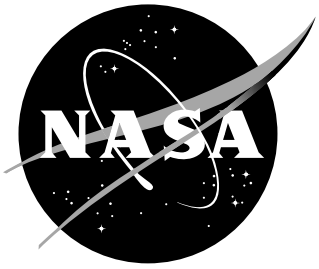
NASA/TM–20220015734



Runtime Assurance of Aeronautical Products: Preliminary Recommendations

Guillaume Brat
Ames Research Center, Moffett Field, California

Ganesh Pai
KBR, Inc.
Ames Research Center, Moffett Field, California

January 2023

NASA STI Program . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA scientific and technical information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI Program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI Program provides access to the NASA Aeronautics and Space Database and its public interface, the NASA Technical Report Server, thus providing one of the largest collection of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

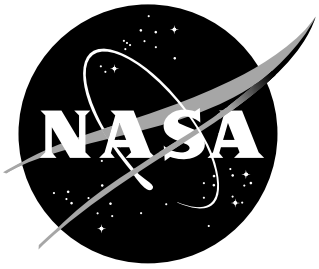
- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers, but having less stringent limitations on manuscript length and extent of graphic presentations.
- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.
- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.
- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include creating custom thesauri, building customized databases, and organizing and publishing research results.

For more information about the NASA STI Program, see the following:

- Access the NASA STI program home page at ***http://www.sti.nasa.gov***
- E-mail your question via the Internet to help@sti.nasa.gov
- Fax your question to the NASA STI Help Desk at 443-757-5803
- Phone the NASA STI Help Desk at 443-757-5802
- Write to:
  NASA STI Help Desk
  NASA Center for AeroSpace Information
  7115 Standard Drive
  Hanover, MD 21076–1320

NASA/TM–20220015734



Runtime Assurance of Aeronautical Products: Preliminary Recommendations

Guillaume Brat
Ames Research Center, Moffett Field, California

Ganesh Pai
KBR, Inc.
Ames Research Center, Moffett Field, California

National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

January 2023

Acknowledgments

This work has been funded by the System-wide Safety (SWS) Project of the Airspace Operations and Safety Program (AOSP) within the Aeronautics Research Mission Directorate (ARMD). We thank Misty Davies who supported this work as SWS PM, and acknowledge the efforts of Adrian Agogino, Alwyn Goodloe, Anastasia Mavridou, Rory Lipkis, Ivan Perez, and Johann Schumann, who helped to review and improve the recommendations documented in this report. Any errors in this report are those of the authors.

Available from:

NASA Center for AeroSpace Information
7115 Standard Drive
Hanover, MD 21076-1320
443-757-5802

Abstract

Runtime assurance (RTA) affords an operational layer of protection against safety hazards to aeronautical products that may include less trusted or untrusted functions. However, any RTA scheme must itself be trusted before it can be deployed into use: i.e., it must be fit for its intended purpose, and it must not itself introduce safety hazards. This report contains preliminary recommendations on the application and integration of RTA into aeronautical products intended for use in civil aviation. The main purpose of these recommendations is to inform regulatory guidance and consensus standards that may be used to meet the safety intent of the applicable regulations.

Contents

List of Figures

List of Tables

1 Introduction

Purpose and Objectives This report aims to inform the aviation community, in particular the US civil aviation regulator—the Federal Aviation Administration (FAA)—and the flight safety foundation, on Runtime Assurance (RTA) and its use in aeronautical products to aid the development of the appropriate regulations and aviation industry-consensus guidance. To that end, this report synthesizes preliminary recommendations on RTA.

Report Outline and Scope Section 2 introduces the terms used, and gives a concise background on RTA, including its scope and a functional organization. This section also briefly summarizes where functionality similar to or the same as RTA has been previously used in aviation.

Section 3 gives the broad characteristics that an RTA scheme should exhibit, followed by the recommendations organized around the development process for RTA, its function, architecture and integration, implementation, usage, and the non-functional attributes it should exhibit. Largely, the recommendations in this report cover the monitoring function in RTA. Recommendations are also made about the assurance of inputs to RTA, the alternative functions invoked, and the decision logic/switching function.

Section 4 concludes with an outline for further maturing the recommendations outlined in Section 3.

2 Background

2.1 Terminology

Conventionally, the literature uses the terms Runtime Monitoring (RTM), Runtime Verification (RV), and RTA interchangeably. In this report, the terms are instead used as follows:

- Runtime Monitoring (RTM) refers to observation of an executing system of interest, its functions, or its environment.
- Runtime Verification (RV) refers to a specialization of RTM, where the response of the monitor is the result of an *online* verification procedure applied to monitor inputs. In general, this report considers RV and RTM to be synonymous.
- Runtime Assurance (RTA) is the combination of RTM and one or more functions triggered by RTM, such as recovery, failover, warning alarm, or shutdown.

