D1.1: Use case scenario definitions and initial requirements

Abstract / Executive summary:
This deliverable presents a description of the use case scenarios, detailed enough to be later used by other work packages to implement and validate the respective functionalities. Driven by project objectives and considering envisioned results from contextual and industrial point of view, the use case providers, namely ATC and Tellu, have chosen two application domains (a media system and a personal security system) in order to cover different technical requirements which address all the challenges of building HD-services. The media domain requires global aggregation and distribution of information from various sources and deals with high volumes of data while the personal security system needs communications with a variety of heterogeneous devices and sensors and has stronger real-time requirements. Furthermore, this deliverable describes the business and technological domains of the case study owners as well as the development processes relevant to each case study. Finally, the scenarios resulted in the extraction and definition of an initial set of requirements from the use case providers that will act as the spearhead towards the implementation of HEADS tools.
Contents

D1.1: Use case scenario definitions and initial requirements ........................................... 1
Table of Figures ................................................................................................................. 4

Version History .................................................................................................................. 5
1 Executive Summary .......................................................................................................... 6
2 Introduction ...................................................................................................................... 7
  2.1 SCOPE AND OBJECTIVES .......................................................................................... 7
  2.2 BIG DATA ANALYTICS: “SENSORS AND THE INTERNET OF THINGS” ................. 7
  2.3 SOCIAL-MEDIA EYE WITNESSING REPORT ............................................................ 8
  2.4 COMMUNICATIONS WITH A VARIETY OF HETEROGENEOUS DEVICES AND SENSORS .......... 9
  2.5 DESIGNING, DEPLOYING AND VALIDATING FUTURE COMPUTING CONTINUUM .................. 9
  2.6 RELATIONSHIPS WITH OTHER WORK PACKAGES (WPS) ............................................. 10
  2.7 USEFUL GUIDELINES FOR READING THE REPORT ...................................................... 10
3 Use case definition: A personal security system ............................................................. 12
  3.1 BUSINESS AND TECHNOLOGICAL DOMAIN OF THE CASE STUDY ......................... 12
  3.2 USE CASE SCENARIO DEFINITION ............................................................................. 12
    3.2.1 High level description of the use case ..................................................................... 12
    3.2.2 Why the scenario cannot be realized with the current state-of-the-art ..................... 13
    3.2.3 Involved infrastructure ......................................................................................... 14
    3.2.4 Involved actors .................................................................................................... 15
  3.3 SCENARIO 1: PATROL GUARD INCIDENT MONITORING AND DETECTION .................... 15
  3.4 SCENARIO 2: ALARM CENTRE SITUATION MONITORING ........................................... 16
  3.5 SCENARIO 3: EMERGENCY ASSISTANCE REQUEST MESSAGE DISPATCHING AND HANDLING ...... 17
4 Use case definition: A media system ............................................................................... 19
  4.1 BUSINESS AND TECHNOLOGICAL DOMAIN OF THE CASE STUDY .............................. 19
  4.2 USE CASE SCENARIO DEFINITIONS ............................................................................ 20
    4.2.1 High level description of the use case .................................................................... 20
    4.2.2 Why the scenario cannot be realised with the current state-of-the-art .................... 25
    4.2.3 Involved infrastructure ....................................................................................... 25
    4.2.4 Involved actors .................................................................................................. 31
Table of Figures

**Figure 1.** Security Company Setup. ........................................................................................................... 12
**Figure 2.** Involved Infrastructure ..................................................................................................................... 15
**Figure 3.** Tellu Personal Security Run-time Use Cases ..................................................................................... 15
**Figure 4.** NewsAsset High Level Architecture .................................................................................................. 22
**Figure 5.** NewsAsset Modules ......................................................................................................................... 23
**Figure 6.** NewsRoom Module Interface ............................................................................................................ 26
**Figure 7.** View of the Archive Module ................................................................................................................ 27
**Figure 8.** Reception Workflow ......................................................................................................................... 29
**Figure 9.** Presentation of NewsAsset Editor’s Role .......................................................................................... 32
**Figure 10.** Presentation of NewsAsset NewsRoom’s Team Role ............................................................ 33
**Figure 11.** Design Time Use Case Diagrams ..................................................................................................... 39
Version History

<table>
<thead>
<tr>
<th>Description</th>
<th>Date</th>
<th>Who</th>
</tr>
</thead>
<tbody>
<tr>
<td>First draft release</td>
<td>2013-12-12</td>
<td>ATC/I.Spais</td>
</tr>
<tr>
<td>Second draft released among editors</td>
<td>2014-01-02</td>
<td>Tellu AS/K.Husa</td>
</tr>
<tr>
<td>Ready for internal review</td>
<td>2014-01-10</td>
<td>ATC/I.Spais</td>
</tr>
<tr>
<td>1st Review</td>
<td>2014-01-16</td>
<td>Software AG/ Walter Waterfeld</td>
</tr>
<tr>
<td>Updates according the 1st Review</td>
<td>2014-01-20</td>
<td>ATC/I.Spais</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tellu AS/K.Husa</td>
</tr>
<tr>
<td>2nd Review</td>
<td>2014-01-24</td>
<td>Inria/Olivier Barais</td>
</tr>
<tr>
<td>Updates according the 2nd Review</td>
<td>2014-01-20</td>
<td>ATC/I.Spais</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tellu AS/K.Husa</td>
</tr>
<tr>
<td>Approval by technical manager</td>
<td>2014-01-27</td>
<td>SINTEF/Franck Fleurey</td>
</tr>
<tr>
<td>Final updates and formatting</td>
<td>2014-01-28</td>
<td>ATC/I.Spais</td>
</tr>
<tr>
<td>Approval by project coordinator</td>
<td>2014-01-29</td>
<td>SINTEF/Trine Seeberg</td>
</tr>
</tbody>
</table>

Copyright

© Copyright 2014 ATC, Tellu

This document has been produced within the scope of the HEADS Project and is confidential to the Project’s participants. The utilisation and release of this document is subject to the conditions of the contract within the 7th Framework Programme, grant agreement no. 611337.
1 Executive Summary

This deliverable presents a description of the use case scenarios, detailed enough to be later used by other work packages to implement and validate the respective functionalities. Driven by project objectives and considering envisioned results from contextual and industrial point of view, the use case providers, namely ATC and Tellu, have chosen two application domains (a media system and a personal security system) in order to cover different technical requirements which address all the challenges of building HD-services. The media domain requires global aggregation and distribution of information from various sources and deals with high volumes of data while the personal security system needs communications with a variety of heterogeneous devices and sensors and has stronger real-time requirements. Furthermore, this deliverable describes the business and technological domains of the case study owners as well as the development processes relevant to each case study. Finally, the analysis of the scenarios resulted in the extraction and definition of an initial set of requirements from the use case providers that will act as spearhead towards the implementation of HEADS tools.
2 Introduction

2.1 Scope and objectives

During the first months of the project, the HEADS consortium analyzed in depth the objectives of the project in order to define those scenarios that will guide the technical developments towards the implementation of the HEADS system. HEADS project will publicly release a number of scenarios coming out of two different domains (a media system and a personal security system) in order to demonstrate the HEADS technology as part of the IDE and methodology. These scenarios are described in details in the following sections of the current report.

The outcome of the scenarios analysis is a list of requirements expressed by the use case owners that will be the spearhead for the developments in the project. Since the requirements (targeted towards the technologies and methods to be developed in HEADS) will serve as key performance indicators to monitor the progress and thus it is crucial to be completely and efficiently defined.

Before describing the scenarios, Section 2 positions the use cases in the scientific world based on their contextual and technological features and describes the core aspects that are addressed. Thus, the assets that HEADS project will issue beyond the state of the art are exposed to the scientific community.

2.2 Big Data analytics: “Sensors and the Internet of things”

Big data analytics hold enormous promise, but in the near term there are simply too many big data solutions that introduce challenges. In the long term, the potential for big data goes way beyond optimizing e-commerce to embrace all kinds of verticals, from manufacturing and environmental monitoring to transportation and electrical grid.

But those vertical areas require the industrial Internet (also known as the “Internet of things”) to come online, where connected sensors deliver huge quantities of telemetry to foster improvements in product design, accurate prediction of failure, and so on. Several companies have been active in this area, but we’re just at the start. Years from now when the industrial Internet is in full swing, big data will be really, really big, and the thirst for big data analytics solutions will be unquenchable. Meanwhile, if any bubbles burst in 2014, big data will go first.

More specifically, regarding environmental monitoring to transportation, today, you only have to turn on Google Maps’ Traffic feature in selected cities to view real-time traffic speed and traffic-jam locations. However, that’s done by Google tracking location of smartphones in cars in relation to cell-phone towers and calculating traffic density and speed.

In the future, the number of sensors in the environment (i.e., going beyond leveraging mobile phones as data-bearing sensors) will grow ever more quickly. More and more sensors, connected to the Internet, will feed real-time data about a variety of things: average traffic speed on roads; slipperiness index for roads receiving rain or snowfall; air pollution levels by location; etc. from government entities. Private businesses will get in on the act, with sensor cameras estimating the number of people in a restaurant, bar, or at the shopping mall, and even the current male/female customer ratio.
The “Internet of Things” is in the early stages of having buses and train cars in some cities capable of sending current position to a web server, utilized by phone apps to inform waiting passengers. Many apps are available to display the exact position of a passenger jet in flight. Further in the future, we should see more mobile “things” connected and displaying their real-time positions, and stationary things reporting on real-time conditions:

- Fire trucks and ambulances. Useful for drivers to learn that an emergency vehicle is about to cross the intersection ahead; handy also for drivers to spot multiple responder vehicles and know to route around the trouble spot.
- Available taxis.
- Temporary road-blocked detour barriers.
- Bridges that sense when a river has risen to or above the road bed.
- Forest-fire detectors that measure smoke levels in the air.
- River-flow sensors, to detect flooding early on; also useful for kayakers and other river-sports enthusiasts.
- Gunfire acoustical sensors, to alert people to stay away from a potentially dangerous situation.
- Athletes’ performance in real-time via worn-on-the-body sensors.

As more sensors enter our world, journalists will be able to access data from more and more of them, aiding not only in disaster coverage (disaster news coverage will be much better in a few years), but being used in all manner of news stories. But not all of them will be accessible.

2.3 Social-media eye witnessing report

In nowadays, when a disaster is happening somewhere in the world there is a lot that is going on, so with power and Internet access available, people can watch the news eco-system to try to learn what damage is where, and what conditions exist the specific time. Between checking local news outlets online, Twitter, Facebook (to a lesser extent), etc., it takes a lot of searching and sifting to get details. Hard as someone can try, maybe he cannot get the details of this local disaster to the degree he desires.

Of course, it’s much better than the “old days” of relying only on broad-path coverage and anecdotal survivor stories from TV and radio news and newspapers; A lot of detailed reports can be picked up from social media (likely some or much of it inaccurate, some incomplete — such as photos of the disaster.

Disaster news coverage will be much better in the future by utilizing information from social networks (SNs).

Many eyewitnesses will snap a disaster photo and post it to Instagram, Twitter, Facebook, etc., explaining what’s going on or the location. Sometimes, a smartphone-toting eyewitness may have location-tracking turned on and physical coordinates of the image are attached when the photo is
published to a social-media site. As smartphone photo apps are set to include location by default, more and more people discover the value of location-tagging the photos they shoot.

