DOMINO

Demonstration of Communication Booking Service
D6.4

Grant Agreement number	101082230
Call identifier	HORIZON-CL4-2022-SPACE-01
Project Acronym	DOMINO-E
Project title	Earth Observation Multi-mission federation layer
Funding Scheme	Horizon Europe Innovation Action (IA)
Project Starting date	01/11/2022
Project Duration	36 months
Project Coordinator	Airbus Defence and Space
Deliverable related number and full name	D26 Demonstration of Communication Booking Service
Delivery Date	17/10/2025
Issue	**V1.0**
Document produced by	Jakub Rezler – ITTI
Document verified by WP Leader	Jakub Rezler - ITTI
Document authorised by Project Coordinator	Airbus Defence and Space
Dissemination Level	PU

* Please indicate the dissemination level using one of the following codes:
PU = Public,
SEN = Confidential, only for members of the consortium (including the Commission Services).

Abstract

This document presents the demonstration materials that has been recorded to demonstrate the functioning of Communication Booking Service.

List of participants

Participant No	Participant organisation name	Country
1 (Coordinator)	Airbus Defence and Space SAS (ADS-FR)	FR
2	Capgemini Technology Services (CG)	FR
3	ITTI SP ZOO (ITTI)	PL
5	Office National d'Etudes et de Recherches Aérospatiales (ONERA)	FR

Applicable documents

Ref. / Document Title	Ref	Date
Domino-E Description of Work	Version 1	29/09/2022
Domino-E Grant Agreement 101082230	Ares(2022)7032529	11/10/2022
Domino-E Consortium Agreement	VF	14/11/2022
Domino-E Design and interface requirement for communication booking	D4.1, version 1.0	14/09/2023
Definition of implementation and verification plan	D4.1, version 1.0	29/09/2023
TN - Domino_E -SCRMS_configuration	Version 1	19/07/2024
Domino-E Communication Booking Service prototype D4.6	Version 1	31/01/2025

Reference documents

Ref. / Document Title	Ref	Date

Document Change Record

Issue	Date	Page / paragraph affected
V1.0	16/10/2024	Document creation

Table of Contents

1 INTRODUCTION

This document presents the demonstration of the Communication Booking Service, in the frame of the DOMINO-E project for the Work Package 6300, deliverable D6.4.

2 DEFINITION AND ACRONYMS

Definition and acronyms can be found in the Domino-X Glossary included in the [RD01].

1.1 Acronyms

The following table lists some additional acronyms used in this document:

Acronym	Meaning
SCRMS	Satellite Communication and Resource Management Service, synonymous with CBS
CBS	Communication Booking Service, synonymous with SCRMS
DB	Database
KPI	Key Performance Indicator
GSaaS	Ground Station as a Service
FDS	Flight Dynamic Service simulator
E2E	End to End

Table 1 : Acronyms

3 TESTING SYSTEM'S DESCRIPTION

In this chapter, we will describe elements that make SCRMS' testing system- that is, elements allowing for verification of SCRMS.

3.1 Virtual time and Timeshift simulator

An important part of testing SCRMS, regardless of its environment, is usage of here called "virtual time". This refers to practice of setting up a value called timeshifting value. This value is taken by SCRMS and various simulators, making them behave as if the system time changed. This allows for rapid and repeated changes to sense of times of used applications, facilitating tests over large spans of time, including those in the past and the future.

The value of virtual time is managed by Timeshift simulator, and changed by Dispatcher and/or direct API calls to Timeshift simulator.

3.1 Postman

Postman is a program allowing users to easily create, send, and test API calls without writing code. It provides a graphical interface to configure HTTP requests, view responses, manage authentication, and automate testing, making API development and debugging faster and more organized.
It is used in testing of singular features of SCRMS that are not part of a bigger testing scenario.

3.2 Dispatcher

Dispatcher is a special simulator, facilitating automation of tests.
Dispatcher has following tasks:

- It sends API calls and waits for specified amount of time, configured using a csv file.
- It controls changing of the timeshift value.
- It uses Mission Programming Simulator to check and save SCRMS' state.
- It ensures that changes in virtual time will not proceed while SCRMS is in middle of performing a time-sensitive task (e.g. computing contact plan or booking contacts).

It is used in long-form testing scenarios.

