►B

# COMMISSION REGULATION (EC) No 482/2008

**of 30 May 2008**

**establishing a software safety assurance system to be implemented by air navigation service providers and amending Annex II to Regulation (EC) No 2096/2005**

**(Text with EEA relevance)**

(OJ L 141, 31.5.2008, p. 5)

Amended by:

| | | Official Journal | | |
|---|---|---|---|---|
| | | No | page | date |
| ►M1 | Commission Implementing Regulation (EU) No 1035/2011 of 17 October 2011 | L 271 | 23 | 18.10.2011 |

▼B

# COMMISSION REGULATION (EC) No 482/2008

## of 30 May 2008

## establishing a software safety assurance system to be implemented by air navigation service providers and amending Annex II to Regulation (EC) No 2096/2005

**(Text with EEA relevance)**

THE COMMISSION OF THE EUROPEAN COMMUNITIES,

Having regard to the Treaty establishing the European Community,

Having regard to Regulation (EC) No 550/2004 of the European Parliament and of the Council of 10 March 2004 on the provision of air navigation services in the single European sky (the service provision Regulation) ([1]), and in particular Article 4 thereof,

Whereas:

(1) Pursuant to Regulation (EC) No 550/2004, the Commission is required to identify and adopt the relevant provisions of the Eurocontrol Safety Regulatory Requirements (ESARRs), taking into account existing Community legislation. ESARR 6 entitled 'Software in ATM systems' provides a set of safety regulatory requirements for the implementation of a software safety assurance system.

(2) Commission Regulation (EC) No 2096/2005 of 20 December 2005 laying down common requirements for the provision of air navigation services ([2]) states in the last sentence of Recital 12 that 'The relevant provisions of ESARR 1 on safety oversight in ATM, and of ESARR 6 on software in ATM systems, should be identified and adopted by way of separate Community acts.'

(3) Annex II to Regulation (EC) No 2096/2005 requires providers of air traffic services to implement a safety management system as well as safety requirements for risk assessment and mitigation with regard to changes. Within the framework of its safety management system, and as part of its risk assessment and mitigation activities with regard to changes, a provider of air traffic services should define and implement a software safety assurance system to deal specifically with software related aspects.

(4) The prime software safety objective to be met for functional systems that contain software is to ensure that the risks associated with the use of software in the European Air Traffic Management network systems (EATMN software) have been reduced to a tolerable level.

([1]) OJ L 96, 31.3.2004, p. 10.
([2]) OJ L 335, 21.12.2005, p. 13. Regulation as amended by Regulation (EC) No 1315/2007 (OJ L 291, 9.11.2007, p. 16).

▼B

(5) This Regulation should not cover military operations and training as referred to in Article 1(2) of Regulation (EC) No 549/2004 of the European Parliament and of the Council of 10 March 2004 laying down the framework for the creation of the single European sky (the framework Regulation) ([1]).

(6) Annex II to Regulation (EC) No 2096/2005 should therefore be amended accordingly.

(7) The measures provided for in this Regulation are in accordance with the opinion of the Single Sky Committee,

HAS ADOPTED THIS REGULATION:

*Article 1*

**Subject-matter and scope**

1. This Regulation lays down the requirements for the definition and implementation of a software safety assurance system by air traffic service (ATS) providers, entities providing air traffic flow management (ATFM) and air space management (ASM) for general air traffic, and providers of communication, navigation and surveillance (CNS) services.

It identifies and adopts the mandatory provisions of the Eurocontrol Safety Regulatory Requirement — ESARR 6 — entitled 'Software in ATM Systems' issued on 6 November 2003.

2. This Regulation shall apply to the new software and to any changes to the software of the systems for ATS, ASM, ATFM, and CNS.

It shall not apply to the software of airborne constituents and to space-based equipment.

*Article 2*

**Definitions**

For the purposes of this Regulation, the definitions in Article 2 of Regulation (EC) No 549/2004 shall apply.

The following definitions shall also apply:

1. 'software' means computer programmes and corresponding configuration data, including non-developmental software, but excluding electronic items, namely application specific integrated circuits, programmable gate arrays or solid-state logic controllers;

2. 'configuration data' means data that configures a generic software system to a particular instance of its use;

([1]) OJ L 96, 31.3.2004, p. 1.

