IMAGE

Interoperable Management of Aeronautical Generic Executive software

PROPOSAL N° GRD1-2001-40137
PROJECT CONTRACT N° G4RD-CT-2002-00850

D 5.3 : Validation Test Report

Table of Contents

1 Purpose of document

The Validation Plan described how the IMAGE software (framework, services, and modules) will be evaluated by the different partners. Both End users & development teams will be involved in this action, working on different aspect of the validation or on providing help toward. This document describes the different tests that have been performed by the partners of the consortium in order to validate the IMAGE system.

2 Real Time Application

2.1 Test Case 1. Specifications

2.1.1 Test items

- Modeling services
- Assembling and Deploying Simulation Component
- Simulation Entity
- Publish and Subscribe
- Scheduler

2.1.2 Input specifications

This test involves two Image components IMG_Scansim and IMG_Flightsim, which are used as gateways between SCANSIM simulator and Flight Simulator.

2.1.3 Output specifications

The result can be verified by checking trace messages produced by both components, and by the scheduler service itself.

2.1.4 Environmental needs

For better realism, both simulators should generate aircraft positions in the same geographic zone. Thus, the result can be visually checked on the end-simulator user interfaces.

2.2 Test Case 1. Procedure

2.2.1 Purpose

This chapter describes the procedure used to carry out the test case.

2.2.2 Procedure steps

- **Modeling**

We create two new components IMG_Scansim and IMG_FlightSim that both publish and subscribe to the Aircraft data entity.

- **Launching**

2.3 Test Case 1. Report

2.3.1 Description

- Modeling services: OK
- Assembling and Deploying Simulation Component : OK
- Scheduler : OK
- Simulation Entity / Publish & subscribe : OK

2.4 Test Case 2. Specifications.

2.4.1 Test items

- Simulation Recording and Replaying

2.4.2 Input specifications

This case test involves the same components as in Test Case1.

There are two steps of this test case :

1. Recording aircraft information
2. Replaying

2.4.3 Output specifications

<u>Recording phase :</u>

The recording phase shall produce a recording file.

<u>Replaying phase :</u>

During the replaying phase, the recording service shall reproduce the same aircraft positions as during the recording phase.

The accuracy can be verified :

- By checking trace messages produced during both phases.
- By visually checking aircraft positions and states within the 2 simulators during both phases.

2.4.4 Environmental needs

Same as the case Test Case1.

2.5 Test Case 2. Procedure

2.5.1 Purpose

This procedure is conducted in 2 separate steps :

1. We run Flight Simulator alone with the IMG_FlightSIM component and the recorder service running.
2. Then we run SCANSIM alone with the IMG_Scansim component and the replaying service running.

2.5.2 Procedure steps

- Modeling
- Assembling and Deploying Simulation Component

2.6 Test Case 2. Report

2.6.1 Description

- Modeling
    1. In GME, open the model defined in Test Procedure Case1.
    2. Import the IMAGE library.
    3. Create 2 new assemblies:
        a. Record_Test assembly
        b. Replay_Test assembly
    4. In the Record_Test assembly add the Recorder and Recorder_UI components from the IMAGE library and connect them together.
    5. In the Replay_Test assembly add the Replay and Replay_UI components form the IMAGE library and connect them together.

- Assembling and Deploying Simulation Components

Record_Test assembly : OK
Replay_Test assembly : OK

- Simulation Entity / Publish & subscribe: OK

- Scheduler: OK

3 Real Time Application

3.1 Test Case 1. Specifications

3.1.1 Test items

- Modelling services
- Assembling and Deploying Simulation Component
- Simulation Entity
- Publish and Subscribe
- Scheduler

3.2 Test Case 1. Procedure

3.2.1 Purpose

This chapter describes the procedure used to carry out the test case.

