0	0
[350; 500)	2	2	4
[500; 700)	2	2	4
[700; 900)	1	1	2
[900; +inf)	0	0	0
Tot	5	5	

Figure 163 Efficiency time class distribution

C.3.4 SATISFACTION - SUS

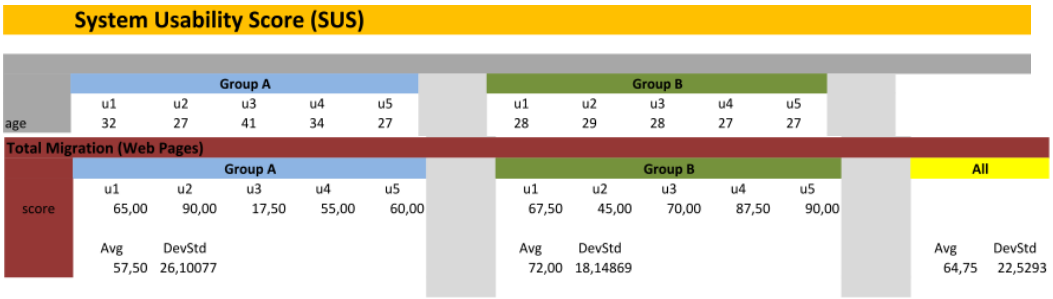


Figure 164 Satisfaction using SUS collected data

C.4. MIGRATION PRC

Here after the complete data about the usability evaluation of the web page migration are reported.

		Group A					Group B					TOT
ID	CARD	u1	u2	u3	u4	u5	u1	u2	u3	u4	u5	
1	Accessible	x										1
2	Advanced		x					x	x			3
3	Annoying			x					x			2
4	Appealing				x							1
6	Attractive		x								x	2
8	Business-like					x						1
12	Clear		x									1
17	Complex			x								1
21	Connected	x										1
25	Creative		x									1
26	Customizable					x					x	2
27	Cutting edge					x						1
29	Desirable		x									1
35	Easy to use	x	x				x	x		x		5
36	Effective					x			x			2
37	Efficient						x					1
39	Empowering					x						1
44	Essential								x			1
48	Familiar		x									1
49	Fast	x										1
50	Flexible					x	x	x		x	x	5
52	Fresh			x								1
53	Friendly									x		1
54	Frustrating			x	x							2
57	Hard to Use			x								1
58	Helpful		x								x	2
61	Impressive					x						1
65	Innovative		x	x		x	x	x		x	x	7
69	Intuitive		x		x		x			x		4
73	Meaningful	x										1
75	Not Secure							x				1
77	Novel					x						1
79	Optimistic								x			1
87	Powerful					x						1
89	Professional					x						1
91	Reliable	x										1
92	Responsive	x										1
93	Rigid			x								1
94	Satisfying						x					1
97	Slow			x	x		x		x	x		5
98	Sophisticated							x			x	2
101	Stimulating			x								1
103	Stressful			x					x			2
105	Time-Saving					x		x				2
111	Unconventional		x									1
116	Usable				x							1
117	Useful				x			x	x		x	4

Figure 165 Satisfaction using PRC number of cards picked at first step by each user

ID	CARD VALUE	Group A					Group B					TOT
		u1	u2	u3	u4	u5	u1	u2	u3	u4	u5	
1	Accessible	everything I see I can use it; it's because of the simplicity of the UI especially on the desktop device										1
2	Advanced							it gives me the idea of an advanced system	because it is not something you can see everyday, it is novel			2
3	Annoying								because I couldn't click in the right place (selection for partial migration)			1
4	Appealing				the idea of the migration both total and partial is appealing because I would like to use it in various situations (eg: work, tourism, leisure, ..)							1
6	Attractive		the idea of partial migration is stimulating								because of the type of novel feature that provides	2
12	Clear		we reach the objective with a few steps									1
21	Connected	it was connected with respect to: i)the content was the same on the different devices ii) there was not problem in the migration itself iii) the UI elements were connected each other										1
27	Cutting edge					I've never seen before a system like this						1
35	Easy to use	I was not familiar, so the first migration was more difficult and I din't know what to do, the second migration I kew what to do	from the first use it is easy to be used				because the steps to do the migration were easy to use and simple	it is not easy to be used at the beginning, but after a while is easy to use		because once you have understood the mechanism it is easy to use		5
36	Effective								because I managed to do what I needed to do			1

44	Essential						because it will make your life easier beacsue you can migrate almost everywhere			1
50	Flexible			because the partial migration allows to select the component to migrate	especially the partial migration is very flexible	it allows to modify the data at any moment even if you change the device		because you can use different devices and do the migration of the various components	because it adapts to different devices	5
52	Fresh		a new approach for the web							1
54	Frustrating		because is not very intuitive	the change of colour of the different parts of the page, together the un-selection of the different parts of the page to do a total migration was really frustrating						2
57	Hard to Use		especially the total migration is complex to use							1
58	Helpful								especially the partial migration is useful beacuse you can select what you actually would like to migrate/to bring with you/move	1
65	Innovative	especially the selection provided by partial migration is innovative			especially the partial migration is innovative			because especially the partial migration is innovative feature		3
69	Intuitive			moving from one tab panel to another one in order to do the various steps (very few) of the migration was intuitive				in general it is very intuitive		2
87	Powerful				because it gives the possibility to continue the interaction on different devices					1
89	Professional				because I think that especially professional users will benefit from this system					1