As that trend plays out, photos and video from social-media sites are becoming increasingly useful for disaster news coverage. In addition to, if all those images spotted on Twitter and Instagram include location metadata, they can be incredibly useful and a detailed interactive map of the disaster could be created. (That can be done now, but we miss the majority of photos and videos posted at a disaster scene because they don’t include location data.).

News organizations can benefit from this future trend and utilize social-media content to provide much better and more detailed coverage of a major disaster. Twitter itself, one of the primary conduits of eyewitness images and reports during disasters, has come to understand that their service provides a valuable news service to the public. The company easily could evolve to become (also) a news operation, utilizing its own tools and analysis to provide the fastest and most-detailed information of a breaking disaster.

2.4 Communications with a variety of heterogeneous devices and sensors

The HEADS use-case scenarios describe systems deployed in environments spanning from cloud computers to handheld smartphones and sensor platforms situated in the field. These platforms are very different when it comes to capabilities like power consumption, computing resources and communication channels. The communication channels will be characterized by unpredictable latencies, low reliability, low bandwidth and messages arriving out of order (e.g. SMS). The protocols will also be very different, some based on IP and other on serial communication protocols for connectivity with peripherals (e.g. temperature sensors). HEADS tools and methodologies will have to address this device heterogeneity and support systems especially adapted to environments with different capabilities.

Based on the envisaged environment for systems designed by HEADS methodologies and tools, we need to design systems that can handle real life environment requirements. Events are happening in real time and the system should be able to respond in real time. The systems are distributed and should be able to handle messages arriving in an asynchronous manner. Also as HEADS systems often will be deployed in environments where they are interacting with real physical entities (light switches, temperature sensors, etc.) the systems should be able to represent state in an appropriate manner.

The designer that uses HEADS methodology and tools will need to perform different tasks in the process of implementing a service. He needs to capture system knowledge and create a sound and functional design, he needs to generate code capable of running on the infrastructure and eventually he needs to deploy the code to the infrastructure nodes. This development cycle will be performed in an iterative manner such that the HEADS tools should also be able to support code update on live systems.

2.5 Designing, deploying and validating future computing continuum

Delivering services in an effective, efficient and reliable manner across the future computing continuum embracing clouds, social data, sensors and smart devices is a daunting challenge due to the tremendous diversity of the infrastructure. It is a challenge that HEADS project has to address. To do so, several architectures, patterns and computing models will be examined and a
novel approach may need to be proposed. Patterns such as publish-subscribe and their variations
designed to fit different application scenarios such as content-based, event-driven data delivery
etc. will be assessed.

Furthermore, designing and integrating appropriate components into an architecture that will be
able to process, make sense and efficiently deliver valuable content from a storm of data from the
social and physical world is another major challenge for HEADS. Streams of data may be
incoming in various forms depending on the underlying sensor network or smart device. Devices
and nodes may need to seamlessly enter the system, register and operate fluently while at the same
time security should be guaranteed on several layers. Moreover, verification of data or metadata
attached to the content is a feature that will guarantee its reliability and added value. The storm
of incoming data must be analysed on the cloud, clustered, aggregated for relaxing redundancy
and then incorporated in HEADS workflows appropriately. Thus, considering that HEADS
context adopts social networks, sensor networks and smart devices that are inherently
heterogeneous also on their own, the design, integration and deployment of such a system
represents major software engineering challenges that HEADS should address to.

2.6 Relationships with other work packages (WPs)
The current report is the outcome of one of the most important activities that work package (WP)
one will carry out during the project’s lifetime. The activities related to requirements gathering
and elicitation are strongly connected with those performed in the other WPs (WP2 to WP5).
Scenarios and requirements definition was the outcome of analysing the technical objectives and
envisioned achievements described in the Document of Work (DoW). Thus, WP1 during its initial
steps received information from technical WPs. This was realised by organizing bilateral meetings
between the technical partners and the use case providers. During the first months of the project,
these meetings were weekly and each group was reporting its achievements during project’s
weekly calls. In sequence, the scenarios and the requirements describing the business and
 technological domains of the use cases, were distributed back to the technical WPs (WP2-WP5)
in order to lead the development of the HEADS system.

2.7 Useful guidelines for reading the report
After describing the objectives of this report, where it is placed in the HEADS context and the
basics of the frameworks utilized in the use cases, this sub-section concludes by providing some
useful guidelines to the readers. The ultimate goal is to provide the capability of having a quick
look at the document as a whole and focus only on the parts that considered as important.

Section 3 describes HEADS first use case, a personal security system where security guards are
continuously monitored when doing patrols. The scenarios defined bring to the surface the need
of communications with a variety of heterogeneous devices and sensors and with strong real-time
requirements.

Section 4 describes the second use case, an end-to-end multimedia cross-channel commercial
solution for evolving news agencies. The scenarios defined bring into the surface the need of
developing a system that will handle global distribution and deal with high volumes of data.
Section 5 describes how different system actors would like to use the portfolio of HEADS tools to perform different tasks. The described scenarios only cover a small selection of the foreseen use cases during system development and are not meant to be exhaustive or very detailed.

Section 6 analyses the scenarios and presents the initial version of the HEADS requirements following a specific format. It must be mentioned that this list will be continuously updated during the project’s lifecycle.
3 Use case definition: A personal security system

This use case describes a personal security system where security guards are continuously monitored when doing patrols. Further, the system automatically detects incidents and performs emergency actions like notifying peer guards, turning on lights, activating alarm bells etc. Peer guards are given instructions how to assist and reach any assaulted guard.

3.1 Business and technological domain of the case study

This use case is well positioned within the needs of the security guard services industry. Systems like described in this use case will improve the working environment and general security of the security guards. Customers of security services will also experience improved quality since guards can continuously coordinate their tasks and achieve an improved workload distribution.

The major customers of Tellu are security companies, both in Norway and in other European countries. Thus, it is very important for Tellu to be able to provide a future-oriented system that makes use of the most modern and advanced technology. When realized the system described will position Tellu in a very favourable market situation being able to offer one of the best services available.

3.2 Use case scenario definition

3.2.1 High level description of the use case

A security service has many tasks to manage. Possible tasks are e.g. controlling buildings at regular intervals, transportation services of valuable goods, or emergency handling in case of fire or assault.

A security company will typically have an alarm centre handling monitoring of security guards in the field and planning tasks to be dispatched. The alarm centre is also the contact point for the security guard in case of an incident and will coordinate assistance if needed. The following figure (Figure 1) shows a typical infrastructure for a security company.

Figure 1. Security company set up.
Intelligent devices with connectivity are becoming cheaper, smaller and more advanced. Smart phones with many built in sensors are examples of such devices. It is assumed that in the near future all guards in security companies will be wearing intelligent devices with various sensors and sufficient bandwidth. Examples of sensors are accelerometer, GPS (outdoor position), radio sensor (indoor position), temperature, compass, gyro, body sensors like heart beat rate sensors, alarm buttons, etc. The intelligent devices will be always connected to services residing in the cloud to report status and receive orders from the alarm centre.

A guard will have daily routines to complete, typically patrolling and value transportation. However, if something unexpected happens, it is very important that the guard and all other relevant parties get informed about actions to be carried out. It is also very important that the system is highly available and reliable in a critical situation, both in terms of computing resources and human resources.

In HEADS we describe three run-time scenarios based on this usage context, namely

1. Patrol guard incident monitoring and detection.
   The sensor device that the guard carries will continuously monitor the guard situation based on local information like position, heart rate, alarm buttons, accelerometer. If several of these sensors indicate abnormal behaviour an event is raised. If for instance the measured GPS position is outside of normal patrol area, heartbeat rate is very low, and the accelerometer measures small impacts, these factors might be an indication that the guard is unconscious and not able to push the alarm button. High heart beat rate and rapidly changing position might be an indication of an escaping guard or the guard being abducted.

2. Alarm centre situation monitoring.
   Cloud services will receive the situation indication from the device and provide decision support for the personnel at the monitoring centre on coordinating efforts to assist the guard. The personnel at the monitoring centre will have an overview of the available personnel and give commands to relevant personnel in the field based on position, availability, equipment etc. The police will also be called/notified if the situation is serious.

3. Alarm dispatching.
   When commands are issued to the relevant personnel that are going to assist, they have to acknowledge the commands. From that moment devices will continuously report the position of the guard so as to be easily reachable by the emergency personnel. Cloud services will supervise the emergency and notify the monitoring centre if the situation is developing in an unexpected way.

3.2.2 Why the scenario cannot be realized with the current state-of-the-art

If we were to realize the described scenario in an all-in-the-cloud solution (implying that all sensor signals are handled distantly from where they were measured) this scenario would suffer greatly w.r.t. response times and reliability. Below we list some reasons for this.

- The cloud must have as much information as possible to make intelligent decisions. Collecting and transmitting this information to the cloud can give increased latency.
- Higher processing requirements: the cloud does not have unlimited power and important computational resources come at a high price when large amount of sensor data is to be processed.
• Power drain on the battery-powered devices would be the result due to having to send all information to the cloud all the time (radio chips being very power consuming).
• Higher bandwidth requirements on sensor, sensors network and cloud in order to transmit all information.
• Implementing all low level device protocols on the cloud is not feasible.
• Not all devices can send events via IP, and we must use gateways.

The points above clearly indicate that in many aspects it would be advantageous to handle logic as close as possible to the sources of information. However, some parts of the logic must also be handled in a centralised manner. This type of logic typically has to do with coordinating distributed resources based on collected sensor information.

3.2.3 Involved infrastructure
The described scenarios involve an infrastructure consisting of several of the components listed below. The final list of components will be developed during the project:

• Sensors (based on Arduino platform, Android platform):
  • Surveillance sensors
  • Video cameras
  • Body condition monitoring sensors
    • Accelerometer
    • Gyroscope
    • Heart rate
    • Perspiration

• Actuators:
  • Light switches
  • Door lock
  • Alarm bell

• Smartphones with GPS, GSM and IP connectivity. Such devices can handle local decision logic and serve as a data transmission unit. The target platform should be Android.
• GSM network. Handles data transmission between smartphones and cloud services.
• Cloud servers. Handles application logic. Should have a redundant setup such that system does not fail if one server gets offline.
• Alarm centres. Displaying situation information and giving decision support for alarm centre personnel. Should have a redundant setup such that several monitoring centres could handle the situation if one centre gets offline.
3.2.4 Involved actors
Security guard, Alarm centre personnel

3.3 Scenario 1: Patrol guard incident monitoring and detection
This use case handles the process of monitoring the conditions related to a security patrol guard.

Identifier: A.1

Involved actors: Security guard
**Involved infrastructure:** Body sensors, Smartphone, GSM network, Cloud servers, Remote controlled light switches

**Requirements:** Common-02, Common-09, Common-12, Tellu-01, Tellu-02, Tellu-03

**Preconditions:** The security patrol guard is out in the field equipped with smartphone and body sensors. Security guard is about to start his/her patrol round. Body sensors are in idle mode.

