Driving Innovation in Crisis Management for European Resilience

D934.31 – DRIVER+ SOLUTION SCENARIOS AND INTEGRATION TEST RESULTS V1

SP93 - SOLUTIONS

NOVEMBER 2018 (M55)

This project has received funding from the European Union's 7th Framework Programme for Research, Technological Development and Demonstration under Grant Agreement (GA) N° #607798

Project information

Project Acronym:	DRIVER+
Project Full Title:	Driving Innovation in Crisis Management for European Resilience
Grant Agreement:	607798
Project Duration:	72 months (May 2014 - April 2020)
Project Technical Coordinator:	TNO
Contact:	coordination@projectdriver.eu

Deliverable information

Deliverable Status:	Final
Deliverable Title:	D934.31 – DRIVER+ Solution scenarios and integration test results v1
Deliverable Nature:	Report (R)
Dissemination Level:	Restricted (RE)
Due Date:	November 2018 (M55)
Submission Date:	11/12/2018
Subproject (SP):	SP93 - Solutions
Work Package (WP):	WP934 - DRIVER+ CM Solutions
Deliverable Leader:	FRQ
Reviewers:	Erik Vullings, TNO Denis Havlik, AIT Maurice Sammels, XVR
File Name:	D934.31_DRIVER+ Solution scenarios and integration test results v1.docx
Version of template used:	V2.0 – June 2018

Revision Table

Issue	Date	Comment	Author
V0.01	16/04/2018	Initial TOC and draft content	Joaquin, Marquez-Bugella, FRQ Task Leader
V0.02	16/07/2018	Creation of sections 2.3, 2.4 and 2.5 based on integration test reports from Trial 1 preparations	Ludwig Kastner, FRQ, Task Leader
V0.03	17/07/2018	Updating to the new template and merging content from different versions	Joaquin, Marquez-Bugella, FRQ Task Leader
V0.04	18/07/2018	Reorganizing TOC: including section 5: Trial independent Test-bed integration of internal solutions	Joaquin, Marquez-Bugella, FRQ Task Leader
V0.05	24/07/2018	Inclusion of Trial 2 – Dry Run 1 integration results	Joaquin, Marquez-Bugella, FRQ Task Leader
V0.06	25/07/2018	Adding contribution of Thales to section 4.2.4.3	Joaquin, Marquez-Bugella, FRQ Task Leader Antoine Leger, Thales, Internal solution provider
V0.07	27/07/2018	Adding Contribution of XVR about Test-bed deployment during T2-DR-1 (Section 4.3.4)	Joaquin, Marquez-Bugella, FRQ Task leader Martijn Hendricks, XVR
V0.08	31/07/2018	Adding raw info from FRQ about LifeX COP	Joaquin, Marquez-Bugella, FRQ Task leader Thomas Obritzhauser, FRQ Internal solution provider
V0.09	31/07/2018	Adding annex 2 information and detailing several sections about the Trial 2	Joaquin, Marquez-Bugella, FRQ Task Leader
V0.10	06/08/2018	Adding contribution from Merlin	Joaquin, Marquez-Bugella, FRQ Task Leader Ruud van den Beukel
V0.11	09/08/2018	Apply amendments out of the Hannah Goeritz comment on V0.9	Hannah Goeritz, FRQ, Software Engineer, Joaquin, Marquez-Bugella, FRQ Task Leader
V0.12	09/08/2018	Add content about Trial 1 – Integration of External solution Providers	Joaquin, Marquez-Bugella, FRQ Task Leader
V0.13	09/08/2018	Mark Sections where Hannah Goeritz shall contribute	Joaquin, Marquez-Bugella, FRQ Task Leader
V0.14	22/08/2018	Added Test-bed description	Hannah Goeritz, FRQ
V0.15	06/09/2018	Added solution descriptions for Trial 2 and GMV contribution for Trial 1; restructured the structure for the Trial 1 section	Hannah Goeritz, FRQ
V0.16	18/09/2018	Added texts and figures from Integration test report Dry Run1 of Trial 2, added information to Trial 1 sections	Ludwig Kastner, FRQ

Issue	Date	Comment	Author
V0.17	19/09/2018	added information to Trial 1 sections	Ludwig Kastner, FRQ
V0.18	20/09/2018	added information to Trial 2 sections	Ludwig Kastner, FRQ
V0.19	26/09/2018	added information to section "Trial independent Test-bed integration of internal solutions"	Ludwig Kastner, FRQ
V0.20	28/09/2018	Section 4.4.7. added (this section was later moved to the Test-bed design documentation because it has more relation to the Test-bed than to solution integration)	Cyril Dangerville, TCS
V0.21	03/10/2018	merge of Dry Run 1 and Dry Run 2 sections, adding information relevant for both Dry Runs	Ludwig Kastner, FRQ
V0.22	19/10/2018	Add information from "DR-2 integration", adding adaptations of each solution provider, adding sections "solution integration results"	Ludwig Kastner, FRQ
V0.23	21/10/2018	Formatting according to DRIVER+ template standards	Ludwig Kastner, FRQ
V0.24	28/10/2018	Add information from project partner ITTI about their PROCeed solution to section 6.1	Ludwig Kastner, FRQ
V0.25	08/11/2018	Extend summaries per main section, final formatting	Ludwig Kastner, FRQ
V0.26	09/11/2018	Add all solution descriptions from PoS to Annex 3	Ludwig Kastner, FRQ
V0.27	22/11/2018	Quality check during internal review	Erik Vullings, TNO
V0.28	23/11/2018	Quality check during internal review	Maurice Sammels, XVR
V0.29	26/11/2018	Quality check during internal review	Denis Havlik, AIT
V0.30	27/11/2018	Adaption after internal reviews	Hannah Goeritz, FRQ
V0.31	29/11/2018	Adaption after internal reviews	Ludwig Kastner, FRQ
V0.32	04/12/2018	Final document formatting	Ludwig Kastner, FRQ
V0.33	06/12/2018	Final document check	Ludwig Kastner, FRQ
V0.34	07/12/2018	Final check and approval for submission	Tim Stelkens-Kobsch, DLR, Quality Manager
V0.35	07/12/2018	Final check	Marijn Rijken, TNO, Project Director
V0.36	10/12/2018	Amending Executive Summary and chapter 5 according to remarks from final check	Ludwig Kastner, FRQ
V1.0	11/12/2018	Final check and submission to the EC	Francisco Gala, ATOS

The DRIVER+ project

Current and future challenges, due to increasingly severe consequences of natural disasters and terrorist threats, require the development and uptake of innovative solutions that are addressing the operational needs of practitioners dealing with Crisis Management. DRIVER+ (Driving Innovation in Crisis Management for European Resilience) is a FP7 Crisis Management demonstration project aiming at improving the way capability development and innovation management is tackled. DRIVER+ has three main objectives:

1. Develop a pan-European Test-bed for Crisis Management capability development:
    a. Develop a common guidance methodology and tool, supporting Trials and the gathering of lessons learnt.
    b. Develop an infrastructure to create relevant environments, for enabling the Trialling of new Solutions and to explore and share Crisis Management capabilities.
    c. Run Trials in order to assess the value of Solutions addressing specific needs using guidance and infrastructure.
    d. Ensure the sustainability of the pan-European Test-bed.
2. Develop a well-balanced comprehensive Portfolio of Crisis Management Solutions:
    a. Facilitate the usage of the Portfolio of Solutions.
    b. Ensure the sustainability of the Portfolio of Solutions.
3. Facilitate a shared understanding of Crisis Management across Europe:
    a. Establish a common background.
    b. Cooperate with external partners in joint Trials.
    c. Disseminate project results.

In order to achieve these objectives, five Subprojects (SPs) have been established. **SP91 *Project Management*** is devoted to consortium level project management, and it is also in charge of the alignment of DRIVER+ with external initiatives on crisis management for the benefit of DRIVER+ and its stakeholders. In DRIVER+, all activities related to Societal Impact Assessment are part of **SP91** as well. **SP92 *Test-bed*** will deliver a guidance methodology and guidance tool supporting the design, conduct and analysis of Trials and will develop a reference implementation of the Test-bed. It will also create the scenario simulation capability to support execution of the Trials. **SP93 *Solutions*** will deliver the Portfolio of Solutions which is a database driven web site that documents all the available DRIVER+ Solutions, as well as Solutions from external organisations. Adapting solutions to fit the needs addressed in Trials will be done in **SP93**. **SP94 *Trials*** will organize four series of Trials as well as the final demo. **SP95 *Impact, Engagement and Sustainability***, is in charge of communication and dissemination, and also addresses issues related to improving sustainability, market aspects of Solutions, and standardization.

The DRIVER+ Trials and the Final Demonstration will benefit from the DRIVER+ Test-bed, providing the technological infrastructure, the necessary supporting methodology and adequate support tools to prepare, conduct and evaluate the Trials. All results from the Trials will be stored and made available in the Portfolio of Solutions, being a central platform to present innovative Solutions from consortium partners and third parties, and to share experiences and best practices with respect to their application. In order to enhance the current European cooperation framework within the Crisis Management domain and to facilitate a shared understanding of Crisis Management across Europe, DRIVER+ will carry out a wide range of activities. Most important will be to build and structure a dedicated Community of Practice in Crisis Management, thereby connecting and fostering the exchange of lessons learnt and best practices between Crisis Management practitioners as well as technological Solution providers.

Executive summary

This document describes the main activities related to the integration of solutions into the Test-bed, focusing on the adaptations and integration activities which were required to prepare the solutions for Trial 1 and Trial 2. Deliverable **D934.32 *DRIVER+ Solution scenarios and integration test results v2*** (1) (due M65) will report about the corresponding activities performed for Trial 3 and Trial 4.

User needs not fulfilled by legacy systems are described in the DRIVER+ project in form of "Crisis Management gaps". These gaps are the starting point for the creation of the Trial scenarios which form the basis for the underlying test cases. The technical effort to design, prepare and finally perform these test cases is reported in this document and is closely related to the overall target to fulfil end user needs. The final achievements for Trial 1 and Trial 2 regarding how the selected solutions and the solution integration could fill the gaps are described in the Trial Evaluation reports (2) and (3).

Integrated IT solutions are of fundamental importance for the efficiency and effectiveness of modern Crisis Management. An automated exchange of data between different IT solutions aims to achieve the following:

- Less time needed for practitioners in their search for crisis relevant information.
- Less time needed for practitioners to read data from one solution and entering data manually into another solution.
- Lower probability for wrong information caused by human errors while reading/entering data from/into a solution.
- More time left for practitioners to analyse and interpret the information, and to define, communicate, execute and supervise crisis response actions.
- Higher quality of the crisis management outcome due to the time savings, better data quality and improvement of communication.

The document gives a brief overview of the Test-bed and summarizes the preparation activities which were required to adapt and integrate the solutions for Trial 1 and Trial 2 from a technical perspective.

The challenges for Trial 1 which were mainly caused by the early timing of this Trial within the project are described in this document. Those were technical procedures which were still in the definition phase and a Test-bed which was still under development. Other challenges resulted from the involvement of external solution providers in such a complex project and the limited experience of the consortium to introduce external solution providers properly to the overall idea of the project and especially to the complex setup of the Trials. The document describes the scenarios for Trial 1 and Trial 2 as well as the selected solutions and the adaptations, integration and testing which were performed to support the Trial scenarios. Each Trial had two preceding Dry-Runs which turned out to be absolutely mandatory in order to prepare the Trials properly, both from a technical and an organizational perspective.

At the end of the document the "Trial independent Test-bed integration" of internal solutions is described. Even though the integration of the solutions in the first year was mainly driven by the first two Trials it was decided that for all remaining DRIVER+ internal solutions (not selected for Trials) a plan shall be created for their integration steps into the Test-bed. For this category of solutions additional "solution scenarios" will be introduced. These solution scenarios and related test cases will be necessary in order to integrate those solutions into the Test-bed. Having all DRIVER+ internal solutions integrated in the Test-bed would provide advantages such as a more comprehensive Test-bed and shorter timespans for the integration work for future Trials.

The conclusions of the document finally summarize the achievements and provide detailed insights into the lessons learnt. In a project of this dimension with lots of actors involved smaller delays in one process step often have a big impact on many other project partners and their dependent consecutive processes. Given the prearranged dates for Trials the timespan in between Dry Runs and for the Solution integration work has often been very demanding. With the lessons learnt from Trial 1 and Trial 2 it can be expected that corrective actions will lead to improvements for the preparation phases of Trial 3 and Trial 4.

Table of Content

List of Figures

List of Tables

List of Acronyms

Acronym	Definition
AMP	Advanced Medical Post
CAP	Common Alerting Protocol, an XML-based data format for exchanging public warnings and emergencies between alerting technologies
CfA	Call for Application
CODIS	Centre Opérationnel Départemental d'Incendie et de Secours (Departmental Operational Center of Fire and Rescue)
COGIC	Centre Opérationnel de Gestion Interministérielle des Crises (Interdepartmental Operations Center for Crisis Management)
COZ	Centre Opérationnel de Zone (Area Operations Center)
DB	Database
DCP	Data Collection Plan
DREAL	Direction Régionale, de l'Environnement de l'Aménagement et du Logement (Regional Department of the Environment, Planning and Housing)
DR-1	Dry Run 1
DR-2	Dry Run 2
EcASC Valabre	École d'Application de Sécurité Civile (Civil Protection trainings site)
EHS HQ	Emergency Health Service Head Quarters
EMS	Emergency Medical Services
EMS-HQ	Emergency Medical Services Head Quarters
EMSI	Emergency Management Shared Information
ERCC	Emergency Response Coordination Centre
EUCPM	European Civil Protection Mechanism
FCP	Field Command Post
FCP-A	Field Command Post – Anticipation
FCP-IC	Field Command Post – Incident Commander
FTP	File Transfer Protocol
HQ	Head Quarters
IC	Incident Commander
JSON	JavaScript Object Notation
KML/KMZ	Keyhole Markup Language

Acronym	Definition
MQ	Message Queue
OPCW	Organization for Prohibition of Chemical Weapon
PMA / AMP	Advanced Medical Post
POK	Partly O.K.
REST	Representational State Transfer
SiTac	Situation Tactique (tactical situation)
SitRep	Situational Report
TGM	Trial Guidance Methodology
TA	Test Activity
TIFF	Tagged Image File Format
TP	Transit Point
UI	User Interface
UML	Unified Modeling Language
URL	Uniform Resource Locator
VM	Virtual Machine
WFS	Web Feature Service
WMS	Web Map Service

1. Introduction

One of the main and implicit objectives of the DRIVER+ project is sustainability. The experiences gained during the Trial preparations and executions are precious information which shall be documented herewith. This document covers the purpose mainly at the technical level of solution integration. The work and results of the solution integration performed for the first two Trials are described, which took place in 2018 at the Szkoła Główna Służby Pożarniczej (SGSP) in Warsaw and at the École d'Application de Sécurité Civile (EcASC) Valabre in Aix en Provence and gives a short overview of the DRIVER+ scenarios for these Trials. In addition the integration of internal solutions which were not yet selected for Trial 1 or Trial 2 is covered in this document. Deliverable **D934.32** (1), which is due in M65, will be an update of the current document and will contain the integration activities which correspond to Trial 3 and Trial 4.

1.1 Identification of intended audience

The intended audiences of this document are:

- Future Trial owners and external users of the Test-bed who want to review past Trials to prepare future Trials.
- Solution providers who aim at integrating their solution to the Test-bed and who want to learn from the integration efforts performed for Trial 1 and Trial 2.

1.2 Scope of the document

The scope of this document is to describe the integration of solutions into the Test-bed as well as the test case scenarios needed to assess and evaluate the integration. The main scope of the document covers Trial-specific integrations but also the integration of solutions that were not part of Trial 1 or Trial 2 is covered in this document in a separate chapter.

Thus, the document shall give insight which technical efforts are required for preparing, testing and running a Trial, and what typically must be done for the integration of solutions into the Test-bed.

1.3 Document structure

The document starts with a short overview of the Test-bed and its components and continues in a chronological order by describing Trial 1 and 2 with all their preparation steps, especially their Dry Run 1 (DR-1) and Dry Run 2 (DR-2) events as they are most relevant for the solution integration work. Work relevant for future Trials is described in section “Trial independent Test-bed integration” which relates to internal solutions which were not yet selected for Trial 1 and Trial 2. Annex 1 lists the DRIVER+ terminology, Annex 2 and 3 document the Test-bed messages for different steps of the integration, Annex 4 and 5 provide details of the testing activities and test scenarios of both Trials. Annex 6 documents the IP addresses and technical users for both Trials; Annex 7 provides the contact details of the solution providers involved in the Trials. Finally Annex 8 provides a description of all solutions which are available in the PoS database at the time of submission of this document (this part is directly exported from the PoS as defined in the DRIVER+ Description of Work).

2. Test-bed

The Test-bed is an important part of the DRIVER+ project and is therefore dealt within its own sub-project, namely **SP92 *Test-bed***. In the document at hand, only a brief overview is given of the Test-bed, its purpose and its design as there are several dedicated Test-bed deliverables[1]. Further, the steps that should be performed in order to use the Test-bed, i.e. the integration to the Test-bed, are described briefly.

In DRIVER+ the Test-bed provides the necessary infrastructure to prepare, execute and manage Trials and in this context, evaluate the solutions which participate in those Trials. It is designed following a modular approach in the sense that several tools and services fulfilling a certain purpose each are connected to build-up the Test-bed. In this way, the Test-bed can be progressively extended during the course of DRIVER+. Also, the modular design should make the task of sustaining the Test-bed beyond the project´s lifetime an easier one. As one of the Test-bed's main purposes is providing the infrastructure for connected systems to exchange information, the architecture of the Test-bed is message-based using the open-source messaging system Apache Kafka. Further, the Test-bed is intended to be deployed using composed Docker images, i.e. ready-to-use installer applications in order to facilitate an easy deployment, see https://github.com/DRIVER-EU/ .

The next paragraph gives an overview the components of the Test-bed. A more detailed description of the Test-bed can be found in **D923.11** regarding the ***Test-bed specification*** (2) as well as in **D923.21** which describes the design of the ***Test-bed reference implementation*** (3). Afterwards a concise summary of the integration process to the Test-bed is provided as part of the ***Solution Testing Procedure*** which is specified in **D934.21** (4).

2.1 Brief description of the Test-bed components

As already mentioned, the Test-bed consists of several modular tools and services. Also, during the course of DRIVER+ the reference implementation of the Test-bed will be constantly improved. Therefore, several versions will be released. As a consequence, not all of the components described here will be part of Trial 1 and Trial 2.

Figure 2.1 gives an overview of the architecture of the Test-bed and its components. The main components are specified in Table 2.1, while some additional components are described in Table 2.2.

In addition to the components (Table 2.1 and Table 2.2) there are adapters specified and implemented in the reference implementation. Those adapters serve the purpose of data transfer, i.e. message exchange between the solutions and the Common Information Space (CIS adapters) and the simulations and the Common Simulation Space (CSS adapters), respectively. These adapters are also shown in Figure 2.1. Several types of adapters are currently available to provide easy integration of solutions and simulations. Those include a REST[2] adapter, TypeScript[3] adapter, a Java adapter, a C# adapter as well as a.

[1] Test-bed functional specification D923.11 (20) and Test-bed reference implementation D923.21 (21)

[2] Representational State Transfer (REST) is a web service for sharing information.

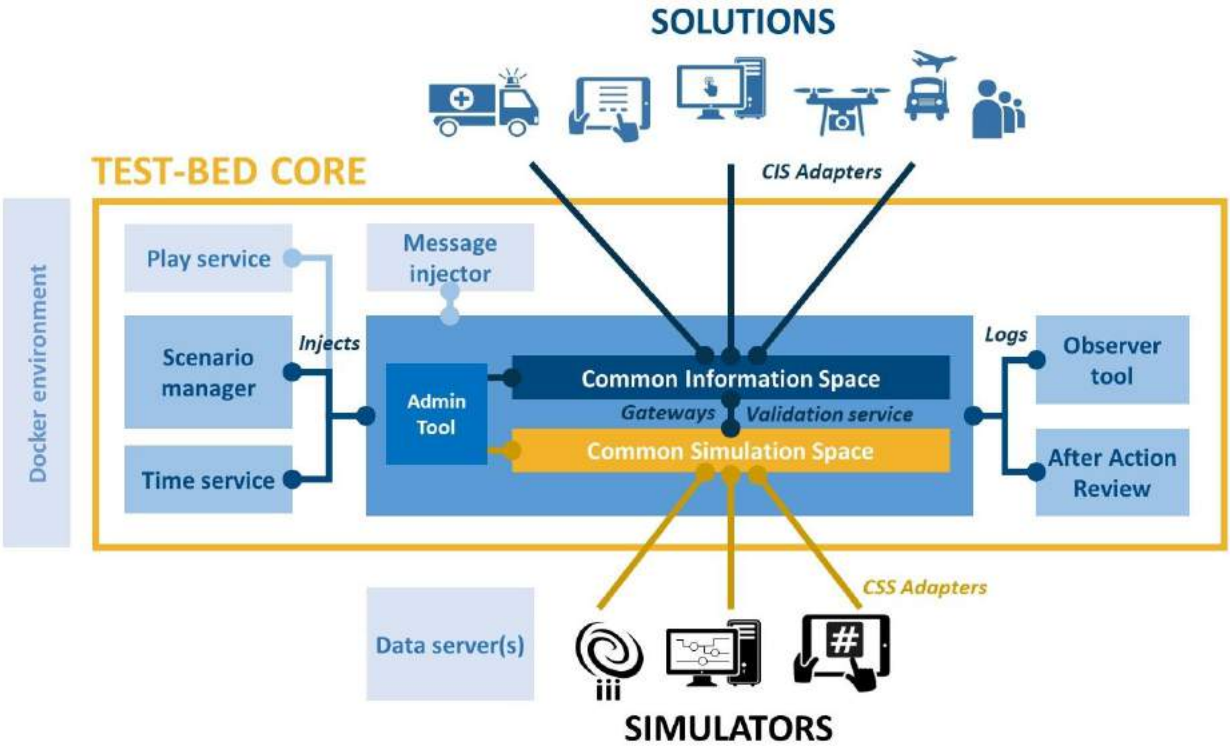


Figure 2.1: The Test-bed and its components

Table 2.1: Main components of the Test-bed

Component	Short description	Responsible
Common Information Space (CIS)	A central messaging bus with standardized CIS adapters to connect solutions to the Test-bed.	FRQ
Common Simulation Space (CSS)	A central messaging bus with standardized CSS adapters to connect simulators to the Test-bed.	TNO, XVR
Gateways	Links between the CSS and the CIS to feed solutions with data from the Simulations and vice versa.	TNO
Validation service	A service that validates the messages sent between CSS and CIS. (optional)	TNO
Test-bed admin tool	A tool providing a user interface to control the CSS and CIS before and during a Trial.	FRQ
Trial scenario manager	A tool to create and manage the timeline of a Trial scenario. Also, messages might be prepared before the Trial and injected during Trial execution.	TNO
Time Service	A service that controls the fictive time during a Trial.	TNO

[3] TypeScript is an open-source programming language developed and maintained by Microsoft. It is a strict syntactical superset of JavaScript.

Component	Short description	Responsible
Observer Support Tool (OST)	With the OST observations can be gathered during a Trial.	TNO, ITTI
After Action Review (AAR)	The AAR module uses data-logs and observations to review the Trial after it was executed.	FRQ

Table 2.2: Additional components of the Test-bed

Component	Short description	Responsible
Message injector	A service to input specific, single messages to test whether these messages are correctly received by another component.	FRQ
Play service	A service to input several messages in chronological order.	TNO
Data server(s)	Components used to feed the simulators and potentially also the solutions with geo-specific information and map-layers.	TNO, iTTi
Docker environment	Using Docker installers of the Test-bed components can be provided in order to facilitate a quick and easy installation.	TNO

2.2 Test-bed integration steps summary

The Test-bed integration steps have been defined in deliverable **D934.21 -** ***Solution Testing Procedure*** (4). The goal of this procedure is to make sure that the solutions and the technical set-up are ready at the end of Dry Run 1 to support the Trial execution.

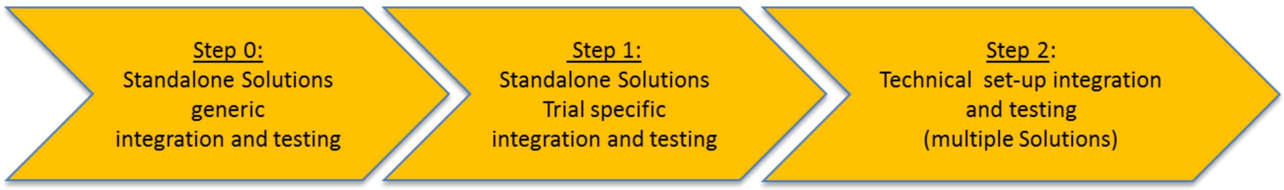


Figure 2.2: Major steps of the solution integration and testing procedure

This procedure is purely technical, it does not address the actual assessment of the solutions which will be performed during the Trial, but it enables it. The whole procedure is under the joint responsibility of the solution coordinator and the Test-bed infrastructure coordinator, who work in coordination with the other members of the Trial Committee.

The Test-bed integration is designed to be supportive, descriptive in nature. It gives recommendations to the solution coordinator who will be able to customise it in agreement with the Trial Committee to consider her/his own familiar methods, the specific constraints of the Trial, the time schedule, the nature of the technical set-up, or the participating solutions themselves. It consists of the following three steps:

- **Step 0:** *Standalone solutions – generic integration and testing*: During this step, the solution owner defines the main user stories of his/her solution, translates them to test cases, and integrates the solution to the Test-bed reference implementation. This integration is tested against the solution test cases, which define the required inputs from the Test-bed, and the expected outputs of the solution. This step is not Trial specific and is performed by each solution owner and under the coordination of **SP93** task leaders as a generic preparation for any Trial.

- **Step 1:** *Standalone solutions – Trial specific integration and testing*: during this step, individual solutions are adapted and integrated to fulfil the Trial specific requirements. Solution owners contribute to the writing of the Trial specific requirements (defined by step 2) and write the corresponding test cases for their own solutions. This step is performed by each solution owner under the coordination of solution coordinators (**SP94**), and leaders of corresponding **SP93** tasks.
- **Step 2:** *Technical set-up integration and testing*: this step aims at designing, integrating and testing the Trial's technical set-up as a whole. The technical set-up consists of a set of solutions, the Test-bed, simulators and measuring tools. step 2 is split in two parts: step 2.1 defines the technical set-up, generates the Trial specific requirements and defines one or more test scenarios, and (after step 1 is completed) step 2.2 performs the integration of multiple solutions and tests it against these test scenarios. A test scenario, in this sense, is a scenario that is dedicated to test the main interactions between different solutions as required by the Trial scenario. This step is coordinated jointly by the solution coordinator (**SP94**) and the Test-bed infrastructure coordinator (**SP92**) with the support of solution and simulation owners.

2.3 Integration information and support

The first steps of the solution provider to integrate their solutions into the Test-bed are:

- Understanding the Test-bed concept.
- Understanding the Test-bed adapter options and choosing the right adapter for their solution.
- Defining the messages to be exchanged between the solution and the Test-bed.
- Connecting the adapter to their solution.
- Exchanging messages between solution and Test-bed according to the integration steps described above.

To support this process an integration information package was created and made available under the following link: https://github.com/DRIVER-EU/Test-bed#integration-process.

With this information, solution providers could start their Test-bed integration with a local version of the Test-bed and try to connect their solution to one of the available Test-bed adapters.

As all technical support questions and answers related to Test-bed integration were assumed to be of interest for all solution providers, a communication channel was established in form of an online forum with the online communication tool SLACK under the following link:
https://driver-eu.slack.com/messages/C6YQK3FUJ/.

The questions discussed in this integration forum were mainly about the following topics:

- Clarifications how to connect to the Test-bed.
- Issues with the local version and the online version of the Test-bed.
- Exchanging information for the first time with the Test-bed.
    - Problems with the Test-bed adapters.
    - Questions about AVRO schemas.

    - Questions about CAP[4] messages.
- Updates of Test-bed adapters.
- Problems with firewalls.
- Questions about file size limitations.

[4] CAP=Common Alerting Protocol, an XML-based data format for exchanging public warnings and emergencies between alerting technologies

3. Trial 1

This section describes the main activities related to the integration of solutions for Trial 1. There were several challenges faced during Trial 1, including that:

- Technical procedures were not finalized, but still in the definition phase.
- The Test-bed was under development in an initial phase.

Another challenge resulted from the involvement of the external solution providers and the limited time and experience of the consortium to introduce external solution providers properly to the overall idea of the project and especially to the complex setup of Trial 1 resulting in an insufficient education of external solution providers.

Before Trial 1 there were two Dry Runs (named DR-1 and DR-2) for the preparation of all technical and organizational matters. Trial 1 was mainly focussing on cross-border tasking and resource management. The Trial demonstrated the potential of a more integrated high-level Crisis Management (CM) system in Europe, especially in cross-border contexts in terms of improved situation assessment, coordination, resource pooling & sharing, and cross border cooperation. The Trial itself also served as a demonstration of the potential of a Common Operational Picture (COP) approach on a European level, and to help evolving the DRIVER+ methodology and the Test-bed for the Trial preparation process.

3.1 Exceptional circumstances and consequences

Trial 1 had some exceptional circumstances which were caused by the early timing of Trial 1:

- The process for involving external solution providers to Trials was not well established which lead to long negotiations with them and a delay in establishing agreements with them. The late joining of external solution providers left very little time for the integration of their solutions.
- Deliverable **D934.21** (4) which defines the solution testing procedure was in draft status only and not approved by the time the Trial 1 preparations started.
- The Test-bed was still under development and had limited functionality, e.g. regarding Web Map Service (WMS)[5] and Web Feature Service (WFS)[6] services, at that point in time leading to some limitations for the solution integration for Trial 1.

Considering these limitations Trial 1 set an initial baseline for the rest of the Trials which will face better prerequisites in these three aspects.

3.2 Scenario description of Trial 1

The Trial 1 scenario consisted of a massive release of liquid toxic substances because of a maintenance failure in a reservoir which collects chemical wastes:

"A valve failure causes pumps that are pumping chemical liquid waste to the reservoir to not switch off. Due to this, there is a rapid inflow of a significant amount of a liquid, mud like toxic chemical to the

[5] Web Map Service (WMS) is a standard protocol developed by the Open Geospatial Consortium for serving georeferenced map images over the Internet.

[6] Web Feature Service (WFS) provides an interface allowing requests for geographical features across the web (14; 13).

retention reservoir. Dikes of the reservoir are weakened after prolonged rainfall during the past few days and under the influence of pressure the dikes break.

The broken control station prevents a quick intervention and the increased pressure causes approximately 700 000 cubic meters of toxic, mud like, fluid as a massive 1-2-meter-high wave that floods nearby localities in a matter of minutes. In the path of the spill are several villages and towns, where initially 15 people died, and 200 people suffer severe toxic injuries. The eventual 30 square kilometres of affected land include a river that crosses the border with a neighbouring country. There is a high risk of contamination for the river water. The river is a source of water intake for various industries, agriculture and fresh water companies. Consequential impact could result in destroyed crops, toxic injuries to livestock and a disturbance in the water supply causing immediate water shortage. The incident requires deployment of different kind of crises management and civil protection assets which are able to realize such functions as medical rescue, search and rescue, evacuation, decontamination, water purification, flood containment, temporary sheltering, etc."

The scenario was based on the disasters which took place in Romania in 2000 (Baia Mare cyanide spill) and in Hungary in 2010 (Ajka alumina sludge spill), and it was consulted with:

- The Emergency Response Coordination Centre (ERCC).
- Representatives from the Joint Environment Unit (JEU UNEP/OCHA).
- The Organization for Prohibition of Chemical Weapon (OPCW).
- Former experts and colleagues who took part in the response for Ajka alumina sludge spill as the European Union Civil Protection Team (EUCPT).
- National Hungarian experts.

For more details about the scenario of Trial 1 please see **D943.11 *Report on Trial Action Plan*** – Trial1 (5).

3.3 CM gaps to be covered in Trial 1

According to DRIVER+ methodology, the corresponding gap analysis was carried out for Trial 1. This analysis was intended to reveal areas that can be improved in the existing practices, processes and daily operations related to Crisis Management in the context described by the Trial scenario.

The gap analysis was conducted by the Trial Owner (SGSP) with the collaboration of their practitioners network. The resulting list of gaps was presented during the DRIVER+ Gap Assessment Workshop where multiple stakeholders and practitioners (over thirty parties) were invited to discuss and assess the relevance of the presented gaps for them. After the received feedback, it was decided to focus on the following gaps listed in Table 3.1

Table 3.1: Crisis Management gaps to be covered in Trial 1

CM Gap No.	Description
T1.1	Limitations in the ability to model real-time (response phase) or pre-event (preparedness phase) dynamics of the chemical and radiological threat and visualisation of obtained results in a form that can be used directly by the Head of the Rescue Operations.
T1.2	Lack of a Common Operational Picture (COP) environment to integrate data sources and calculation results from different models crucial for decision making process from the perspective of Head of Rescue Operation.
T1.3	Limitations in the cross vulnerabilities (people, property, environment) assessment to optimize task prioritization and decision making.
T1.4	Insufficiencies in terms of resource management (human resources, hardware, etc.) during multi-stakeholder long-term rescue operations.

CM Gap No.	Description
T1.5	Lack of effective public warning system with the ability to verify whether the information reached the recipient.

These CM gaps drove the solution selection process, which aimed to select the solutions best suited to close the gaps.

3.4 Participating solutions and their integration into the Test-bed

Three solutions participated in Trial 1, one solution from a DRIVER+ partner and two external solutions:

- Socrates OC is a web-based tool for generating a COP in Crisis Management; it enables the exchange of information amongst nodes as well as doing tasking and resource management.
- 3Di is an interactive water simulation model which enables flood forecasting and exploring various future scenarios in a very short time frame (minutes).
- Drone Rapid Mapping (DRM) solution enables fast generation of orthophoto maps based on imagery acquired by a remotely piloted aircraft system which is available to Crisis Management actors.

These solutions were intended to achieve a certain level of automated data exchange to have practitioners take advantage of an integrated "system of systems" which combined some of the benefits brought by each solution. The integration of solutions consisted of Socrates OC ingesting data generated by DRM and 3Di. Concretely, the latter ones provided georeferenced image files which were displayed as map layers in Socrates OC. From a technical perspective, this was found to be the soundest approach; from the operational perspective, it was expected to allow high-level decision makers to only focus on the COP provided by Socrates OC, including some of the 3Di and DRM's products (as map layers) combined with other operational information about events, missions and resources.

3.4.1 Socrates OC – GMV

Socrates OC, provided by the DRIVER+ partner GMV, is a solution aimed to support the command & control and decision-making processes by enabling COP-based sharing of situational awareness and providing features for tasking and resource management in crisis scenarios. The Socrates OC solution permits setting up a network of nodes spread amongst different operation centres belonging to organizations involved in crisis response (cf. Figure 3.1). This enables the exchange of relevant information about the operational situation which can be geolocated in a COP supported by a Geographic Information System (GIS).

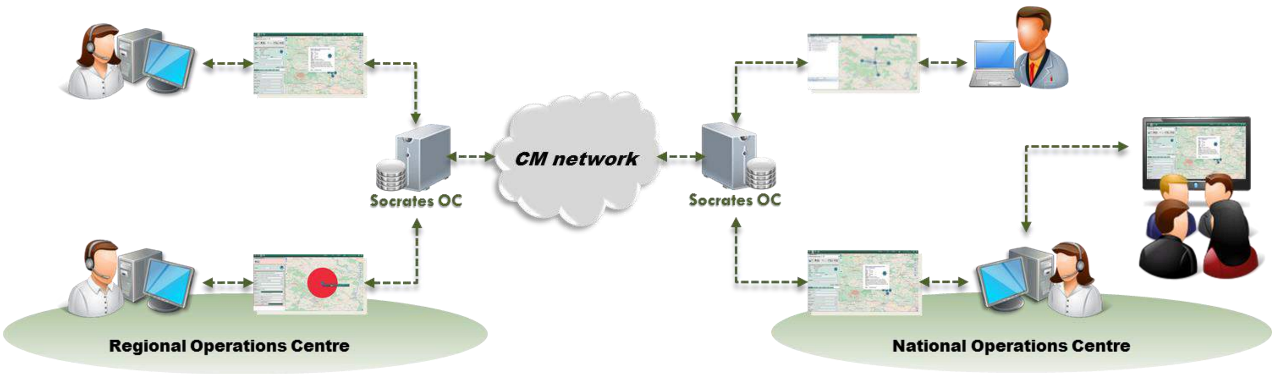


Figure 3.1: Example of a Socrates CM network

3.4.1.1 Integration into the Test-bed

The integration of the Socrates OC solution into the Test-bed concerns two main purposes:

1. Receiving simulated resources from the Test-bed and displaying them in Socrates OC.
2. Receiving map layer update notifications to inform the Socrates OC operator about the availability of a new map layer which can be retrieved and displayed in the solution.

Receiving simulated resources from the Test-bed had several implications. The first and obvious one was the need for developing a dedicated Test-bed adapter (which built on the generic Java Test-bed adapter developed by the project partner FRQ). It listened to XVR's[7] GeoJSON[8] messages distributed by the Test-bed, processed them and provided Socrates OC with the corresponding resource updates (cf. Figure 3.2).

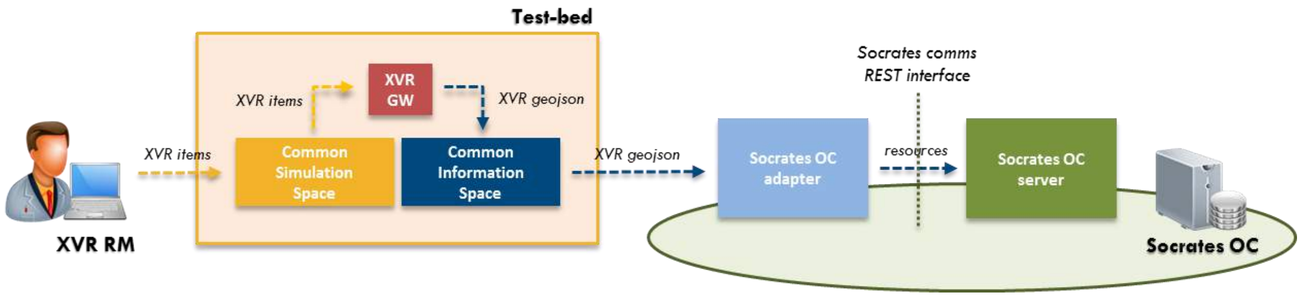


Figure 3.2: Components involved in the reception of resource updates by Socrates OC

According to the scenario of Trial 1, resource updates could come from two different sources, the Test-bed (which provides the name of the resource, its type and position updates) and the operator of the solution (who might introduce additional data such as a textual description of the resource or the modification of its status like "available" etc.). Therefore, it might be the case that the information introduced by the operator was overwritten when a new update of the resource came from the Test-bed. To avoid this, it was required to develop an extra feature for managing partial updates in Socrates OC, which was not originally implemented in Socrates OC.

Additionally, during Dry Run 2 of Trial 1, a performance issue was detected in Socrates OC when it had to deal with the reception of hundreds of resources at once. The performance issue resulted from the fact that Socrates OC uses an elaborate data model for its objects, including versioning, support for dynamic attributes, cross-references, etc., and relies on a complex data base (DB) model which is not intended to handle high data rate updates. In operational contexts requiring high-rate updates (e.g. in other projects for the reception of AIS traces), different custom approaches have been followed (e.g. using a dedicated lighter data model for the affected objects or avoiding real-time DB storage).

The second objective of the integration of Socrates OC into the Test-bed was the reception of the map layer update notifications. These map layer update notifications were intended to inform the Socrates OC operator when a new map layer was made available at some WMS/WFS-compliant server, so the layer could be retrieved and displayed by the solution.

---

[7] XVR Simulation is a DRIVER+ project partner responsible for simulation in the trials

[8] GeoJSON is an open standard format designed by DRIVER+ partner XVR for representing simple geographical features, together with their non-spatial attributes.

During DR-1, it was agreed that the operator of these solutions would supervise the uploading of the corresponding GeoTIFF files (digital elevation, flood arrival times and flood water depth models in the case of 3Di and orthophotos in the case of DRM) to a File Transfer Protocol (FTP) server and send a so-called large data update notification message through the Test-bed. This notification (which included the name and description of the file and a link) had to be received by the operator of the map layer server, who would download the file from the FTP, process it accordingly and finally publish the resulting shapefile to a GeoServer[9] instance included in the map layer server.

To handle these notifications, a web application connected to the Test-bed (and also based on the Java Test-bed adapter) was developed to listen to the large data update notifications and generate the corresponding map layer update notifications once the map layers had been published. The message exchange schema is depicted in the following Figure 3.3.

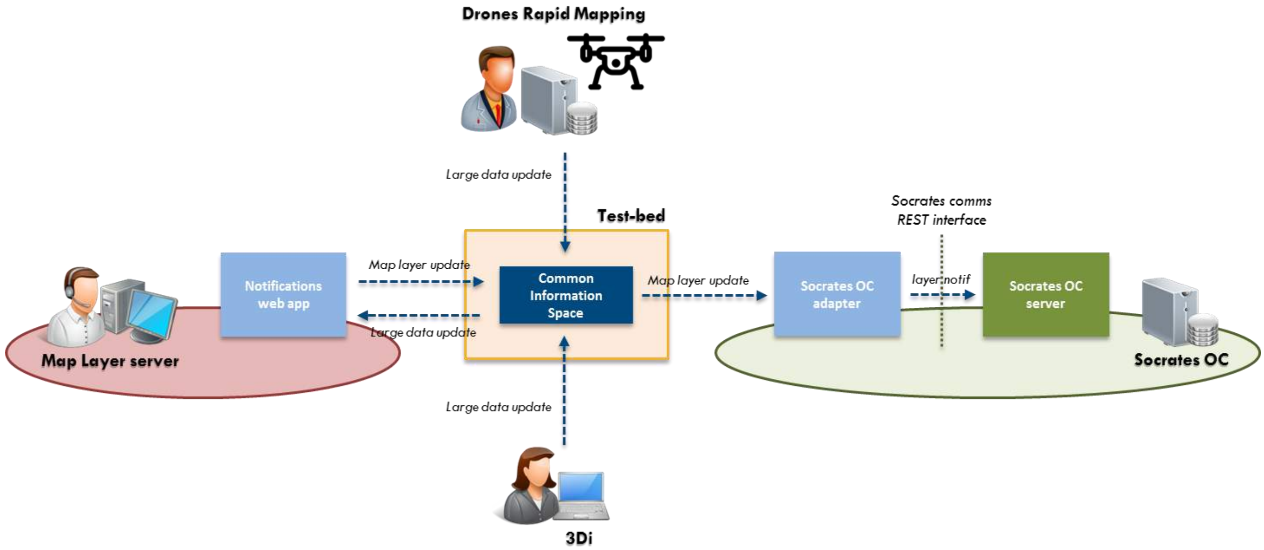


Figure 3.3: Components involved in the reception of map layer updates by Socrates OC

The processing of GeoTIFF files consisted of the following:

- The DRM orthophotos were published as shapefile in the GeoServer.
- The 3Di GeoTIFF, it was required to change the original reference coordinates to the EPSG:4326 [10] system used by Socrates OC and colouring of the original files using QGIS[11], to improve their interpretability by the Socrates OC operator.

Once the map layer was published and the map layer server operator has generated the corresponding map layer update message (including the name and description of the layer and the URL (Uniform Resource Locator) of the GeoServer), this message has been captured by the Socrates adapter which forwarded it to Socrates OC. The Socrates OC operator was then able to retrieve and display the map layer in Socrates OC according to the information provided within the notification.

[9] GeoServer is an open source server for sharing geospatial data (8).

[10] EPSG:4326 is a geodetic coordinate system.

[11] QGIS is a geographic information system (GIS) application that supports viewing, editing, and analysis of geospatial data.

Summing up, the integration of Socrates OC into the Test-bed required the following functionalities, spread between the Notifications web app, the Socrates adapter and the Socrates OC solution itself.

Table 3.2: Required functionality for the integration of Socrates OC solution into the Test-bed

Component	ID	Required functionality	Since
Socrates OC	SOC#01	Partial update of resources.	Workshop "0"
	SOC#02	Receptions of map layer update notifications.	Workshop "0"
	SOC#03	Display of key legend for the interpretation of the colour codes associated to the flood arrival times and flood water depth map layers.	DR-1
Socrates adapter	SOC#04	Reception of XVR's GeoJSON messages and gathering of the resource info inside them.	Workshop "0"
	SOC#05	Creation of Socrates resources update messages based on the info inside XVR GeoJSON messages and transmission of them to the corresponding Socrates OC node.	Workshop "0"
	SOC#06	Reception of map layer update messages and generation of the associated notification to the corresponding Socrates OC node.	Workshop "0"
	SOC#07	Identification of the owner (owning node) of each resource based on its geographical location (provided by the simulation).	DR-2
	SOC#08	Management of the Test-bed messages update rate to avoid performance issues in Socrates OC.	DR-2
Notifications web app	SOC#09	Reception and display of large data update notifications.	DR-1
	SOC#10	Transmissions of map layer update notifications.	DR-1

3.4.1.2 Integration testing and results

This section provides a description of the integration tests and their results regarding the integration of Socrates OC into the Test-bed. The tests described here were carried out locally at the solution provider premises.

The tests were mainly performed in the time periods between DR-1 and DR-2 and between DR-2 and the Trial execution. During DR-1 some preliminary "connection tests" of the Socrates adapter with the Test-bed were also performed. During DR-2 the functionalities developed and tested in the period between DR-1 and DR-2 were demonstrated on site to the Test-bed team.

The last activity was due to a problem found during Trial 1 execution, located in the Socrates adapter and related to the reception of resources from the Test-bed. Further details about TA#1 – TA#7 are provided in Annex 4.

Table 3.3: Summary: results of the tests related to the integration of Socrates OC into the Test-bed

Period	ID	Testing activities	Results
DR-1	TA#0	"Connection tests" between the Socrates adapter and the Test-bed.	successfully passed before DR-2
DR-1-DR-2	TA#1	Reception of resources from Test-bed and simultaneous update by Socrates OC and the Test-bed.	successfully passed before DR-2
	TA#2	Reception of large data update messages by the Notifications web app.	successfully passed before DR-2
	TA#3	Sending of map layer update message by the Notifications web app, reception of the corresponding notification by Socrates OC and display of the map layer.	successfully passed before DR-2
DR-2	-	On site validation (and demonstration to the Test-bed team) of the functionalities developed and tested in the period between DR-1 and DR-2.	On site validation successful during DR-2
DR-2-Trial 1	TA#4	Identification of the owner of resources reported by the Test-bed according to their geolocation.	successfully passed before Trial 1
	TA#5	Reception of resources from the Test-bed at high rate.	successfully passed before Trial 1
After Trial1	TA#6	Debugging of reception of resources from the Test-bed at high rate.	The cause of the problem was with the Socrates adapter, not with the Test-bed

3.4.2 3Di and Lizard – Nelen-Schuurmans

3Di is an interactive water simulation model which enables flood forecasting (flooding locations, water depths, water arrival times, etc.) by constructing a COP of floods and exploring various future scenarios in a very short time frame (minutes).