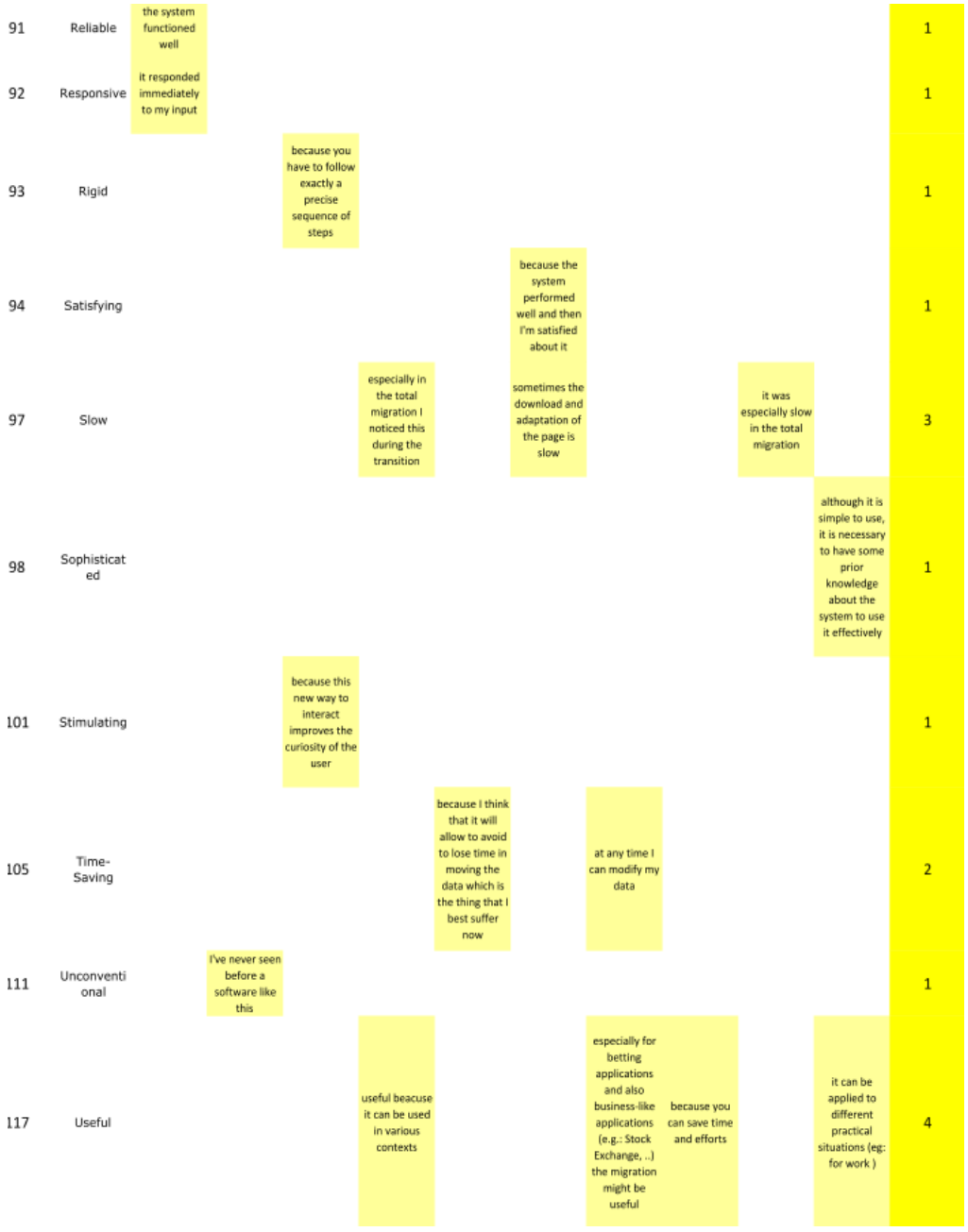


Figure 166 Satisfaction using PRC number of cards picked at second step by each user. It is reported also the motivation given by each user

LEGENDA (interpretazione fatta in base ai comemnti, notare che alcuni sono border line)

Cards refers to the Platform

Cards refers to the Applicaiton and not to the paltform

ID	CARD VALUE	Group A					Group B					TOT
		u1	u2	u3	u4	u5	u1	u2	u3	u4	u5	
1	Accessible	everything I see I can use it; it's because of the simplicity of the UI especially on the desktop device										0
2	Advanced							it gives me the idea of an advanced system	because it is not something you can see everyday, it is novel			2
3	Annoying								because I couldn't click in the right place (selection for partial migration)			0
4	Appealing				the idea of the migration both total and partial is appealing because I would like to use it in various situations (eg: work, tourism, leisure, ..)							1
6	Attractive		the idea of partial migration is stimulating								because of the type of novel feature that provides	2
12	Clear		we reach the objective with a few steps									1
21	Connected	it was connected with respect to: i)the content was the same on the different devices ii) there was not problem in the migration itself iii) the UI elements were connected each other										1
27	Cutting edge					I've never seen before a system like this						1
35	Easy to use	I was not familiar, so the first migration was more difficult and I din't know what to do, the second migration I kew what to do	from the first use it is easy to be used				because the steps to do the migration were easy to use and simple	it is not easy to be used at the beginning, but after a while is easy to use		because once you have understood the mechanism it is easy to use		5


36
Effective
because I managed to do what I needed to do
1
44
Essential
because it will make your life easier beacsue you can migrate almost everywhere
1
50
Flexible
because the partial migration allows to select the component to migrate
especially the partial migration is very flexible
it allows to modify the data at any moment even if you change the device
because you can use different devices and do the migration of the various components
because it adapts to different devices
5
52
Fresh
a new approach for the web
1
54
Frustrating
because is not very intuitive
the change of colour of the different parts of the page, together the un-selection of the different parts of the page to do a total migration was really frustrating
1
57
Hard to Use
especially the total migration is complex to use
1
58
Helpful
especially the partial migration is useful beacuse you can select what you actually would like to migrate/to bring with you/move
1
65
Innovative
especially the selection provided by partial migration is innovative
especially the partial migration is innovative
because especially the partial migration is innovative feature
3
69
Intuitive
moving from one tab panel to another one in order to do the various steps (very few) of the migration was intuitive
in general it is very intuitive
2
87
Powerful
because it gives the possibility to continue the interaction on different devices
1
89
Professional
because I think that especially professional users will benefit from this system
1
91
Reliable
the system functioned well
1
92
Responsive
it responded immediately to my input
1

Figure 167 Satisfaction using PRC number of cards picked at second step by each user. It is reported also the motivation given by each user. In RED the motivation strongly related to the application while in YELLOW the ones related to the user migration experience

C.5. ADAPTATION COMPARISON

Here after the complete data about the usability evaluation of the web page migration are reported.

C.5.1 EFFECTIVENESS

Safari

		errors			Abandoned				
	tasks tot	User error	Application	Successfull		user	app	successfull	abandoned
part 1	72	16	0		0	22,22%	0,00%	77,78%	0,00%
part 2	90	0	0		0	0,00%	0,00%	100,00%	0,00%
part 3	90	0	0		0	0,00%	0,00%	100,00%	0,00%
tot	252	16	0	236	0	6,35%	0,00%	93,65%	0,00%

WebUIAdaptation

		errors			Abandoned				
	tasks tot	User error	Application	Successfull		user	app	successfull	abandoned
part 1	72	0	1		0	0,00%	1,39%	98,61%	0,00%
part 2	90	0	7		0	0,00%	7,78%	92,22%	0,00%
part 3	90	0	2		0	0,00%	2,22%	97,78%	0,00%
tot	252	0	10	242	0	0,00%	3,97%	96,03%	0,00%

OperaMini

		errors			Abandoned				
	tasks tot	User error	Application	Successfull		user	app	successfull	abandoned
part 1	72	7	1		0	9,72%	1,39%	88,89%	0,00%
part 2	90	0	0		0	0,00%	0,00%	100,00%	0,00%
part 3	90	0	0		0	0,00%	0,00%	100,00%	0,00%
tot	252	7	1	244	0	2,78%	0,40%	96,83%	0,00%