**Steps:**

1. Security guard presses “Start Round” on Smartphone app to indicate that patrol round is about to start.
2. Body sensors are set in work mode to continuously register different body conditions (heart rate, body position …).
3. GPS sensor registers position and speed.
4. Positions are saved to a track record that can prove the execution of the patrol round.
5. Security guard is being assaulted and left unconscious on ground.
6. Body sensors react to new abnormal body condition and reports condition and position to peer guards patrolling nearby. The alarm centre is also notified about situation. For high availability different means of reporting should be used. IP should be the preferred method, but other means of communication should be considered based on the current QoS requirements on bandwidth.
7. All lights that are remotely controllable in the surroundings of the assaulted guard are turned on to improve visibility for video coverage.
8. Alarm bell is chiming.

**Post-conditions:** Security guard is lying unconscious on ground. Peer guards and alarm centre have received report. Lights are turned on and alarm bell is chiming.

3.4 Scenario 2: Alarm centre situation monitoring

This use case handles the process of monitoring security personnel out in the field from an alarm centre.

**Identifier:** A.2

**Involved actors:** Monitoring centre, Security guard

**Involved infrastructure:** Body sensors, Smartphone, GSM network, Cloud servers, Monitoring centre

**Requirements:** Common-09, Tellu-02

**Preconditions:** Alarm centre personnel is situated in front of monitors showing real time information from all security personnel out in the field in graphs and maps.

**Steps:**
1. Guards are positioned in maps showing real time position and patrol areas as well as body conditions.
2. One guard is assaulted and left unconscious on ground.
3. Marker representing assaulted guard flashes red on monitor screen and abnormal condition is displayed.
4. A silent voice/video call (no ringing) is established with the assaulted guard such that the monitoring centre can get an overview of the situation at the location.
5. Alarm centre monitors that at least two of the peer guards acknowledge the assistance indication within 1 min. If not acknowledged, the alarm centre starts to page guards to ask explicitly for assistance.

**Post-conditions:** Alarm centre is notified about alarm condition. Video conference is established between alarm centre and assaulted guard.

### 3.5 Scenario 3: Emergency assistance request message dispatching and handling

This use case handles the process of dispatching assistance requests and supervising an emergency situation.

**Identifier:** A.3

**Involved actors:** Monitoring centre, Security guard

Involved infrastructure: Body sensors, Smartphone, GSM network, Cloud servers, Monitoring centre

**Requirements:** Tellu-01, Tellu-02, Tellu-03

**Preconditions:** Security guard on duty has received an emergency assistance request message.

**Steps:**

1. Security guard presses yes to assist assaulted guard.
2. Security guard receives acknowledgment that he/she has been assigned to the emergency.
3. The assisting guard is attached to the voice/video conference call between the alarm centre and assaulted guard. Assisting guard can then have continuous dialogue with alarm centre at the same time as he/she gets an overview of the situation. The guard will also continuously be informed about the risk involved.
4. A route is created for the guard to follow to reach the assaulted guard as fast as possible.
5. The guard follows the route and the alarm centre monitors that the guard approaches the assaulted guard. Positions are reported continuously as the assisting guard approaches.

**6. Alternative 1**
   a. When reaching the assaulted guard, doors are automatically unlocked such that the assisting guard can reach the assaulted guard as fast as possible.
   b. The assisting guard reports that situation is under control.
   c. Flashing marker is turned off.

**7. Alternative 2**
a. The assisting guard is hindered in reaching the assaulted guard (traffic jam, other incidents) or connectivity is lost (out of battery, broken phone...). This is detected by the system.
b. The system automatically dispatches an emergency assistance request to nearby guards.
c. continue at point 1

Post-conditions: Assaulted guard receives assistance and alarm centre is notified.

It is important to realise that if this scenario were to be implemented using an all-in-the-cloud approach as discussed in Section 3.2.1, it would not be as reliable and responsive as it can be with a distributed implementation. Examples on this are:

- The continuous monitoring of body conditions of the guards is based on large amount of sensor data. Doing the event processing in the central service would not be reasonable due to high demand on bandwidth to transmit data.
- The notification to patrolling guards nearby the assaulted guard is initiated from the assaulted guard terminal. If a central service were to initiate this notification it would mean a high dependability on a distant that can fail.
- The control of lights and doors in the buildings will suffer both w.r.t. latency and reliability if they were to be controlled by a central service. Direct control from the assaulted guard terminal will be to prefer to ensure correct behaviour.
- The call-setup between the alarm centre and assaulted guard should not be initiated from a central service, but rather from the assisting guard terminal. If failure of the central service and/or alarm centre, the assisting guard will still get contact with the assaulted guard.
4 Use case definition: A media system

News as an Asset (NewsAsset) is an end-to-end multimedia cross-channel solution for evolving news agencies, broadcasters and publishers. It is an innovative suite of media asset management solutions and it is available in three editions: Publishing, Agency and Broadcast; each one of them providing specialized modules/functionalities targeted to the different media sectors. NewsAsset is a commercial product being marketed by ATC with success both inside and outside Greece with the major clients being the national news agencies of Greece, Portugal, Poland and Russia.

With respect to HEADS objectives and considering several business and technological aspects of the platform, the Agency Edition will be utilized during project’s lifetime. NewsAsset Agency Edition is a modular, configurable, all-in-one multimedia solution for news agencies. It has been built to meet the needs of an evolving news agency to help plan, create, manage and distribute breaking news quickly and efficiently to a wide array of customers through multiple delivery channels. It delivers the core functionality required for event management & editorial planning, content creation, ingestion & aggregation, production monitoring, archiving, and distribution/publishing through multiple channels.

NewsAsset Agency Edition supports the whole life-cycle of news item from planning, through creating, gathering and selecting, editing, to producing, distributing and archiving. An all-in-one system covers all internal, user-generated, Web-acquired or wired multimedia assets, all types of workflow and all news asset management activities, via a friendly, industry-standard compliant interface. Workflow is flexible and configurable. Channel independent content and support for a wide range of formats make NewsAsset ideal for cross channel publishing. Assets can be combined as multimedia stories for enhanced multimedia production. NewsAsset co-operates with widely used tools, (Microsoft Word, Open Office, Photoshop) and is based on industry standards (IPTC, ANPA, NewsML, EventsML), making it a very powerful and flexible solution.

4.1 Business and technological domain of the case study

NewsAsset suite constitutes an innovative management solution for handling large volumes of information in the following forms: text, picture, graphics, audio, video, stories, etc., offering a complete and secure electronic environment for storage, management and delivery of sensitive information in the news production environment. The functional and efficient means of management for the distribution of content that the suite offers streamlines the internal operations and organization of news enterprises with absolute security and reliability.

The suite is capable of handling material already published or to be published, that have either been created within the agency, or originating from external associates (correspondents, photographers, agencies, press offices, etc.). Such material may be of various types, such as text, image, graphic, audio, and video and in several supported electronic formats. Moreover, it offers the ability of creating multimedia news Items, the contents of which may be a combination of all the aforesaid types and formats.

The initial engine of the suite was developed in a Greek State R&D project (in 1996), whereas new collaborations enabling versions with strong search capabilities are currently being researched in EU Projects. The developed solution was able to aggregate the requirements and expressed needs of the most demanding enterprises in the areas of digital news content production,
management and delivery. Since then, the suite was updated by following the present technological achievements. Times change fast, and news agencies are already feeling the impact of the web and the on-line world in their operations. At the same time there is a rapid increase of demand for cross-media news publishing tools and ways to make management of any type of files an easy and risk free process.

In this environment NewsAsset production team realized that news agencies are looking for additional ways to create and distribute their content, whether it is text, images, graphics, video or other digital data. Furthermore, based on the analysis provided in Sections 2.2 and 2.3, the future computing continuum, composed of a wide set of heterogeneous platforms, promises to be an environment that will definitely affect the media industry in terms of creating, managing and exploiting trustworthy media items. Nowadays, interesting media info is out there in the digital world and can be possessed and exploited by utilizing a variation of nodes and software engineering practices like network nodes, gateways, smart-phones, cloud computing and service-oriented methods, etc. The challenge for NewsAsset is to catch up with this evolution and provide services that can handle the developing new situation in the media industry. HEADS innovations are envisioned to offer this opportunity to NewsAsset if they will be efficiently adopted during project’s lifecycle.

4.2 Use case scenario definitions

4.2.1 High level description of the use case

NewsAsset gives news organizations all the tools necessary to streamline the newsgathering, editing, production and distribution process. It delivers the core functionality required for receiving news, archiving, managing the production process and delivering services through multiple channels. The system can send and receive information through a variety of devices such as FTP, fax, wireless LAN, web, email, with future extensions to mobile phones with SMS or WAP, PDAs etc.

The following technical characteristics apply to NewsAsset:

- Open architecture and design allowing an economical and smooth means for future expansion of the system within the enterprise.
- Parameterization, and easy adaptation of the system for the particular environment and requirements of each organization and its users.
- Friendliness, through an ergonomic and functional user oriented graphic interface and working environment.
- Ability to support the simultaneous use of the application by a large number of users with no loss in performance, speed of search results or compromises in security.

NewsAsset core module is based on a three tier distributed client/server architecture. The three tier architecture provides increased performance, flexibility, maintainability, reusability, and scalability, while hiding the complexity of distributed processing from the user. NewsAsset Application Server is a multi-tier architecture application that hosts an API to expose the logic and business processes for use by the NewsAsset Client.
The standard system includes the modules required to cover the basic day-to-day operations and activities of most news agencies. Through the addition of the optional modules, news agencies can expand their functionality as their operations and business needs grow.

The following NewsAsset modules are combined with each other in order to expose applications functionalities:

- **Event Management and Editorial Planning Layer**, an event driven engine enabling fully interactive co-ordination of all agency activities and editorial planning, embedded into the workflow, allowing linkage of events and editorial content.

- **Content Creation, Ingestion and Aggregation Layer**, reception of unlimited number of news wires in various formats, automatic sorting & archiving, providing immediate access to all (authorized) users of the system in real time. The following modules are included:
  - Integrated Multimedia Newsroom module,
  - Archiving & Searching module, safely storing and managing infinite amount of text, image, graphic, audio, video files and multimedia stories and powerful searching mechanisms for supporting advanced search queries with multiple parameters such as date, subject, keyword, author, provider etc, including full text search and a flexible & smart query builder,
  - Multiple Receiving Services module,
  - News Aggregation module.

- **Adaptation for multiple channels layer**, The following modules are included:
  - WEB Content Management & Delivery module,
  - Real Time Monitoring module,
  - Subscribers & Services Management module.

- **Distribution Channels**, The following modules are included:
  - WEB Publishing, Social Media,
  - WEB Services, RSS & PODCASTING,
  - FTP, Emails, Serial Ports, Shared folders,
  - MOBILE Devices & E-Readers.

