

**OFFICIAL
Uncontrolled If Printed**



Australian Government
Defence

**AUSTRALIAN NAVAL CLASSIFICATION AUTHORITY MANUAL
(VOLUME 2)**

**DIVISION 2: CORE DESIGN RULES
CHAPTER 03: SOFTWARE SYSTEMS
PART 1: ANC RULES**



This document is issued for use by Defence and Defence Industry personnel and is effective forthwith.

A handwritten signature in black ink, appearing to read 'CN Dagg'.

CN Dagg, CSC
Assistant Secretary
Australian Naval Classification Authority
Department of Defence
CANBERRA ACT 2600
May 2024 Edition

**OFFICIAL
Uncontrolled If Printed**

OFFICIAL
Uncontrolled If Printed

© Commonwealth of Australia 2024

This work is copyright. Apart from any use as permitted under the [Copyright Act 1968](#)¹, no part may be reproduced by any process without prior written permission from the Department of Defence.

All classified Defence information is protected from unauthorised disclosure and it is an offence to release classified information under the [Criminal Code Act 1995](#)² and the [Privacy Act 1988](#)³. Information contained in Defence publications may only be released in accordance with the [Defence Security Principles Framework](#)⁴.

ANCA Manual (Volume 2)

Division 2: Core Design Rules, Chapter 03: Software Systems, Part 1: ANC Rules, May 2024 Edition

Developer:

Australian Naval Classification Authority

¹ <https://www.legislation.gov.au/Series/C1968A00063>

² <https://www.legislation.gov.au/Series/C2004A04868>

³ <https://www.legislation.gov.au/Series/C2004A03712>

⁴ <http://drnet/AssociateSecretary/security/policy/Pages/dspf.aspx>

AUSTRALIAN NAVAL CLASSIFICATION RULES

First issued	May 2024
Reissue date	N/A
Issued by	CN Dagg, CSC, AS ANCA
Document management	This volume will be reviewed periodically from the date of issue, but sooner if necessitated by business requirements, and to ensure it continues to meet the intent of Defence policy.
Availability	The latest version of this volume is only available from the Defence Australia website. Its currency cannot be guaranteed if sourced from other locations. It is available for public release.
Policy domain	Defence Seaworthiness
Accountable Officer	Australian Naval Classification Authority
Publication Owner	Defence Seaworthiness Authority (DSWA)
Policy contact	anca.communications@defence.gov.au
Structure	see Contents ⁵
Cancellation	N/A
Definitions	Definitions that apply to this volume are located in the Division 1, Part 1 Annex A.

⁵ <https://www.defence.gov.au/business-industry/industry-governance/australian-naval-classification-authority/australian-naval-classification-rules>

AMENDMENTS

Proposals for amendments to the ANCA Manual (Volume 2) may be sent to:

Australian Naval Classification Authority

Mail to: anca.correspondence@defence.gov.au

EDITIONS

Edition	Edition	Amendment type	Effective
May 2024	Original issue		May 2024

Division 2: Core Design Rules
Part 1: ANC Rules

Chapter 03: Software Systems

Contents

Rule 0.	Goal.....	2
Rule 1.	General Software Requirements.....	2
Rule 2.	Software Engineering Lifecycle Processes.....	3
Rule 3.	System and Software Design Safety.....	3
Rule 4.	Software Assurance Processes.....	4
Rule 5.	Software Security.....	4
Rule 6.	Software Control Environmental Compliance Aspects.....	5

Australian Naval Classification Rules**Rule 0. Goal**

- 0.1 The Naval Vessel's software systems shall be designed, developed, and maintained to:
- 0.1.1 Minimise personnel safety risk;
 - 0.1.2 Minimise environmental risk;
 - 0.1.3 Minimise security risk;
 - 0.1.4 A Software Assurance Process;
 - 0.1.5 Operate predictably and fail safely in the event of failure;
 - 0.1.6 Minimise adverse impact to mission critical functions and other integrated systems in the event of failure, damage, or loss; and
 - 0.1.7 Provide post damage capability as required by the Operating and Support Intent (OSI).

Rule 1. General Software Requirements**Functional Objective**

- 1.1 The Naval Vessel's systems which incorporate software shall be designed, developed, tested, deployed and maintained to meet the goal of this chapter.

Performance Requirements

- 1.2 Software systems shall be designed, developed, tested, and maintained throughout life, so that when operated as intended, the following are satisfied:
- 1.2.1 For the Foreseeable Operating Conditions, the Naval Vessel shall be safe to operate and prevent loss of assets and injury to embarked personnel;
 - 1.2.2 For foreseeable damage events, the Naval Vessel shall maintain the availability of Essential Safety Functions; and
 - 1.2.3 For foreseeable failure events, the Naval Vessels systems shall fail safely and predictably and not cause failures in integrated and dependant systems.
- 1.3 Software shall be integrated as part of the overall Naval Vessel system architecture in accordance with Chapter 1 Rule 4 *Systems Architecture*.