Figure 168 Effectiveness analysis final total results

C.5.2 EFFICIENCY

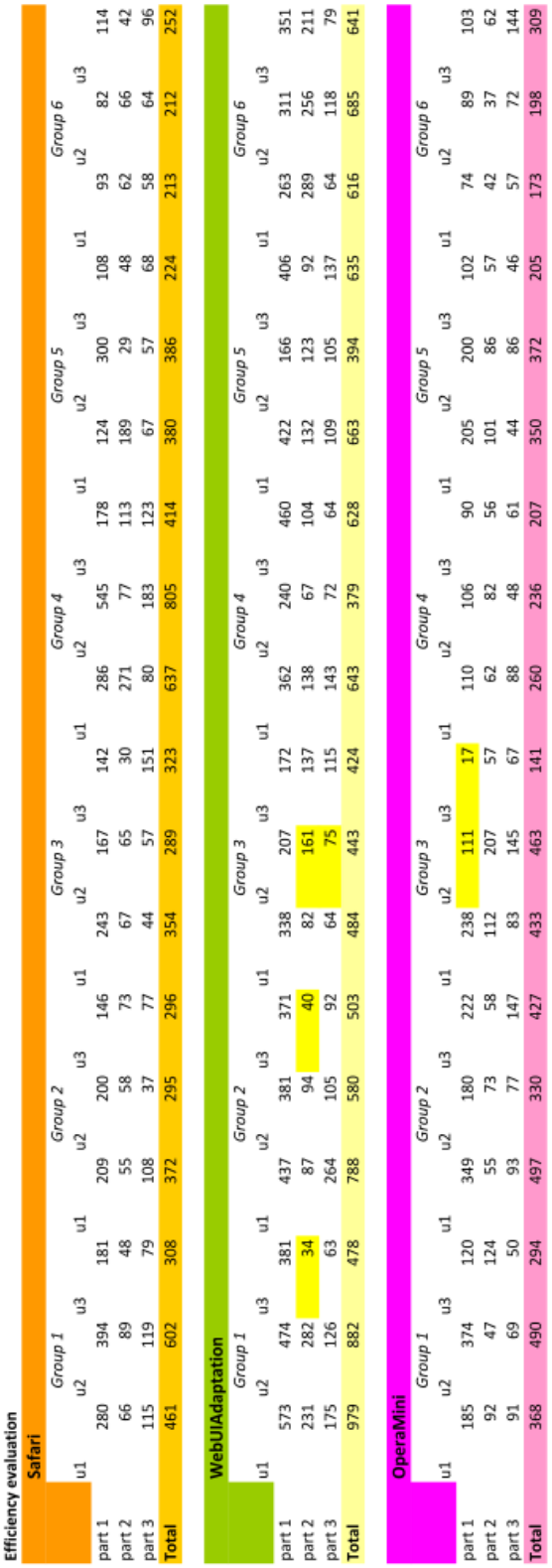


Figure 169 Efficiency roughly collected data

Class [s]	Safari	bUIAdapta	OperaMini
[0; 100)	0	0	0
[100; 200)	0	0	3
[200; 300)	7	0	5
[300; 400)	6	2	5
[400; 500)	2	4	5
[500; 600)	0	2	0
[600; 750)	2	7	0
[750; 1000	1	3	0
[1000; 125	0	0	0
[1250; + in	0	0	0
Total	**18**	**18**	**18**

Figure 170 Efficiency time class distribution

C.5.3 SATISFACTION AD-HOC QUESTIONNAIRE

Adaptation usability evaluation

Questionnaire

Date (gg/mm/yyyy)_____Time (hh:mm)_____Place_____

Section I: General information

In this section a set of general information about you are required.

Gender: ☐ Male ☐ Female

How old are you? _____

What is your qualification?

☐ Primary School
☐ Secondary School
☐ High School
☐ Bachelor degree
☐ Master degree
☐ Phd

Have you already surfed the Internet using a *desktop PC*? ☐ Yes ☐ No

How frequently? ☐ Yearly ☐ Monthly ☐ Weekly ☐ Daily

Have you already surfed the Internet using a *mobile device*? ☐ Yes ☐ No

How frequently? ☐ Yearly ☐ Monthly ☐ Weekly ☐ Daily

Have you already accessed *British Airways site (*http://www.ba.com*) from a desktop PC*?

☐ Yes ☐ No

Have you already accessed *British Airways site from a mobile device*?

☐ Yes ☐ No

Have you already accessed *Ebay site (*http://www.ebay.com *) from a desktop PC*?

☐ Yes ☐ No

Have you already accessed *Ebay site from a mobile device*? ☐ Yes ☐ No

Have you already accessed *W3c site (*http://www.w3.org*) from a desktop PC*?

☐ Yes ☐ No

Have you already accessed *W3c site from a mobile device*? ☐ Yes ☐ No

How familiar are you with the use of IPhone?

☐ Never used before ☐ A bit familiar ☐ Familiar ☐ Very familiar

How familiar are you with the use of other touch-based smartphones?

☐ Never used before ☐ A bit familiar ☐ Familiar ☐ Very familiar

Section II: Usability evaluation

Fill in only one response for each question below:

Very Poor (Low) = 1, Poor = 2, Fair = 3, Good = 4, Very good (High) = 5

If you have some additional comments you can add them in the available space below each question, remember that they are not mandatory.

1. Is it clear how the considered web sites are organized?

Safari	WebUIAdaptation	OperaMini
Poor Very Good 1 2 3 4 5	Poor Very Good 1 2 3 4 5	Poor Very Good 1 2 3 4 5
Free comments: _____ _____ _____	Free comments: _____ _____ _____	Free comments: _____ _____ _____

2. Is it easy to identify the different elements into the web pages?

Safari	WebUIAdaptation	OperaMini
Poor Very Good 1 2 3 4 5	Poor Very Good 1 2 3 4 5	Poor Very Good 1 2 3 4 5
Free comments: _____ _____ _____	Free comments: _____ _____ _____	Free comments: _____ _____ _____

2.a.What about menus? Are they comprehensible?

Safari	WebUIAdaptation	OperaMini
Poor Very Good 1 2 3 4 5	Poor Very Good 1 2 3 4 5	Poor Very Good 1 2 3 4 5
Free comments: _____ _____ _____	Free comments: _____ _____ _____	Free comments: _____ _____ _____

2.b. What about links? Are they easily identifiable in the page?

Safari	WebUIAdaptation	OperaMini
Poor Very Good 1 2 3 4 5	Poor Very Good 1 2 3 4 5	Poor Very Good 1 2 3 4 5
Free comments: _____ _____ _____	Free comments: _____ _____ _____	Free comments: _____ _____ _____

2.c.What about forms? Is it simple to understand and correctly compile the form fields?

Safari	WebUIAdaptation	OperaMini
Poor Very Good 1 2 3 4 5	Poor Very Good 1 2 3 4 5	Poor Very Good 1 2 3 4 5
Free comments: _____ _____ _____	Free comments: _____ _____ _____	Free comments: _____ _____ _____

2.d. What about images? Are they comprehensible?

Safari	WebUIAdaptation	OperaMini
Poor Very Good 1 2 3 4 5	Poor Very Good 1 2 3 4 5	Poor Very Good 1 2 3 4 5
Free comments: _____ _____ _____	Free comments: _____ _____ _____	Free comments: _____ _____ _____

2.e. What about data tables? Are they clearly readable?

Safari	WebUIAdaptation	OperaMini
Poor Very Good 1 2 3 4 5	Poor Very Good 1 2 3 4 5	Poor Very Good 1 2 3 4 5
Free comments: _____ _____ _____	Free comments: _____ _____ _____	Free comments: _____ _____ _____

3. Do you think that is correctly balanced the quantity of information available in a single web page?

Safari	WebUIAdaptation	OperaMini
Poor Very Good 1 2 3 4 5	Poor Very Good 1 2 3 4 5	Poor Very Good 1 2 3 4 5
Free comments: _____ _____ _____	Free comments: _____ _____ _____	Free comments: _____ _____ _____

4. What do you think about the time required to obtain the adapted version of the web page?

Safari	WebUIAdaptation	OperaMini
Poor Very Good	Poor Very Good	Poor Very Good
1 2 3 4 5	1 2 3 4 5	1 2 3 4 5

5. What do you think about the overall usability of the adapted web site?

Safari	WebUIAdaptation	OperaMini
Poor Very Good	Poor Very Good	Poor Very Good
1 2 3 4 5	1 2 3 4 5	1 2 3 4 5

Section III: Page Splitting support (WebUIAdaptation)

In this section you are required to evaluate page splitting, a feature supported only by the WebUIAdaptation module.

1. If a web page is split in more pages, how is the navigability of the split version of the page?

WebUIAdaptation				
Poor				Very Good
1	2	3	4	5

Free comments:

_____

_____

_____

2. In which cases do you think that page splitting is useful?

……

……

……

Section IV: Overall Strong and Weak points

In this section you are required to put at least 1 (but it is possible up to 3) strong and weak points for each adaptation modality that you have tried.

(*) means that it is mandatory.

Strong points:

Safari	WebUIAdaptation	OperaMini
1. *_____	1. *_____	1. *_____
2. _____	2. _____	2. _____
3. _____	3. _____	3. _____

Weak points:

Safari	WebUIAdaptation	OperaMini
1. *_____	1. *_____	1. *_____
2. _____	2. _____	2. _____
3. _____	3. _____	3. _____

D. APPENDIX: TECHNOLOGICAL TEST CASES

The following requirement is implicitly considered in all test cases. It is not repeated because it is clear that this requirement is success for all the test cases. Note that it is so because it represents the innovative idea on which the entire platform and this project is developed.
[5.01.03] The OPEN Migration Service Platform must provide for several types of migration of UI elements or data:

- TOTAL: all components of the application migrate
- PARTIAL: only a part of the application migrates. The migration can be from multiple source devices to multiple target devices.