The following two figures present the NewsAsset high level architecture and NewsAsset modules respectively.
Figure 4: NewsAsset high level architecture
In the context of the HEADS project we will focus on the following three scenarios:

1. **Broadcasting traffic conditions.**
   Chaos is expected to descend on Athens on July 16, when a 10km stretch of “Attiki Odos”, a major highway running from the airport through the center of town that carries an average of 100,000 cars on a summer business day, will be shut down for more than 3 hours due to the official visit of a foreign Prime Minister. But “NewsBeast Radio Station”, a local radio station of a large news agency, sees the closure as an opportunity to experiment with technology tools as it plans to report on the mess as it unfolds. Back
in 2013, the radio station would have hired a helicopter to fly for half an hour above the city centre making the availability of this information extremely costly. But this is not what is happening now! The station has partnered with HEADS, a Greek service provider that has recently developed a service for creating a traffic/navigation map to give drivers a real-time picture of what is happening on the roads. HEADS’s service, after aggregating and analysing information from several sources to get a sense of traffic, incorporates them in a map. Subsequently, it directs the drivers to quicker routes based on the data it collects. During the highway closure, “NewsBeast” will use HEADS data to give its audience updates on what is happening on a variety of alternate routes. “NewsBeast” will use real-time traffic maps, which draw data from sensors located on several roads near the highway and expected to carry heavy traffic. Besides information coming from the sensors, HEADS’s data includes information on spot where a critical mass of people are driving, and presumably many of them are willingly uploading media items to the web via social networks. While “NewsBeast” started looking for information hosted in SNs, the company’s eyes have increasingly turned toward other media sources of information. Thus, information coming from different news wires and delivery channels are aggregated and analysed.

2. Broadcasting environmental conditions
More and more citizens are gradually fond of using bicycles for their own everyday transportation in the city. But the existing transportation routes and infrastructures are currently designed and built to serve vehicle drivers and public transportation users. In addition, environmental conditions in the city center and especially air pollution levels, although improved due to multiple measures and precautions are still preventing people from using their bicycles as an everyday means of transportation. Especially people suffering from various respiratory diseases such as asthma, but still wishing to follow the cycling transportation option, are in great need of information that will allow them to optimize their cycling experience. “NewsBeast” is using again the HEADS service to draw the most preferable cycling routes to the city centre by accessing not only transportation related information (previous scenario) but more importantly environmental information, namely air pollution levels along the potential cycling route. Wireless sensor networks have been already deployed in Athens city centre to monitor the concentration of dangerous gases for citizens. More specifically, environmental sensors measuring: carbon monoxide (CO), carbon dioxide (CO2), ozone (O3), nitrogen dioxide (NO2), temperature, humidity and noise. In addition to or alternatively, HEADS’s service receives information from cyclists who may act as sensors themselves when cycling. The outcome is a map that displays the latest air pollution levels recorded by sensors, described by people and reported by respective authorities.

3. Live disaster news coverage
Forest fires are one of the main causes of environmental degradation nowadays. In the summer of 2007, more than 80 people dead in Greece and 670,000 acres (2,711 km²) burned because of fires. The same year in California, 500,000 acres (2,027 km²) were devastated by the flames, causing at least 14 fatalities.

1http://www.libelium.com/wireless_sensor_networks_to_detect_forest_fires/
Current surveillance systems for forest fires lack in supporting real-time monitoring of every point of a region that a fire has spread. Solutions using wireless sensor networks, on the other hand, can be promoted. A network of sensor nodes can be installed in a forest to monitor fire’s evolution and spreading. The nodes can be equipped with sensors to measure temperature, humidity and gases from all points of a field continuously, day and night, and, provide fresh and accurate data quickly not only to the fire-fighting centre but to several other stakeholders like citizens that want to avoid any potential danger.

4.2.2 Why the scenario cannot be realised with the current state-of-the-art
With respect to the descriptions of Sections 2.2 and 2.3, NewsAsset scenarios refer to the adaptation and management of media items that leverage from heterogeneous wires and sources which have been brought into surface the last recent years. Social networks, mobile devices and sensor networks are some indicative examples of nodes which are being used by more and more users and services the last few years making them extremely important in the media industry. Even though the existing NewsAsset software framework manages info coming from several other wires mentioned in the previous section, there is still the need for an open source environment that will be able to incorporate heterogeneous platforms and networks and exploit the information each one provides in an integrated way.

4.2.3 Involved infrastructure
NewsAsset (NA) modules and other nodes that will be utilized for the following scenarios are presented in this section:

**NewsAsset general info**

NA is mostly built on MS .NET technology. Its database is accessed directly by the Application Server or peripheral applications using ADO.NET and ADO.

NA Application Server is the back-end component that communicates with the database, repository, the extended API, the web server, and the front-end windows desktop application, namely the NA Client. The web based front-end application and NA WebAccess offer certain client features through the web server. The extended API is a set of web-services that offer core functionality (news creation, editing, retrieval, and notifications) to third-party systems.

News items volumes created, received or distributed vary depending on the specific system organization needs. News created typical volume is approx. 2000 per day. News reception may vary from 5000 to 20000 items per day. News distribution may be 50000 to 200000 items per day.

The default import format is based on standard NewsML format. This format schema can be found in [http://www.iptc.org](http://www.iptc.org) site.

**NewsRoom module**

The NewsRoom module provides the core functionality for the writing, editing, approval and management of individual news items, stories and document folders.
The NewsRoom module includes the following features:

- Virtual role-based work areas managed by customized workflow processes and permissions
- Advanced text processing through an advanced built-in RTF text editor
- Spell checking available in multiple languages
- Support of various formats (text, photos, graphics, audio, video, stories, etc.)
- Identification and categorization of articles, images, graphics or stories with multiple identifiers, unlimited keywords, IPTC metadata, etc.
- Creating Stories through the linking of text with photos, graphics, and video files
- Saving multiple versions of files
- Organizing work and research in customisable folders in Windows style environment

Figure 6: NewsRoom module interface

**Archive module**

The Archive Module includes powerful functionality for storing, searching and managing safely the valuable digital assets of news agencies.

Through a simple to use interface users can quickly search and access all agency news items, all subscribed wire services as well as custom or personalized search results.

The Archive Module includes the following features:
• A comprehensive module for the management of all editorial content
• Creation of advanced keywords based thesaurus
• Classification according to IPTC standards
• Support of various formats (text, photos, graphics, audio, video, stories, etc.)
• Performing unlimited concurrent full text queries with expression builder using Boolean Operators
• Performing various advanced search queries with multiple parameters
• Saving complex search templates for future use (user defined filters)

Figure 7: View of the Archive module

NewsAsset Sender module

The NA Sender distribution module uses a configurable engine for distributing news assets to various delivery channels and formats.

NA Sender is the main content distribution platform implementing push protocols such as FTP, SMTP, and SOAP (web services). NA Sender is a multithreaded windows application which sends news items to subscribers by using delivery channels according to subscriptions.
The default import/export format is based on standard NewsML format\(^2\). NA Sender supports also XSL transformations, allowing the agency to easily define and support any new format by specifying the XSL transformations that will be applied over the native NewsML format.

The NA Sender module is intended for defining server installation parameters such as the maximum number of threads and distribution channels to be used, while provider distribution parameters are handles by system administrators. Over the user interface administrators are also able to check and handle (remove/stop) the items in the distribution queue, to check for error related to connection pulls and proceed with the appropriate actions. An embedded service named ‘Watcher’ comes together with NA Sender. This service constantly checks if the Sender application is running. When the Sender application stops or restarts, the Watcher sends to the administrators a relevant email with the relevant log file attached. When the NA Sender is not running, Watcher tries to restart it. If the restart succeeds then the watcher sends the relevant email to the administrators.

**NewsAsset Reception workflow**

NA includes two modules for receiving and ingesting news: NA Receiver and NA Aggregator. All imported data of any format are stored as relational data in a database. Large raw files (photos, graphics, video and audio) are stored in a file repository. Conversely, NewsML format is constructed on the fly from relational data. NewsML-G2 is the current IPTC news standard format that covers older news standards: NewsML 1, IIM, IPTC-7901, ANPA, Dublin Core, etc.

The following figure presents the reception workflow of NewsAsset.

\(^2\)NewsML samples are included in the Annex
**NewsAsset Receiver**

NA Receiver is a multithreaded windows application/service allowing unlimited number of receiving services working in parallel. It provides automated reception of news assets from multiple media, sources and wires, using a configurable engine for ingesting news assets (files) from any source and format that lie on specified file folders. It supports various standard industry data formats and also features serial port capability.

Through the NA Receiver Module, documents from an external Agency/Provider can be imported directly into Archive independent of the receiving channel (FTP, satellite etc.).

The agencies implementing IPTC/IIM/XMP/NewsML/EventsML/ANPA format (such as Reuters, Associated Press, etc) are fully supported. Metadata are automatically extracted for indexing. Items can be found later in the system based on their metadata (IPTC subjects, Keywords etc).

The default import format is based on the standard NewsML format. NA Receiver supports XSL transformations, allowing the Agency easily to define and support any new format by specifying the XSL transformations that will be applied over the original transforming it into NewsML format.
Image files are automatically recognized and imported with the NA Receiver manager. Images supporting embedded metadata (IPTC/IIM/XMP) are extracted automatically as part of import. Metadata of images are fully indexed for later search and retrieval.

Multiple Photos could be also imported using NA Receiver manager and directed to a specific user, group of users (role), or into the agency’s archive.

**NewsAsset Aggregator**

NA Aggregator is a multithreaded windows service allowing unlimited number of receiving services working in parallel. This module deals exclusively with RSS news feeds. A configurable set of RSS feeds is constantly checked for new items. RSS metadata and content are imported employing HTML filtering tools.

**A wireless sensor network (traffic parameters)**

For the following scenarios, a sensor network (for instance digital cameras, radar, active infrared and laser radar, etc.) will be utilized. The sensor network will be suited to derive various traffic parameters such as vehicle density, average vehicle velocity, beginning and end of congestion, length of congestion etc. It will be based on an Arduino/Android platform.

**A wireless sensor network (environmental parameters)**

As before, a wireless sensor network will be utilized to measure: carbon monoxide (CO), carbon dioxide (CO2), ozone (O3), nitrogen dioxide (NO2), temperature, humidity and noise. These sensors are installed either to specific areas or/and can be attached to a human body.

**Cloud environment**

A cloud environment will be utilized to host and process the real-time data that are being transmitted by the sensor network. It handles the application logic and should have a redundant setup such that system does not fail if one server gets offline.

**Social Networks (SNs)**

Social networks (SNs) (for instance Twitter, Instagram and Facebook) will be utilized to aggregate information posted by the users. Consequently, using an event detection system, information that is being aggregated by SNs is processed accordingly in order to discover potential information relevant to specific events that are happening and are connected to the information received by the sensor networks.

**Event detection system**

A software capable of analysing big data and hosted in a cloud environment, it exposes a web service to users/subscribers and provides the opportunity to share, collaborate, and make use of information generated from SNs via the Internet.

**Smartphone (mobile devices)**
A number of smartphones with GPS, GSM and IP connectivity are necessary for the following scenarios as well. Such devices can handle local decision logic and serve as a data transmission unit. The target platform should be Android. Furthermore, smartphone photo apps default to include location can be used since more and more people discover the value of location-tagging the photos they take. Indicative technical specifications are: Android OS: 4.0+, RAM: 0.5G+, 3G/4G connectivity, Camera: 5MP+.

**GSM network**

It handles data transmission between smartphones and cloud services.

