RBAT
Method description

The Risk-Based Assessment Tool (RBAT) method description

Rev. 4.2

Date: 24/04/2025

EMSA

About this study:

This report was commissioned by the European Maritime Safety Agency (EMSA) under framework contract EMSA/OP/10/2020.

Authors:

Kenneth Kvinnesland (DNV)
Åsa Snilstveit Hoem (DNV)
Sondre Øie (DNV)
Remi Brensdal Pedersen (DNV)

Recommended citation:

European Maritime Safety Agency (2024), *RBAT - Method description*, EMSA, Lisbon

Legal notice:

Copyright notice[1]:

This report and any associated materials are available online at www.emsa.europa.eu

[1] The copyright of EMSA is compatible with the CC BY 4.0 license.

Document History

Version	Date	Changes	Prepared	Approved
1.0	18/10/2024	DRAFT issue	Åsa S. Hoem	Kenneth Kvinnesland
2.0	27/11/2024	Updated after comments from EMSA	Åsa S. Hoem	Kenneth Kvinnesland
3.0	17/02/2025	Updated to match the RBAT Tool	Åsa S. Hoem	Kenneth Kvinnesland
3.1	25/02/2025	Updated cover page and table for Operational states	Åsa S. Hoem	Kenneth Kvinnesland
4.0	02/04/2025	Updated figures and tables. Corrected definitions and terms after QA	Åsa S. Hoem	Kenneth Kvinnesland
4.1	08/04/2025	Updated example and guidance on mitigation analysis	Åsa S. Hoem	Kenneth Kvinnesland
4.2	24/04/2025	Updated Appendix E	Åsa S. Hoem	Kenneth Kvinnesland

Table of Contents

List of Tables

List of Figures

List of Abbreviations

AIS	Automatic Identification System
ALARP	As Low As Reasonably Practicable
ANF	Autonomous Navigation Function
ANS	Autonomous Navigation System
ConOps	Concept of Operation
ECDIS	Electronic Chart Display and Information System
ESD	Emergency Shutdown Systems
CPU	Central Processing Unit
FDIR	Fault Detection, Isolation and Recovery
FMEA	Failure Mode and Effect Analysis
FPGA	Field Programmable Gate Array
FSA	Formal Safety Assessment
GCAS	Ground Collision Avoidance System
GNSS	Global Navigation Satellite System
HMI	Human Software Interface
HSE	Health, Safety and Environment
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
IR	Infrared
PSF	Performance Shaping Factor
RBAT	Risk Based Assessment Tool
ROC	Remote Operation Centre
RCM	Risk Control Measure
UoA	Use of Automation
VPS	Voyage Planning System

Definitions

Terms	Definitions
abnormal situation	Any unexpected event or condition that deviates from normal operations and poses a potential risk to the safety of the vessel, its crew, passengers, or the environment. These situations require immediate attention and appropriate action to prevent accidents or mitigate their consequences. In RBAT, unsafe conditions/modes represent such deviations.
accident	An unintended event involving fatality, injury, ship loss or damage, other property loss or damage, or environmental damage (IMO, 2018).
accident category	A designation of accidents reported in statistical tables according to their nature, e.g., fire, collision, grounding, etc. (IMO, 2018).
accident scenario	A set of events and conditions which, in the way they are combined, results in losses.
active human supervisory control	A human agent is responsible for continuously supervising the automated performance of a control function with the purpose of being able to intervene at any stage based on judgements about how to best act upon the situation. Because active supervision provides an opportunity for the human agent to continuously create situational awareness, it can be beneficial in cases where there is limited time available to intervene.
agent	Human or software (computer or computer system) responsible for performing or supervising control actions.
annunciated failure	An annunciated failure is one which fails 'actively', i.e., in such a manner as to inform operators of the failure by virtue of system generated cues such as visual and/or audible alerts. In RBAT, annunciation refers to cues generated by the performing agent, or other systems involved in performing the control function. Cues may also be generated by independent systems (e.g., supervisory control agents), if implemented.
automation	The execution by a 'software' *agent* (usually a computer) of a function that was previously carried out by a human" (Parasuraman & Riley, 1997).
autonomy	"Technology operates alone". See sub-chapter 3.3.1 in Report 1oo2 for Part 1 of RBAT (DNV GL, 2020a).
causal factors	A single or the minimum combination of causes which, in the presence of an enabling condition or event, can initiate an accident scenario. In RBAT, causal factors concern the system under control, and not events or conditions in the operating environment (see "Enabling condition/event").
common cause failures	Failures of multiple items, which would otherwise be considered independent of one another resulting from a single cause (IEC, 2018).
ConOps	Document describing the characteristics of a proposed system from the viewpoint of an individual who will use that system (ISO/IEC/IEEE 15288:2015).
context	External and internal environment in which the organization seeks to achieve its objectives (ISO, 2009).
control	Purposeful action on or in a process to meet specified objectives (IEC, 2013).
control action	Acquisition of information, analysis of information, decision-making, or implementation of physical actions performed as part of a *control function.*
control function	Control exerted by humans or software for the accomplishment of a function (adapted from IEC, 2000).
degraded state	A state where performance capabilities are degraded e.g., by failures or inadequate capabilities, but normal operations can be safely continued if operational restrictions and/or compensating measures are implemented.
effect on [safety, environment, ship or uptime]	The consequence of a worst-case outcome in terms of losses.
enabling conditions	Conditions which must occur or be present in the operation for causal factors to initiate scenarios which result in losses (i.e., accidents). In RBAT: - Enabling conditions are not to be confused with causal factors

Terms	Definitions
	- Losses cannot occur without the presence of enabling conditions
essential continuous function	A function which is required to continuously perform according to its specifications to maintain the safety of the vessel during one or more of its normal type of operations.
exposure level	The estimated occurrence of an enabling event or duration of an enabling condition.
fallback state	Designed state that can be entered when the autoremote vessel system cannot stay within the operational envelope (DNV, 2024) In previous RBAT reports the term *Minimum Risk Condition (MRC)* was used with the same definition.
failure	Loss of the ability of an item to perform the required (specified) function within the limits set for its intended use (DNV, 2021b).
failure cause	Set of circumstances that leads to failure (IEC, 2018).
failure effect	A description of the operation of a system or an item as the result of a failure, i.e., the consequence(s) a failure mode has on the operation, function or status of a system or an item (SAE, 1996).
failure frequency	The number of failures expressed in failures per unit of time (calendar or operational).
failure mechanism	Process that leads to failure (IEC, 2018). The process may be physical, chemical, logical, psychological or a combination thereof.
failure mode	The observed way in which the failure (of an item) occurs (adapted from SAE, 1996 and DNV, 2021b).
fault detection, isolation, and recovery (FDIR)	A control function's internal capacity to withstand, isolate, self-recover or initiate recovery from a failure situation. In case system self-monitoring identifies a fault, what type, and its location, examples of recoveries include: - Switch-off of a faulty equipment - Switch-over from a faulty equipment to a redundant equipment In RBAT, FDIR represents a type of *mitigation* that fully or partly rely on mechanisms and resources located within the agent responsible for performing the control action being analysed.
function	Specific purpose or objective to be accomplished, that can be specified or described without reference to the physical means of achieving it (IEC, 2020). In RBAT functions refer to how systems perform to successfully accomplish operations. Sub-functions are offspring (sub-goals) of higher-level, parent function.
functional goal	The performance objectives that shall be satisfied to achieve a higher-level corresponding function (adapted from IEC, 2009). In RBAT, navigation, manoeuvring, and communication are examples of functional goals located at the highest level in the Function Tree.
function tree	Hierarchical breakdown of high-level functional goals into a set of sub-functions.
hazard	A potential to threaten human life, health, property or the environment (IMO, 2018). For the purpose of RBAT, this is interpreted as the source of harm which, unless managed, has the potential to cause accidents involving harm or losses. In terms of *safety*, a hazard therefore often refers to conditions, situations, or states in which various sources of energy, biological or chemical agents are present.
hierarchical goal structure	Relationship between a goal and sub-goals structured in a hierarchical order (adapted from IEC, 2009). In RBAT, the function tree has a hierarchical goal structure.
human-automation interaction	The way a human is affected by, controls, and receives information from automation while performing a task (Sheridan & Parasuraman, 2006).
human error	Discrepancy between the human action taken or omitted, and that intended or required to achieve a task goal (adapted from IEC, 2018).
incident	Occurrence of any event, other than an accident, that is associated with a ship or its required infrastructure and affects or could affect its safety.

Terms	Definitions
initiating event	The first of a sequence of events leading to a hazardous situation or accident (IMO, 2018).
item	Subject being considered (IEC, 2018).
loss	A loss involves something of value to stakeholders. Losses may include a loss of human life or human injury, property damage, environmental pollution, loss of mission, loss of reputation, loss or leak of sensitive information, or any other loss that is unacceptable to the stakeholders (Leveson & Thomas, 2018).
mission	The commercial, political (e.g., defence) or public intentions which have contributed to and justifies the vessel concept development and operation.
mission model	Hierarchical breakdown of a vessel mission into a set of mission phases and operations.
mission phase	Subdivisions of the mission typically characterized by a recognizable shift in where the vessel is located in terms of geographical surroundings, or the start and end of one or more operations.
mitigation	The goal of preventing initiating events from resulting in accidents. In RBAT “mitigations” refer to either FDIR or additional mitigation measures.
mitigation measure	Specific goal of a single function or combined set of different functions to prevent an unsafe condition from resulting in an accident (i.e., losses). In RBAT, mitigation measures shall not be adversely affected by the initiating event or the actions of any other mitigation measures which have already been in effect.
mitigation effectiveness	The effectiveness of the set of FDIR and/or mitigation measures that is identified as relevant when it comes to preventing a specific accident scenario from occurring. In RBAT mitigation effectiveness is determined qualitatively, and the effectiveness scale has a range from Low to Extremely High.
operations	Activities performed as part of a mission phase in order to achieve the mission goal. Sub-operations are offspring (sub-goals) of higher level, parent operations.
operational envelope	Boundaries of pre-defined operational, environmental and system conditions in which an autonomous or remotely operated vessel can safely execute its normal operations (adapted from DNV, 2024). Operational envelope is used towards the overall concept or the vessel operations.
operational goals	The ultimate purposes of a vessel (adapted from IEC, 2009). In RBAT operational goals are explained in terms of the mission, mission phases and operations.
operational restrictions	Measures taken to stay within the *operational envelope* in the presence of hazardous *enabling conditions* or when experiencing *failures* which puts the controlled system in a *degraded state.*
other roles involved	Humans which, in addition to the performing agent, must act for the control action to be executed.
other systems involved	Systems which, in addition to the performing agent, must function for the control action to be executed.
passive human supervisory control	A human agent is responsible for being available to supervise the automated performance of a control function and intervene upon requests (e.g., an alert) generated by the system according to pre-defined parameters.
performance	The performance of a technology is its ability to provide its specified functions (DNV, 2021b). These functions contribute to safety/reliability as well as the output or value generated by the system, equipment, or component when in operation.
performance shaping factors	Human, workplace, or other contextual factors which have a significant effect on an operator’s or crew of operator’s performance.
performing agent	A human or software responsible for performing a control action.
preventive safeguards	Measures implemented to prevent an initiating event from occurring. In RBAT, inspection, testing, and maintenance are examples of preventive safeguards.

Terms	Definitions
process	Set of interrelated or interacting activities that transforms inputs into outputs (IEC, 2018)
reliability	The ability of an item to perform a required function under given conditions for a given time interval or at a specified condition (DNV, 2021b). In quantitative terms, it is one (1) minus the failure probability.
recovery actions	Actions taken to recover the system from a degraded or failed state and back to a state which allow normal and safe operations to be continued.
redundancy (of a system)	Having multiple capabilities for performing the same function, typically in parallel (DNV, 2021b).
risk control measure	A means of controlling a single element of risk (IMO, 2018). This may refer to measures taken to reduce the risks to the operation of the system, and to the health and safety of personnel associated with it or in its vicinity by (DNV, 2021b): — reduction in the probability of failure — mitigation of the consequences of failure In RBAT, the order of preference of risk control measures for a function is: a) inherent safety b) prevention (only given risk reduction credit for if alternative methods are used, see 2.3.5) c) built-in detection and d) built-in control (FDIR) e) mitigation in form of detection and control that is independent of the function being analysed. f) operational restrictions aiming to reducing the consequences of an accident if it were to occur Note that due to the problem of determining the probability for systematic failures in control systems, the standard RBAT approach focus on risk control measures that mitigate the consequences of failure, rather than measures that aim to reduce the probability of failure. However, if a credible argument for reduction in the probability of failure can be made, it is possible to take credit for that. See *Step 17: Alternative approaches for determining risk levels.*
risk level	In RBAT, the risk level is determined based on three key factors: worst-case outcome severity, mitigation effectiveness, and exposure to enabling conditions.
scenario	See “accident scenario”.
severity (level)	Relative ranking of potential or actual consequences of a failure or a fault (IEC, 2018).
situational awareness	Situational awareness or situation awareness (SA) is the perception of environmental elements and events with respect to time or space, the comprehension of their meaning, and the projection of their future status (Endsley, 1995).
supervision	A role with an explicit responsibility to supervise system performance and detect abnormalities so that the desired outcome can be achieved through implementation of corrective responses. In RBAT, mitigation measures are corrective responses.
supervisory control agent	An *agent* with an explicit responsibility to perform supervision. A supervisory control agent must be independent of the performing agent it is supervising.
system	Combination of interacting elements organized to achieve one or more stated purposes, i.e., goals (IEC, 2018).
systematic failure	An event which occurs even if no individual component in the system has failed. See Appendix D for definition.
task	A set of [control] actions taken by humans to enable functions and perform operations. A task may involve interactions with several different functions, but also with humans. Task goals is the same as *operations*.
unannunciated failures	An unannunciated failure is one which is hidden, latent or in any way fails ‘passively’, i.e., in such a manner as to not inform operators of the failure by virtue of system generated cues, or the provided information is misleading, incomplete, or not presented in due time.

Terms	Definitions
unsafe condition	The state or mode when a system operates outside its operational envelope due to functional failures or exceeded capabilities and, which if left unmitigated, has the potential to cause an accident.
uptime	Measure of system reliability, expressed as the percentage of time a machine, typically a computer, has been working and available. Uptime is the opposite of downtime (source: https://en.wikipedia.org/wiki/Uptime)
worst-case outcome	The most severe credible outcome of an unsafe condition when assuming there is no mitigation. In RBAT, worst-case outcomes assume the contextual presence of a hazardous enabling condition or event. For example, loss of steering (an unsafe condition) close to shore (a hazard) results in a grounding (a worst-case outcome).

1. INTRODUCTION

1.2 Background

EMSA has contracted DNV to perform a functional study for developing a Risk-Based Assessment Tool (RBAT) for Maritime Autonomous Surface Ships (MASS) concepts. The purpose of the RBAT methodology is to risk assess whether increased or new ways of using automation and remote operation is as safe or safer than conventional shipping.

As outlined in DNV's proposal (DNV, 2020a) and EMSA's Tender Specifications (EMSA, 2020), the RBAT study consists of three parts:

- Part 1: Develop a framework for a generic MASS risk assessment tool.
- Part 2: Test the risk assessment tool on specific cases and develop a software tool prototype.
- Part 3: Re-iterate testing on more complex cases and finalise the software tool.

This report builds on the Final report (issued December 6th, 2024).

2. METHODOLOGY STEP-BY-STEP GUIDANCE TO THE RBAT METHODOLOGY

The RBAT methodology consists of five main parts:

1. Describe use of automation (and remote control)
2. Perform hazard analysis
3. Perform mitigation analysis
4. Perform risk evaluation
5. Address risk control

The following sub-chapters present these five main parts as consisting of 19 steps.