Additionally, this report uses the term System Under Observation (SUO) to refer to the system of interest for RTA, and *integrated system* to refer to an SUO that includes RTA mechanisms.

2.2 Overview of Runtime Assurance

2.2.1 Schematic

Figure 1 shows a simplified schematic where a complex function—which may be implemented using emerging untrusted technologies such as Machine Learning (ML)—is *wrapped* by high assurance functions that, together, facilitate RTA. The runtime (safety) monitor can observe (one or all of) the inputs, outputs, and computation of the less-trusted or untrusted complex function. Upon detecting conditions that can violate safety, e.g., invalid inputs, deviant outputs, or errant execution traces, the monitors triggers an intervention: that is, to disconnect the complex function and to switch to a high assurance alternative function to maintain safe system state.

Figure 1 can be seen as (the schematic of) an *architectural pattern* for RTA that abstracts more complex configurations, e.g., with multiple monitors, multiple alternative functions, or hierarchies of monitors and alternative functions. In this schematic, note that the monitor receives both trusted and untrusted inputs. Depending on what is being monitored, the application context, its safety criticality, and the level of assurance required, the concrete architecture may have the monitor receive only trusted inputs. For example, the architecture in [1] is one where the monitor only receives trusted inputs.

2.2.2 Functional Scope

This report is focused on *safety-critical* RTA, rather than *mission-critical* RTA whose focus is the assurance of functional performance of the integrated system [2]. Thus, for the purposes of this report, the emphasis of the RTA function

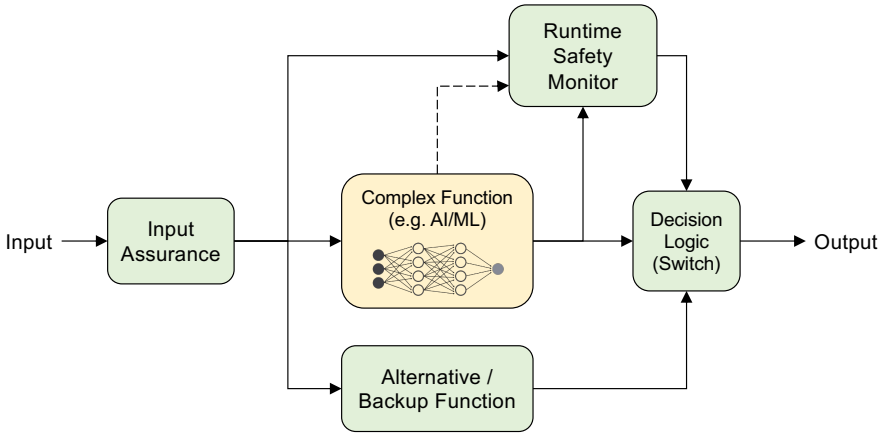


Figure 1. Runtime Assurance architecture pattern

is the provision of safety, so that there is confidence in operation that a specific function, capability, or service provided by the SUO can be safely delivered by the integrated system.

The RTA function can be decomposed into the following (implicitly runtime) functions: input assurance, monitoring, switching decision logic, and alternative or backup function.

Input Assurance The main function of Input Assurance is to ensure that not only do the Monitoring and Backup functions receive trusted inputs, but also that they receive the *right* inputs. That is, that the inputs are correctly routed to the intended functions. The trusted inputs could also (but need not) be provided to the primary Complex Function.

Monitoring Functionally, the (runtime) Monitoring function is meant to detect (safety-related) deviations/violations by observing (specifications or properties of) an SUO. These can be:

- emergent interactions at the system boundary, for example, as in the *safety envelope* concept for RV [3];
- violations of function outputs or guarantees, for instance, as in the *safety shield* concept for RTA [4];
- invalid or incorrect function inputs or assumptions of environmental conditions;
- computational deviations from required internal states, state changes, and guards in state transitions, or undesired state transitions.

Switching Decision Logic The main function of the Switching Decision Logic is to effect a risk mitigation intervention triggered by the monitor; that is, to disconnect the complex function and engage the alternative/backup function instead. Depending on the concept of operations, the usage scenarios, and kind of complex function involved, the switching/decision logic may involve a simplistic *source selection* or more sophisticated protocols.