**Other media sources and wires**

Multiple media sources and wires will be accessed to investigate if other news agencies have published useful information relevant to the scenarios. For instance, Athens News Agency, a Greek national journalism organization that reports information about traffic. These sources can also be other delivery channels (not connected to media domain) that are providing different kind of information, for instance real time weather forecast. A heavy rain in a specific area can be the reason for a traffic congestion.

**4.2.4 Involved actors**

The actors involved in the following scenarios are the editor, the journalist and the newsRoom team. The following figures (Figure 9 and Figure 10) present NewsAsset actors that participate in HEADS respective scenarios and the role that each one of them perform.
Figure 9: Presentation of NewsAsset editor’s role
4.3 Scenario 1: Broadcasting traffic conditions
This scenario describes how the system handles the process of creating news items by analysing and in sequence broadcasting real-time information aggregated by various media sources.

**Identifier:** B.1

**Involved actors:** Editor, journalist, news Room team

**Involved infrastructure:** NA NewsRoom, NA Receiver, NA Aggregator, sensor network (measuring traffic parameters), smartphone, GSM network, cloud servers, social networks, event detection system, media wires

**Requirements:** Tellu-01, Tellu-03, ATC-0x, Common-0x

---

3 “x” stands for all the requirements of the specific category. For this case, all the requirements issued by ATC use case provider.
**Preconditions:** The editor is monitoring NewsAsset (NA) basic console that is utilized to create a news item. This module is called integrated multimedia NewsRoom. The newsRoom team is aggregating information from various resources and posts them into the system. The newsRoom team is also managing NA reception workflow.

**Steps:**

1. The editor is creating a news item. He wants to broadcast traffic information and the potential detection of a traffic congestion due to the shutdown of a major highway in Athens.
2. The editor fills in the creation form providing the news item type, the name and the initial content. The new item will mention “......due to the official visit of a foreign Prime Minister, a 10km stretch of “Attiki Odos”, a major highway running from the airport through the center of town that carries an average of 100,000 cars on a summer business day, will be shut down for more than 3 hours....”.
3. The editor notifies the newsRoom team that is interested first of all in broadcasting a traffic/navigation map to give drivers a real-time view of what is happening on the roads and secondly in enhancing the news item with relevant material. The notification includes the basics of the news item that is being created.
4. The newsRoom team receives the notification and instantiates the HEADS service.
5. A sensor network delivering traffic parameters such as vehicle density, average vehicle velocity, beginning and end of congestion, length of congestion etc is being utilized.
6. By using a wireless data network the information is immediately sent back to a central system (for instance meshium/arduino radio) which will then transmit the data onto a dedicated server-hosting environment located in a cloud based platform.
7. The data are being processed/formatted and forwarded to a web-based service (HEADS service) built to manage the real-time data. The service gives users/subscribers the power to share, collaborate, and make use of information generated from a sensor network via the Internet.
8. The newsRoom team receives the input from the HEADS service (aggregation and analysis of the sensor network traffic information) and gets a sense of the traffic.
9. The newsRoom team creates a map that depicts the view of the roads near to “Attiki odos” regarding traffic conditions. Traffic congestions are discovered.
10. The map is sent back to the editor. The news item is now extended by the editor “......due to the official visit of a foreign Prime Minister, a 10km stretch of “Attiki Odos”, a major highway running from the airport through the center of town that carries an average of 100,000 cars on a summer business day, will be shut down for more than 3 hours. During the highway closure, you can use the following alternate routes to reach your destination. It is important to avoid the following as traffic congestion already exists ”
11. The editor needs to elaborate further and specifically he needs to know how bad the situation is at specific spots and what the people think about it (measure the tense). He sends a notification to the newsRoom team.
12. The newsRoom team instantiates again the HEADS service looking information on spot where a critical mass of people are driving, and presumably many of them are willingly uploading media items to the web via social networks.
13. Social networks (SNs) (for instance Twitter, Instagram and Facebook) are utilized to discover potential information relevant to the traffic congestions of the day.
14. In sequence, the data are transmitted to a cloud environment where they are being processed by the event detection system. The latter exposes a web service (HEADS service) to users/subscribers and provides the opportunity to share, collaborate, and make use of information generated from SNs via the Internet.

15. Smartphone photos can also be found to justify the posted information. Smartphone apps default to include location can be used since more and more people discover the value of location-tagging the photos they take.

16. The newsRoom team sends information to the editor which subsequently updates the news item: “......due to the official visit of a foreign Prime Minister, a 10km stretch of “Attiki Odos”, a major highway running from the airport through the center of town that carries an average of 100,000 cars on a summer business day, will be shut down for more than 3 hours. During the highway closure, you can use the following alternate routes to reach your destination. It is important to avoid the following as traffic congestion already exists. Especially at the specific road the drivers are mentioning delays that are more than 20 minutes. Photos uploaded at SNs show that roadblocks there badly increase the situation”.

17. newsRoom team have increasingly turned toward other media sources of information as well. Thus, information coming from different news wires and delivery channels are aggregated and analysed from HEADS service as well. These sources can also be other delivery channels (not connected to media domain) that are providing different kind of information, for instance real time weather forecast. A heavy rain that is predicted in the following minutes will be another reason of increasing the traffic congestion(s).

18. All these “nodes” mentioned above are sending information to an enhanced and configurable engine for ingesting news assets (files) from several sources, delivery channels and format. NA Receiver module receives the processed data and imports them to the newsRoom.

19. The editor is analyzing the information and passes the info to the journalist.

20. The journalist broadcasts the news.

**Post-conditions:** The news item is transmitted via the NA sender to all relevant channels.

### 4.4 Scenario 2: Broadcasting environmental conditions

This scenario is similar to the previous one. Thus, in this section we provide a brief description focusing only on the aspects that are different.

**Identifier:** B.2

More and more citizens are gradually fond of using bicycles for their own everyday transportation in the city. But the existing transportation routes and infrastructures are currently designed and built to serve vehicle drivers and public transportation users. In addition, environmental conditions in the city center and especially air pollution levels, although improved due to multiple measures and precautions are still preventing people from using their bicycles as an everyday means of transportation. Especially people suffering from various respiratory diseases such as asthma or other, but still wish to follow the cycling transportation option, are in great need of information that will allow them to optimize their cycling experience.
The editor is using again HEADS service to draw the most preferable cycling routes to the city center by accessing not only transportation related information (previous scenario) but more importantly environmental information, namely air pollution levels along the potential cycling route.

Wireless sensor networks have been already deployed in Athens city centre to monitor the concentration of dangerous gases for citizens. In addition to or alternatively, HEADS service, receives information from cyclists who may act as sensors themselves when cycling.

The outcome is a map that displays the latest air pollution levels recorded by sensors, described by people and reported by respective authorities.

4.5 Scenario 3. “Live disaster news coverage”

As already mentioned above, forest fires are one of the main causes of environmental degradation nowadays. Current surveillance systems lack in supporting real-time monitoring of every point of a region that a fire has spread. Solutions using wireless sensor networks, on the other hand, can be promoted. A network of sensor nodes can be installed in a forest to monitor fire’s evolution and spreading. The nodes can be equipped with sensors to measure temperature, humidity and gases from all points of a field continuously, day and night, and, provide fresh and accurate data quickly not only to the fire-fighting center but to several other stakeholders like citizens that want to avoid any potential danger.

**Identifier:** B.3

**Involved actors:** Editor, journalist/reporter/photographers, newsRoom team

**Involved infrastructure:** NA NewsRoom, NA Receiver, NA Aggregator, NA Sender, sensor network (measuring temperature, humidity, gases and optionally weather conditions), mobile devices, GSM network, cloud servers, social networks, event detection system, media wires.

**Requirements:** Tellu-01, Tellu-03, ATC-0x, Common-0x

**Scenario description:**

Athens News Agency, a Greek national journalism organization, reports that a forest fire in Peloponnese has been detected. “NewsBeast”, a Greek TV channel wants to cover live the event and thus want to send field journalists and reporters to specific locations near the file in order to broadcast useful information for all. This aspect is considered “breaking news” and several actions must be performed very quickly.

While the field personnel is on their way to the location of the event, in the newsRoom several people are trying to aggregate as much information as possible in order to define the most appropriate locations to distribute the team.

More specifically the following sources are applicable:

- First a sensor network that is able to measure temperature, humidity, gases and optionally weather conditions. The info can be processed as mentioned before.
• Secondly, social networks (SNs) (for instance Twitter, Instagram and Facebook) are utilized to discover a disaster photo, explaining what’s going on at photo’s location. Smartphone photo apps set to include location by default can be used since more and more people discover the value of location-tagging the photos they take. The info can be processed as mentioned before.
• Finally, multiple media other sources and wires are accessed to investigate if other delivery channels (news agencies, public authorities, etc.) have published useful information regarding the event. For instance, National Weather Service is broadcasting that the direction of the wind in the area will change and its speed will increase.

The newsRoom team analyzes all info and draws a map that presents the current location of the fire and its potential spreading to the nearby area. In order to cover live the event, it is extremely important for the field personnel (journalists, photographers, etc.) to be guided to the areas/locations where the fire is going to evolve. To do so, the newsRoom team requests from the personnel to define their location. A respective message was sent (NA Sender module is utilized) to their mobile devices (the mobile devices are using Android operating system) and all are replying accordingly. Based on that information the newsRoom team decides where each field journalist should be appointed. Furthermore, since it is important to send them to these locations as quickly as possible, the newsRoom team creates the most efficient routes to guide people. This information can be retrieved from existing web services, like for instance Google maps. Finally, field personnel is receiving on their mobile devices the location that should be moved to during the following minutes and the most appropriate route to follow. A confirmation message is sent when they reach their destination.
5 Design Time Scenarios

The scenarios in this section describe how different system actors would like to use the portfolio of HEADS tools to perform different tasks. The described scenarios only cover a small selection of the foreseen use-cases during system development and are not meant to be exhaustive or very detailed. Nevertheless, they give an idea on how a system developer would like to work while developing services with HEADS tools/methodology.

5.1 Involved HEADS tools
The described scenarios involve HEADS tools consisting of:

- HEADS design tool - the tool where the service designer designs the services
- HEADS code generation tool - the tool where the service designer generates code to targeted platforms
- HEADS service provisioning tool - the tool that handles provisioning of code to distributed systems/devices/platforms

5.2 Involved actors
Platform expert, service designer, service operator
5.3 Scenario 1: System design by module based graphical programming using wizards

This scenario handles how the service designer captures his/her knowledge of the system design using the HEADS design tool.

**Identifier:** C.1

**Involved actors:** Service designer
**Involved infrastructure**: HEADS design tool

**Requirements**: Tellu-06, ATC-01, ATC-04, Common-11, Common-15

**Preconditions**: The service designer has an initial understanding of the system architecture and functionality. The service designer has identified a set of system modules/components that can be modeled in the HEADS design tool. The modules represent well-defined parts of system architecture and hold both behavioral and structural information (e.g., state machines, activity diagrams, class structures). Modules can represent physical equipment as well as logic control parts of system.

**Steps:**

1. Service designer creates system project in HEADS tool.
2. Service designer opens new system wizard and creates new module.
3. Service designer starts to specify module connections and module behavior using graphical design tool.