2.1 Part 1: Describe use of automation (and remote control)

The purpose of describing the Use of Automation (UoA) and remote control is to:

- Describe the vessels mission (operational goals) (Step 1)
- Describe the control functions that are affected by automation or remote control (Step 2)
- Understand how the control functions are allocated to different Performing Agents (human or software) (Step 3)
- Map which other systems and other roles are involved in performing the control action (Step 4)

This process should preferably be done as an integrated part of developing and documenting the *Concept of Operations* (ConOps). It is therefore an advantage if the ConOps adopts the RBAT terminology and approach to modelling vessel missions using hierarchical goal structures, as explained in Steps 1 and 2 below.

The UoA's *context* (e.g., geography, environmental conditions, infrastructure etc.) is also expected to be described in the ConOps. In addition, the manning and operational philosophy should be outlined for all parts of the vessel's mission, including the use of supervisory control and fleet modes in case of abnormal situations on one or more vessels.

2.1.1 Step 1: Describe the vessel's mission (operational goals)

The first step of the process is to model the *mission* of the vessel or fleet of vessels. A mission can be described as consisting of three levels organized as a *hierarchical goal structure*, e.g.:

Mission: Safe and timely transport of cargo from Port X to Port Y

Mission phase: Arrival in port

Operation: Perform docking

Control function: Perform manoeuvring

Control action Y: Adjust speed

Control action Z: Adjust heading

The three levels can be explained as follows:

- The overall mission goal(s), i.e., the commercial, political (e.g., defence) or public intentions which have contributed to and justifies the vessel concept development and operation. A (simplified) example can be *"Ensure safe and timely transport of cargo from Port X to Port Y"*.

- The mission phases, i.e., subdivisions of the mission, are typically characterised by a recognisable shift in the location of the vessel, in terms of geographical surroundings, or the start and end of one or more operations. An example can be *"Arrival in port"*.

- The operations, i.e., activities performed as part of a mission phase to achieve the mission goal. An example can be *"Perform docking".*

The identified mission phases and operations are used to determine which functions to include in the risk assessment. Together with the details provided in the ConOps, they form the *operational context* (circumstances) under which the functions are required to perform. Considering the context is an important part of understanding the severity of potential accident scenarios (Step 9) and qualifying which mitigation measures can be considered effective for preventing losses from unsafe conditions (Step 13).

The generic RBAT mission model (Appendix A) can be used as a starting point. If needed, descriptions can be added and/or rephrased. It is recommended to check that abnormal situations and emergency responses are covered through the functions analysed. If not, they should be included as separate operations, as listed in Table 1 below.

Table 1 Overview of identified mission phases and operations.

Mission Phases	Operations
All phases	All operations
Port arrival	Port arrival general
	Perform port/harbour manoeuvring
	Perform docking/berthing
Port activities	Port activities general
	Perform unloading/loading
	Perform disembarkation/embarkation
	Manage passengers
	Re-plenish consumables
	Prepare vessel for voyage, incl. start-up
	Port stay, incl. shutdown
	Lay-up of vessel
Port departure	Port departure in general
	Perform port/harbour manoeuvring
	Perform undocking/un-berthing
Transit to location	Transit to location in general
	Navigate along coast
	Navigate on open ocean/deep sea
	Navigate on inland waterways
Abnormal situations and emergency responses	Perform damage control
	Respond to loss of stability/flooding
	Limit emission/spills to environment
	Mitigate fire/explosion
	Perform evacuation
	Assist emergency towing of own vessel
	Rescue man overboard
	Assist vessel in distress
	Handle blackout/loss of main power
	Handle loss of communication link
	Handle sabotage/piracy
	Respond to cyber attack
	Maintain ship safety in extreme weather
	Perform emergency repair at sea
	Abnormal situations not covered above (see Table 6)
Inspection, maintenance & repair	Inspection, maintenance & repair in general
	Perform planned maintenance
	Perform corrective maintenance
	Perform/support inspections
Waterborne operations	To be decided by User

2.1.2 Step 2: Describe the automated and/or remotely controlled functions (functional goals)

The second step of the process is to describe the functions that are subject to or affected by automation and remote control. This includes identifying:

- *control functions* required to successfully carry out the operations in each mission phase, and
- *control actions* allocated to various agents (human or software) involved in performing the control function.

Control functions and actions make up the *functional goals* of the hierarchical goal structure (letters in **bold**) in Figure 1 below:

Mission: Safe and timely transport of cargo from Port X to Port Y

Mission phase: Arrival in port

Operation: Perform docking

Control function: Perform manoeuvring

Control action Y: Adjust speed

Control action Z: Adjust heading

Figure 1 Use of Automation module in RBAT

The generic RBAT Function Tree (see Appendix B) can be used as a starting point for this process. For each operation described in Step 1, review and identify which of the (highest level) *functional goals* are required to achieve a successful outcome. Then, for each relevant functional goal, drill down the tree branches **to a sub-function level where automation can be made sense of, i.e., to a level where there is only one agent (human or software) responsible for performing the control action** (see Step 3).

Note that the generic RBAT Function Tree does not contain any detailed description of the functions and control actions, since the way the operational goals are achieved may vary from vessel to vessel. Thus, for each control function/ action selected for risk assessment, a short description should be added, which also includes other systems and roles are involved (see Step 4) in addition to the performing agent.

The lower-level functions in the RBAT Function Tree should primarily be considered as suggestions. Functions can be re-phrased and/or added on a need-to basis. For this the list of verbs provided in Appendix C can be useful.

When identifying and describing functions it is important to not only include those exerting direct control of a process. Care should be taken to also consider functions which serve more supportive purposes (often across several other functions), such as auxiliary and system monitoring functions.

Note that most functions are being used in more than one operation and in more than one mission phase. This has two significant implications. First, the severity of an unmitigated unsafe condition will typically vary with operations, depending on the context. Second, and similarly, the requirements put on a system are also likely to vary (e.g., traffic complexity may differ). Analysing all combinations of functions and operations in each mission phase will however result in an unnecessarily large analysis. I**n RBAT, it is therefore suggested to first identify generic unsafe conditions for each function subject to risk assessment and then identify the relevant phase and operation where this may lead to worst-case outcomes.** See section 2.2.2 for more details about this.

Functions which involve exchange and interaction with external agents or systems should also be considered for inclusion, such as those provided by surrounding infrastructures, e.g., navigational aids.

It is helpful if the ConOps or other relevant design documents[2] includes functional block diagrams, see Figure 2, illustrating the relationships and dependencies between the affected control actions (both internal and external).

Important: The level of function decomposition impacts the assessed risk level of the control actions. When the analysis is done on a (relatively) high level, the function adopts the risk level of its most critical functionality (i.e., sub-function). This is normally addressed in Step 18 (section 2.4.3) as part of allocating the risk level.

[2] E.g., safety and design philosophies, functional descriptions etc.

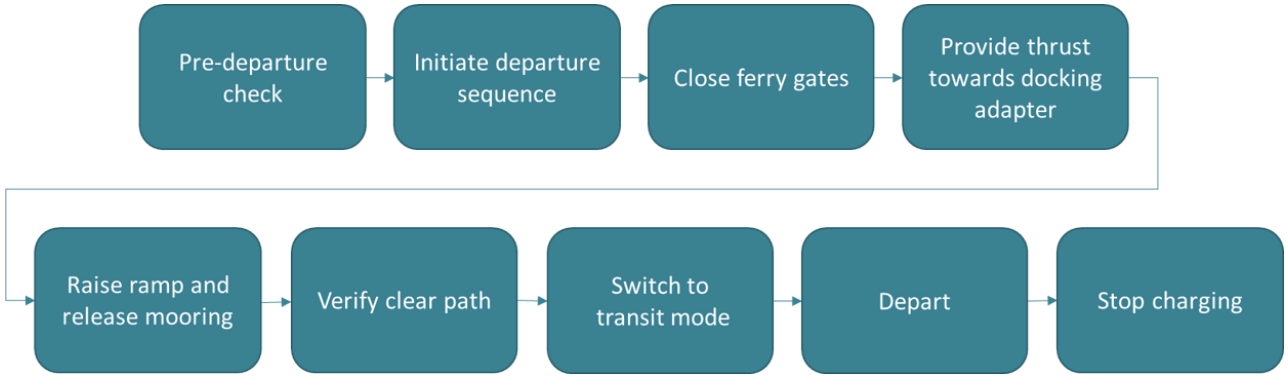


Figure 2 Example of control actions illustrated in a functional block diagram format.

2.1.3 Step 3: Describe how control functions are allocated to agents

The third step of the process is to describe how control functions are allocated to different *agents* by indicating who is responsible for performing the various required control actions.

Agents can be a computerised system (i.e., software) or a human operator, and only one agent can be listed as responsible for *performing* a control action under normal operations. However, depending on which level of detail control actions are described, cases may arise where more than one agent is involved. **In principle, this calls for further decomposing the control action until it can be distinguished which agent is the performing agent. If this appears to be too detailed, the agent making the decision should be nominated.**

It is recommended to create a list of performing agents that includes a brief description and comments as shown in Table 2 below.

Table 2 Description of a performing agent

Performing agents	Description	Comments
Situation awareness system	The situation awareness system manages and utilises information about the vessel's surroundings from AIS, ECDIS, GNSS, radar, lidar, IR, cameras, speed log, echo sound, gyro compass, microphone, thermometer, anemometer, and inertial measurement unit (IMU).	It is assumed that the situational awareness system is separate from the Autonomous Navigation System (ANS) in such a way that errors cannot propagate between the systems through memory, use of CPU time or shared I/O.

The comment in the table above reflect that the initial risk assessment may be performed during early concept evaluation, at a point when system vendors have not yet been selected.

2.1.4 Step 4: Identify other systems and roles involved

The fourth step of the process is to identify other systems and roles which are required to perform the control action, in addition to the performing agents. These are systems that, in case of failure, cause incorrect performance or unavailability of the intended control action. This step benefits from clear descriptions of the system architecture, including the relationships and interactions between various systems. Examples are system hierarchies, block diagrams, and system/function matrices. Such details and visualisation are expected to be included in the ConOps.

At a minimum, a table listing the control functions and actions, along with a description (Step 2) and the performing agent of each action/function (Step 3), as shown for one control action in Figure 3 below, should be prepared as input for the RBAT assessment.

Function hierarchy			Description and Performing Agents			
Control function (Level 1)	Control function (Level 2)	Control action (Level 3)	Description	Performing Agents	Other systems involved	Other roles involved
Perform manoeuvring	Provide acceleration/ deceleration	Adjust speed	This function controls the azimuth thrusters, it receives manoeuvring commands from ANS or fallback systems	Propulsion and Motion Control System	Autonomous Navigation System, Fallback Systems, Power Generation and Distribution System,	Backup system w/Joystick in ROC

Figure 3 Example of a functional breakdown with a description (step 2), performing agent (step 3) and other systems and roles involved (step 4) defined for one control action.

2.2 Part 2: Perform hazard analysis

An RBAT assessment can be performed at different levels of abstraction. When performing the hazard analysis this is done by first selecting the control function or action to be assessed. The selected control function/action should be presented with the following information as shown in Figure 4 below. The Hazard Analysis for the selected function/action consist of the following sub-parts: Identify unsafe conditions, analyse the operational aspects, and classify the severity. The purpose of each sub-part of the hazard analysis is to:

1 - Identify unsafe condition (Figure 5)
- Identify *unsafe conditions* associated with control actions (operations or functions) (Step 5)
- Identify *causal factors* which may initiate the unsafe conditions (Step 6)

2 - Analyse operational aspects (Figure 6)
- Describe relevant *operational restrictions* (Step 7)
- Determine *enabling conditions* and *exposure* to such circumstances (Step 8)
- Describe the *worst-case outcomes* from (unmitigated) unsafe conditions (Step 9)

3 - Severity classification (Figure 7)
- Rank the worst-case outcomes *severity* (Step 10)

Figure 5 – Figure 7 show one example of a hazard analysis (i.e. scenario) of the control function *Perform manoeuvring*, where the control action *Adjust speed* is performed by the *Propulsion and Motion Control System* (Figure 4). More examples and additional guidance are provided in the following sub-chapters. A “row” in the tables (Figure 4 – Figure 7), is referred to as a “scenario” describing how the causal factors triggered the unsafe condition, which in the presence of enabling conditions and in the absence of mitigation, caused the worst case outcome/ accident. Note that the *Operation (mission phase)*, which is defined in step 1, is first listed under the Operational Specific Analysis (step 7 and step 8) in Figure 6.

Function to be analysed			
Control function	Control action	Performing agent	Other systems involved (onboard, onshore)
Perform manoeuvring	Adjust speed	Propulsion and Motion Control System	Autonomous Navigation System, Fallback Systems, Power Generation and Distribution System

Figure 4 Selected function to be analysed, presented with an example

HAZARD ANALYSIS		
1 – Unsafe Conditions/Modes		
Guidewords	Unsafe condition	Causal factors
Not provided	Vessel fails to reduce speed	random hardware failure OR systematic failure OR systemic failure

Figure 5 Hazard analysis module in RBAT (Unsafe condition) with an example

HAZARD ANALYSIS					
2 – Operation Specific Analysis					
Operation (mission phase)	Enabling conditions	Exposure to enabling conditions	Operational restrictions	Worst Case outcome	Accident category
Perform port/harbour manoeuvring	Navigational: Onshore structure	High	Speed kept below 5 knots	Impact with dock in transit speed	Contact with shore object

Figure 6 Hazard Analysis module in RBAT (Operation specific analysis) with an example

HAZARD ANALYSIS					
3 – Severity Classification					
Effect on human safety	Effect on environment	Severity level safety & environ.	Effect on ship	Effect on uptime	Severity level ship & uptime
Single serious or multiple injuries	Temporary effect on a confined area	Significant	Minor	< 1 day delay	Minor

Figure 7 Hazard Analysis module in RBAT (Severity classification)

2.2.1 Step 5: Identify unsafe conditions associated with control actions/functions

The fifth step of the process is to identify unsafe conditions associated with the various control actions identified in Step 2. Unsafe conditions manifest as incidents where a system operates outside its operational envelope due to functional failures or exceeded capabilities and, which if left unmitigated, has the potential to cause an accident (i.e., losses).

Identification of unsafe conditions is done by assigning a guideword (see Table 3) found relevant and credible to the control action under consideration. What finally characterises a condition or mode as *unsafe* depends on the severity of worst-case outcomes (see Step 10).

To prevent relevant unsafe conditions being missed, it's recommended to create at least one scenario for each of the guidewords in Table 3 for each control function/action. **However, if an identified scenario is equivalent to an already assessed one, or has negligible impact on safety and environment, it may not require further evaluation.** See guidance in section 2.2.2.

It is also frequently necessary to create several scenarios associated with a single guideword. For example, there may be several different scenarios related to the guideword "too much" due to different degrees of "too much". In addition, the need to split scenarios may come from:

- Different causal factors, requires different forms of mitigation. See section 2.2.2 for guidance.
- The necessity of showing the risk picture for different mission phases and operations. See section 2.2.2 and 2.2.5 for guidance.
- The effect of operational restrictions being different for different type of worst-case outcomes (accidents). See section 2.2.5 for guidance.

Important: All the unsafe conditions are associated with the function being analysed, i.e., they represent functional failures.

A typical starting point for a hazard analysis is selecting the control function/ action and identifying the performing agent. For example, the Autonomous Navigation System (ANS) may be the *performing agent* for the control function "Provide Manoeuvring Commands", while the Thruster Control System could be the performing agent for "Perform Manoeuvring". When analysing the "Provide Manoeuvring Commands" function, it should be assumed that the "Perform Manoeuvring" function receiving the commands is working as intended. Similarly, all other related functions, including *essential common functions* such as power generation and distribution, should also be assumed to operate as intended.