Alternative/Backup Function Functionally, RTA includes (one or more) *Alternative* or *Backup* functions that serve to replace or failover from the primary complex function. The backup function(s) need not be a completely functional equivalent of the primary and may rather deliver a reduced level of service or capability. In either case, safety is the main focus of the alternative function(s).

2.3 Use in Aviation

The use of RTA in aviation is not new, and it has been successfully deployed and used in a variety of platforms, and usage scenarios. Some noteworthy examples follow.

The Automatic Ground Collision Avoidance System (Auto-GCAS) is an award-winning example of safety-critical RTA in service for avoiding Controlled Flight into Terrain (CFIT) accidents [5]. Deployed into military aircraft platforms, this system includes monitoring capabilities that detect and alert the pilot of imminent ground impact, taking control when there is no pilot response to alerts, to correct the aircraft trajectory to a safe (i.e., non-collision) trajectory.

The Receiver Autonomous Integrity Monitoring (RAIM) [6] framework provides a set of algorithms that use redundancy in Global Positioning System (GPS) pseudorange[1] measurements to establish their measurement consistency. RAIM thus provides a means to monitor the *integrity* of GPS signals in air navigation applications. In turn, that facilitates the detection, possible isolation, and subsequent alerting of faulty signals. In effect, RAIM implements a form of RTA.

Engine Health Monitoring (EHM) [7] represents a suite of monitoring capabilities deployed to observe a variety of subsystems of modern turbofan aircraft engines. The overall goal of EHM is to facilitate early detection of potential engine faults through the monitoring of engine parameters and conditions, such as airspeed, fuel flow, vibrations, and fan revolutions per minute (RPM). EHM is a specialized application of the more general paradigm of Integrated Vehicle Health Management (IVHM), which uses a suite of diagnostic and prognostic techniques for detecting faults and degradation conditions, alerting, and recovery, in a manner similar to RTA. IVHM is used for health monitoring not just at run time, but across the (air) vehicle lifecycle. A specialized application of EHM/IVHM across the lifecycle of jet aircraft engines is Engine Condition Trend Monitoring (ECTM).

Analogous to IVHM is Fault Detection, Isolation, and Recovery (FDIR), a framework for system monitoring to detect faults, accurately isolate the fault to a failed component as rapidly as possible, to reduce system unavailability, and effecting repair. FDIR can be considered as a hardware centric variant of RTA, or analogously, RTA can be considered as a software variation of FDIR [2].

3 Recommendations

The recommended characteristics of trusted RTA are first given (Section 3.1), followed by recommended requirements for the development process, the provided functions, architecture and integration, implementation, usage, and non-functional attributes (Sections 3.2 – 3.5).

3.1 General Characteristics

Trusted RTA is:

Simple the RTA scheme is simpler to specify and validate than the SUO, and is of lower complexity[2] than the SUO.

Benign the RTA scheme does what is intended functionally, does not do what is not intended, and does not harmfully interfere with the SUO.

Realizable a valid RTA specification exists that can be correctly implemented.

Verifiable the implementation of the RTA specification can be verified to be correct with high assurance.

Dependable the RTA scheme exhibits one or more (or all) of the following expected non-functional attributes.

Safety RTA does not contribute to system hazards. This is related to the characteristic of being benign.

Safety Integrity RTA measurably reduces risk.

Reliability RTA continues to delivers correct service.

Availability RTA is ready for service on demand.

Resilience RTA recovers from outages to an acceptable level of service.

Robustness RTA is reliable under well-characterized and bounded perturbations.

Stability RTA attains and maintains a desired steady state within a defined time period.

Security Integrity RTA does not contain improper alterations.

Security RTA exhibits security integrity, availability, and an absence of unauthorized information disclosure.

[1] An approximation of the distance between a GPS satellite and a GPS receiver.

[2] The *Cynefin* framework [8] gives a useful characterization of complexity: a *simple* or *clear* domain is characterized by stability, linearity, and well-understood cause and effect relations. Meanwhile, a complex domain is more unstable, with nonlinearities, unknown or less-understood cause and effect relations, and emergent behavior.

The characteristics above suggest that an RTA scheme that is fit for its intended purpose is one that has a valid, and feasibly implementable specification of its intended function. Additionally, that implementation is verifiably correct and harmless when integrated with the SUO, dependably delivering its expected service (under all foreseeable operating conditions). The non-functional attributes that apply are to be determined based on the application context, and the assurance needs that emerge from the overall concept of operations.

In addition to the preceding characteristics, the recommendations that follow stem from the following principles: (i) RTA must itself by highly trusted; (ii) assurance of RTA must be commensurate to the level of safety risk (of the worst-case effects that RTA is intended to mitigate); (iii) assurance of RTA cannot be a lower level than that of the SUO; and (iv) the use of RTA does not replace or eliminate SUO assurance obligations.