**Post-conditions**: Service designer has captured his/her knowledge in a system design model.

5.4 Scenario 2: System design using module library
This scenario handles how the service designer can use module libraries for prebuilt functionalities.

**Identifier**: C.2

**Involved actors**: Service designer

**Involved infrastructure**: HEADS design tool

**Requirements**: ATC-04, Tellu-05, Common-01

**Preconditions**: The service designer is working on a system model where he/she wants to include prebuilt modules

**Steps:**

1. Service designer opens module library in HEADS tool.
2. Service designer chooses module to import into system model.
3. Service designer graphically connects imported module to existing design.

**Post-conditions**: Service designer has imported a library module to a system design model.

5.5 Scenario 3: Verifying design by automated simulation/inspection
This scenario handles how the service designer can use simulations to verify system design.

**Identifier**: C.3

**Involved actors**: service designer
Involved infrastructure: HEADS design tool

Requirements: Common-04, Common-06

Preconditions: service designer has a system model where he/she wants to check system design. Service designer has recorded a set of system inputs.

Steps:
1. Service designer chooses a system model and clicks the Verify button.
2. HEADS design tool gives visual feedback on design flaws, e.g. deadlocks, unreachable code/states

Alternatively
1. Service designer plays the set of system inputs to the system model.
2. Service designer verifies the set of system outputs. (This verification could perhaps be automatically performed by checking that operational semantics (e.g. state machinery) fulfill denotational semantics (e.g. sequence/activity diagrams).

Post-conditions: Service designer has verified system model.

5.6 Scenario 4: Generating platform code from tool using platform library
This scenario handles how the service designer can generate platform specific code

Identifier: C.4

Involved actors: Service designer

Involved infrastructure: HEADS code generation tool

Requirements: ATC-01, Common-05, Common-08

Preconditions: Service designer wants to generate code for specific platforms. Platform expert provides target platform in platform library.

Steps:
1. Service designer specifies that module will be targeted to selected platform.
2. HEADS code generation tool generates code.
3. HEADS code generation tool saves code to HEADS provisioning system.

Post-conditions: Service designer has generated platform specific code and saved it in HEADS provisioning system.

5.7 Scenario 5: Provision/upgrade module functionality to devices
This scenario handles how the service operator can distribute module functionality to devices.

Identifier: C.5
**Involved actors:** Service operator, device users (security guards)

**Involved infrastructure:** HEADS system provisioning tool

**Requirements:** Tellu-04

**Preconditions:** Service operator wants to distribute generated module functionality to devices registered in HEADS service provisioning system.

**Steps:**

1. Service operator selects devices that will have new functionality downloaded.
2. Device user accepts new download of new functionality.
3. Device downloads platform specific module functionality.

**Post-conditions:** Generated module functionality is downloaded to device.
6 Requirements elicitation

The following tables present the requirements defined in the previous scenarios. The objectives mentioned in the following tables are those defined by the project consortium at this early phase of the project. These objectives will guide technical activities throughout project’s duration.

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Tellu – 01</th>
<th>Type</th>
<th>Functional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Short name</td>
<td>Body sensors</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objectives</td>
<td>Primary: Objective 4: Manage the flow of big data across the future computing continuum  Secondary: Objective 5: Provide seamless dynamic deployment and evolution of HD-Services</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>Body sensors monitor body conditions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rationale</td>
<td>HEADS should demonstrate that big data can be handled</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Related scenario</td>
<td>A.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contributors</td>
<td>Primary: WP3 Secondary: WP4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Priority</td>
<td>MUST</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Satisfaction criteria</td>
<td>Utilization of specific body sensors that monitor the body conditions defined in the relevant scenario</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Tellu – 02</th>
<th>Type</th>
<th>Functional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Short name</td>
<td>P2P media communication</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objectives</td>
<td>Primary: Objective 4: Manage the flow of big data across the future computing continuum</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>P2P media communication must be possible</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rationale</td>
<td>P2P communication should be a communication mode to make system robust</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Related scenario</td>
<td>A.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contributors</td>
<td>Primary: WP3 Secondary: WP5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Priority</td>
<td>SHOULD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Satisfaction criteria</td>
<td>P2P communication utilized during system’s operation</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Data security (integrity and authenticity of data)

**Identifier**: Tellu – 03  
**Type**: Non-Functional  
**Short name**: Data security (integrity and authenticity of data)  
**Objectives**
- **Primary**: Objective 4: Manage the flow of big data across the future computing continuum  
- **Secondary**: Objective 5: Provide seamless dynamic deployment and evolution of HD-Services  
**Description**: Data security (integrity and authenticity of data)  
**Rationale**: Information through the HEADS platform should be handled in a secure way to ensure integrity and authenticity of data.  
**Related scenario**: A.3  
**Contributors**: Primary: WP3  
**Priority**: MUST  
**Satisfaction criteria**: Secure exchange of information

### Life-cycle management for distributed code

**Identifier**: Tellu – 04  
**Type**: Functional  
**Short name**: Life-cycle management for distributed code  
**Objectives**
- **Primary**: Objective 3: Make the HD-service lifecycle safe predictable and consistent  
- **Secondary**: Objective 5: Provide seamless dynamic deployment and evolution of HD-Services  
**Description**: Life-cycle management for distributed code  
**Rationale**: HEADS should have a life-cycle management system for handling distribution and upgrade of code  
**Related scenario**: C.5  
**Contributors**: Primary: WP5  
**Priority**: MUST  
**Satisfaction criteria**: Life-cycle management for distributed code achieved
<table>
<thead>
<tr>
<th>Identifier</th>
<th>Tellu – 05</th>
<th>Type</th>
<th>Functional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Short name</td>
<td>Module library</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objectives</td>
<td>Primary: Provide service developers with new abstractions to develop HD-service</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Secondary: Provide platform experts with efficient means to capture their knowledge</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>A design tool equipped with a module library for service designers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rationale</td>
<td>HEADS design tool should have a module library where service designers can search for existing modules and integrate them in design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Related scenario</td>
<td>C.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contributors</td>
<td>Primary: WP6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Priority</td>
<td>MUST</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Satisfaction criteria</td>
<td>A module library that can search for existing modules and integrate them in design is implemented</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Tellu – 06</th>
<th>Type</th>
<th>Functional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Short name</td>
<td>Graphical tool</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objectives</td>
<td>Primary: Provide platform experts with efficient means to capture their knowledge</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Secondary: Provide service developers with new abstractions to develop HD-service</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>HEADS design tool should be graphically oriented</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rationale</td>
<td>HEADS design tool should be graphically oriented</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Related scenario</td>
<td>C.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contributors</td>
<td>Primary: WP6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Priority</td>
<td>SHOULD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Satisfaction criteria</td>
<td>A graphically oriented design tool is implemented</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identifier</td>
<td>ATC – 01</td>
<td>Type</td>
<td>Functional</td>
</tr>
<tr>
<td>------------</td>
<td>----------</td>
<td>------</td>
<td>------------</td>
</tr>
<tr>
<td>Short name</td>
<td>Generate executable code</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Objectives</strong></td>
<td>Primary</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Objective 2: Provide platform experts with efficient means to capture their knowledge</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Secondary</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Objective 5: Provide seamless dynamic deployment and evolution of HD-Services</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td>HEADS transformations should generate executable code in an automated way regardless the underlying devices and platforms</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Rationale</strong></td>
<td>Once a “node” is incorporated into the system, the service developer would be able to use it without knowing its specifications</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Related scenario</strong></td>
<td>C.x⁴</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Contributors</strong></td>
<td>Primary: WP4  Secondary: WP2</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Priority</strong></td>
<td>MUST</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Satisfaction criteria</strong></td>
<td>A graphically oriented design tool is implemented</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

⁴ “x” stands for all the sub-scenarios of the category “Design Time Scenarios”.

Version 1.0: Final version 2014-01-29
<table>
<thead>
<tr>
<th>Identifier</th>
<th>ATC – 02</th>
<th>Type</th>
<th>Non-Functional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Short name</td>
<td>Secured transactions and data</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objectives</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Primary</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objective 1: Provide service developers with new abstractions to develop HD-service</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Secondary</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objective 2: Provide platform experts with efficient means to capture their knowledge</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>In the media industry, all transactions between personnel must be secured. Moreover, all the media items must be hosted in a secure environment.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rationale</td>
<td>The security influences the selection of the appropriate nodes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Related scenario</td>
<td>A.x - B.x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contributors</td>
<td>Primary: WP4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Priority</td>
<td>MUST</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Satisfaction criteria</td>
<td>Devises and platforms selected for the respective scenario are offering secured transactions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identifier</td>
<td>ATC – 03</td>
<td>Type</td>
<td>Non-Functional</td>
</tr>
<tr>
<td>------------</td>
<td>----------</td>
<td>------</td>
<td>----------------</td>
</tr>
<tr>
<td>Short name</td>
<td>Scalability and performance</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Objectives</strong></td>
<td>Primary</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Objective 4: Manage the flow of big data across the future computing continuum</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Secondary</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Objective 2: Provide platform experts with efficient means to capture their knowledge</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td>In order to address the requirements of the media industry when it comes to aggregating and analysing media items, storage and compute resources can be added and removed within a very short time on demand.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Rationale</strong></td>
<td>Utilize “nodes” that adopt rapid scaling</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Related scenario</strong></td>
<td>A.x - B.x</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Contributors</strong></td>
<td>Primary: WP4  Secondary: WP2</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Priority</strong></td>
<td>MUST</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Satisfaction criteria</strong></td>
<td>In case of cloud platforms, elasticity is adopted</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identifier</td>
<td>ATC – 04</td>
<td>Type</td>
<td>Functional</td>
</tr>
<tr>
<td>------------</td>
<td>----------</td>
<td>------</td>
<td>------------</td>
</tr>
<tr>
<td>Short name</td>
<td>Efficient incorporation of new data sources</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objectives</td>
<td>Primary</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Objective 4: Manage the flow of big data across the future computing continuum</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Secondary</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Objective 2: Provide platform experts with efficient means to capture their knowledge</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>Efficiently process and filter high volume of data (coming from social and sensor networks) in order to avoid flooding and congesting the system with low quality unprocessed data.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rationale</td>
<td>Media-related information emerging from new sources like sensor and social networks can be efficiently used by the system</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Related scenario</td>
<td>B.x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contributors</td>
<td>Primary: WP4  Secondary: WP2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Priority</td>
<td>MUST</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Satisfaction criteria</td>
<td>Sensor and social networks are utilized. Highly ranked usability of the information emerging from these data sources</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**Identifier**: ATC – 05  
**Type**: Functional

<table>
<thead>
<tr>
<th><strong>Short name</strong></th>
<th>Lock-in interoperability</th>
</tr>
</thead>
</table>

**Objectives**

- **Primary**
  - Objective 2: Provide platform experts with efficient means to capture their knowledge
  
- **Secondary**
  - Objective 6: Demonstrate its interest and generability and disseminate HEADS result

**Description**

When composing a service, it is important for the designer to be aware of the architecture and specifications of the platforms he is going to use.