Figure 3.4: Screenshot of the 3Di and Lizard – Nelen-Schuurmans solution

The solution has three core innovative aspects:

1. Short computation times, high spatial resolution and accurate prediction of floods.
2. Easy to apply flood-affecting measures and compute their effects.
3. Realistic visualization of model outputs.

This interactive solution enables flood forecasting (prediction of flooding locations, water depths, and water arrival times, among others) by constructing a COP of floods and exploring various future scenarios. Scenarios can be explored, and measures can be tested in a very short time frame (minutes).

3.4.2.1 Integration into the Test-bed

The main outputs of the 3Di solution are timestamped GeoTIFF files which include detailed geographical information about the location of the water during a flood. These GeoTIFF files easily exceed 100s of megabytes and were too big to be exchanged through the Test-bed (now and in the future). Instead, the GeoTIFF files were uploaded to an FTP service, and the Test-bed provided the means to distribute the corresponding notification messages needed for the interaction of solutions. The uploaded GeoTIFF files were subsequently, upon receiving a notification, converted to ESRI shapefiles and served as map layers via GeoServer (elaborated in Section 3.4.1). The resulting architecture is shown in Figure 3.5.

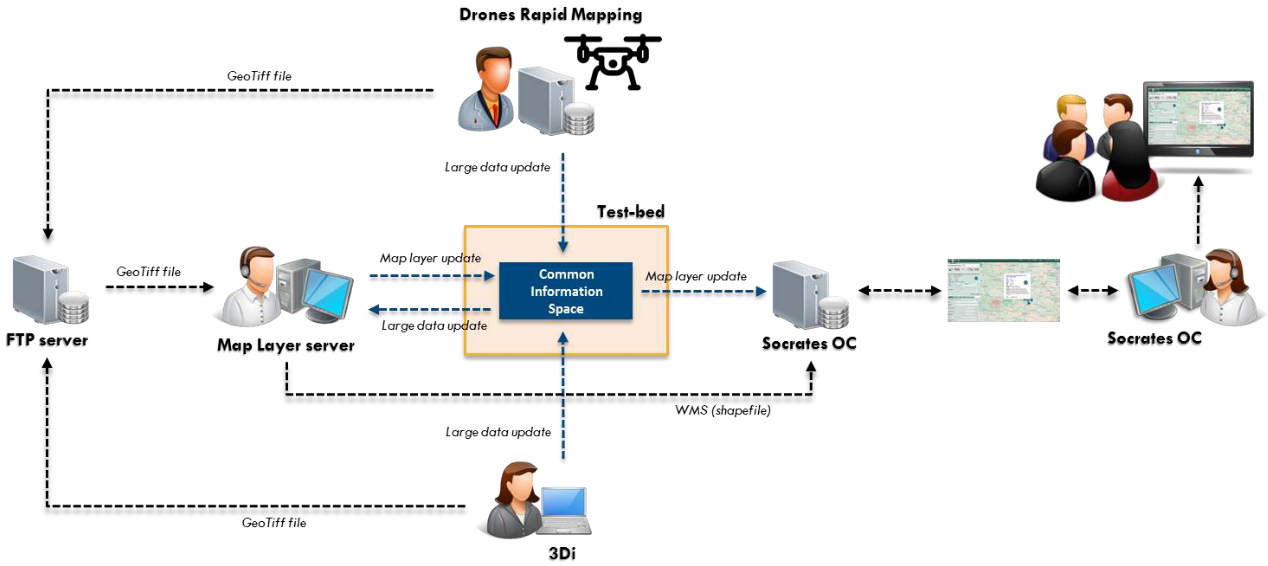


Figure 3.5: 3Di solution integration into the Test-bed

The format and contents of the large data update and the map layer update messages were defined by the Test-bed team and incorporated to the Test-bed's infrastructure as ad-hoc standards for notifications. This way, the Test-bed supervised distributing received messages to those recipients which were subscribed to the corresponding Kafka topics. Project partner FRQ developed a single-web-page application (the "notification WebApp") based on the Test-bed's REST adapter.

The web application was tested for the interaction of 3Di with the Test-bed. No problems were reported during the testing of this web application. Two additional components were needed in order to enable the sharing of map layers: a FTP server where GeoTIFFs generated by 3Di solutions would be uploaded and a GeoServer where those GeoTIFFs would be published as shapefiles after the corresponding data processing. The FTP server was set up by GMV, the GeoServer was also set up by GMV. During the testing of the interaction between 3Di and the FTP server no problems were reported during testing. 3Di could export their GeoTIFF files to the FTP server where a GMV member would run the conversion to a shapefile which then was reported to the Socrates solution so that it could import this new layer.

3.4.3 Drone Rapid Mapping – Hexagon/Creotech

Aerial views of a crisis area can be of a high value for an improved situational understanding of the incident scene and its dynamics as well as for monitoring specific elements of a tactical situation. However, video images are suboptimal for general awareness of larger-scale situation and for understanding of a general topography of the crisis area. For such purposes, the orthophoto map is a significantly more effective form of presenting imagery information. The Drone Rapid Mapping solution enables access to up-to-date spatial information about the area of interest, its topography and changes resulting from the on-going crisis. It significantly improves situational awareness of higher-level commanders and it enables larger-scale damage assessment. It can also be used as reference information at the tactical level.

While information about the current situation (e.g. distribution of vehicles or extent of water during flooding) is not real-time, it can still be used as an approximation and the map can be regularly updated. Furthermore, a map is much more intuitive than video for persons not familiar with the area. Therefore, the orthophoto map seems to be an optimal form of presenting spatial information. The sharing of such maps should contribute to an improved COP and more effective coordination, particularly at silver command level. They can also be used for reporting of general situation at strategic level.

The Drone Rapid Mapping solution offers a quick generation of the orthophoto maps and 3D visualisations of incident/crisis area. It processes data acquired by any kind of drone available to rescue or crisis management actors, that is capable of capturing nadir (vertical) imagery and can be programmed for autonomous flight mission.

The resulting orthophoto maps of the DRM solution can be viewed online via a standard internet browser without any additional software required. In that case, it is displayed in a dedicated geoportal (considered as a part of the solution). It can also be easily integrated with any other GIS environment with the use of WMS service or it can also be downloaded for further processing. 3Di used the Java Adapter, Socrates the C# Adapter and Drone Rapid Mapping the REST adapter to connect to the Test-bed.

3.4.3.1 Integration into the Test-bed

The main output of the Drone Rapid Mapping solution consists of GeoTIFF files (orthophoto maps) which include detailed geographical information. As with the 3Di solution, these GeoTIFF files were too big to be exchanged through the Test-bed at the current status of Test-bed development. Due to this fact the Test-bed was not going to directly support the exchange of "operational" data between solutions but was providing the means to distribute the corresponding notification messages needed for the interaction of solutions. This schema required putting two additional components in place in order to enable the sharing of map layers: a FTP server where GeoTIFFs generated by the 3Di solution would be uploaded and a GeoServer where those GeoTIFFs would be published as shapefiles after the corresponding data processing.

The format and contents of the large data update and the map layer update messages were defined by the Test-bed team and incorporated to the Test-bed's infrastructure as ad-hoc standards for notifications. This way, the Test-bed was in charge of distributing received messages to those recipients which were subscribed to the corresponding Kafka topics. Project partner TNO developed a single-web-page application (the "notification WebApp") for the TypeScript adapter.

The exchange of "operational" data between Drone Rapid Mapping and the Test-bed was not directly supported, only corresponding notification messages were exchanged via the Test-bed. The web application was tested for the interaction of Drone Rapid Mapping with the Test-bed. No problems were reported during the testing of this web application. Contact details of all solution providers can be found in Annex 7.

3.5 Dry Run 1 and 2

Dry Run 1 (DR-1) was devoted to the technical integration of solutions and the scenario validation. However, the progress achieved before DR-1 was not as expected due to several reasons:

- The Trial scenario was not available in the final form for DR-1, only a reduced version of the scenario was defined.
- The formal commitment by external solution providers (and so the start of their work) was not achieved until one week before the DR-1.

The technical integration and testing during DR-1 was focusing on the internal solution Socrates OC for these reasons. For details about the integration steps with the Socrates OC solution please see section 3.4.1.

To prepare the input data for the Trial, the Trial owner provided a list of initial information which was used by solution providers prior to the Trial:

- List of resources (e.g. fire trucks, hospitals, ambulances) with their geolocations.
- List of events/missions/operational zones.

- Artificial border between Poland and Germany.
- List of injects that will trigger actions by practitioners.
- Definition of the flight area for drone.
- Definition of breaches and levees for flooding simulation.

In the preparation for DR-2, the scenario for Trial 1 has been worked out in more detail. From a technical perspective, it was requested to draw an overall information flow diagram showing all actors and their actions in a UML (Unified Modelling Language) diagram to achieve a detailed picture of the kind and format of the data to be exchanged.

Dry Run 2 was also focusing on practitioners to learn the solutions´ capabilities in the solution trainings.

From the technical perspective the overall objectives of Dry Run 2 were:

- Solution maturity check, organisational and technical constraints analysis.
- Solutions final integration with Test-bed.
- Theoretical and practical training on use of the solutions for Trial 1 participants.
- Running a pilot Trial 1 with practitioners' contribution.
- Final check of readiness for Trial 1.

3.5.1 Test-bed deployment

The Test-bed was deployed on a laptop which was positioned in the staff room at the SGSP facility. The laptop was connected via LAN cable to the SGSP network, and as such available to all other solutions which were also on the SGSP network. On the laptop the Test-bed ran inside a Docker[12] container, a program allowing quick instancing of pre-made container setups. This allowed for a quick deployment in advance and quick update during deployment. The solutions were connected to the Test-bed software via so called "adapters", program parts made to fit on one side to the Test-bed standards and on the other side to allow solutions to easily transfer their required data formats and standards. 3Di used the Java Adapter, Socrates the C# Adapter and Drone Rapid Mapping the REST adapter. During Dry Run 1 it was decided to use a very simple notification setup for the adapters of 3Di and Drone Rapid Mapping. This decision was made with the purpose to support them with a quick and simple integration due to the very limited amount of time in between their commitment to join Trial 1 and the Dry Run 1. One requirement of their solutions was the necessity of sending large data files, for which the Test-bed was not yet ready. The files were therefore sent directly to the FTP server of GMV and a notification that these files were sent had to be mentioned through the Test-bed for logging purposes. During Dry run 2 the adapter and Test-bed connection were tested and no problems were found.

3.5.2 Solution Integration results

An automatic exchange of data among involved IT solutions shall avoid that practitioners have to manually collect data from several solutions or even enter some output data from solutions into other solutions for further data processing. Map centric solutions such as the Socrates solution play a central role in the integration process as they are the medium of choice to display results from other solutions which can be geo-referenced. Using a map as the primary information layer opens the possibility to achieve a very structured presentation of information, with similar information types grouped in layers which can individually be switched on and off for best visibility of information. The output data of Drone Rapid

[12] Docker is a software that performs operating-system-level virtualization, also known as "containerization" (18).

Mapping as well as the output data of the 3Di solution were made available to Socrates and supported practitioners in gaining better situation awareness compared to a situation where the outputs of both solutions would have been shown separately. The overall information flow diagrams in Figure 3.6 and Figure 3.7 show the involved solutions in Trial 1 and the information exchanged among them.

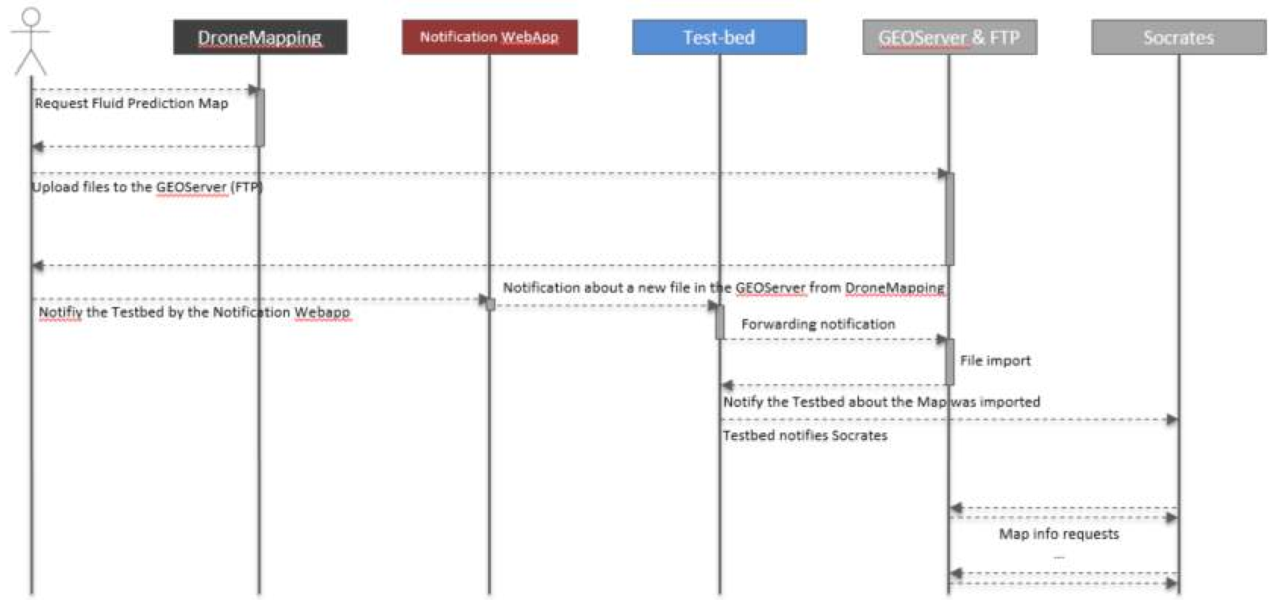


Figure 3.6: Sequence diagram for DRM during Trial 1 (part 1)

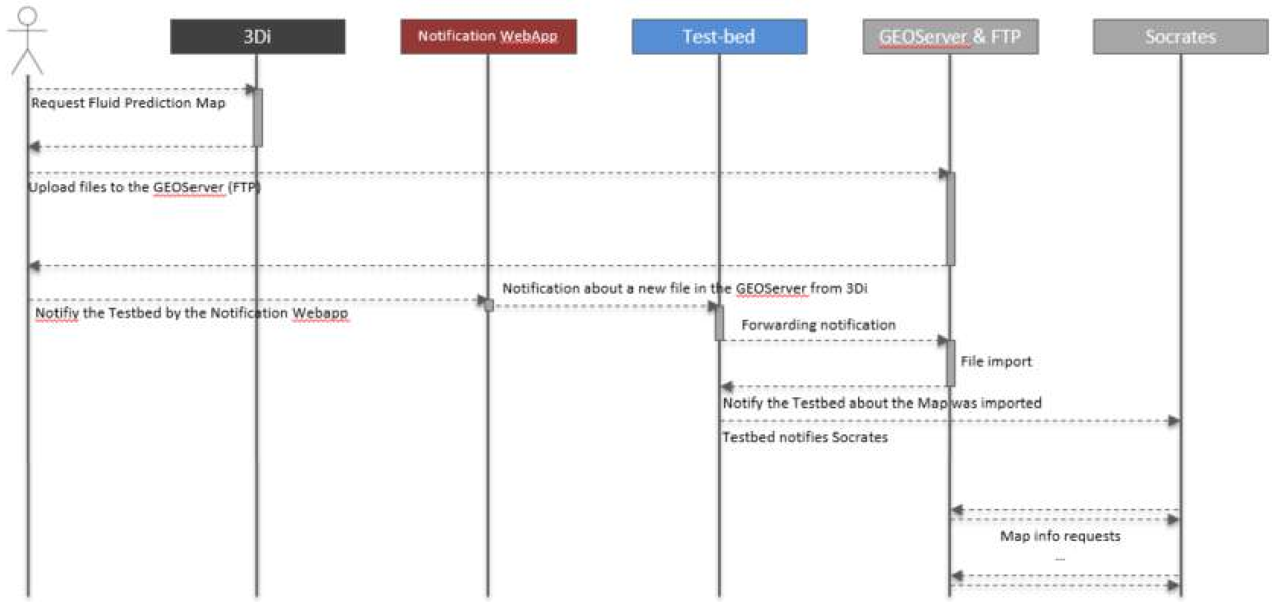


Figure 3.7: Sequence diagram for 3Di during Trial 1 (part 2)

The overall integration status of the involved solutions in Trial 1 can be considered limited. The limitation is mainly related to two facts:

- The amount of time available for integration after having the requirements for integration frozen. Only the final scenario (with all details available) set the final requirements for integration.
- The status of the Test-bed development at the time of Trial 1 execution.

4. Trial 2

This section describes the main activities related to the integration of solutions for Trial 2. This includes the Trial scenario, the selected solutions and the adaptations, integration and testing which were performed with these solutions to support the Trial scenarios. Trial 1 had two preceding Dry Runs for the preparation of technical and organizational matters. The preconditions for Trial 2 were a more mature status of the documentation relevant for the preparations (e.g. technical procedures described in **D934.21**) (4) and a more mature Test-bed. The performance of the Test-bed in the preparation and also during Trial 2 was good and proved that its architecture and functionality is a great support for performing Trials in the future. Goal of Trial 2 was to improve the cooperation and coordination among different agencies, organizations and actors involved in a large crisis and to evaluate the solutions which were selected for this purpose. The fictitious crisis consists in a chain of multiple events that takes place in a natural area shared by multiple countries, which will take part in the Crisis Management process with different protocols and hierarchies.

4.1 Scenario description of Trial 2

The Trial 2 scenario starts from an initial description presented in the Call for Application (CfA): It includes multiple incidents with cross-border dimension occurring on several sites. The main event is a large forest fire, threatening wildland urban interfaces in a Mediterranean environment. The main mission objective is therefore to “suppress the fire” to protect (according to the French doctrine):

- People (this includes dealing with casualties).
- Goods and infrastructures.
- The environment.

A support from the EU Civil Protection Mechanism is requested. Additional man-made and natural events will complete the scenario to increase the coordination needs and exchanges of information at horizontal level between agencies and countries, as well as vertical level along the command chain. The hazard is developing rapidly, and the resources of the departmental fire brigade are overcharged soon, requiring support from other fire brigades and national means (water bombers airplanes). As the fire is located near the border of Italy and the wind is oriented towards them, an international warning is issued. Also terrestrial support is requested through a pre-existing bilateral agreement, because of the geographical proximity. In addition, the EU Civil Protection Mechanism is asked to send additional airplanes.

This initial scenario description serves as a baseline to which more details were added from the Data Collection Plan (DCP) and the capabilities of the selected solutions. The scenario has been detailed to six consecutive sessions, called A-B-C-D-E-F.

For more details about the scenario of Trial 2 please see **D944.11** ***Report on Trial Action Plan – Trial 2*** (6).

4.2 Participating solutions

To coordinate the Test-bed integration of solutions in the Trial a set of meetings took place via teleconferences.

The list of the selected solutions taking part in the Trial is presented below together with a summary of each one, motivation for being in the Trial, Crisis Management gaps that it covers and the responsible contact details.

4.2.1 LifeX COP – FRQ

The COP provided by Project partner FRQ with the LifeX COP prototype is a map-based web-application that aims to fulfil the need of having overall situation awareness during a crisis. It enables the users to check on important factors such as the location or area of an incident, its severity, the type of incident (e.g. a fire) as well as available resources and many more. In addition, users can enter new information and share it with others. This becomes particularly important when many parties are affected, especially during cross-border disasters.

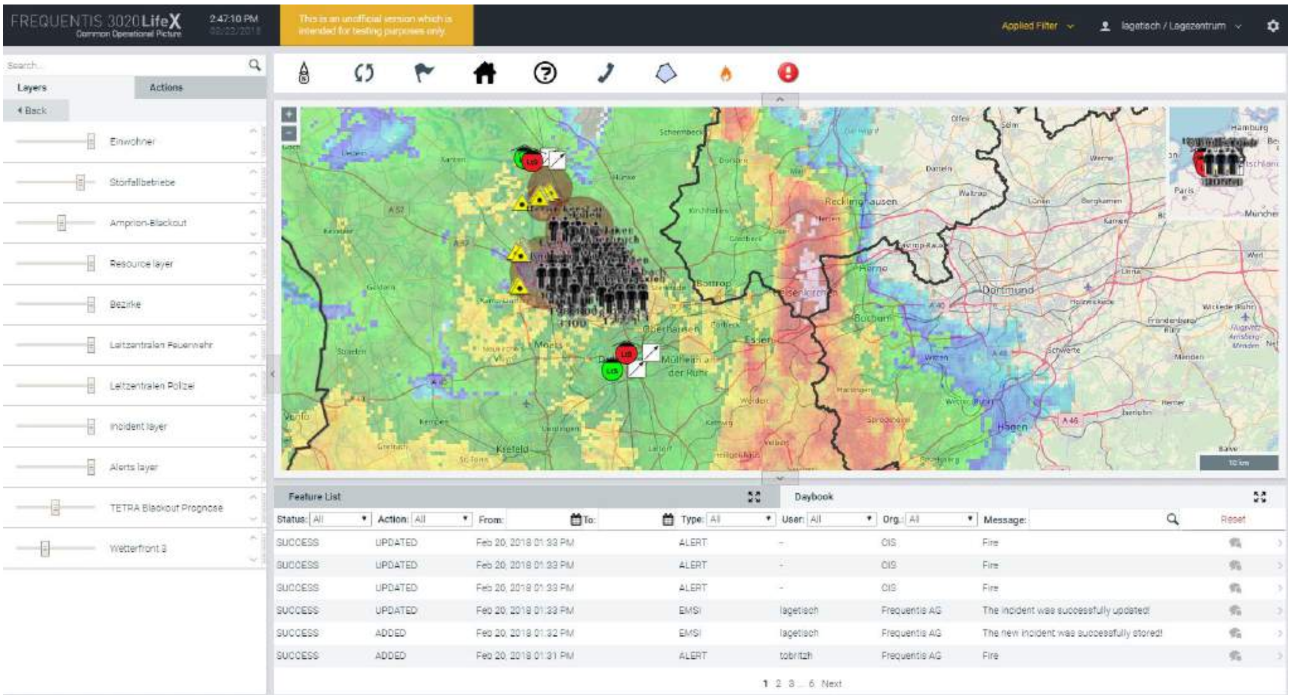


Figure 4.1: Screenshot of the LifeX COP solution

Using the LifeX COP all information shared between users is always consistent and up-to-date. Also, as some information might be more important than other to certain users, operators using the LifeX COP have the possibility to select what they want to see at any point in time by applying filters to the information. In addition to the information shown on the map, more detailed information can be displayed, filtered and searched for in optional list views, see Figure 4.1.

4.2.1.1 Covered CM Gaps

The LifeX COP solution was selected as it covers the following Crisis Management gaps:

Table 4.1: Crisis Management gaps covered by LifeX COP in Trial 2

CM Gap No.	Description
4	Real-time data and information fusion to support incident commander decision making.
5	Exchanging crisis-related information among agencies and organisations.
6	Common understanding of the information exchanged in response operations.

4.2.2 MDA C2 – MDA (Magen David Adom)

The solution MDA C2 provided by MDA (Magen David Adom) allows for an efficient, real time response to tasks on the field (e.g. people in need for medical assistance), by allocating the site, allocating the resources needed, tasking the resources and following up the accomplishment. This can be achieved for large number of incidents simultaneously and for large number of resources to the same task, grouping them if needed. The system receives and disseminates information to dedicated apps both used by the general public as well as by the team members and volunteers. The workflow of the MDA C2 solution is described in Figure 4.2.

Figure 4.2: MDA C2 solution workflow

4.2.2.1 Covered CM Gaps

The solution MDA C2 was selected as it covers the following Crisis Management gaps:

Table 4.2: CM gaps covered by MDA C2 in Trial 2

CM Gap No.	Description
19	Coordination in dealing with large numbers of severely burned casualties. Lack of efficient coordination mechanism to overcome the limited capacity to deal with large numbers of severely burned casualties at member state level.
20	Locating casualties in large forest fires. Limited ability to identify the location of injured/trapped/deceased casualties in large forest fires.
21	Providing medical assistance to casualties. Barriers in capability to provide medical assistance to casualties either by transporting them to a safe place or bringing emergency medical service to the scene (when medical care is not provided by firefighters' units).

4.2.3 SMAP – Thales

The SMAP solution aims at supporting crisis managers in the processing of social media for situation assessment purposes. Social media contains precious information which can bring an important contribution to situation assessment. This information can concern the incident(s) itself, the impact, or the needs of the population affected by the crisis. When trying to take this information into account, crisis managers face a major challenge which is the finding of relevant information in a huge volume of messages. Social media processing automates a user-defined collection process and proposes content mining tools based on content analysis and network analysis to find the relevant information.

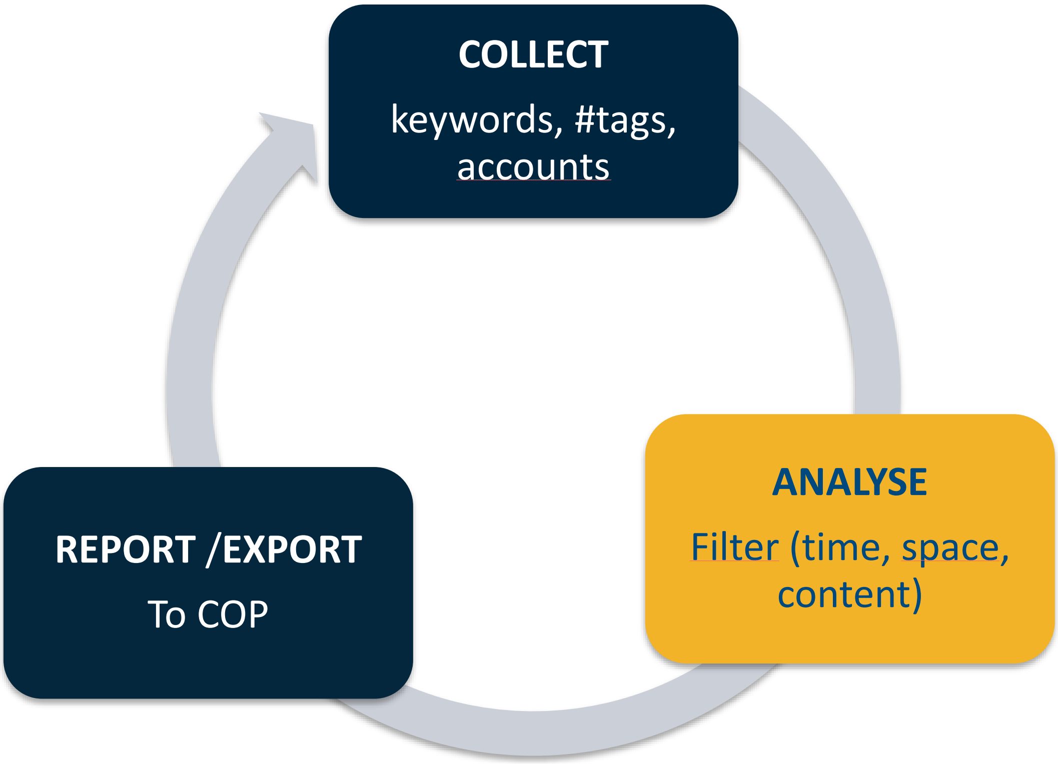


Figure 4.3: SMAP solution workflow

4.2.3.1 Covered CM Gaps

The solution SMAP was selected as it covers the following Crisis Management gaps:

Table 4.3: CM gaps covered by SMAP in Trial 2

CM Gap No.	Description
12	Incorporating information from multiple and non-traditional sources. Insufficiency in the ability to incorporate accurate and verified information from multiple and non-traditional sources (e.g. crowdsourcing and social media) into response operations.

4.2.4 CrisisSuite – Merlin

The solution CrisisSuite consists of a web application and an App for mobile devices. Main target is to provide a COP to several crisis teams working on the same crisis. CrisisSuite supports several file formats for pictures and is organized in a dashboard supporting crisis organisations in their plans for response actions. Figure 4.4 gives an overview of the structure of the solution which provides a COP in form of a textual logbook with attached pictures.

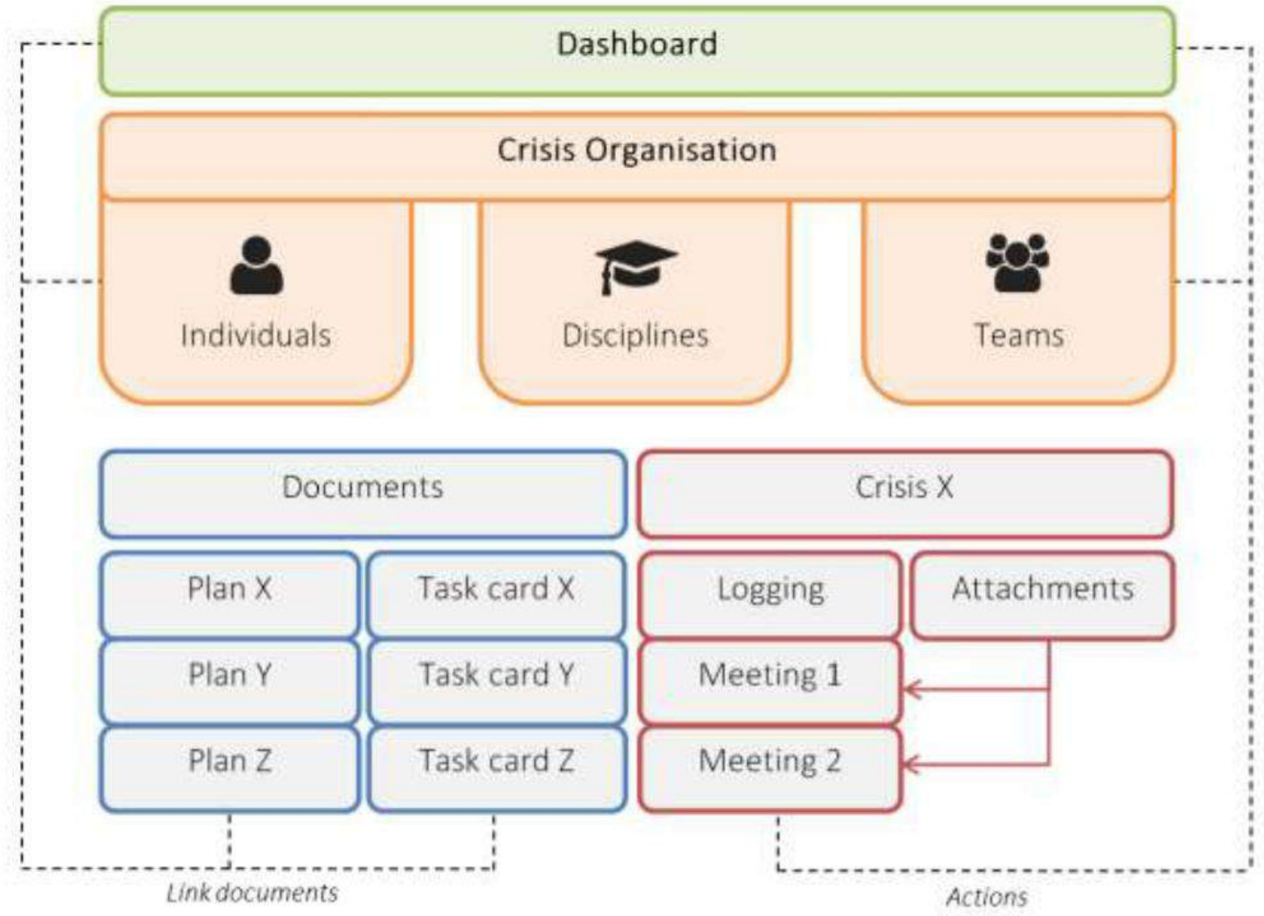


Figure 4.4: CrisisSuite overview

4.2.4.1 Covered CM Gaps

The solution CrisisSuite was selected as it covers the following Crisis Management gaps:

Table 4.4: CM gaps covered by CrisisSuite in Trial 2

CM Gap No.	Description
5	Exchanging crisis-related information among agencies and organisations.
6	Common understanding of the information exchanged in response operations.

4.3 Dry Run 1 and 2 integration of solutions

Due to the time limitations to prepare DR-1 and its conditions – more complexity, more solutions than in Trial 1 – Dry Run 1 focused on the technical set-up. However, the Test-bed integration was completed by all solution providers based on the current Trial scenario and the Test scenarios were carried out successfully to validate the step 2.2 of the Test-bed integration as described in **D934.21** (4).

The MDA C2 solution was deployed in several "boxes" (these are small rooms equipped with one practitioner working position consisting of one PC with several monitors) as the solution requires a multi-monitor setup (minimum three separate HD monitors) and thus cannot be installed in training rooms with standard PC setup (which include one HD monitor only).

4.3.1 Scenario and workflow diagram

In the scenario (described in section 4.1) the wind pushes the fire towards human settlements creating cascading effects; i.e. people trying to escape, getting trapped and injured and the industrial plant faces power outage (due to the fire effect on electricity transport and distribution lines).

The mission is carried out under the command of the Incident Commander (senior fire fighter officer). Coordination between agencies (fire fighters chain of command from two countries, environmental protection agency and emergency medical support) and countries, as well as horizontal and vertical information exchange are identified as major issues.

A set of online meetings took place for consolidating the scenario, to kick off the interaction of all the selected solutions and roll out the integration process. The information workflow diagrams as well as the test scenarios are key information for the solution providers to commence the adaptations in their solutions required to satisfy their role in the Trial scenario.

Subsequently a set of test scenarios (see Annex 5) was defined to group the intermediate actions in the Trial. Every test scenario is represented as a sequence diagram (implemented in UML) to illustrate the interaction between actors and solutions.

According to the scenario details and the capabilities of the selected solutions the information workflow diagram was generated. Figure 4.5 shows the interaction of the different actors in the Trial scenario

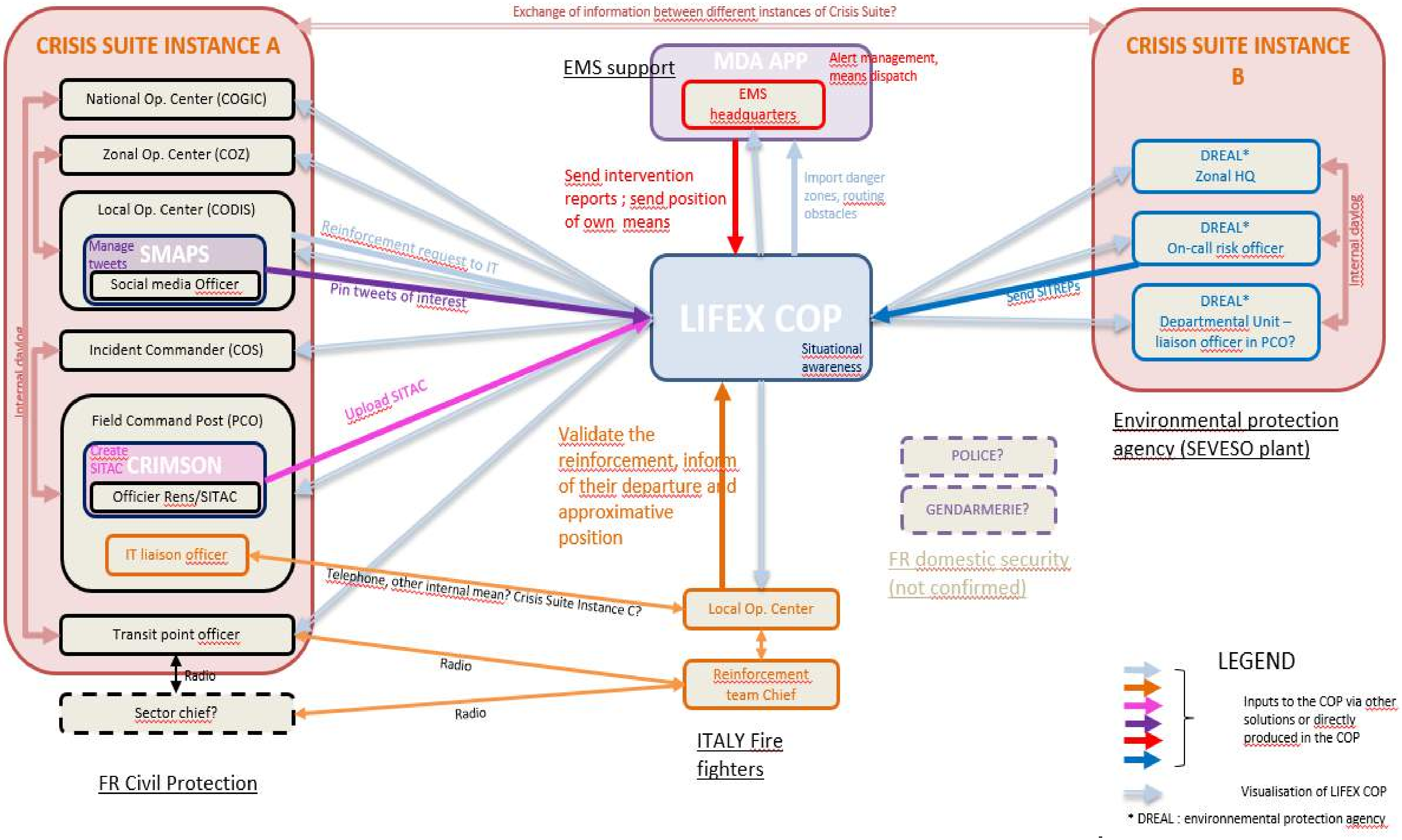


Figure 4.5: Information workflow diagram – Trial 2

4.3.2 Coordination and integration steps

4.3.2.1 Integration steps summary

The Table 4.5 shows a summary of the basic integration steps for all selected solutions and when they were achieved.

Table 4.5: Test-bed integration steps timestamps according to D934.21[13] for Trial 2

Steps	LifeX COP - FRQ	SMAP - THALES	MDA C2 - MDA	CrisisSuite - Merlin
Step 0	Done by 04/06/2018.	Done by 01/06/2018.	Done by 01/06/2018.	Done by 01/06/2018.
Step 1	Done by 20/06/2018.	Done by 14/06/2018.	Done by 20/06/2018.	Done by 14/06/2018.
Step 2[14]	Done by 21/06/2018, and officially validated in Dry Run 1.	Done by 21/06/2018, and officially validated in Dry Run 1.	Done by 21/06/2018, and officially validated in Dry Run 1.	Done by 21/06/2018, and officially validated in Dry Run 1.

4.3.2.2 Step 0

According to the **D934.*21 Solution testing procedure*** (4), step 0 consists of the generic integration of standalone solutions to the Test-bed. During this step the solution owner defines the main user stories of his/her solution, translates them to test cases, and integrates the solution to the Test-bed reference implementation.

However, this step is not Trial specific and given the short preparation time between selection of solutions and Dry Run 1, it was simplified to a connectivity check-up to guarantee that the solution providers were technically able to connect to the Test-bed.

An information package was delivered to each solution provider summarizing all the needed documentation (links to Github, PowerPoint how-to's and access to the Test-bed support channels on Slack) to carry out this step. On top of this, a series of independent meetings were scheduled with each solution provider to guide them through the process.

Annex 2 contains the technical Test-bed messages of each solution provider during the integration step 0.

4.3.2.3 Step 1 and step 2.2

Due to time limitations, step 1 and the step 2.2 were merged. This task was facilitated using an online instance of the Test-bed, making possible the integration of multiple solutions (step 2.2) by the time the standalone integration (step 1) was carried out.

---

[13] D934.21 Solution testing procedure (4)

[14] According to Deliverable **D934.21 *Solution testing procedure***, step 2 consists of two sub-steps. As step 2.1 is a technical design set-up, carried out by the Trial Committee and not by the solution providers, step 2 only mentions the work done in step 2.2.

It turned out that the existence of an online Test-bed is a more than convenient way to facilitate a collaborative methodology to develop the Test-bed integration of those solutions that implied data exchanged between solutions, as some solutions consumed information produced by other solutions.

Once the scenario details were released the information workflow diagram was generated collaboratively - a series of meetings were organized for those sequence diagrams that involved the interaction between two solution providers so they could check these interactions. In these meetings each solution provider received a sub-set of these sequence diagrams in which they were individually involved so to simplify the development of their adaptations into the Test-bed and the scenario.

4.3.2.4 Test scenarios list

From those test scenarios that involve data exchange between two or more solutions, a technical data exchange diagram (see Figure 4.6) was created to represent the data exchanged between solutions through the Test-bed.

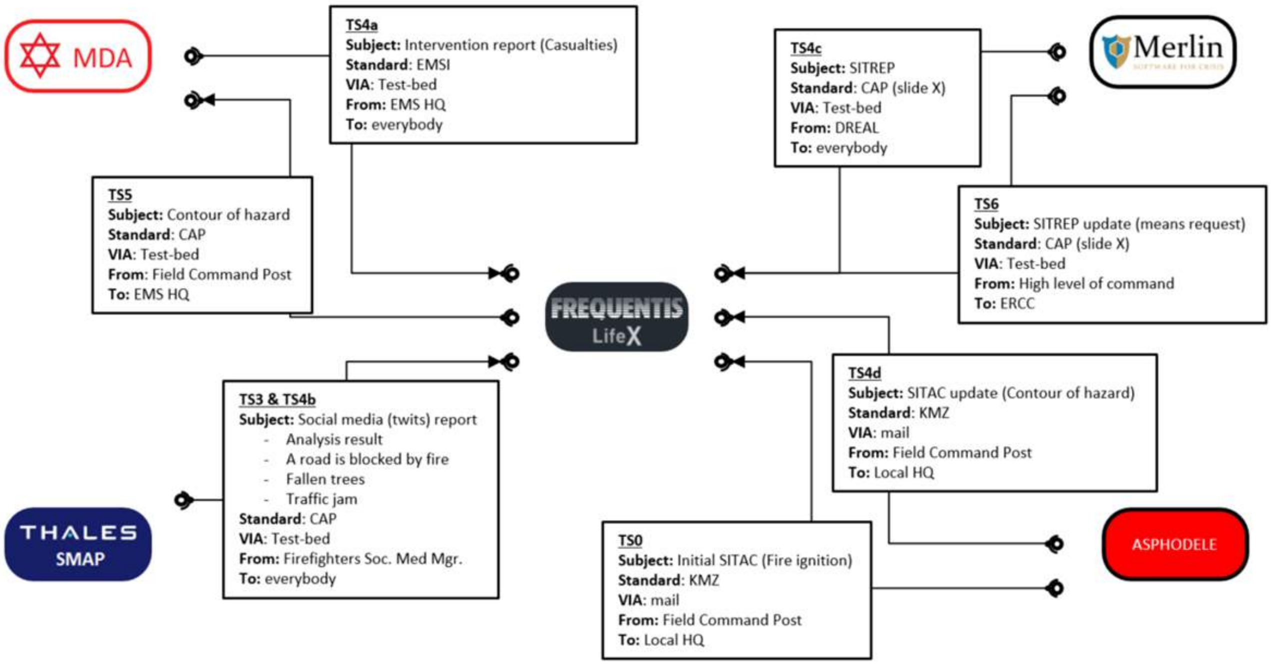


Figure 4.6: Data Exchange Diagram of Trial 2 from Dry Run 1

For the sake of readability, here again the test scenarios in list format

Table 4.6: Summary of test scenarios

Test scenario	Test scenario brief description/objective
TS0	The information in the SiTac[15] is displayed correctly in LifeX COP.
TS1a	A new client from a different organization can be opened.

[15] Situation Tactique (SiTac) is a french term for the standardization of Firefighting Tactical Situation Management

Test scenario	Test scenario brief description/objective
TS1b	Information of one organization is not shown in the view of a different organization.
TS2	A SitRep[16] from a CrisisSuite organization can be shared with another CrisisSuite organization.
TS3	SMAP analysis results are available and accessible in LifeX COP.
TS4a	To visualize the information in the intervention report in the LifeX COP.
TS4b	SMAP analysis results are available and accessible in LifeX COP.
TS4c	SitRep is available and accessible in LifeX COP.
TS4d	The information in the SiTac is displayed correctly in LifeX COP.
TS5	The contour of hazard is visualized in MDA C2, so vehicles can be routed taking the hazard contour into account.
TS6	SitRep is available and accessible in LifeX COP.

Test scenarios TS0 – TS6 are described in detail in Annex 5.

4.3.3 Solution providers' adaptations and integration technical details

This section contains descriptions of each solution provider regarding the adaptations which were needed in the solution to enable the support of the Trial 2 scenario. Necessary adaptations are described in terms of UI (User Interface) adaptations as well as back-end adaptions, i.e. changes that were made in controllers and the design of the Test-bed connection.

4.3.3.1 LifeX COP – FRQ

The performed UI adaptations and back-end adaptations were the following:

Resource information retrieved via CAP message from the Test-bed were added: the adaptations of LifeX COP for Trial 2 are based on the additional data coming in the CAP messages received from CrisisSuite and SMAP consisting of producing a linkable text in the related resource information panel which produces a pop-up of the link remote content. Figure 4.7 shows the resource details and content pop-up adaptation; Figure 4.7 shows the resource details and content pop-up adaptations of the LifeX COP.

[16] Situation Report (SitRep) describes a document outputted by CrisisSuite.