3.3 Simulators

Simulators refer to programs that are part of the testing system, that are meant to behave like parts of Domino-X ecosystem. Aside from previously mentioned simulators, there is also:

- Flight Dynamic Service Simulator – responsible of providing orbit ephemerides to SCRMS.
- Mission Programming Service – responsible for querying SCRMS for Global Schedule
- Ground Station as a Service (GSaaS) simulator – responsible for receiving of booking contacts, as well as evaluating and potentially rejecting them, based on services the contacts are assigned to.

Below diagram shows the current state of SCRMS and surrounding simulators:

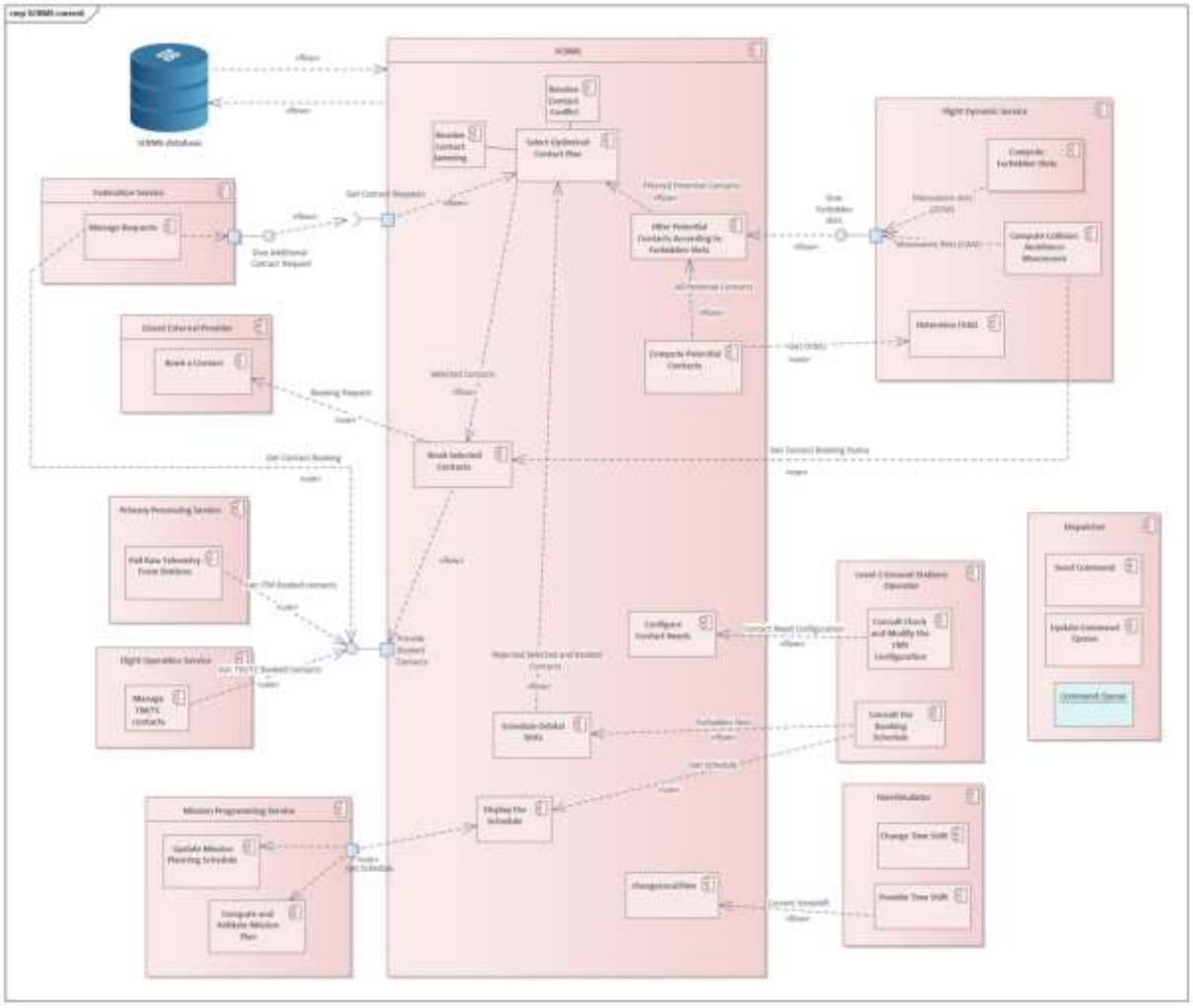

Figure 1 SCRMS and surrounding simulators, with functions higlighted

It must be noted that the place of Federation Service Simulator in stand-alone tests is taken by Dispatcher and Postman, as its function is limited to sending API calls. In E2E test, it's role is taken by Coverage Service.