**Rationale**

Consider how today’s decisions regarding the selected nodes (for instance a cloud platform) will affect options in the future – Avoid lock-ins

**Related scenario**

B.x

**Contributors**

- **Primary**: WP4

**Priority**

MAY

**Satisfaction criteria**

Utilize “nodes” with open architecture and specifications

---

**Identifier**: ATC – 06  
**Type**: Functional

<table>
<thead>
<tr>
<th><strong>Short name</strong></th>
<th>Open data sets</th>
</tr>
</thead>
</table>

**Objectives**

- **Primary**
  - Objective 4: Manage the flow of big data across the future computing continuum
  
- **Secondary**
  - Objective 3: Resource-constrained devices and networks

**Description**

The realization of a useful media scenario highly depends on the data that are used. Thus, the data sets should be opened and available for further analysis

**Rationale**

HEADS services should offer open data sets relevant to the selected scenarios

**Related scenario**

A.x - B.x

**Contributors**

- **Primary**: WP4  
- **Secondary**: WP5

**Priority**

SHOULD

**Satisfaction criteria**

Sensor and social networks offer for further analysis open data sets
### Heterogeneous data sets

**Identifier** | ATC – 07
---|---
**Type** | Functional

**Short name** | Heterogeneous data sets

**Objectives**
- **Primary**
  - Objective 4: Manage the flow of big data across the future computing continuum
  - Objective 5: Methodology and tool integration
- **Secondary**

**Description**
HEADS system should be able to incorporate heterogeneous data sets and address the respective challenges of analysing them.

**Rationale**
Heterogeneous infrastructure is valid only if it exposes heterogeneous data sets

**Related scenario** | B.x

**Contributors**
- **Primary**: WP4
- **Secondary**: WP5

**Priority** | MUST

**Satisfaction criteria**
Heterogeneous data sets are utilized

---

### Timeliness

**Identifier** | ATC – 08
---|---
**Type** | Non-Functional

**Short name** | Timeliness

**Objectives**
- **Primary**
  - Objective 4: Manage the flow of big data across the future computing continuum
  - Objective 5: Methodology and tool integration
- **Secondary**

**Description**
The larger the data set to be processed, the longer it will take to analyze. The timeliness challenge of analyzing big data should be addressed in the context of HEADS.

**Rationale**
The design of a system that effectively deals with size is likely also to result in a system that can process a given size of data set faster. This can be acceptable only for the relevant to the scenarios analyzed information

**Related scenario** | A.x - B.x

**Contributors**
- **Primary**: WP4
- **Secondary**: WP5

**Priority** | SHOULD

**Satisfaction criteria**
HEADS system’s response time satisfies the end users of the scenarios
<table>
<thead>
<tr>
<th>Identifier</th>
<th>ATC – 09</th>
<th>Type</th>
<th>Functional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Short name</td>
<td>Rich palette of visualizations</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objectives</td>
<td>Primary</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Objective 2: Provide platform experts with efficient means to capture their knowledge</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Secondary</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Objective 1: Provide service developers with new abstractions to develop HD-service</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>Designers should be able to offer services that pack and present results in powerful visualizations that assist interpretation, and support user collaboration</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rationale</td>
<td>Systems with a rich palette of visualizations become important in conveying to the users the results of the queries in a way that is best understood in the particular domain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Related scenario</td>
<td>A.x - B.x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contributors</td>
<td>Primary: WP4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Priority</td>
<td>MAY</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Satisfaction criteria</td>
<td>Rich palette of visualization for the analysed data is used</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identifier</td>
<td>COMMON – 01</td>
<td>Type</td>
<td>Functional</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------</td>
<td>---------------</td>
<td>------------</td>
</tr>
<tr>
<td>Short name</td>
<td>Integration with legacy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objectives</td>
<td>Primary</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Objective 2: Provide platform experts with efficient means to capture their knowledge</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Secondary</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Objective1: Provide service developers with new abstractions to develop HD-service</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>HEADS should consider the possibility of integrating legacy APIs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rationale</td>
<td>HEADS should enable platform experts to reuse existing third-party/legacy APIs and should not impose to redevelop them from scratch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Related scenario</td>
<td>A.x – B.x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contributors</td>
<td>Primary: WP2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Priority</td>
<td>MUST</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Satisfaction criteria</td>
<td>Legacy APIs are used</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Identifier</th>
<th>COMMON – 02</th>
<th>Type</th>
<th>Functional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Short name</td>
<td>Local decision making</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objectives</td>
<td>Primary</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Objective 4: Manage the flow of big data across the future computing continuum</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Secondary</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Objective 5: Provide seamless dynamic deployment and evolution of HD-Services</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>Local decision making</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rationale</td>
<td>HEADS should enable decisions to be made as close as possible to the place where events supporting those decisions are produced</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Related scenario</td>
<td>C.x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contributors</td>
<td>Primary: WP5  Secondary: WP6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Priority</td>
<td>MUST</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Satisfaction criteria</td>
<td>HEADS supports local decision making</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identifier</td>
<td>COMMON – 03</td>
<td>Type</td>
<td>Functional</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
<td>---------------</td>
<td>------------</td>
</tr>
<tr>
<td><strong>Short name</strong></td>
<td>Reduced development time</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Objectives</strong></td>
<td></td>
<td>Primary</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Objective 3: Make the HD-service lifecycle safe predictable and consistent</td>
<td>Secondary</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Objective 6: Demonstrate its interest and generability and disseminate HEADS result</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td>Reduce development time compared to common practice</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Rationale</strong></td>
<td>HEADS should reduce the development time of heterogeneous and distributed services compared to the state of the art/practice</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Related scenario</strong></td>
<td>C.x</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Contributors</strong></td>
<td>Primary: WP2</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Priority</strong></td>
<td>MUST</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Satisfaction criteria</strong></td>
<td>HEADS reduces development time of distributed services</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identifier</td>
<td>COMMON – 04</td>
<td>Type</td>
<td>Functional</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>Short name</td>
<td>Consistent application</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objectives</td>
<td></td>
<td>Primary</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Objective1: Provide service developers with new abstractions to develop HD-service</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Secondary</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Objective 5: Provide seamless dynamic deployment and evolution of HD-Services</td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ensure consistency of total application</td>
<td></td>
</tr>
<tr>
<td>Rationale</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>HEADS should provide tools to allow the design of applications (HD-services) that can be distributed and deployed in the infrastructure in a sound way. I.e. the application should behave as expected when e.g. asynchronous channels are used.</td>
<td></td>
</tr>
<tr>
<td>Related scenario</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>C.x</td>
<td></td>
</tr>
<tr>
<td>Contributors</td>
<td></td>
<td>Primary: WP2 Secondary: WP4</td>
<td></td>
</tr>
<tr>
<td>Priority</td>
<td></td>
<td>MUST</td>
<td></td>
</tr>
<tr>
<td>Satisfaction criteria</td>
<td></td>
<td>Consistency is ensured regardless of the infrastructure</td>
<td></td>
</tr>
</tbody>
</table>
### Use case scenario definitions and initial requirements

<table>
<thead>
<tr>
<th>Identifier</th>
<th>COMMON – 05</th>
<th>Type</th>
<th>Functional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Short name</td>
<td>Selecting the nodes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objectives</td>
<td>Primary</td>
<td>Objective 2: Provide platform experts with efficient means to capture their knowledge</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Secondary</td>
<td>Objective 3: Make the HD-service lifecycle safe predictable and consistent</td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>Specify, search and select the appropriate nodes of targeted infrastructure</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rationale</td>
<td>The methodology should consider how the nodes of the targeted infrastructure will be selected. HEADS should provide a tool/way that node’s characteristics, for instance a cloud platform, will be specified, searched and at the end selected. The selected platform should have the most compatible offerings</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Related scenario</td>
<td>A.x – B.x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contributors</td>
<td>Primary: WP3 Secondary: WP5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Priority</td>
<td>MUST</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Satisfaction criteria</td>
<td>With respect to HEADS scenarios, the appropriate nodes are selected</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identifier</td>
<td>COMMON – 06</td>
<td>Type</td>
<td>Functional</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
<td>------</td>
<td>------------</td>
</tr>
<tr>
<td><strong>Short name</strong></td>
<td>Evaluation process</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Objectives</strong></td>
<td>Primary</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Objective 5: Provide seamless dynamic deployment and evolution of HD-Services</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Secondary</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Objective 3: Make the HD-service lifecycle safe predictable and consistent</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td>Support the evaluation of the process</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Rationale</strong></td>
<td>The system should be verified against its prior behaviour. HEADS should provide the ways to conduct assessments through both functional and non-functional tests: a) both systems have the same functionalities, b) verify that the required targeted node's requirements are met</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Related scenario</strong></td>
<td>A.x – B.x</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Contributors</strong></td>
<td>Primary: WP4</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Priority</strong></td>
<td>MUST</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Satisfaction criteria</strong></td>
<td>The “new” system complies to the functionalities of its prior behaviour</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Identifier
**COMMON – 07**

**Type**
Functional

<table>
<thead>
<tr>
<th>Short name</th>
<th>Capabilities and Requirements</th>
</tr>
</thead>
</table>

**Objectives**
- **Primary**
  Objective 2: Provide platform experts with efficient means to capture their knowledge
- **Secondary**
  Objective 1: Provide service developers with new abstractions to develop HD-service

**Description**
HEADS should provide means to document and formalize the capabilities of the different platforms (e.g. in terms of CPU/RAM resources, I/O throughput, etc.) as well as define requirements (possibly in an automated way e.g. by statistical analysis) about the individual components composing the HD-Services.

**Rationale**
This should provide guidance to drive the deployment of the HD-services.

**Related scenario**
A.x – B.x

**Contributors**
Primary: WP5  Secondary: WP2

**Priority**
MUST

**Satisfaction criteria**
Different platforms are efficiently documented from the HEADS offered means

---

### Identifier
**COMMON – 08**

**Type**
Functional

<table>
<thead>
<tr>
<th>Short name</th>
<th>Integration of new platforms</th>
</tr>
</thead>
</table>

**Objectives**
- **Primary**
  Objective 2: Provide platform experts with efficient means to capture their knowledge

**Description**
HEADS should provide practical means to target new platforms that have not been integrated as part of the "standard HEADS library".

**Rationale**
Sustainability of the HEADS library is ensured.

**Related scenario**
C.x

**Contributors**
Primary: WP5  Secondary: WP2

**Priority**
MUST

**Satisfaction criteria**
Different platforms are efficiently documented from the HEADS offered means
### Use case scenario definitions and initial requirements

#### Identifier: COMMON – 09

**Type:** Non-Functional

**Short name:** Combination of independent data streams

**Objectives:**

- **Primary**
  - Objective 4: Manage the flow of big data across the future computing continuum

**Description:**

HEADS should provide means to combine independent sensor streams to achieve higher precision output e.g. combine for fire fighter scenario sensors for smoke particles, temperature, and carbon monoxide centration in order to raise an alarm based on the combination of only almost dangerous changes in each sensor in the same time period.

Verify the functioning of the sensor by continuously receiving events. Furthermore combine those alarms with unusual occurrence of reports about fire in social media based on text analysis.