UI Adaptations (front-end)

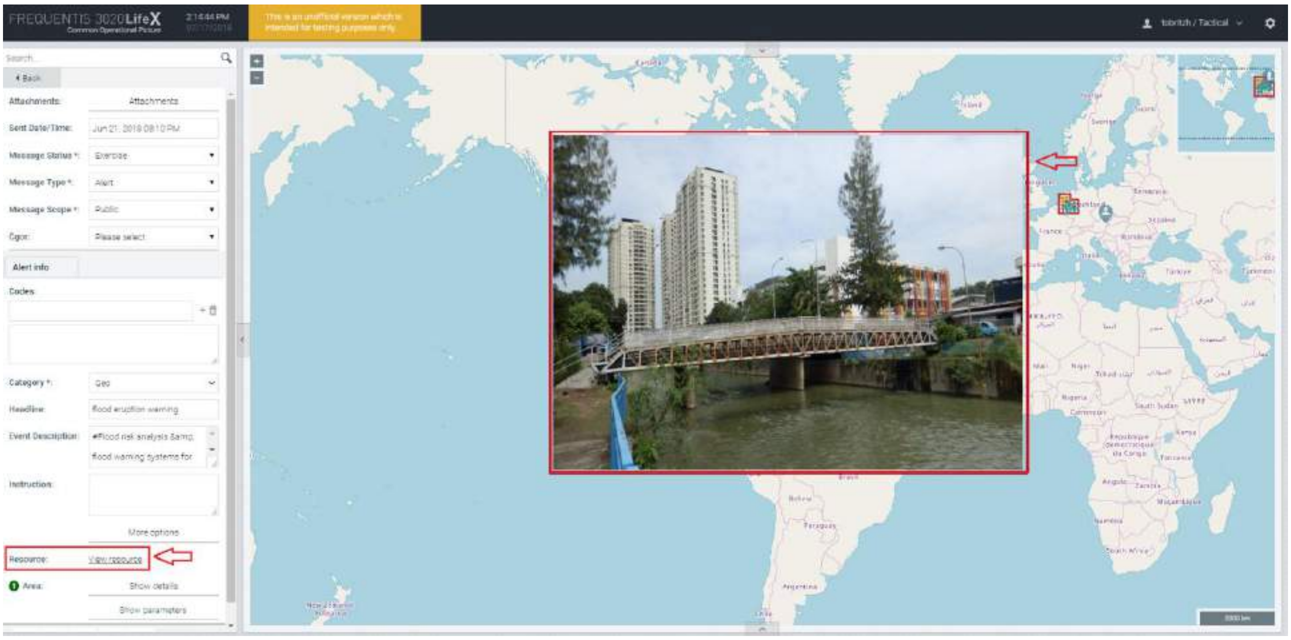


Figure 4.7: Resource details and content pop-up adaptation of LifeX COP

As users of LifeX COP can switch each information layer on and off separately, it can happen that new information is missed when the respective layer is not switched on. To notify a user that new information became available, a pop-up notification was created in LifeX COP, see Figure 4.8.

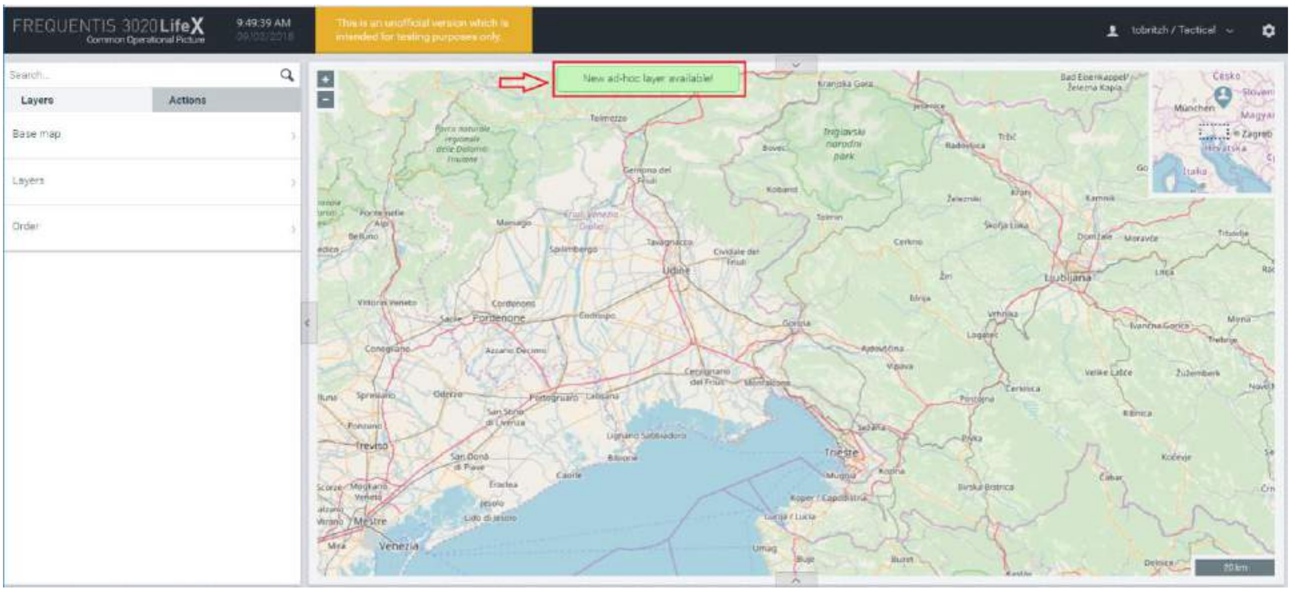


Figure 4.8: A pop-up notifies the user that a new KML layer became available

Controllers (back-end)

Following back-end components have been changed in LifeX COP:

- CAP controller
  - CAP extraction: extracting the CAP message from the received AVRO message.
  - CAP creation: create from the extracted CAP message the LifeX COP internal CAP message structure

    - CAP XML creation: as the LifeX COP needs also the original CAP XML message, and as this is not part of the received message, the XML had to be created. The Google CAP Library was used to enable this. The XML message is also validated and sorted as part of the internal CAP message.
- EMSI controller
    - EMSI extraction: extracting the EMSI message from the received AVRO message.
    - EMSI event compose: create the LifeX COP internal EMSI event object structure.
    - EMSI resource compose: In case of available resource information in the received AVRO message, also the internal EMSI resource object structure is created.
- KML Layer:
    - KMZ/KML layer extraction: the KML layer and the resources are extracted from the KMZ file and are copied to the server to be provided for the clients.
    - New layer notification: when a new layer becomes available, the connected clients are notified so that they can load and display the layer.
    - The back-end was extended by providing a simple drag & drop KMZ upload page to inject the KML layers into the COP.
- PowerPlant information layer:
    - A PowerPlant should be displayed as an overlay layer. This is uploaded as GeoTIFF to the geoServer and exposed as WMS layer to the LifeX COP client.
- Station layer:
    - Stations are retrieved as geoJson from the gateway translator via the DRIVER+ Test-bed.
    - The stations are going to be stored in the LifeX COP internal database.
    - A notification is send to each client that a station occurred or changes any information.
- Unit layer:
    - Units are retrieved as geoJson from the gateway translator via the DRIVER+ Test-bed.
    - The units are going to be stored in the LifeX COP internal database.
    - A notification is send to each client that a unit occurred or changes any information/location.
- Item layer:
    - Items are retrieved as geoJson from the gateway translator via the DRIVER+ Test-bed.
    - The items are going to be stored in the LifeX COP internal database.
    - A notification is send to each client that an item occurred or changes of information/location happened.

These changes were required in order to enable the data exchange requested by the Trial scenario, the integration of data coming from other solutions is shown in the UML diagram in Figure 4.9.

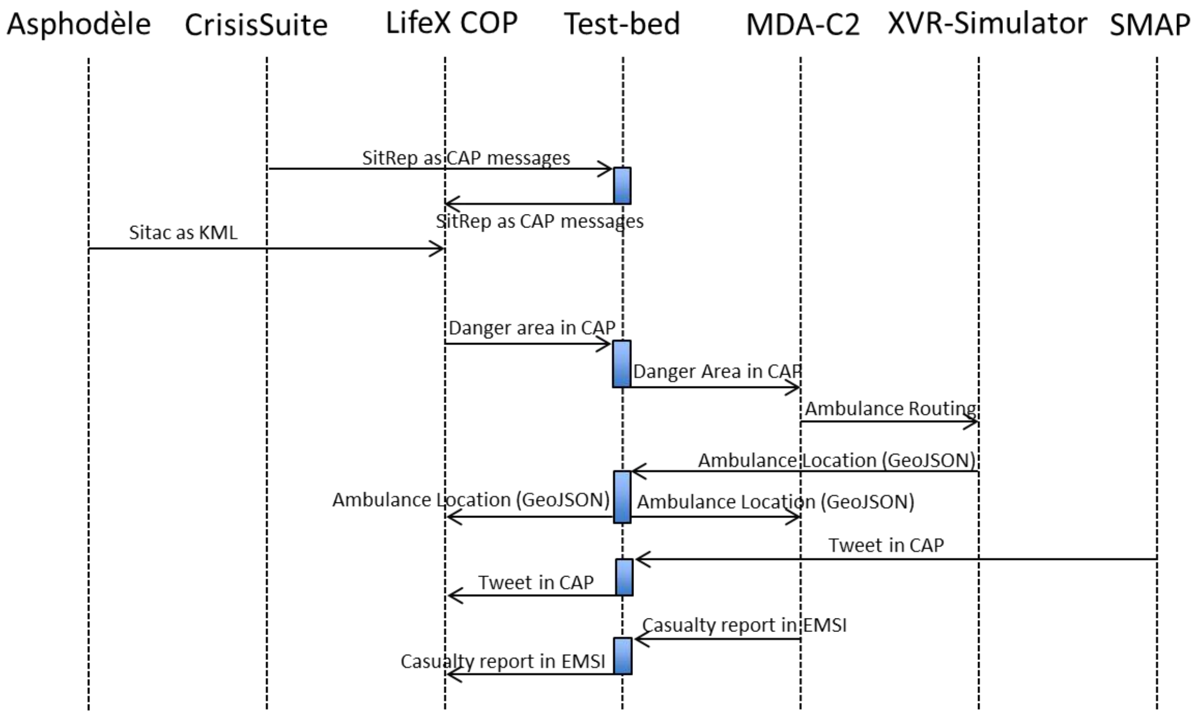


Figure 4.9: UML diagram for the LifeX COP integration

Test-bed connection design (interface and use of the adapter)

The Test-bed connection can be implemented either with a Test-bed adapter embedded in the solution or with a Test-bed adapter as middleware in between the solution and the Test-bed. For LifeX COP an embedded Test-bed adapter design has been implemented, see Figure 4.10.

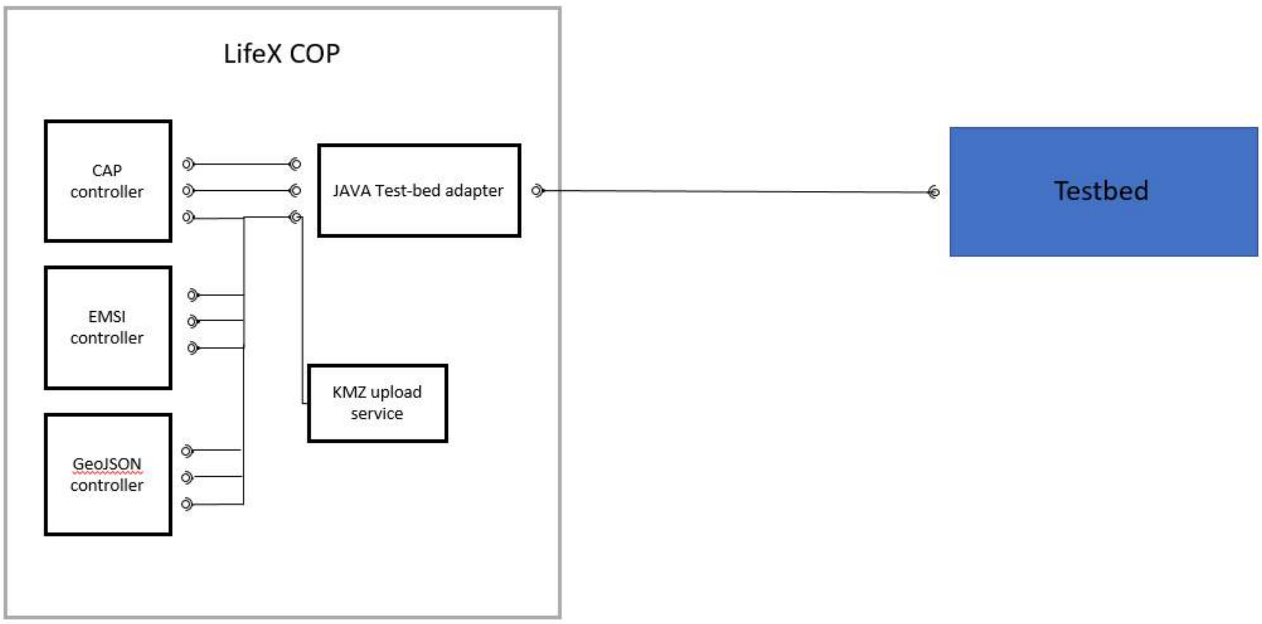


Figure 4.10: Test-bed connection diagram for LifeX COP in Trial 2

4.3.3.2 MDA C2 – MDA

During the preparation phase for Trial 2 the following adaptations were performed on the MDA C2 solution.

UI Adaptations (front-end):

Map adaptations: adaptation of maps of France and Italy in MDA’s command and control system. Calculation of the routes for emergency vehicles while taking into accord the danger zones. Enabling the representation of the polygons on the map.

Controllers (Back-end):

Command conversions were implemented from/to the simulator according to the Trial scenario to MDA’s command and control system for receiving and establishing the evacuation destination (e.g. hospitals). This allowed to establish an incident in MDA’s command and control system and to transfer the information to the simulator in real time.

Preparing a command and creating a simulation of receiving a picture form the incident.

Preparing a command and creating a simulation of receiving an update location form the incident and update the location in the map of MDA’s command and control system.

Receiving the ambulance location from the simulator and update the ambulance location on the map in real time.

Reporting of the casualty number. Preparing a command of the injury description of each casualty in the incident.

Preparing an environment of Italy and France in MDA’s command and control system.

These changes were required in order to enable the data exchange requested by the Trial scenario. The integration of data coming from other solutions is shown in the UML diagram in Figure 4.11.

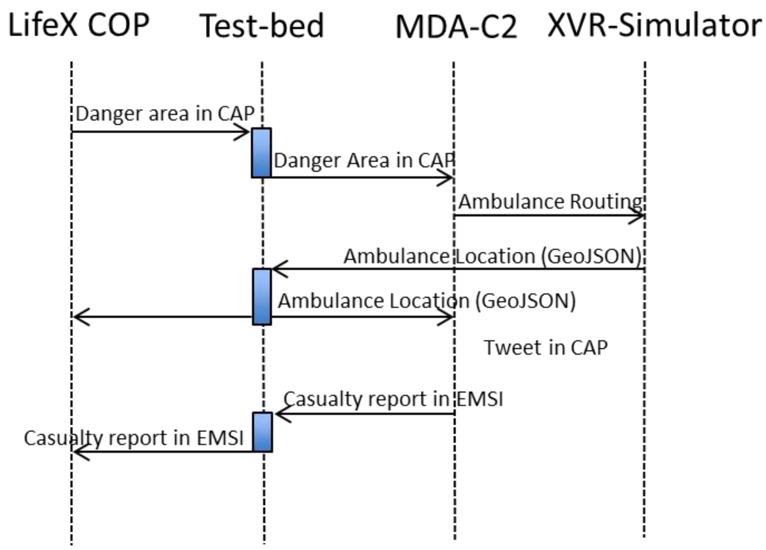


Figure 4.11: UML diagram for MDA C2 integration

4.3.3.3 SMAP – Thales

The adaptation performed on SMAP had the following objectives:

- Connect SMAP to the Test-bed.
- Adapt the HMI of SMAP for Trial 2.
- Enable SMAP to export information (required by Trial 2).
- On the fly collect optimisation (required by Trial 2).

SMAP is connected to the Test-bed via the REST adapter. Figure 4.12 shows the information flow for SMAP: the XVR simulator creates tweets, the Twitter gateway publishes them to Twitter (privately), and SMAP sends CAP messages to the Test-bed which are consumed by LifeX-COP. Other information such as views of the SMAP Dashboard is exchanged directly by a URL link sent to CrisisSuite. There is no direct link with XVR's simulator but for the Trial scenario, Tweets need to be injected in Twitter. They are created on Twitter following a scripted sequence by XVR. This chain has been tested successfully.

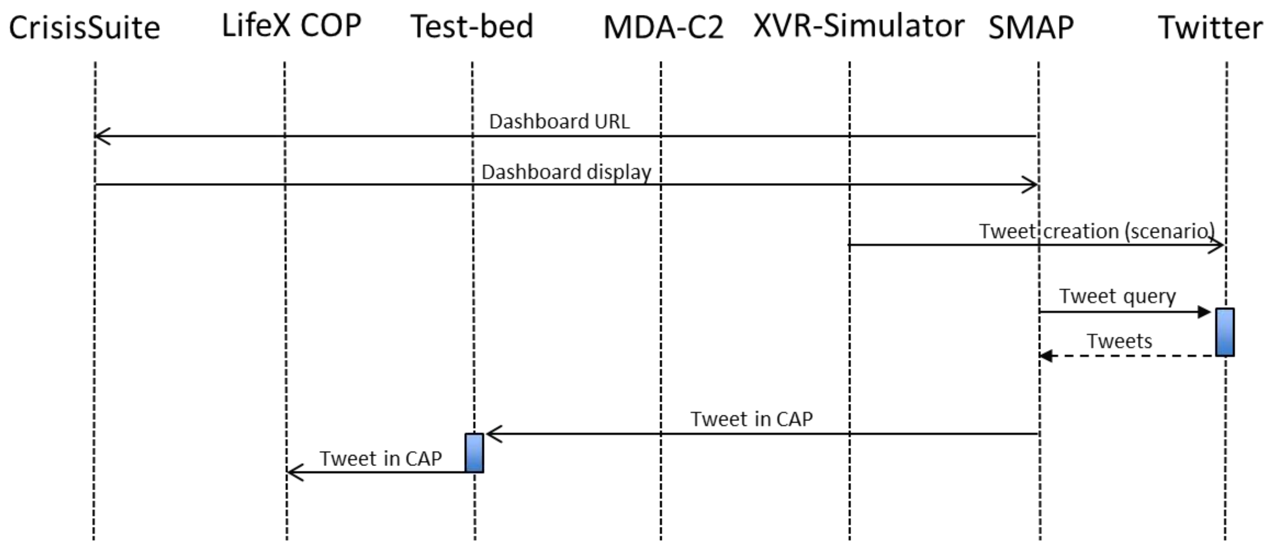


Figure 4.12: SMAP integration UML diagram

UI Adaptations (front-end)

The initial HMI (Dashboard) has been simplified. Only the ones corresponding to the functions activated by Trial 2 script have been kept. This simplification was necessary due to the limited time available for the training on the SMAP solution. The original dashboard contained too much information thus confusing users with limited SMAP experience.

The following functions have been removed from the dashboard: event detection and community detection.

The following functions have been simplified: filtering (the filtering on account names has been removed as it is not needed in the Trial scenario but could lead to wrong search results).

Controllers (Back-end)

The back-end service has been adapted to enable the export of selected tweets from SMAP and to generate CAP messages.

The following changes were made in the back-end service:

- Allow usage from remote browser (not only localhost as previous version).
- Transform selected tweet info into CAP message.
- Send CAP message to the REST Test-bed adaptor.
- Set up REST adaptor.
- Beta testing of REST adaptor (content type issue + typo).

The initial use that was envisaged by SMAP was to collect continuously on some particular keywords and refine the collection when the crisis emerges and the social media manager is activated. In this process, most passed tweets were collected in advance.

Because of the GPDR analysis (which showed that the storage time of the collected information needed to be reduced as much as possible) it was decided not to collect in advance, but to start the collection at the moment the social media manager is activated. This in consequence required enhanced collection power as the interesting tweets needed to be collected as fast as possible to improve operational velocity. This required some optimisations in the collection process.

Test-bed connection design (interface and use of the adapter)

For SMAP a Test-bed adapter as middleware in between the solution and the Test-bed has been used as shown in Figure 4.13.

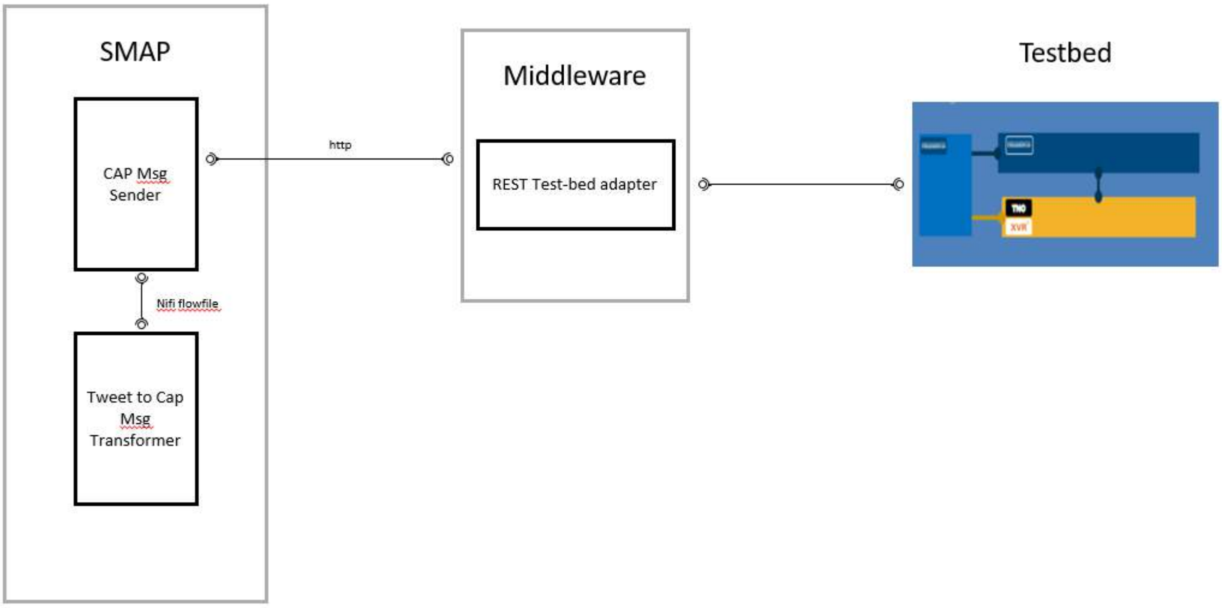


Figure 4.13: Test-bed connection diagram required for SMAP in Trial 2

4.3.3.4 CrisisSuite – Merlin

CrisisSuite's architecture has been designed considering already the export of certain information to external 3rd party solutions. In the context of the Trial it contains a structure that allows sending updated SitReps to an external provider. It was therefore just required to develop a component that knows how to transmit SitReps as a CAP message to the Test-bed so it can be consumed by the solution that needs it.

During the preparation phase for Trial 2 the following adaptations were performed on the CrisisSuite solution.

UI Adaptations (front-end)

Two new types of SitRep questions were added: location and location area.

Result: when editing a SitRep with existing location information the map window displays the location when it opens up now.

After developing a custom extension component to export SitReps and configuring the back-end settings the regular usage of the application includes the integration into the Test-bed and the export of the required data according to the Trial 2 scenario.

Controllers (Back-end)

Implementation of changes to allow for updating a CAP message (instead of adding a new message when a SitRep is updated).

Use of the location given in SitRep question as origin of CAP message. When a SitRep contains multiple location fields, they can be sent as separate CAP messages to the Test-bed.

Display of the most important information in de CAP headline field instead of the description field.

Mapping of SitRep location fields to different CAP categories based on field name (used by the COP tool to select the proper icon).

The components added or modified in CrisisSuite for participating in the scenario – and hence, integrating into the Test-bed – are displayed in Figure 4.14.

The development team had to create a new connector (DriverPlusConnector), this creates a CAP XML based on a SitRep and sends it to the Test-bed REST adapter.

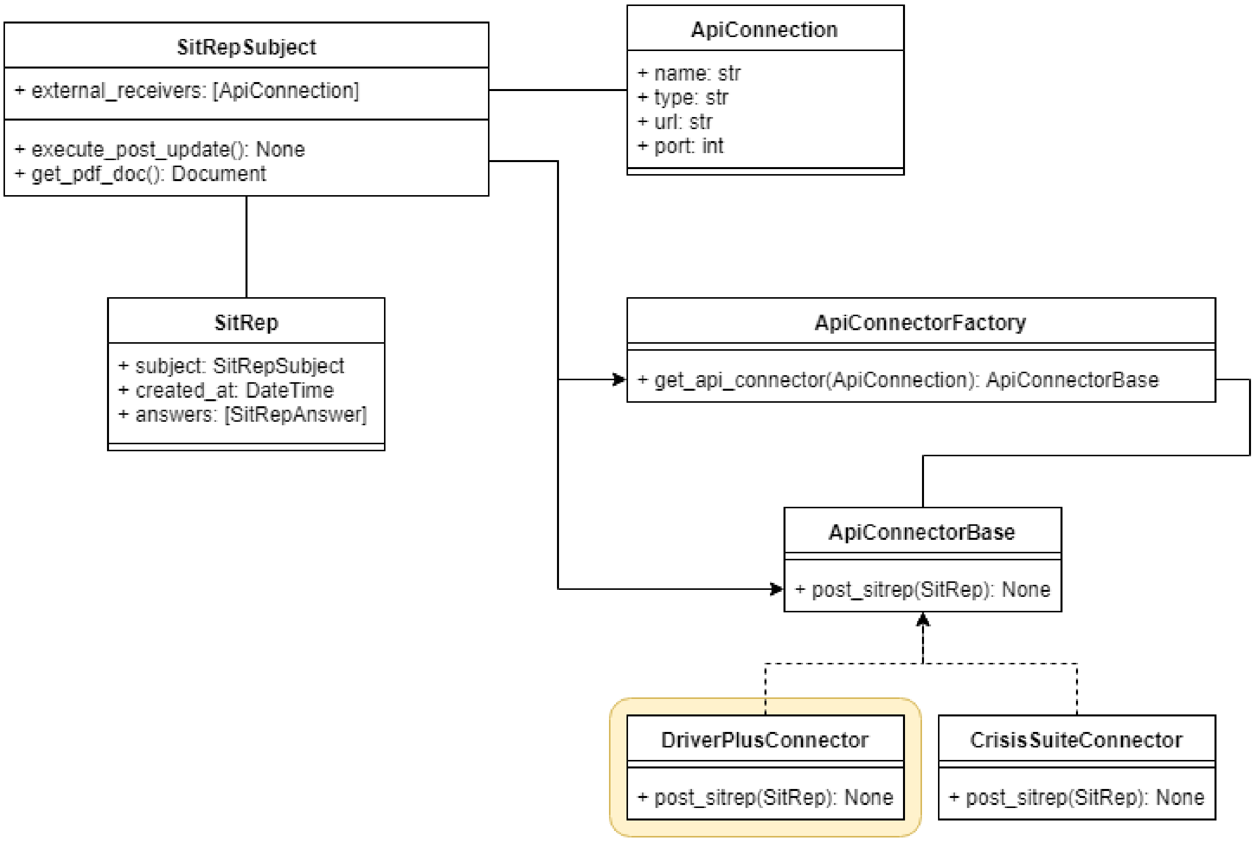


Figure 4.14: UML diagram of back-end components for CrisisSuite adaptation in Trial 2

Other effort

A separate instance of CrisisSuite was required, requesting a separate virtual machine which was used by the Italian fire fighters in the Trial. Three given SitRep questionnaires were rebuilt in CrisisSuite in order to be used during the Trial.

Test-bed connection design (interface and use of the adapter)

CrisisSuite is a centralised web-based application following a server-client (1:N) architecture allowing multiple clients. The Trial runs in an environment as isolated as possible, to minimize the dependency on the internet and to avoid the influence of discontinuities of the internet service. Thus, the set-up required for Dry Run 1 didn't use the default setup with a server in the cloud to connect to the Test-bed. Instead, a virtual server that runs locally was deployed.

As the Figure 4.15 shows, a Test-bed adapter instance ran on this virtual machine, which required only some small changes to the properties files and a source code update due to different Java versions.

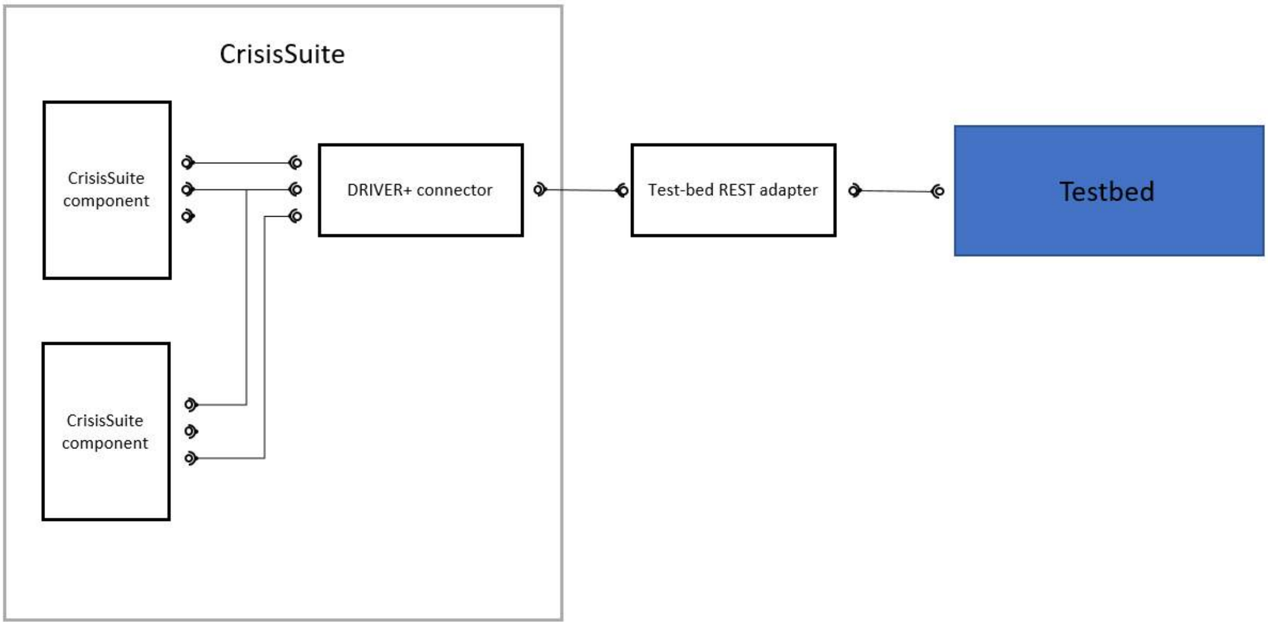


Figure 4.15: Test-bed connection diagram required for CrisisSuite in Trial 2

4.3.4 Test-bed deployment

The Test-bed was deployed on a laptop for Trial 2. Annex 6 lists the IP address and the technical users of the Test-bed.

The Test-bed components used in Trial 2 were:

1. The Test-bed itself.
2. The admin tool.
3. The Twitter gateway.
4. The online observer support tool (only for testing).

Active Test-bed features:

1. Core topics
   a. System_heartbeat (all solutions sending out their "alive" status).
   b. System_admin_heartbeat (the admin tool sending out its "alive" status).
   c. System_logging (logs for all solutions and admin tool).
   d. System_topic_access_invite (allowing solutions to listen to all standard topics).
2. Standard topics (https://github.com/DRIVER-EU/avro-schemas/tree/master/standard).
   a. Standard_cap (used by LifeX – CrisisSuite – MDA – SMAP).
   b. Standard_emsi (used by MDA - LifeX).
   c. Standard_geojson-xvr (not used during Dry Run 1).
   d. Standard_mlp (not used during Dry Run 1).

The bugs/problems that were encountered in the preparation of Trial 2 are: The Test-bed ran correctly. One problem was that the database of the admin tool needed a reset before configuring the solution client IDs. Most of the bug fixing was necessary for the exchange of messages between the different solutions.

4.3.5 Scenario simulation

Preparations of XVR for the scenario simulation relied heavily on the exact content of the stories to run in the Trial, these stories were developed step by step before, during and in between the Dry Runs. The simulation which was envisaged consisted of four parts as input to the Trial sessions:

1. The simulation of resources available to the crisis teams like fire truck teams, ambulances, airplanes etc.
2. The visualisation of the on-site incidents like forest fires, the fire line with flames and smoke, wounded people on a camping site who inhaled smoke and got burned.
3. The role-played information injected to the Trial at given times, to give the proper information to the connected solutions and thus provide a story to practitioners of the Trial.
4. Everything else that is part of setting the Trial but not played by the participants.

The simulation used in Trial 2 touched on all four aspects but in a reduced setup. Only the ambulance units are simulated as moving resources. The SiTac injects (a sort of battle plan drawn by the firefighters) were shown on the COP solution. These injects were sometimes simulated by the XVR simulator, sometimes they were events created by practitioners. The visualisations of incident locations were created by EcASC Valabre and were not linked to the Test-bed or other simulation aspects.

4.3.6 Solution Integration results

An automatic exchange of data among involved IT solutions shall avoid that practitioners have to manually collect data from several solutions or even enter some output data from one solution into any other solutions for further data processing. Map centric solutions such as LifeX COP play a central role in the integration process as they are the medium of choice to display results from other solutions which can be geo-referenced. Using a map as the primary information layer opens the possibility to achieve a very structured presentation of information, with similar information types grouped in layers which can individually be switched on and off for best visibility of information. The availability of layers also depends on the role of the user (e.g. one user group in Trial 2 was not allowed to see all layers of the COP). The overall information flow diagram in Figure 4.16 shows the involved solutions and the information exchanged among them. In this diagram Asphodele represents a legacy solution for the creation of Sitacs. Twitter represents the social media platform Twitter. The output data of MDA-C2 as well as the output data of SMAP were made available to LifeX COP and supported practitioners in gaining better situation awareness compared to a situation where the outputs of both solutions would have been shown separately. CrisisSuite as a dashboard-oriented COP was used in parallel to provide an overview of relevant data coming from the SMAP Solution and to create situation reports.

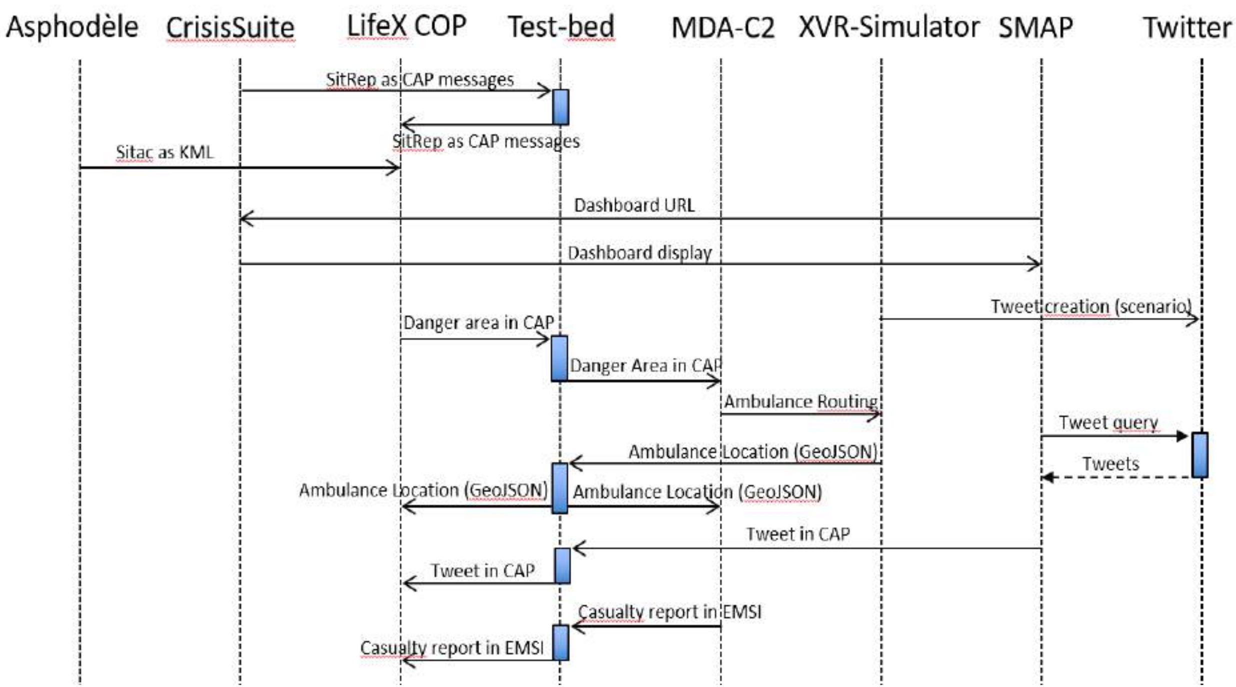


Figure 4.16: Overall information flow diagram for Trial 2

4.3.6.1 Test-bed integration details of the trialled solutions

There was an instance of LifeX COP server running on a Virtual Machine hosted by a laptop allowing multiple clients.

There was a local instance of SMAP server running on a laptop allowing multiple clients (although just one client was needed).

There was a local instance of MDA C2 server running on a setup hosted by a laptop. No client instances were deployed for the Dry Run 1 due to a more detailed and extensive scenario being designed during the Dry Run 1 for the following Dry Run 2.

There was a local instance of CrisisSuite server running on a Virtual Machine hosted by a laptop allowing multiple clients.

The details for deployment with the IP addresses and technical users for all solutions can be found in Annex 6.

4.3.6.2 Results of the test scenarios

The validation of step 2 consists of the consecutive execution of the test scenarios described in section 4.3.2.4.Table 4.7 summarizes the results of this Test-bed integration validation.

Please note that Test-bed messages for test scenarios which involved data exchange through the Test-bed are included in Annex 2.

Table 4.7: List of test scenario results of Trial 2 - Dry Run 1

Test ID	Test scenario ID	Test description and objective	Results	Comment
1	TS0	The information in the SiTac is displayed correctly in LifeX COP.	OK	Icons included in KMZ don't include the text shown in Asphodele. No Test-bed messages in this test scenario.
2	TS1a	A new client from a different organization can be opened.	OK	No Test-bed messages in this test scenario.
3	TS1b	Information of one organization is not shown in the view of a different organization.	Not applicable	There were no rights defined yet for Dry Run 1, hence, this test scenario couldn't be carried out. No Test-bed messages in this test scenario.
4	TS2	A SitRep from a CrisisSuite organization can be shared to another CrisisSuite organization.	OK	No Test-bed messages in this test scenario.
5	TS3	SMAP analysis results are available and accessible in LifeX COP.	OK	Technical Test-bed messages are listed in Annex 2.

Test ID	Test scenario ID	Test description and objective	Results	Comment
6	TS4a	Visualize the information in the intervention report in LifeX COP.	POK	MDA sends the message properly, but LifeX COP doesn't display all information that comes with it. Technical Test-bed messages are included in Annex 2.
7	TS4b	SMAP analysis results are available and accessible in LifeX COP.	OK	Technical Test-bed messages are included in Annex 2.
8	TS4c	SITREP is available and accessible in LifeX COP.	OK	Technical Test-bed messages are included in Annex 2.
9	TS4d	Information in the SiTac is displayed correctly in LifeX COP.	OK	Technical Test-bed messages are included in Annex 2.
10	TS5	Contour of hazard is visualized in MDA C2.	OK	Technical Test-bed messages are included in Annex 2.
11	TS5	MDA can route vehicles considering this contour of hazard.	OK	It is considered partial because there's nothing to be routed from the simulation. Technical Test-bed messages are included in Annex 2.
12	TS6	SITREP is available and accessible in LifeX COP.	OK	Technical Test-bed messages are included in Annex 2.

The integration tests refer to six consecutive sessions (these are parts of the Trial scenario - for details see **D944.12** (7)), the sessions being:

- **Session A:** Alert calls Identification of the ignition point, sending of a first group, Arrival of the Incident Commander (IC) on-scene, Check of the ignition point.
- **Session B:** Arrival of new IC, Situation update and request for additional means, Check and dispatch of additional means from CODIS (Centre Opérationnel Départemental d'Incendie et de Secours = Departmental Operational Center of Fire and Rescue), COZ (Centre Opérationnel de Zone = Area Operations Center), COGIC (Centre Opérationnel de Gestion Interministérielle des Crises = Interdepartmental Operations Center for Crisis Management); Dispatch of Italian sections and integration into the operations; integration of French additional means into the operations; injured Italian fire-fighter - exchanges of information between French & Italian authorities.
- **Session C:** Power outage in a SEVESO plant related to the Fire, plan informs CODIS and DREAL (Direction Régionale, de l'Environnement de l'Aménagement et du Logement = Regional Department of the Environment, Planning and Housing), DREAL provides a danger analysis, plant request support.
- **Session D:** Alert call from victims, dispatch of first means, request activation.
- **Session E:** Victims' extraction: AMP to Hospital, integration of Italian red cross in the operations, triage of victims and transport to hospital.
- **Session F:** activation of EUCPM for aerial means.

Table 4.8 provides an overview of the assignment of test scenarios to the sessions of Trial 2. The identifiers of these test scenarios are prefixed by TS (test scenario) and in some cases DR-1 (for Dry-Run 1). Detailed descriptions of the test scenarios of Trial 2 can be found in Annex 5.

Table 4.8: Assignment of test scenarios to sessions of Trial 2

Test scenario	Relevant to Session
TS DR-1-03 (test scenario of Dry-Run 1 number 03): LifeX COP symbology compliant with Valabre symbology.	A, All
TS DR-1-01 Call and opening of incident.	A
TS DR-1-02 Arrival of Incident Commander (IC) & request for reinforcement using CrisisSuite.	A, B
TS DR-1-04 Updating of the incident location on COP (before SiTac arrives).	A
TS 0: first SiTac into the LifeX COP.	A
TS DR-1-05 New SiTac is sent by Asphodèle to LifeX COP.	B, All
TS 4d: FCP sends SITAC update to COP.	B, All
TS 5: Sharing of hazard contour.	B, All
TS1: Additional LifeX COP actors enter in scene.	C
TS DR-1-08: DREAL CrisisSuite uploads GP Form.	C
TS 2: DREAL can use CrisisSuite internally between different levels (SITREPS).	C
TS DR-1-07: DREAL CrisisSuite event form (input for risk analysis).	C
TS 4c: DREAL sends new SITREP to LifeX COP.	C
TS DR-1-09: Dispatching and routing of ambulance.	D, E
TS 4a: EHS HQ reports on victims' status.	E
TS 6: Request for aerial support from ERCC.	F
TS 3: Detection of information of interest through social media.	B, D
TS 4b: Social media manager send tweets of interest to COP.	B, D
TS DR-1-06: Social media summary report to FCP.	B, D

Table 4.9 provides an overview of the testcases and test results achieved in DR-1 and DR-2. All test scenarios could be successfully tested either in DR-1 or in DR-2.

Table 4.9: Testcases and test results achieved in DR-1 and DR-2

TC	Title	Messages #	DR-1	DR-2
TS-DR-1-01	Call and opening of incident.	-		OK
TS DR-1-02	Arrival of Incident Commander (IC) & request for reinforcement using CrisisSuite.	7		OK
TS DR-1-04	Updating of the incident location on COP (before	7		OK

TC	Title	Messages #	DR-1	DR-2
	SiTac arrives).			
TS0	first SiTac into the LifeX COP.	8	OK	
TS DR-1-05	New SiTac is sent by Asphodèle to LifeX COP.	8		OK
TS DR-1-06	Social media summary report to FCP.	9-5		OK
TS DR-1-07	DREAL CrisisSuite event form (input for risk analysis.	7		OK
TS DR-1-08	DREAL CrisisSuite uploads GP form.	-		OK
TS6	Request for aerial support from ERCC.			OK
TS DR-1-09	Dispatching and routing of ambulance.	8-4-3		OK
TS DR-1-03	LifeX COP symbology compliant with Valabre symbology.	-		OK
TS1a	Additional LifeX COP actors enter in scene.	-	OK	OK
TS1b	Additional LifeX COP actors enter in scene.		KO	OK
TS2	DREAL can use CrisisSuite internally between different levels (SITREPS).	-	OK	
TS3	Detection of information of interest through social media.	1-10	OK	
TS4a	EMS HQ reports on victims' status.	2	POK	OK
TS4b	Social media manager sends tweets of interest to COP.	1-10	OK	OK
TS4c	DREAL sends new SITREP to LifeX COP.	6	OK	OK
TS4d	FCP sends SITAC update to COP.	7	OK	OK
TS5 (hazard)	Sharing of hazard contour.	4	OK	OK
TS5 (route)	Sharing of hazard contour.	4	POK	OK
TS DR-1-10	EMS is warning CODIS there are victims.	-		OK
TS DR-2-11	CS is able to generate SITREPS (typical report message, event form, ERCC request).	-		OK

5. Trial independent Test-bed integration of internal solutions

Independently from the Trials and in parallel to their preparation and execution, the DoW indicates that all DRIVER+ internal solutions included and described in the PoS must be integrated in the Test-bed. This shall be done for the sake of readiness in case some of these solutions will be selected in future Trials. Another aspect for this integration is that it will enable an easier evaluation of its Crisis Management functions also after the end of the DRIVER+ project.

For these reasons a time plan for the integration of these internal solutions (independent from the Trials) was drafted.

5.1 Solution integration time plan

With the start of DRIVER+ and the related technical work packages, the development of the Test-bed had a very high priority as the Test-bed provides the major services for the integration of DRIVER+ solutions and for the preparation of the Trials. The integration of the DRIVER+ solutions in the first year was mainly driven by the first two Trials. The integration of these already consumed a big amount of the planned time from the technical project partners. An additional challenge in the early phase of the Test-bed was the fact that during integration some weak spots and bugs were becoming visible in the Test-bed and the related bug-fixing consumed considerable – not well foreseeable - amounts of time. In the period after Trial 2 the Test-bed was still under continuous development and the solutions integration remained challenging. It was decided that not only for the solutions selected for Trials but also for all remaining DRIVER+ internal solutions a plan shall be created with the timeline for their integration into the Test-bed. Having all DRIVER+ internal solutions integrated in the Test-bed would provide the following advantages:

- During a Trial preparation phase the timespan between the solution selection and the Dry Run 1 is usually very short. Any preparation of a solution in advance (being already integrated into the Test-bed) would definitely support the progress in this phase
- Integration of all DRIVER+ internal solutions – regardless of their participation in Trials – is one of the project targets. Therefore, not only the Trials shall be considered as trigger to start their integration but the general target to create a comprehensive Test-bed shall be achieved.

Early June 2018 the inputs for the basic integration testing steps were collected from all solution providers. The information is summarized in Table 5.1 below containing the selection of Test-bed adapter types, data exchange formats and the complexity of the integration.