Primary Processing Service, Flight Operation Service and Level-2 Ground Stations Operator are also simulated using Postman with appropriate commands, due to relative simplicity of their interaction with SCRMS.

4 TESTING ENVIRONMENTS

In this chapter, we will describe the different environments in which tests have been performed. To achieve all the objectives described during the introduction, two environments of tests have been set up:

- Stand-alone environment
- Testbed environment

4.1 Stand-alone environment

Stand-alone environment is used for testing SCRMS itself. Tests ran on a personal mobile workstation.
The environment is composed of SCRMS and simulators set up in docker images.

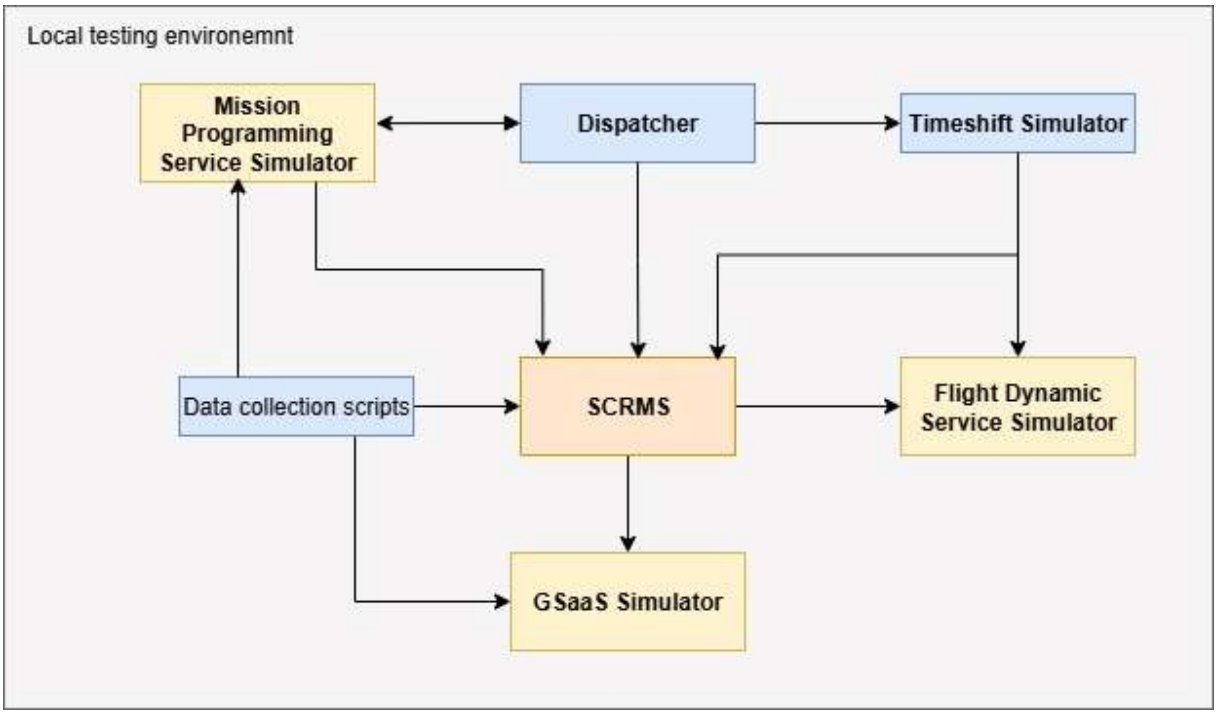

Figure 2 Stand-alone environment

KPI collection is facilitated by automated scripts collecting information thorough scenarios functioning.

4.2 Testbed environment

In order to test interaction between SCRMS and other modules developed in Domino-E, SCRMS is launched on testbed infrastructure.
Diagram below shows the components of E2E testing environment (for sake of clarity, some simulators that do not have a direct contact with SCRMS and its simulators has been omitted).