Procedure steps are the following:

- Modelling
    - creation of a blank project with IMAGE meta-model
    - creation of two new components that both publish and subscribe to the Aircraft data entity
    - in the simulation entities folder, definition of aircraft entity (position, orientation, etc.)
    - creation of a new assembly
    - definition of a new scheduling diagram
- Launching

3.3 Test Case 1. Report

3.3.1 Description

- Modelling services: OK
- Assembling and Deploying Simulation Component: OK
- Scheduler: OK
- Simulation Entity / Publish & subscribe: OK
- Comparing aircraft positions in SCANSIM and in FlightSim

3.4 Test Case 2. Specifications.

3.4.1 Test items

- Simulation Recording and Replaying

3.5 Test Case 2. Procedure

3.5.1 Purpose

This procedure is conducted in 2 separate steps:

3. Running of Flight Simulator alone with the IMG_FlightSIM component and the recorder service running.
4. Then running of SCANSIM alone with the IMG_Scansim component and the replaying service running.

Only one host machine is used.

3.6 Test Case 2. Report

3.6.1 Description

- modelling
- assembling and deploying simulation components: OK
- simulation entity / publish and subscribe / scheduler: OK

4 Numerical Application

4.1 Test Case Specifications

4.1.1 Test items

- Modelling services
- Assembling and Deploying Simulation Component
- Wrapper
- Gateway

4.1.2 Input specifications

This test involves five IMAGE components (NASTRAN, ESARAD, ESATAN, SENsys and MATLAB components):

- Modelling services :
  These 5 components had been modelled using IMAGE's modelling service.
- Assembling and Deploying Simulation Component :
  The deployment task shall be handled by IMAGE.
- Wrapper
  Four wrappers have been developed to encapsulate the legacy code tools NASTRAN, ESATAN, ESARAD and MATLAB.
- Gateway
  A Gateway has been developed to exchange the data between NASTRAN and ESATAN.

4.1.3 Output specifications

- Modelling service :
  In GME, user shall be able to define the 5 new components, their interfaces and the deployment diagram.

- Assembling and Deploying Simulation Components :
  The 5 components will be deployed and assembled according to the deployment diagram defined in the modelling service.

4.2 Test Case Procedure

4.2.1 Purpose

This chapter describes the procedure used to carry out the test case.

4.2.2 Procedure steps

- Modelling

⇒ In GME (Generic Modelling Environment), a new project, based on the IMAGE paradigm, is created.
⇒ We create five new components (NASTRAN, SENsys, ESARAD, MATLAB and ESATAN)
⇒ The interfaces and new complex types needed for exchanging data and connecting the application components are defined.
⇒ We create a new assembly that involves 1 instance of each component type.
⇒ Each component is deployed

- Launching

The system is launched.

4.3 Test Case Report

4.3.1 Description

- Modelling services: OK
- Deployment: OK but with the CIAO_Daemon launched manually on the remote PCs.
- Assembling Component : OK
- Wrapper and Gateway: OK

The defined methodology is correct.