Essential continuous functions must be identified and analysed separately to check for available mitigations and varying degrees of severity more thoroughly, across different mission phases and operations. If the risk level is still assessed as unacceptable, alternative approaches need to be considered (e.g., qualifying the control function as having sufficient integrity, and thus not having to rely on mitigation measures to ensure an acceptable level of safety).

Table 3 Unsafe condition/mode categories and guidewords

Categories	Guidewords
Not providing the control action leads to an unsafe condition	Not provided
Providing the control action leads to an unsafe condition	Provided when not required
	Incapable/not fit for purpose
Incorrectly provided control actions lead to an unsafe condition *Regarding too much/too little it should be considered that output provided by a function may be within expected range and provided timely but still be incorrect.*	Too early/late or in wrong of order
	Too much/ too little
	Stops too soon
	Applied too long
Control action not being followed leads to an unsafe condition	Not followed/Rejected

Note that the intention behind the guideword "Not followed/rejected" in Table 3 is not to look at scenarios where the function receiving commands has failed, but to look for scenarios where that function is working as intended but for different reasons may reject or ignore commands. Examples includes corruption in communication, and scenarios where the performing agent responsible for the function receiving commands must be in a specific state to accept them.

The category in Table 3 described as "Incorrectly provided control actions leads to an unsafe condition" refers to control parameters either being out of range, or within range but invalid or incorrect. The latter refers to unsafe conditions or modes caused by systematic or systemic failures which may be difficult to detect. A more detailed explanation regarding the implications of such failures is provided as part of Step 12 and 13.

2.2.2 Step 6: Identify causal factors which can trigger unsafe conditions/modes

The sixth step of the process is to identify causal factors which can trigger the unsafe condition. While unsafe conditions shall describe *why* the system is unsafe, the causal factors shall describe *how* the system became unsafe. These can be *internal failures* in the vessel or ROC systems or *insufficient capabilities* when it comes to handling external hazards (e.g., unfamiliar objects or strong currents). Hazards external to the vessel, relevant for the operation in question, should therefore always be considered when identifying failures which represent insufficient capabilities. These will in most cases be related to the conditions which enable potential accidents to occur (see Step 7).

The following categories have been defined to represent causal factors:

- Random (hardware) failures,
- Systematic failures,
- Systemic failures,
- Operator failures,
- Failures due to environmental conditions,
- Failures due to deliberate actions.

See Appendix D for a more detailed explanation of these categories.

Note that the risk associated with the system being incapable of handling external hazards (waves, winds, current, traffic etc.) is covered by the "systematic/systemic failures" category.

A second note is that the "operator failures" category only applies when a human is identified as the performing agent. Cases where the operator makes an error when preparing/configurating/maintaining the system is covered by "systematic failures".

Although the causal factor categories overlap and correlate[3] to some extent, somewhat depending on which function level they are applied to, they are useful as a guide to identify a wide range of failures that may pose risk. However, the purpose is not to try to identify very specific causes, but to identify mechanisms capable of detecting

[3] E.g., operator failure is a result of human errors caused by performance shaping factors (PSFs), such unannunciated failures with no alarms being presented.

and acting upon the type of unsafe conditions that may occur due to different types of causal factors. This is further elaborated below.

If a function is control-related, all unsafe conditions identified in Table 3 may be caused by any of the causal factor categories listed above. This leads to a total of 8 “unsafe condition types” x 6 “causal factor types” = 48 combinations of unsafe conditions and causal factor types per function. In addition, the potential for specific failures caused by the different causal factors is typically large, in particular when it comes to causal factors that are systematic in nature.

Guidance to avoid analysing a very large set of different scenarios per function is included below:

1) All guidewords to unsafe conditions identified in Table 3 should be reflected per control function included in the analysis to demonstrate that they have been considered. However, not all of them need to be elaborated into scenarios that are analysed in detail, see 2) and 3) below.

2) Some guidewords may represent scenarios which for a specific control action have no safety effect. For example, control action “Applied too long” may have no negative effect on safety for a specific function. In such cases, it is recommended to make a note saying that the scenario will not have any negative impact on safety or environment and that the scenario therefore does not need to be elaborated further.

3) For specific control actions, different guidewords may produce (near-)equivalent scenarios, i.e., the worst-case outcome (see 2.2.5), severity (see 2.2.6), operational restrictions (see 2.2.4), enabling conditions (see 2.2.3), relevant mitigations (see 2.3) are the same. For example, scenarios where the control action “Stops too soon” are in many cases equivalent to scenarios where the control action is “Not provided”. In such cases, it is recommended to make a note saying that the scenario is equivalent to an already analysed scenario, and that the scenario therefore does not need to be elaborated.

4) RBAT is designed in such a way that random hardware failures, systematic failures, and systemic failures, typically can be handled together within a single scenario as follows:

    a) For each guideword that leads to scenarios being elaborated, there will typically be a scenario where it is stated that the cause may be **random hardware failure OR systematic failure OR systemic failure.**

    b) Unsafe conditions associated with single random hardware failures are mainly managed through the function being made redundant (see 2.3.1 which discusses Fault Detection Isolation and Recovery (FDIR) mechanism within the performing agent).

    c) Unsafe conditions associated with systematic and systemic failures and residual risks associated with random hardware failure are typically managed through use of independent mitigation measures (see 2.3). If the performing agent is also redundant, this will contribute to reducing risks associated with double random hardware failures.

5) Operator failures will only be relevant for functions concerned with mission planning and for functions where the operator may change function input values during its operation. Thus, this cause is only relevant for a subset of the functions. Scenarios where the operator at the ROC fails to activate a mitigating measure as required is covered during evaluation of available mitigation measures (see 2.3). Thus, this type of operator failure is not considered a causal factor that could trigger the unsafe condition.

6) In the generic RBAT function tree there is a function called Manage Security that covers both physical and cyber security. Experience from testing of the method suggests that when RBAT is applied to high-level functions, unsafe conditions caused by deliberate actions related to unauthorised physical assessment, hacking and viruses may be covered when analysing this function. Thus, this does not need to be repeated for the other functions.

    a) Security threats will typically be managed by applying specific schemes applicable for the whole vessel and the ROC, e.g., cyber security class notations. Such schemes are expected to require specific risk analyses at a more detailed level than the more high-level function analysis performed using RBAT.

b) Other forms of deliberate actions, such as jamming, may be better dealt with when analysing specific functions. E.g., unsafe conditions associated with GNSS jamming and spoofing should be considered when analysing functions associated with observing the vessel's geographical location.

7) Unsafe conditions caused by some environmental conditions can also be covered when analysing specific functions such as: provide cooling, provide heating, provide fire protection, maintain watertight integrity etc.

See section 2.2.5 for further elaboration on combination of scenarios to be analysed for a specific control action.

2.2.3 Step 7: Determine Enabling Conditions and Exposure to such conditions

Due to the difficulties associated with determining the likelihood of unsafe conditions caused by systematic or systemic causal factors, the RBAT methodology does not make any attempt to determine the probability of the initiating event that could lead to an unsafe condition. However, it is possible to say something about exposure to enabling conditions and use that to determine the risk level in different phases of the operation.

As a principle, the accidents identified in Table 6 can only occur if an enabling condition or event is present. For example, collision with other ships can only occur if there are other ships relatively nearby, and groundings cannot occur when the water is deep. Thus, how long or frequent the vessel is exposed to enabling conditions influences the risk level associated with a potential unsafe condition (see Table 4).

Exposure towards enabling conditions may vary with each mission phases and its operations, and therefore, a specific mission phase shall be selected as relevant for the scenario. A mission phase or operation where the worst-case outcome (see step 9) is considered most likely to occur shall be selected in the scenario.

As explained in Step 6 and 9, it may be necessary to create several accident scenarios for the same unsafe condition to find out which scenario(s) have the highest residual risk when considering exposure to enabling conditions, operational restrictions, and the availability of mitigation measures. This means that Step 6 to 9 needs to be performed in an iterative way to determine what combinations of unsafe conditions, mission phases/operations, and accident types should be covered in specific scenarios.

Furthermore, the mitigation analysis of each scenario (Step 11 to 14) may later reveal the need for some scenarios to be split into two or more, as different mitigations measures are effective against different causes. Thereby the number of accident scenarios for a specific unsafe condition may further increase.

Table 4 below shows how exposure levels should be determined for a specific enabling condition or event in a specific operational phase. The scheme is adapted from a similar scheme used in the automotive industry, with reference being made to ISO 26262:2018 Road vehicles – Functional safety.

Table 4 Exposure levels/rates.

Levels	Frequency per mission phase	Duration per mission phase
Low	Occurs once every hundredth mission or less	<1 % of average operating time
Medium	Occurs once every tenth mission or less	1 % to 10 % of average operating time
High	Occurs more often than every tenth mission	>10 % of average operating time
Not relevant	The condition does not occur in the mission phase	

Table 5 below shows how relevant enabling conditions can be mapped against mission phases. Those considered relevant for the risk assessment should be assigned an exposure level per phase.

Section 2.4.1 contains examples of three different risk matrices that are used to determine risk level for scenarios where risk exposure is Low, Medium, and High, respectively.

Table 5 List of enabling conditions.

Category	Enabling conditions	Mission phase 1	Mission phase 2, etc.
Navigational	------NAVIGATIONAL------		
	Onshore structures (bridges, dock, pier, jetty)		
	Offshore structures (windmills, rigs, platforms)		
	Seabed structures/obstructions (subsea installations, pipelines, shipwrecks)		
	Shallow waters (reefs/ rocks, sandbanks, shore, beach)		
	Narrow waters/ shoreline		
	Ship traffic (large vessels)		
	Pleasure crafts with low maneuverability (sailboats, rowboats, canoes)		
	Pleasure crafts with high maneuverability (motorboats, jet skis etc.)		
	Vessels and crafts with restricted or lost maneuverability (e.g. engine failure).		
	Floating foreign objects (logs, barrels, containers, fishing equip., buoys, ice)		
	Unclear/ missing navigational marks		
	Swimmers, surfers		
Environmental	------ENVIRONMENTAL------		
	Strong currents		
	Strong winds		
	Reduced visibility		
	Large waves/ heave/ swell (w/o green sea on deck)		
	Green seas on deck		
	Heavy icing on vessel		
	Floating ice		
	Tsunami		
	Lightning		
	Floods (onshore)		
	Landslides (onshore)		
	Earthquake (onshore)		
Onboard	------ONBOARD------		
	High voltage/ electricity/ sparks		
	Flammable materials and liquids		
	High pressure (e.g. liquid/gas storage)		
	Chemically harmful/toxic substances (incl. exhaust/ emissions)		
	Biological hazards (virus, bacteria)		
	Passengers & crew conditions (heights, confined spaces etc.)		
	Cargo loads		
	Extreme temperature (incl. hot surfaces)		
	Radiation		
	Naked flames		
	Movement (e.g. rotating machinery)		
	Presence of humans (crew, passengers)		

2.2.4 Step 8: Describe operational restrictions

The eighth step of the process is to identify any operational restrictions associated with the control function and action being analysed. This can be maximum allowed speed limits, requirements for keeping distance to ship traffic and small crafts, prohibited sailing areas, weather condition and sea state sailing restrictions, and more.

Assumptions about operational restrictions are important because they can have an impact on the potential severity of worst-case outcomes, as well as influence which mitigation measures are available in different mission phases. This can in turn have a direct influence on the risk level and whether it is within acceptable limits.

Operational restrictions should not however be used to argue for excluding certain design features. For example, even though a ship is never meant to sail in specific weather conditions, it may still be designed to cope with such conditions (to a reasonable extent).

Furthermore, in case operational restrictions are not documented (e.g., in the ConOps), they should either be logged as an assumption to be validated at later stages or alternatively be proposed as a possible risk control measures which has not yet been taken credit for. See Step 19 regarding use of assumptions and actions.

2.2.5 Step 9: Describe the worst-case outcomes from unmitigated unsafe conditions

The ninth step of the process is to determine the worst foreseeable outcome of an unsafe condition in case there is no mitigation available (this includes Fault Detection, Isolation and Recovery, FDIR, see Step 11). In RBAT, worst-case outcomes assume the contextual presence of a credible *hazard* (i.e., an enabling condition or event, see Step 7). For example, loss of steering (an unsafe condition) close to shore (a hazard) results in a grounding (a worst-case outcome).

Worst-case outcomes should be adjusted taking operational restrictions into account as shown Figure 8 below. The severity of worst-case outcomes when considering operational restrictions will be used when assessing the risk level, see section 2.4.1.

Operation Specific Analysis					
Operation (mission phase/)	Enabling conditions	Exposure level	Operational restrictions	Worst Case outcome	Accident category
Perform port/harbour manoeuvring	Ship traffic	High	Speed kept below 5 knots	Losing control leads to collision with other ship	Collision with other ship
Perform port/harbour manoeuvring	Pleasure crafts with low manoeuvrability (sailboats, rowboats)	Medium	Speed kept below 5 knots	Losing control leads to collision with smaller vessel/craft	Collision with small craft/leisure vessel

Figure 8 Example of the operation specific part of the RBAT process: describing the worst-case outcomes taking operational restrictions into account.

Note however, that the exposure is considered lower for the latter scenario. Thus, the risk matrix for *Medium exposure* will be applied when determining the residual risk for this scenario in contrast to the ship collision where the risk matrix for *High exposure* will be used. See section 2.4.1.

In case an argument is made that a hazard is not present, e.g., through operational restrictions, this must be clearly stated either as part of the prevention analysis (Step 15) or in the comments for addressing risk control (Step 19).

Finally, an accident main category is assigned to each worst-case outcome, using the taxonomy in the list below (Table 6). This is done by matching the worst-case outcome against the accident main category which includes the most suitable accident sub-categories.

Table 6 Accident main and sub-categories

General

- No effect on safety
- Injuries/loss of life (general)

Degraded/Loss of control

- Degraded/Loss of directional control
- Degraded/Loss of propulsion power
- Degraded/Loss of electrical power
- Degraded/Loss of communication link
- Degraded/Loss of containment
- Degraded/Loss of stability
- Degraded/Loss of control (other)

Collision

- Collision with other ship
- Collision with multiple ships
- Collision with small craft/leisure vessel
- Collision with canoe, kayak, paddleboard etc.

Contact

- Contact with floating object
- Contact with flying object
- Contact with shore object

Damage to/ loss of ship equipment

Hull failure (i.e., structural failure)

Leakage

- Leakage of hydrocarbons
- Leakage of chemicals
- Leakage of hazardous substance (other)

Fire/explosion

- Fire
- Explosion

Grounding/stranding

- Grounding
- Stranding

Capsize/listing

- Capsize
- Listing

Flooding/foundering

- Massive flooding
- Progressive flooding
- Foundering

Non-accidental event

- Acts of war
- Criminal acts
- Illegal discharge
- Other

Missing vessel

The accident categories are mutually exclusive and only one shall be assigned to each worst-case outcome. To help with this, the following principles apply:

- *Injuries/loss of life* shall only be used when this happens outside any of the other accident categories. For example, in the case of the crew being exposed to a disease.
- *Loss of control* shall only be used when it is not credible that the unsafe condition can evolve into any one of the other accident categories.
- *Damage to/ loss of ship equipment* shall only be used when this occurs in absence of the other accident categories.
- *Hull failure* (i.e., structural failure) shall only be used in case this occurs without being the direct cause of other accident categories (e.g., capsize or foundering).