D.1. CONTEXT, DISCOVERY AND MIGRATION

D.1.1 VERIFIABLE

<table>
<tr><td>ID</td><td>FE01</td><td>Status</td><td>Passed</td></tr>
<tr><td>Executor</td><td>Vodafone Team</td><td>Execution date</td><td>5 May 2010</td></tr>
<tr><td>Item</td><td>Context Awareness</td><td>Prototype</td><td>Social Game
Emergency
PacMan
Web Page Migration</td></tr>
<tr><td>Description</td><td colspan="3">[20] The OPEN Migration Service Platform must provide applications and users with information about devices in their vicinity.</td></tr>
<tr><td>Input</td><td colspan="3">Device A and device B can be (note that not all prototypes support all devices): <A=PC, B=PC>; <A=PC, B=PDA>; <A=PDA, B=PC>

<table>
<tr><th>Num</th><th>Device</th><th>Action</th></tr>
<tr><td>1</td><td>A</td><td>Device A is on</td></tr>
<tr><td>2</td><td>B</td><td>Device B is off</td></tr>
<tr><td>3</td><td>A</td><td>The application is running on device A</td></tr>
<tr><td>4</td><td>A</td><td>The user asks for devices to migrate</td></tr>
<tr><td>5</td><td>B</td><td>Switch on the device B and its OrchestrationClient</td></tr>
<tr><td>6</td><td>A</td><td>The user asks for device to migrate</td></tr>
</table></td></tr>
<tr><td>Expected output</td><td colspan="3">Device A and device B can be (note that not all prototypes support all devices): <A=PC, B=PC>; <A=PC, B=PDA>; <A=PDA, B=PC>

<table>
<tr><th>Num</th><th>Device</th><th>Expected output</th></tr>
<tr><td>1</td><td>A</td><td></td></tr>
<tr><td>2</td><td>B</td><td></td></tr>
<tr><td>3</td><td>A</td><td></td></tr>
<tr><td>4</td><td>A</td><td>No target devices available</td></tr>
<tr><td>5</td><td>B</td><td></td></tr>
<tr><td>6</td><td>A</td><td>At least device B available as target</td></tr>
</table></td></tr>
<tr><td>Actual output</td><td colspan="3">[Emergency] as device A and device B <A=PC, B=PC>;
<table>
<tr><th>Num</th><th>Device</th><th>Actual output</th></tr>
<tr><td>1</td><td>A</td><td></td></tr>
<tr><td>2</td><td>B</td><td></td></tr>
<tr><td>3</td><td>A</td><td></td></tr>
<tr><td>4</td><td>A</td><td>No target devices available</td></tr>
</table></td></tr>
</table>

	5 B 6 A Device B available as target
General consideration	

ID	FE02	**Status**	*Passed*
Executor	Vodafone Team	**Execution date**	5 May 2010
Item	Context Awareness	**Prototype**	Social Game Emergency
Description	[22] The OPEN Migration Service Platform must provide applications and users with context information. [136] The OPEN Migration Service Platform must collect environmental information, using all possible kinds of sources (sensor, interaction panel, network parameters).		
Input	Prototype is running on devices. It means that on the screen the complete application user interface is available without any error.		
Expected output	Context information available in the considered prototypes: **Prototype** / **Context information** Social Game / Name of the registered devices Emergency Scenario / IP address of the registered devices		
Actual output	Context information experienced in the considered prototypes: **Prototype** / **Context information** Social Game / PC-OPEN Emergency Scenario / 192.168.1.5, 192.168.1.10, 192.168.1.12		
General consideration	The OPEN MSP correctly manages the context information. The kind of available information strongly depends from the data that the device is possible to detect and record.		

<table>
<tr><td>ID</td><td>FE03</td><td>Status</td><td>Passed</td></tr>
<tr><td>Executor</td><td>Vodafone Team</td><td>Execution date</td><td>5 May 2010</td></tr>
<tr><td>Item</td><td>Migration</td><td>Prototype</td><td>Emergency
Social Game [PARTIAL only]
PacMan [TOTAL only]</td></tr>
<tr><td>Description</td><td colspan="3">[44] The OPEN Migration Service Platform must enable users to migrate the user interface to another device, fully or partially.</td></tr>
<tr><td>Input</td><td colspan="3">Device A and device B can be:
<A=PC, B=PC>; <A=PC, B=PDA>; <A=PDA, B=PC>

<table><tr><th>Num</th><th>Device</th><th>Action</th></tr><tr><td>1</td><td>A</td><td>Device A is on and registered to OPEN MSP</td></tr><tr><td>2</td><td>B</td><td>Device B is on and registered to OPEN MSP</td></tr><tr><td>3</td><td>A</td><td>Migrate a component or the whole application to device B</td></tr></table></td></tr>
<tr><td>Expected output</td><td colspan="3">The application or a part of it runs on device B with an adapted interface.</td></tr>
<tr><td>Actual output</td><td colspan="3">[Emergency] the “Flooding Control” migrate completely to the Target device, while the “Traffic Control” migrate partially, keeping synchronization to the Target device.

[Social Game] the “betting” area migrates on the mobile device.

[PacMan] the complete PacMan application migrate to mobile device.</td></tr>
<tr><td>General consideration</td><td colspan="3"></td></tr>
</table>

<table>
<tr><td>ID</td><td>FE04</td><td>Status</td><td>Passed</td></tr>
<tr><td>Executor</td><td>Vodafone Team</td><td>Execution date</td><td>5 May 2010</td></tr>
<tr><td>Item</td><td>Platform initiated Migration</td><td>Prototype</td><td>Emergency</td></tr>
<tr><td>Description</td><td colspan="3">[82] The OPEN platform must provide four types of migration initiation:</td></tr>
</table>

<table>
<tr><td></td><td>Platform initiated
The platform proposes the destination device and the UI components to migrate;
The user can confirm or reject the migration;
In case the user rejects the migration, she/he can eventually proceed with a user-initiated migration.</td></tr>
<tr><td>Input</td><td>Set up a RFID reader connected to a device A and an RFID tag connected to a device B.
<A=PC, B=PDA>
<table><tr><th>Num</th><th>Device</th><th>Action</th></tr><tr><td>1</td><td>A</td><td>Device A is on and registered to OPEN MSP Emergency application with Traffic Simulation is running</td></tr><tr><td>2</td><td>B</td><td>Device B is on and connected to OPEN MSP Emergency application with Flooding Simulation is running</td></tr><tr><td>3</td><td>B</td><td>Put the device B close to the RFID reader</td></tr></table></td></tr>
<tr><td>Expected output</td><td><table><tr><th>Num</th><th>Device</th><th>Expected output</th></tr><tr><td>1</td><td>A</td><td></td></tr><tr><td>2</td><td>B</td><td></td></tr><tr><td>3</td><td>B</td><td>The OPEN MSP proposes the Flooding simulation migration from device B to device A, device A user can accept or reject the incoming migration request.</td></tr></table></td></tr>
<tr><td>Actual output</td><td>Devices selected are <A=PC with RFID reader, B=PDA with RFID tag>
<table><tr><th>Num</th><th>Device</th><th>Action</th></tr><tr><td>1</td><td>A</td><td></td></tr><tr><td>2</td><td>B</td><td></td></tr><tr><td>3</td><td>B</td><td>The OPEN MSP effectively proposes the Flooding simulation migration from device B to device A. To user on device A it is required to accept the incoming migration request.</td></tr></table></td></tr>
<tr><td>General consideration</td><td></td></tr>
</table>