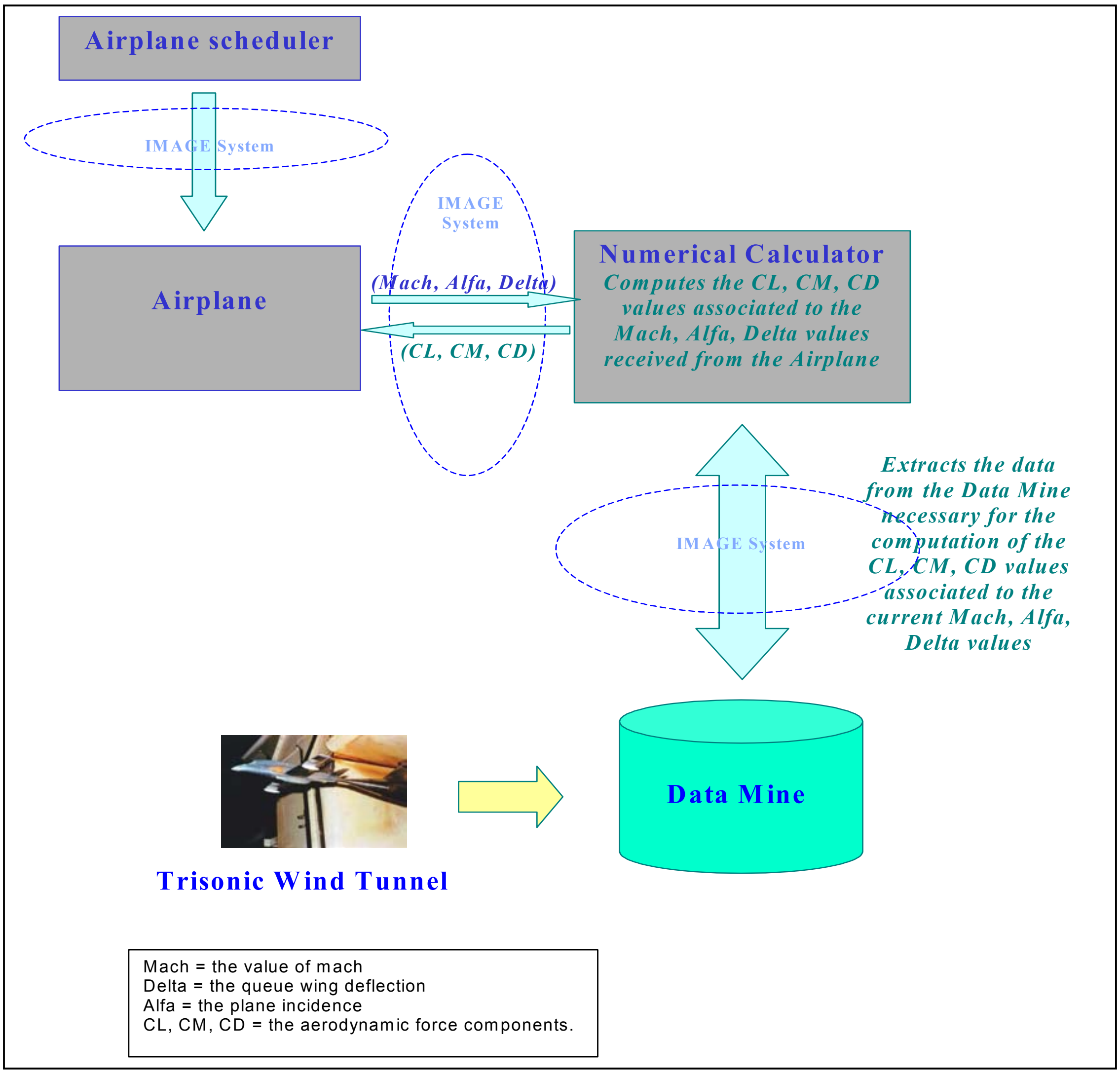
5 Mixed Real Time – Numerical Application

5.1 Test Case 1. Specifications

5.1.1 Test items

- Modelling services
- Assembling and Deploying Simulation Component
- Time Management
- Gateway
- Simulation State
- Simulation Management
- Error Handling and Fault Tolerance

5.1.2 Input specifications



- Modelling services:
  The three IMAGE components shall be modelled using IMAGE 's modelling service.
- Assembling and Deploying Simulation Component:
  The deployment task shall be handled by IMAGE. The three components will be deployed on a standard Windows XP host machine.
- Time Management
  The three components will be pulsed as follows:
  - Airplane Scheduler: 3Hz
  - Airplane: 1Hz
  - Numerical Calculator: 1Hz
- Gateway
  A Gateway will be implemented to ensure that the IMAGE Numerical Calculator can read the polynomials provided by the wind tunnel and its post processing applications
- Simulation State
  Adding of some trace to show that the simulation components go through the right states and correctly pass them to the simulation manager
- Simulation Management
  Adding of trace to show that the simulation management works properly.
- Error Handling and Fault Tolerance
  Creation of, catch and report an error both in IMAGE internal code and in the user code.