3.2 Development Process

Table 1 lists the recommended requirements on a development process applied for achieving RTA.

Table 1. Recommendations on the development process for RTA

ID	Recommended Requirement
SWS-RV-1	RTM specifications shall derive from system level requirements and assumptions that have been validated by domain experts. **Rationale**: See [9, 10].
SWS-RV-1.SA1	RTM specifications shall include requirements and assumptions deriving from safety analyses. **Rationale**: Accounts for inclusion of results of safety analysis in the system-level requirements from which RTM specifications are derived.
SWS-RV-1.SA1.1	RTM contributions to hazards of the SUO, including RTM failure conditions, shall be identified and mitigated. **Rationale**: Refines SWS-RV-1.SA1, and accounts for the safety impact of introducing RTM to observe the SUO.
SWS-RV-1.SA1.2	Identified RTM contributions to system hazards shall be considered in the system safety analysis. **Rationale**: Closes the loop of system safety analysis to include safety analysis on RTM for identifying architectural mitigations for RTM contributions to system hazards.
SWS-RV-1.SA1.2.1	Dependencies between RTM and the SUO shall be identified and managed. **Rationale**: Refines [SWS-RV-1.SA1.2]. Associated to [SWS-RV-5.SA4] and [SWS-RV-6-SA6]. Accounts for potential failure paths or hazardous interactions between RTM and the SUO.
SWS-RV-2	Safety properties shall be formulated precisely, and if possible, should be formalized in a (possibly probabilistic) logic. **Rationale**: Refinement of [SWS-RV-A2] to support high assurance by using formal methods. Also see [9–13].
SWS-RV-A2	RTM shall be assured to a level commensurate with the risk posed (by the worst-case consequence or failure condition being mitigated by RTM) **Rationale**: Reflects the confidence needed to use RTM as part of an RTA-based fallback solution.

Table 1. Recommendations on the development process for RTA (Continued).

ID	Recommended Requirement
	Note. This recommendation is analogous to requirement F3269-RTAS-002 in [1], i.e., *the designer shall develop the RTA components to the necessary level of assurance as determined by the System Safety Analysis*.
SWS-RV-3	The functions and their associated variables being monitored shall be observable and shall be specified in RTM requirements. **Rationale**: See [9, 10].
SWS-RV-4	RTM development shall support bidirectional traceability from the requirements and system level analysis to the actual RTM code. **Rationale**: Supports confidence that RTM requirements are implemented, and that RTM implementation only realizes RTM requirements. Also see [9, 10].
SWS-RV-7	The correctness of the RTM specification shall be assured (to a level commensurate with the level of safety-criticality of the RTM). **Rationale**: Supports confirmation of validity of RTM functionality. Also see [9, 10].
SWS-RV-7.1	Assurance arguments with evidence should be used for assurance that the executable monitors correctly implement their specification. **Rationale**: Refinement of [SWS-RV-7] to support assurance by using structured rationale that clarifies why RTM implementation is correct given verification evidence.
SWS-RV-7.2	If possible, the assurance arguments should be included in what is monitored for safe operation at runtime. **Rationale**: Refinement of [SWS-RV-7] to support dynamic assurance by including monitor responses into assessment of safety performance of the SUO.
SWS-RV-10	The failover / fallback / recovery function shall be verified to be correct and safe from any viable system state once a specification violation is detected (i.e., upon being invoked by RTM). **Rationale**: Supports confirmation of validity of failover functionality. Also see [9, 10].
SWS-RTM-DP1	The operational domain assumed for developing RTM shall be verified to be consistent with the actual operational domain of RTM. **Rationale**: Assurance of consistency between the operational constraints assumed for the RTM during design and during its deployment.
SWS-FO-DP1	The operational domain assumed for developing the failover function shall be verified to be consistent with the actual operational domain for failover. **Rationale**: Analogous to [SWS-RTM-DP1]. Assurance of consistency between the operational constraints assumed for the failover function during its design and its deployment.

3.3 RTA Functions

3.3.1 Monitoring

Table 2 gives the preliminary recommended requirements on the Monitoring function in RTA.