▼B

3. ‘non-developmental software’ means a software not developed for the current contract;

4. ‘safety assurance’ means all planned and systematic actions necessary to provide adequate confidence that a product, a service, an organisation or a functional system achieves acceptable or tolerable safety;

5. ‘organisation’ means either an ATS provider, a CNS provider or an entity providing ATFM or ASM;

6. ‘functional system’ means a combination of systems, procedures and human resources organised to perform a function within the context of ATM;

7. ‘risk’ means the combination of the overall probability, or frequency of occurrence of a harmful effect induced by a hazard and the severity of that effect;

8. ‘hazard’ means any condition, event, or circumstance which could induce an accident;

9. ‘new software’ means a software that has been ordered or for which binding contracts have been signed after the entry into force of this Regulation;

10. ‘safety objective’ means a qualitative or quantitative statement that defines the maximum frequency or probability at which a hazard can be expected to occur;

11. ‘safety requirement’ means a risk-mitigation means, defined from the risk-mitigation strategy that achieves a particular safety objective, including organisational, operational, procedural, functional, performance, and interoperability requirements or environment characteristics;

12. ‘cutover or hot swapping’ means the approach of replacing European air traffic management network (EATMN) system components or software while the system is operational;

13. ‘software safety requirement’ means a description of what is to be produced by the software given the inputs and constraints, and if met, ensures that EATMN software performs safely and according to operational need;

14. ‘EATMN software’ means software used in the EATMN systems referred to in Article 1;

15. ‘requirements validity’ means the confirmation by examination and provision of objective evidence that the particular requirements for a specific use are as intended;

16. ‘achieved with independence’ means, for software verification process activities, that the verification process activities are performed by a person(s) other than the developer of the item being verified;

▼B

17. ‘software malfunction’ means the inability of a programme to perform a required function correctly;

18. ‘software failure’ means the inability of a programme to perform a required function;

19. ‘COTS’ means a commercial available application sold by vendors through public catalogue listings and not intended to be customised or enhanced;

20. ‘software components’ means a building block that can be fitted or connected together with other reusable blocks of software to combine and create a custom software application;

21. ‘independent software components’ means those software components which are not rendered inoperative by the same failure condition that causes the hazard;

22. ‘software timing performances’ means the time allowed for the software to respond to given inputs or to periodic events, and/or the performance of the software in terms of transactions or messages handled per unit time;

23. ‘software capacity’ means the ability of the software to handle a given amount of data flow;

24. ‘accuracy’ means the required precision of the computed results;

25. ‘software resource usage’ means the amount of resources within the computer system that can be used by the application software;

26. ‘software robustness’ means the behaviour of the software in the event of unexpected inputs, hardware faults and power supply interruptions, either in the computer system itself or in connected devices;

27. ‘overload tolerance’ means the behaviour of the system in the event of, and in particular its tolerance to, inputs occurring at a greater rate than expected during normal operation of the system;

28. ‘correct and complete EATMN software verification’ means all software safety requirements which correctly state what is required of the software component by the risk assessment and mitigation process and their implementation is demonstrated to the level required by the software assurance level;

29. ‘software life cycle data’ means the data that is produced during the software life cycle to plan, direct, explain, define, record, or provide evidence of activities; this data enables the software life cycle processes, system or equipment approval and post-approval modification of the software product;

▼B

30. ‘software life cycle’ means:

(a) an ordered collection of processes determined by an organisation to be sufficient and adequate to produce a software product;

(b) the period of the time that begins with the decision to produce or modify a software product and ends when the product is retired from service;

31. ‘system safety requirement’ means a safety requirement derived for a functional system.

*Article 3*

**General safety requirements**

1. Whenever an organisation is required to implement a risk assessment and mitigation process in accordance with applicable Community or national law, it shall define and implement a software safety assurance system to deal specifically with EATMN software related aspects, including all on-line software operational changes, and in particular cutover or hot swapping.

2. The organisation shall ensure, as a minimum, that its software safety assurance system produces evidence and arguments that demonstrate the following:

(a) the software safety requirements correctly state what is required by the software, in order to meet safety objectives and requirements, as identified by the risk assessment and mitigation process;

(b) traceability is addressed in respect of all software safety requirements;

(c) the software implementation contains no functions which adversely affect safety;

(d) the EATMN software satisfies its requirements with a level of confidence which is consistent with the criticality of the software;

(e) assurances are provided confirming that the general safety requirements set out in points (a) to (d) are satisfied, and the arguments that demonstrate the required assurances are at all times derived from:

(i) a known executable version of the software;

(ii) a known range of configuration data;

(iii) a known set of software products and descriptions, including specifications, that have been used in the production of that version.

3. The organisation shall make available the required assurances, to the national supervisory authority, demonstrating that the requirements provided for in paragraph 2 have been satisfied.