**Rationale:**

Achieve higher precision output

**Related scenario:** A.x – B.x

**Contributors:**

- **Primary:** WP5
- **Secondary:** WP2

**Priority:** MUST

**Satisfaction criteria:**

Several independent sensor streams are combined to achieve higher precision output

---

#### Identifier: COMMON – 10

**Type:** Non-Functional

**Short name:** Virtual devices

**Objectives:**

N/A

**Description:**

HEADS must provide data processing and persistence mechanism for the platforms that can't provide required level of service

**Rationale:**

The contextual areas of the scenarios can be extended by using virtual devises. HEADS applicability increases

**Related scenario:** A.x – B.x

**Contributors:**

N/A

**Priority:** SHOULD

**Satisfaction criteria:**

Virtual devices are utilized if needed

---

Version 1.0: Final version 2014-01-29
## Use case scenario definitions and initial requirements

<table>
<thead>
<tr>
<th>Identifier</th>
<th>COMMON – 11</th>
<th>Type</th>
<th>Functional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Short name</td>
<td>Service composition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objectives</td>
<td>N/A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>HEADS must support easy service composition. For example it should be possible to access media storage via published API from external applications like Early Warning System.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rationale</td>
<td>HEADS application will be accepted as a promising solution by the relevant communities – Easy to adopt</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Related scenario</td>
<td>C.x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contributors</td>
<td>N/A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Priority</td>
<td>SHOULD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Satisfaction criteria</td>
<td>Service composition can be done easily</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Identifier</th>
<th>COMMON – 12</th>
<th>Type</th>
<th>Non-Functional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Short name</td>
<td>QoS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objectives</td>
<td>N/A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>HEADS must provide mechanism for definition and consumption of different service quality levels (~Service Level Agreement, Connectivity/Availability). For example priority level, capacity, throughput or latency can be addressed as QoS attributes.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rationale</td>
<td>HEADS will be able to increase its acceptance from all stakeholders if it addresses concrete QoS attributes.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Related scenario</td>
<td>C.x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contributors</td>
<td>N/A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Priority</td>
<td>MUST</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Satisfaction criteria</td>
<td>QoS attributes are defined for the HEADS services</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Use case scenario definitions and initial requirements

#### Identifier:
| COMMON – 13 |
| Type | Non-Functional |

#### Short name:
Monitoring

#### Objectives:
N/A

#### Description:
Models should be updated with monitoring information coming from the running system (load, etc.) in order to support decision making e.g., should the system be adapted/re-deploy

#### Rationale:
Several metrics (like loading, fault tolerance, response time) should be monitored to ensure system’s efficient behaviour

#### Related scenario:
A.x – B.x – C.x

#### Contributors:
N/A

#### Priority:
SHOULD

#### Satisfaction criteria:
Models are enhanced with monitoring information from the running system

### Identifier:
| COMMON – 14 |
| Type | Non-Functional |

#### Short name:
Load – management

#### Objectives:
N/A

#### Description:
It should be possible to easily extend system and balance the load

#### Rationale:
Ensure load balancing to minimize the cost of resources

#### Related scenario:
A.x – B.x – C.x

#### Contributors:
N/A

#### Priority:
SHOULD

#### Satisfaction criteria:
Load managing is achieved
<table>
<thead>
<tr>
<th>Identifier</th>
<th>COMMON – 15</th>
<th>Type</th>
<th>Functional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Short name</td>
<td>A methodology blueprint</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| Objectives | **Primary**
  Objective1: Provide service developers with new abstractions to develop HD-service
  **Secondary**
  Objective 6: Demonstrate its interest and generability and disseminate HEADS result |
<p>| Description | HEADS should provide a methodology blueprint consisting of: a) a graphical representation (outline) of the process integrated within the main tooling suite, and easily accessible throughout the suite, b) a cheat sheet explaining the overall process and details for each task and c) links to launch required tools for each task from the cheat sheet or the graphical outline |
| Rationale   | Ease the complexity of the process and increase HEADS acceptance from all stakeholders |
| Related scenario | C.x |
| Contributors | <strong>Primary</strong>: WP4  <strong>Secondary</strong>: WP2 |
| Priority    | MUST |
| Satisfaction criteria | A methodology blueprint is offered to the service designers |</p>
<table>
<thead>
<tr>
<th>Identifier</th>
<th>COMMON – 16</th>
<th>Type</th>
<th>Functional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Short name</td>
<td>Robust testing and validation of HD-services</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objectives</td>
<td>N/A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>HEADS should provide the means (methods, tools and infrastructure) to test and subsequently validate HD-services against their intended objective and scope. This requirements is connected with Common 13 since monitoring is also part of the testing. By continuous monitoring of HD-services, deviations can be detected and corrective actions can be taken by the HD-service designers.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rationale</td>
<td>Minimize the costs and the risks of failures</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Related scenario</td>
<td>N/A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contributors</td>
<td>Primary: WP4  Secondary: WP2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Priority</td>
<td>MUST</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Satisfaction criteria</td>
<td>Deployment and execution of large-scale test beds for HD-services</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
7 Annex

7.1 Annex 1. NewsML samples

Sample 1.

```xml
<?xml version="1.0" encoding="Windows-1253"?>
<NewsML>
  <NewsEnvelope>
    <DateAndTime>20120217T130221+0200</DateAndTime>
    <NewsProduct FormalName="Οικονομικό Δελτίο"/>
    <Priority FormalName="8"/>
  </NewsEnvelope>
  <NewsItem>
    <Identification>
      <NewsIdentifier>
        <ProviderId>ana-mpe.gr</ProviderId>
        <DateId>20120217</DateId>
        <NewsItemId>11077</NewsItemId>
        <RevisionId Update="N" PreviousRevision="0">1</RevisionId>
        <PublicIdentifier>urn:newsml:ana-mpe.gr:20120217:11077:1</PublicIdentifier>
      </NewsIdentifier>
      <NameLabel>
        <![CDATA[]]>
      </NameLabel>
      <DateLabel>17/2/2012 13:02:21</DateLabel>
    </Identification>
    <NewsManagement>
      <NewsItemType FormalName="News"/>
      <FirstCreated>20120217T130202+0200</FirstCreated>
      <ThisRevisionCreated>20120217T130202+0200</ThisRevisionCreated>
      <Status FormalName="Usable"/>
    </NewsManagement>
    <NewsComponent Essential="no" EquivalentsList="no">
```

Version 1.0: Final version 2014-01-29
<TopicSet FormalName="NewsSubjects">
  <Topic>
    <TopicType FormalName="Subject" Scheme="IptcTopicType"/>
    <FormalName Scheme="IptcSubjectCodes">04000000</FormalName>
    <Description xml:lang="en" Variant="Category">EBF</Description>
    <Description xml:lang="el" Variant="Name">Economy-Business-Finance</Description>
    <Description xml:lang="en" Variant="Name">Economy-Business-Finance</Description>
  </Topic>
</TopicSet>

<NewsLines>
  <HeadLine xml:lang="x-default">
    <![CDATA[test]]>
  </HeadLine>
  <DateLine>17/2/2012</DateLine>
  <CreditLine>credit</CreditLine>
  <CopyrightLine>© 2012 ΑΠΕ</CopyrightLine>
</NewsLines>

<AdministrativeMetadata>
  <FileName>11077.xml</FileName>
  <Source>
    <Party FormalName="mySource"/>
  </Source>
</AdministrativeMetadata>

<DescriptiveMetadata>
  <Language FormalName="EL"/>
  <Genre FormalName="Τρέχουσα είδηση"/>
  <SubjectCode>
    <Subject FormalName="04000000" Scheme="IptcSubjectCodes"/>
  </SubjectCode>
  <Property FormalName="Country" Value="GRC"/>
  <Property FormalName="City" Value="Αθήνα"/>
</DescriptiveMetadata>
Sample 2.

```xml
<?xml version="1.0" encoding="utf-8"?>
<NewsML>
    <NewsEnvelope>
        <DateAndTime>20081203T162852+0200</DateAndTime>
        <NewsProduct FormalName="Texts"/>
        <Priority FormalName="5"/>
    </NewsEnvelope>
    <NewsItem>
        <Identification>
            <NewsIdentifier>
                <ProviderId>pap.pl</ProviderId>
                <DateId>20081203</DateId>
                <NewsItemId>959130</NewsItemId>
            </NewsIdentifier>
        </Identification>
    </NewsItem>
</NewsML>
```
D1.1. Use case scenario definitions and initial requirements

<RevisionId Update="N" PreviousRevision="O">1</RevisionId>

</RevisionId>

<PublicIdentifier>urn:newsml:pap.pl:20081203:959130:1</PublicIdentifier>

</NewsIdentifier>

</NameLabel>

<![CDATA[[]]]>

</NameLabel>

<DateLabel>3/12/2008 16:28:52</DateLabel>

</Identification>

<NewsManagement>

<NewsItem Type FormalName="News"/>

<FirstCreated>20081203T154034+0200</FirstCreated>

<ThisRevisionCreated>20081203T154034+0200</ThisRevisionCreated>

<Status FormalName="Usable"/>

<AssociatedWith FormalName="SeeAlso" NewsItem="urn:newsml:pap.pl:20000927:1359792:1">

<Comment FormalName="Headline">"Medyczna ośmiornica" z Wałbrzycha załatwiała renty za łapówki</Comment>

<Comment FormalName="Dateline">27/9/2000</Comment>

<Comment FormalName="MediaType">Text</Comment>

</AssociatedWith>

</NewsManagement>

</NewsComponent Essential="no" EquivalentsList="no">

<TopicSet FormalName="NewsSubjects">

<Topic>

<TopicType FormalName="Subject" Scheme="IptcTopicType"/>

<Description xml:lang="en" Variant="Category">ACE</Description>

<Description xml:lang="pl" Variant="Name">Sztuka, Kultura, Rozrywka</Description>

<Description xml:lang="en" Variant="Name">arts, culture and entertainment</Description>

</Topic>

</TopicSet>

<TopicSet FormalName="Events">

</TopicSet>

<Topic Duid="PAEvents:337">

Version 1.0: Final version 2014-01-29
<TopicOccurrence Topic="#PAEvents:337" HowPresent="Prominent"/>

<Property FormalName="Country" Value="ALB"/>
<Property FormalName="City" Value="Tirana"/>
<Property FormalName="Region" Value="sdfsdfsdf"/>
</DescriptiveMetadata>

<NewsComponent Essential="no" EquivalentsList="no">
<Role FormalName="Main"/>
<ContentItem>
<MediaType FormalName="Text"/>
<Format FormalName="HTML"/>
<Characteristics>
<Property FormalName="WordCount" Value="0"/>
</Characteristics>
<DataContent><![CDATA[]]></DataContent>
</ContentItem>
</NewsComponent>
</NewsItem>
</NewsML>
Contributing partners

Trine M. Seeberg
Trine.Seeberg@sintef.no

Franck Fleurey
Franck.Fleurey@sintef.no

Olivier Barais
barais@irisa.fr

Vladimir Cvetkovic
vlad@ozoneroam.com

Eamonn Roarke
eamonn.roarke@ozoneroam.com

Knut Eilif Husa
Knut.Eilif.Husa@tellu.no

Walter Waterfeld
walter.waterfeld@softwareag.com

Udo Hafermann
Udo.Hafermann@softwareag.com

Ilias Spais
i.spais@atc.gr