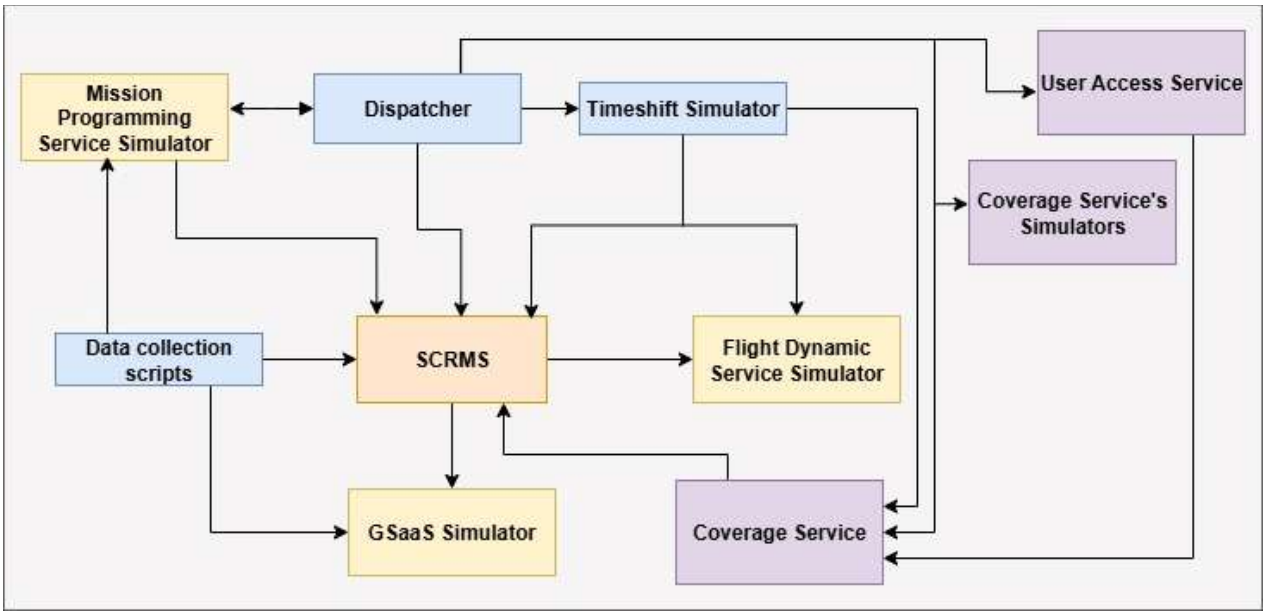

Figure 3 E2E testing environment

Every element present in Stand-alone scenario is present. Main difference is the presence of other modules, mainly Coverage Service, which serves the function of Federation Service.

Similar to stand-alone environment, the environment is composed of docker images of SCRMS and its simulators.

5 FORMS OF DEMONSTRATION

In order to validate CBS', a number of scenarios has been devised:

1. Commentated showcase of SCRMS functions.
2. Long-form stand-alone scenario–
    a. Minimal configuration variant
    b. Full configuration variant
3. Long-from E2E scenario
    a. Minimal configuration variant
    b. Full configuration variant

Each of said demonstrations has been recorded. The recordings are part of this deliverable.

The difference between Minimal and Full configuration variants is the usage of ground stations within SCRMS configuration – Minimal configuration only uses a singular, owned site. It's meant to contrast Domino-X approach to traditional approach.

Below is a short explanation of each of the form of demonstrations.

The specifics of the long-form scenarios has been described in SCRMS Configuration document (TN - Domino_E -SCRMS_configuration).