Table 5.1: Test-bed integration plan

Project Partner	Solution	Responsible person for tech integration to Test-bed	Data exchange formats	Test-bed adapter	Estimated Complexity: (5 is highest)	Q2 2018 (Trial 1)	Q2 2018	Q3 2018	Q4 2018	Q1 Q2 2019
TCS	SMAP (Social Media Analysis Platform)	antoine.leger(at)thalesgroup.com	CAP	REST adapter	2		"Hello world" integration DONE	Full integration		
FRQ	LifeX COP	Thomas.obritzhauser(at)frequentis.com	CAP, EMSI, KML, KMZ, MLP	JAVA adapter	3		"Hello world" integration DONE	Full integration		
AIT	Crowd Tasker	Sebastian.Sippl(at)ait.ac.at	CAP, EMSI	Node.Js adapter	Not estimated			"Hello world" integration	Full integration	
AIT	Rumor Debunker	Joachim.Klerx(at)ait.ac.at	CAP, EMSI	JAVA adapter	Not estimated				"Hello world" integration	Full integration
ITTI	PROCEED	grzegorz.taberski(at)itti.com.pl	EMSI	JAVA adapter	3			"Hello world" integration	Full integration	Full integration
GMV	Socrates OC	rvalencia(at)gmv.com	GeoJSON, large data update, map layer update	JAVA adapter	3	Done (Trial 1)				
WWU	GDACSmobile	michael.middelhoff(at)wi.uni-muenster.de	CAP	REST adapter	2			"Hello world" integration	Full integration	
WWU	HumLog	michael.middelhoff(at)wi.uni-muenster.de	KML (to be implemented)	Java, (Rest) adapter	4			"Hello world" integration	Full integration	
ARMINES	IO-DA	aurelie.conges(at)mines-albi.fr	JSON or XML (needs adaptation based on the needs	JAVA adapter or REST	4				"Hello world" integration	Full integration

Project Partner	Solution	Responsible person for tech integration to Test-bed	Data exchange formats	Test-bed adapter	Estimated Complexity: (5 is highest)	Q2 2018 (Trial 1)	Q2 2018	Q3 2018	Q4 2018	Q1 Q2 2019
			of the Test-bed)	adapter						
EDI	PRoTect	Antonio.Chagas(at)edisoft.pt	EMSI	JAVA adapter or REST adapter	4				"Hello world" integration	Full integration
DLR	Airborne and Terrestrial Situation Awareness	eric.neidhardt(at)dlr.de	JSON (needs an adaptation based on the needs of the Test-bed)	JAVA adapter or REST adapter	5			"Hello world" integration	Full integration	
DRC	Scenario based psychological First Aid (PFA) training	Pia Tingsted Blum piblu(at)rodekors.dk	Text entries via observer tool	via Observer tool	2				"Hello world" integration	Full integration
DWR	Debris Tool	aiden(at)urplatform.eu cedric(at)urplatform.eu	json	Node.Js adapter	Not estimated				"Hello world" integration	Full integration
MDA	MDA C2	itamarl(at)mda.org.il rand(at)mda.org.il	C#	C# adapter	3		"Hello world" integration DONE	Full integration		

Each solution integration consists of the first step called "hello world" (Step 0) integration and the full integration with a more complex data exchange between the solution and the Test-bed (both steps are described in detail in **D934.21** (4)).

The "hello world" (Step 0) integration does not need a concrete Trial scenario and it only enables a basic data exchange between a solution and the Test-bed. To achieve higher levels of integration, dedicated "solution scenarios" will be introduced with "related solutions". The related solutions are solutions of the PoS database with good potential for data exchange. The solution scenarios and related test cases will be necessary in order to integrate those solutions deeper into the Test-bed.

The priority for the solution integration was influenced by

- Their use in the Trials.
- The used Test-bed adapter types.
- The estimated complexity of integration.
- The availability of the technical staff on the side of the solution providers.

As an overall target it was envisaged that the integration of all DRIVER+ internal solutions (if not already done for Trial 1 or already started for Trial 2) shall start during the second half of 2018 and the last steps of the Test-bed integration were targeted to be finished by mid of 2019. Table 5.2 shows the time plan for the solution integration in a more graphical view (yellow=1st step of integration, green=2nd step of integration).

Table 5.2: Time plan solution integration

	Q1 2018	Q2 2018	Q3 2018	Q4 2018	Q1 Q2 2019
SMAP					
LifeX COP					
Crowd Tasker					
Rumor Debunker					
PROCEED					
Socrates OC					
GDACSmobile					
HumLog					
IO-DA					
PRoTect					
Airborne and Terrestrial Situation Awareness					
PFA training					
Debris					
MDA C2					

A specific challenge was how to "integrate" non-technical solutions into the Test-bed like the "Scenario based psychological First Aid (PFA) training", provided by consortium partner DRC. It was decided that the results of such a training (which consists of answers from interviewed persons) shall be logged via the observer tool of the Test-bed. This would provide an option for exporting and processing the results from the PFA training into other solutions. The integration of PFA is ongoing.

Besides the 14 internal solutions of DRIVER+ partners the target was to enable also external solutions to be selected for specific Trials. As external solution providers always had to apply for participation to Trials and needed to undergo a selection process, their integration into the Test-bed could always be started only after a successful selection process before each Trial. For this reason, the effort for the integration process of external solutions was not predictable in a long-term forecast.

5.2 Solution integration progress for internal solutions not yet selected for Trials

For a faster integration into a future Trial environment the solutions will have to be integrated into the DRIVER+ Test-bed first. Exchanging data with the Test-bed is a first important step which will support the communication with other solutions via the Test-bed at a later stage and thus speed up the technical preparation process in future Trials.

For the DRIVER+ internal solutions not yet selected for Trials the status of the integration process into the Test-bed, the necessary adaptations, and the testing progress is currently (until 01/11/2018) done as described in the following sections.

5.2.1 Crowd Tasker

Crowd Tasker is a solution provided by DRIVER+ partner AIT.

CrowdTasker uses the NodeJS adapter to connect to the Test-bed.

The following back-end adaptations have been performed on the solution for the Test-bed integration:

- Installation of DRIVER+ typescript adapter on the CrowdTasker backend:
    - Implementation of wrapper objects for the Test-bed adapter.
    - Implementation of the logic to forward CAP messages from Kafka Test-bed to the internal zero MQ (Message Queue) interface.
- Implementation of the logic to forward CAP messages from the internal zero MQ interface to Kafka Test-bed.
- Creation of a configuration on the back-end to embed wrapper objects.
- Creation of NodeJS/Gulp Script for DRIVER+ to load the respective configuration.
- Adaptation of the existing mapping for incoming EMSI messages (external data) from XML/JSONIX format to the DRIVER+ AVRO schema.
- Amendments/corrections of AVRO schema for EMSI messages (optional fields instead of mandatory fields).

The following front-end adaptations have been performed on the solution for the Test-bed integration:

- Amendments/corrections to “external data hub” to display CAP messages.
- Amendments/corrections in the reporting front-end to send a CAP “Hello World”-message to the DRIVER+ Gateway.

The following IT infrastructure related work has been performed during the Test-bed integration process:

- Opening of the respective firewall ports to access the online Test-bed.
- Installation of the local DRIVER+ Test-bed on a virtual machine (VM) while the online Test-bed was not reachable.

The testing process of the Test-bed integration consists of the following steps:

- Load the back-end adapter with the configuration.
- Create a report for a CrowdTasker event.
- Send the report (“Hello World”) in EMSI/CAP format.
- Check in the console-log if the Kafka adapter has sent successfully.
- Check in the console-log if the Kafka adapter has received successfully.
- Check on the “external data hub” if the message is displayed.

- Copy the received string from console-log, amend the string with polygon-data, re-send the updated string via Kafka.
- Create a CrowdTasker event from the received EMSI messages with polygon and repeat the process.

All tests have been performed successfully.

5.2.2 Airborne and Terrestrial Situation Awareness

Airborne and Terrestrial Situation Awareness is a solution provided by DRIVER+ partner DLR.

A local Test-bed instance was deployed and step 0 of the Test-bed integration process was performed. The normally used KeepOperational web-front-end was substituted with an adapter so that the Test-bed can access the underlying services directly. Currently the adapter supports the isochronous routing as well as the regular routing. The messages are in a customized schema because they are quite solution specific and meant to substitute the user interface.

Next steps:

For the Trial specific Test-bed integration (step 1) a new connection to the adapter has been added so that the Test-bed can update the flood mask used for routing. Furthermore, existing map data (e.g. OpenStreetMap) for the area of The Hague is obtained and added to the DLR database.

The following solution adaptions have been performed and are planned:

For the Test-bed integration a new adapter was created so that the Test-bed can directly connect to the underlying services of the Airborne and Terrestrial Situation Awareness solution. For the Trial specific implementation DLR will add a Dutch translation to the front-end of the solution.

The following two test cases have been performed to verify a successful step 0 with regard to Test-bed integration:

- Test case: connecting routing component to the Test-bed.
  - Start adapter with back-end of KeepOperational-scenario Magdeburg and Test-bed configured.
  - Adapter produces a test event for a routing request
(format: keep_operational_routing_request-value.avsc).
  - Adapter subscribes to the topic keep_operational_routing_request.
  - Adapter sends a test event on the topic keep_operational_routing_request.
  - Adapter receives a test event.
  - Adapter verifies the integrity of the received event.
  - Adapter sends routing request to the back-end of KeepOperational-scenario Magdeburg.
  - Adapter receives routing information from back-end.
  - Adapter verifies that no error has occurred.
  - Adapter send routing information to Test-bed
(topic: keep_operational_routing_response,
format: keep_operational_routing_response-value.avsc).
  - Verify that the routing information was received in the Test-bed topic browser.
- Test case: connecting isochrone routing component to the Test-bed.
  - Start adapter with back-end of KeepOperational-scenario Magdeburg and Test-bed configured.
  - Adapter produces a test event for an isochrone routing request
(format: keep_operational_isochrone_request-value.avsc).
  - Adapter subscribes to the topic keep_operational_accessibility_request.

  - Adapter sends a test event on topic keep_operational_accessibility_request.
  - Adapter receives the test event.
  - Adapter verifies integrity of the received event.
  - Adapter sends isochrone request to back-end of KeepOperational-scenario Magdeburg.
  - Adapter receives isochrone information from back-end.
  - Adapter verifies that no error has occurred.
  - Adapter sends isochrone information to Test-bed (topic: keep_operational_accessibility_response, format: keep_operational_isochrone_response-value.avsc).
  - Verify that accessibility information was received in Test-bed topic browser.

The following test case has been performed with the solution to verify a successful step 1 with the Test-bed integration:

- Test case: connecting net-restriction component to the Test-bed.
  - Start adapter with back-end of KeepOperational-scenario Magdeburg and Test-bed configured.
  - Adapter produces a test event for creation of a new net-restriction (format: keep_operational_net_restriction_create-value.avsc).
  - Adapter subscribes to topic keep_operational_net_restriction_create_request.
  - Adapter sends a test event on topic keep_operational_net_restriction_create_request.
  - Adapter receives the test event.
  - Adapter verifies integrity of the received event.
  - Adapter sends a request for a new net-restriction to the back-end of KeepOperational-scenario Magdeburg.
  - Adapter receives response from back-end, containing all currently active net-restrictions.
  - Adapter verifies that the new net-restriction is in the response.
  - Adapter sends information on net-restrictions to Test-bed (topic: keep_operational_net_restriction_response, format: keep_operational_net_restriction_response-value.avsc).
  - Adapter produces a test event for deletion of the newly created net-restriction (format: keep_operational_net_restriction_delete-value.avsc).
  - Adapter subscribes to topic keep_operational_net_restriction_delete_request.
  - Adapter sends test event on topic keep_operational_net_restriction_delete_request.
  - Adapter receives test event.
  - Adapter verifies integrity of received event.
  - Adapter sends request to delete the net-restriction to back-end of KeepOperational-scenario Magdeburg.
  - Adapter receives response from back-end, containing all currently active net-restrictions.
  - Adapter verifies that the new net-restriction is no longer in the response.
  - Adapter sends information on net-restrictions to Test-bed (topic: keep_operational_net_restriction_response, format: keep_operational_net_restriction_response-value.avsc).
  - Verify that information on net-restrictions was received in Test-bed topic browser.

5.2.3 HumLog

HumLog is a solution provided by DRIVER+ partner WWU. The HumLog solution uses the Java adapter to connect to the Test-bed.

HumLog was integrated using the online Test-bed and is already able to send and receive messages from/to the Test-bed. For testing purposes, a local Test-bed installation was installed first, this was achieved using the Docker environment and the provided software components from the Test-bed repository. Using a local Test-bed required some changes to the configuration of the HumLog solution in order to address the local distribution.

IT/Software related issues:

The integration required minor changes to version numbers in the Java Maven[17] configuration of the adapter which might be due to the current development state. Since the Java-based solution cannot import Maven projects, an export of all dependencies was needed and a direct import to the solution. This required a manual update with every update of the Adapter. The configuration files containing the connection address and other settings are not directly accessible in this solution which requires further adaptation.

Overall the integration to the Test-bed was achieved with a successful communication. Further adaptions to a Trial scenario and relevant data exchanges will be performed after the solution will be selected for a Trial.

The testing process for HumLog consisted of the following steps:

- Connect the adapter using the configuration for the Test-bed connection.
- Create a dummy solution to answer to proper solution messages.
- Create custom schema using Kafka schema browser.
- Send routing information as test case in CAP and custom format.
- Check console-log of the Kafka adapter if message was sent.
- Check console-log of the Kafka adapter if the message was received.
- Check if the dummy solution receives HumLog message via Test-bed.
- Send answer message using CAP and custom format.
- Check if the solution has received messages from Test-bed.

All test steps have been performed successfully.

5.2.4 GDACS mobile

GDACS mobile is a solution provided by DRIVER+ partner WWU. The GDACS mobile solution uses the REST adapter to connect to the Test-bed. GDACS Mobile currently is under development and therefore not in a stable condition to be integrated into the Test-bed. The initial plan to integrate the solution until the end of the fourth quarter of 2018 was postponed for this reason.

5.2.5 PROCeed

PROCeed is a simulation solution provided by DRIVER+ partner ITTI. This solution enables its users to simulate the cascading effects and predict future events based on current situations. PROCeed handles a set of object classes which have their capabilities and a set of rules which must be defined. They are not based on the individual objects but on the classes of objects. Each rule connects change of object capabilities, user interactions and events and leads to consequences. Those consequences can change the capabilities of some objects, create an event or have other impacts on the future in the simulated world.

[17] Maven is a build automation tool used primarily for Java projects

To use PROCeed in the integrated environment the synchronization of the state of the objects (their capabilities) is needed. This synchronization must be made in the moment from which the user would like to see the future consequences. After that synchronization the PROCeed laboratory enables the user to see simulated future events based on the rules defined in the system. Those rules connect the capabilities of the objects.

The PROCeed solution uses the Java adapter to connect to the Test-bed. The integration process consists of the following steps:

- Setup of the local Test-bed environment using Docker.
- Setup of the Time Service.
- Connection to the time service using the Java Adapter.
- Collection of data from topics published on the Test-bed.

These steps were performed successfully and PROCeed is currently adapted in order to handle data collected from the Test-bed. The PROCeed laboratory solution is adapted to be able:

- To use the time service to adjust the time in the application.
- To collect EMSI messages describing the state of the objects in the simulation and transfer those data to the internal database of the PROCeed laboratory which shall include also a translation mechanism.

Currently the scope of EMSI messages handling is under discussion since the PROCeed integration is made independently of the DRIVER+ Trials.

During the integration process the following issues/problems were detected:

- Not up-to-date library used in Test-bed implementation.

6. Conclusions

This document has presented the main activities related to the integration of solutions into the Test-bed, focusing on the adaptations and activities which were required to prepare and integrate the solutions for Trial 1 and Trial 2. Deliverable **D934.32** ***DRIVER+ Solution scenarios and integration test results v2*** (1) (due M65) will report about the corresponding activities performed for Trial 3 and Trial 4. Whilst in Trial 1 the overall integration of the involved solutions was still limited due to the very short time for integration works and due to the status of the Test-bed, Trial 2 achieved a good level of integration, especially with regards to the amount of time available after having the requirements for integration frozen (which result from the final Trial scenario).

Overall goal of the solution integration is that crisis relevant information is presented to practitioners in the most suitable and comfortable way considering the challenges they face during their crisis response actions e.g. the information overload from many information sources with various data formats. An automated exchange of data among involved IT solutions shall avoid that practitioners manually collect data from several solutions or even enter some output data from solutions into other solutions for further data processing. The solution integration leads to an aggregation of subsystems which cooperate in an ideal way so that the system can deliver an overarching functionality.

During preparation and conduction of the Trials there was an automated exchange of data performed between different IT solutions and there is a tendency visible that integration of solutions (once it will be performed in an operational environment with fully mature products) will lead to

- Less time needed for practitioners in their search for crisis relevant information.
- Lower probability for wrong information caused by human errors while reading/entering data from/into a solution.
- More time left for practitioners to analyse and interpret the information, and to define, communicate, execute and supervise crisis response actions.
- Higher quality of the Crisis Management outcome due to the time savings, better data quality and improvement of communication.

It has to be mentioned that many solutions used in the Trials were prototypes and thus do not have the maturity level of a product. A direct comparison between the mature products currently used by practitioners and the new DRIVER+ solutions cannot be objective for this reason.

However, there are several lessons learned which also reveal some aspects to be improved:

- The concept of having two Dry Runs before each Trial has proven to be necessary and valuable. The interaction between a scenario and the involved solutions requires those physical meetings (on top of weekly telephone conferences and remote testing) where the major actors prepare the Trial and iteratively improve scenario details and technical integration details.
- The Dry Runs did not provide enough time to perform all tests at a timescale corresponding to the original Trial scenario. Some tests were shortened for this reason but did not produce the same behaviour of solutions as detected later at original scenario length. This caused problems which were only detected during the execution of Trial 1. In the future a Dry Run 2 should perform all tests at original length as they are planned for the Trials.
- Even though the solution trainings were performed during Dry Run 2 and again during the week of the Trials, they were often perceived to be too short by the practitioners. To profit from all the benefits a solution could offer, even more focus shall be put on the solution trainings and correspondingly more time should be foreseen (minimum one hour net training time per solution).
- In general, the Trial 1 and Trial 2 schedules were perceived as too tight from the solution integrator perspective - one month between the initial acceptance of solutions and Dry Run 1, and three weeks between Dry Run 1 and Dry Run 2 might be not enough in many cases.

- The process for the descriptions of user stories and information workflow diagrams was not defined well enough for external solutions and for those internal solutions which did not take part in the Trials. This has put a special challenge on their integration into the Test-bed as information workflow diagrams represent the main requirements for solution integration.
- Required adaptations were very demanding and not validated enough by detailed test cases.
- An earlier definition of the Trial scenario, research questions, data collection and evaluation plans and the needed adaptions of the solutions are required. Only minor modifications should be allowed during the rest of the process in order not to endanger already achieved progress.
- More information needs to be distributed to solution providers in the Call for Applications or during the initial acceptance of solutions. The information about the required interoperability capabilities of solutions needs to be defined better and more detailed information about the Test-bed has to be provided in advance. This mainly impacted external solution providers, and had the following consequences:
    - The formal commitment by external solution providers (and so their start of the work) was not achieved until one week before Dry Run 1 of Trial 1 (this led to an even tighter schedule).
    - External solution providers were not familiar with the whole concept of the Test-bed and it turned out that a complete “introduction into DRIVER+” was needed before any steps for integration of their solutions could be taken. External solution providers needed to understand the concept of the Trials and the Test-bed before concrete steps could be taken to integrate their solutions.
    - External solution providers needed longer than expected to prepare the integration in the required timeline.
- The solution integration and validation process for approval at Dry Run 1 and Dry Run 2 needs to be defined and monitored in a more efficient way. It must be also realistic and achievable, given the corresponding Trial timeframe.
- The solution providers should be informed well in advance about:
    - The preliminary technical and interoperability requirements for participating in Trial 1 (i.e. the requirements to connect to the Test-bed and the kind and format of the data to be exchanged.
    - The adaptions to be made on their solutions (if any), as required by the Trial Owner or practitioners.
    - The deadline for the readiness of solutions.
- It is highly desirable to set up an infrastructure, which allows remote testing at least between the initial acceptance of the solutions and the Dry Run 2.
- Specific checklists should be elaborated for the main milestones of the Trial process (e.g. the Dry Runs).

References

1. **DRIVER+ project.** *D934.32 - Report on the Solution scenarios and integration test results.* 2019.

2. —. *D923.11 - Report on thr Functional Specification of the Test-Bed Infrastructure.* 2018.

3. —. *D923.21 - Report on the Test-Bed reference Implementation.* 2018.

4. —. *D934.21 - Solution Testing Procedure.* 2018.

5. —. *D943.11 - Report on Trial Action Plan - Trial 1.* 2018.

6. —. *D944.11 - Report on Trial Action Plan - Trial 2.* 2018.

7. —. *D944.12 - Report on Trial Evaluation- Trial 2.* 2018.

8. **DRIVER+ consortium.** *Description of Work - DRiving InnoVation in crisis management for European Resilience.* 7th Framework Programme of the European Commission. 2017. Grant agreement no.: 607798..

9. **ISO.** *Societal security - Emergency management - Message structure for exchange of information.* ISO. 2015. ISO/TR 22351:2015.

10. **DRIVER+ project.** *Test-bed specification.* 2018. https://driver-eu.gitbook.io/test-bed-specification.

11. —. *Test-bed design.* 2018. https://driver-eu.gitbook.io/test-bed-design.

12. **GeoServer.** *GeoServer for sharing geospatial data.* http://geoserver.org/.

13. **PostgreSQL.** *Open Source Relation Database: Datastorage.* https://www.postgresql.org/.

14. **JAVA-Technologie.** *Development Language and Runtime environment.* https://java.com/en/.

15. **OpenLayers.** *OpenLayers high-performance library for mapping needs.* https://openlayers.org/.

16. **The Open Source Geospatial Foundation (OSGeo).** *GeoTIFF.* 2018. https://trac.osgeo.org/geotiff/.

17. **The Open Geospatial Consortium (OGC).** *Web Map Service.* 2018. http://www.opengeospatial.org/standards/wms.

18. **Web Feature Service.** 2018. http://www.opengeospatial.org/standards/wfs.

19. **The Apache Software Foundation.** *Apache Kafka - A distributed streaming platform.* 2017. https://kafka.apache.org/.

20. **DRIVER+ project.** *The Java Test-bed adapter.* 2018. https://github.com/DRIVER-EU/java-test-bed-adapter.

21. **The European Commission (EC).** *EUR-Lex - 2014/762/ EU Commission Implementing Decision of 16 Oct. 2014 laying down rules for implementation of Decision No 1313/2013/EU of the European Parliament and the Council on Union Civil Protection Mechanism and repealing.* 2014. Commission Decisions 2004/277/EC, Euratom and 2007/606/EC, Euratom (notified under document C(2014) 7489) Text with EEA relevance - https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=uriserv%3AOJ.L_.2014.320.01.0001.01.ENG.

22. **Docker Inc.** 2018. https://www.docker.com/.

23. **DRIVER+ project.** *D946.12 - Report on Trial Evaluation - Trial 4.* 2019.

24. —. *D945.12 - Report on Trial Evaluation - Trial 3.* 2019.

25. —. *D943.12 - Report on Trial Evaluation - Trial 1.* 2018.

Annexes

Annex 1 – DRIVER+ Terminology

In order to have a common understanding within the DRIVER+ project and beyond and to ensure the use of a common language in all project deliverables and communications, a terminology is developed by making reference to main sources, such as ISO standards and UNISDR. This terminology is presented online as part of the Portfolio of Solutions and it will be continuously reviewed and updated[1]. The terminology is applied throughout the documents produced by DRIVER+. Each deliverable includes an annex as provided hereunder, which holds an extract from the comprehensive terminology containing the relevant DRIVER+ terms for this respective document.

Table A1.1: DRIVER+ Terminology

Terminology	Definition	Source
Crisis management function	Crisis management functions aim at achieving effects, e.g. coordination, a direction of effort, shared awareness, etc., in a crisis management system-of-systems. The “function” focuses on what is to be achieved, not how or by whom. Several systems, tools, building blocks, etc. may individually or in concert deliver a given function and, conversely, may support several different functions. Crisis management functions are grouped in three functional areas: operational (protection, response, recovery), preparatory (mitigation, capability development, strategic adaptiveness) and common (security management, logistics, C3, comms & Info management).	Initial DRIVER+ definition
Dry run 1	First rehearsal of a Trial, focusing on the technical integration of solutions, reference implementation of the Test-bed, and scenario validation; it also serves as a readiness review to approve the maturity of technical solutions.	Initial DRIVER+ definition
Dry run 2	Full scale rehearsal of a Trial without external end-users participation, aimed at detection of technical issues and last second fine-tuning; Dry Run 2 is organised as a complete mirror of the Trial.	Initial DRIVER+ definition
Gap	Gaps between the existing capabilities of responders and what was actually needed for effective and timely response.	Project Responder 5
Interoperability	The ability of diverse systems and organisations to work together, i.e. to interoperate.	ISO 22397

[1] The Portfolio of Solutions and the terminology of the DRIVER+ project are accessible on the DRIVER+ public website (https://www.driver-project.eu/). Further information can be received by contacting coordination@projectdriver.eu.

Terminology	Definition	Source
Legacy systems	(Crisis management) system currently in operational use.	Initial DRIVER+ definition
Operator	(Human) operator Person engaged in task performance, considered as a monitoring, controlling or directing element in a system or process capable of a dynamic response to system inputs and disturbances.	ISO 9996:1996(en) Mechanical vibration and shock — Disturbance to human activity and performance — Classification, 3.5
Portfolio of Solutions (PoS)	A database driven web site that documents the available Crisis Management Solutions. The PoS includes information on the experiences with a solution (i.e. results and outcomes of Trials), the needs it addresses, the type of practitioner organisations that have used it, the regulatory conditions that apply, societal impact consideration, a glossary, and the design of the Trials.	Initial DRIVER+ definition
Test-bed	The software tools, middleware and methodology to systematically conduct Trials and evaluate solutions within an appropriate environment. An “appropriate environment” is a testing environment (life and/or virtual) where the trialling of solutions is carried out using a structured, all-encompassing and mutual learning approach. The Test-bed can enable existing facilities to connect and exchange data, providing a pan-European arena of virtually connected facilities and crisis labs where users, providers, researchers, policy makers and citizens jointly and iteratively can progress on new approaches or solutions to emerging needs.	Initial DRIVER+ definition

Annex 2 – Trial 2 – integration step 0 Test-bed messages

LifeX COP

Table A2.1: Trial 2 - integration's step 0 Test-bed messages of LifeX COP

Test-bed messages	
Emisor	LifeX COP


```
{
    "topic": "standard_cap",
    "key": {
      "distributionID": "lifex_cop-27",
      "senderID": "lifex_cop",
      "dateTimeSent": 1528360023889,
      "dateTimeExpires": 1528360023889,
      "distributionStatus": "Unknown",
      "distributionKind": "Unknown"
    },
    "value": {
      "identifier": "a1b68e3c-21c0-4436-8821-fe8f8e498767",
      "sender": "COP",
      "sent": "2018-06-07T09:44:22+02:00",
      "status": "Test",
      "msgType": "Alert",
      "source": null,
      "scope": "Public",
      "restriction": null,
      "addresses": null,
      "code": null,
      "note": null,
      "references": null,
      "incidents": null,
      "info": {
        "eu.driver.model.cap.Info": {
          "language": null,
          "category": {
            "eu.driver.model.cap.Category": "Geo"
          },
          "event": "AlertEVENT",
          "responseType": null,
          "urgency": "Unknown",
          "severity": "Unknown",
          "certainty": "Unknown",
          "audience": null,
          "eventCode": null,
          "effective": null,
          "onset": null,
          "expires": null,
          "senderName": {
            "string": "3020 LifeX COP"
          },
          "headline": {
            "string": "asd"
          },
          "description": {
            "string": "asdf"
          },
          "instruction": {
            "string": ""
          },
          "web": null,
          "contact": null,
          "parameter": null,
          "resource": null,
          "area": {
            "eu.driver.model.cap.Area": {
              "areaDesc": "Area",
              "polygon": {
                "string": "51.25761789054249,6.1248779296875 51.17417731875821,6.12762451171875
```

```
51.18450806106816,6.4654541015625 51.23612705252839,6.455841064453124 51.294558643257886,6.2786865234375
51.25761789054249,6.1248779296875"
              },
              "circle": null,
              "geocode": null,
              "altitude": null,
              "ceiling": null
            }
          }
        }
      }
    },
    "partition": 0,
    "offset": 0
  }
```


MDA C2

Table A2.2: Trial 2 - integration's step 0 Test-bed messages of LifeX COP

Test-bed messages	
Emisor	MDA C2


```
{
    "topic": "csharp-test",
    "key": {
      "distributionID": "9abffbd8-8df5-41e8-a62e-e8913d721c1f",
      "senderID": "CSharpTest-bedAdapter",
      "dateTimeSent": 63663286028763,
      "dateTimeExpires": 63663286088763,
      "distributionStatus": "Unknown",
      "distributionKind": "Unknown"
    },
    "value": {
      "sender": "CSharpExampleProducerCustom",
      "message": ""
    },
    "partition": 0,
    "offset": 0
  },
```


SMAP

Table A2.3: Trial 2 - integration's step 0 Test-bed messages of LifeX COP

Test-bed messages	
Emisor	SMAP


```
{
    "topic": "standard_cap",
    "key": {
      "distributionID": "SMAP_at_THALES-18",
      "senderID": "SMAP_at_THALES",
      "dateTimeSent": 1528451096697,
      "dateTimeExpires": 1528451096697,
      "distributionStatus": "Unknown",
      "distributionKind": "Unknown"
    },
    "value": {
      "identifier": "SMAP_1",
      "sender": "smap(at)driver.eu",
      "sent": "2018-06-08T10:15:27-07:00",
      "status": "Exercise",
      "msgType": "Alert",
      "source": null,
```

```
      "scope": "Public",
      "restriction": null,
      "addresses": null,
      "code": null,
      "note": null,
      "references": null,
      "incidents": null,
      "info": {
        "eu.driver.model.cap.Info": {
          "language": null,
          "category": {
            "eu.driver.model.cap.Category": "Geo"
          },
          "event": "Tweet",
          "responseType": null,
          "urgency": "Past",
          "severity": "Unknown",
          "certainty": "Observed",
          "audience": null,
          "eventCode": null,
          "effective": null,
          "onset": null,
          "expires": null,
          "senderName": {
            "string": "Social Media Analysis Platform from DRIVER+ Trial2"
          },
          "headline": {
            "string": "flood eruption warning"
          },
          "description": {
            "string": "#Flood risk analysis &amp; flood warning systems for #Malaysia - #Ambiental
https://t.co/TslZCVXLeB https://t.co/3bNb45ihCT"
          },
          "instruction": null,
          "web": {
            "string": "http://twitter.com/!/status/939187831372689408"
          },
          "contact": null,
          "parameter": null,
          "resource": null,
          "area": {
            "eu.driver.model.cap.Area": {
              "areaDesc": "flood",
              "polygon": null,
              "circle": {
                "string": "2.4999999348074198,112.5 0"
              },
              "geocode": null,
              "altitude": null,
              "ceiling": null
            }
          }
        }
      }
    },
    "partition": 0,
    "offset": 4
  },
```

CrisisSuite

Table A2.4: Trial 2 - integration's step 0 Test-bed messages of LifeX COP

Test-bed messages	
Emisor	CrisisSuite


```
{
    "topic": "standard_cap",
    "key": {
      "distributionID": "crisissuite-672",
      "senderID": "crisissuite",
      "dateTimeSent": 1527788395790,
      "dateTimeExpires": 1527788395790,
      "distributionStatus": "Unknown",
      "distributionKind": "Unknown"
    },
    "value": {
      "id": "crisissuite",
      "dateTimeSent": 1527788395790,
      "level": "DEBUG",
      "log": "User admin requested this test msg to be send at 2018-05-31 17:39:55.784773+00:00 using
server RBE-MSI-GE62",
     },
    "partition": 0,
    "offset": 0
  },
```

Annex 3 – Trial 2 – Dry Run 1 – test scenario's Test-bed messages

Test scenario 3

Test-bed messages sent by **SMAP** to **LifeX COP** in the test scenario 3:

Table A3.1: Test scenario TS3's Test-bed messages

Test-bed messages			
Emisor	SMAP	**Receptor**	LifeX COP


```
{
    "topic": "standard_cap",
    "key": {
      "distributionID": "SMAP_at_THALES-781",
      "senderID": "SMAP_at_THALES",
      "dateTimeSent": 1530091194794,
      "dateTimeExpires": 1530091194794,
      "distributionStatus": "Unknown",
      "distributionKind": "Unknown"
    },
    "value": {
      "identifier": "SMAP-648a3ac6-7cf3-4649-afe8-18873813151f",
      "sender": "smap-driver.eu",
      "sent": "2018-06-27T12:22:38+0200",
      "status": "Exercise",
      "msgType": "Alert",
      "source": null,
      "scope": "Public",
      "restriction": null,
      "addresses": null,
      "code": null,
      "note": null,
      "references": null,
      "incidents": null,
      "info": {
        "eu.driver.model.cap.Info": {
          "language": null,
          "category": {
            "eu.driver.model.cap.Category": "Geo"
          },
          "event": "Tweet",
          "responseType": null,
          "urgency": "Past",
          "severity": "Unknown",
          "certainty": "Observed",
          "audience": null,
          "eventCode": null,
          "effective": null,
          "onset": null,
          "expires": null,
          "senderName": {
            "string": "Social Media Analysis Platform from DRIVER+ Trial2"
          },
          "headline": {
            "string": ""
          },
          "description": {
            "string": "#NiUnaMenos: In #Argentina, women declare a general strike against all violence
against women https://t.co/lfhPdnyrUC"
          },
          "instruction": null,
          "web": {
            "string": "http://twitter.com/!/status/1004016543989223424"
          },
          "contact": null,
          "parameter": null,
          "resource": {
            "eu.driver.model.cap.Resource": {
              "resourceDesc": "Image",
```

```
              "size": null,
              "uri": {
                "string": ""
              },
              "mimeType": {
                "string": "image/jpeg"
              },
              "deferUri": null,
              "digest": null
            }
          },
          "area": {
            "eu.driver.model.cap.Area": {
              "areaDesc": "Argentina",
              "polygon": null,
              "circle": {
                "string": "-34.00000006891787,-64.0000000409782 0"
              },
              "geocode": null,
              "altitude": null,
              "ceiling": null
            }
          }
        }
      }
    },
    "partition": 0,
    "offset": 2
  },
```


Test scenario 4a

Test-bed messages sent by **MDA C2** to **LifeX COP** in the test scenario 4a:

Table A3.2: Test scenario TS4a's Test-bed message

Test-bed messages			
Emisor	MDA C2	**Receptor**	LifeX COP


```
{
    "topic": "standard_emsi",
    "key": {
      "distributionID": "035c06ac-8663-4740-9d25-ad0649f0483c",
      "senderID": "mda-c2",
      "dateTimeSent": 63665702031378,
      "dateTimeExpires": 63665702091378,
      "distributionStatus": "Unknown",
      "distributionKind": "Unknown"
    },
    "value": {
      "CONTEXT": {
        "ID": "d2f85e52-2bc4-40f6-bc3b-d6a13bd7d507",
        "MODE": "EXERCISE",
        "MSGTYPE": "INCIDENT",
        "CREATION": {
          "long": 63665712831277
        },
        "LINK": [],
        "LEVEL": null,
        "SECLASS": null,
        "FREETEXT": null,
        "URGENCY": {
          "string": "NOT_URGENT"
        },
        "ORIGIN": null,
        "EXTERNAL_INFO": []
      },
      "EVENT": {
        "eu.driver.model.emsi.EVENT": {
          "ID": "c868e580-bf43-4f6a-bb76-57a1cefb8280",
```

```
          "NAME": {
            "string": "forest fire"
          },
          "MAIN_EVENT_ID": null,
          "ETYPE": {
            "eu.driver.model.emsi.ETYPEEVENT": {
              "CATEGORY": [
                "/FIR"
              ],
              "ACTOR": [
                "/BEV/OTH"
              ],
              "LOCTYPE": [
                "/OTH"
              ],
              "ENV": []
            }
          },
          "SOURCE": null,
          "SCALE": null,
          "CERTAINTY": null,
          "DECL_DATIME": null,
          "OCC_DATIME": null,
          "OBS_DATIME": null,
          "STATUS": null,
          "RISK_ASSESSMNT": null,
          "REFERENCE": [],
          "CASUALTIES": [
            {
              "CONTEXT": "PRELIM_STAT",
              "DATIME": null,
              "DECONT": null,
              "TRIAGERED": {
                "string": "0"
              },
              "TRIAGEYELLOW": {
                "string": "0"
              },
              "TRIAGEGREEN": {
                "string": "0"
              },
              "TRIAGEBLACK": {
                "string": "0"
              },
              "MISSING": {
                "string": "0"
              }
            }
          ],
          "EVAC": [],
          "EGEO": [
            {
              "DATIME": null,
              "TYPE": "/GEN/INCGRD",
              "POSITION": {
                "LOC_ID": null,
                "NAME": null,
                "TYPE": {
                  "string": "POINT"
                },
                "COORDSYS": null,
                "COORD": [
                  {
                    "LAT": 43.29337,
                    "LON": 5.3713,
                    "HEIGHT": null
                  }
                ],
                "HEIGHT_ROLE": null,
                "ADDRESS": [
                  "marseille      "
                ]
              },
              "WEATHER": [],
```

```
              "FREETEXT": null,
              "ID": null,
              "STATUS": null
            }
          ],
          "CAUSE": null,
          "FREETEXT": null
        }
      },
      "RESOURCE": [],
      "MISSION": []
    },
    "partition": 0,
    "offset": 3
  },
  {
    "topic": "standard_emsi",
    "key": {
      "distributionID": "755b53a8-2061-44a4-a96e-d7d4c547921d",
      "senderID": "mda-c2",
      "dateTimeSent": 63665704755539,
      "dateTimeExpires": 63665704815539,
      "distributionStatus": "Unknown",
      "distributionKind": "Unknown"
    },
    "value": {
      "CONTEXT": {
        "ID": "720dab27-590a-4f88-9f90-e48a0d65652a",
        "MODE": "EXERCISE",
        "MSGTYPE": "INCIDENT",
        "CREATION": {
          "long": 63665715555443
        },
        "LINK": [],
        "LEVEL": null,
        "SECLASS": null,
        "FREETEXT": null,
        "URGENCY": {
          "string": "NOT_URGENT"
        },
        "ORIGIN": null,
        "EXTERNAL_INFO": []
      },
      "EVENT": {
        "eu.driver.model.emsi.EVENT": {
          "ID": "308e36e8-cf78-4b09-90f1-538bd4cdda0a",
          "NAME": {
            "string": ""
          },
          "MAIN_EVENT_ID": null,
          "ETYPE": {
            "eu.driver.model.emsi.ETYPEEVENT": {
              "CATEGORY": [
                "/FIR"
              ],
              "ACTOR": [
                "/BEV/OTH"
              ],
              "LOCTYPE": [
                "/OTH"
              ],
              "ENV": []
            }
          },
          "SOURCE": null,
          "SCALE": null,
          "CERTAINTY": null,
          "DECL_DATIME": null,
          "OCC_DATIME": null,
          "OBS_DATIME": null,
          "STATUS": null,
          "RISK_ASSESSMNT": null,
          "REFERENCE": [],
          "CASUALTIES": [
```

```
          {
            "CONTEXT": "PRELIM_STAT",
            "DATIME": null,
            "DECONT": null,
            "TRIAGERED": {
              "string": "3"
            },
            "TRIAGEYELLOW": {
              "string": "0"
            },
            "TRIAGEGREEN": {
              "string": "0"
            },
            "TRIAGEBLACK": {
              "string": "4"
            },
            "MISSING": {
              "string": "0"
            }
          }
        ],
        "EVAC": [],
        "EGEO": [
          {
            "DATIME": null,
            "TYPE": "/GEN/INCGRD",
            "POSITION": {
              "LOC_ID": null,
              "NAME": null,
              "TYPE": {
                "string": "POINT"
              },
              "COORDSYS": null,
              "COORD": [
                {
                  "LAT": 43.29337,
                  "LON": 5.3713,
                  "HEIGHT": null
                }
              ],
              "HEIGHT_ROLE": null,
              "ADDRESS": [
                "marseille       "
              ]
            },
            "WEATHER": [],
            "FREETEXT": null,
            "ID": null,
            "STATUS": null
          }
        ],
        "CAUSE": null,
        "FREETEXT": null
      }
    },
    "RESOURCE": [],
    "MISSION": []
  },
  "partition": 0,
  "offset": 4
}
```

Test scenario 4b

Test-bed messages sent by **SMAP** to **LifeX COP** in the test scenario 4b:

Table A3.3: Test scenario TS4b's Test-bed messages

Test-bed messages			
Emisor	SMAP	**Receptor**	LifeX COP


```
{
    "topic": "standard_cap",
    "key": {
      "distributionID": "SMAP_at_THALES-3199",
      "senderID": "SMAP_at_THALES",
      "dateTimeSent": 1530091194794,
      "dateTimeExpires": 1530091194794,
      "distributionStatus": "Unknown",
      "distributionKind": "Unknown"
    },
    "value": {
      "identifier": "SMAP-88948883-b50d-4a86-91cb-062012209de0",
      "sender": "smap-driver.eu",
      "sent": "2018-06-27T15:43:59+0200",
      "status": "Exercise",
      "msgType": "Alert",
      "source": null,
      "scope": "Public",
      "restriction": null,
      "addresses": null,
      "code": null,
      "note": null,
      "references": null,
      "incidents": null,
      "info": {
        "eu.driver.model.cap.Info": {
          "language": null,
          "category": {
            "eu.driver.model.cap.Category": "Fire"
          },
          "event": "Tweet",
          "responseType": null,
          "urgency": "Past",
          "severity": "Unknown",
          "certainty": "Observed",
          "audience": null,
          "eventCode": null,
          "effective": null,
          "onset": null,
          "expires": null,
          "senderName": {
            "string": "Social Media Analysis Platform from DRIVER+ Trial2"
          },
          "headline": {
            "string": ""
          },
          "description": {
            "string": "RT (at)morphin15: #incendie #paris ça fume !! http://t.co/Nxd48gKrSx"
          },
          "instruction": null,
          "web": {
            "string": "http://twitter.com/!/status/955597816629481474"
          },
          "contact": null,
          "parameter": null,
          "resource": {
            "eu.driver.model.cap.Resource": {
              "resourceDesc": "Image",
              "size": null,
              "uri": {
                "string": "http://pbs.twimg.com/media/BSMl0P-IcAEOsCC.jpg"
              },
```

```
              "mimeType": {
                "string": "image/jpeg"
              },
              "deferUri": null,
              "digest": null
            }
          },
          "area": {
            "eu.driver.model.cap.Area": {
              "areaDesc": "Incendie",
              "polygon": null,
              "circle": {
                "string": "48.85340999811888,2.3487998731434345 0"
              },
              "geocode": null,
              "altitude": null,
              "ceiling": null
            }
          }
        }
      }
    },
    "partition": 0,
    "offset": 7
  },
```


Test scenario 4c

Test-bed messages sent by **CrisisSuite** to **LifeX COP** in the test scenario 4c:

Table A3.4: Test scenario TS4c's Test-bed messages

Test-bed messages			
Emisor	CrisisSuite	**Receptor**	LifeX COP


```
{
    "topic": "standard_cap",
    "key": {
      "distributionID": "crisissuite-53",
      "senderID": "crisissuite",
      "dateTimeSent": 1530095293774,
      "dateTimeExpires": 1530095293774,
      "distributionStatus": "Unknown",
      "distributionKind": "Unknown"
    },
    "value": {
      "identifier": "cbd57590-79f4-11e8-a317-080027225070",
      "sender": "no-reply(at)crisissuite.com",
      "sent": "2018-06-27T12:28:05.420321+02:00",
      "status": "Exercise",
      "msgType": "Alert",
      "source": null,
      "scope": "Private",
      "restriction": null,
      "addresses": null,
      "code": {
        "string": "crisissuite_sitrep"
      },
      "note": null,
      "references": null,
      "incidents": null,
      "info": {
        "eu.driver.model.cap.Info": {
          "language": null,
          "category": {
            "eu.driver.model.cap.Category": "Safety"
          },
          "event": "publish_sitrep",
          "responseType": null,
          "urgency": "Immediate",
```

```
          "severity": "Severe",
          "certainty": "Observed",
          "audience": null,
          "eventCode": null,
          "effective": null,
          "onset": null,
          "expires": {
            "string": "2018-06-27T13:28:05.420321+02:00"
          },
          "senderName": null,
          "headline": null,
          "description": {
            "string": "Field Command Post"
          },
          "instruction": null,
          "web": {
            "string": "http://192.168.34.179"
          },
          "contact": null,
          "parameter": null,
          "resource": {
            "eu.driver.model.cap.Resource": {
              "resourceDesc": "Situation report",
              "size": null,
              "uri": {
                "string": "http://192.168.34.179/documents/download/98cde067dda6478db0d8efaab7f2bba2/"
              },
              "mimeType": {
                "string": "application/pdf"
              },
              "deferUri": null,
              "digest": null
            }
          },
          "area": {
            "eu.driver.model.cap.Area": {
              "areaDesc": "SitRep Origin",
              "polygon": null,
              "circle": {
                "string": "52.037972,4.499742 0"
              },
              "geocode": null,
              "altitude": null,
              "ceiling": null
            }
          }
        }
      }
    },
    "partition": 0,
    "offset": 3
  },
```


Test scenario 5

Test-bed messages sent by **LifeX COP** to **MDA C2** in the test scenario 5:

Table A3.5: Test scenario TS5's Test-bed messages

Test-bed messages			
Emisor	LifeX COP	**Receptor**	MDA C2


```
{
    "topic": "standard_cap",
    "key": {
      "distributionID": "lifex_cop-1037",
      "senderID": "lifex_cop",
      "dateTimeSent": 1530061920755,
      "dateTimeExpires": 1530061920755,
      "distributionStatus": "Unknown",
```

```
      "distributionKind": "Unknown"
    },
    "value": {
      "identifier": "e5b6d2ed-4c4c-4fc3-97bc-09db3a977e3e",
      "sender": "COP",
      "sent": "2018-06-27T03:12:00+02:00",
      "status": "Exercise",
      "msgType": "Alert",
      "source": null,
      "scope": "Private",
      "restriction": null,
      "addresses": null,
      "code": null,
      "note": null,
      "references": null,
      "incidents": null,
      "info": {
        "eu.driver.model.cap.Info": {
          "language": null,
          "category": {
            "eu.driver.model.cap.Category": "Geo"
          },
          "event": "AlertEVENT",
          "responseType": null,
          "urgency": "Unknown",
          "severity": "Unknown",
          "certainty": "Unknown",
          "audience": null,
          "eventCode": null,
          "effective": null,
          "onset": null,
          "expires": null,
          "senderName": {
            "string": "3020 LifeX COP"
          },
          "headline": {
            "string": "Hazard Area"
          },
          "description": {
            "string": ""
          },
          "instruction": {
            "string": ""
          },
          "web": null,
          "contact": null,
          "parameter": null,
          "resource": null,
          "area": {
            "eu.driver.model.cap.Area": {
              "areaDesc": "Danger Area",
              "polygon": {
                "string": "43.77968350917803,4.743141177459619 43.771453294936094,4.746093749999999
43.763965791852314,4.7569942474365225 43.76351948643568,4.764066698262466
43.76525509696552,4.767911917297169 43.76793279804667,4.770658499328418
43.771998706744625,4.771894452860578 43.77765085061387,4.768804548075421
43.78290587474342,4.766126634785905 43.78419477252859,4.749578479095361
43.77968350917803,4.743141177459619"
              },
              "circle": null,
              "geocode": null,
              "altitude": null,
              "ceiling": null
            }
          }
        }
      }
    },
    "partition": 0,
    "offset": 8
  }
```

Test scenario 6

Test-bed messages sent by **CrisisSuite** to **LifeX COP** in the Test scenario 6:

Table A3.6: Test scenario TS6's Test-bed messages

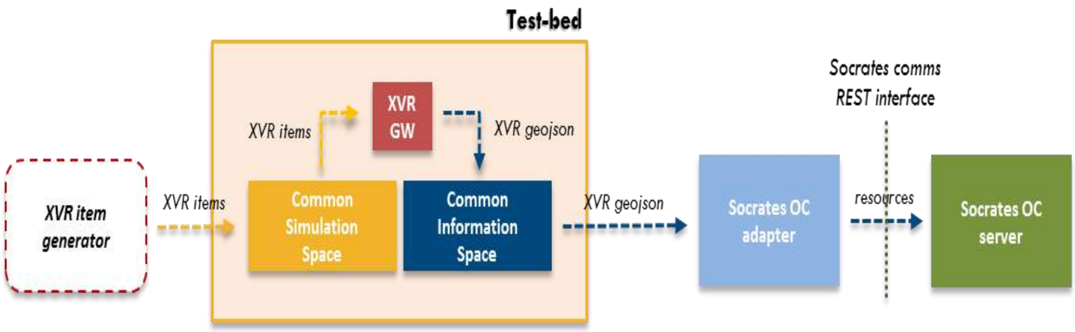
Test-bed messages			
Emisor	CrisisSuite	**Receptor**	LifeX COP


```
{
    "topic": "standard_cap",
    "key": {
      "distributionID": "crisissuite-392",
      "senderID": "crisissuite",
      "dateTimeSent": 1530106486470,
      "dateTimeExpires": 1530106486470,
      "distributionStatus": "Unknown",
      "distributionKind": "Unknown"
    },
    "value": {
      "identifier": "db28bf10-7a0e-11e8-a317-080027225070",
      "sender": "no-reply(at)crisissuite.com",
      "sent": "2018-06-27T15:34:44.835734+02:00",
      "status": "Exercise",
      "msgType": "Alert",
      "source": null,
      "scope": "Private",
      "restriction": null,
      "addresses": null,
      "code": {
        "string": "crisissuite_sitrep"
      },
      "note": null,
      "references": null,
      "incidents": null,
      "info": {
        "eu.driver.model.cap.Info": {
          "language": null,
          "category": {
            "eu.driver.model.cap.Category": "Safety"
          },
          "event": "publish_sitrep",
          "responseType": null,
          "urgency": "Immediate",
          "severity": "Severe",
          "certainty": "Observed",
          "audience": null,
          "eventCode": null,
          "effective": null,
          "onset": null,
          "expires": {
            "string": "2018-06-27T16:34:44.835734+02:00"
          },
          "senderName": null,
          "headline": null,
          "description": {
            "string": "Field Command Post"
          },
          "instruction": null,
          "web": {
            "string": "http://192.168.34.179"
          },
          "contact": null,
          "parameter": null,
          "resource": {
            "eu.driver.model.cap.Resource": {
              "resourceDesc": "Situation report",
              "size": null,
              "uri": {
                "string": "http://192.168.34.179/documents/download/4028363ccfb54b65abd29771762e51df/"
              },
```

```
            "mimeType": {
              "string": "application/pdf"
            },
            "deferUri": null,
            "digest": null
          }
        },
        "area": {
          "eu.driver.model.cap.Area": {
            "areaDesc": "SitRep Origin",
            "polygon": null,
            "circle": {
              "string": "52.037972,4.499742 0"
            },
            "geocode": null,
            "altitude": null,
            "ceiling": null
          }
        }
      }
    }
  },
  "partition": 0,
  "offset": 6
},
```

Annex 4 – Details of testing activities for Trial 1

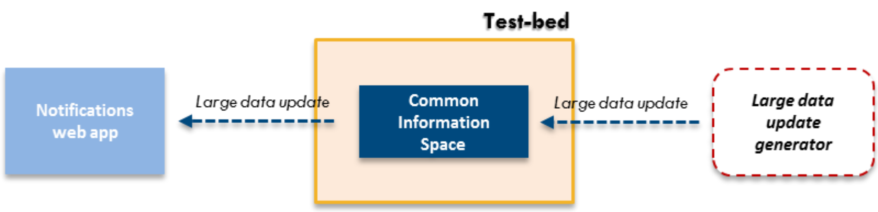
Table A4.1: TA#1 - reception of resources from Test-bed and simultaneous update of them

TA#1	
Purpose	Validate functionalities **SOC#01**, **SOC#4** and **SOC#5** in Table 3.2: • Partial update of resources by Socrates OC. • Reception of XVR's GeoJSON messages and extracting the resource info inside them by the Socrates adapter. • Creation of Socrates resource update messages by the Socrates adapter, based on the info inside XVR's GeoJSON messages and transmitting them to the corresponding Socrates OC node.
Procedure	Previous functionalities were tested by sending the corresponding XVR messages to the Test-bed and observing their reception by the Socrates OC solution immediately after. To perform the tests, a local Test-bed instance was deployed at GMV premises. The components used in these tests would be connected to the local Test-bed.  **Figure A4.1: Test set-up for reception of resources from Test-bed and simultaneous update of them** As shown in Figure A4.1, these tests involved the Socrates adapter and Socrates OC (the components under test) and it was also required to develop an XVR entity item generator which allowed injecting entity item messages into the Test-bed. This generator consisted of a very basic software stub which sent a set of predefined XVR entity item messages simulating the presence of the XVR RM tool[18]. These messages would include information about Trial 1 resources. Several tests were run using this set-up. They included from the most basic configuration, consisting of sending a single vehicle and updating its position at low rate, to more sophisticated ones where up to 20 resources of different types were reported. During the test runs, additional information was added to resources received from the Test-bed by the Socrates OC operator. Later test runs consisted of position update messages being sent at 1 update/second during about 30 minutes.