ID	FE05	**Status**	*Passed*
Executor	Vodafone Team	**Execution date**	5 May 2010
Item	User initiated migration	**Prototype**	Social Game Emergency PacMan Web Page Migration

Description	[82] The OPEN platform must provide four types of migration initiation: User initiated • Upon user request, the OPEN Migration Service Platform proposes a list of available devices and the UI components to migrate; • The user selects the destination devices and the UI components to migrate. [98] The OPEN MSP must ensure that a migration occurs only if either the user explicitly consents to it, or it conforms to a predefined policy that allows it.
Input	Through the application specific interfaces, the user selects the UI components to be migrated and the destination device among the available ones.
Expected output	The UI components selected (and only it) effectively migrates to the destination device, after that the user accept the incoming migration request.
Actual output	[Emergency] the selected "Flooding Control" completely migrates to the Target device after the user accepts the incoming migration request. [Social Game] only the selected "betting" area migrates on the mobile device after the user confirms the migration. [PacMan] the complete PacMan application migrates to mobile device after the user accepts the incoming migration request. [Web Page Migration] the selected page areas migrate from PC to mobile device after the user accepts the incoming migration request.
General consideration	The detailed steps to do in order to perform migration depend from the application used to execute the test case. About requirement 98 it is also possible that the platform automatically accepts the incoming migration request on the bases of the policy coded in the trigger management module.

ID	FE06	**Status**	*Passed*
Executor	Vodafone Team	**Execution date**	5 May 2010
Item	Adaptation	**Prototype**	PacMan
Description	[78] The system provides the users playing games with different interactive support.		

<table>
<tr><td>Input</td><td>Device A and device B can be:
<A=PC, B=PDA>
<table><tr><th>Num</th><th>Device</th><th>Action</th></tr><tr><td>1</td><td>A</td><td>Device A is on and connected to OPEN MSP
PacMan is running</td></tr><tr><td>2</td><td>B</td><td>Device B is on and connected to OPEN MSP</td></tr><tr><td>3</td><td>A</td><td>Migrate the PacMan game to device B</td></tr><tr><td>4</td><td>B</td><td>Continue the game on device B</td></tr></table></td></tr>
<tr><td>Expected output</td><td>When the application migrates from device A (PC) to device B (PDA) it changes the interaction mode from keyboard to touch screen.</td></tr>
<tr><td>Actual output</td><td>On the PC the game interaction is through keyboard, while once the game migrates on mobile it uses specific application buttons available on the touch screen.</td></tr>
<tr><td>General consideration</td><td>Using the PacMan prototype this requirement is very well stressed because the difficulty of the game changes according to the usability of the device in which the game is running. Technically it consists of a change of the game logic.</td></tr>
</table>

ID	FE07	**Status**	*Passed*
Executor	Vodafone Team	**Execution date**	5 May 2010
Item	Context Awareness	**Prototype**	Social Game Emergency PacMan Web Page Migration
Description	[157] The OPEN Migration Service Platform must be aware of the availability of devices as migration sources or targets and must coordinate the migration between source and target devices. [5.01.03] The OPEN Migration Service Platform must provide for several types of migration of UI elements or data: • TOTAL: all components of the application migrate • PARTIAL: only a part of the application migrates. The migration can be from multiple source devices to multiple target devices. [47] The OPEN Migration Service Platform must enable adaptation of UI according to the interaction capabilities and modalities of the target devices. It must enable the adaptation of the choice of the layout structure and interaction technique to the available resources of the device.		

<table>
<tr><td>Input</td><td>Device A and device B can be:
<A=PC, B=PC>, <A=PDA, B=PC>, <A=PC, B=PDA> for Social Game
<A=PC, B=PC> for Emergency
<A=PC, B=PDA> for PacMan
<A=PC, B=PDA> for Web Page Migration

<table>
<tr><th>Num</th><th>Device</th><th>Action</th></tr>
<tr><td>1</td><td>A</td><td>Device A is on and connected to OPEN MSP
PROTOTYPE is running</td></tr>
<tr><td>2</td><td>B</td><td>Device B is on and connected to OPEN MSP</td></tr>
<tr><td>3</td><td>A</td><td>Migrate the application TOTALLY or PARTIALLY (depends on the considered prototype) to device B</td></tr>
</table>
</td></tr>
<tr><td>Expected output</td><td>Before to trigger the migration observe that correctly all the prototypes show a list with the available devices clearly indicating if they are source and/or target. The fact that they are in the list means that they are available for migration.

TOTAL MIGRATION: When the total migration is performed you must have all the components of the application correctly migrated from device A to device B.

PARTIAL MIGRATION: When the partial migration is performed you must have the selected parts of the application correctly migrated from device A to device B, while the other components remains running on device A.

In both cases (TOTAL and PARTIAL migration) have to be observed that the migrated application adapt its interface to the device to which it is migrated. This has to be valid for all the prototypes under test.</td></tr>
<tr><td>Actual output</td><td>[Emergency] the selected “Flooding Control” migrates completely to the Target device, deleting the components from the start device.

[Social Game] only the selected “betting” area migrates on the mobile device and adapts its rendering.

[PacMan] the complete PacMan application migrates to mobile device and change its rendering and also its interaction modality.

[Web Page Migration] the selected page areas migrate from PC to mobile device adapting the web page itself.</td></tr>
<tr><td>General consideration</td><td></td></tr>
</table>

<table>
<tr><td>ID</td><td>FE08</td><td>Status</td><td>Passed</td></tr>
<tr><td>Executor</td><td>Vodafone Team</td><td>Execution date</td><td>5 May 2010</td></tr>
<tr><td>Item</td><td>Data Streaming Migration</td><td>Prototype</td><td>Social Game</td></tr>
<tr><td>Description</td><td colspan="3">[126] Streaming and copying of large data volumes must be supported by the OPEN Migration Service Platform.

[91] The OPEN Migration Service Platform should predict the data and applications needed when going mobile, so that the user can take them with him, if desired (*).</td></tr>
<tr><td>Input</td><td colspan="3">Device A and device B can be: <A=PC, B=PDA>

<table>
<tr><th>Num</th><th>Device</th><th>Action</th></tr>
<tr><td>1</td><td>A</td><td>Device A is on and registered to OPEN MSP
Start Social Game</td></tr>
<tr><td>2</td><td>A</td><td>Login to the Social Game</td></tr>
<tr><td>3</td><td>B</td><td>Device B is on and registered to OPEN MSP</td></tr>
<tr><td>4</td><td>A</td><td>Migrate the IPTV streaming to device B</td></tr>
</table>
</td></tr>
<tr><td>Expected output</td><td colspan="3">The IPTV stream is supported by the OPEN Migration Service Platform because on device B it correctly continues the TV program that was previous running on device A.</td></tr>
<tr><td>Actual output</td><td colspan="3">Once the IPTV is migrated on the target device it is kept the channel selected on the source and also the progress of the broadcast is correctly maintained.</td></tr>
<tr><td>General consideration</td><td colspan="3">(*) From the OPEN MSP point of view this feature is supported by the Migration Orchestration module. Actually there isn’t any application on top of the middleware that exploits this feature.</td></tr>
</table>

<table>
<tr><td>ID</td><td>FE09</td><td>Status</td><td>Passed</td></tr>
<tr><td>Executor</td><td>Vodafone Team</td><td>Execution date</td><td>5 May 2010</td></tr>
<tr><td>Item</td><td>Continuity</td><td>Prototype</td><td>Social Game</td></tr>
<tr><td>Description</td><td colspan="3">[162] The OPEN Migration Service Platform should be able to maintain the data</td></tr>
</table>

<table>
<tr><td></td><td>inserted by the user in the source device and show them in a consistent way after migration on the target device.