5.1 Commentated showcase of SCRMS functions

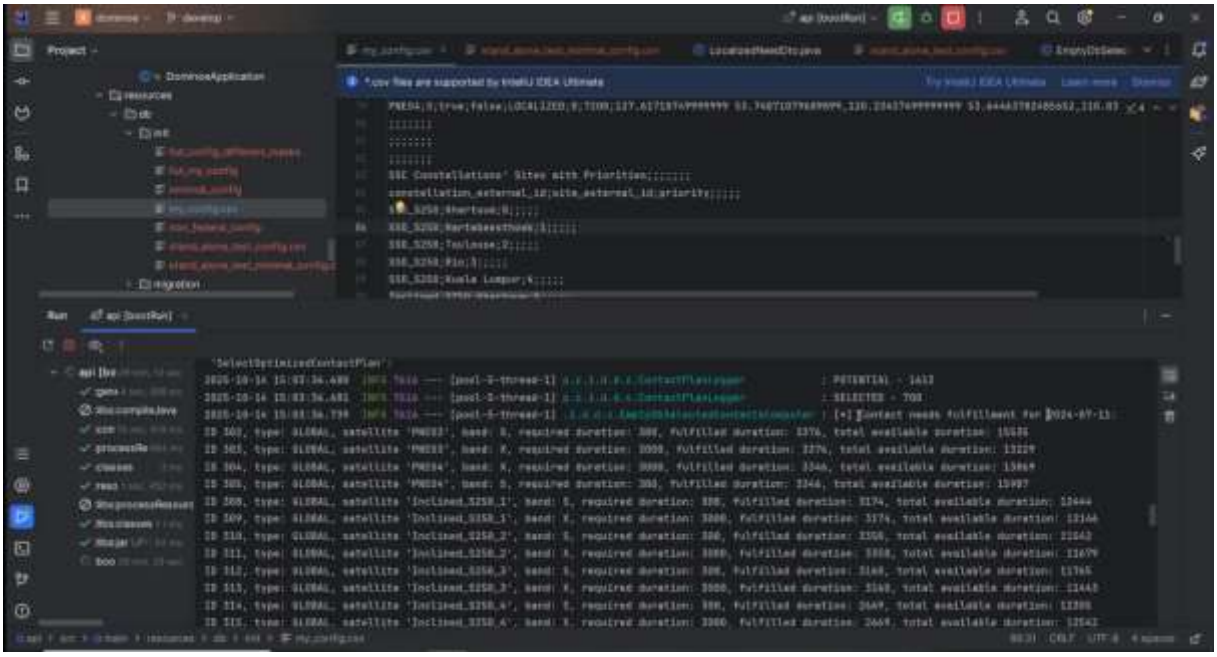

Figure 4 Screenshot from SCRMS demonstration

A demonstration where main functions of SCRMS has been described.

5.2 Long-form stand-alone scenario

A 60-day scenario has been devised to test SCRMS capability to function in scenario approach. This scenario utilizes both Global needs and Localized needs.

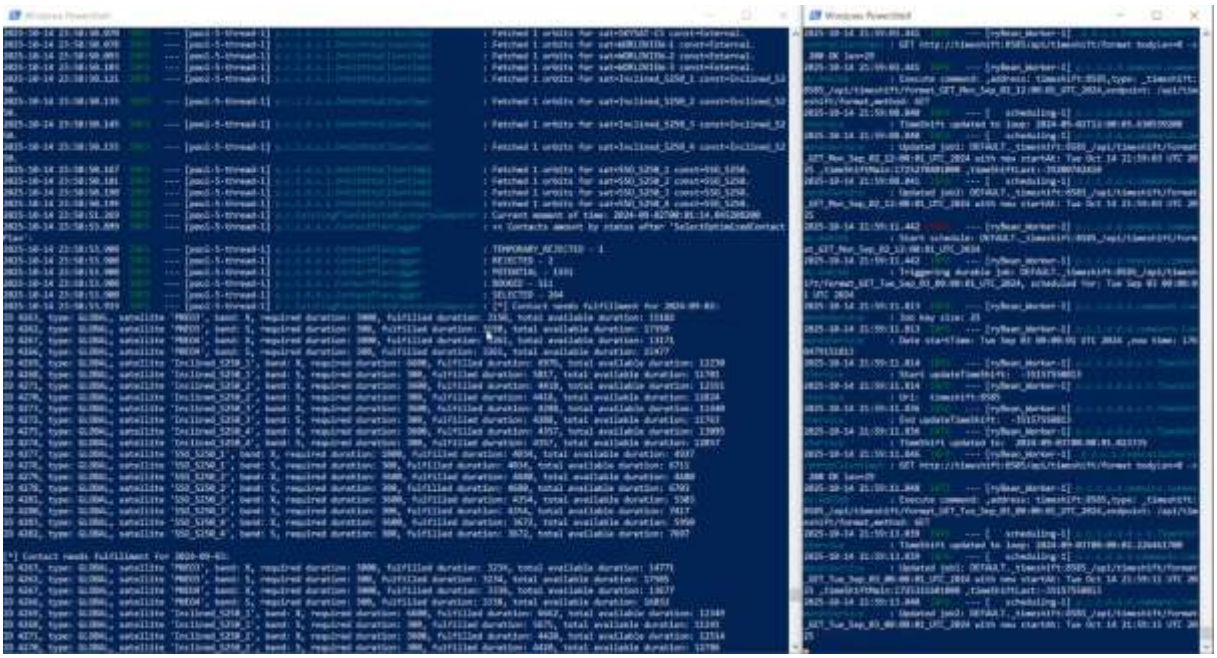

Figure 5 Screenshot from stand-alone scenario (on left SCRMS logs, on right Dispatcher)