[18] Project partner TNO developed a Kafka Replay Service which provided XVR entity item messages representing the movement of a single vehicle (recorded at the Mini Trial carried out during Workshop "0"). However, for testing purpose it was decided to use an ad-hoc XVR item generator which could be easily configured according to the testing needs.

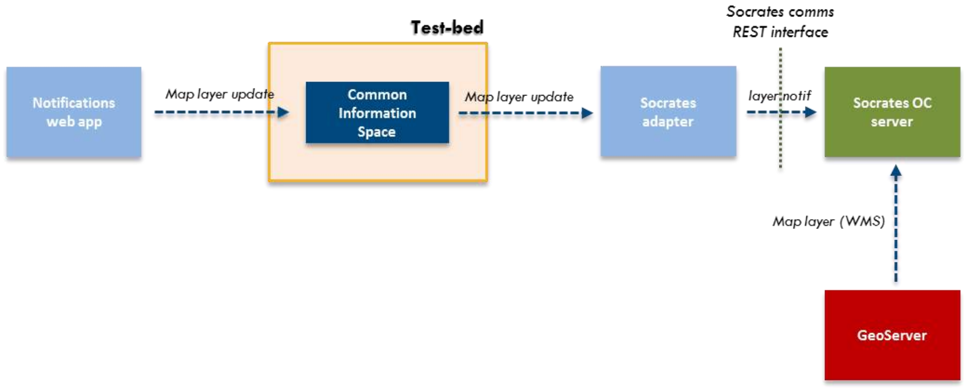
TA#1	
	The acceptance criteria for the tests were: • Resources reported by the XVR item generator were displayed in Socrates OC and updated accordingly when position updates were provided by the XVR item generator. • The types and icons associated to each resource in Socrates OC were the correct ones, according to the type assigned to the resource in the corresponding XVR entity item messages. • The information related to resources introduced by the operator of Socrates OC was not overridden by subsequent resource updates coming from the Test-bed. In case Socrates OC was not properly receiving resources, there were two mechanisms that helped identify the source of the error: • The Socrates adapter logger, which printed all messages being received. • The Test-bed development mode, which allowed subscribing directly to topics (i.e., types of messages) in the Common Simulation Space (see Figure 2.1). This way, the Socrates adapter could be directly receiving XVR entity item messages instead of their conversion into GeoJSON.
Test results	Tests mentioned above were all successfully passed before DR-2. It required, however, several iterations. First a problem was detected in the entity item to GeoJSON gateway due to conflicting SW versions (the gateway had not been updated according to the latest Test-bed release). This caused the GeoJSON messages not being received by the Socrates adapter. Once this issue was solved, some corrections were also needed in the Socrates adapter to solve some errors in the assignment of types and icons to resources (Socrates OC was not correctly displaying the expected types and icons for resources as the adapter was not properly mapping XVR item types to Socrates resource types). After these corrections, all tests fulfilled the criteria mentioned above and the functionality was validated.

Table A4.2: TA#2 - Reception of large data update messages

TA#2	
Purpose	Validate functionality **SOC#09** in Table 3.1 Reception and display of large data update notifications by the notifications web app. It was carried out during the period between DR-1 and DR-2, after successful "connection tests" of the Socrates adapter with the Test-bed in DR-1.
Procedure	Previous functionality was tested by sending the corresponding large data update messages to the Test-bed and observing whether they were displayed by the Notifications web app immediately after. Again, the local Test-bed instance deployed at GMV premises was used.  **Figure A4.2: Test set-up for reception of large data update messages**

TA#2	
	As shown in the previous Figure A4.2, this test involved the notifications web application (the component under test) and a large data update generator which allowed injecting those messages into the Test-bed. This generator consisted of a software stub which sent a set of predefined large data update messages as it would be done by the operators of 3Di and DRM solutions after one of their GeoTIFF files had been uploaded to the FTP server. The test consisted of two consecutive large data update messages being sent to the Test-bed. The acceptance criterion for the test was: large data update notifications were displayed by the Notifications web app according to the information provided by the large data update generator. The notifications web app also included a logger that printed messages coming from the Test-bed.
Test results	The test was successfully passed before DR-2.

Table A4.3: TA#3 - sending and reception of map layer update messages

TA#3	
Purpose	The tests comprised by this testing activity were aimed to validate functionalities **SOC#2**, **SOC#3**, **SOC#6** and **SOC#10** in Table 3.2 Reception of map layer update notifications by Socrates OC. Display of key legend in Socrates OC for the interpretation of the colour codes associated to the flood arrival times and flood water depth map layers. Reception of map layer update messages by the Socrates adapter and generation of the associated notification to the corresponding Socrates OC node. Transmission of map layer update notifications by the notifications web app. They were carried out during the period between DR-1 and DR-2 after successful "connection tests" of the Socrates adapter with the Test-bed in DR-1.
Procedure	Previous functionalities were tested by sending the corresponding map layer update messages using the notifications web app to the Test-bed and observing the reception of the corresponding notification in Socrates OC solution immediately after. Once the message was received the corresponding map layer was loaded Socrates OC. The same local Test-bed instance than in previous tests were used.  **Figure A4.3: Test set-up for sending and reception of map layer update messages**

TA#3	
	As shown by Figure A4.3, these tests involved the notifications web app, the Socrates adapter, Socrates OC (the components under test) and a GeoServer where map layers were published. These map layers were based on some example GeoTIFFs provided by 3Di (three files: a water depth, a digital elevation and a flood arrival times model) and DRM (one file: an orthophoto of SGSP premises). Four tests were run under this set-up, one for each of the previous GeoTIFF files. On each test the corresponding image was processed and published in the GeoServer (in the case of DRM orthophoto it could be directly published as a shapefile; in the case of 3Di products some additional processing was required before and the related map layer update message was sent using the Notifications web app. The criteria to consider the tests passed were: The map layer update messages sent with the Notifications web app were received and correctly displayed by Socrates OC. The map layers which previous update messages referred to were properly loaded and displayed by Socrates OC. The corresponding key legend (in the case of 3Di products, for interpreting their colours) was properly displayed by Socrates OC at the moment of displaying the map layer.
Test results	Tests mentioned above were all successfully passed before DR-2.

Table A4.4: TA#4 - Identification of the owner of resources reported by the Test-bed

TA#4	
Purpose	The test comprised by this testing activity was aimed to validate functionality **SOC#07** in Table 3.2: • Identification (by the Socrates adapter) of the owner (node) of each resource based on its geographical location. This required functionality was identified during DR-2 after discarding the possibility of the owner node being directly provided as part of the resource information provided by the simulation tool. The owner nodes would be either the Crisis Management Centre in LANDPOL or the Crisis Management Centre in MANYGER, according to the Trial scenario. Each of these CMCs had the responsibility of responding to those crisis events occurred in their respective regions (i.e., LANDPOL or MANYGER). This way the owning node of a given resource would be inferred from the initial position of that resource (CMC in LANDPOL if the resource was located inside LANDPOL area, CMC in MANYGER otherwise[19]). This functionality was implemented and tested during the period between DR-2 and Trial 1 (a first version, further elaborated afterwards, was already developed during DR-2). As the functionality was fully ad-hoc for the Trial 1's scenario and the list of resource to be used during the Trial execution was known in advance (elaborated and distributed by the Trial Owner and XVR after DR-2), the implemented functionality just took into account the

[19] No resources outside the areas of LANDPOL and MANYGER were considered in Trial 1.

<table>
<tr><th colspan="2">TA#4</th></tr>
<tr><td></td><td>expected positions of the concrete resources to be used, in order to minimize the effort to be put on the implementation of the functionality. It has to be noted that, due to the fact that a resource could move and cross the border between LANDPOL and MANYGER, the owner of resources had to be established only the first time the resource was reported, and not being modified afterwards by the Socrates adapter.</td></tr>
<tr><td>Procedure</td><td>Previous functionality was tested by sending the corresponding XVR entity item messages to the Test-bed and examining the information associated to the corresponding resources once received by the Socrates OC solution.

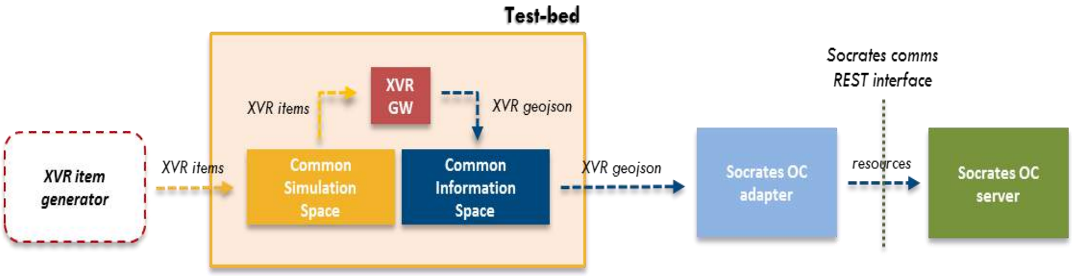


Figure A4.4: Test set-up for the identification of the owner of resources reported by the Test-bed

As shown by previous figure, these tests involved the Socrates adapter (the component under test), Socrates OC and the XVR item generator used for previous tests.
The test consisted of several resources being reported from the XVR item generator (some of them located in LANDPOL while others were in MANYGER), whose positions were later updated to make them cross the border between the two regions.
The resources and their initial positions were based on the list of resources developed by the Trial Owner and XVR after DR-2.
The criteria to consider the tests passed were:

The owner of resources displayed by Socrates OC was the expected one according to the position of the resource (LANDPOL if the resource was inside LANDPOL area; MANYGER otherwise).
When a resource moved from one area to the other (according to the position updates sent by the XVR item generator), its owner was not modified, remaining the original one.
</td></tr>
<tr><td>Test results</td><td>After two iterations, tests mentioned above were all successfully passed before Trial 1 execution. Some minor corrections were required to determine the owner of resources which were very close to the border between LANPOL and MANYGER.</td></tr>
</table>

Table A4.5: TA#5 - Reception of resources from the Test-bed at high rate

<table>
<tr><th colspan="2">TA#5</th></tr>
<tr><td>Purpose</td><td>The tests comprised by this testing activity were aimed to validate functionality SOC#8 in Table 3.2:

Management of the Test-bed messages update rate by the Socrates adapter, in order to avoid performance issues in Socrates OC.

This functionality was implemented as a result of some performance issues detected</td></tr>
</table>

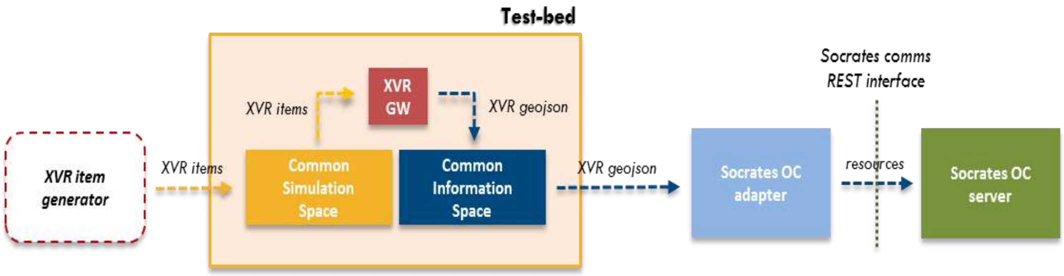
TA#5	
	during DR-2. The Trial set-up during DR-2 included hundreds of resources (vehicles) being simulated by XVR RM. These resources were all reported at once when the scenario started resulting in *Socrates OC* underperforming (in a matter of seconds) and appearing to be blocked for several minutes. The solution adopted (the only one feasible in the given timeframe) consisted of temporarily storing resources received in the Socrates adapter, and send them to *Socrates OC* at a lower rate (the final rate was 50 resources every 10 seconds; determined empirically according to the observed solution behaviour). This would make the initialization process take longer but would avoid Socrates OC being collapsed. The associated tests were run during the period between DR-2 and Trial 1. It has to be noted that during that period, and according to the feedback provided by practitioners during DR-2, the Trial Owner decided to use modules instead of vehicles as the resources to be managed in Trial 1. A module can be seen as a unit providing some predefined response capabilities and may include several vehicles, etc. This way, the total amount of resources to be managed during Trial 1 execution became significantly reduced (from several hundreds to around 40).
Procedure	Previous functionalities were tested by sending several hundreds of XVR entity item messages, each one reporting a different resource, to the Test-bed and observing their reception by the Socrates OC and the subsequent behaviour of the solution.  **Figure A4.5: Test set-up for reception of resources from the Test-bed at high rate** As shown by Figure A4.5, these tests involved the Socrates adapter and Socrates OC (the components under test) and the XVR item generator. Several tests were run under this set-up, each of them increasing the number of resources being sent at once by the XVR item generator (according to the following series: 50, 100, 200, 300, 400, 500). On each test, the reception of resources by Socrates OC was verified and the behaviour of the solution within the following two or three minutes approx. was observed (more or less the time the Socrates adapter took to send all resources to Socrates OC plus the time the latter took to stabilize). The criteria to consider the tests passed were: • Resources reported by the XVR item generator were displayed in Socrates OC at the expected rate. • The solution did not get blocked during the reception of resources and ran smoothly after it.
Test results	Tests mentioned above were all successfully passed before Trial 1 execution. As mentioned in the purpose description of this testing activity, it was first required to determine a suitable resource update rate for Socrates OC, which was determined empirically by using a resource injector directly connected to the solution. Once the update rate was fixed, the procedure above was followed according to the test set-up depicted in Figure A4.5.

Table A4.6: TA#6 - Debugging of reception of resources from the Test-bed at high rate

<table>
<tr><th colspan="2">TA#6</th></tr>
<tr><td>Purpose</td><td>During the execution of Trial 1 (approximately 15 minutes the first session with Socrates OC started) an issue was detected in the communications of Socrates with the Test-bed, which caused Socrates OC receiving thousands of resource updates from the Test-bed. This issue overloaded Socrates OC and prevented the Test-bed from being used for reporting resources for the rest of the Trial execution (as finally only 40 resources were used during Trial 1, they were manually inserted into Socrates OC, instead of automatically receiving them form the Test-bed).</td></tr>
<tr><td>Procedure & Outcome</td><td>After a deeper analysis and debugging at the solution provider’s premises, it was concluded that the problem was originated by the Socrates adapter, concretely by the functionality which was developed for managing resource updates at high rate. This functionality properly fulfilled its objective of distributing large bursts of resource updates (i.e., more than 50 resource updates being received at once from the Test-bed) in periods of 10 seconds, so Socrates OC did not get overloaded; however, it was introducing an error which caused messages coming from the Test-bed being duplicated over and over (due to a message queue whose messages were not being properly removed). This way, there was no problem with the update rate (as demonstrated by tests documented in previous section), but the Socrates adapter was sending updates endlessly (without having received new updates from the Test-bed). After about 15 minutes, Socrates OC started to underperform due to continuous reception of resource updates being stored in its DB (several thousand at that time).</td></tr>
<tr><td>Additional notes</td><td>This issue was not detected during previous tests due to the following reasons:
The faulty code was added just after DR-2, so it could not be tested until a few days before the execution of the Trial.
Tests performed after implementing this modification were not stressing Socrates to the point it was when the real scenario was executed. All tests took no more than 2 or 3 minutes, while the issue was detected during Trial 1 after about 15 minutes playing the scenario.
This reveals not enough testing in the latter case, as longer tests would have detected the issue. Added to this is the fact that the late change in the resource set-up for the Trial (replacing vehicles with modules, reducing the number of resources being used from several hundreds to some dozens) caused that the functionality which introduced the error was not finally paid too much attention.</td></tr>
</table>

Annex 5 – Test scenarios of Trial 2

Test scenario 0: first SiTac into the LifeX COP

After a fire has been detected, the Field Command Post (FCP) opens the LifeX COP to report it. A first SiTac is created by using the legacy system ASPHODELE. The SiTac in KMZ[20] format is imported into the LifeX COP.

Objective: visualize the elements of the first SiTac (i.e. fire ignition point) in LifeX COP.

As the legacy system ASPHODELE was not foreseen to be evaluated in the Trial ASPHODELE was not integrated into the Test-bed, thus the information exchange in this test scenario is not done through the Test-bed.

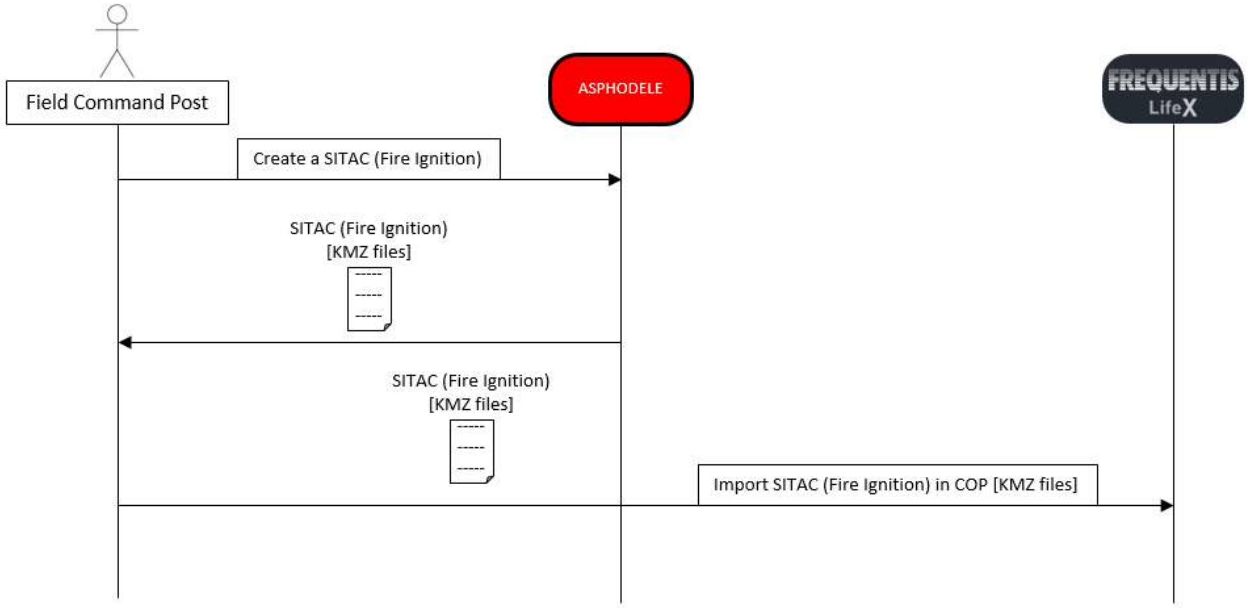


Figure A5.1: Test scenario 0 sequence diagram – Trial 2 – Dry Run 1

Test scenario 1: Additional LifeX COP actors enter in scene

Several actors from different organizations interact with several instances of LifeX COP, each of them having different access rights to information in the COP.

TS1a: Opening of COP (CM System) to (cross border) organization.

A new organization is involved in the crisis (can be either DREAL (**D**irection **R**égionale, de l'**E**nvironnement de l'**A**ménagement et du **L**ogement), or fire fighters – liaison officer / local HQ (headquarter), or EMS (**E**mergency **M**edical **S**ervices) HQ and needs to access information in LifeX COP.

[20] Keyhole Markup Language (KML) is an XML notation for expressing geographic annotation and visualization. KML files are very often distributed in KMZ files, which are zipped KML files.

Objective: This new organization can open and work with the LifeX COP with a different Organization account.

TS1b: COP access rights.

Different organizations maintain a private view of their alerts and resources and can share information with each other.

Objective: actors can only see the map layers for which they were authorized to access in the LifeX COP.

Test scenario 2: DREAL can use CrisisSuite internally between different levels (SITREPS)

Managers from DREAL open CrisisSuite and are able to exchange SitReps to other actors/organizations in different levels.

Objective: Sharing information between different instances of CrisisSuite.

Test scenario 3: Detection of information of interest through social media

A set of tweets related to an incident are published on Twitter (in a private account so they will not upset the public). The firefighting social media manager uses SMAP to collect them automatically by using search/filtering criteria based on geographical information and thematic area to elaborate a social report of the incident or an early detection of new events.

The relevant information out of this social media digestion (representative tweets) is shared among the rest of participants through the Test-bed to LifeX COP.

Example for relevant social media information: "A relevant number of tweets about a blocked road by fire or fallen trees", "a traffic jam", etc.

Objective: to share information from the social media manager to the rest of actors in the scenario.

The tweets were simulated by XVR and pushed into the Test-bed to guarantee a proper scenario.

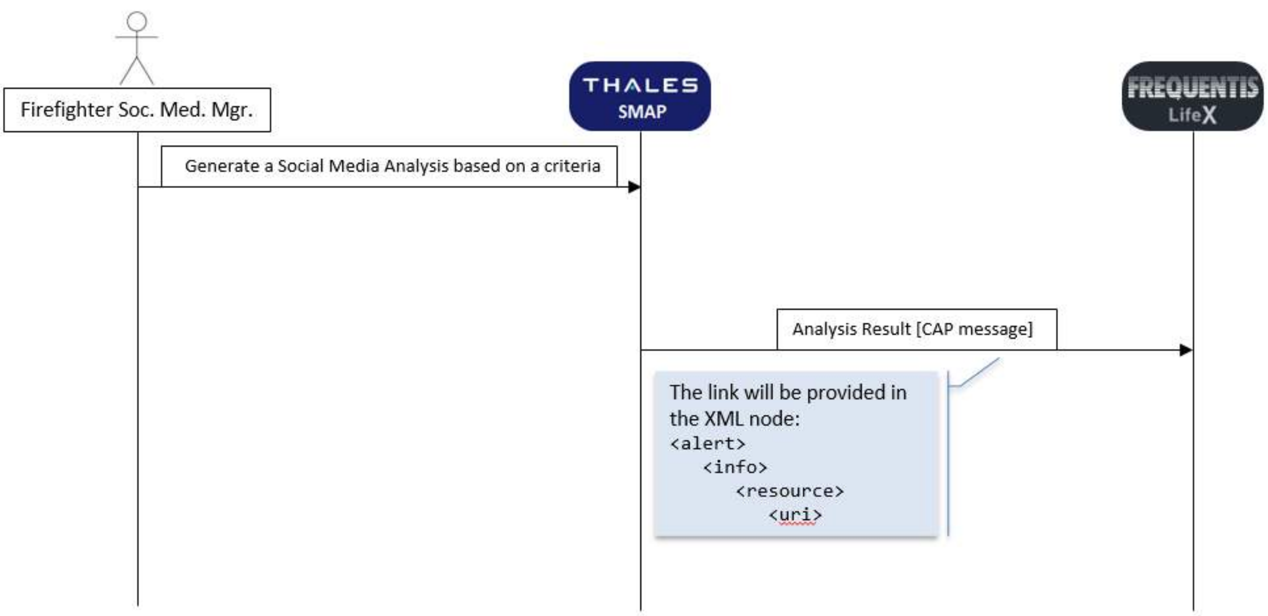


Figure A5.2: Test scenario 3 sequence diagram – Trial 2 – Dry Run 1

Test scenario 4: Inputs into LifeX COP

Different actors update the information visualized in LifeX COP from diverse sources (other solutions). All actors can update/visualize only according to their access rights in COP.

TS4a: inputs from EMS.

The EMS (Emergency Medical Servive) HQ sends through the MDA C2 solution an intervention report about victims and casualties.

Objective: The report is shared with other organizations and geolocated in the LifeX COP (via EMSI[21] messages).

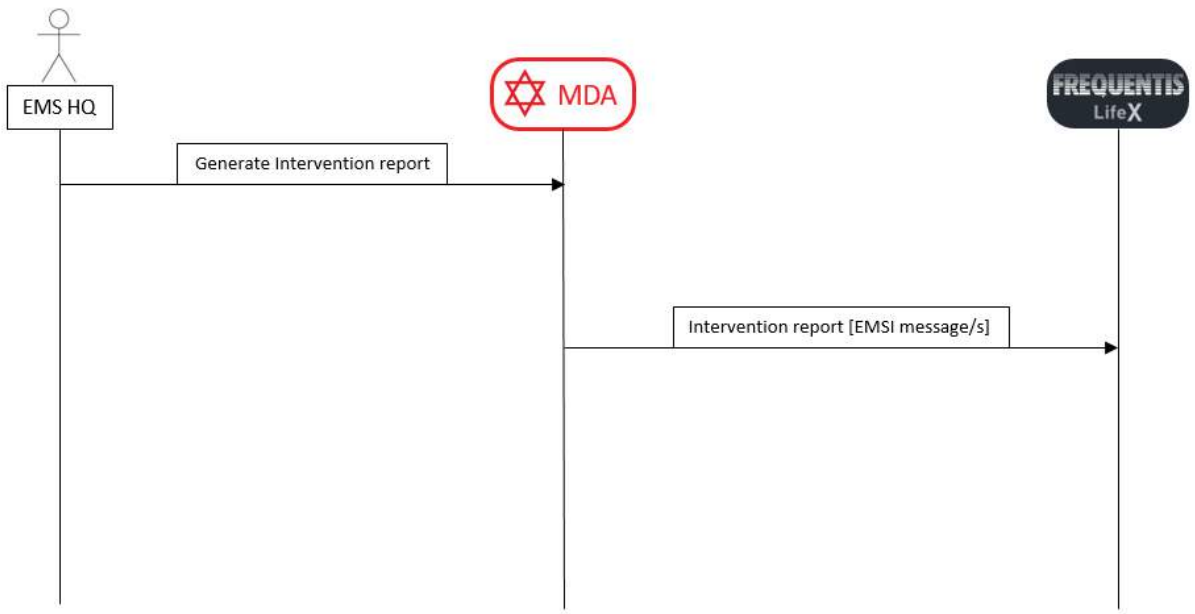


Figure A5.3: Test scenario 4a sequence diagram – Trial 2 – Dry Run 1

TS4b: inputs from FR firefighters

The French firefighter social media manager selects tweets and shares this information with the rest of the actors.

Objective: FR firefighter social media manager can report new findings in the social networks.

[21] Emergency Management Shared Information (EMSI) are messages structured according to ISO/TR 22351:2015.

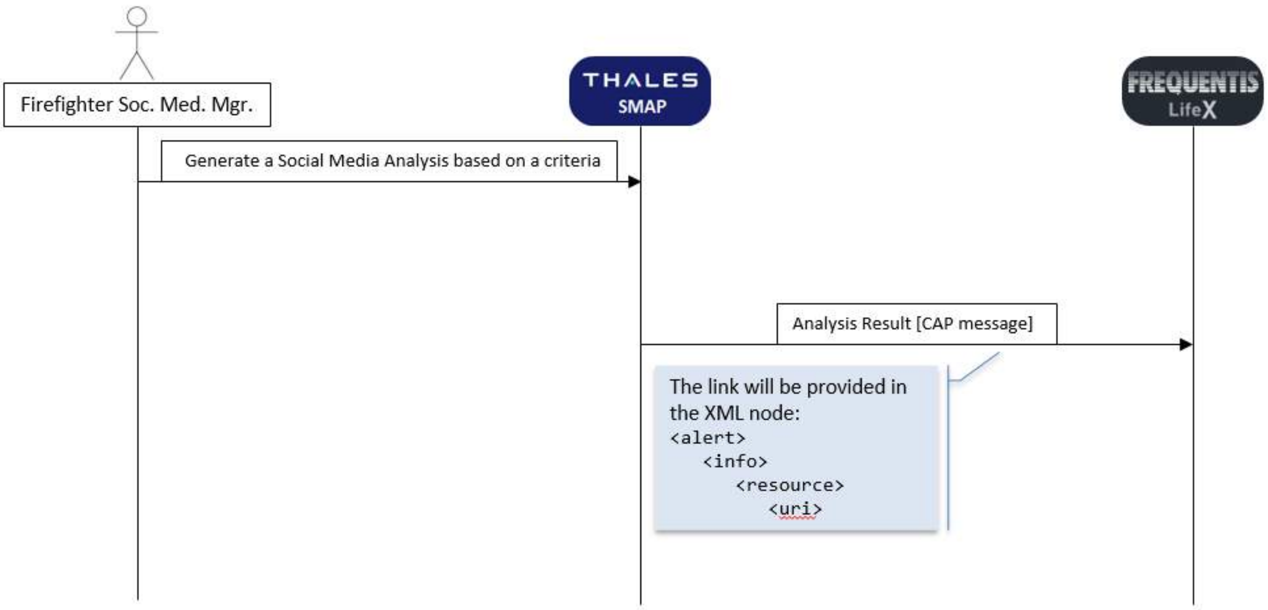


Figure A5.4: Test scenario 4b sequence diagram – Trial 2 – Dry Run 1

TS4c: inputs from DREAL.

DREAL sends a new SitRep from CrisisSuite which is made accessible via the LifeX COP.

Objective: Sharing information from CrisisSuite into the LifeX COP.

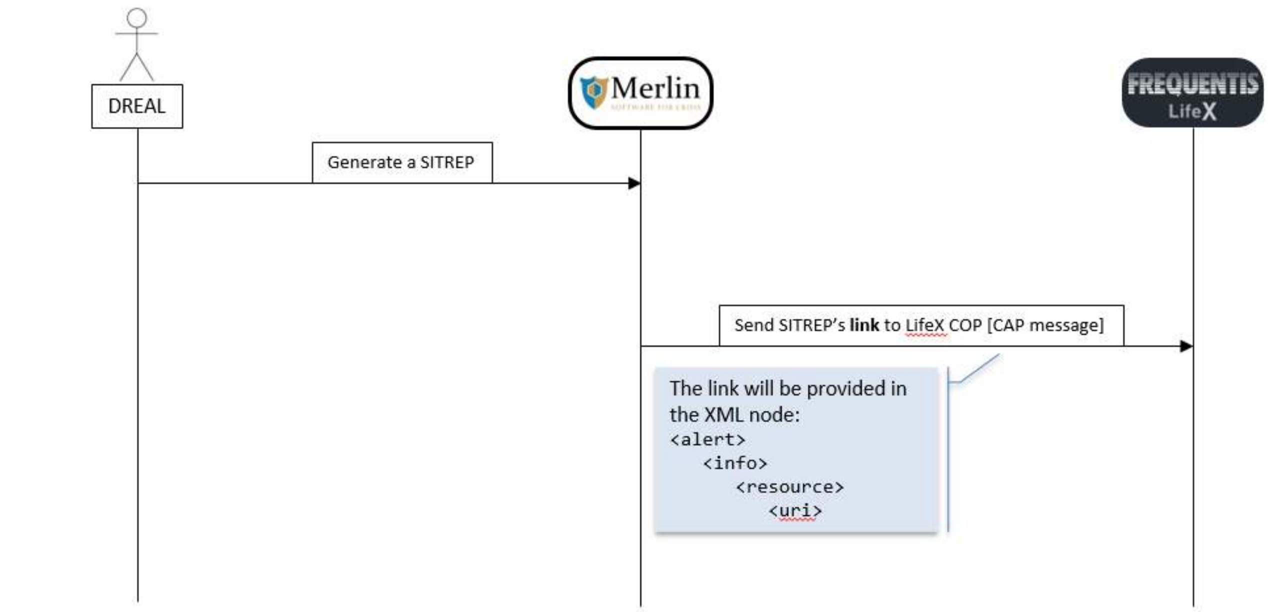


Figure A5.5: Test scenario 4c sequence diagram – Trial 2 – Dry Run 1

TS4d: inputs from FCP.

An updated SiTac is generated from the Field Command Post including the progression of the fire: a contour of Hazard is defined. This new SiTac will not replace to the old one but will extend the information previously notified.

Objective: field command post can send an updated SITAC updating the previous one.

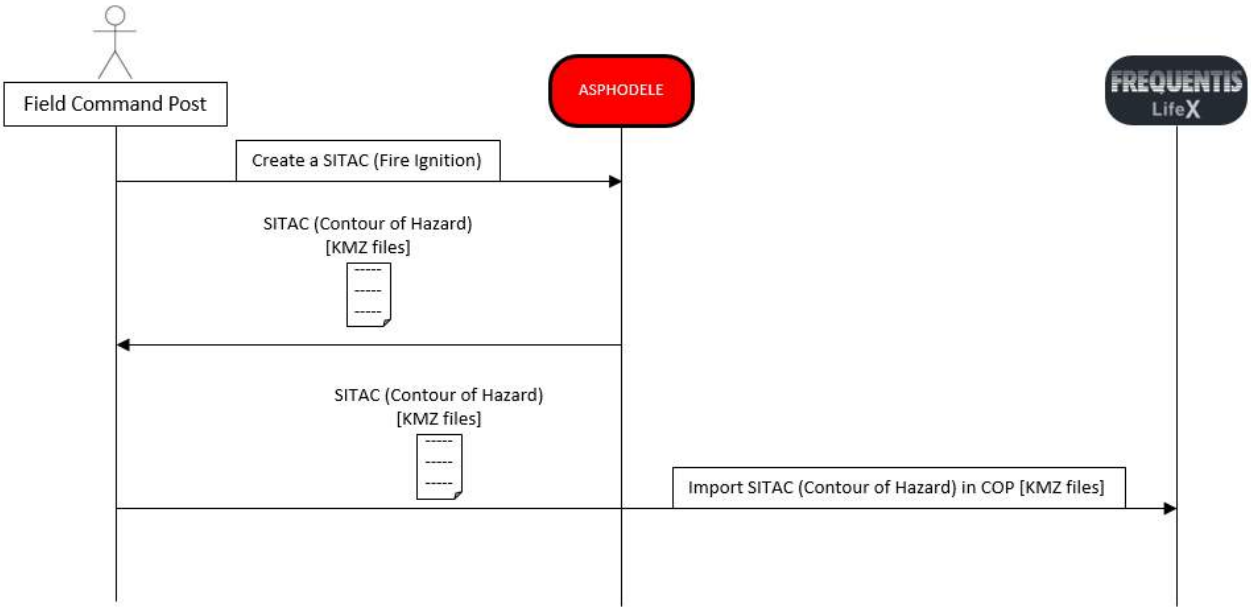


Figure A5.6: Test scenario 4d sequence diagram – Trial 2 – Dry Run 1

Test scenario 5: Sharing of Hazard contour

Once the hazard contour becomes available in LifeX COP (as one of the elements contained in the SiTac described in the test scenario 4d) this information is shared to the MDA C2 solution in order to enable the emergency services (EMS) to route the emergency resources accordingly to avoid dangerous and/or inaccessible areas.

Objective: EMS teams can take this information into account in its routing decisions.

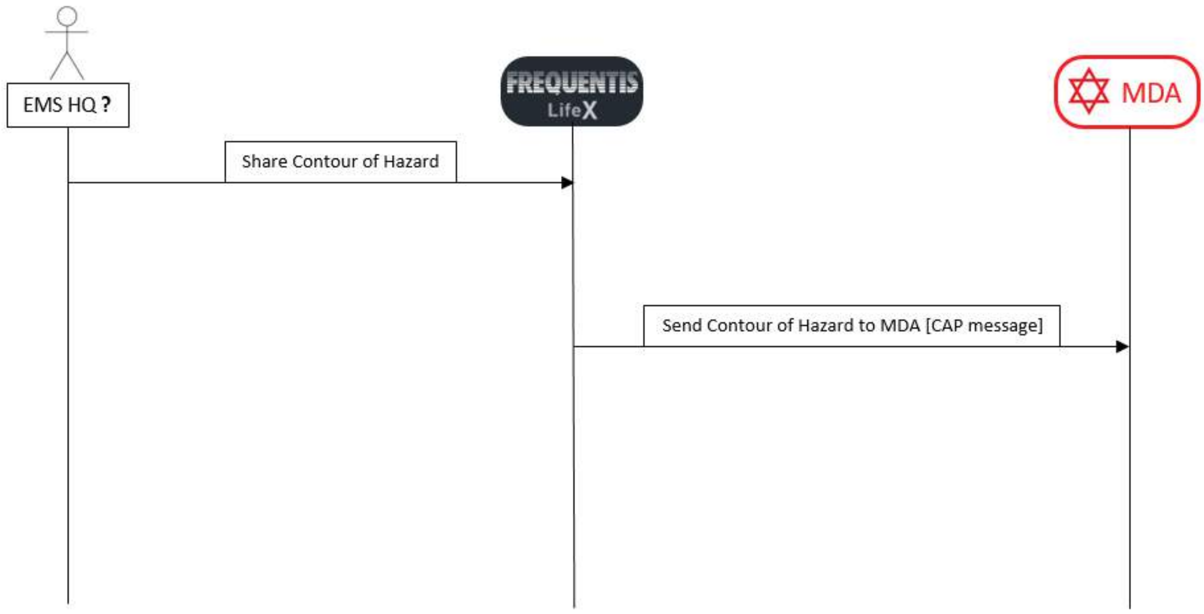


Figure A5.7: Test scenario 5 sequence diagram – Trial 2 – Dry Run 1

Test scenario 6: Request for aerial support from ERCC

A specific kind of SitRep is created following the model of the ERCC request for means.

Objective: Any position in the higher levels of the command chain (i.e. COGIC) can prepare their reinforcement request to ERCC via the information available in CrisisSuite and in the COP.

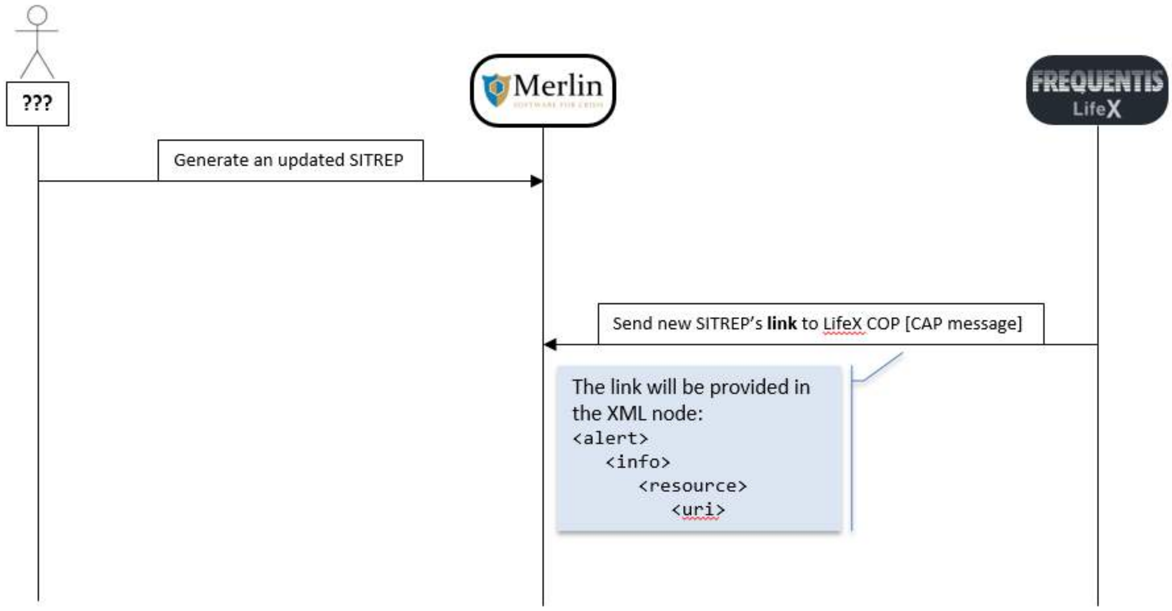


Figure A5.8: Test scenario 6 sequence diagram

Trial 2 – Dry Run 1 Trial 2 test scenario details

TS DR-1-01: Call and opening of incident

CODIS receives a call (not to be tested) and creates a new event in CrisisSuite. Characteristics of event shall contain:

1. Tentative location.
2. Type of event.
3. Type of means (reconnaissance) that are sent on first intention.
4. Movements of Means moving to scene are displayed on map.

N.B: Locations are entered using the map.

TS DR-1-02: Arrival of Incident Commander (IC) & request for reinforcement using CrisisSuite

1. Incident Commander sends (using CrisisSuite) first IC report (I am, I see, I foresee, location of Field Command Post (FCP) and Transit Point (TP).
2. COP automatically extracts locations from « IC » report.
3. Location of FCP and Transit post are displayed automatically in COP with adequate symbology.