[163] The OPEN Migration Service Platform must enable the presentation of the last data inserted by the user on the source device in the first presentation provided to the user in the target device.

[54] The OPEN Migration Service Platform must make it possible to continue the user's current service seamlessly across multiple devices.

[74] The OPEN Migration Service Platform must enable users to migrate identified logical parts of the application to other devices. The user must be able to migrate more than the user interface.</td></tr>
<tr><td>Input</td><td>Device A and device B can be: <A=PC, B=PC>

<table>
<tr><th>Num</th><th>Device</th><th>Action</th></tr>
<tr><td>1</td><td>A</td><td>Device A is on and connected to OPEN MSP
Start Social Game</td></tr>
<tr><td>2</td><td>A</td><td>Login to the Social Game</td></tr>
<tr><td>3</td><td>B</td><td>Device B is on and connected to OPEN MSP</td></tr>
<tr><td>4</td><td>A</td><td>Set the following bet parameters:
Championship: Best Lap
Driver: Felipe Mass
Bet Import: 50,00
Don't click on bet</td></tr>
<tr><td>5</td><td>A</td><td>Migrate Betting components to device B</td></tr>
<tr><td>6</td><td>B</td><td>Click on Bet button and confirm your bet</td></tr>
</table></td></tr>
<tr><td>Expected output</td><td>Device A and device B can be: <A=PC, B=PC>

<table>
<tr><th>Num</th><th>Device</th><th>Expected output</th></tr>
<tr><td>1</td><td>A</td><td></td></tr>
<tr><td>2</td><td>A</td><td></td></tr>
<tr><td>3</td><td>B</td><td></td></tr>
<tr><td>4</td><td>A</td><td></td></tr>
<tr><td>5</td><td>A</td><td></td></tr>
<tr><td>6</td><td>B</td><td>The Betting parameters are maintained during migration. They are:
Set the following bet parameters:
Championship: Best Lap
Driver: Felipe Mass
Bet Import: 50,00</td></tr>
</table></td></tr>
</table>

<table>
<tr><td>Actual output</td><td>Device A and device B are: <A=PC, B=PC>

<table>
<tr><th>Num</th><th>Device</th><th>Expected output</th></tr>
<tr><td>1</td><td>A</td><td></td></tr>
<tr><td>2</td><td>A</td><td></td></tr>
<tr><td>3</td><td>B</td><td></td></tr>
<tr><td>4</td><td>A</td><td></td></tr>
<tr><td>5</td><td>A</td><td></td></tr>
<tr><td>6</td><td>B</td><td>The Betting parameters are maintained during migration. They are:
Set the following bet parameters:
Championship: Best Lap
Driver: Felipe Mass
Bet Import: 50,00</td></tr>
</table>
</td></tr>
<tr><td>General consideration</td><td></td></tr>
</table>

<table>
<tr><td>ID</td><td>FE10</td><td>Status</td><td>Passed</td></tr>
<tr><td>Executor</td><td>Vodafone Team</td><td>Execution date</td><td>5 May 2010</td></tr>
<tr><td>Item</td><td>Application Logic Reconfiguration</td><td>Prototype</td><td>PacMan</td></tr>
<tr><td>Description</td><td colspan="3">[128] For migration occurring between devices in a PAN, the OPEN MSP must enable application-logic reconfiguration between multiple devices.

[59] The OPEN MSP must enable the application to adapt its behavior to the context/user’s needs</td></tr>
<tr><td>Input</td><td colspan="3">Device A and device B can be: <A=PC, B=PDA>

<table>
<tr><th>Num</th><th>Device</th><th>Action</th></tr>
<tr><td>1</td><td>A</td><td>Device A is on and connected to OPEN MSP
Start playing with PacMan</td></tr>
<tr><td>2</td><td>B</td><td>Device B is on and connected to OPEN MSP</td></tr>
<tr><td>3</td><td>A</td><td>Execute the migration of the complete application to device B</td></tr>
<tr><td>4</td><td>B</td><td>Continue playing PacMan on device B</td></tr>
</table>
</td></tr>
<tr><td>Expected output</td><td colspan="3">When the migration is occurred and the game continues on device B the ghost and Pacman logic has to be simpler than the one used on device A. This</td></tr>
</table>

	because the device B (context) on which the game is migrated is harder to use, so also the logic changes to a simplified one.
Actual output	Once the migration occurred the game correctly continues on device B. The ghost seems to be a bit slower respect its behavior on the source device.
General consideration	

ID	FE11	**Status**	*Passed*
Executor	Vodafone Team	**Execution date**	5 May 2010
Item	Migration Recovery	**Prototype**	TextMigration
Description	The OPEN Migration Service Platform enables the possibility to migrate from a device to another and recover the migration coming back to the original starting devices.		
Input	Device A and device B can be: <A=PC, B=PC>		

Num	*Device*	*Action*
1	A	Device A is on and connected to OPEN MSP Start TextMigration application
2	B	Device B is on and connected to OPEN MSP
3	A	Put the string “Hello world!” in TextMigration
4	A	Execute the total migration of the complete application to device B
5	B	Continue inserting a new string “Hi from device B” on device B
6	B	Execute the total migration of the complete application to device A
7	A	Continue to use the application on device A

Expected output	When the migration is occurred and the TextMigration application continues the application state is maintained. It means that the written strings are maintained and correctly migrated on the corresponding device.
Actual output	Once the migration occurred strings that represent the application state are correctly maintained.
General consideration	TextMigration is an ad hoc application developed to test some specific features already available in the OPEN MSP but not yet exploited by any official OPEN application prototype.

D.1.2 NOT VERIFIABLE

Item	REQ - 135 Context and Discovery
Description	[135] The OPEN Migration Service Platform must make updated context information available to applications and users. E.g. so that new context information about a user can cause a change in another participants' capabilities (file accessibility, GPS positioning, WiFi localization,…)
Reason	This feature is supported by the Context Management Framework (CMF) module, but up to now there isn't prototype that exploits this feature.

Item	REQ - 82 Context and Discovery
Description	[82] The OPEN platform must provide four types of migration initiation: 3) Automatic platform initiated A platform-initiated migration that fulfils a user-specified policy for being performed automatically without user involvement;
Reason	This feature is implemented in the current version of the platform but up to now there isn't any prototype exploiting this feature.

D.1.3 NOT SUPPORTED

Item	REQ - 82 Context and Discovery
Description	[82] The OPEN platform must provide four types of migration initiation: 4) Third party initiated One user requests the platform to initiate migration for another user.
Reason	This feature is not implemented in the current version of the platform because it has not been found any use case that requires it.

Item	REQ - 129 Application Logic Reconfiguration
Description	[129] The OPEN platform must be able to perform application logic reconfiguration between multiple devices and verify the presence of all necessary modules. If one or more modules are missing on the target device, the platform must: • Set the connection to the modules repository • Download the required modules Install and configure the modules on the device
Reason	The actual version of the OPEN Platform doesn't provide any mechanism to automatically discover, download and install the eventually missing modules. However new modules can be manually plugged to the platform. This process can be made automatic, but it is more a technical implementation while an innovative/research aspect.

Item	REQ - 41 and 93 Usability and Humanity
Description	[41] The OPEN Migration Service Platform must be able to learn the user's preferred modality for a given application and device. [93] The OPEN Migration Service Platform should remember previous migration settings, when it recognizes that the conditions are the same (e.g. when the user comes to the office).
Reason	The actual version of the OPEN Platform doesn't provide any learning mechanism, so these requirements are not implemented.