▼B

*Article 4*

**Requirements applying to the software safety assurance system**

The organisation shall ensure, as a minimum, that the software safety assurance system:

1. is documented, specifically as part of the overall risk assessment and mitigation documentation;

2. allocates software assurance levels to all operational EATMN software in compliance with the requirements set out in Annex I;

3. includes assurances of:

   (a) software safety requirements validity in compliance with the requirements set out in Annex II, Part A;

   (b) software verification in compliance with the requirements set out in Annex II, Part B;

   (c) software configuration management in compliance with the requirements set out in Annex II, Part C;

   (d) software safety requirements traceability in compliance with the requirements set out in Annex II, Part D;

4. determines the rigour to which the assurances are established; the rigour must be defined for each software assurance level, and increase as the software increases in criticality; for that purpose:

   (a) the variation in rigour of the assurances per software assurance level must include the following criteria:

      (i) required to be achieved with independence;

      (ii) required to be achieved;

      (iii) not required;

   (b) the assurances corresponding to each software assurance level must give sufficient confidence that the EATMN software can be operated tolerably safely;

5. uses feedback of EATMN software experience to confirm that the software safety assurance system and the assignment of assurance levels are appropriate. For that purpose, the effects from a software malfunction or failure reported according to the relevant requirements on reporting and assessment of safety occurrences shall be assessed in comparison with the effects identified for the system concerned as per the severity classification scheme established in Section 3.2.4 of Annex II to ►**M1** Commission Implementing Regulation (EU) No 1035/2011 [1] ◄.

[1] OJ L 271, 18.10.2011, p. 23.

▼B

*Article 5*

**Requirements applying to changes to software and to specific software**

1. For any changes to the software or for specific types of software such as COTS, non-developmental software or previously used software for which some of the requirements of Article 3(2)(d) or (e) or of Article 4(2), (3), (4) or (5) cannot be applied, the organisation shall ensure that the software safety assurance system provides, through other means chosen and agreed with the national supervisory authority, the same level of confidence as the relevant software assurance level whenever defined.

Those means must give sufficient confidence that the software meets the safety objectives and requirements, as identified by the safety risk assessment and mitigation process.

2. In the assessment of the means referred to in paragraph 1, the national supervisory authority may use a recognised organisation or a notified body.

▼M1

———————

▼B

*Article 7*

**Entry into force**

This Regulation shall enter into force on the 20th day following that of its publication in the *Official Journal of the European Union*.

It shall apply from 1 January 2009 to the new software of EATMN systems referred to in Article 1(2), first subparagraph.

It shall apply from 1 July 2010 to any changes to the software of EATMN systems referred to in Article 1(2), first subparagraph, in operation by that date.

This Regulation shall be binding in its entirety and directly applicable in all Member States.

▼B

*ANNEX I*

**Requirements applying to the software assurance level referred to in Article 4(2)**

1. The software assurance level shall relate the rigour of the software assurances to the criticality of EATMN software by using the severity classification scheme set out in Section 4 of point 3.2.4 of Annex II to ►M1 Implementing Regulation (EU) No 1035/2011 ◄ combined with the likelihood of the occurrence of a certain adverse effect. A minimum of four software assurance levels shall be identified, with software assurance level 1 indicating the most critical level.

2. An allocated software assurance level shall be commensurate with the most severe effect that software malfunctions or failures may cause, as referred to in Section 4 of point 3.2.4 of Annex II to ►M1 Implementing Regulation (EU) No 1035/2011 ◄. This shall, in particular, take into account the risks associated with software malfunctions or failures and the architectural and/or procedural defences identified.

3. EATMN software components that cannot be shown to be independent of one another shall be allocated the software assurance level of the most critical of the dependent components.

▼B

*ANNEX II*

**Part A: Requirements applying to the software safety requirements validity assurance referred to in Article 4(3)(a)**

1. Software safety requirements shall specify the functional behaviour in nominal and downgraded modes, of the EATMN software, timing performances, capacity, accuracy, software resource usage on the target hardware, robustness to abnormal operating conditions and overload tolerance, as appropriate.

2. Software safety requirements shall be complete and correct, and compliant with the system safety requirements.

**Part B: Requirements applying to the software verification assurance referred to in Article 4(3)(b)**

1. The functional behaviour of the EATMN software, timing performances, capacity, accuracy, software resource usage on the target hardware, robustness to abnormal operating conditions and overload tolerance, shall comply with the software requirements.

2. The EATMN software shall be adequately verified by analysis and/or testing and/or equivalent means, as agreed with the national supervisory authority.

3. The verification of the EATMN software shall be correct and complete.

**Part C: Requirements applying to the software configuration management assurances referred to in Article 4(3)(c)**

1. Configuration identification, traceability and status accounting shall exist such that the software life cycle data can be shown to be under configuration control throughout the EATMN software life cycle.

2. Problem reporting, tracking and corrective actions shall exist such that safety related problems associated with the software can be shown to have been mitigated.

3. Retrieval and release procedures shall exist such that the software life cycle data can be regenerated and delivered throughout the EATMN software life cycle.

**Part D: Requirements applying to the software safety requirements traceability assurances referred to in Article 4(3)(d)**

1. Each software safety requirement shall be traced to the same level of design at which its satisfaction is demonstrated.

2. Each software safety requirement, at each level in the design at which its satisfaction is demonstrated, shall be traced to a system safety requirement.