TS DR-1-04: Updating of the Incident location on COP (before SiTac arrives)

1. Incident Commander updates the Incident location on COP.

<u>Test scenario 0: first SiTac into the LifeX COP:</u>

After a fire has been detected, the Field Command Post (FCP) opens the LifeX COP to report it.

1. A first SiTac is created by using a legacy system called ASPHODELE.

2. The SiTac, in KMZ format is imported into the LifeX COP.

Objective: visualize the elements of the first SiTac (i.e. fire ignition point) in the LifeX COP.

As ASPHODELE is not being evaluated in the Trial the information exchange in this test scenario is not done through the Test-bed.

TS DR-1-05: New SiTac is sent by Asphodèle to LifeX COP

1. Asphodèle sends a new SiTac.
2. The new SiTac is displayed in LifeX COP.
3. The SiTac is loaded in a new specific overlay.
4. The display of overlays can be managed by user.
5. Other SiTac layers are hidden when new layer is displayed.

TS DR-1-06: Social Media Summary report to FCP

1. Social media manager (CODIS) generates a dashboard (by applying various filters).
2. A link to dashboard is manually uploaded in CrisisSuite (CS) by CODIS.
3. CS sends the social media summary report SITREP to Field Command Post (FCP).

TS DR-1-07: DREAL CrisisSuite event form (input for risk analysis)

1. DREAL fills event form in CrisisSuite (including factory location and danger area). DREAL sends SITREP (based on Event Form).
2. Event form is automatically distributed to all DREAL levels and to COP.
3. Factory Location and Danger Area are displayed on map (automatically extracted by LifeX from received Event Form).
4. Event Form is displayed at factory location in COP.
5. Event form can be opened from COP map.

TS DR-1-08: DREAL CrisisSuite uploads GP Form

1. DREAL uploads (attachment /mail) GP (Severity Perception) form (received from external stakeholder).
2. This form is now visible in CrisisSuite.
3. The User can open GP form from CrisisSuite.

TS DR-1-03: LifeX COP Symbology compliant with Valabre Symbology

1. All tactical entities are displayed on a map with adequate symbology (provided with Valabre).

N.B: The list of tactical entities is currently to be validated by Valabre (e.g. Fire trucks ambulances, roadblock...).

TS1: Additional LifeX COP actors enter in scene

Different actors would interact with LifeX COP, who doesn't belong to the same organizations. Therefore, they are allowed to visualize their own content and what has been shared with them.

<u>Test scenario 1a: Opening of COP (CM System) to (cross border) organization:</u>

A new organization is involved in the crisis (can be either DREAL, or IT fire fighters – liaison officer / local HQ, or EMS HQ) and needs to see the LifeX COP.

Objective: This new organization is able to open and work with the LifeX COP with a different Organization account.

<u>Test scenario 1b: COP access rights:</u>

Different organizations maintain a private view of their alerts and resources and are able to share information one to each other.

Objective: actors can only see the layers for which they are authorized in the *LifeX COP*.

TS 2: DREAL can use CrisisSuite internally between different levels (SITREPS)

Managers from DREAL open CrisisSuite and are able to exchange SitReps to other actors/organizations in different levels.

Objective: Sharing information between different instances of CrisisSuite.

TS 3: Detection of information of interest through Social media

The firefighting social media manager uses SMAP to:

1. Collect automatically Tweets according its topic.
2. Search/filtering criteria based on geographical information and thematic area.
3. Push the relevant information out of this social media digestion to LifeX COP and shared among the rest of participants.

Example: "A relevant number of tweets about a blocked road by fire or fallen trees", "a traffic jam", etc…

Objective: to share information selected by the social media manager with other participants.

TS 4a: EMS HQ reports on victims' status

The EMS HQ sends through MDA C2 an intervention report about the victims and casualties.

Objective: The report is shared to the rest of organizations and geolocated in the LifeX COP (via EMSI messages).

TS 4b: Social Media manager send tweets of interest to COP

The firefighting social media manager selects pins tweets and shares information among the rest of actors.

Objective: FR firefighter social media manager can report new findings in the social networks.

TS 4c: DREAL sends new SITREP to LifeX COP

DREAL sends a new SitRep from CrisisSuite that is accessible via the LifeX COP.

Objective: Sharing information from CrisisSuite into the LifeX COP.

TS 4d: FCP sends SITAC update to COP

An updated SiTac is generated from the Field Command Post including the progression of the fire: a contour of Hazard is defined.

This new SiTac will not replace to the old one, but will extend the information previously notified.

Objective: field command post can send an updated SiTac updating the previous one.

TS 5: Sharing of Hazard contour

Once the contour of hazard is available in the LifeX COP (as one of the elements contained in the SiTac described in the test scenario 4d) this information is shared to the MDA C2 app, so the emergency services (EMS) can route accordingly the emergency resources to avoid the area.

Objective: EMS teams can take this information into account in its routing decisions.

TS DR-1-09: Dispatching and routing of ambulance

This test-scenario is an update of TS5.

- Danger area is sent by Asphodèle through SITAC
- SITAC is updated in LifeX COP.
- Danger area is automatically sent to MDA.
- MDA routes ambulances taking Danger Area into account (avoiding).
- Movements of ambulances are simulated in XVR.
- Movements of ambulances are shown in LifeX COP.

TS 6: Request for aerial support from ERCC

A specific kind of SitRep is created following the model of the ERCC request for means.

Objective: The higher levels of the command chain (i.e. COGIC) can prepare their reinforcement request to ERCC via the information available in CrisisSuite and the COP.

TS DR-1-10 EMS is warning CODIS there are Victims (CrisisSuite)

CODIS (through MDA) receives call saying that fire is making victims

- CODIS sends notification there are victims in CrisisSuite (with of requests activation of Novi Plan).

This paragraph describes the content of the messages exchanged during the Trial.

Table A5.1: exchanged messages 424

#	Output	Input	Final recipient	Data type	Via	Format
1	SMAP	Test-bed	LifeX COP	Tweet	Test-bed	CAP
2	MDA	Test-bed	LifeX COP	Casualties report	Test-bed	EMSI
3	MDA		XVR	Routing		
4	LifeX COP	Test-bed	MDA	Danger zone	Test-bed	CAP+AREA
5	CrisisSuite	Test-bed	LifeX COP	SitRep	Test-bed	CAP+URL
6	CrisisSuite	Test-bed	LifeX COP	SitRep	Test-bed	CAP+ pdf
7	CrisisSuite	Test-bed	LifeX COP	SitRep	Test-bed	CAP+ geo
8	Asphodèle/Crimson	LifeX COP		Sitac		KML
9	SMAP	CrisisSuite		Dashboard link		URL

Annex 6 – IP addresses and technical users for Trial 1 and Trial 2

Table A6.1: IP address and technical user of the Test-bed in Trial 2

Key	Value
IP	192.168.34.177
Technical user	Martijn Hendriks (XVR)

Table A6.2: Solutions and Test-bed IP addresses and responsible technical users

Solution	Test-bed Sender ID	IP	Technical User
LifeX COP	LifeX_COP	192.168.34.175	Thomas Obritzhauser (FRQ)
SMAP	SMAP_at_THALES	192.168.34.174	Antoine Leger & Bruno Quéré (TCS)
MDA C2	mda_c2	192.168.34.176	Ran Dahan & Itamar Laist (MDA)
CrisisSuite	crisissuite	192.168.34.179	Rudolf van den Beukel (MERLIN)

Annex 7 – Contact details of solution providers of Trial 1 and Trial 2

Table A7.1: Contact Details

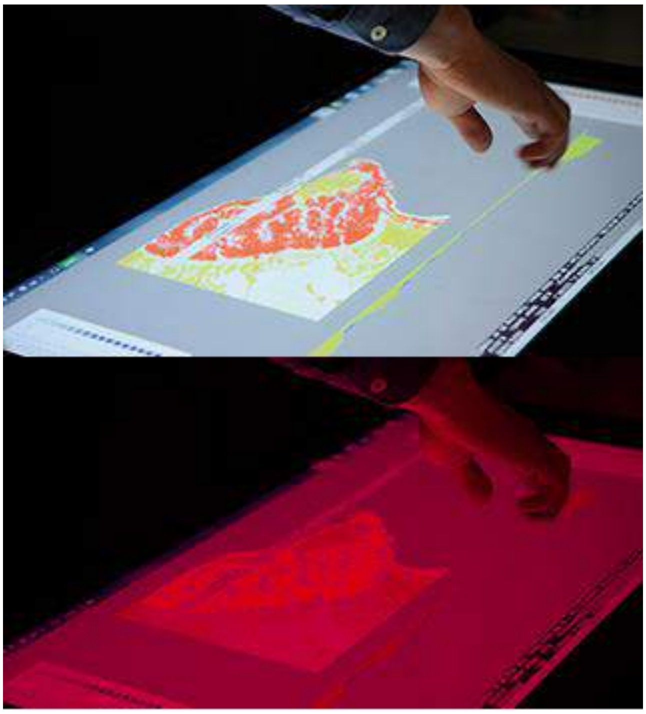
Details for	Name	Role	Email	Phone number
FRQ LifeX COP	Thomas Obritzhauser	Software engineer	Thomas.OBRITZHAUSER(at)frequentis.com	+43 664 608 502 333
3Di and Lizard	Ciska Overbeek	Solution Provider 3Di	ciska.overbeek(at)nelen-schuurmans.nl	+31 30 233 0200
GMV	Raul Valencia Perez	Solution Provider Socrates OC	rvalencia(at)gmv.com	+34 918 072 100
Drone Rapid Mapping	Marcin Marjasiewicz	Solution provider Drone Rapid Mapping	marcin.marjasiewicz(at)hexagonsi.com	+48 22 495 88 84
MDA C2	Itamar Laist	Software engineer	itamarl(at)mda.org.il	
MDA C2	Ran Dahan	Software engineer	rand(at)mda.org.il	
SMAP	Antoine Leger	Software engineer	antoine.leger(at)thalesgroup.com	+33 173 322 058
SMAP	Laurent Dubost	Project Manager	laurent.dubost(at)thalesgroup.com	+33 682 945 185
CrisisSuite	Ruud van den Beukel	CEO	ruud(at)merlincrisis.com	+31 615 541 315

Annex 8 – Description of DRIVER+ solutions (Export of PoS database)

This Annex is an export of the PoS (Portfolio of Solutions) database containing all available information concerning the DRIVER+ solutions available at 01/11/2018. The solution descriptions are listed in alphabetical order. Please note that the formatting of this Annex is the original formatting applied within the online PoS database and does not follow DRIVER+ template standards.

3Di

This solution is an external solution.



Team

Real Name

Anne Leskens

Roles

Owner

Summary

3Di enables flood forecasting on the basis of a detailed model. The model is able to predict flooding locations, water depths, and water arrival times, among others. The output of 3Di models is provided in GeoTIFF and netCDF form. The results can easily be analysed and processed via a ready-to-use plugin in open GIS software tool QGIS.

The maps generated by the 3Di models can be used to adjust local risk management. 3Di is a fast, accurate and interactive modelling suite, in which users can easily adapt the model during runs. For example, users can open breaches, or add portable levees to investigate the effects of implementing possible mitigation measures for flood scenarios.

Innovation stage

Stage 6: Wide-scale Adoption

Readiness

TRL 9

Crisis size

Local

Regional

Cross-border

Large scale

Crisis Cycle Phase

Preparedness

Response

Recovery

Mitigation

Trials

3Di trialled in Poland

Open configuration options

DRIVER+ Trial 1

Trial type

Trial

3Di was trialled in the first DRIVER+ Trial

Open configuration options

CM-Functions

Mitigation of effects through identification of vulnerabilities

Assess vulnerabilities to hazards

The model can predict the extent and depth of the flooding/chemical spill, and thus with some postprocessing also the vulnerable roads (i.e. inaccessible to normal and/or calamity traffic), also across borders, and thereby sites with high priority for evacuation. Since the model is interactive and fast, with an easy interface, the effect of optimizing/mitigation measures can be calculated and visualised easily.

Raise awareness and anticipate to support decision makers with protection and response measures

Raise awareness and anticipate

Combine “current operational status” with consequence analysis to build a real-time geospatial framework in support of effective and timely decision-making. Provide predictive analysis and deliver situational awareness to the decision makers and first responders by using modelling. Investigate the effects of optioneering/mitigation measures to support decision makers.

Communication between stakeholders for shared situational awareness

Provide communications and information support to C3

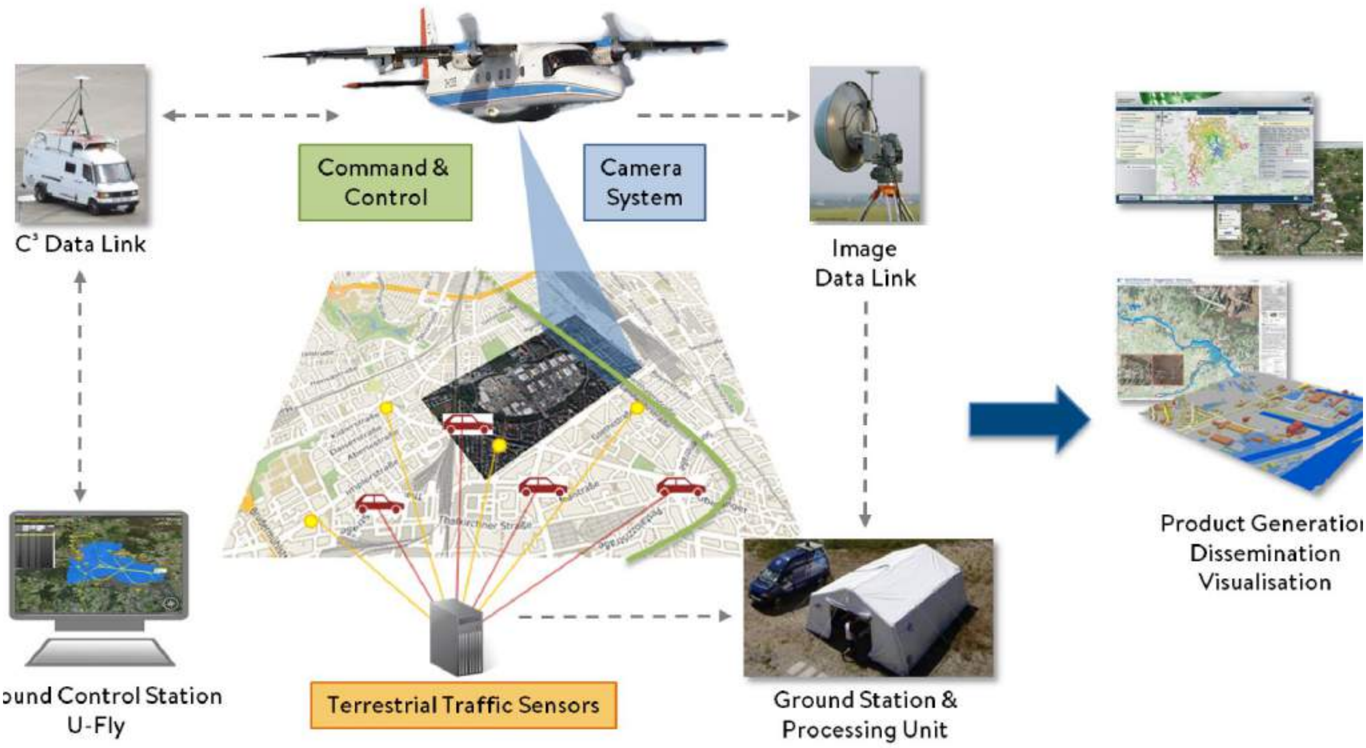
3Di provides an intuitive and interactive framework which shows the extent of the flooding/chemical spill, including water depths and arrival times, which can assist in clear communication between stakeholders.

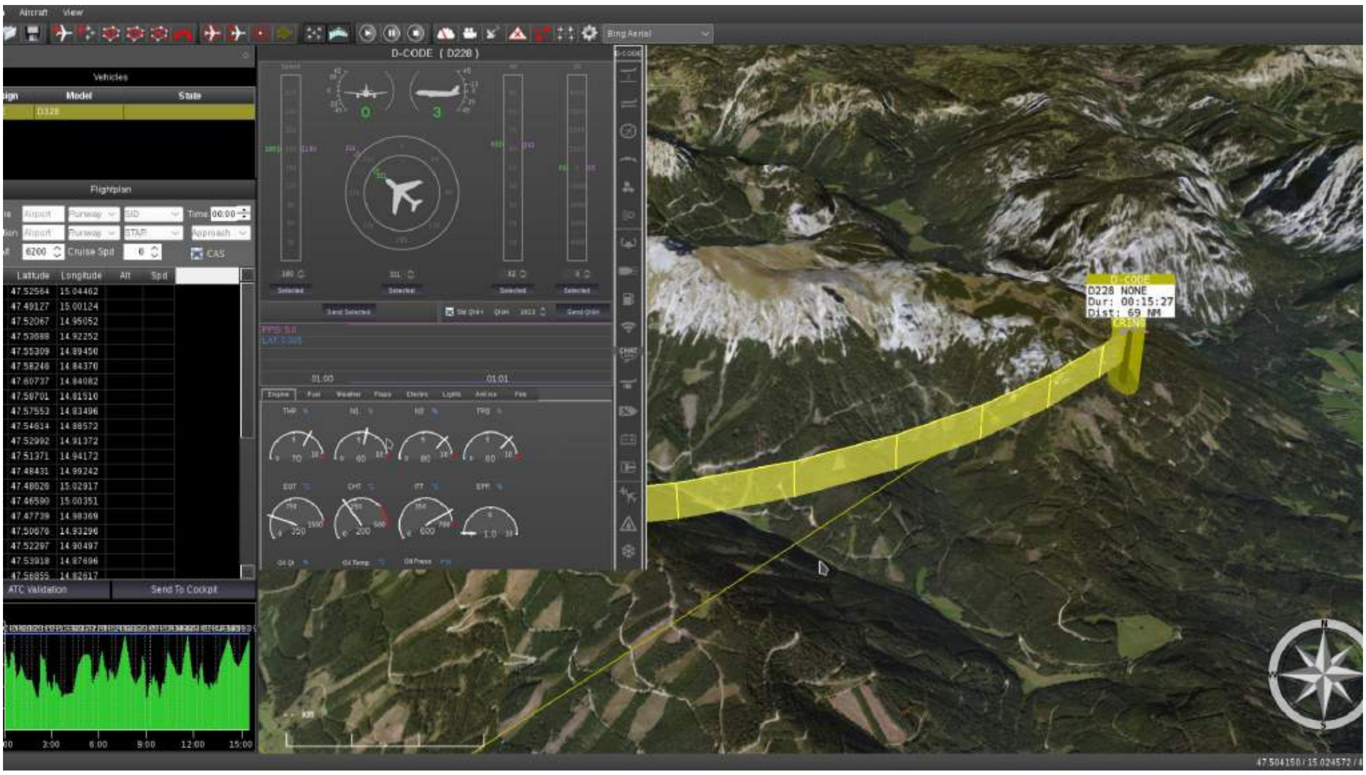
Support C3 decision making

Support communication among stakeholders because of the interactive and intuitive model. Easily visualize extent of hazard, support in decision making for evacuation sites and routes. Determine areas with possible contamination. Enables the testing of effects of optioneering and mitigation measures which will support decision makers.

Airborne and Terrestrial Situational Awareness

This solution is an internal solution.





Team

Real Name

Christian Niermann

Roles

Owner

About

Main D+ contact at DLR

Summary

The solution “Airborne and Terrestrial Situational Awareness” is composed of several individual components and tools, which are integrated into a complete system, ready to be deployed in different scenarios. DLR's optical 3K camera system is integrated into the research aircraft D-CODE, a modified Dornier Do228 with a digital autopilot, which will be operated as a remotely piloted vehicle (RPV) during the selected Trials. The flight path planning and remote control will be executed by the ground control station U-Fly, which is connected to the RPV using a data link for command, control, and communication (C3). The RPV will provide aerial images and send the data over a separate payload data link to the ground system. Based on the aerial images and additional data, information layers relevant for crisis management will be derived and provided as geowebservices and map products using the resources of the ZKI-Tool.

KeepOperational will combine terrestrial sensor data with the information layers derived from the aerial images and provide traffic analysis and route planning capabilities.

Innovation stage

Stage 2: Research and Developement

Readiness

TRL 7

Crisis size

Local

Regional

Cross-border

Large scale

Crisis Cycle Phase

Preparedness

Response

CM-Functions

Open configuration options

Determine the nature of the crisis

After any disaster with visible impacts, aerial and satellite data of the affected area support the determination of the nature and extent of the crisis and facilitate the damage assessment

Conduct monitoring and anticipation

Repeated flights and acquisition of aerial or satellite data can support monitoring (e.g. of remote and large areas) and facilitate change detection analyses

Provide decision support

Aerial imagery as well as derived geo-information (e.g. disaster extent, damage assessment, traffic data and simulation, routing information) can contribute significantly to decision-making

Conduct SAR operations

“Search” part of SAR: Analysis of aerial imagery provides the operator with possible locations of people/groups enclosed in disaster areas or AOIs

Maintain shared situational awareness

Aerial Images assist in up-to-date Information for actors - After initial assessment, progress in relief actions and change in size or extent of the affected area (e.g. retreating water) can be monitored

Conduct damage and needs assessment

After any disaster with visible impacts, aerial and satellite data of the affected area support the determination of the nature and extent of the crisis and facilitate the damage assessment

Plan, organise, and resource transportation logistics

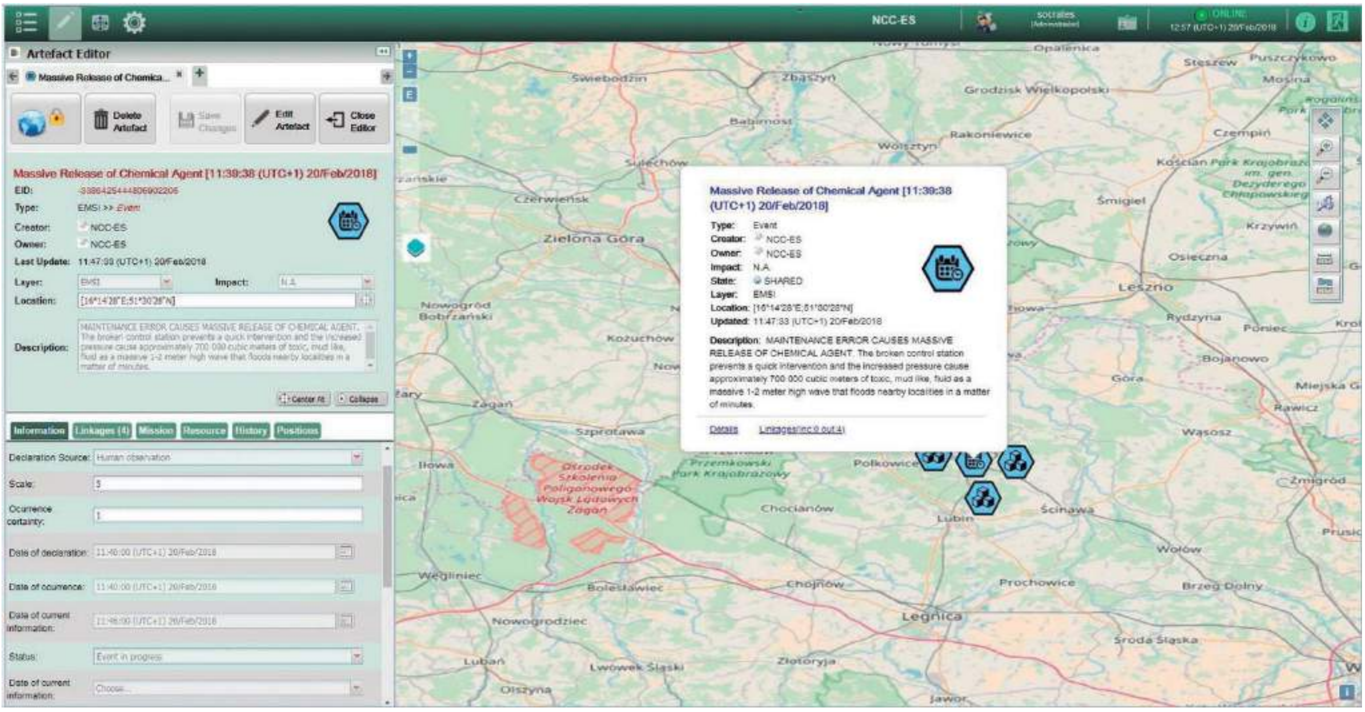
Route planning capabilities can offer optimized routes for (first) responders and logistics

Conduct flights to collect information

Solution integrates an aerial surveillance platform for information gathering.

C3 support solution for Emergency Management Operations Centres

This solution is an external solution.



Team

Roles

Owner

Summary

The solution provides a COP based on a GIS which displays operational information regarding crisis events (or incidents), missions and resources and allows sharing this information with other nodes (coordination centres, command posts, etc.) in the Crisis Management network. It also allows the user to assign missions to other nodes and ask them for additional resources which are needed to deal with the crisis event.

Innovation stage

Stage 4: Early Adoption/ Distribution

Readiness

TRL 8

Crisis size

Local

Regional

Crisis Cycle Phase

Response

CM-Functions

Conduct coordinated tasking and resource management

The solution allows to register the existing resources and updating its status and position, assign resources to missions, assign tasks to other nodes or request them for their available resources, and share all this information with all nodes in the CM network.

Maintain shared situational awareness

The solution is able to gather (collect), store (sustain) and share (disseminate) operational information about the crisis situation (regarding crisis events, missions and resources) and exchange it with other nodes in the CM network.

Support C3 decision making

Supports crisis managers and commanders in decision making by sustaining and sharing the COP with relevant information about crisis events, on-going missions, available (and in use) resources, etc.

CrisisSuite

This solution is an external solution.

Video link: https://youtu.be/0VWgjxk7Meo

Team

Real Name

Ruud van den Beukel

Roles

Owner

About

Solution owner, co-founder and CEO of Merlin Software

Summary

The main objective of CrisisSuite (online crisis management software) is to enable organisations to successfully manage information during a crisis. CrisisSuite is a tool that supports the netcentric working methods of crisis teams by creating an universal picture of the crisis and share it horizontally and vertically with all the other teams in the crisis organisation. CrisisSuite also assists in maintaining an effective crisis meeting structure and it decreases the administrative workload for the people managing the crisis.

The main functions of CrisisSuite are:

Logbooks & Actions –Log and process the meeting information, to get a grip on the crisis and remain in control

SitRep's–Updated Situation Reports (SitReps) give the latest overview of the state of affairs regarding a specific topic

Attachments –Share images and documents with all people involved in a particular crisis

Maps –Give a geographic overview of the relevant information (still in development)

Organisation–Make it as easy as possible for people to reach out to another

Plans –Share plans and scenario's so that everyone involved knows what is expected

Innovation stage

Stage 5: Market Growth

Readiness

TRL 9

Crisis size

Local

Regional

Cross-border

Crisis Cycle Phase

Preparedness

Response

Trials

DRIVER+ Trial 2

Trial 2

Trial type

Trial

CM-Functions:

CM function 1

Maintain shared situational awareness

CM function 2

Disseminate COP and assessments

CM function 3

Communicate operational information across chain of command

CM function 4

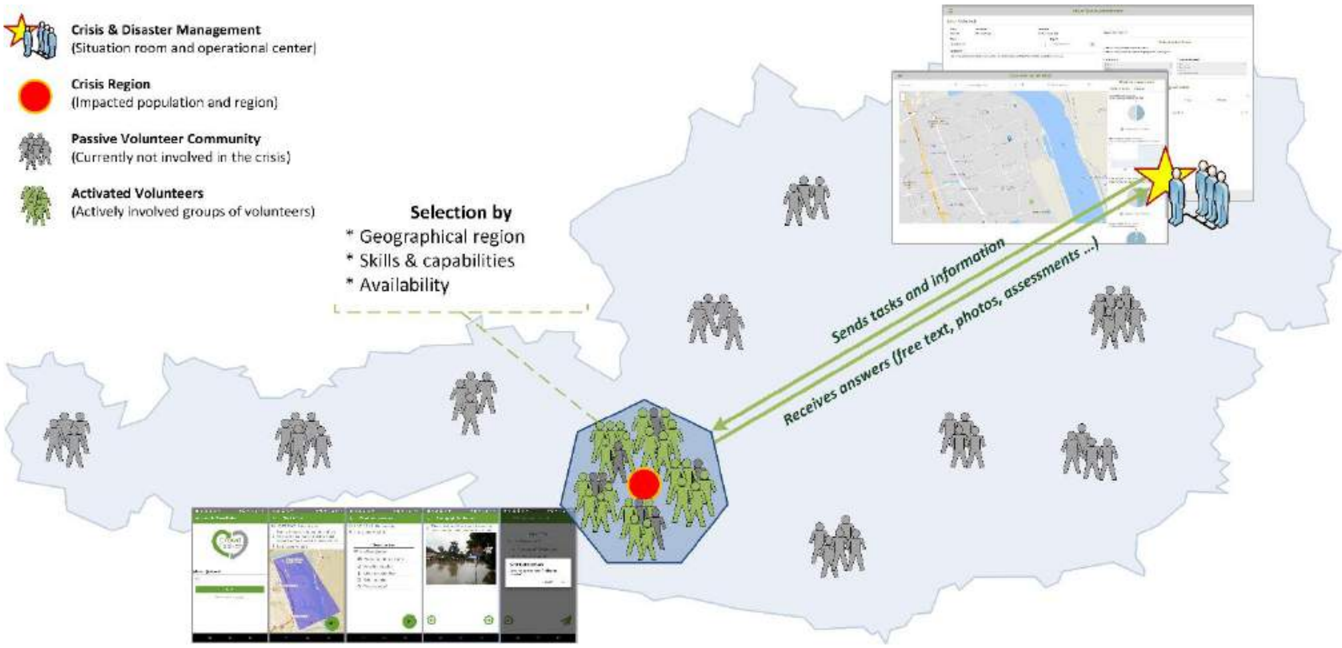
Support C3 decision making

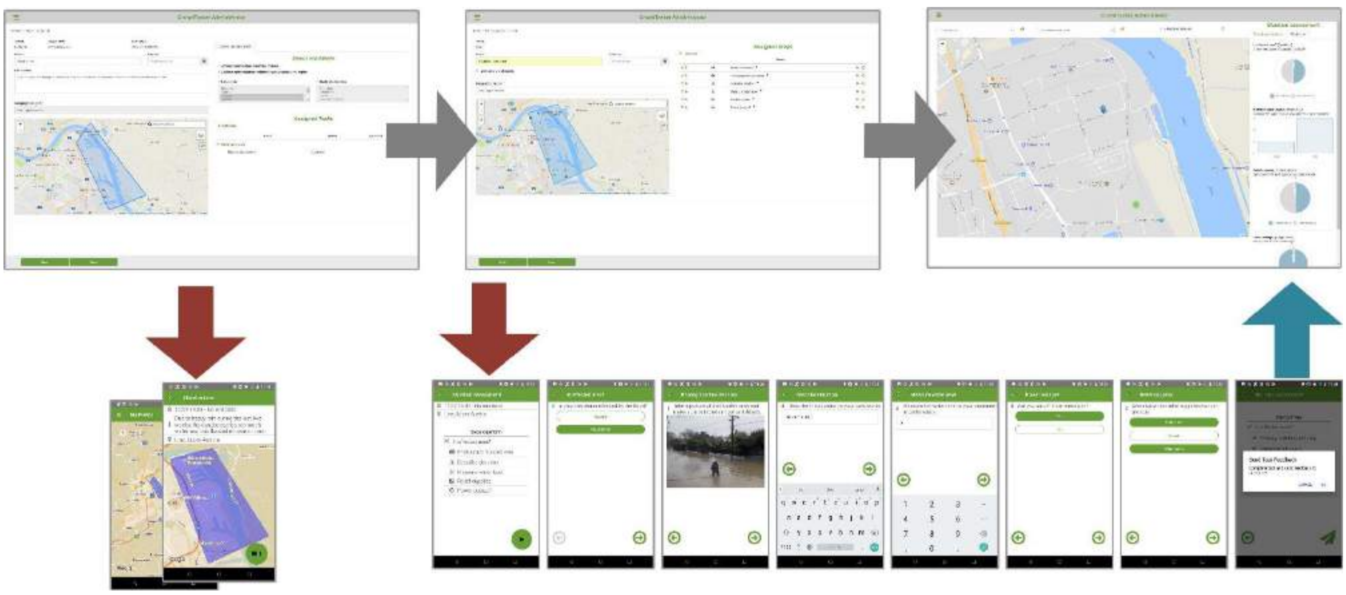
CM function 5

Provide situational awareness, share COP

CrowdTasker

This solution is an internal solution.





Team

Real Name

Georg Neubauer

Roles

Owner

About Main AIT contact for DRIVER+

Real Name

Daniel Auferbauer

Roles

Team

About

Team Member

Summary

CrowdTasker enables crisis managers to instruct large numbers of non-institutional volunteers with customizable tasks.

The received feedback is evaluated and visualized and provides crisis managers with a detailed overview of the situation, which is used in turn to trigger adequate disaster relief services.

Information is provided by volunteers that are already at a disaster site allowing to exploit numerous benefits:

Innovation stage

Stage 4: Early Adoption/ Distribution

Crisis size

Local

Regional

Crisis Cycle Phase

Preparedness

Response

CM-Functions

Task and manage volunteers using smartphone app

Conduct coordinated tasking and resource management

This solution addresses the management of the informal volunteers ("citizen as a volunteer")

Manage spontaneous volunteers

Manage spontaneous volunteers

CrowdTasker provides a way to manage a large group of non-institutional volunteers with a help of a a smartphone app.

Task volunteers using a smartphone app

Task volunteers

CrowdTasker provides appropriate and relevant tasks to selected pre-registered volunteers, at the right time and location and according to their respective skills.

Organise volunteers during recovery

Organise volunteers and communities for recovery

Through it's tasking system, CrowdTasker offers options to organise individual spontaneous volunteers as well as existing and emergent communities in all phases of the disaster management cycle.

Inform app users on hazards and respective services

Maintain public awareness on hazards and respective services

This solution provides realtime warnings, information and guidelines to the public. Recipients can be selected based on their location to warn them of hazards in certain areas or provide guidance once they enter a pre-defined area.

Use volunteers for monitoring and data collection

Conduct systematic monitoring and data collection

This solution addresses the use of the informal volunteers as sources of information ("citizen as a sensor")

Lightweight and scalable volunteer organisation

Establish organisation for spontaneous volunteers

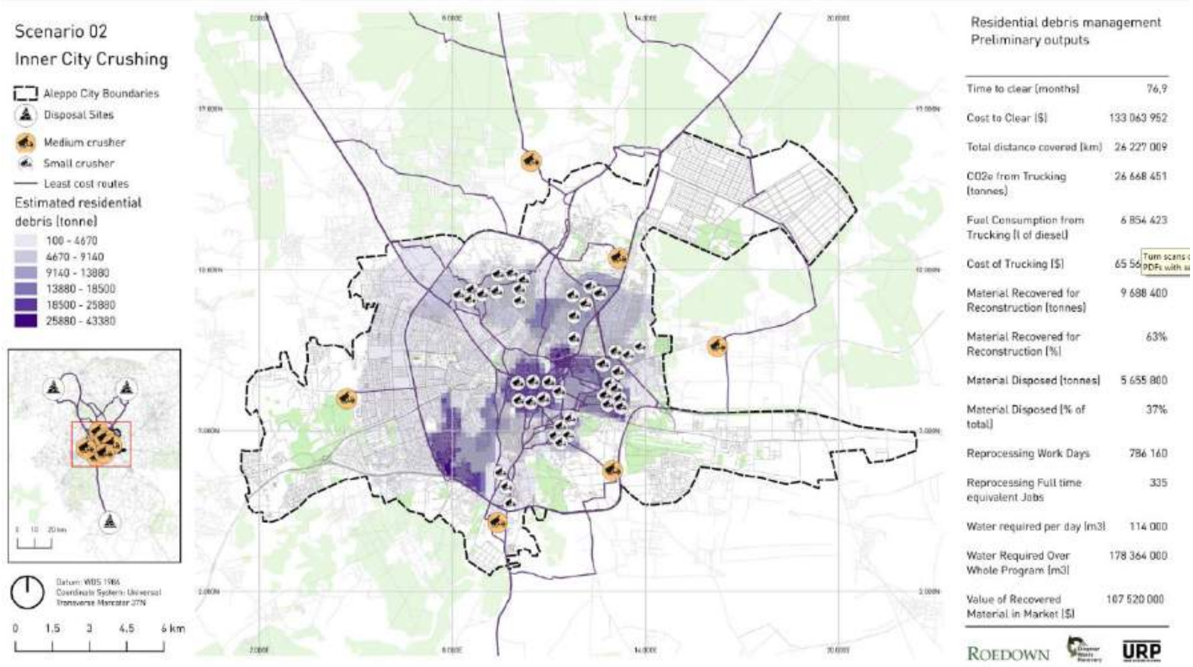
CrowdTasker provides features to support and enable spontaneous volunteer organisations by installing and registering via mobile app.

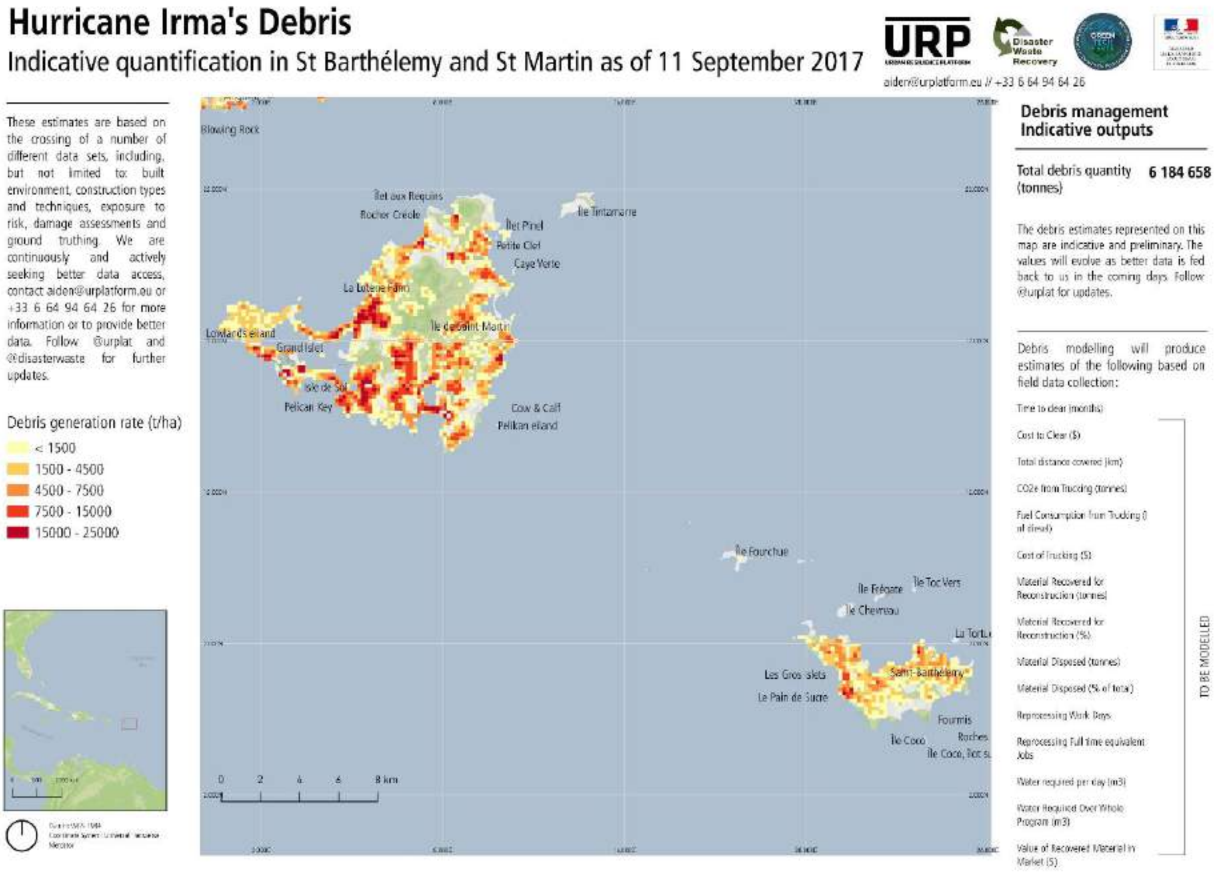
Communicate directly with media and citizens

CrowdTasker provides a scalable "one to many" and "many to one" communication between the crisis managers and the non-institutional volunteers.

Debris Tool

This solution is an internal solution.





Team

Real Name

Aiden Short

Roles

Owner

About

Solution owner

Summary

The Debris Tool is a software based solution designed to amalgamate various defined inputs from the field, historic survey data and other sources, for the prediction and modelling of waste and debris removal options in a post-crisis environment. The tool also adapts to applications in pre-emptive or prevention scenarios by providing an accurate, up to date map that can be used as a keystone in preparedness planning. By the modelling of the standard management system the marginal benefits of any potential improvements are pivotal in decision support particularly where logistics and access are a concern. In a live crisis context, the maps are invaluable in aiding resource planning and analysis of humanitarian logistics options.

Innovation stage

Stage 5: Market Growth

Readiness

TRL 6

Crisis size

Local

Regional

Crisis Cycle Phase

Preparedness

Recovery

Mitigation

CM-Functions

Restore the solid waste collection system

Restore the solid waste collection system

By mapping and granting a visual overview of the quantities of waste involved and the limitations of the physical environment. The solution can help model different cleanup scenario's and analyse the resulting resources required including economic, financial and technological. The responsible authorities can then strategically implement the most efficient restoration plan.

Transport debris and waste

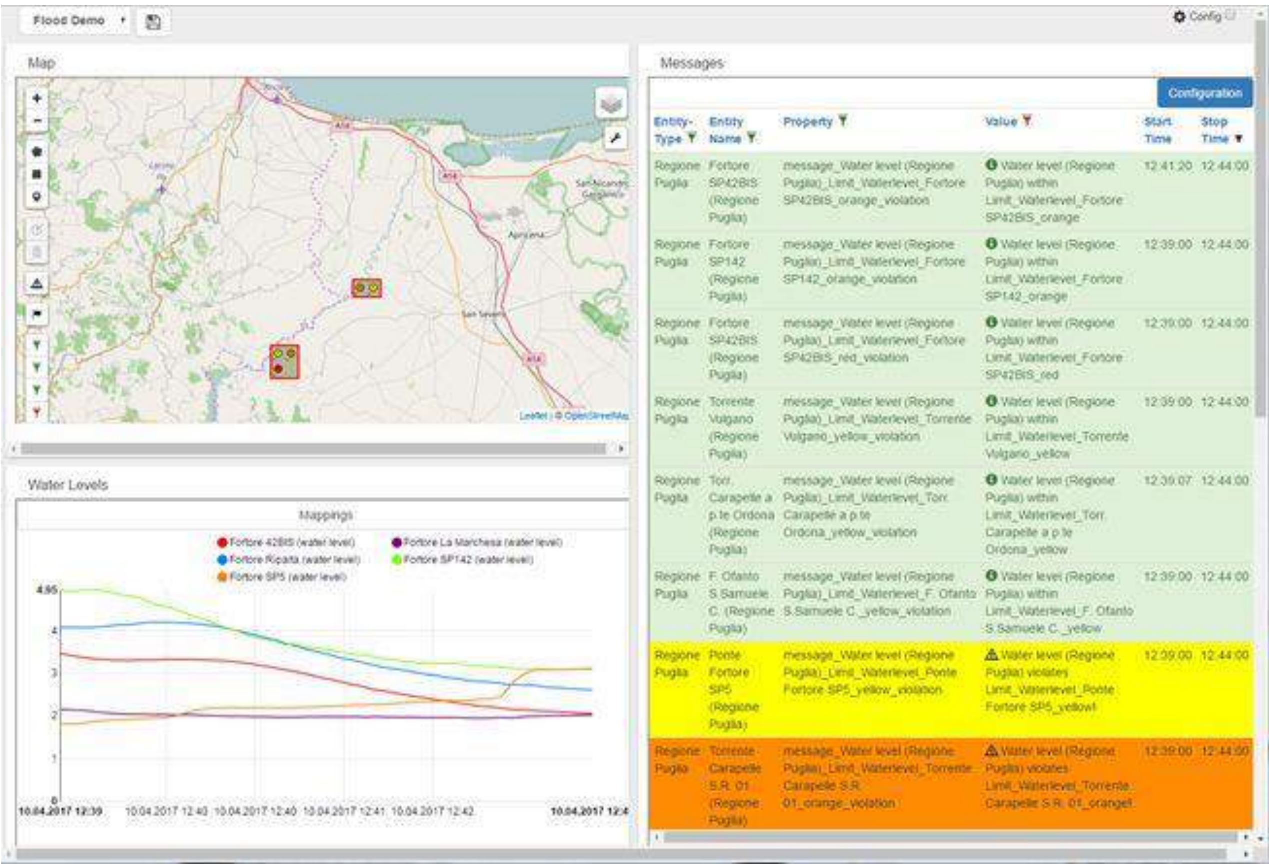
In a similar process to the rational for the CM function above, the solution offers a practical and efficient planning tool that allows waste and debris to be removed in a sustainable and transparent way. Different predefined options for transport and removal can be modeled and compared.