Table 2. Recommendations on the Monitoring Function of RTA

ID	Recommended Requirement
SWS-RV-3.SA3	RTM shall function under all foreseeable operating conditions of the SUO. **Rationale**: Provides assurance that monitoring occurs under all known nominal and off-nominal operating conditions under which the SUO will be used.
SWS-RV-5	RTM shall not be impacted, rendered inoperable, or unavailable by the conditions that also impact (lead to failure conditions of) the SUO. **Rationale**: Accounts for loss of function through common mode and common causes of failure conditions of the SUO. Also see [9, 10, 12].
SWS-RV-1.SA4	To the extent possible, RTM shall be independent of the SUO. **Rationale**: Refines [SWS-RV-5]; associated to [SWS-RV-5.SA1.2.1]. Independence mitigates the potential impact that events or conditions affecting the SUO could have on RTM.
SWS-RV-3.SA1.2.1	RTM shall confirm that the inputs it receives are valid and trusted. **Rationale**: Refines [SWS-RV-3.SA1.2]. Supports identifying abnormal deviations in inputs, e.g., false positive/negative values. **Note.** This is analogous to the requirement F3269-IM-004 from [1]: *The Input Manager shall output assured data for Safety Monitor Inputs*.
SWS-RV-12	RTM shall include the means to clean up input data originating from different, distributed streams. **Rationale**: Associated to [SWS-RV-3.SA1.2.1]. Also see [14].
SWS-RTM-A2	The scope of RTM shall be covered by the state space of the integrated system. **Rationale**: Provides assurance that specification being observed is contained within the state space of the integrated system. **Note.** This is analogous to the requirement F3269-RTAS-005 from [1]: *RTA system fully implements the desired RTA system coverage*.
SWS-RTM-A3	RTM performance criteria shall be consistent with its level of safety criticality and the performance obligations of the SUO. **Rationale**: Accounting for the tradeoff between monitor safety criticality and functional performance of the SUO whilst defining monitor performance (e.g., warning / detection accuracy, and recovery activation rate). Higher safety criticality may mean a higher rate of false positive warnings is tolerable.

3.3.2 Backup Function

Table 3 lists the preliminary recommended requirements on the Backup function in RTA.

Table 3. Recommendations on the Backup Function of RTA

ID	Recommended Requirement
SWS-FO-1	Failover / fallback / recovery behavior shall cover all foreseeable operating conditions of the SUO. **Rationale**: Analogous to recommended requirement SWS-RV-3.SA3 (See Table 2).

Table 3. Recommendations on the Backup Function of RTA (Continued).

ID	Recommended Requirement
SWS-FO-2	Failover / fallback / recovery shall not be impacted, rendered inoperable or unavailable by the conditions that also impact (lead to failure conditions of) the SUO. **Rationale**: Analogous to recommended requirement SWS-RV-5 (See Table 2).
SWS-FO-3	The level of failover / fallback / recovery function availability shall be commensurate with the safety-criticality of the RTA capability. **Rationale**: Accounts for readiness of failover / fallback / recovery on demand or continuously. **Note.** This is analogous to the requirement F3269-RF-002 from [1]: *The Recovery Function shall ensure Recovery Function output is available at the time of selection*.
SWS-FO-4	The failover / fallback / recovery function shall not conflict with the (primary) function(s) of the SUO for which it is a fallback. **Rationale**: Accounts for consistency with primary function of the SUO. **Note.** This is analogous to the requirement F3269-RS-003 from [1]: *The only source of RTA output is the function selected by RTM, and there is always exactly one RTA output source*.

3.3.3 RTA Capability

Table 4 gives preliminary recommended requirements on the overall RTA capability.

Table 4. Recommendations on the RTA capability.

ID	Recommended Requirement
SWS-RV-3.SA1.1	RTA behavior shall include responses to normal deviations of the targets of RTM (specifications including variables and states). **Rationale**: Accounts for legitimate violations of the monitored specification, to trigger failover/contingency response.
SWS-RV-3.SA1.2	RTA behavior shall include responses to abnormal deviations of the targets of RTM. **Rationale**: Accounts for false positives/negative violations of the monitored specification, to trigger self-test.
SWS-RV-3.SA2	RTA behavior shall cover all foreseeable operating conditions of the SUO. **Rationale**: Provides assurance that failover occurs under all known nominal and off-nominal operating conditions under which the failover response will be used.
SWS-RTA-F1	RTA behaviors shall be risk-informed. **Rationale**: Provides for a risk-informed basis for RTA response, i.e., weighted cost of failure versus weighted benefit of success of RTA responses.

3.4 Architecture and Integration

Table 5 gives preliminary recommended requirements on the architecture and integration of the RTA capability.

Table 5. Recommendations on the architecture and integration of RTA.