More than one worst-case outcome may need to be considered for each control action. The background is described through an example below.

For a vessel which does not have passengers onboard and does not carry dangerous cargo, the collision scenarios will typically represent worst-case outcomes, since they may lead to fatalities onboard other ships, small boats, kayaks etc.

- Without considering operational restrictions, collision with another ship will typically represent a worst-case outcome since the other ship may sink with many people onboard. However, for this scenario the risk may be significantly reduced by introducing operational restrictions in form of speed limits that may lower the severity of a collision in mission phases where exposure to ship traffic is high, e.g., during in shore operations. Depending on the type of unsafe condition that could lead to collision, there may also be a range of mitigation measures that can be used to manage the unsafe condition.

Collision with canoe, kayak, paddleboard etc. may affect fewer people, and exposure to such crafts may be Medium or Low even during inshore operations, see Table 4 regarding exposure rates. However, a collision may lead to one or more fatalities even at low speed and there may be fewer mitigation measures available as these

crafts may not appear on radar, meaning no collision alarm would be triggered based on radar information. Consequently, after considering all relevant factors, the residual risk may be higher here than residual risk associated with ship collision, even if the absolute worst-case outcome is less severe and the exposure to kayaks and similar crafts is lower than exposure to ship traffic. Thus, there may be a need for additional risk reducing measures for this scenario, further highlighting the importance of systematically addressing the different enabling conditions.

Furthermore, it is recommended to select worst-case outcomes for different mission phases, as a range of factors used in determination of residual risk may vary with operational phases. Examples include use of speed limits, exposure to ship traffic and leisure crafts, whether there is enough time for anchoring, whether station keeping is an effective MRC, etc.

There will always be several scenarios to analyse for each control action, and experiences from testing RBAT shows that available operational limitations and mitigations are often repeated. This can be utilized to select different worst-case outcomes and different operational phases for the different combinations of guidewords and causal factors that are discussed in 2.2.2. Thereby the number of scenarios to analyse per control action can be reduced.

2.2.6 Step 10: Rank the worst-case outcome severity

The tenth step of the process is to rank the worst-case outcome severity. For impact on safety and the external environment, this is done by assigning a degree of severity using the index in Table 7 and Table 8.

When it comes to the indexes for asset damages and delays and downtime (Table 9 and Table 10), each company can adjust the scales and add specific monetary values for each level to calibrate what they consider to be acceptable losses. The limits for what define acceptable levels of risk is presented in the risk matrix shown as part of Step 18 (section 2.4.1).

Table 7 Severity index for worst-case outcomes in terms of peoples' safety

Severity	Effects on human safety
Negligible	Single minor injury
Minor	Single injury or multiple minor injures
Significant	Single serious or multiple injuries
Severe	Single fatality or multiple serious injuries
Catastrophic	Multiple fatalities (more than one)

Table 8 Severity index for worst-case outcomes in terms of environmental impact

Severity	Effects on environment
Negligible	Spills onboard vessel or emissions with no noticeable effect on the environment
Minor	Spills or emissions with a brief effect on the environment surrounding the vessel
Significant	Spills or emissions with a temporary effect on the environment limited to a confined area
Severe	Spills or emissions with a long-lasting effect on the environment reaching some distant areas
Catastrophic	Spills or emissions with a permanent effect on the environment reaching a widespread distant area

Table 9 Severity index for worst-case outcomes in terms of damage to ship

Severity	Effects on ship[4]
Negligible	Superficial damage
Minor	Local equipment damage
Significant	Non-severe ship damage
Severe	Severe ship damage
Catastrophic	Loss of ship

Table 10 Severity index[5] for worst-case outcomes in terms of delays and downtime

Severity	Effects uptime[6]
Negligible	< 2 hours delay
Minor	< 1 day delay
Significant	1 – 10 days downtime
Severe	10 – 60 days downtime
Catastrophic	> 60 days downtime

The severity of worst-case outcomes when considering operational restrictions will be used when assessing the risk level, see section 2.4.1. Figure 9 below builds on the operation specific analysis from Figure 8 and shows the severity classification of the two examples:

- The first row reflects a ship collision scenario where the severity of a potential collision is reduced to a level where fatalities are no longer expected, through a speed limit.
- The second row reflects a scenario where the same speed limit is applied, however, the severity is not reduced, since a collision with small crafts may still lead to multiple fatalities.

Severity Classification		
Effect on human safety	Effect on environment	Severity Level HSE
Single serious or multiple injuries	Temporary effect on a confined area	Significant
Multiple fatalities (more than one)	No effect	Catastrophic

Figure 9 Example where Severity is adjusted based on an operational restriction

[4] Here “ship” also extends to include assets required for remote control, such as remote-control centres and other infrastructure (if relevant).
[5] Scale is adopted from DNV-RP-203 Technology Qualification (DNV, 2021b).
[6] Uptime is a measure of system reliability, expressed as the percentage of time a machine, typically a computer, has been working and available. Uptime is the opposite of downtime (source: https://en.wikipedia.org/wiki/Uptime).

2.3 Part 3: Perform mitigation analysis

The purpose of the mitigation analysis is to identify mechanisms that can prevent unsafe conditions from escalating into accidents.

The analysis consists of the following steps:

- Check whether Fault Detection, Isolation and Recovery (FDIR) is planned to be part of control functions' design (Step 11)
- Identify which mitigation measures are in place to prevent the unsafe condition or mode from resulting in losses (Step 12).
- Assess and determine whether mitigation measures can be qualified as effective in achieving their intended purpose (Step 13).
- Rank how effective the mitigations are at preventing potential losses (Step 14).
- Identify measures which are in place to prevent the direct cause of an unsafe condition or mode from occurring (Step 15, optional)

Mitigation measures may involve i) using alternative means of control re-entering the operational envelope (albeit in a potentially degraded state), or ii) entering a fallback state as a way to stay as safe as possible while attempting to regain the desired level of control. Fallback states are operational states to which the system (vessel) should transition to when experiencing an abnormal situation which make it impossible to stay within the operational envelope. Entering a fallback state can be achieved by use of mitigation measures realised by a single function or several different functions. The same or additional functions may also be responsible for recovering the system to a normal or degraded (but safe) condition. However, note that all degraded states are not necessarily fallback states. A situation where an autonomous ship must be manually controlled remotely does not necessary qualify as a fallback state (but could be regarded a degraded state).

Summarised, in this context mitigations can involve the following types of responses:

- Withstanding or recovering from a failure before it turns into an unsafe condition
- Re-entering to a normal operational envelope by regaining control of an unsafe condition
- Enter some form of fallback state to prevent escalation and reduce the likelihood of further losses.

The role of mitigation measures is illustrated in the RBAT accident model (see Figure 15 **Error! Reference source not found.**in Appendix E).

The Mitigation analysis is illustrated Figure 10 below, where an example of the FDIR and three mitigation measures are listed.

MITIGATION ANALYSIS									
Fault detection, isolation & recovery (FDIR)	1st mitigation measure	Operational state if successful outcome	2nd mitigation	Operational state if successful outcome	3rd mitigation	Operational state if successful outcome	4th mitigation	Operational state if successful outcome	Mitigation Effectiveness
The format of a voyage plan shall be such that the ANS will be able to recognise an invalid plan and imitate station keeping.	ANS initiating evasive manoeuvre or station keeping based on input from Grounding and Collision Avoidance System	Autonomous control	ROC operator initiating station keeping or evasive manoeuvre based on information in the camera feed and/or from other relevant sources such as radar picture, ECDIS picture etc.	Station keeping	In case of grounding alarm from ECDIS, the ROC operator shall initiate either station keeping or evasive manoeuvre based on operational judgement If the ROC operator does not react to the alarm within a specific time span, the ANS shall initiate station keeping	Station keeping			High

Figure 10 Mitigation analysis module in RBAT.

2.3.1 Step 11: Check for Fault Detection, Isolation and Recovery (FDIR)

The eleventh step is checking whether Fault Detection, Isolation and Recovery (FDIR[7]) is part of the control functions' design and can (for the assessed scenario) prevent losses when the unsafe condition is caused by random hardware failures, and some (but not all) other types of failure causes.

A binary assessment of FDIR is part of the input used to rank mitigation effectiveness (see Step 16) – "Yes" if FDIR is planned for and "No" if not.

- If "Yes", this should be based on what is documented in technical reports (e.g., ConOps or a Safety Philosophy).
- If the use of FDIR is not documented anywhere, but the risk analysts are quite sure that relevant FDIR mechanisms will be implemented, it is possible to state "Yes" and record an assumption that is subsequently validated in the project. See section 2.5 for details.
- If the assessment is "No", an action to implement FDIR as part of the design can be noted down as a potential risk control measure, if relevant (see Step 19).

When doing the assessment, it is important to be aware of typical challenges associated with FDIR mechanisms that are implemented in the performing agent responsible for the control action being analysed:

a) Built-in FDIR mechanisms may be vulnerable to common cause related problems, e.g., a weakness in the software may lead to an unsafe condition and at the same time inhibit functionality needed for detection and/or recovery. Some examples of such scenarios are included below.
    - Logic intended for handling of a specific possible failure situation may as a side-effect disable one or more FDIR mechanisms implemented in another part of the software. Such negative influence may occur due to dependencies in the software's internal dataflow that has not been identified and therefore not explored in verification and validation activities.
    - A memory overwrite may occur e.g., when specific input combinations and/or input sequences is received as part of a software which is not robust with that input. If a memory overwrite should occur, this could negatively affect other parts of the software using the same part of the memory which is overwritten. Memory-overwrite often leads to a software failure which in some operational scenarios may be mitigated through the use of a hardware watchdog automatically initiating a reboot. However, memory overwrite may also have more subtle effects which may be harder to detect and mitigate.
    - Specific parts of software may under certain input conditions use too much processing time and thereby slow down or inhibit FDIR mechanisms in other parts of the software. This is particularly relevant in software applications utilising multitasking, but the problem may also occur in single task applications, e.g., the software execution may stay too long in an internal loop.

b) Some types of unannunciated failures may only be detected at higher levels in the system that have a broader overview of the system state and the current operational mode, for example by comparing output from different controllers in functionally diverse subsystems, comparing measurement from physical processes with expected performance, or through operator observation of system behaviour.
    - Systemic failures caused by missing or inadequate system requirements are examples of failures that may be difficult to detect through FDIR mechanisms built into the performing agent.
    - Note that unannunciated failures may also be a challenge for some software supervisors that are considered independent of the performing agent, see section 2.3.3 regarding functionality in mitigating measures.

These challenges are the reason why FDIR mechanisms built into the performing agent being analysed are considered to provide only a moderate level of risk mitigation.

Regarding the common cause challenges described in a) above, it should be noted that it may be possible to decompose functions into subfunctions and analyse these subfunctions and control actions at a more detailed level. This may typically lead to identification of a hierarchy of more low-level performing agents supporting the top-level performing agents and control actions. If the lower level performing agents are located at different physical

[7] Wikipedia includes a useful article about FDIR, see https://en.wikipedia.org/wiki/Fault_detection_and_isolation

controllers, these performing agents may potentially act as supervisors for each other. In such an architecture the top-level performing agent may also be located at a separate controller and act as a supervisor for all lower level performing agents. Thus, through decomposition of a high-level function, more detailed independent mitigating measures may be identified. Such a distributed system architecture may reduce the number of FDIR mechanisms considered vulnerable to common cause. However, it may not remove the problem completely.

A highly integrated system architecture where several performing agents are sharing hardware and resources like memory and processor time will in principle be more vulnerable to common cause issues than a physically distributed one. Note however that there are controllers certified for usage in highly safety critical systems in other industries that can provide so-called time and space partitioning, sometimes also referred to as logical separation. Such controllers allow tasks of different criticality to be executed on the same hardware as unwanted interference through timing, memory, or I/O is prevented by the certified controller hardware in combination with the certified commercial off the shelf software provided by the controller vendor.

2.3.2 Step 12: Nominate mitigation measures which can prevent losses

The twelfth step of the process is to identify which mitigation measures are in place to prevent the unsafe condition from resulting in an accident (and losses). This is done by nominating potential 1st, 2nd, 3rd, and 4th mitigation measure(s) for each combination of unsafe condition and causal factor(s) (see one example in Figure 10). Preferably, a preliminary set of mitigation measures have already been described *prior* to using RBAT, e.g., as part of drafting the first version of a ConOps. If new mitigation measures are identified as part of the process, these should be added to the list of existing ones, and then nominated in the analysis.

It is crucial to consider that when assessing the need for a 2nd mitigation measure, it should be based on the assumption that the 1st mitigation measure was ineffective in addressing the unsafe condition effects, rather than assuming it has simply “run out”. Taking the latter approach may lead to numerous scenarios and introduce a level of uncertainty. As such, the 2nd (and any subsequent) mitigation measure must be able to respond to the initiating event, and not to a scenario where the 1st mitigation measure was successfully initiated, before eventually failing. For example, picture a scenario where the initiating event is a drive-off and that the 1st mitigation measure is to bypass the DP system by taking manual control of the thrusters. If this fails, it must be assumed that the drive-off is still occurring and the 2nd mitigation measure must cope with this.

2.3.2.1 Information required per mitigation

The information below shall be made available per mitigation in a mitigation measure register. Such a register should be kept in a table or a database. It is recommended to have one column or database field per entry, to allow consistency checks. For example, one may want to filter all mitigating measures associated with a special supervisory agent, or a specific resulting operational state.

Table 11 Register of mitigation measures

No	Information	Comments
1	ID/Name	As it may be relevant to refer specific mitigation measures from various documents it is recommended that a mitigation measure should have a unique identifier containing a number and a name. Further, it is recommended that: ■ The combination of name and number is unique for the overall concept being analysed. ■ The numbering is unique per concept or per supervisory agent responsible for activating the mitigation. Since the same mitigation measure may be used for several functions, it is not recommended to use numbering per function.

No	Information	Comments
2	Short description	The description shall identify: ■ How the unsafe condition is detected and by which supervisory agent, see 0 below. ■ What recovery action is to be performed, and which supervisory agent is responsible for deciding whether it shall be activated or not.
3	Supervisory Agent responsible for the function	The supervisory agent that is responsible for deciding whether it shall be activated or not, see also 2.3.2.2 below.
4	Supervisory control category	One of: Active Human Supervision, Passive Supervision, Software Supervision or No Supervision, see also 2.3.2.2 below.
5	Operational State	Operational state after mitigation measure completed, see 2.3.2.4.
6	Applicability of the mitigation measure	
6.1	For which mission phases the mitigation measure is applicable	See section 2.1.1 regarding mission phases.
6.2	For which mission phases the mitigation measure is NOT applicable	e.g. due to being ■ Being potentially unsafe ■ Restricting use of other mitigation measures ■ Not being relevant (i.e., effective) See section 2.1.1 regarding mission phases.
7	System and human involvement in the mitigation measure	
7.1	Other systems which must function and be available for execution of the mitigation measure	Reference to relevant performing agents outside the function being analysed which need to work as expected for the mitigating measure to be effective. See section 2.1.3 regarding performing agent
7.2	How humans are involved in executing the mitigation measure):	How humans are involved in information acquisition and analysis, decision making, and implementation of actions. See section 2.3.3.4 for guidance regarding human involvement.
8	Limitations to the mitigation measure	
8.1	External/environmental limitations to the mitigation measure	For example, limitations related to: Sea state, visibility, day/night, suitability for anchoring, or availability of external resources
8.2	Resource limitations in the mitigation measure	For example, time, fuel, energy reserves, manpower, etc.
8.3	Limitations in the sequence mitigation measures can be introduced	e.g., a mitigation measure should only be activated after another has been exhausted

2.3.2.2 Identify supervisory control agents for each mitigating measure

Supervisory control is a role with an explicit responsibility to monitor control action performance and detect unsafe conditions so that the desired outcome can be achieved through implementation of corrective responses. Examples of unsafe conditions can be system failures and malfunction, or external conditions which exceed pre-defined criteria for what are considered operational limits (e.g., weather conditions). In case the agent performing the control-action does not have the capacity to withstand or self-recover from a failure, the designated supervisory agent is responsible for ensuring that mitigation measures are effective, as described in Steps 13 and 14.