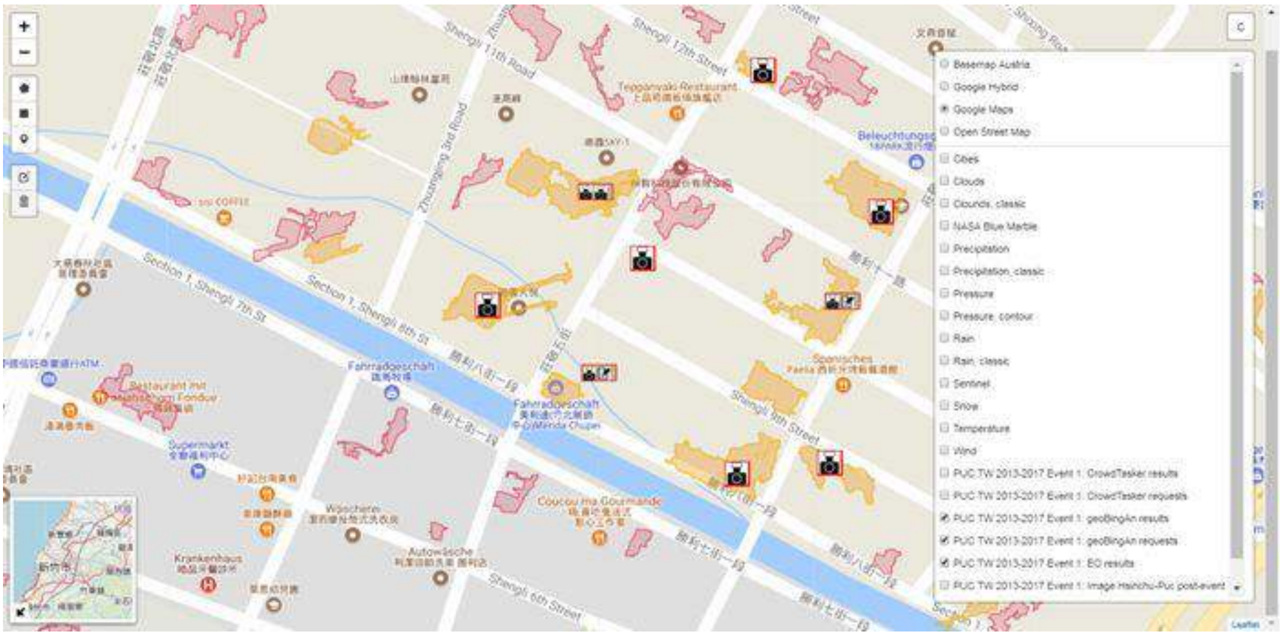
Remove damaged structures and debris

By accurately predicting the quantity and quality of debris different demolition and cleanup scenario's can be modeled to test out the practicality and requirements of a structural removal plan. This takes into account the architectural make-up of a structure as well as the limitations of a given terrain.

Emergency Mapping Tool (EMT)

This solution is an internal solution. The EMT tool was only recently added to the internal DRIVER+ solutions and was therefore not yet included in the planning for solution integration.





Video link: https://youtu.be/dfUR8lkE3VU

Team

Real Name

Kutschera Peter

Roles

Owner

About

Main EMT developer

Real Name

Georg Neubauer

Roles

Team

About

DRIVER PM at AIT

Summary

EMT faciliates seamless exchange of information for stakeholders in the crisis management.

Information is presented to the EMT users in the most suitable and comfortable way, taking into account their background and role in the crisis management. This is accomplished by a user domain tailored message representation. Each user organisation can see the same data in its usual working environment, e.g. using their specific icons, colours, and textual representation.

By closing the information gap EMT enables an efficient management of a crisis, even in ad-hoc multi-organisational and multilingual settings.

- EMT is not meant to be a full and complex COP tool, that can only be used by specialists and require extended training. It is designed for simple usage by everyone.
- EMT is accessed using a web browser and requires no local installation.
- Local installations can be provided and configured to exchange information with other EMT instances if needed for redundancy and to overcome internet outages.

EMT provides a multi-user web interface that can be configured to present crises relevant information to users in their own language and using the users' domain specific textual representation, text-colours, icons, map features, etc., while filtering out information that is of no concern to them. Information that is typically shared through EMT includes the number, type, location and status of available resources, sensor values, hazard areas, alerts, commands sent and fulfilled and situation reports. Also references to external data like photos or map layers can be handled.

EMT users can enter information into the system in their own language. This information is translated on the fly and shared with other users. Translation relies on predefined "message templates" consisting of predefined text that can be translated but also variables that users can enter on the fly and that are forwarded "as is" to other users.

Innovation stage

Stage 4: Early Adoption/ Distribution

Readiness

TRL 6

Crisis size

Local

Regional

Cross-border

Crisis Cycle Phase

Response

Recovery

Mitigation

Previous Resume Next

CM-Functions

EMT CMF-02 C3 Communications and information support

Provide communications and information support to C3

EMT facilitates a seamless exchange of relevant information for CM professionals at all levels of command and across all organisations included in the management of a crisis. Provision of tailored information to users from other organisations can be easily achieved, by pre-defining which information they should receive (and how) and then providing them with the EMT URI and login information.

EMT CMF-01 Crisis communication across organisational and language barriers

Establish crisis communications capabilities

EMT provides a mean to *communicate across organisational and language barriers, without a need to fully harmonize the terms, symbols or formats of the messages*. This is accomplished by user domain tailored message representation. Each user or organisation can see the same data but the presentation of the data can be modified in terms of the icons, colours, and textual representation.

EMT CMF-04 Cross-X dissemination and information sharing

Set-up dissemination and information sharing

EMT provides a mean to share information across organisational and language barriers, without a need to fully harmonize the terms, symbols or formats of the messages. This is accomplished by user domain tailored message representation. Each user organisation can see the same data but with there usual icons, colours, and textual representat*ion*

EMT CMF-03 Regulated access to CM communication and information

Regulate access to CM communications and information

EMT can be configured to provide each user with the information this user needs, while blending out the information they do not need or aren't allowed to receive. Likewise, the users can only generate pre-defined types of messages and the choice of available messages can depend on the user profile.

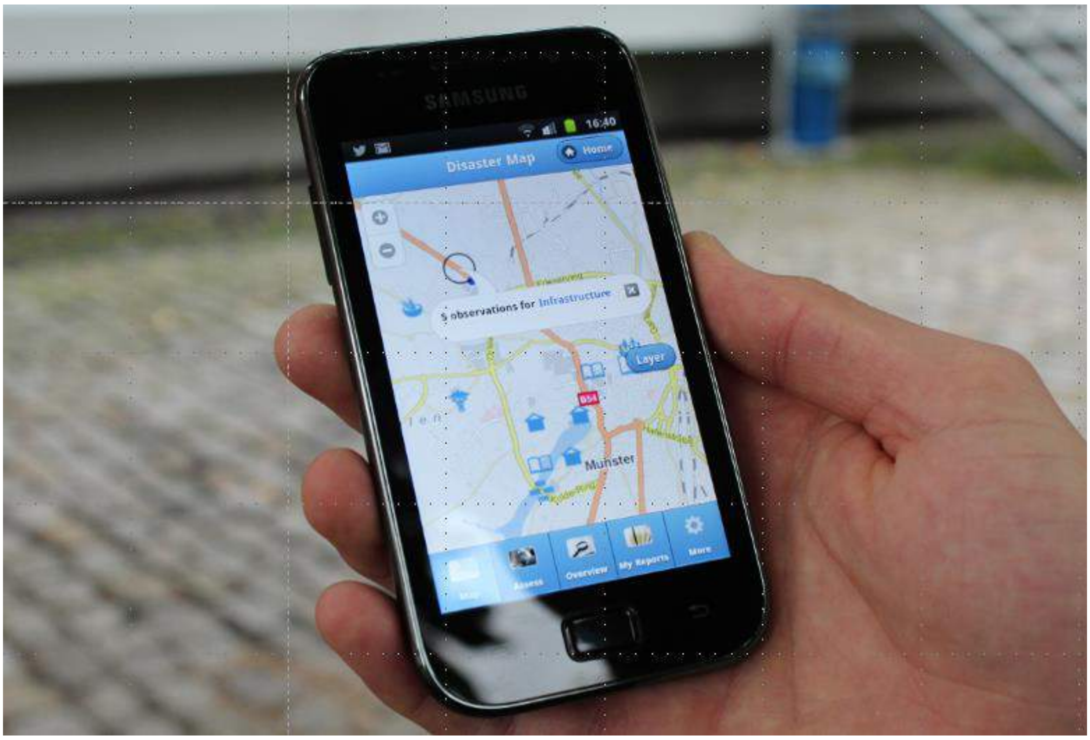
EMT CMF-05 Enforced communication and information exchange policy

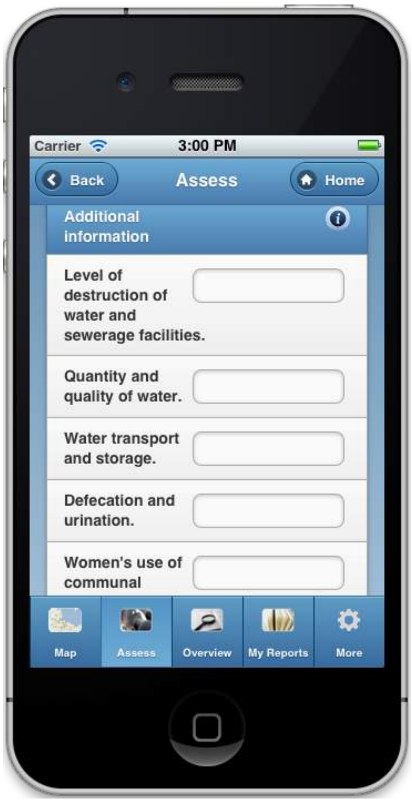
Develop communications policy

EMT relies on existence of the pre-defined message templates and user or organisation profiles and assures that the information is only shown to intended recipients, based on their user profile.

GDACSmobile

This solution is an internal solution.





Team

Real Name

Michael Middelhoff

Roles

Owner

About

Solution owner

Summary

GDCASmobile is a support platform for volunteers and crisis commanders in collecting and sharing situational awareness information. During and especially in the first phase after a major sudden-onset disaster, situation assessment is of high importance for crisis management. The solution supports this activity in establishing an additional open communication channel between the affected population and responder organizations. Using the mobile application, public unregistered users can enter short observation reports on damages, supply needs or available resources, which are collected and reviewed by a control team. The creator can categorize the report based on its topic and add information like geo-location, images, text and others. Special report templates can be predefined and adapted to the information needs for each topic category. Registered responders can publish these reports or add additional information, like evacuation points, shelter locations and others. Thereby, the tool aims at supporting two main target groups having different rights and roles: people concerned with disaster relief (e.g. Command and Control Center) and the (affected) population. In order to control the information flow and to enable quality management on the collected data, a central server interface manages a review process for each observation. Thereby, professionals and trusted informed reviewers can decide which information is accepted as valid and set the visibility for each target group.

"GDACSmobile" is linked to the "Global Disaster Alert and Coordination System (GDACS)". The initial delveopment was motivated together with the "Joint Research Center (JRC)" of the EU Commission.

Innovation stage

Stage 3: Initial Piloting

Crisis size

Local

Regional

Crisis Cycle Phase

Response

CM-Functions

Maintain population's operational awareness

CM professionals can share relevant information about the crisis situation with the app users on to guide them to shelters or evacuation points.

Provide communications with volunteers

App users can interact in an undirected communication with CM professionals and those can share public information on the crisis.

Conduct damage and needs assessment

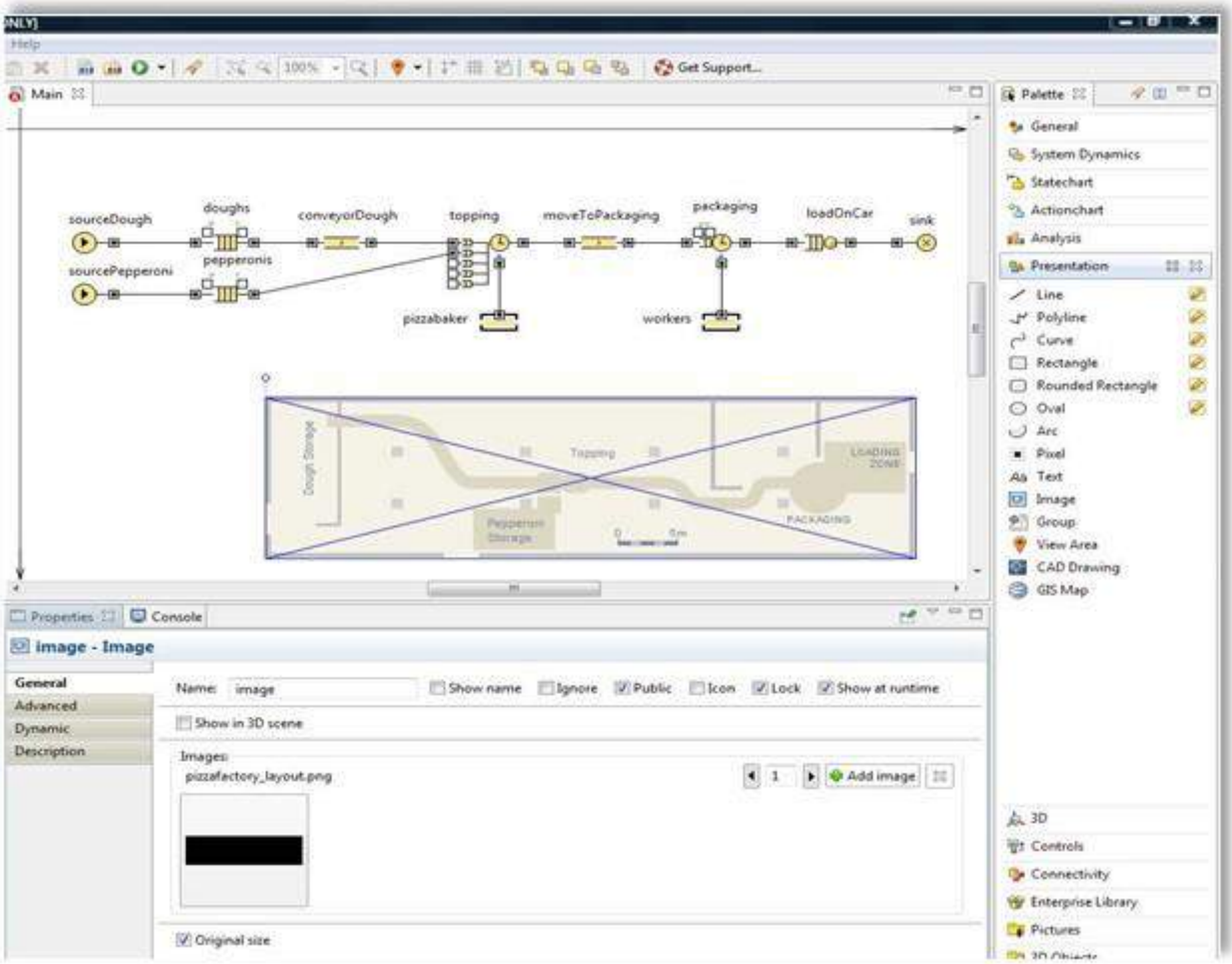
CM professionals can evaluate reports from volunteers to assess information on urban infrastructure.

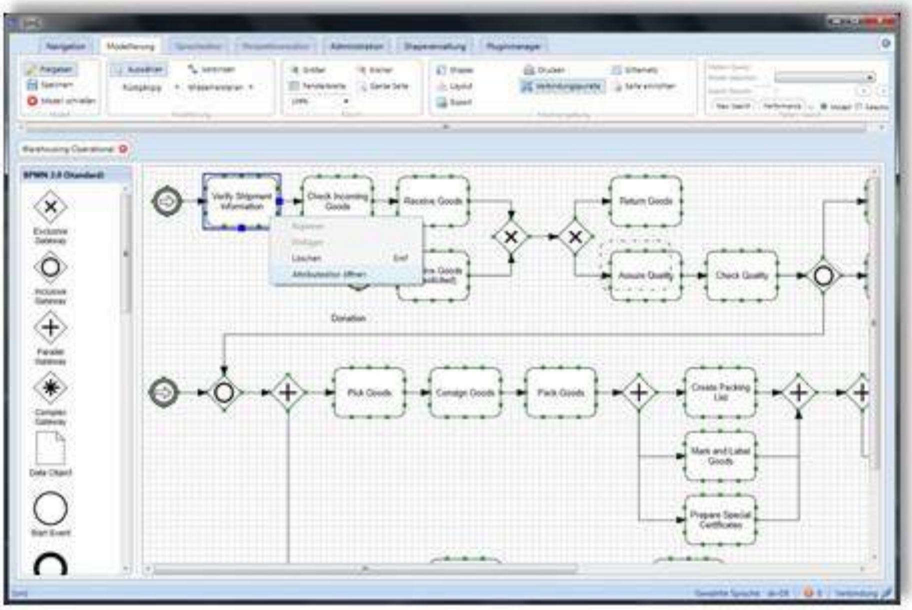
Monitor the affected area

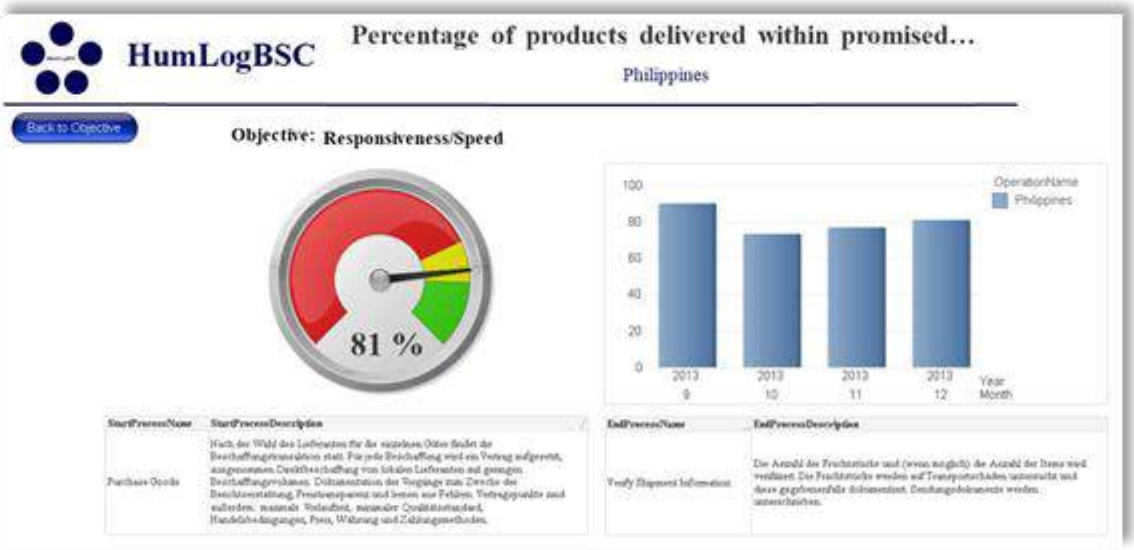
App users can send reports on current developments in the affected area to CM professionals.

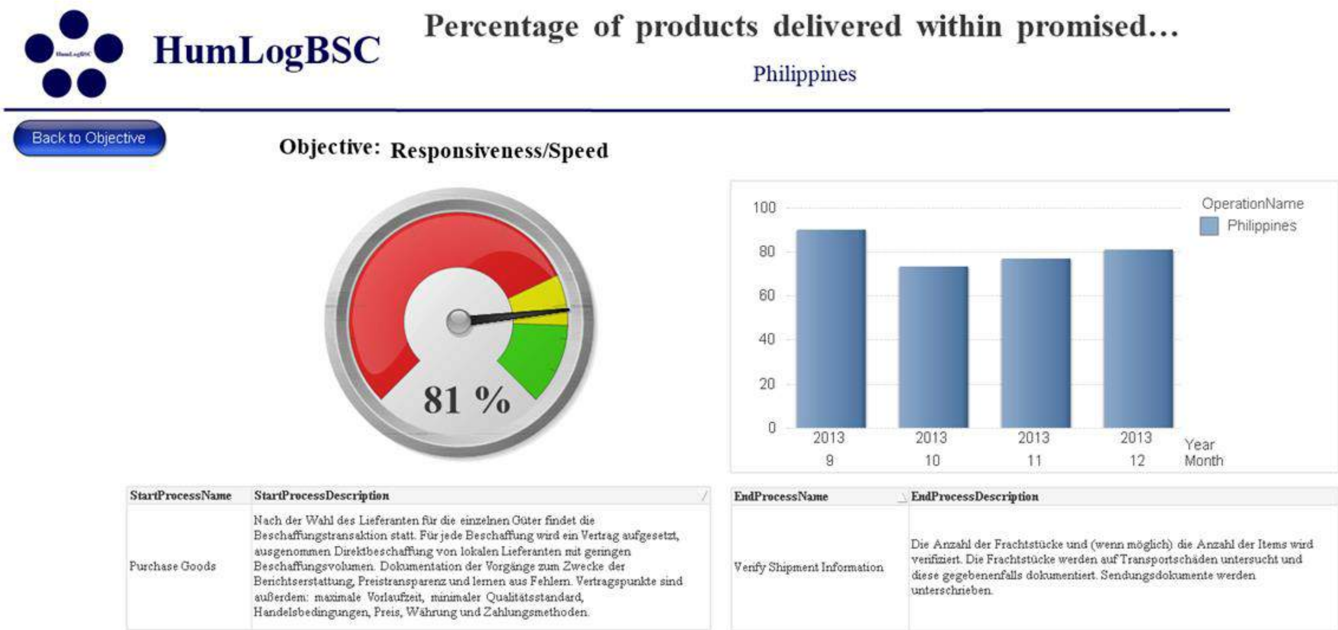
HumLogSuite

This solution is an internal solution.









Team

Real Name

Michael Middelhoff

Roles

Owner

Summary

HumLog Suite is a performance assessment platform for logistics processes in crisis management. It can operate on both current operational logistics networks and fictional or planned network configurations. Based on this functionality, it offers strategic planning support as well as tactical and operational decision support by assessing and comparing the network performance under given situations and realistic crisis management actions. The name “HumLog” refers to the humanitarian logistics context as the planning, implementation and controlling of efficient, cost-effective flow and storage of goods, materials and equipment as well as related information, from point of origin to point of consumption for the purpose of meeting the beneficiaries’ requirements under given resource capacities.

The solution implements this objective from three distinct but interconnect viewpoints:

- HumLog[em] is a modelling tool able to support various modeling languages. Current or planned operational processes are recorded and designed in here. It can be used for the application of reference models as well as for the model reporting and pattern search.
- HumLogBSC is an IT supported performance measurement system for humanitarian logistics. It is a processdriven and modular Balanced Scorecard and based on the reference model of humanitarian logistics. It utilizes the modelled processes to derive a set of performance values.
- HumLogSim is an adaptable simulation environment for discrete eventbased and agent-based simulations, which represent crisis management activities within and between humanitarian organizations to achieve a defined objective, while assessing the overall performance.

HumLog Suite can be used in different settings and operation levels of crisis management. On a strategic level, one capability is the support of creating or updating a relief network of one or multiple responder organizations. Using the process modelling features, one can design and evaluate crisis management operations to plan daily operations or to prepare emergency actions in case of a crisis event. Beyond this, the simulation environment can analyze different possible network configurations in terms of facility locations, relief goods stock values, warehouse capacities, human resources and others. By executing so-called “what-if” scenarios, the performance of a planned network can be simulated under a fictive crisis event, which could for example make use of historical crisis event data to represent a realistic scenario.

On a tactical level, these capabilities can also be utilized to compare possible execution plans and to estimate their expected outcomes. Given a relief network of one or multiple responder organizations in an actual crisis event, crisis management commanders can test different strategies in addressing the crisis without any influence on the realworld resources. The simulation capabilities can thereby be freely adapted to the needs of an individual organization, like for example the construction of sandbag barriers, evacuation of the population or allocation and transportation of human resources. The simulation will take all available resources and capacity restrictions into account to provide a complete and sophisticated execution protocol, which can then be used by command staff to compare execution strategies.

Innovation stage

Stage 3: Initial Piloting

Readiness

TRL 7

Crisis size

Local

Regional

Cross-border

Large scale

Crisis Cycle Phase

Preparedness

Response

Recovery

CM-Functions

Provide decision support

Provide decision support

Users estimate the effects of decision options on the logistics network and compare the outcomes.

Estimate resource requirements

Users can setup fictional scenarios to assess the effect of changes in the resources available.

Open configuration options

Conduct transportation logistics

Users can plan several interconnected transports based on location-based needs of any kind of resources.

Identify the components of crisis logistics support

User can model the current processes within and between organizations.

Identify and analyse bottlenecks

User can identify which resources are needed at which locations to fulfil the current logistics objectives.

Determine materiel requirements

Users can design and test logistics scenarios to evaluate resources needed to operate the scenario successfully.

Develop decision support systems

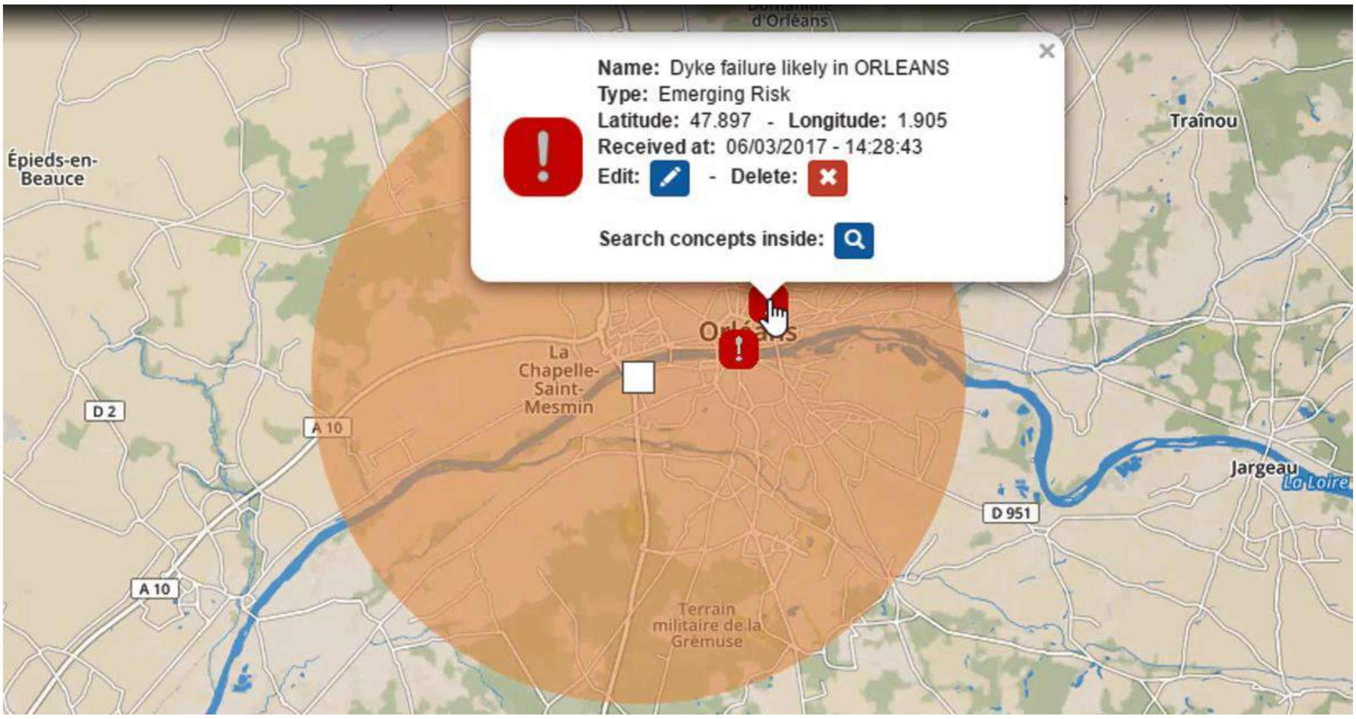
Users can adapt the simulation suite to represent a specific network for future logistics decision making.

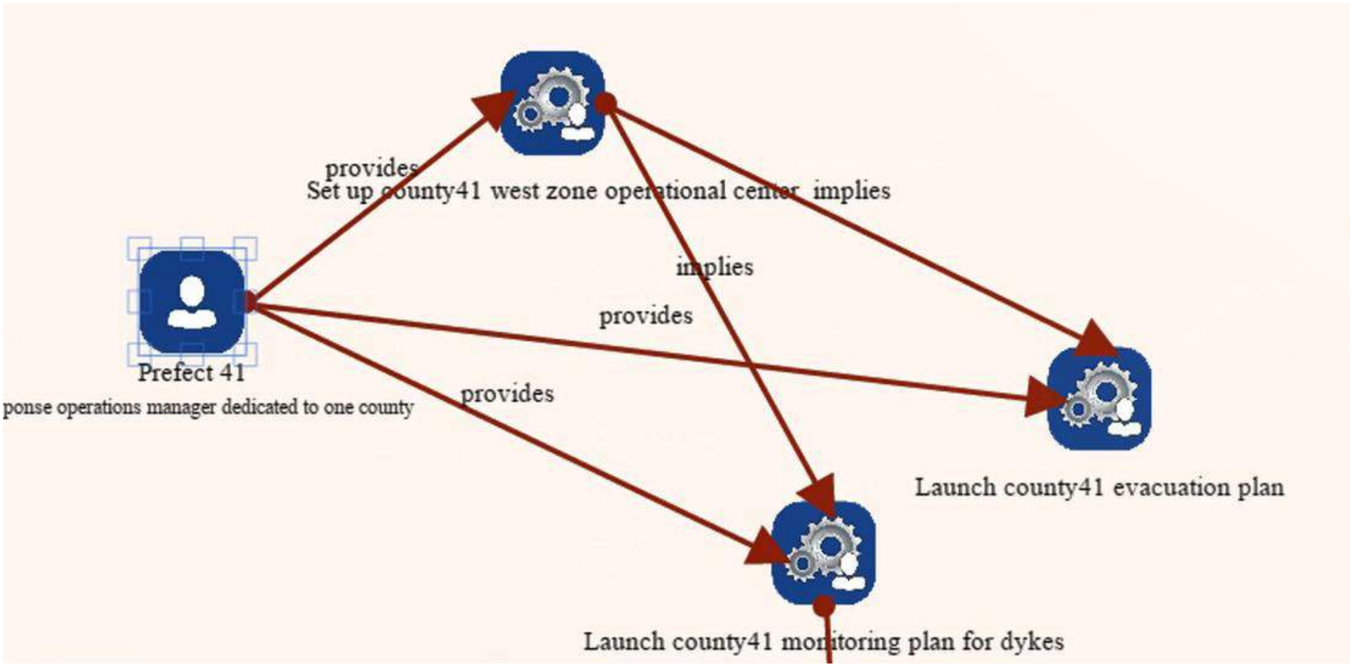
Select storage and distribution facilities

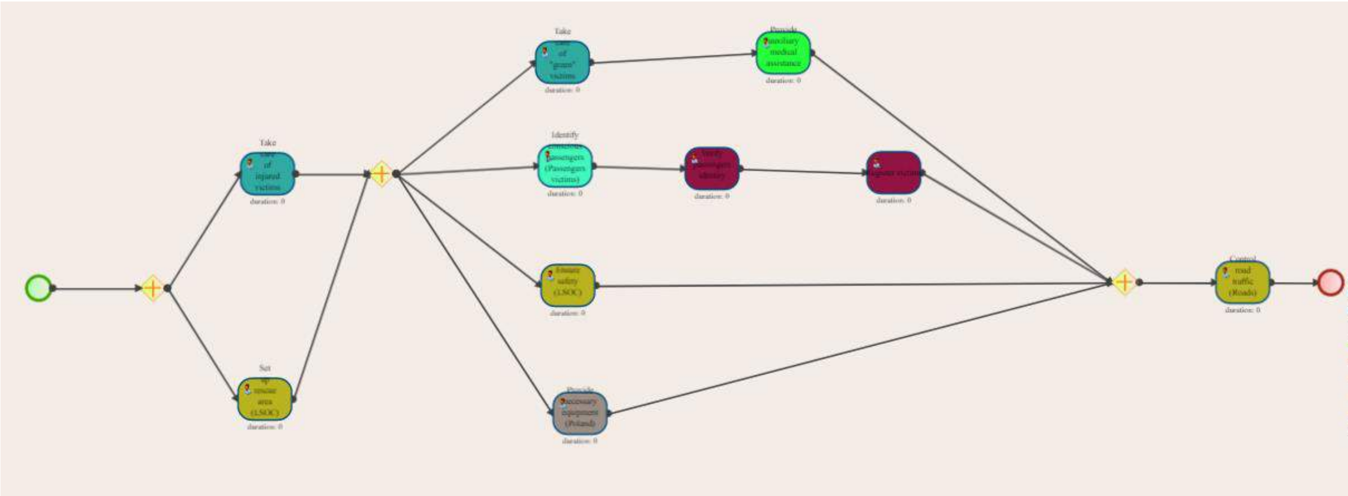
Users can compare options of available facilities and is provided with the optimal selection.

IO-DA

This solution is an internal solution.







Team

Real Name

Sebastien Truptil

Roles

Owner

Team

About

Main owner of the solution

Summary

On the one hand, the information about the crisis situation is brought thanks to the use of dedicated modelers: · Partner modeler: This modeler allows the crisis manager to model crisis management stakeholders that can be mobilized in case of crisis situation and their capabilities. · Context modeler: The models obtained with this modeler present characteristics of the impacted systems, including goods, people, intrinsic risks and any other specific feature. · Objective modeler: This third modeler is dedicated to describe precisely the crisis situation. The obtained models contain mainly objectives to satisfy (as missions) which are basically based on effects to treat (like injured people to evacuate or fire to extinguish) or risks to prevent (like building collapse or tank explosion). This model could be seen as a map on a GIS. On the other hand, the IO-DA tool embeds an inference engine, which allows the user to apply different strategies to deduce, from the previous models, collaborative process models. Regarding deduction strategy, for the moment, only one simple strategy has been implemented as a set of business rules to deduce collaborative process models, but the current work is about implementing more complex strategies embedding doctrine and law principles.

Innovation stage

Stage 3: Initial Piloting

Readiness

TRL 5

Crisis size

Local

Regional

Crisis Cycle Phase

Preparedness

Response

CM-Functions

Conduct coordinated tasking and resource management

Conduct coordinated tasking and resource management

Based on the model of the situation (Cop), this solution aims to deduce a collaborative process in order to support the orchestration of the stakeholders' activities. Thus it could be helpful to define, prioritise and assign tasks.

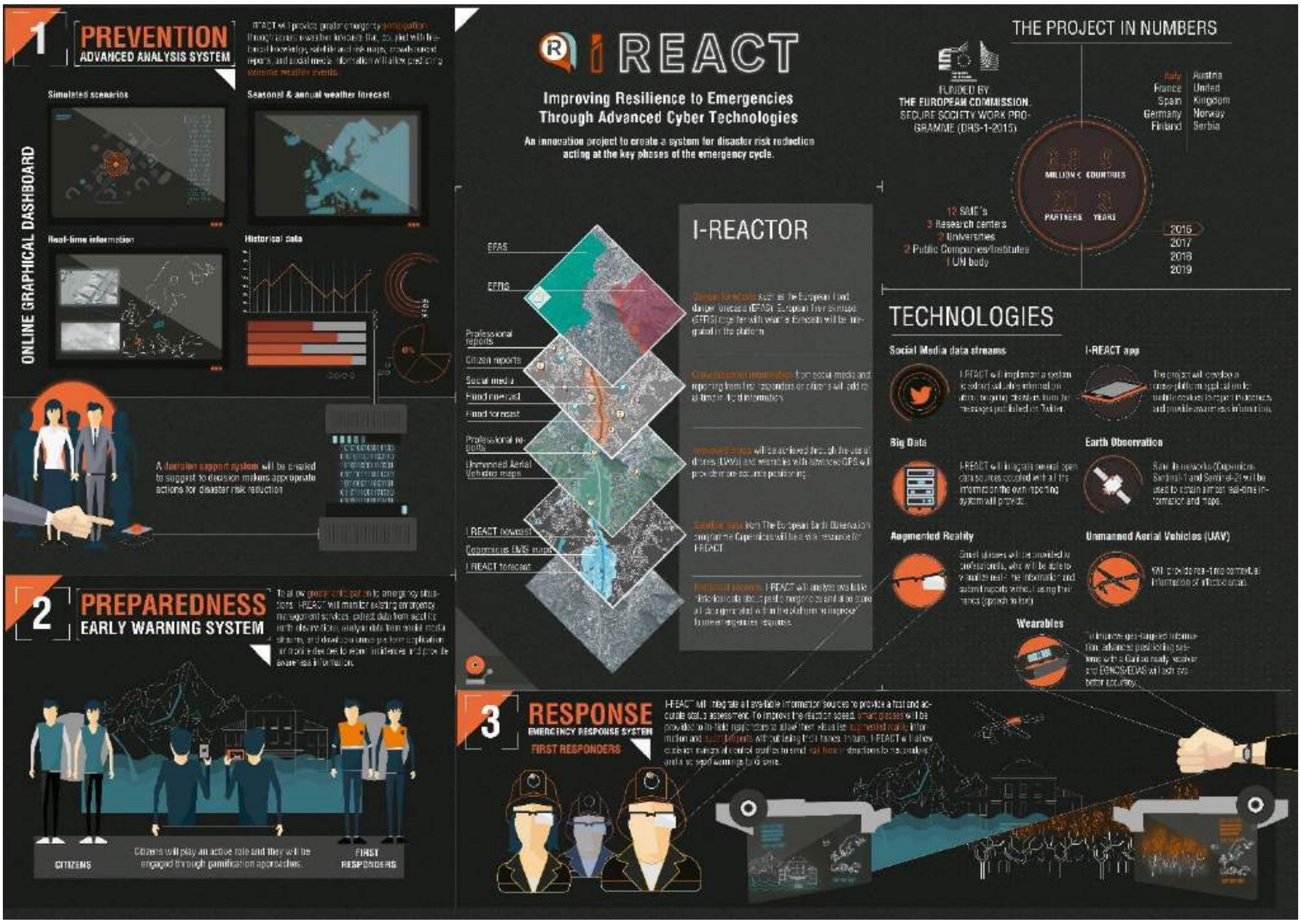
Provide decision support

Provide decision support

The idea is to help the deduction of Crisis response activities based on situational information.

I-REACT

This solution is an external solution.



Video link: https://youtu.be/4t5ScCh6XU0

Team

Real Name

Claudio Rossi

Roles

Owner

Summary

Due to climate change, floods, wildfires and other extreme weather events are becoming more frequent and intense. This scenario poses a challenge for current risk management systems.

I-REACT aims to develop a solution through the integration and modelling of data coming multiple sources. Information from European monitoring systems, earth observations, historical information and weather forecasts will be combined with data gathered by new technological developments created by I-REACT. These include a mobile app and a social media analysis tool to account for real-time crowdsourced information, drones to improve mapping, wearables to improve positioning, as well as augmented reality glasses to facilitate reporting and information visualization by first responders. With this approach I-REACT will be able to empower stakeholders in the prevention and management of disasters. Citizens will be involved in reporting first-hand information, policymakers will be supported in the decision making process, and first responders will be equipped with essential tools for early warning and response.

Overall, I-REACT aims to be a European-wide contribution to build more secure and resilient societies to disasters.

Innovation stage

Stage 3: Initial Piloting

Readiness

TRL 6

Crisis size

Local

Regional

Cross-border

Large scale

Crisis Cycle Phase

Preparedness

Response

CM-Functions

Social media engine

Conduct systematic monitoring and data collection

People share emergency related data on social media and this happens in all kind of emergencies (local up to international).

The I-REACT social media data engine is able to automatically gather Twitter data, classify the, and detect the occurrence of emergency events.

It features a named entity module the is able to improve the localization of tweets, which does not often contain exact location (lat,lon)

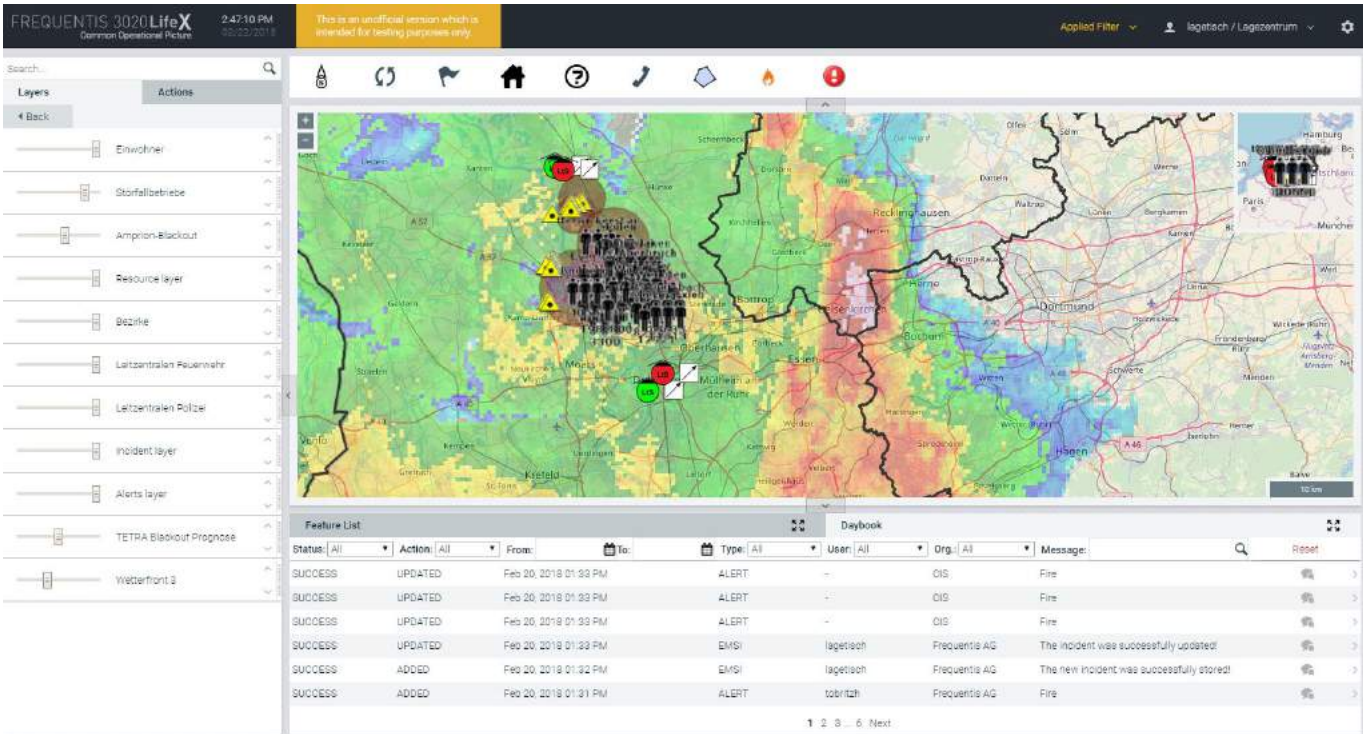
Crowdsourcing

Conduct systematic monitoring and data collection

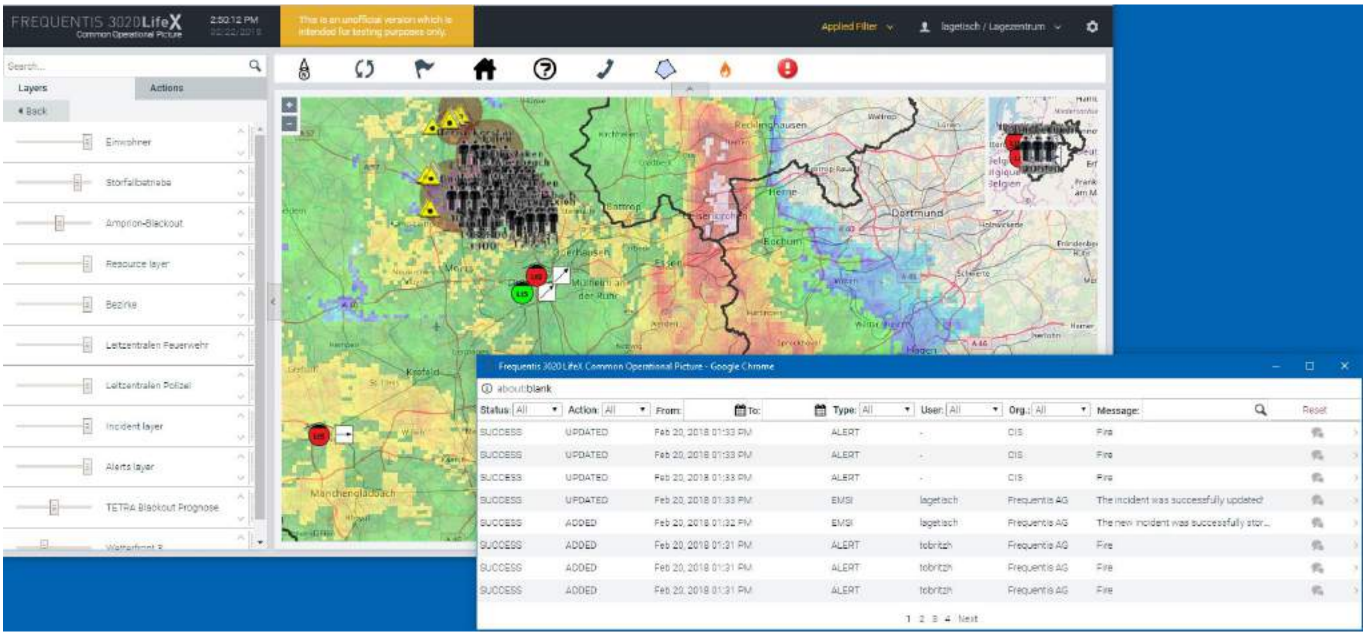
In-field agents, volunteers and citizens can provide valuable real-time multimedia in-field observations (reports). I-REACT features a crowdsoucing solution through a gamified mobile application. The I-REACT mobile application allows spontaneous reporting, reports validation (both peer-to-peer for citizens and official for authorities), it delivers geolocated alerts and warnings, and supports command and control features for professionals.

LifeX COP

This solution is an internal solution.



The picture above shows the standard layout of the solution.



The picture above shows the layout prepared for a multi-monitor setup of the solution.

Team

Real Name

Ludwig Kastner

Roles

Owner

About

Solution owner

Summary

LifeX COP is a web-centric multi-user solution developed by FRQ to address the lack of a Common Operational Picture in the field of Crisis Management. LifeX COP is able to collect information from various data sources (static or dynamic) and present them in a map-centric user interface.

This information can be grouped in layers which can be arranged to improve the visibility of the data. Additionally, information can be filtered so non-interesting data can be hidden from the user view. LifeX COP enables each user to add comments in a logbook that can be reviewed at any time. Information is presented both in a map view and list view. In terms of visual design, the graphic user interface is very dynamic allowing the user to decouple windows (map, logbook and list) to be arranged in a multi-monitor operation center. Technically, the LifeX COP handles the following formats input formats:

Innovation stage

Stage 4: Early Adoption/ Distribution

Readiness

TRL 6

Crisis size

Local

Regional

Crisis Cycle Phase

Preparedness

Response

Recovery

Mitigation

CM-Functions

Determine the principles of information exchange

Semantic mapping of codes can be configured, so simultaneous use of different icon sets for different user groups is possible.

Map the hazards per geographic area

The COP GUI presents in a map all information related to an event: incidents, alerts, resources, observations and sensor data; added manually and/or automatically. Information is organized in layers that can be changed individually (show/hide, sort and set transparency).