D.2. MULTICORE, MULTIMODAL AND MULTIUSER

D.2.1 VERIFIABLE

<table>
<tr><td>ID</td><td>FE12</td><td>Status</td><td>Passed</td></tr>
<tr><td>Executor</td><td>Vodafone Team</td><td>Execution date</td><td>5 May 2010</td></tr>
<tr><td>Item</td><td>Multicore
and other devices</td><td>Prototype</td><td>Social Game</td></tr>
<tr><td>Description</td><td colspan="3">[4] The OPEN MSP must enable the user to split the display screen of a device and use the different parts for different applications.</td></tr>
<tr><td>Input</td><td colspan="3">Device A and device B can be: <A=PC, B=PC>

<table>
<tr><th>Num</th><th>Device</th><th>Action</th></tr>
<tr><td>1</td><td>A</td><td>Device A is on and connected to OPEN MSP
Start Social Game</td></tr>
<tr><td>2</td><td>A</td><td>Login to the Social Game</td></tr>
<tr><td>3</td><td>B</td><td>Device B is on and connected to OPEN MSP</td></tr>
<tr><td>4</td><td>A</td><td>Migrate the IPTV streaming application to device B</td></tr>
<tr><td>5</td><td>A</td><td>Migrate the chat application to device B</td></tr>
</table></td></tr>
<tr><td>Expected output</td><td colspan="3">On device B in the same screen, which results splitted in two parts, must be available the IPTV application and also the chat one that work at the same time.</td></tr>
<tr><td>Actual output</td><td colspan="3">Device A and device B are: <A=PC, B=PC>. On device B in the same screen is splitted in two parts that contains respectively the IPTV and the chat that work at the same time.</td></tr>
<tr><td>General consideration</td><td colspan="3"></td></tr>
</table>

<table>
<tr><td>ID</td><td>FE13</td><td>Status</td><td>Passed</td></tr>
<tr><td>Executor</td><td>Vodafone Team</td><td>Execution date</td><td>5 May 2010</td></tr>
<tr><td>Item</td><td>Multimodal</td><td>Prototype</td><td>PacMan</td></tr>
<tr><td>Description</td><td colspan="3">[40] The OPEN MSP must support multimodal input and output.</td></tr>
<tr><td>Input</td><td colspan="3">Device A and device B can be: <A=PC, B=PDA>

<table>
<tr><th>Num</th><th>Device</th><th>Action</th></tr>
<tr><td>1</td><td>A</td><td>Device A is on and connected to OPEN MSP
PacMan</td></tr>
<tr><td>2</td><td>A</td><td>Start playing PacMan using keyboard as input device</td></tr>
<tr><td>3</td><td>B</td><td>Device B is on and connected to OPEN MSP</td></tr>
<tr><td>4</td><td>A</td><td>Migrate the game to device B</td></tr>
<tr><td>5</td><td>B</td><td>Continue playing the game using as input device the touch screen controls</td></tr>
</table></td></tr>
<tr><td>Expected output</td><td colspan="3">The action provided on device A using the keyboard input modality must remain the same also after the migration when the input modality changes to touch screen.</td></tr>
<tr><td>Actual output</td><td colspan="3">Device A and device B are: <A=PC, B=PDA>. The action to go straight on which on device A is provided by the straight arrow on the keyboard it is provided by a button available on the touch screen of the device B.</td></tr>
<tr><td>General consideration</td><td colspan="3"></td></tr>
</table>

<table>
<tr><td>ID</td><td>FE14</td><td>Status</td><td>Passed</td></tr>
<tr><td>Executor</td><td>Vodafone Team</td><td>Execution date</td><td>5 May 2010</td></tr>
<tr><td>Item</td><td>Multiuser</td><td>Prototype</td><td>Emergency</td></tr>
<tr><td>Description</td><td colspan="3">[58] The OPEN Migration Service Platform must be able to handle (co-ordinate and synchronize) input from different users.</td></tr>
<tr><td>Input</td><td colspan="3">Device A and device B can be: <A=PC, B=PDA>

<table>
<tr><th>Num</th><th>Device</th><th>Action</th></tr>
<tr><td>1</td><td>A</td><td>Device A is on and connected to OPEN MSP</td></tr>
<tr><td>2</td><td>A</td><td>The traffic control is running on this device</td></tr>
<tr><td>3</td><td>B</td><td>Device B is on and connected to OPEN MSP</td></tr>
<tr><td>4</td><td>B</td><td>The flood control is running on this device</td></tr>
</table></td></tr>
</table>

	5 | A | The user decide to migrate to device B the traffic manager information keeping them synchronized
Expected output	On device B the user has the originally flood control information and after the migration are available in overlay the information about the traffic situation. Verify that the data are effectively synchronized between device A which has the control and device B which views the shared data.
Actual output	[Emergency] When the user migrate to device B the traffic control information keeping them synchronized effectively they remain synchronized because the control is kept by the user with device A.
General consideration	

ID	FE15	**Status**	*Passed*
Executor	Vodafone Team	**Execution date**	5 May 2010
Item	Multiuser	**Prototype**	TwitterWall
Description	[66] The user must be able to specify migration policies, e.g. automatic migration when switched off. [36] The OPEN MSP must enable users to define privacy policies, e.g. for application migration to shared/public displays, and it must ensure those policies are honored. Each time a migration occurs, OPEN MSP must adhere to privacy policies and settings for files/directories and applications. [48] The OPEN MSP must enable users to control the access rights to application data displayed in a migrated UI. For example if output data could be partially displayed on a public screen because of privacy settings, each data should be associated with a privacy metadata. [52] The OPEN Migration Service Platform should provide a policy setting for deciding who can edit data in a multiuser scenario.		
Input	Device A and device B can be: <A=MiniPC, B=Wall> Before it is necessary to fix a word blacklist that if present in the tweets reverts the migration to the user terminal only. The word blacklist has to contain the word: “exampleBlacklist” ***Num*** | ***Device*** | ***Action*** 1 | A | Device A is on and connected to OPEN MSP		

	<table><tr><td>2</td><td>B</td><td>Device B is on and connected to OPEN MSP</td></tr><tr><td>3</td><td>A</td><td>Migrate the Twitter contains to TwitterWall</td></tr><tr><td>4</td><td>B</td><td>Twitter Wall presents also the user A Tweets</td></tr><tr><td>5</td><td>A</td><td>Write a tweet containing the word “exampleBlacklist”</td></tr><tr><td>6</td><td>B</td><td>The OPEN MSP migrate the Twitter output from the Wall to the MiniPC</td></tr></table>
Expected output	The OPEN MSP migrate the Twitter output from the Wall to the MiniPC.
Actual output	Twitter output migrates from the Wall to the MiniPC.
General consideration	In the actual OPEN implementation there is a basic policy management implemented through scoring functions. This requirement is developed and hardcoded for a specific use case. It is not yet available in a generic manner. However it is supported, even if considering only a specific implementation. The fact that it is available in a specific use case shows how the structure of the platform is designed to support this feature.

D.2.2 NOT VERIFIABLE

No requirement of this class falls into this category.

D.2.3 NOT SUPPORTED

Item	REQ - 88 Multiuser
Description	[88] The OPEN MSP must recognize conflicts, e.g. when multiple users try to use the same devices. It must provide mechanism to resolve such conflicts.
Reason	In the current version of the platform really to put in evidence that the system is multiuser it is possible that a device can support more than only one application running on it. From the point of view of the user it is important to remember that it is always possible to the user to accept or reject an incoming request as destination for a given application.