An important principle is that the supervisory agent cannot be the same as the agent performing the control action(s) being supervised. See section 2.3.3. for an overview of limitations to the agents involved in FDIR and mitigating measure (Table 17).

Supervisory control can be performed by either a software or human agent. It is important to consider the strengths and weaknesses of both agents before assigning supervision responsibilities. In cases where humans are the supervising agent of a control action they will often rely on a system for monitoring and detection, while analysis,

decision-making and implementation of actions require cognitive efforts and manual actions. A software agent will perform all actions.

Four different categories of supervisory control are defined in RBAT:

- *Active human supervisory control*: A human agent is responsible for continuously[8] monitoring the automated performance of a control action with the purpose of being able to successfully intervene at any stage based on judgements about how to best act upon the situation. Because active supervision provides an opportunity for the human agent to continuously create situational awareness, it can be beneficial in cases where there is limited time available to intervene.
- *Passive human supervisory control*: A human agent is responsible for being available[9] to monitor the automated performance of a control action and successfully intervene upon requests (e.g., an alarm) generated by the system according to pre-defined parameters. Because passive supervision (often) requires the human agent to obtain situational awareness about the events preceding the request, it is best suited for cases where there is sufficient time available to intervene.
- *Software supervisory control*: A software agent is responsible for continuously monitoring the performance of a control action with the purpose of being able to successfully intervene on demand, without involvement of a human agent, for example if pre-defined parameters are exceeded, or if there is disagreement in voting between separate functions/components.
- *No supervisory control*: No agent is responsible for monitoring the performance of a control action.

One of these categories shall be selected for each mitigation measure.

As such, the supervisor is the agent responsible for making decisions about interventions. Note that for some of the mitigating measures it will be a software supervisor that detects the unsafe condition and raises an alarm, while the decision-making is performed by a human. In these cases, Passive human supervisory control shall be selected since it is sufficient for this particular mitigation which relies on a software supervisor for its detection part.

It is important to emphasize that the supervisory control categories represent a specific operational responsibility. This means that if an operator is responsible for actively supervising a control action, this must be reflected in job descriptions, procedures, routines, etc. Selection of supervisory control categories should therefore be based on the overall philosophy about monitoring and control described in the ConOps, which also includes a more detailed description of the supervisory roles. Such descriptions should consider the influence from factors such as fleet size, manning level, competencies, human-software interfaces (e.g., information representation) when assigning supervision responsibilities to human agents. A preliminary solution for supervisory control should therefore be decided upon and described in the ConOps before commencing with the hazard and mitigation analysis (Part 2 and 3). The hazard analysis may however provide insights which can call for the initial supervisory agent and type of control to be revised. For example, different unsafe conditions associated with a specific function may require different supervisory agents as specific software supervisory agents may not be capable of detecting all unsafe conditions.

2.3.2.3 Detection of unsafe conditions

Unsafe conditions may not manifest themselves as detectable anomalies, e.g., in case they are a result of control parameters being within range of incorrectly defined parameters. This may cause a scenario where no control signal is sent that demands automatic activation of mitigation measures and/or the operators are unable to intervene due to unannunciated failures. As such, it is important to systematically check for these types of unsafe conditions and how they impact the availability and qualification of mitigation measures (see Step 13), due to how this is determined by which supervisory control is required for successful detection.

Table 12 below lists the detection methods that typically will be available to supervisors.

Note that all mitigation measures have to utilise one of the methods numbered 3, 4 or 5, as these are capable of identifying unsafe conditions even if there are no alarms from the performing agent.

[8] 'Continuously' implies that the agent is responsible for, and expected to, direct his/her/its attention to a function for as long as it is being executed.
[9] 'Available' implies that the agent is responsible for, and expected to, be in close enough proximity to intervene upon a demand from the system.

Table 12 Detection methods typically available to supervisors

No.	Detection of unsafe condition	Comments and Examples
1	Alarms from the performing agents	In RBAT, measures triggered by such alarms are considered part of the FDIR within the performing agent, and thereby not independent mitigation measures.
2	Alarms from other supervisors	For example, a human supervisor may receive and act upon an alarm from a software supervisor that only has the authority to detect and report the problem. If the alarm shall be able to trigger an independent mitigation measure, the other supervisor raising the alarm must be capable of detecting unsafe conditions even if there are no alarms from the performing agent, see 3,4 and 5 below.
3	Detection through analysis of the data received from the performing agent	Examples of unsafe conditions that may be detected are: no data received, corrupted data received, data received too late, invalid data received, data received out of sequence, unexpected trend in in the data received etc. This detection method requires that some form of software supervisor is involved, and reflects that different performing agents typically are acting as supervisors for each other when exchanging data. Note that this detection method may not detect all forms of unsafe conditions that could occur. For example, a thruster control system may be able to detect a range of problems with the data received from an Autonomous Navigation system. However, data may be received on time, not corrupted, within expected range and sequence, but still wrong or not fit for purpose. In such a case independent observation of the process being controlled may be needed to detect the problem, see no. 5 below.
4	Detection by comparison of data from functions having different performing agents	An agent acting as a supervisor for several subfunctions performed by different performing agents, may be able to compare data received from these subfunctions and conclude that there is a problem in one of them, even if there is no alarm. In a high-level analysis where the function is not decomposed into subfunctions this method may be considered part of FDIR. However, a more detailed analysis may identify that the performing agent for the overall function, acts as an independent supervisor for specific failure modes in the subfunctions. To qualify a mitigation measure that utilizes this detection method, the subfunctions providing the data need to be sufficiently independent, see section 2.3.3.2 for more guidance related to independence. Depending on what data is compared, this method may also be used for independent observation of the performance of the process being controlled.
5	Independent observation of the performance of the process being controlled	Examples: A Human supervisor may compare input from different information sources, like electronic charts, camera, radar etc. and detect that there is an unsafe condition even if there is no alarm. This requires active supervision, see section 2.3.2.2. An Autonomous Navigation Function (ANF) may detect that there is a problem in the Perform Manoeuvring function by comparing input from the situational awareness system with expected performance and conclude that there is a problem. If no alarms are present the navigation function may, in such a case, not be able to identify the detailed cause of the problem, and a human supervisor may need to be involved in the decision making. Note that a mitigation measure utilising the latter method for detection, would only be considered independent for unsafe conditions associated with functions where ANF is not the performing agent, for example problems relate to thruster control, power management etc. It may also be able to detect unsafe conditions causing unwanted output from the ANF itself. However, such a measure may not be independent of

No.	Detection of unsafe condition	Comments and Examples
		the ANF and therefore it may be considered a part of FDIR, see section 2.3.3.2 for more guidance related to independence.

By looping through the five methods above for each function, it is possible to build up a preliminary table that shows which supervisors are relevant for each function, before doing the risk analysis. This is recommended to do as a part of establishing the ConOps.

Further consideration related to efficiency of mitigation measures depending on software supervisors is included below.

Regarding detection method 4 and 5 in Table 12 above: A software supervisor that is capable of detecting and mitigating critical effects of all possible failure causes in a specific performing agent, may need to be equally as advanced as the performing agent itself and also be functionally diverse. It can also rely on another performing agent that is equally as advanced and functionally diverse. This is to be able to detect and act upon output that is within expected range and timing but is still wrong. A typical example where the latter strategy is used, is for position reference systems where outputs from positioning systems utilizing different principles are compared to each other. For example, output from GPS may be compared to output from Inertial Navigation Systems (INS) and other position reference sources. Consequently, critical failures in one of the position reference systems can be detected and handled regardless of failure cause. See also discussion about functional diversity in section 2.3.3.2 which is providing guidance related to independence.

In some cases, a relatively simple software supervisor can detect and mitigate critical effects of all possible failure causes in the performing agent. A typical example is fully automated Emergency Shutdown Systems (ESD systems). Such systems are not monitoring output from the performing agent directly. Instead, failures in the process control system are detected indirectly through the ESD system monitoring the status of the process being controlled while using its own sensors. If critical parameter limits are exceeded, the ESD system will shut down the process being controlled. This is an example of detection method 5 in Table 12 above.

The ANF related example used to illustrate detection method 5 in in Table 12 above, shows that some software supervisors may also have very strong capabilities when it comes to detecting when a problem is present, but less capability for detecting the cause and independent mitigation. Regardless of what caused the unsafe condition, and even if there are no alarms, a supervisor in an autonomous system may through monitoring the ships motion be able to detect critical problems in one or more of the other performing agents involved in the manoeuvring function. This resembles ESD systems in that failures are detected indirectly through monitoring of the process being controlled.

The supervisor may also try to identify and isolate the cause of the unsafe conditions based on trend analyses or similar, for example it may decide to exclude a thruster from being used based on available statistics. It may also initiate a “station keeping” command as an attempt at bringing the vessel to safe state. However, such measures may rely on the same performing agents that may have failed, and consequently such mitigations may only be considered effective for specific failure causes.

Such a supervisor may also have the authority to cut power to the thrusters as a subsequent option if other measures are not effective. In that case the mitigation measure would be independent of the performing agents having failed. However, whether such a measure would lead to a safe state will be highly dependent on the type of operation and operational phase.

Note that a further decomposition of the functions may have led to more software supervisory control agents being identified.

2.3.2.4 Identify relevant operational states after the mitigating measure have been applied

The use of fallback states is an important way to manage unsafe conditions. Thus, such states should be identified in the ConOps and in RBAT. To complete the picture, the operational states that are relevant when there are no unsafe conditions should also be identified.

As part of the preparations for an RBAT assessment the following information below shall be made available for each operational state in an Operational state register. Such a register should be kept in a table or a database. It is recommended to have one column or database field per entry, to allow consistency checks. For example, one may want to filter all operational state that are considered fallback states.

Table 13 Register of Operational states

No	Information	Comments
1	ID/Name	An Operational state should have a unique identifier containing a number and a name. The name shall reflect the operational state the system (vessel) has transitioned to.
2	Short description	The description shall identify: ■ which performing agent is involved and how ■ action taken by the system (vessel) in the operational state
3	Applicability of the Operational State	
3.1	Is the operational state a fallback state?	Is the operational state a (designed) state the vessel can go to when it is outside the operational envelope? Reference to the fallback chain, as described in section 2.3.3.3.
3.2	For which mission phases is the Operational State applicable	Reference to the mission phases where the Operational state can be reached. See section 2.1.1 regarding mission phases.
3.3	For which mission phases is the Operational State NOT applicable	Reference to the mission phases where the Operational state cannot be reached. See section 2.1.1 regarding mission phases.
3.4	Events to which the Operational State is a planned response	
4	System and human involvement in the operational state	
4.1	Systems which must function and be available for executing the operational state	Reference to relevant systems which need to work as expected for the operational state to be reached.
4.2	How humans are involved in executing the operational state	How humans are involved in in executing the operational state (information acquisition and analysis, decision making, and implementation of actions). See section 2.3.3.4 for guidance regarding human involvement.
5	Limitations to the operational state	
5.1	External/environmental limitations to the mitigation measure	For example, limitations related to: Sea state, visibility, day/night, suitability for anchoring, or availability of external resources
5.2	Resource limitations in the mitigation measure	For example, time, fuel, energy reserves, manpower, etc.
5.3	Limitations in the sequence operational state can be introduced	

The list of states is Table 14 below is a theoretical example inspired by the fallback chain typically found onboard manned DP vessels.

Table 14 Example of Operational states

ID	Operational state	Short description
OpState-0	Not Relevant	Used in scenarios where the vessels overall operational state has little relevance.
OpState-1	Moored	Vessel is moored at quay.
OpState-2	Autonomous control	Vessel is controlled by the Autonomous Navigation System
OpState-3	Autopilot control	Vessel is controlled by the Autonomous Navigation System, but settings related to speed and heading is decided by the operator at the ROC. This state may be relevant in scenarios where there is a technical problem with the collision and grounding system and in scenarios where the traffic situation suggest that manual navigation is preferable.
OpState-4	Drifting after thrusters being set in idle	This is a typical intermittent state that may be relevant when managing specific unsafe conditions. Transition to this state may be imitated automatically by the Autonomous Navigation Function, by the perform manoeuvring function and by the ROC operator. In all three cases the ROC operator will typically be expected to take further action within a short time limit.
OpState-5	Station keeping	Vessel is controlled by the Autonomous Navigation System, and kept in a stationary position
OpState-6	Joystick control (ANS independent)	Vessel is controlled by the ROC operator using a joystick. The ROC system is communicating with a dedicated backup controller onboard that seen from thruster control has a higher priority than the Autonomous Navigation System
OpState-7	Operator controlling thrusters individually	Vessel is controlled by the ROC operator using induvial levers per thrusters. The ROC system is communicating with a dedicated backup controller onboard that seen from thruster control has a higher priority than both the Autonomous Navigation System and the backup controller used for joystick control
OpState-8	Safely drifting after shutdown while preparing for towing or emergency anchoring	Vessel is drifting after ROC has initiated an Emergency Shutdown of thruster drives using a dedicated last resort communication channel
OpState-9	Anchored	Vessel is anchored while waiting for assistance
OpState-10	Being towed	Vessel is being towed

2.3.3 Step 13: Qualify the nominated mitigation measures

The thirteenth step of the process is to assess and qualify the nominated mitigation measures against a set of performance criteria which characterises them as effective in accident prevention. This includes:

Functionality: The mitigation measure's design and intended use makes it effective at preventing the unsafe condition or mode from resulting in (safety) losses.

Integrity: The mitigation measure is available, its condition is intact, and it can be relied upon to work under the expected circumstances.

Robustness: The mitigation measure will remain functional after the unsafe condition or mode has occurred, taking any disturbances and/or accidental loads into account.

Independence:

- of the event which initiated the unsafe condition/ mode
- of each other (in case a mitigation fails)

A mitigation measure cannot depend on an agent which has already failed as part of the accident scenario. This means that it cannot depend on the performing agent of the failed control action or on the supervisory agent responsible for a preceding mitigation measure to function successfully. Human can provide the decision-making when it comes to activation for more than one mitigating measure identified as relevant for an accident scenario. But credit can only be given provided that these mitigating measures rely on different supervisors when it comes to detection of the relevant unsafe conditions. In other words, credit can be given for multiple mitigation measures, but not more than the number of independent detection methods available to the human; otherwise, there will be vulnerability to common cause failures. Similarly, if there is a typical recovery problem, credit can be given for multiple fallback states involving humans, provided they are independent of each other. It is important to distinguish between scenarios where the issue is maintaining control once a problem is detected and scenarios where the challenge is detecting the problem in the first place. The latter often relates to navigation problems, with further guidance provided in section 2.3.3.2 and 2.3.3.4.

Systems performing essential continuous functions across the failed control action and (several) mitigation measures, and for which independence cannot be demonstrated, must be identified, and analysed separately.

Human involvement: A final criterion is that the mitigation measures are designed and implemented in such a way that it ensures successful human-automation interaction. At the time that a conceptual analysis like RBAT is performed, the details in this area may not be fully known. Thus, assumptions may have to be made, see section 2.3.3.4 for more about human involvement and 2.5.1 for more about assumptions.

Additional guidance for assessing functionality, independence, and human involvement is provided below in sub-chapters, 2.3.3.2 and 2.3.3.4.