5.1.3 Output specifications

- Modeling services:
  In GME, the user shall be able to define the 3 components AirplaneScheduler, Airplane and NumericalCalculator, as well as the facet and receptacles and its methods to allow them to communicate and the deployment diagram.
- Assembling and Deploying Simulation Component:
  The deployment task shall be handled by IMAGE. The three components will be deployed on a standard Windows XP host machine according to the deployment diagram.
- Time Management
  The three components shall be pulsed as follows:
  - Airplane Scheduler: 3Hz
  - Airplane: 1Hz
  - Numerical Calculator: 1Hz

  As a result of the design defined in GME. As a result, we should check by means of trace messages that airplane would always receive the aerodynamic force associated to any new parameter set received by the Airplane Scheduler (if the numerical calculation itself is not higher than one cycle).
- Gateway
  The Numerical Calculator should be able to read the polynomial coefficients without error and the computed aerodynamic force should be correct (the same as the one computed directly by PP3Sv2.0).
- Simulation State

Trace messages should show that the components go through the right states and correctly pass them to the simulation manager

- Simulation Management
  Trace messages should show that the simulation management works properly. The successful execution of the overall simulation demonstrate that the simulation manager correctly works.
- Error Handling and Fault Tolerance
  Trace messages should show that the created errors have been cached and reported correctly to the simulation manager.

5.1.4 Environmental needs

The data produced by INCAS ‘s wind tunnel and its applications are provided on the form of a Microsoft Access Database, which is then post-processed by the PP3Sv2.0 application that has been coupled with a C program to generate the modelling polynomials. The modelling polynomials are provided on the form of coefficients stored in a text file loaded on an ftp site accessible by the IMAGE Numerical Calculator ‘s Gateway.

The IMAGE Numerical Calculator, Airplane Simulator and Airplane Scheduler are running on standard computers (e.g.: DELL INSPIRON 8500) connected to an Ethernet LAN Network.

5.2 Test Case 1. Procedure

5.2.1 Purpose

This chapter describes the procedure used to carry out the test case.

5.2.2 Procedure steps

5.2.2.1 Modelling

In GME, creation of a blank project with the IMAGE meta-model with three components:

- Airplane CCM: defined in the IMAGE middleware in order to represent the simulated aircraft.
- Airplane Scheduler: defined in the IMAGE middleware in order to represent both the pilot and weather application actions. It is responsible for changing the aircraft parameters on a regular time basis.
- Numerical Calculator: defined in the IMAGE middleware in order to represent the numerical calculator.

The method “getAerodynamicVector” computes each aerodynamic force component (CL, CD, CM) by introducing the polynomials coefficients in the formulas

The IMAGE Component Assembly of the current simulation case is composed of three CCM components instances called MyAirplaneScheduler, MyAirplane and MyNumericalCalculator connected as shown here below.

5.2.2.2 Launching

The system launch is done by different steps.

5.3 Test Case 1. Report

5.3.1 Description

- Modelling services

The entire modelling task is done in GME as specified in IMGTest_Procedure01:

  - Creating a new project : OK
  - Creating the 3 components: OK
  - Adding the specified facets and receptacles to each components: OK
  - Defining the assembly diagram: OK
  - Defining the deployment diagram: OK

- Assembling and Deploying Simulation Component : OK

Both component are correctly deployed and assembled on the selected host machine(s).

- Time Management
  The three components generate trace messages according to their pulse frequency.

- Gateway
  The IMAGE Numerical Calculator can read the polynomials provided by the wind tunnel and its post processing applications and we can check that they are the same as the one we can directly acquire using the PP3Sv2.0 application.

- Simulation State
  Trace messages of each component and the simulation manager component, show that the simulation components go through the right states and correctly pass them to the simulation manager

- Simulation Management
  Trace messages show that the simulation manager is correctly started, correctly creates the simulation components and then is correctly stopped at simulation shutdown.

- Error Handling and Fault Tolerance
  Trace messages show that the error created in the internal component and in the user components are correctly cached and reported.