ID	Recommended Requirement
SWS-RV-6	Assured RTM shall safely compose with the SUO. **Rationale**: See [9, 10]. Associated to recommended requirement SWS-RTA-C1 (See Table 6).
SWS-RV-6.1	RTM shall safely compose with the failover / fallback / recovery function. **Rationale**: Analogous to recommended requierment SWS-RV-6. Provides assurance that the RTM integration with the failover function does not lead to RTA failure conditions.
SWS-RV-6-SA5	RTM shall not exhibit unintended behavior when integrated with the SUO. **Rationale**: Refines recommended requirement SWS-RV-6. For assurance of the absence of hazardous emergent behavior of the integrated system.
SWS-RV-6-SA5.1	RTM shall not exhibit unintended behavior on integration with the failover / fallback / recovery function. **Rationale**: Analogous to recommended requirement SWS-RV-6-SA5. For assurance of the absence of hazardous emergent behavior of the RTA.
SWS-RV-6-SA6	RTM shall not compromise correct behavior of the SUO upon integration. **Rationale**: Refines recommended requirement SWS-RV-6. Associated to recommended requirements SWS-RV-5.SA4 (see Table 2) and SWS-RV-1.SA1.2.1 (see Table 1). For assurance that RTM function does not conflict with functions of the SUO.
SWS-RV-6-SA6.1	RTM shall not compromise correct behavior of the failover / fallback / recovery function upon integration as an RTA function. **Rationale**: Analogous to recommended requirement SWS-RV-6-SA6. Provides assurance that the RTM function does not conflict with failover / fallback / recovery function.
SWS-RV-13	For monitoring complex distributed systems, RTM shall span the entire distributed system. **Rationale**: See [15].
SWS-RTA-AC1	RTM or RTA being integrated into an aeronautical application should follow the system design and analysis guidance in Advisory Circular 25.1309-1A [16]. **Rationale**: Leverages fail-safe design principles codified in FAA AC 25.1309-1A [16].

3.5 Other

This section gives preliminary recommendations on the implementation, configurations, usage, and non-functional attributes, as listed in Table 6

Table 6. Recommendations on the implementation, configurations, usage, and non-functional attributes of RTA.

ID	Recommended Requirement
SWS-RV-SWI1	Assured RTM A software-only implementation of RTA should account for the considerations on software contributions to system hazards as documented in the NASA Software Assurance and Safety Standard, NASA-STD-8739.8B [17].

Table 6. Recommendations on the implementation, configurations, usage, and non-functional attributes of RTA (Continued).

ID	Recommended Requirement
	Rationale: Leverages domain knowledge and best practices for developing safety-critical software, subsuming individual implementation specific requirements. For example, (a) SWS-RV-8: RTM implementation shall not be susceptible to unsafe or undefined behaviors such as buffer and floating-point overflows; (b) SWS-RV-9: Assured RV shall not introduce security vulnerabilities into a system; (c) SWS-RV-11: RTM shall detect impending violations of the specification and invoke the safety controller in time to preserve safe operation.
	Note. NASA-STD-8739.8B.complements other assurance guidance documents used in safety-critical aeronautical applications (such as DO-178C and DO-278).
SWS-RTA-C1	An RTA scheme employing a multi-level monitoring configuration shall be internally consistent and consistent with the functions of the SUO.
	Rationale: Associated to recommended requirement SWS-RV-6 (see Table 5). Also gives assurance that multiple monitors and respective failover functions do not conflict with each other and with functions provided by the SUO.
SWS-RTA-R1	RTM data relevant for analysis performed post operations shall be recorded.
	Rationale: Support for variety of post operational analysis including detection of leading indicators for incidents, post-incident analysis, diagnosis of failure conditions.
SWS-RTA-NFA1	RTA responses shall be stable.
	Rationale: RTA switching between failover function and primary function provided by the SUO should not introduce instability.
	Note. Related to requirement F3269-RTAS-004 from [1]. Transitions while switching between different sources for RTA output adequately addresses issues of transients, timing, latency, and stability.

4 Concluding Remarks

Discussion and Summary This report gives a preliminary collection of recommendations on RTA, which has been synthesized from work primarily performed under support from NASA [9,10,12–15,18–20]. These recommendations have been examined in a lightweight way against ASTM F3269-21 [1], a consensus-based industry standard that specifies the requirements and practices for using an RTA architecture to bound the behavior of complex aircraft functions. The intent was to use the standard as an external *sanity check* to examine and validate how the recommended requirements in this report align with those in the standard.