How to perform the qualification:

The qualification itself is qualitative and based on the knowledge available at the time RBAT is used. The conclusions are binary – a mitigation measure is either qualified or disqualified based on the user(s)[10] judgement.

In principle, a mitigation measure can be considered qualified when the user(s) feels confident that all the above-mentioned criteria are fulfilled, across any causal factors identified as relevant.

If knowledge is available which indicates that one or more of the criteria cannot be met, the mitigation measure is disqualified and shall be removed from the RBAT mitigation analysis (i.e., it shall not be taken credit for as part of risk evaluations, Step 18).

It is acknowledged that limited information may be available about the mitigation measures, particularly in the preliminary design stage. In cases where assumptions must be made about the mitigation measures' performance and pre-requisites, these should be noted down (e.g., as part of a Safety Philosophy) so that they can be used to update the concept and included as part of verification and validation (V&V) efforts at a later stage.

In case a mitigation measure disqualifies, a comment should be made about why. If a risk is found unacceptable (see Step 18), disqualified mitigation measures can then be re-visited as the design matures and more knowledge is obtained. The approach therefore benefits from being conservative in the early stages, by not having to disqualify mitigation measures at a later stage which potentially may result in unacceptable risks.

2.3.3.1 Choice of Active vs Passive Human Supervision

The ROC operator may compare information from many sources and determine that an unsafe condition is present, even if there are no alarms. Information sources may be camera images, radar picture, position plots, etc. This is similar to an operator being present on the bridge of a manned ship.

[10] *Users* here also includes potential reviewers and approvers.

Mitigation measures relying on this form of detection may be utilised to reduce risk in a large number of scenarios in RBAT, however in line with the guidance in section 2.3.3.3 there should not be more than one mitigation measure that utilises this form of detection per scenario.

If passive human supervision is selected, the consequence in the RBAT analysis may be many scenarios where the level of effectiveness of mitigations is **reduced by one level** compared to a situation where active human supervision is being used.

Note that in many scenarios there may be a residual risk of unsafe conditions caused by systematic or systemic failures that can only be detected in this way, see section 0 for more about detection of unsafe conditions. For this reason, active human supervision may be preferable in situations where detection of unsafe conditions is time critical.

See section 2.4.3 for more about the use of different forms of human supervision in different mission phases.

2.3.3.2 Additional guidance on independence

Additional guidance about how to assess mitigation measure independence is provided in Table 15 below.

Table 15 Perspectives on mitigation measure independence

Perspective	Descriptions	Examples
Composition	This perspective is used to evaluate whether there are any physical or software components used in the mitigation measure that may be affected by failures in components where an unwanted event has manifested itself.	A mitigation measure that relies on thrusters being reversed will not be independent if the initiating event occurred in the thruster itself or in the thruster control system. Two different types of software applications executed on the same controller will typically be dependent because they will share hardware and software components[11] A system may have several and different types of sensors which can trigger a safety function representing a mitigation measure. However, if the same controller and actuators are used regardless of type of initiation, there may only be one full mitigation measure available.
Environment	This perspective evaluates whether there are items outside the system and/or external events that may act upon the system, cause an unsafe condition and impair the mitigating layer.	■ Loss of cooling in control rooms ■ Fire ■ Water ingress or flooding ■ Lightning strike ■ Radio communication jamming ■ Electrostatic discharge ■ Unexpected wind or wave conditions
Structure	This perspective looks at the relationships and interdependencies between the system constituents, and between the system constituents and the environment.	Two systems/functions that are otherwise considered independent may both rely on the Power Management System being operational. An equipment-specific protection mechanism may have the authority to reduce capacity to prevent equipment damage in a situation where the mitigation measure requires full capacity from that equipment to be effective. An operator may depend on alarms from the main control system to understand that a failure has occurred, and that activation of a mitigation measure is needed. If an unexpected scenario for which no alarm has been defined should occur (i.e., an annunciated failure), the mitigation measure may not be activated in time to prevent a mishap.
Mechanisms	This perspective evaluates dependencies that may be introduced through systems/functions or components having common requirements, common design, or common implementation*.	The controllers in a redundant control system are typically not independent of each other if a failure has systematic or systemic causes. This is because the two controllers typically will have common requirements, common design, and common implementation. Consequently, they will react in the same way to unexpected input: values, input combinations or input sequences. Two different GPS-based positioning reference systems may have different designs and implementations. However, in case of unexpected input the systems may still fail in the same way as the functional requirements for such systems may be very similar**.

*Avoiding these kinds of dependencies may require some form of diversification, as described below.

** It is common to combine information from positioning references based on different principles to mitigate this kind of common cause through functional diversity as discussed below.

[11] Note that there are safety controllers that provide so called logical separation. In such cases the Commercial Of The Shelf (COTS) hardware and software components such as the operating system have been qualified for use in high-integrity systems and designed in such a way that individual software tasks cannot negatively influence each other through timing, memory space or I/O.

1) Functional diversity involves solving the same problem in different ways.

- This kind of diversity reduces the likelihood, that functional requirements which are inadequate for one or more operational scenarios will lead to dangerous systematic or systemic faults.
- Use of functional diversity may in some cases also lead to use of design diversity as discussed below, but not always.

2) Design diversity involves the use of multiple components, each designed in a different way but implementing the same function. E.g., one may use a CPU in combination with a Field Programmable Gate Array (FPGA).

This kind of diversity may be used to detect, isolate, and recover from systematic failures introduced at software design and coding level, as well as in hardware design and manufacturing. It may also be used to detect random hardware faults.

This kind of diversity is not effective against systematic/systemic failures introduced in functional requirements specifications in the same way as functional diversity. It should be noted that software and hardware in controller(s) comparing and/or merging information from diverse functions and/or diverse components may introduce common mode failures.

2.3.3.3 Unsafe conditions related to navigation, for which detection is the main challenge

When it comes to unsafe conditions that could lead to loss of control over the vessel and unsafe conditions that could lead to collision or grounding due to inadequate navigation, there is typically a fallback chain which, depending on scenario, may include for example:

1. Evasive manoeuvre initiated by the Autonomous Navigation System
2. Station keeping initiated by the Autonomous Navigation System
3. Autonomous Navigation System in autopilot mode (heading and speed decided by ROC operator)
4. Station keeping initiated by the ROC operator
5. Idling of thrusters by autonomy system while waiting for ROC intervention
6. Joystick control by ROC operator utilising onboard backup system
7. Direct control of azimuth thrusters from ROC utilising second onboard backup system
8. Emergency shutdown of thruster initiated from ROC with subsequent preparation for anchoring or towing.

This example fallback chain corresponds to the fallback states present in the list of operational states in Table 14 in section 2.3.2.4 *Identify relevant operational states after the mitigating measure have been applied.* Guidance on how to take credit for the fallback chain is included below.

When performing a functional risk analysis like RBAT for the subfunctions and control actions inside the overall “Perform Navigation” function, one will identify several scenarios where the vessel is fully controllable when it comes to performing manoeuvres, but where the risk is that inadequate navigation may lead to collisions or groundings.

For such scenarios, the strength of mitigations should typically not be determined by the availability of the full above-mentioned fallback chain, but by several independent mitigations available that can detect the need for and subsequently activate evasive manoeuvre or station keeping. To achieve this, only the first 4 elements in the fallback chain may be relevant. If evasive manoeuvre or station keeping is not attempted, the measures in the remaining fallback chain will not be attempted, and therefore, they do not provide any real risk reduction when it comes to scenarios where the vessel is fully operational, but navigation is inadequate.

In such scenarios, the number of independent mitigations should not be considered higher than:

- The number of independent supervisors capable of detecting the unsafe conditions, e.g. Collision and Grounding Avoidance System, Radar system, and ROC operator utilising Camera system are examples of possible supervisors.
- The number of different ways the unsafe condition can be detected, e.g. use of camera, radar, lidar, position plot etc.

When it comes to the first limitation, a human supervisor may be involved in more than one mitigation measure, as long as the supervisors providing the detection are different.

See Table 21 in section 2.5.1 for an example where 3 different mitigating measures have been proposed, but credit has only been taken for two, since the last mitigation would be independent of the others for some causes of unsafe conditions, but not all.

In some cases, scenarios associated with navigation may also affect the controllability of the vessel. For example, a scenario which is concerned with total loss of positional reference would fall into this category. The latter scenario will be detected in many ways, as the vessel for such a case will be significantly operationally degraded.

In such a case focus should not be put on ways to detect the problem, but on mitigation measures 5- 8 in the fallback chain which are not fully depending on positioning reference being available. In particular one should evaluate whether there could be a common cause problem capable of creating the unsafe condition and also inhibit that part of the fallback chain. Network storms may be an example of the latter.

2.3.3.4 Additional guidance on human involvement

For a mitigation measure to be qualified as effective, it must be designed and implemented in such a way that reliable human-automation interactions can be expected, assuming that operator actions are required.

This is assessed by asking whether it is possible for the operator(s) to:

- Detect and observe (perceive) the situation (information acquisition)?
- Make sense of the situation and predict future outcomes (information analysis)?
- Select a course of action among several alternative options (decision making)?
- Execute activities required to achieve the desired outcome (implementation of actions)?

Answers to these questions are found by determining whether one or more hindrances are present (see Table 16) and if their effect(s) on human-automation interaction is so negative that the required operator action(s) can be argued to fail.

During the design process the hindrances will concern technical *performance shaping factors* (PSFs) such as alarms, control panels and other human-software interfaces (HMI), communication systems, automation design, equipment performance and tolerances, and more.

Particular attention should be devoted to examining dependencies between the system failures which initiate the unsafe condition, and the systems which operators rely on to perform actions required for mitigation measures to be successful. For example, in case a software-related error causes an unannunciated failure, the chances for an operator to act diminishes significantly.

Towards and during the operational phase the influence from other non-technical PSFs will emerge, such as procedures, training, and supervision. Although such factors can have a positive effect on human performance, they should not be an excuse to allow sub-optimal solutions at the earlier design stages.

If there are uncertainties about whether successful human-automation interaction can be expected, a more detailed analysis of the required operator actions should be done prior to qualifying the mitigation measure. For this purpose, it is recommended to use a recognized human reliability analysis technique (Blackett et al., 2022), or a similar risk analysis method based on task analysis.

Table 16 Hindrances for successful human-automation interaction

Information processing stages	Hindrances
Information acquisition *Perception of sensory information about the situation*	■ There is no information available ■ There is too much information available ■ Information can easily be missed ■ Information can easily be misperceived (e.g., misheard, misread) ■ Information is misleading (e.g., expected but incorrect)
Information analysis *Making sense of the situation and predicting future events*	Information analysis requires large amounts of information to be interpreted and memorized/recalled ■ Information analysis requires significant interpretations of uncertainties in parameters (incl. future events) ■ Information analysis requires understanding complex dependencies between different parameters ■ Information analysis requires factoring in the impact of unpredictable events (e.g., environment)
Decision-making *Selecting a course of action among several possible alternative options*	■ The decision basis is insufficient and/or unclear ■ There are too many paths, options, goals and/or they are contradicting, conflicting, or competing ■ How to prioritize paths, options, goals is unclear ■ The plan (e.g., a procedure) does not match the situation ■ Outcomes from decisions are uncertain
Implementation of action(s) *Executing activities required to achieve desired outcome*	■ Opportunities for successfully exerting control is limited, e.g., due to being remotely located ■ There is insufficient time (or other required resources) available to successfully perform the required actions ■ Expected amount of training and experience is not likely to raise and maintain required skills at an adequate level ■ There are few or no feasible opportunities to recover and correct an erroneous action.

2.3.3.5 Guidance related to how agents can be involved in mitigation measures

Table 17 below provides guidance on which agents than be involved in mitigating measure, and to what extent such agents also can be involved in more than one mitigation measure and/or in FDIR mechanisms. and their limitation to what extent they can be part of a mitigation measure and given credit for.

Table 17 Limitations to the role of the agent in FDIR and mitigating measures

Lim.no.	Type of agent	Limitations	Comment
1	The agent is the performing agent for the scenario being analysed, and it is a software agent	A software performing agent can be involved in FDIR mechanisms, but not in any mitigating measures identified as relevant for the scenario.	
2	The agent is the performing agent in the scenario being analysed, and it is a human agent.	A human performing agent can be involved in FDIR mechanisms and also in one or more mitigation mechanisms.	See limitation no. 4 below, regarding human involvement in mitigation mechanisms.
3	The agent is one of the supervisory agents identified as relevant in the scenario being analysed, and it is a software agent	A software supervisory agent can be involved in either a FDIR mechanism or a single mitigation measure.	Modifier: If (a mitigation is used frequently during normal operation so that its status is known) AND (the detection part of the mitigation measure is independent from the detection part of the FDIR mechanism) then the supervisory agent may be involved in

Lim.no.	Type of agent	Limitations	Comment
			both the FDIR mechanism and in that single mitigating measure. Such a claim should be substantiated, see section 2.4.2 regarding alternative methods for risk classification.
4	The agent is one of the supervisory agents identified as relevant in the scenario being analysed, and it is a human agent.	A human agent can be involved in FDIR mechanisms and also in all mitigation mechanisms, unless risk analyses or regulatory requirements should require the involvement of additional human supervisors.	Note that the number of mitigation measures shall not be higher than the number of independent detection mechanism. In many scenarios that will limit how many mitigations that can be taken credit for in the scenario. See also limitation no. 5 below.
5	The agent is providing detection of the unsafe condition.	The detection provided by the agent can be taken credit for either as a part of FDIR or as a part one specific mitigation measure.	This limitation exists to make sure that a specific detection mechanism is only credited once in a scenario. To what extent the agent can be involved in FDIR mechanism or mitigation measures beyond providing detection is covered by the other limitations.
6	The agent is not a performing agent, nor a supervisory agent, but is involved in control of a function that need to work as intended if FDIR and/or one or more of the mitigation measures shall be effective	If the agent is used continuously during normal operation so that the status of the function that it is involved in is known, then the agent can be involved both in FDIR mechanisms and in the mitigation measures. If the agent is not used continuously, it can be involved in either a FDIR mechanism OR a single mitigation measure.	Agents continuously involved in thruster control and power management are example of agents that can be involved both in FDIR and in several mitigating measures. It is more likely that such essential functions will fail without any failure in other functions, than that they will fail at the same time that another function has also failed. The severity will be the same in both cases, and therefore, the scenarios where such agents are failing shall be explored in analyses of functions where they are the performing agents.

2.3.4 Step 14: Rank the mitigation measures' effectiveness

The fourteenth step of the process is to rank how effective the mitigation(s) is/are at preventing losses, using the index provided in Table 18 below. For control systems the thinking behind the index is as follows:

For a control function which is not redundant, the effectiveness provided by FDIR mechanisms within the performing agents is considered *Low* when it comes to management of unsafe conditions caused by hardware or software failures inside the performing agent. There may be mitigation measures that can prevent losses from some types of random hardware failures, but the function being analysed is not single hardware fault tolerant nor fully tolerant to systematic/ systemic faults.

- Typically, there are FDIR mechanisms within the performing agents, which are included to manage unsafe conditions in the input provided to the performing agent from external components. Such FDIR mechanisms provide more independence than mechanisms aimed at managing failures inside the performing agent. However, for such an FDIR mechanism, there will typically be types of systematic/systemic faults in the input that cannot be mitigated without external intervention. Thus, the effectiveness of such FDIR mechanisms in the system should at the most be classified as *Moderate*.
- A standard critical control system used in the maritime industry is expected to be redundant. This implies that there is least one internal mitigation (i.e., FDIR) that can prevent losses from various types of random hardware

failures within the redundant performing agents. There may also be mitigation measures that can prevent losses from some types of systematic faults, but for such systems there will typical be types of systematic/systemic faults that cannot be mitigated without external intervention. Thus, the effectiveness of the internal mitigations in the system should at the most be classified as *Moderate*.