Note: This chapter specifies the generic requirements for software and applies to any generic software on Defence vessels that could pose a safety or environmental risk.

Note: For the formal definition of Foreseeable Operating Conditions and other definitions refer to ANCR Division 1 Annex A – *Definitions and Abbreviations*.

Rule 2. Software Engineering Lifecycle Processes**Functional Objective**

- 2.1 The Naval Vessel's software systems shall be designed, developed, tested, deployed and maintained according to Software Engineering lifecycle processes.

Performance Requirements

- 2.2 Development of software systems shall be part of the Systems Engineering and Software Engineering process for the Naval Vessel development lifecycle.
- 2.3 Safety-significant functions shall be identified and managed at all stages of the software system life cycle.
- 2.4 Software Engineering lifecycle processes shall be performed on Software Systems at the respective stages of the system development and operational phases.

Rule 3. System and Software Design Safety**Functional Objective**

- 3.1 The Naval Vessel's software systems shall be designed to operate safely and predictably in all Foreseeable Operating and Support Conditions.

Performance Requirements

- 3.2 Software functionality and correct software behaviour shall be part of the scope of Systems Engineering and Software Engineering Program.
- 3.3 Software Systems shall be identified where they are either the cause of a potential safety hazard, or the control to a safety hazard.
- 3.4 Safety hazards and controls attributable to software shall be identifiable separately from other system safety hazards within the system Safety Case.

Note: Chapter 1 Rule 3 *System Safety* details the requirements for the Safety Case.

- 3.5 The Software Engineering Program shall perform Software Assurance Processes to ensure the technical quality and safety of Software Systems.
- 3.6 The failure of safety-significant software functions shall be detected, isolated (where achievable without/limited degradation and mission compromise) and recovered from such that catastrophic hazardous events do not ensue.
- 3.7 A failure or unspecified behaviour of the software system shall not result in:
- 3.7.1 An event that escalates to a safety hazard;
- 3.7.2 Impairment of the control of a hazard;
- 3.7.3 Impairment of recovery from a hazard; or
- 3.7.4 Impairment or detrimental effect on other integrated and dependent systems.

Rule 4. Software Assurance Processes**Functional Objective**

- 4.1 Software Assurance Processes shall provide confidence in the Software System across the software lifecycle, and objective evidence that demonstrates safety requirements and software quality requirements are met.

Performance Requirements

- 4.2 The Software Engineering Program shall ensure that all Software Systems are classified into Assurance Levels. The Software Systems shall be designed, developed, tested, and managed according to those levels.
- 4.3 Software Assurance Processes shall be performed throughout the life of the software to ensure risk to personnel, systems, and the environment is eliminated or reduced SFARP.
- 4.4 Where there is potential for a Software System to invoke a hazard, impair the mitigation/control of a hazard, or impair recovery of a hazardous event, the Software System shall meet the conditions of the respective Assurance Level.
- 4.5 Where a Software System is identified as the potential cause of a hazard, the Software System shall be designed, developed, and implemented consistent with the assessed Assurance Level and associated safety risk.
- 4.6 Where a Software System is used as the control to a hazard, the Software System shall be designed, developed and implemented consistent with the assessed Assurance Level and associated safety risk.
- 4.7 Software shall have safeguards to ensure continued operations of Mission Critical functions and services during error states and damage to systems hosting the software.

Rule 5. Software Security**Functional Objective**

- 5.1 The Naval Vessel's software systems shall be designed to operate securely in all Foreseeable Operating Conditions.

Performance Requirements

- 5.2 The design of the Naval Vessel's security architecture shall include the Cyber security aspects of Software Systems.
- 5.3 Software security shall be addressed as part of the Cyberworthiness and Software Engineering programs to ensure systems are designed as intrinsically secure.

Note: Chapter 02 *Cyberworthiness* details the applicable requirements for Cyber security.

- 5.4 Software Systems shall be designed and developed to support the following Cyber security functions:
- 5.4.1 Identification of Cyber security threats and incidents;
- 5.4.2 Protection against Cyber security threats;
- 5.4.3 Detection of Cyber security incidents; and

5.4.4 Respond to Cyber security threats and incidents.

5.4.5 Recover from Cyber security incidents

Rule 6. Software Control Environmental Compliance Aspects

Functional Objective

6.1 Systems that are controlled by software, or include software subsystems, shall be designed so as not to adversely affect the environment.

Performance Requirements

6.2 A failure or unspecified behaviour of Software Systems shall not result in:

6.2.1 Inadvertent sensor emissions either above or below water posing risk to marine wildlife;

6.2.2 Inadvertent release of weapons or effectors;

6.2.3 Inadvertent release of liquid pollutants including sewerage, contaminants, fuel, oil, waste, or ballast; and

6.2.4 Release of unacceptable levels of airborne pollution and contaminants.