For the most part, the recommendations here align with those in the ASTM F3269-21 standard, though there are some notable differences: first, the ASTM F3269-21 standard advocates the use of RTA with untrusted functions that have not undergone traditional development assurance processes used to aid airworthiness and type certification activities. In contrast this report explicitly does not relieve the assurance obligations on the SUO despite the use of RTA. The rationale is that safety must be demonstrated for the *integrated system*, and assurance of the SUO is a mitigation mechanism that provides defense in depth against potential hazardous interactions that may emerge.

In fact, ASTM F3269-21 states that "*The designer should not use RTA as a substitute for poor complex function design and/or implementation*" which, this report considers to be consistent with the intent of recommending that assurance processes continue to apply to the SUO. Indeed, the corresponding assurance objectives serve as the basis to determine whether or not a complex function design and/or implementation is poor. Moreover, for complex functions that are implemented using novel emerging technologies such as ML, there are assurance challenges as yet unanswered despite using RTA [21].

Secondly, the ASTM F3269-21 standard provides specific requirements on the RTA architecture, also suggesting

particular assurance techniques such as *dynamic consistency checking*. In contrast, the recommendations here intentionally refrain from the application of particular techniques, rather recommending broad classes of techniques, e.g., the use of formal methods, or assurance arguments (see recommended requirements SWS-RV-2, and SWS-RV-7.1, SWS-RV-7.2, respectively, in Table 1). The intent here is to leave the choice and suitability of specific assurance techniques to, respectively, the applicants and regulators, who must collectively determine whether or not those techniques are sufficient to meet the safety intent of any applicable regulations.

Next Steps The recommendations presented here are not a complete collection, and more work is need to close the gaps in their coverage of scope. In particular, the recommendations can be made more comprehensive across all categories of recommendations, i.e., the development process, the provided functions, architecture and integration, and especially for each of RTA implementation, configurations (e.g., RTA architectures that involve multiple levels of monitoring and backup functions), usage (e.g., applying RTA under complex mode switching rules that could potentially defeat the monitoring and failover capabilities), and non-functional attributes (Table 6).

Additionally, the recommended requirements across the different recommendation categories are related and some of these associations have been identified. A natural next step is to provide complete coverage of these interrelations and to identify and reconcile potential requirements conflicts.

The rationale that justifies the recommendations made can be constructed in the form of a meta-assurance case shows how the proposed recommendations support the general characteristics for RTA (Section 3.1), and why, collectively, they are consistent with the intent of RTA.

Along these lines, since RTA is one among many candidate architectural options for achieving high assurance, an additional next step is to explore the relationship of the recommended requirements with the requirements that emerge from other system-level development and safety processes used, such as ARP 4754 [22] and ARP 4761 [23].

References

[1] ASTM International, "Standard Practice for Methods to Safely Bound Behavior of Aircraft Systems Containing Complex Functions Using Run-Time Assurance." ASTM F3269-21, November 2021.

[2] J. D. Schierman, M. D. DeVore, N. D. Richards, and M. A. Clark, "Runtime assurance for autonomous aerospace systems," *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 12, pp. 2205–2217, 2020.

[3] A. Tiwari, B. Dutertre, D. Jovanović, T. de Candia, P. D. Lincoln, J. Rushby, D. Sadigh, and S. Seshia, "Safety envelope for security," in *Proceedings of the 3rd International Conference on High Confidence Networked Systems*, HiCoNS '14, pp. 85–94, ACM, 2014.

[4] R. Bloem, B. Könighofer, R. Könighofer, and C. Wang, "Shield Synthesis: Runtime Enforcement for Reactive Systems," in *Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2015)* (C. Baier and C. Tinelli, eds.), vol. 9035 of *LNCS*, pp. 533–548, Springer, 2015.

[5] J. Carpenter, K. Gahan, and R. Cobb, "Automatic-Ground Collision Avoidance System (Auto-GCAS) for Performance Limited Aircraft," in *AIAA Aviation 2019 Forum*, AIAA, 2019.

[6] F. van Diggelen and A. Brown, "Mathematical aspects of GPS RAIM," in *Proceedings of 1994 IEEE Position, Location and Navigation Symposium - PLANS'94*, pp. 733–738, 1994.

[7] H. Powrie and C. Fisher, "Engine health monitoring: Towards total prognostics," in *1999 IEEE Aerospace Conference. Proceedings (Cat. No.99TH8403)*, vol. 3, pp. 11–20, 1999.

[8] D. J. Snowden and M. E. Boone, "A leader's framework for decision making," *Harvard Business Review*, vol. 85, no. 11, p. 68, 2007.