- A mitigation measure will increase the strength of the mitigating measures by one level. For example, an independent emergency function that can mitigate a control failure in a standard control system will raise the strength from Moderate to Medium. A further strengthening to High will require a second independent mitigation, and so on.

Note that a further decomposition of a function and a subsequently more detailed analysis may reveal that there are several independent supervisors within a high-level function which may improve the risk picture.

Also note that is possible to explore alternative justifications for determining risk levels, see section 2.4.2.

Table 18 Effectiveness of Mitigations

Effectiveness		Description
Extremely high	Very high	At least four effective mitigation measures can for the assessed scenario prevent losses *regardless* of what caused the unsafe condition.
Very high	High	At least three effective mitigation measures can for the assessed scenario prevent losses *regardless* of what caused the unsafe condition.
High	Medium	At least two effective mitigation measures can for the assessed scenario prevent losses *regardless* of what caused the unsafe condition.
Medium	Moderate	At least one effective mitigation measure can for the assessed scenario prevent losses *regardless* of what caused the unsafe condition.
Moderate	*FDIR not available*	FDIR mechanisms built into the performing agent can prevent losses when the unsafe condition is caused by single random hardware failure or by some types of systematic or systemic failures*.
Low	Low	No or limited capacities for fault detection, isolation, and recovery are available, however (if present), for the assessed scenario these are expected to have a limited effect.

*The list below contains some examples of effects that may be caused by systematic or systemic failures, which FDIR functionality realized within the performing agent for the function being analysed typically may be capable of detecting and mitigating.

The list is by no means exhaustive:

- Software crash or software hang up
- Expected data not being received in internal communication
- Data received in internal communication being out of range, corrupted, or out of sequence
- Date received in internal communication being received too late
- Internal tasks performing too slow Internal data that is unexpected from a statistical point of view, e.g., temporarily unexpected variations in received data
- Internal commands that are illegal in the current system state
- Stack overruns

Note that there are alternative approaches for determining risk levels.

2.3.5 Step 15: Identify prevention measures (optional)

An (optional) step of the process is to identify any measures which exist to *prevent* the occurrence of unsafe conditions. This includes activities which provide assurance that the required performance can be expected, such as maintenance, testing and inspection for technical equipment. As with mitigation measures, only measures which have already been documented prior to the assessment should be included.

Prevention measures should not be mistaken for operational restrictions.

2.4 Part 4: Perform risk evaluation

The purpose of performing risk evaluation is to compare the risk level for each assessed scenario against a set of risk acceptance criteria to determine the need for risk control.

- Determine risk level for each assessed scenario (Step 16)
- Describe alternative approaches for determining risk levels (Step 17)
- Run sensitivities to check for supervisory control effects (Step 18)

2.4.1 Step 16: Determine risk level for each assessed scenario

The sixteenth step of the process is to determine the risk level for each assessed scenario.

In RBAT the level of risk for each scenario being analysed is determined based on 3 factors:

1. Exposure to Enabling conditions, see section 2.2.3.
2. How severe the worst-case outcome of a scenario is after considering the effect of operational restrictions, see section 2.2.6.
3. Effectiveness of Mitigations, see section 2.3.4.

This is illustrated through three example risk matrixes shown in Table 19 below. These are intended for risks associated with Health, Safety and Environment (HSE).

It is also recommended to capture risks associated with the ship itself, uptime, reputation etc. However, different risk matrixes may typically be used to evaluate such risks.

Table 19 Risk as a measure of Exposure to Enabling Condition, Severity and Mitigation effectiveness

High exposure – no influence (baseline)

Mitigation	Severity				
	Negligible	Minor	Significant	Severe	Catastrophic
Low	Medium	High	High	High	High
Moderate	Low	Medium	High	High	High
Medium	Low	Medium	Medium	High	High
High	Low	Low	Medium	Medium	High
Very high	Low	Low	Low	Medium	Medium
Extremely high	Low	Low	Low	Low	Medium

Medium exposure – one level (from baseline)

Mitigation	Severity				
	Negligible	Minor	Significant	Severe	Catastrophic
Low	Low	Medium	High	High	High
Moderate	Low	Medium	Medium	High	High
Medium	Low	Low	Medium	Medium	High
High	Low	Low	Low	Medium	Medium
Very high	Low	Low	Low	Low	Medium
Extremely high	Low	Low	Low	Low	Low

Low exposure – two levels (from baseline)

Mitigation	Severity				
	Negligible	Minor	Significant	Severe	Catastrophic
Low	Low	Medium	Medium	High	High
Moderate	Low	Low	Medium	Medium	High
Medium	Low	Low	Low	Medium	Medium
High	Low	Low	Low	Low	Medium
Very high	Low	Low	Low	Low	Low
Extremely high	Low	Low	Low	Low	Low

As requested by EMSA, it is here recommended that the “as low as reasonably practicable” (ALARP) principle is applied for risk evaluation[12]:

- High (red region): Risk cannot be justified and must be reduced, irrespectively of costs.
- Medium (yellow ALARP region): Risk is to be reduced to a level as low as reasonably practicable.
- Low (green region): Risk is negligible, and no risk reduction is required.

The term *reasonable* is interpreted to mean cost-effective. Risk reduction measures should be technically practicable, and the associated costs should not be disproportionate to the benefits gained. The FSA guideline extensively explains how to perform cost-benefit assessments (IMO, 2018) and is therefore not repeated here.

2.4.2 Step 17: Alternative approaches for determining risk levels

The seventeenth step of the process is to explore alternative justifications for determining risk levels. While this is not expected to be a standard part of using RBAT, cases may arise where arguments for lowering the risk level appears to be justifiable.

When comparing the risk picture associated with a specific function and corresponding risk mitigation measures to relevant acceptance criteria, the following alternatives for risk evaluation can be considered:

1. If the initiating event[13] is not related to software control, it may be possible to argue for a lower probability than what has been generally anticipated for control functions. In that case, fewer independent risk mitigation measures may be required to meet the acceptance criteria. For such events, the classical type of risk matrix shown in Table 20 can be used as a starting point to determine the initial risk picture before looking at available mitigation measures.
2. It should be possible to argue that a single mitigation will increase the effectiveness of the mitigation by more than one level. One example may be that if it can be demonstrated that an emergency stop function for machinery has a Performance Level (PL) = *d* performance according to the ISO 13849 safety standard for machinery, this would be considered a two-level increase.
3. It should also be possible to demonstrate that critical control functions have a better performance than what is anticipated in the default scheme in RBAT. Such claims should be substantiated in an Assurance Case or similar. More advanced forms of risk analysis, carefully selected components, and sharper development processes than what have traditionally been applied in the maritime industry may be required to substantiate such claims.
4. The assumption that a single control-related mitigating measure that has not been developed to a high integrity level according to standards like IEC61508, IEC 61508, and ISO 13849 will increase the effectiveness of the mitigation by only one level, is based on the scheme used for low-demand safety functions in IEC 61511. However, if a mitigating measure is a control function that is frequently used also during normal operation, it may be treated as a high-demand function. Subsequently, a quantitative analysis that considers the number of hours per year that the mitigation is likely to be needed, and the number of hours per year that the mitigation is estimated to be not working as intended due to failure, can be used to calculate the likelihood that the latter will occur in one of the time periods where the mitigation is needed. If this method is used, a conservative estimate of the likelihood both for the unsafe condition to occur and for failure in the mitigating measure should be utilised. With this approach, the classical type of risk matrix shown in Table 20 can be employed. Station keeping is an example of a mitigation measure that may be frequently used also during normal operation.

[12] MSC-MEPC.2/Circ.12/Rev.2, chapter 4.
[13] Causal factor(s) initiating the event which results in an unsafe condition

The pursuit of any such alternative approaches needs to be thoroughly argued for and carefully documented. As it is not within the scope of RBAT to suggest how this is done in practice, each user must determine what is the best possible approach to meet the expectations of approvers and other stakeholders.

Table 20 Example of a classical risk matrix

			Severity				
Probability of experiencing an unsafe condition per year			Negligible	Minor	Significant	Severe	Catastrophic
Frequent	>=1		Medium	High	High	High	High
Probable	>=1/10	To <1	Low	Medium	High	High	High
Occasional	>=1/100	To <1/10	Low	Medium	Medium	High	High
Remote	>=1/1000	To <1/100	Low	Low	Medium	Medium	High
Very remote	>=1/10000	To <1/1000	Low	Low	Low	Medium	Medium
Improbable		<1/10000	Low	Low	Low	Low	Medium

2.4.3 Step 18: Run sensitivities to check for supervisory control effects

The eighteenth step in RBAT is to run sensitivities to check for effects in changes to how supervisory control is used. Supervisory control has a direct impact on the risk level through which mitigation measures can be relied on and qualified for certain scenarios, see section 2.3.2.2, 0 and 2.3.3.1, when it comes to supervisory control are in turn a result of the:

- number of vessels compared to the number of available operators (vessel-supervisor ratio),
- when and how vessels require attention during normal operation (operational philosophy),
- the degree of automation in specific functions, and
- the capability and reliability of automated systems.

A wish to assess the impact from multi-vessel concepts on the risk level is assumed to be the driving incentive for running sensitivities on effects from changes in supervisory control. RBAT, as a starting point, does not directly handle multi-vessel scenarios. This can, however, be evaluated indirectly, e.g., by making judgements about how an incident on one vessel creates supervision demands, which influences the supervision/ monitoring capacity of other vessels. For example, in case there are only two operators present in a remote-control room, and both must actively supervise at least two vessels during normal operations for certain mitigation measures to be qualified as effective, this is not valid in case one vessel requires the complete attention of one operator.

Implications from multi-vessel effects on supervisory control is illustrated in Figure 11, Figure 12 and Figure 13. Assuming there is only one operator available to supervise two vessels, the concept illustrated in Figure 11 could potentially disqualify mitigation measures which require active supervisory control to be successful in the mission phases “Arrival in port” and “Depart from port”. This is because one operator alone will have difficulties following two vessels simultaneously. For the concept illustrated in Figure 12 this is solved logistically by not having the two vessels entering a mission phase requiring active supervisory control at the same time. Figure 13 shows a concept like the one in Figure 11. However, this has solved the supervision conflict by enabling passive supervisory control throughout all the mission phases. This means that none of the mitigation measures require active supervisory control to perform successfully (and thus to be qualified).

For the moment, the method supports that different supervisory control types can be assigned to different mission phases. Note that this may lead to more scenario analysis where the type of supervision is the same for all phases. However, there are strategies that can be used to keep the number of scenarios down.

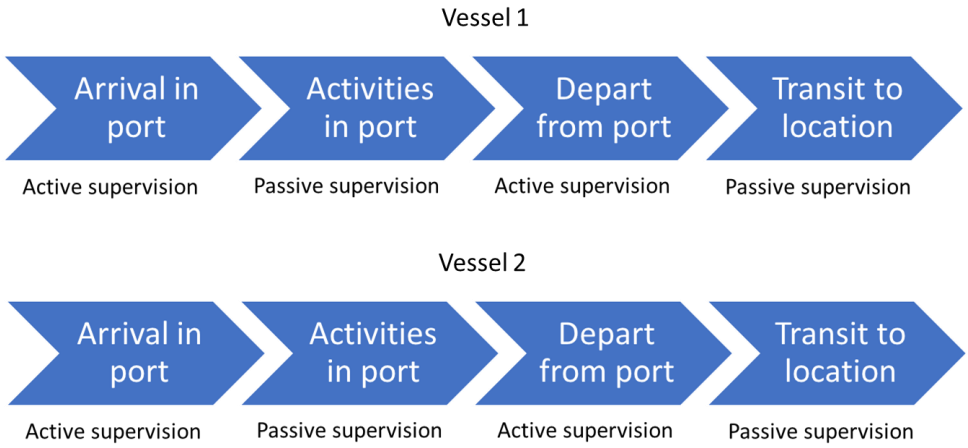


Figure 11 Two vessels simultaneously entering the same mission phases – mixed supervisory control

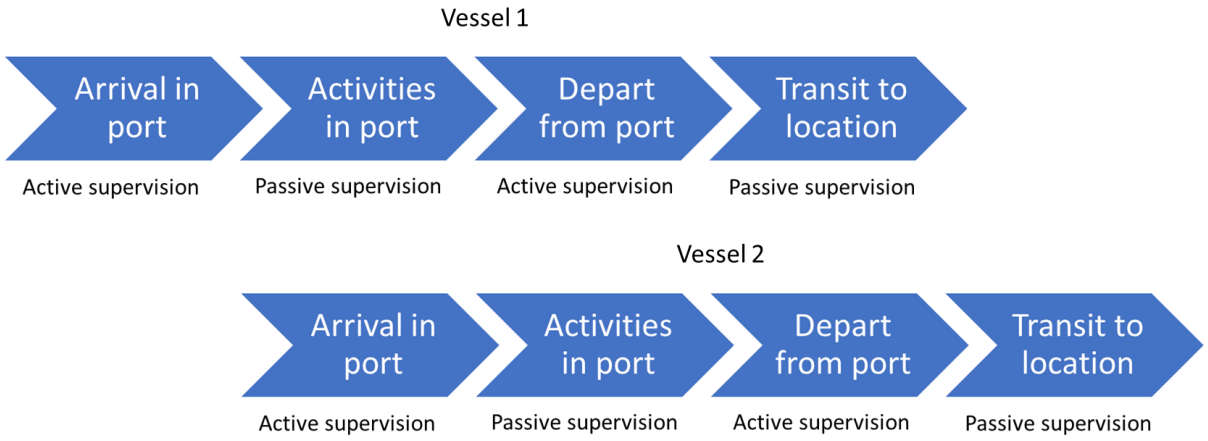


Figure 12 Two vessels simultaneously entering different mission phases – mixed supervisory control

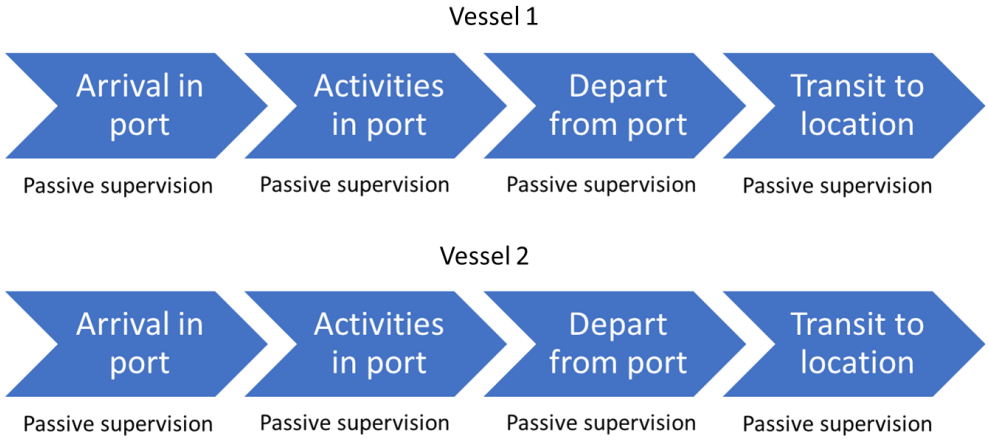


Figure 13 Two vessels simultaneously entering the same mission phases – passive supervisory control

2.5 Part 5: Address risk control

The purpose of risk control is to:

- Identify and document risk control measures ensuring that unacceptable (high) and tolerable (medium) risks are made as low as reasonably practicable (ALARP) (Step 19)

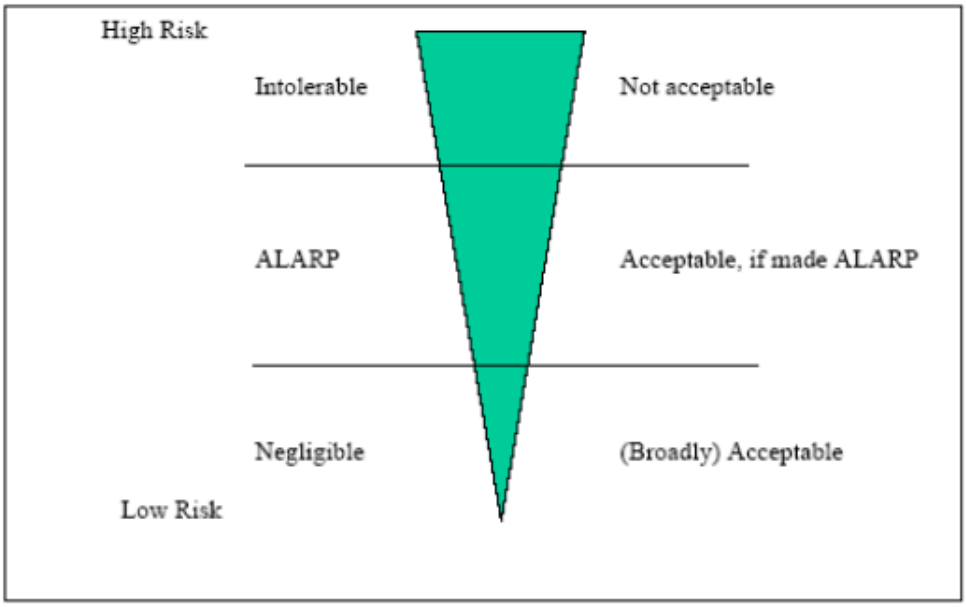


Figure 14 ALARP principle (IMO, 2018)

2.5.1 Step 19: Identify and document risk control measures

The nineteenth step in RBAT is to identify and document risk control measures.