D.3. PLATFORM FACILITIES

ID	FE16	**Status**	*Passed*
Executor	Vodafone Team	**Execution date**	5 May 2010
Item	Maintance and Support	**Prototype**	
Description	[123] The OPEN MSP must collect data, e.g. create a log file with the purpose of measuring the framework and application execution as well as migration events		
Input	Start the OPEN Server (eventually redirecting the standard output and the standard error to a file) and then execute some actions, like migration, device registration, application registration or what you want.		
Expected output	On the OPEN Server side a set of log files are available, eventually the file in which the standard output is redirected. They have to contain information about the actions executed.		
Actual output	All the stack trace and also additional information like timestamps, and XML-RPC calls are registered in comprehensible log files.		
General consideration			

Item	REQ - 161 and 46 and 100 Development
Description	[161] The OPEN MSP must be able to support migration amongst different implementation languages in order to exploit the interaction possibilities offered by the various environments. [46] The OPEN MSP must be able to access an application that can be dynamically implemented in different implementation languages (e.g. Web, Java). [100] The OPEN MSP should provide mechanisms for developers to easily write code for multiple platforms.
Reason	All the requirements reported above refer to the capability of the OPEN MSP to support and offer the migration features to different kind of applications.

D.4. MULTICORE

Item	REQ - 124 Multicore and other devices
Description	[124] The OPEN MSP must enable applications to leverage additional cores on multicore CPUs to provide full performance to the user
Reason	This requirement is not testable because the multicore extension of the OPEN MSP is not implemented due to the closure, since Dec 31st 2009, of the NEC laboratory focused on the development of this features.

Item	REQ - 125 Multicore and other devices
Description	[125] The OPEN MSP must make it possible for the application developer to utilize dedicated hardware for video (de)coding and processing, 3D graphics and audio.
Reason	This requirement becomes testable only if high performance applications that need to use hardware acceleration are available. Actually it is not identified any use case to test this features because none of the available prototypes requires high performance computation and also the OPEN MSP not doesn't directly manages dedicated resources.

D.5. NO ADDED VALUE

Item	REQ - 90 Overall and UI Migration
Description	[90] The OPEN Migration Service Platform must give the user a way to control which UI elements and data are migrated.
Reason	The OPEN Migration Service Platform allows managing the migration of parts of an application. How this feature is available to the user depends how the considered prototype renders it in term of interface. All the considered prototypes implement this feature. All the considered prototype displays in form of a list (drop-down rather fixed) the possible destination devices and makes selectable directly from the user interface the components of the application to migrate. From the platform point of view this requirement doesn't add any value.

Item	REQ - 108 Data migration
Description	[108] The OPEN Migration Service Platform must be able to handle the migration of temporarily changing information in a 3D dimensional map (visual modality).
Reason	This requirement refers to the presentation layer of the applications rather to OPEN middleware implementation. From the OPEN MSP point of view the migration of data that can vary changing the context in which the application run is supported.

Item	REQ - 159 and 94 Usability and Humanity
Description	[159] The OPEN Migration Service Platform must ensure that the application user interface is as homogeneous as possible when migrating from one device to another. [94] The user must be able to set preferences (font size, colors, audio volume and brightness) and defaults for migration.
Reason	These two initial requirements are a bit a countersense respect to the OPEN vision. The adaptation of the logic and user interface to a device is one of the strong points of the entire project itself, so the idea is to leave the platform to select which is the best adaptation strategy.

Item	REQ - 69 and 71 Usability and Humanity
Description	[69] The OPEN Migration Service Platform must ensure that the applications like games are migrated in such a way that the user does not experience any loss of continuity. Interruption of game continuity should be avoided, especially in real time game. [71] The interaction with migration phases must be as unobtrusive as possible.
Reason	These initial requirements are more related to the user experience than to a technological point of view. However they are already evaluated in the Usability Validation tests. From the technological point of view they are partially covered by the performance indicators that record the migration time.

Item	REQ - 27 Usability and Humanity
Description	[27] The OPEN Migration Service Platform must enable applications to clearly show who has control in a multi-user scenario.
Reason	This feature is related to the application and not to the platform itself.

Item	REQ - 101 Usability and Humanity
Description	[101] The OPEN Migration Service Platform should provide the user information like how much a migration would potentially cost.
Reason	Since in term of time the duration of the migration it is always very low, it has not been spent any effort to develop a mechanism to predict how much each migration potentially cost. As remark it is important to remember that the perception of how well the migration maintains continuity of the application experience is evaluated with Usability test.

Item	REQ - 24 Usability and Humanity
Description	[24] Some graphical user interface or visual clue must be used to • Trigger a partial migration • Recover a migration, so the application come back to the device For intuitiveness sake, this cue must be consistent throughout devices.
Reason	This feature is related to the application and not to the platform itself. The platform manage the case in which a migratory application comes back to the source device, but to use this features it has to be exported by the application user interface.

Item	REQ - 118 Usability and Humanity
Description	[118] The OPEN Migration Service Platform must enable the user to know (seamlessly over migration), which information is important and which is not.

Reason	The platform does not hide any information so, also after a migration, the user found all the information available in the source device in the target device, too.

Item	REQ - 57 Multimodal
Description	[57] The user must select which modality is migrated.
Reason	This requirement is a nonsense because is the platform that automatically adapts the modality that the user is using to interact with the OPEN MSP.

Item	REQ - 114 Multimodal
Description	[114] The OPEN MSP must enable the user to compare data sets in different way.
Reason	This requirement refers to the presentation layer of the applications that use the OPEN middleware.

Item	REQ - 26 Multiuser
Description	[26] The OPEN MSP must enable users and application to share a device and to divide its resources amongst themselves, e.g. two users may split a large screen between them.
Reason	This requirement it is not strongly related to the OPEN platform, because the resources management is provided by the Operative System of the device on which the platform run.

Item	REQ - 127 and 132 Network

Description	[127] The OPEN MSP must configure network settings but before the migration from a network to another, it must alert the user and wait for authorization. [132] For migration involving devices in a WPAN and network servers, OPEN platform must manage new communication among: • devices • devices and servers • servers
Reason	All the requirements reported above refer not to the capability of the OPEN MSP but to the Operative System on which the OPEN MSP is installed and runs. For this reason it has no sense to consider the above ones as platform requirements but it is more sense to consider them as pre-requirements.

Item	REQ - 84 Maintance and Support
Description	[84] The OPEN Migration Service Platform must allow automatic update of components.
Reason	This feature related to the components update is implemented through the OSGi mechanism that provide in the CMF module to activate the necessary retriever to obtain the actual context information.

Item	REQ - 140 Security and Privacy
Description	[140] Personal data stored on any OPEN MSP public database must be protected.
Reason	This requirement is not directly implemented in OPEN MSP, but it is considered as a pre-requirement to use OPEN. This because data protection is a complex issue that is already well managed by the DBMS and it is un-useful to re-implement it inside the OPEN MSP.

Item	REQ - 141 and 15 Security and Privacy
Description	[141] Any personal data transferring (device-device, device-platform and platform-platform) should be safe.

	[15] Wireless communications should be encrypted.
Reason	These requirements are not directly implemented in OPEN MSP, because at application level there is not available any encryption mechanism. However they can be considered as a sort of pre-requirement to use OPEN in a safe way. Note that they refer to the network level, which is a lower level than application one at which OPEN MSP work.

Item	REQ - 143 Security and Privacy
Description	[141] Any personal data transferring (device-device, device-platform and platform-platform) should be safe. [143] The user interface of the migratory application has to allow managing permission settings.
Reason	This requirement is from the point of view of the migratory application rather than the OPEN Platform point of view.

Item	REQ - 63 Offline mode
Description	[63] The OPEN Migration Service Platform should work with or without internet connection when the components necessary are locally available.
Reason	Considering any one of the available prototypes and installing on you LAN all the components necessary to correct run the application, if also the OPEN MSP server is inside your LAN, it is possible to use the application with the OPEN MSP also without the Internet connection.