[9] A. Goodloe, "Challenges in high-assurance runtime verification," in *Leveraging Applications of Formal Methods, Verification and Validation: Foundational Techniques ISoLA 2016* (T. Margaria and B. Steffen, eds.), vol. 9952 of *LNCS*, (Cham), pp. 446–460, Springer International Publishing, 2016.

[10] I. Perez, F. Dedden, and A. Goodloe, "Copilot 3," Technical Report NASA/TM-2020-220587, NASA Langley Research Center, April 2020.

[11] A. Dutle, C. Muñoz, E. Conrad, A. Goodloe, L. Titolo, I. Perez, S. Balachandran, D. Giannakopoulou, A. Mavridou, and T. Pressburger, "From Requirements to Autonomous Flight: An Overview of the Monitoring ICAROUS Project," *Electronic Proceedings in Theoretical Computer Science*, vol. 329, pp. 23–30, December 2020.

[12] K. Y. Rozier and J. Schumann, "R2U2: Tool Overview," in *RV-CuBES 2017. An International Workshop on Competitions, Usability, Benchmarks, Evaluation, and Standardisation for Runtime Verification Tools* (G. Reger and K. Havelund, eds.), vol. 3 of *Kalpa Publications in Computing*, pp. 138–156, EasyChair, 2017.

[13] I. Perez, A. Mavridou, T. Pressburger, A. Goodloe, and D. Giannakopoulou, "Automated translation of natural language requirements to runtime monitors," in *Tools and Algorithms for the Construction and Analysis of Systems* (D. Fisman and G. Rosu, eds.), (Cham), pp. 387–395, Springer International Publishing, 2022.

[14] P. Mehlitz, D. Giannakopoulou, and N. Shafiei, "Analyzing airspace data with race," in *2019 IEEE/AIAA 38th Digital Avionics Systems Conference (DASC)*, pp. 1–10, 2019.

[15] N. Shafiei, K. Havelund, and P. Mehlitz, "Actor-Based Runtime Verification with MESA," in *Runtime Verification* (J. Deshmukh and D. Ničković, eds.), (Cham), pp. 221–240, Springer International Publishing, 2020.

[16] Federal Aviation Administration, ANM-112, "System design and analysis." AC 25.1309-1A, June 1988.

[17] NASA, "Software Assurance and Software Safety." NASA-STD-8739.8B, August 2022.

[18] A. Dutle, C. Muñoz, E. Conrad, A. Goodloe, L. Titolo, I. Perez, S. Balachandran, D. Giannakopoulou, A. Mavridou, and T. Pressburger, "From Requirements to Autonomous Flight: An Overview of the Monitoring ICAROUS Project," in *Proceedings of the 2nd Workshop on Formal Methods for Autonomous Systems* (M. Luckcuck and M. Farrell, eds.), vol. 329 of *Electronic Proceedings in Theoretical Computer Science (EPTCS)*, pp. 23–30, arXiv preprint arXiv:2012.03745, 2020.

[19] P. Moosbrugger, K. Y. Rozier, and J. Schumann, "R2U2: Monitoring and Diagnosis of Security Threats for Unmanned Aerial Systems," *Formal Methods in System Design*, vol. 51, pp. 31–61, August 2017.

[20] S. Bharadwaj, S. Carr, N. Neogi, and U. Topcu, "Decentralized Control Synthesis for Air Traffic Management in Urban Air Mobility," *IEEE Transactions on Control of Network Systems*, vol. 8, no. 2, pp. 598–608, 2021.

[21] J. Fenn, M. Nicholson, G. Pai, and M. Wilkinson, "Architecting Safer Autonomous Aviation Systems," in *Proceedings of the 31st Safety-Critical Systems Symposium (SSS 2023)* (M. Parsons and M. Nicholson, eds.), SCSC, February 2023.

[22] S-18, Aircraft And System Development And Safety Assessment Committee, *ARP 4754A, Guidelines for Development of Civil Aircraft and Systems*. SAE International, Dec. 2010.

[23] S-18, Aircraft And System Development And Safety Assessment Committee, *ARP 4761, Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment*. SAE International, Dec. 1996.

A Acronyms

Auto-GCAS	Automatic Ground Collision Avoidance System
CFIT	Controlled Flight into Terrain
ECTM	Engine Condition Trend Monitoring
EHM	Engine Health Monitoring
FAA	Federal Aviation Administration
FDIR	Fault Detection, Isolation, and Recovery
GPS	Global Positioning System
IVHM	Integrated Vehicle Health Management
ML	Machine Learning
RAIM	Receiver Autonomous Integrity Monitoring
RTA	Runtime Assurance
RTM	Runtime Monitoring
RV	Runtime Verification
SUO	System Under Observation