Detect pending emergencies and provide early warning

The COP receives CAP messages from external systems (e.g. C2, Sensors) via feeds.

Maintain shared situational awareness

Based in a Server-Client architecture, all information immediately available for all users: info layers are automatically refreshed. The COP allows multiple clients to access to the Web GUI (just requiring a standard HTML5 internet browser) for tactical and operational users. Users can add information (observations, alerts, resource needs,...) related to an incident, which can be shared among the rest of users.

Conduct damage and needs assessment

Data acquisition: Data is dynamically collected from the field, which will be both be presented as user friendly information as well as machine readable info. Situation analysis: There are alerts triggered by sensor values, using a configurable rules engine. Information layout: All information is available in two combined ways: Map and List view. The Map view provides information organized by layers, which can be enabled/disabled and manipulated. The Geo-referenced data is dynamically imported to create the layers. The List view, on the other hand, organizes the information in lists, Additionally, a set of data filters is available so information can be shown/hidden from the map at will.

MDA command and Control system (MDA C2)

This solution is an internal solution.

Team

Real Name

Itamar Laist

Roles

Owner

About

Solution owner

Summary

MDA C4I system allows for efficient, real time response to tasks on the field (e.g. people in need for medical assistance), by allocating the site, allocating the resources needed and avialable, tasking the resources and following up the acomplishment. This can be achieved for large number of incidents simultaniousely and for large number of resources to the same task, grouping them if needed. The systems recieves and disseminates information to dedicated apps both used by the general public as well as by the team members and volunteers.

Innovation stage

Stage 6: Wide-scale Adoption

Readiness

TRL 9

Crisis size

Local

Regional

Cross-border

Crisis Cycle Phase

Response

CM-Functions

Activate crisis management bodies

Activate crisis management bodies

The system allows to dispatch emergency response resources to the scene

Manage volunteers

Volunteers are activated through the system and the apps, and thier actions are followed

Conduct incident or emergency response

The main objective of the system is to conduct a response to an emergency situation or to many emergency situations.

Deploy first responders

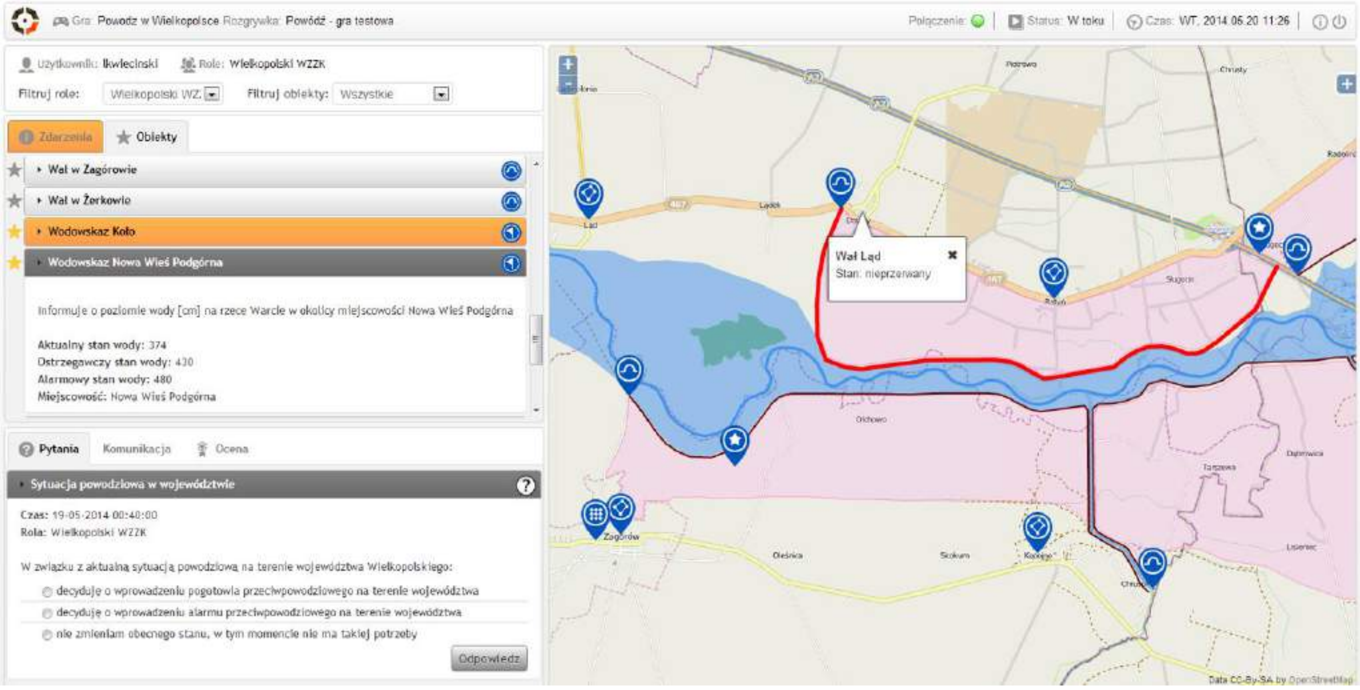
The system is about the deployment and followup to the functions of the first responders on the scene

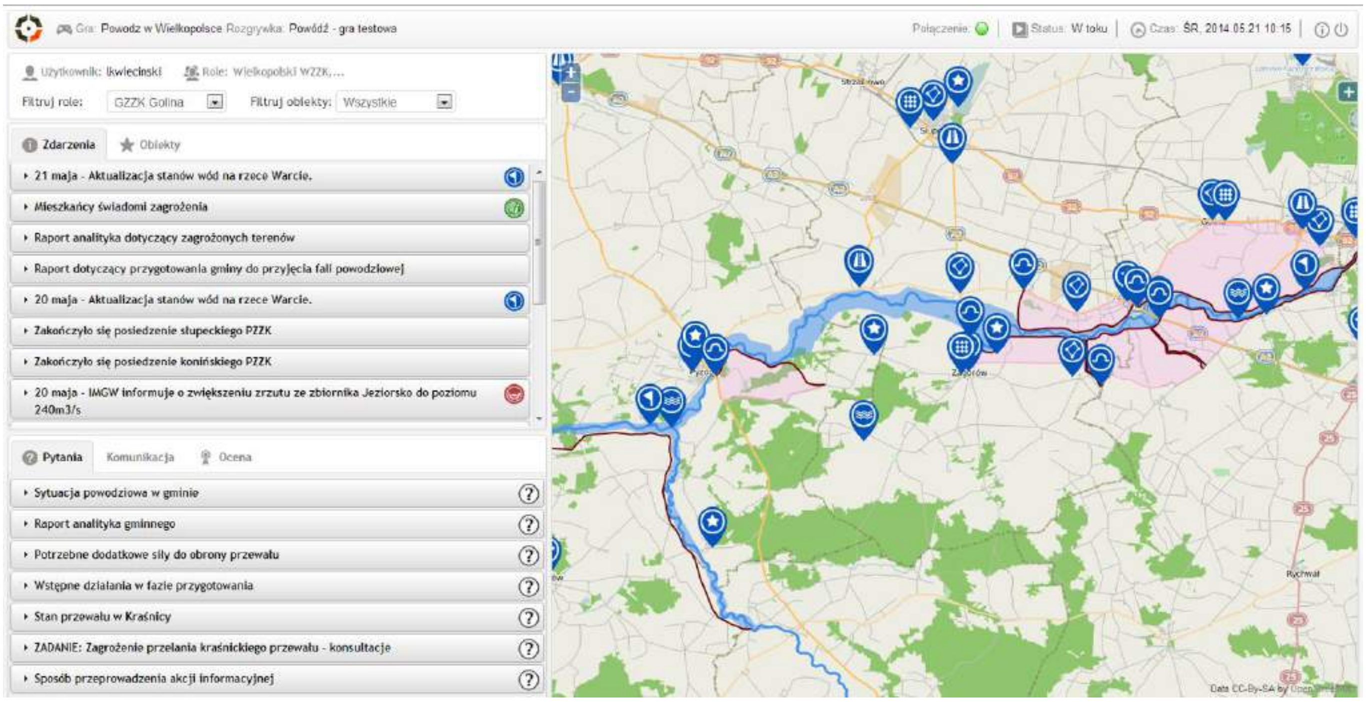
Develop and sustain COP

the system integrating information from numerous sources on real time basis, creats a real time cop, that is shared through the dedicated apps with all the respective managers.

Preparedness for Decision Making in Crisis (PROCeeD)

This solution is an internal solution.




Gra: Powodz w Wielkopolsce Rozgrywka: Powódź - gra testowa
Połączenie:
Status: W toku
Czas: ŚR, 2014.05.21 10:15
Użytkownik: lkwiecinski
Role: Wielkopolski WZZK,...
Filtruj role: GZZK Golina
Filtruj obiekty: Wszystkie
Zdarzenia
Obiekty
21 maja - Aktualizacja stanów wód na rzece Warcie.
Mieszkańcy świadomi zagrożenia
Raport analityka dotyczący zagrożonych terenów
Raport dotyczący przygotowania gminy do przyjęcia fali powodziowej
20 maja - Aktualizacja stanów wód na rzece Warcie.
Zakończyło się posiedzenie słupeckiego PZZK
Zakończyło się posiedzenie konińskiego PZZK
20 maja - IMGW informuje o zwiększeniu zrzutu ze zbiornika Jeziorsko do poziomu 240m3/s
Pytania
Komunikacja
Ocena
Sytuacja powodziowa w gminie
Raport analityka gminnego
Potrzebne dodatkowe siły do obrony przewału
Wstępne działania w fazie przygotowania
Stan przewału w Kraśnicy
ZADANIE: Zagrożenie przelania kraśnickiego przewału - konsultacje
Sposób przeprowadzenia akcji informacyjnej

Team

Real Name

Andrzej Adamczyk

Roles

Owner

About

Solution Owner

Summary

PROCeed is a computer system which prepares its users for proper decision-making in crisis situations. It enables creating and running all kinds of simulation applications and can be used as an interactive decision-making training game, as well as a tool for multi-variant analysis. Implementation of simulation techniques enables accurate modeling of the actual emergency proceedings by providing all the necessary roles, flashpoints, events, physical objects, or the environment. While observing dynamically changing situation, application users can influence on the other users engagement and make various decisions affecting future course of events. Training and analysis carried out in the real world are the best solution, however they consume time and tend to be very expensive, particularly in terms of necessary repetition. Therefore, acting in simulated conditions is the optimal way of training and analysis of crisis situations. Within the costructed environment training and analysis are based on the fulfiling tasks in real-like conditions, using appropriate methods and tools, observing the consequences of one's decisions and trying to assess the undertaken actions. Due to the high degree of repeatability and interactive features of training and analysis within the PROCeed system, it is possible to significantly reduce costs and increase effectiveness of the coaching process. In addition, this type of training and analysis can be provided more often and may consider wider range of possible options.

Innovation stage

Stage 5: Market Growth

Readiness

TRL 9

Crisis size

Regional

Crisis Cycle Phase

Preparedness

Response

CM-Functions

Provide predictive analysis and situational awareness

By analysis and playing decision games you acquire a situational awareness.

Establish CM doctrine and train organisations and people

Decision games help exercise in simulated environment the consequences of one's decisions.

Maintain shared situational awareness

Decision games involving various roles in the crisis situation helps to work out cooperative patterns of proceeding.

Estimate cascading effects

Analysis of scenario of events generated by objects' states changes builds the awareness of cascading effects

Certify personnel, train_ing and education

Certify personnel, training and education

Decision games help exercise in simulated environment the consequences of one's decisions.

PROTECT

This solution is an internal solution.

Team

Real Name

Antonio Chagas

Roles

Owner

About

Solution Owner

Summary

Using the know-how and expertise acquired during the development of the CECIS tool, the PROTECT application is a web-based alert and notification system for emergency (and early warnings) situations concerning civil protection. The main concept behind is to monitor and control Emergencies and to manage a pool of Resources to support the assistance provided during emergencies. PROTECT uses a map oriented user approach powered by the know-how and skills from Alert4All, featuring the monitoring and reporting on the development of each scenario, management of all documents related to the scenario, management and dissemination of messages and notifications, collection and retrieval of lessons learnt.

Innovation stage

Stage 4: Early Adoption/ Distribution

Crisis Cycle Phase

Response

CM-Functions

Establish resource management and mutual aid system

PROTECT supports the decision making steps, allowing to collect, update and process data, and track resources (resource pool management). Furthermore, PROTECT establishes a system for provision of mutual aid between different levels of crisis command and management, allowing also to track flows of requests and responds at the entity level (local, national or international entity).

Maintain shared situational awareness

PROTECT allows to collect information (both manually and automatically) , regarding deployed operational assets and requests. Furthermore, PROTECT also features a Common Operational Picture, disseminating and updating it throughout all applicable entities.

Communicate operational information across chain of command

PROTECT communicates its operational information throughout all entities involved in the crisis at hand, namely resources available, requests, messages and notifications

(Drone) Rapid Mapping (DRM)

This solution is an external solution.

Summary

Drone Rapid Mapping enables rapid mapping of incident/crisis area.

The solution enables very fast generation of orthophoto maps based on imagery acquired by any drone (RPAS) available to rescue or crisis management actors. The resulting maps can be viewed and analysed in the dedicated geoportal or any GIS environment already utilised by crisis management institutions. The additional product is a 3D terrain model, enabling better and more intuitive understanding of the area of interest.

Rapid generation of maps is enabled by cloud computing. A drone operator is expected to conduct a flight over area of interest and acquire imagery (using standard on-board camera) in line with the standard operational procedures. Data is uploaded into cloud and automatically processed. The resulting orthophoto map is available within the dedicated geoportal that can also provide access to other maps (satellite, topographic, etc.). The 3D model can be viewed in any standard program (3D viewer).

The efficiency of rapid mapping requires the Internet access with sufficient throughput. Validated efficiency: mapping of 10ha with 2cm pixel and LTE Internet access requires 26 minutes. This timing covers all activities: mission request (crisis manager's briefing for a drone operator), flight preparation, conduct of the flight, landing, data retrieval and upload, all calculations, preparation of geoportal content.

Generation of the high-quality 3D model requires additional 20+ minutes.

The Drone Rapid Mapping as a whole integrated solution represents TRL 7 maturity level and the 3rd Stage of Innovation (Initial Piloting). While it has not yet been implemented fully operationally, its high maturity results from the approach integrating already operational (TRL 9) building blocks. Imagery is acquired by commercially available drones and transmitted to the Internet using existing GSM or SatCom solutions.

Orthophoto maps calculations are conducted by Hexagon software components using the innovative operational IT cloud infrastructure (EO Cloud). Results are delivered by operational geoportal software.

Currently the solution does not have a dedicated user interface for data upload and configuration of calculation parameters - these actions are conducted by experts supporting use of the system.

Input for orthophoto map:

Series of raw photographs, made perpendicularly (camera angle=90') to the ground with overlap of 60/70 (forward / lateral overlap [%]), suggested flight height depends on the resolution of camera sensor and expected pixel size. For 4k camera and 2cm pixel we recommend 50-75m flight height, for forest and water surface - not less than 150m.

Series of raw photographs, 90', 80/80 overlap, preferable with fish-eye lenses – for high quality 3D model.

Output for orthophoto map:

2D georeferenced map as a GeoTIFF file to download.

2D georeferenced map as WMS link (or displayed on solutions owner portal for public access).

3D files in .obj format to be downloaded

Innovation stage

Stage 4: Early Adoption/ Distribution

Readiness

TRL 7

Crisis size

Local

Regional

Crisis Cycle Phase

Preparedness

Response

Trials

Trial 1

DRIVER+ Trial 1

Trial type

Trial

CM-Functions

CM function 1

Maintain shared situational awareness

Maintain shared situational awareness by providing (for every level of CM, accessible everywhere) a up-to-date mosaic of hi-resolution (1pix=1cm) imagery in less than 30 minutes from drone start.

CM function 2

Manage environmental recovery

Manage environmental recovery by comparing already existing maps and ad hoc generated orthophoto and detailed 3D model of terrain. Plan decontamination and clean-up actions eye-witnessing damage from safe distance of your Command Post.

CM function 3

Monitor the affected area

Monitor area & Provide situational awareness by updating any COPs and geoportals anywhere on the world without any additional hassle.

CM function 4

Provide information to media and the public

Provide information to media, decision makers and public in clear and comprehensible way of displaying online 3D models of affected areas (that can also display the response action)

CM function 5

Conduct flights to collect information

Conduct flights to collect information & assess damage by providing (for every level of CM, accessible everywhere) a up-to-date mosaic of hi-resolution (1pix=1cm) imagery in less than 30 minutes from drone start.

Rumour Debunker

This solution is an internal solution.

Team

Summary

The Rumor Debunker offers a solution for internet news analytics, that prevents from being part of online mis- or disinformation campaigns.

Background:

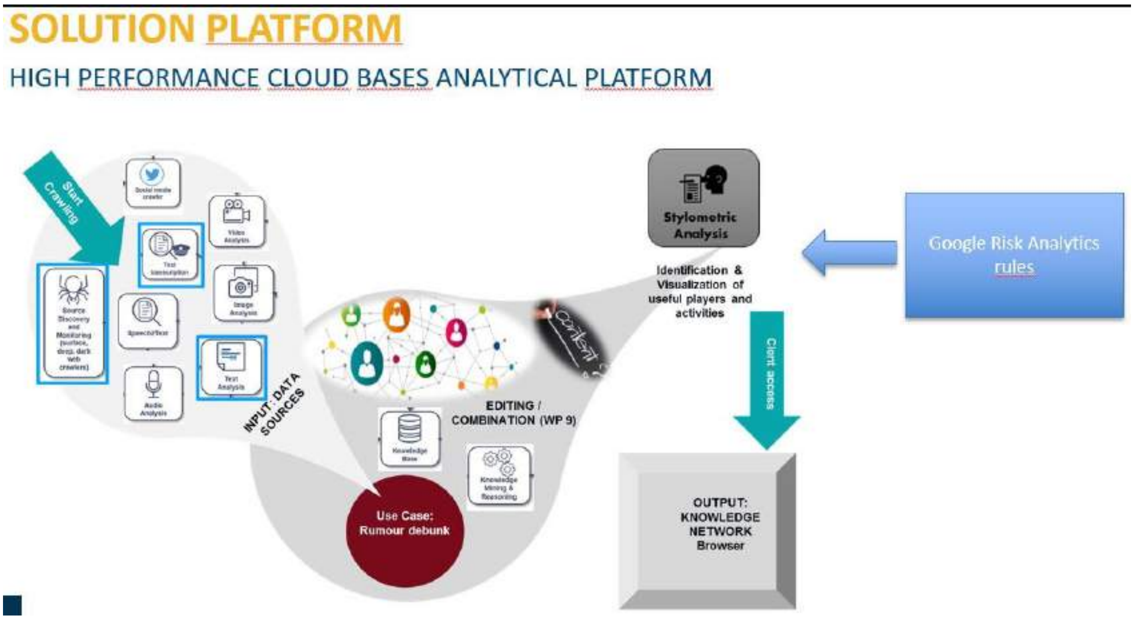
In the last time an increasing amount of misinformation and disinformation becomes visible in all different forms of large scale crisis. For example, in a flooding event in Austria, someone posted on social media, that cholera bacteria are in the water, which was not true. In the Munich gunman event an unprecise situational awareness picture caused a very large amount of bound forces (about 2300) and an unknown number of potential "terrorists". Social media was a facilitator of a very sensitive public reaction. Even with public communication via social media, it was not possible to calm down the situation. These are examples for a new type of crisis communication with new rules. Traditional journalists lost their role as a gateway in crisis communication. With social media in place, crisis communication is always network centric and needs crisis and disaster manager to use communication tools which keep up with the pace of innovation.

Problem:

For network centric communication in crisis and disaster management (CDM) it is very important to gather reliable open source information and to identify sources of misinformation and disinformation quickly in order to be able to react in an efficient manner.

A typical problem is, that the internet provides such a quantity and variety of information that it is impossible, manually to evaluate efficiently, whether information is true or not. It is very easy to overlook whole parts of the internet communication, relevant for a specific crisis, in particular, if these are manmade. Traditional media monitoring methods do not have a proper grade of saturation for strategic communication. Automated methods have proven to be more effective. However, most effective is the combination of automated tools and human intelligence, which is applied in Rumor Debunker.

Solution:



The Rumor Debunker offers a solution against the actual trend of constantly improving online mis- and disinformation campaigns in crisis situations. It includes all phases of crisis management by preparing reliable data sets of media communication for unexpected events.

By pointing out actual trends it allows crisis and disaster managers to discover relevant information from internet news for their operational responsibility.

Within the data set, it offers a solution for:

1. Detect and debunk deception and rumours in social media

   In the Rumour Debunker platform, mis – and disinformation is valued by a compound index, based on source reliability, message reliability and tampering detection (not yet integrated). By building up and providing access to a quality checked news data set for the relevant operational information space, the Rumour Debunker platform shows news, marked with the value of the compound index. As such, the news can be valued according to their reliability.

2. Monitor media coverage

   Based on the data set of classified news messages, users of the Rumour Debunker platform get an insight on how they are present in digital media. They get an impression on whether they are target of a hyper-personalized disinformation campaign. As it is expected to become more important for CDM, in the future user might even be warned if they are such a target.

3. Conduct information analysis and evaluation

   By tracking the sources of specific news, back to their origin, the structure behind the disinformation campaign becomes visible. The first publisher might have a specific interest for publishing the information. The publisher, with the highest multiplicator rate is responsible for spreading the disinformation. For strategic network communication in crisis situations, it is very important to know the motivation behind disinformation campaigns, as well as first source and multiplicator sources, so that they can be addressed appropriately and effectively.

Method:

The Rumor Debunker collects news from open internet sources and rates their reliability (1-10) based on source reliability, message reliability and tamper detection (ongoing research in other projects).

At the actual state, the Rumor Debunker is built from the following three technical components:

- A module for data acquisition from news messages from internet sources
- A mobile app for distributing and accessing quality checked news with transparent quality measures.
- A web-based platform for administration.

This kind of evaluated news may support the decision-making process within crisis and disaster management. The Rumour Debunker will therefore improve the situational awareness by detecting real topics, misinformation and disinformation in web mass media.

Innovation stage

Stage 3: Initial Piloting

Readiness

TRL 6

Crisis size

Local

Regional

Cross-border

Large scale

Crisis Cycle Phase

Preparedness

Innovation stage

Stage 3: Initial Piloting

Readiness

TRL 6

Crisis size

Local

Regional

Cross-border

Large scale

Crisis Cycle Phase

Preparedness

CM Functions

Detect and debunk deception and rumours in social media

In the Rumour Debunker platform, mis – and disinformation is valued by a compound index, based on source reliability, message reliability and tampering detection (not yet integrated). By building up and providing access to a quality checked news data set for the relevant operational information space, the Rumour Debunker platform shows news, marked with the value of the compound index. As such, the news can be valued according to their reliability.

Monitor media coverage

Based on the data set of classified news messages, users of the Rumour Debunker platform get an insight on how they are present in digital media. They get an impression on whether they are target of a hyper-personalized disinformation campaign. As it is expected to become more important for CDM, in the future user might even be warned if they are such a target.

Conduct information analysis and evaluation

By tracking the sources of specific news, back to their origin, the structure behind the disinformation campaign becomes visible. The first publisher might have a specific interest for publishing the information. The publisher, with the highest multiplicator rate is responsible for spreading the disinformation. For strategic network communication in crisis situations, it is very important to know the motivation behind disinformation campaigns, as well as first source and multiplicator sources, so that they can be addressed appropriately and effectively.

Scenario enabled Psychological First Aid (PFA) training

This solution is an internal solution.

Team

Real Name

Pia Tingsted Blum

Roles

Owner

About

Solution owner

Summary

The scenario enabled psychological first aid (PFA) training comprises knowledge on what PFA is, guidelines on how to perform PFA and an experiential training package to build the capacity of individuals, groups and organizations on how to deliver quality PFA.

Incorporating the World Health Organization's Look Listen Link principles for PFA, the tools in this solution are a) Introduction to Psychological First Aid in booklet form, b) Guidelines on caring for staff and volunteers after crises, c) Position paper for policy makers and d), Training package in four parts. Training package part 1: 4 hour basic training in PFA - suitable for all staff and volunteers in any sector Training package part 2: Full day basic training in PFA – experiential training suitable for all staff and volunteers in any sector Training package part 3: Full day training on PFA for children - experiential training for staff and volunteers who work with children and their caregivers and who have completed the full day basic training Training package part 4: Full day training on Group PFA - experiential training for managers and other who have responsibility for the wellbeing of groups of staff and volunteers and who have completed the full day basic training The trainings make use of experiential learning, and trainees practice through role plays, consideration of various situations, from small private crises such as divorce, to large scale emergency events, for example after natural disasters. All trainings use case stories and scenarios. The scenarios are scripted and can be realised by actors or digital solutions such as XVR. The materials are developed by the International Federation of Red Cross Red Crescent Societies' Reference Centre for Psychosocial Support. They address the needs of the Red Cross Red Crescent Movement and other humanitarian and volunteer-based crisis response organizations to build the capacity for providing PFA to people affected by crisis and to care for staff and volunteers after crises. Materials build on the extensive experience and practice of the Red Cross Red Crescent staff and volunteers globally, are research based, and encompass international guidelines and standards.

Innovation stage

Stage 5: Market Growth

Readiness

TRL 9

Crisis Cycle Phase

Response

CM-Functions

Provide MHPSS

The Group PFA training for managers and the guidelines for caring for staff and volunteers enable managers and organizations to create supported working environments that allow staff and volunteers to maintain psychosocial wellbeing in crisis situations. With managers trained in group PFA, leaders and organization are equipped to actively, quickly and safely address the mental health and psychosocial support issues arising in staff and volunteers during a distressing crisis intervention.

Manage spontaneous volunteers

The PFA training of volunteers support spontaneous volunteers to understand and fulfill their roles better and safely. Managers trained in group PFA are able to identify and address the psychosocial reactions experienced by volunteers after a distressing crisis intervention.

Manage organised volunteers

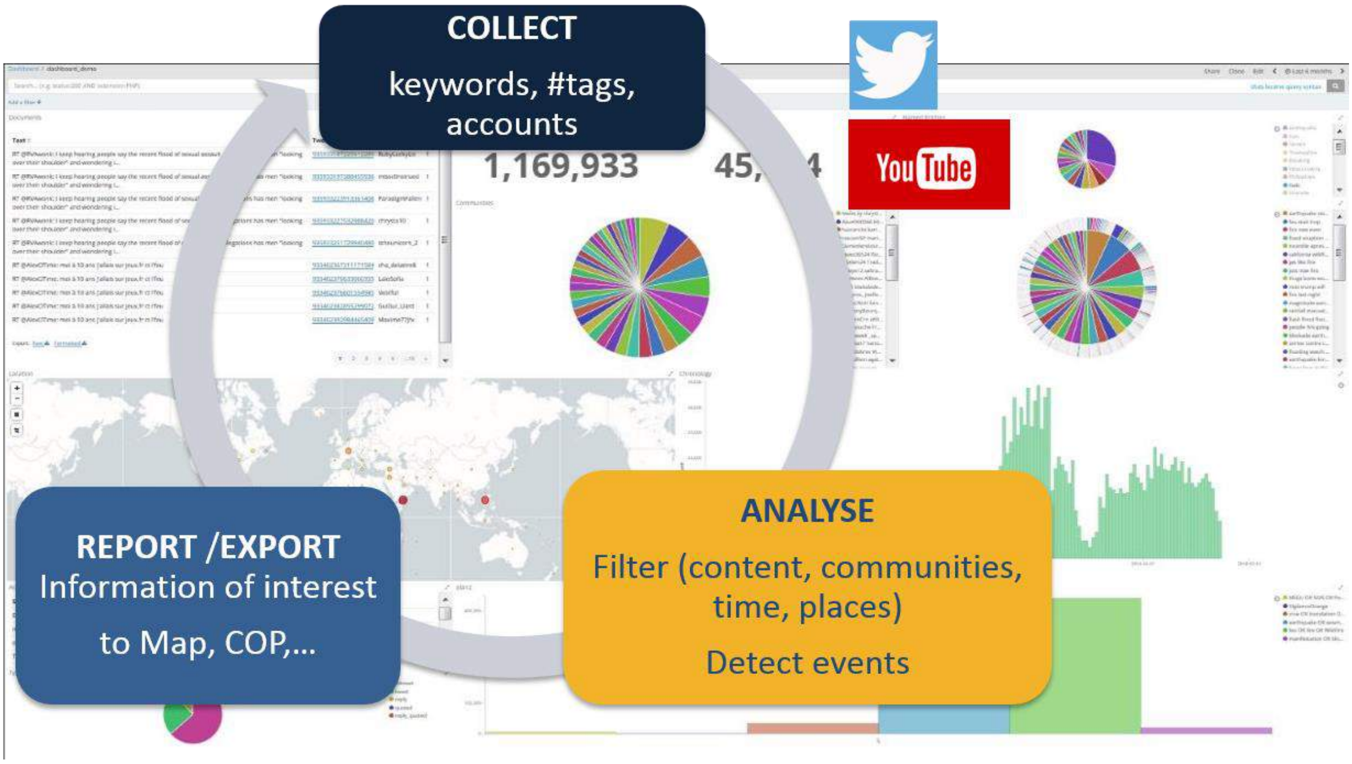
The PFA training of volunteers support organized volunteers to understand and fulfill their roles better and safely. Managers trained in group PFA are able to identify and address the psychosocial reactions experienced by volunteers after a distressing crisis intervention.

Provide off-site health and MHPSS services

The Group PFA training for managers and the guidelines for caring for staff and volunteers enable managers and organizations to create supported working environments that allow staff and volunteers to maintain psychosocial wellbeing in crisis situations. With managers trained in group PFA, leaders and organization are equipped to actively, quickly and safely address the mental health and psychosocial support issues arising in staff and volunteers following a distressing crisis intervention. The solution also support organisations to set up referral mechanisms, enabling them to address more severe mental health issues arising in staff and volunteers following a distressing crisis intervention.

Social Media Analysis Platform (SMAP)

This solution is an internal solution.



Team

Laurent DUBOST

Roles

Owner

About

I am SMAP's solution owner

Summary

Social Media contains precious information which can bring an important contribution to situation assessment. This information can concern the incident(s) itself, the impact, or the needs of the population affected by the crisis. When trying to take this information into account, Crisis Managers face a major challenges which is finding relevant piece of information in a huge volume of information. Social Media Processing automates a user-defined collection process, and proposes content mining tools based on content analysis and network analysis to find the relevant information.

Thanks to its Big Data architecture SMAP is able to store and process the volumes of data generated by Social Media.

Innovation stage

Stage 3: Initial Piloting

Readiness

TRL 5

Crisis Cycle Phase

Response

Recovery

CM-Functions

Detect and debunk deception and rumours in social media

This CM Function is supported by automatically detecting the main themes (potentially rumours) addressed in the social media and the communities of accounts relaying them.

Provide for crowd sourcing

This CM function is supported by assisting the CM managers to mine the social media data and find incident related information. This CM function is also supported by enabling the collection of specific accounts (potentially pre-registered trusted volunteers) and giving the ability to filter the social media data by accounts.

Maintain shared situational awareness

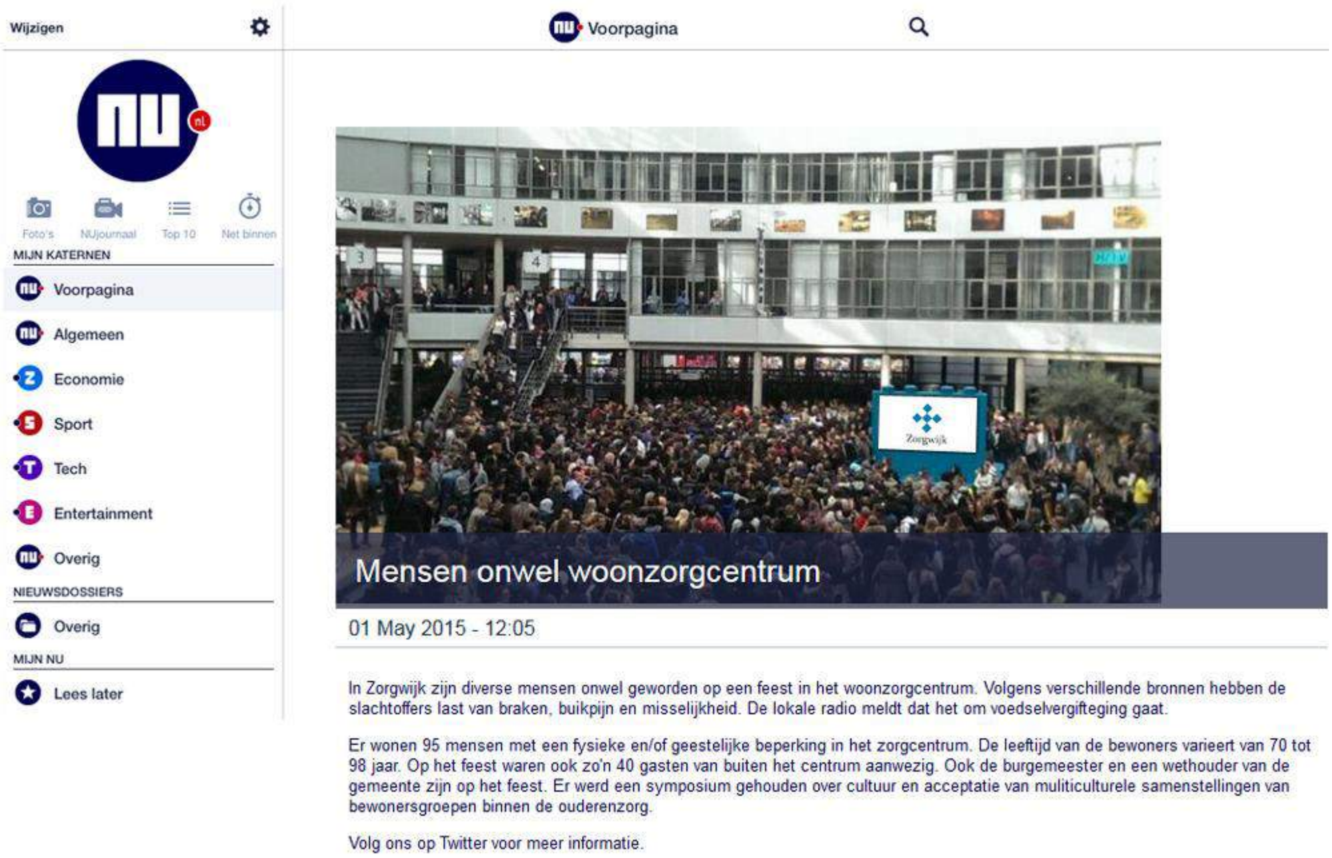
This solution supports Situation awareness by providing a view on incident, needs, damages, and population moral and rumours as reflected in the social media.

Address the needs of vulnerable populations

This CM functionality is supported by the ability to collect information related to needs of the population which is posted in the social media.

Social Media Trainer

This solution is an external solution.



Team

Real Name

Ruud van den Beukel

Roles

Owner

Summary

In the Social Media Trainer (SMT) various Social Media platforms and media outlets can be simulated. This is used to trigger crisis management teams and crisis communication teams and train them on how to respond to this.

Innovation stage

Stage 5: Market Growth

Readiness

TRL 9

Crisis size

Local

Regional

Cross-border

Large scale

Crisis Cycle Phase

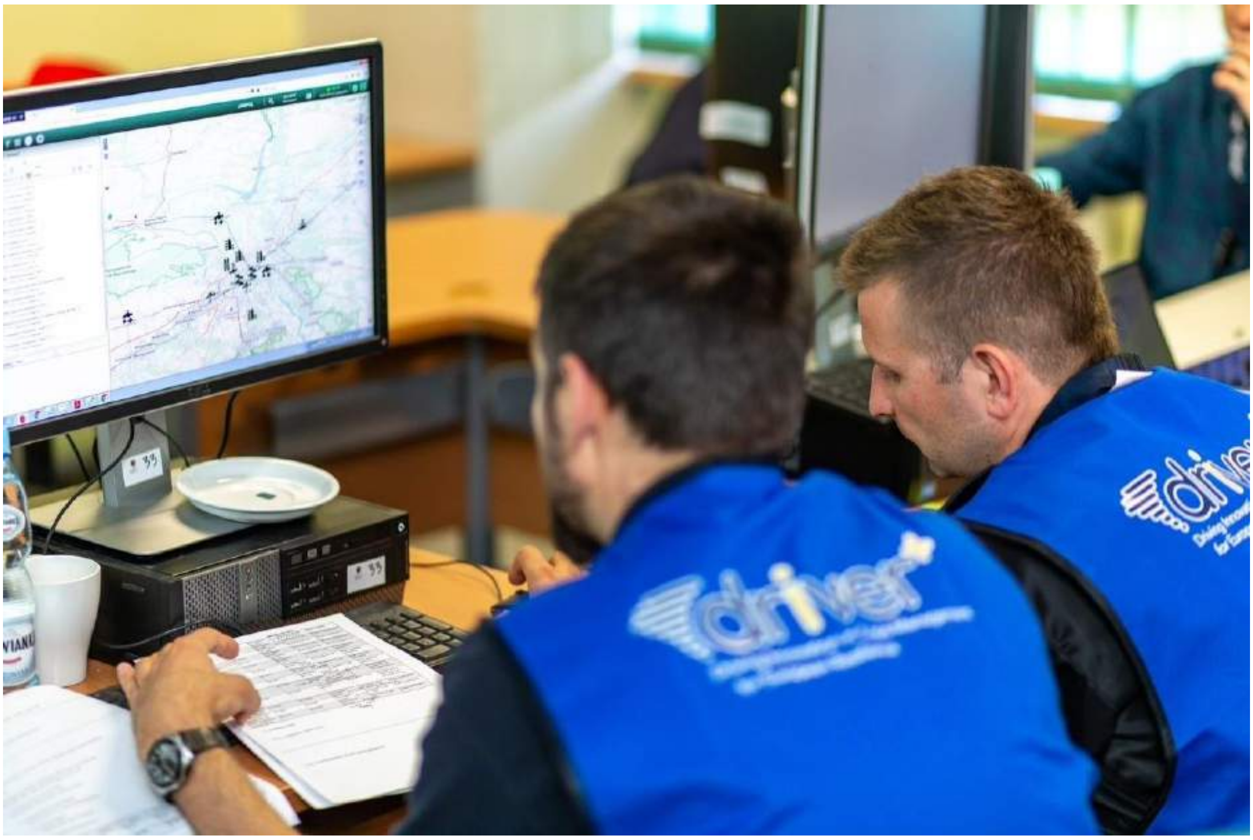
Preparedness

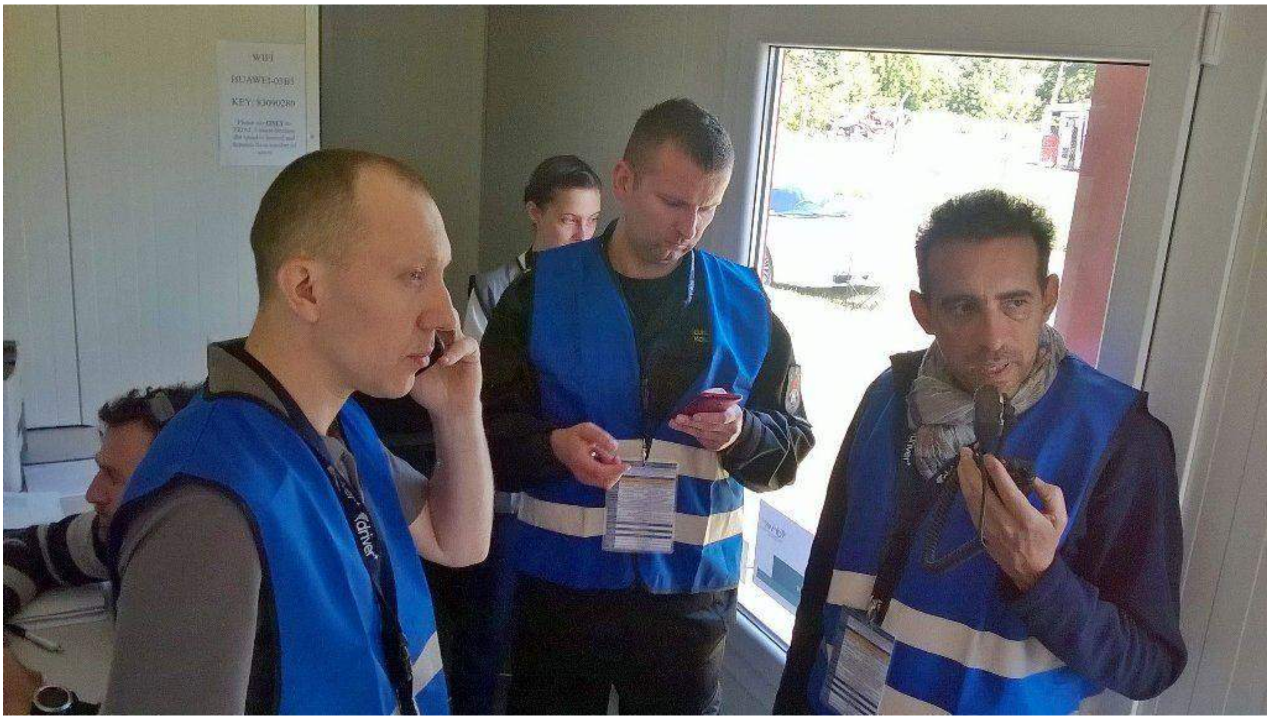
CM-Functions

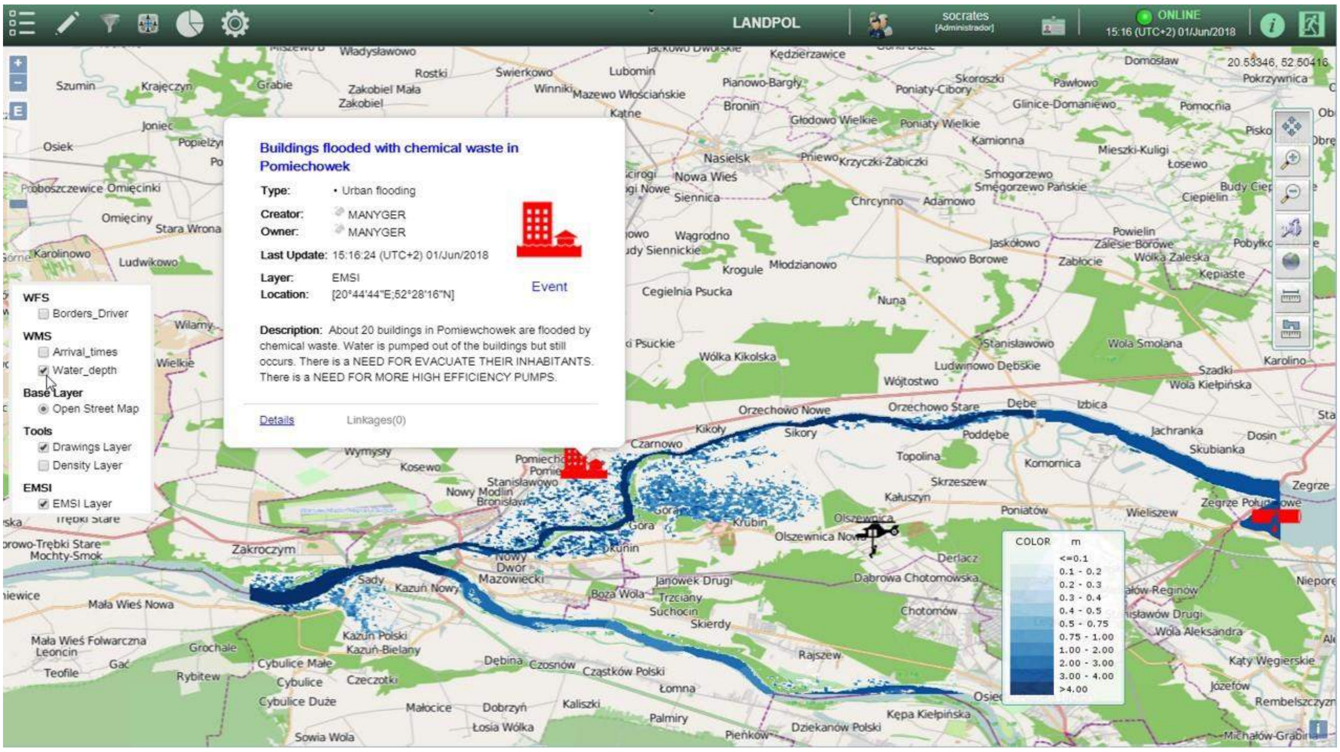
No content has been made publicly available so far.

Socrates OC

This solution is an internal solution.







Team

Real Name

Hector Naranjo Setien

Roles

Owner

About

Solution owner

Real Name

Raúl Valencia Pérez

Roles

Owner

About

owner

Summary

Socrates OC enhances analysis and decision-making capabilities by means of an improved shared situational awareness based on relevant information about the operational situation including crisis events, missions and resources, created by the operator or coming from external sources. The information is displayed on a Common Operational Picture (COP). Socrates OC enables the exchange and sharing of the information (expandable and customizable) among Socrates nodes and with other external systems (using existing standards like ISO's EMSI – Emergency Management Shared Information –) enabling the reporting and tracking of events and inter-organisational tasking (mission assignment) and resource management:

- Events and their associated missions and resources are displayed in a GIS (Geographic Information System).
- Data about them are stored in a DB fully compliant with ISO's EMSI (Emergency Management Shared Information).
- These data are replicated amongst connected Socrates OC instances, which can additionally interact for tasking and resource management.

Input data format:

User input:
EMSI (other system/sensor)
Other Socrates OC node
WMS
WFS

Output data format:
EMSI

Innovation stage

Stage 4: Early Adoption/ Distribution

Readiness

TRL 8

Crisis size

Local

Regional

Cross-border

Large scale

Crisis Cycle Phase

Preparedness

Response

Recovery

Trials

DRIVER+ Trial 1

Trial type

Trial

CM-Functions

CM function 3

Conduct coordinated tasking and resource management

The solution allows registering the existing resources and updating its status and position, assign resources to missions, assign tasks to other nodes or request them for their available resources, and share all this information with all nodes in the CM network.

CM function 1

Maintain shared situational awareness

The solution is able to gather (collect), store (sustain) and share (disseminate) operational information about the crisis situation (regarding crisis events, missions and resources) and exchange it with other nodes in the CM network.

CM function 2

Support C3 decision making

Supports crisis managers and commanders in decision making by sustaining and sharing the COP with relevant information about crisis events, on-going missions, available (and in use) resources, etc.