The scenario moves through those specific steps:

- An initialized version of SCRMS and its simulators is launched. Computation of potential contacts and selection of contact plan concludes.
- Dispatcher simulator successively changes time to new day, allowing SCRMS to compute new contacts and contact plans.
- Dispatcher successively sends new Contact Needs to SCRMS, forcing it to recompute contact plans.

5.3 Long-form E2E scenario

A 20-day scenario where SCRMS functions alongside other applications of Domino-E.

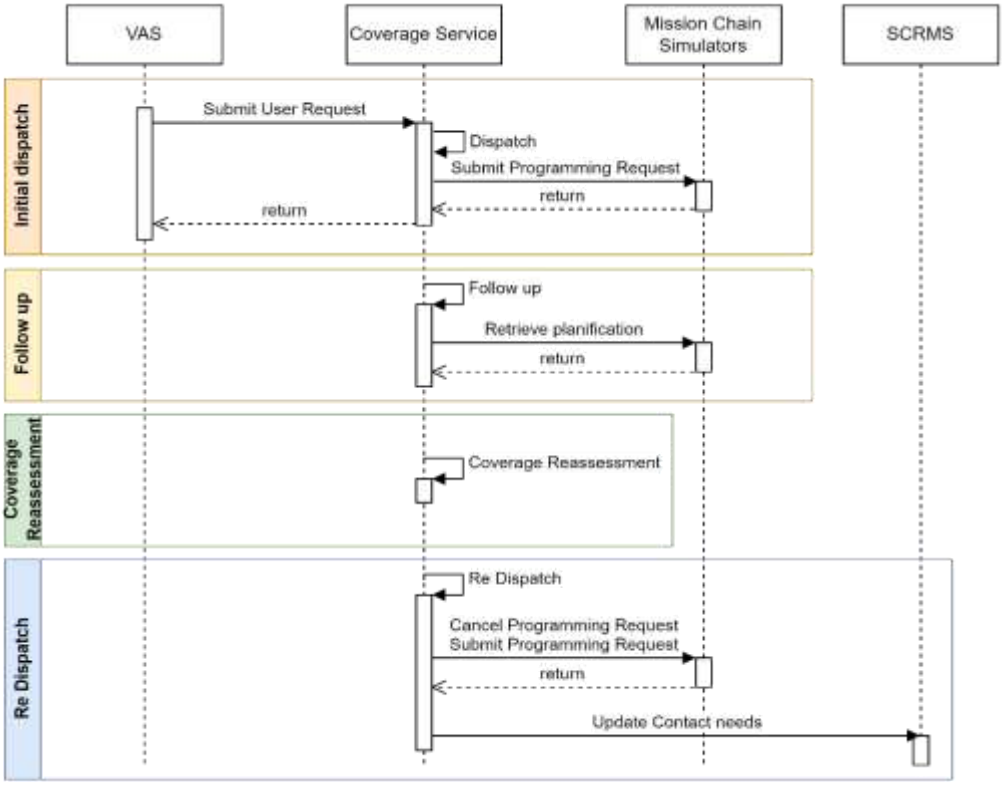

Figure 6 End-to-end interactions between components

SCRMS interacts only with Coverage Service, which pushes updated Contact needs at set times. The scenario begins with SCRMS computing its routine contact needs. Dispatcher then runs, advancing virtual time across several days and triggering multiple contact plan computations, while Coverage Service issues several updates to contact needs.
In the End-to-End scenario, SCRMS relies solely on Global needs.

The scenario moves through those specific steps:

- An initialized version of SCRMS, its simulators, other applications and their simulators is launched. Computation of potential contacts and selection of contact plan concludes.
- Dispatcher simulator successively changes time to new day, allowing SCRMS to compute new contacts and contact plans.
- In meantime, Coverage Service sends new Contact needs, forcing SCRMS to recompute contact plans.

The scenario lasts for 20 days.