In a high-level perspective, risk control measures can include:

- Updating the design by introducing FDIR and/or qualifying additional mitigation measures as effective so that they can be taken credit for as part of the risk evaluation.
- Removing or reducing the hazard associated with the control function, e.g., the fewer or less flammable hazards onboard, the less severe accident outcomes.
- Introducing operational restrictions which reduce the hazards potential impact, e.g., not allowed to sail close to shore in certain weather conditions or in high speed through traffic dense areas.
- Some operational restrictions related to speed and navigation may also reduce the exposure to enabling conditions, for example exposure to small crafts, kayaks etc.
- Improving the control functions integrity (and thus reducing its failure frequency) through design, component manufacturing and maintenance processes backed up by thorough assurance cases.

An elaborate description of generic RCM attributes (categories) can be found in the FSA guideline (IMO, 2018) and is therefore not described in any more detail here.

Important: If the analysis is done on a high function level, it will adopt the criticality of the most critical sub-function. In some cases, it may therefore be necessary to perform a more detailed risk analysis to confidently identify which control functions and actions are the most critical and should be targeted for risk control measures. This can be done using RBAT, but also other risk analysis techniques such as Failure Mode and Effect Analysis (FMEA) may be relevant.

Since the number of independent mitigation measures may be limited in many scenarios, it is quite typical that the resulting risk is classified as medium. For such risks, the scenario should be compared to a scenario where the same unsafe condition occurs onboard a manned vessel with a normal size crew. If the two scenarios are equivalent the risk may be acceptable.

For each scenario, a rationale for the risk classification shall be added in the form of a comment. See example in Table 21. Note that this particular rationale reflects the guidance related to typical fallback chains in 2.3.3.3 and the considerations related to independence in section 2.3.3.2.

Table 21 Example of risk classification with rationale

Unsafe condition/mode	Worst-case outcome (in case of no mitigation and no operational restrictions)	1st mitigation	2nd mitigation	3rd mitigation	Mitigation Effectiveness	Predicted Risk for HSE	Comments
No valid voyage plan present	Groundings due to arbitrary selection of course.	ANS initiating evasive manoeuvre or station keeping based on input from Grounding and Collision Avoidance System	ROC operator initiating station keeping or evasive manoeuvre based on information in the camera feed and/or from other relevant sources such as radar picture, ECDIS picture etc.	In case of grounding alarm from ECDIS, the ROC operator shall initiate either station keeping, or evasive manoeuvre based on operational judgement If the ROC operator does not react to the alarm within a specific time span, the ANS shall initiate station keeping	High	Medium	Once problems are detected there is also a chain of fallback that can be used to avoid collision in case of ANS related problems. These are joystick control, direct thruster control and emergency shutdown of thrusters. However, in this scenario no credit has been taken for that fallback chain, since detection of grounding and collision risk is the main challenge when it comes to managing voyage planning. This philosophy has been followed for all scenarios related to voyage planning There are potential common cause problems between mitigation 1, and 3. Thus credit has only been taken for FDIR and mitigation 1 and 2.

When it comes to documentation, the risk control measures coming out of RBAT will typically be represented in the following way:

- The list of operational restrictions will represent a set of safety requirements.
- The list of qualified mitigating measures will be a major source of safety requirements. Typically, some of these measures will already have been considered in the ConOps, but experience shows that this kind of analysis will typically identify the need for additional mitigation measures.

- The list of assumptions should also be treated as safety requirements, as some of the mitigating measures may be invalid if the assumptions are not correct.
- The list of actions will in a real-life project be used to register many different types of topics but may typically contain items concerned with candidate risk controls that have not yet been credited. Thus, the items on the action list typically reflect opportunities for further risk reduction.

Table 22 below contains an assumption and an action considered relevant for the risk classification in Table 21 above. If the assumption is correct, the Autonomous Navigation System can be considered an independent software supervisor for the Voyage Planning System which is something that has been credited in the analysis. However, if the assumption is incorrect, a more detailed common cause related analysis, may be required to evaluate if the RBAT analysis is valid. The action is aimed at checking the validity of the third mitigation measure in Table 21. As evident from the comment in that table, no credit has yet been taken for that last mitigating measure.

This example illustrates that assumptions are used to record and validate information that has already been credited in the analysis, while actions may be used to follow up on a mechanism that has not yet been credited.

Table 22 Example of assumptions and actions.

Assumption	Action
In this theoretical analysis, it is assumed that the Voyage Planning System (VPS) is separate from the Autonomous Navigation System (ANS) so that errors cannot propagate between them through memory, CPU time, or shared I/O.	A detailed analysis of possible common cause mechanisms between the GCAS, ANS, and the ROC operator when analysing input from positioning/digital chart and echosounder should be performed.

3. REFERENCES

DNV GL (2020a). Proposal for A functional study developing a Risk-Based Assessment Tool for MASS (RBAT MASS). DNV GL doc No: 1-1HPDRGR-M-N-ADSS-1.

DNV (2021b). Technology Qualification. Recommended Practice: DNV-RP-A203. Edition September 2021

DNV (2024). DNV-CG-0264 Class Guidelines on Autonomous and remotely operated vessels. Edition December 2024

EMSA (2020). Invitation to tender No. EMSA/OP/10/2020 for the functional study developing a Risk-Based Assessment Tool for MASS (RBAT MASS).

Endsley, M.R. (1995). "Toward a theory of situation awareness in dynamic systems". Human Factors. 37 (1)

International Electrotechnical Commission, IEC (2000). IEC 61839 Nuclear power plants – Design of control rooms – Functional analysis and assignment. First edition.

International Electrotechnical Commission, IEC (2009). IEC 60964 Nuclear power plants – Control rooms – Designs. Edition 2.0.

International Electrotechnical Commission, IEC (2013). IEC 60050-351 International Electrotechnical Vocabulary (IEV) - Part 351: Control technology.

International Electrotechnical Commission, IEC (2018). IEC 60812 Failure modes and effects analysis (FMEA and FMECA).

International Electrotechnical Commission, IEC (2020). IEC 61226 Nuclear power plants – Instrumentation, control and electrical power systems important to safety – Categorization of functions and classification of systems. Edition 4.0.

International Maritime Organization, IMO (2018). MSC-MEPC.2/Circ.12/Rev.2 – Revised guidelines for formal safety assessment (FSA) for use in the IMO rule-making process.

International Standard Organisation, ISO (2000). ISO 11064 Ergonomic design of control centres – Part 1: principles for the design of control centres. First edition.

International Standard Organisation, ISO (2009). ISO 31000:2009(E) Risk management – Principles and guidelines. First edition.

ISO/IEC/IEEE 15288:2015 Systems and software engineering -- System life cycle processes.

International Standard Organisation, ISO (2018). ISO 26262:2018 Road Vehicles – Functional safety. Second edition.

Leveson, N.G. & Thomas, J.P. (2018). STPA Handbook. Downloaded from: https://psas.scripts.mit.edu/home/get_file.php?name=STPA_handbook.pdf

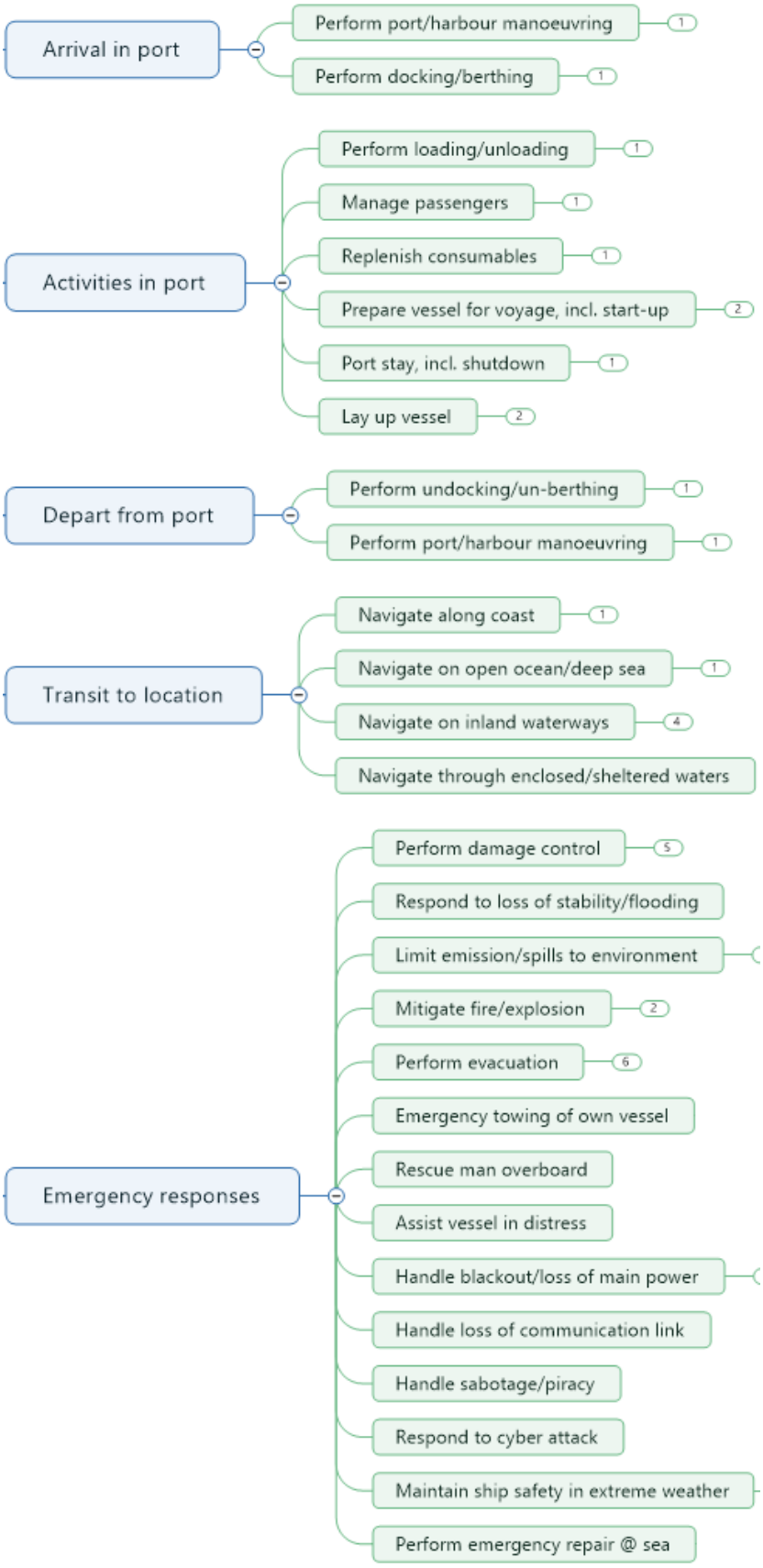
Parasuraman, R., Sheridan, T.B., Wickens, C.D. (1997). A model for types and levels of human interaction with automation. IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans 30, 286–297. https://doi.org/10.1109/3468.844354.

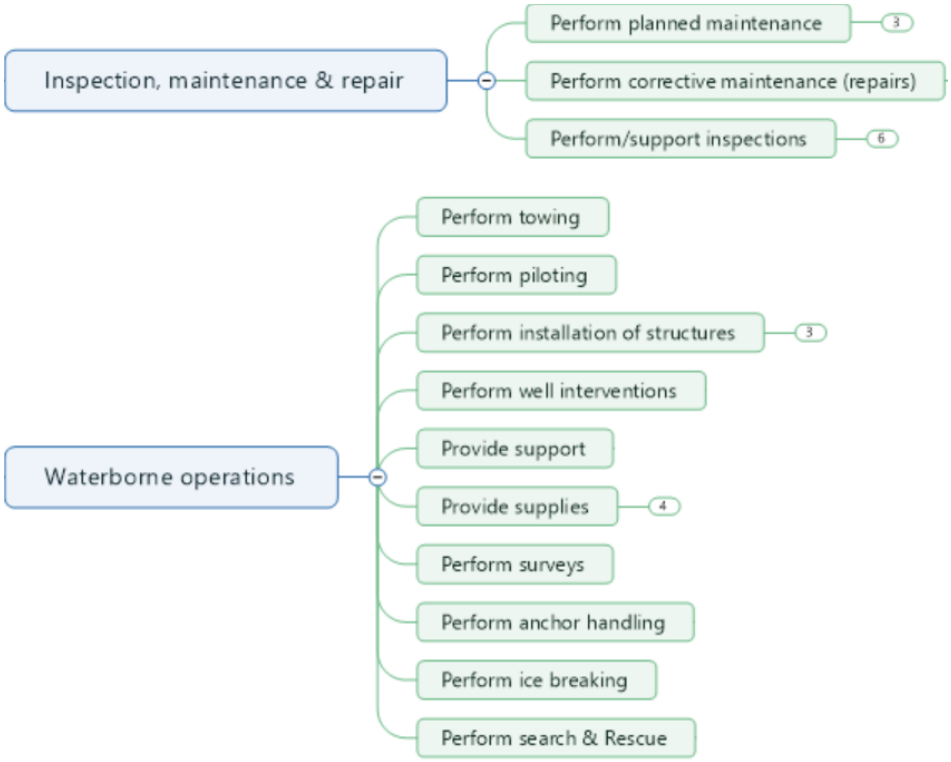
SAE Aerospace (1996). Guidelines and methods for conduction the safety assessment process on civil airborne systems and equipment. Aerospace Recommended Practice ARP4761. First edition.

Sheridan T. B., Parasuraman R. (2006). Human-automation interaction. In Nickerson R. S. (Ed.), Reviews of human factors and ergonomics (Vol. 1, pp. 89–129). Santa Monica, CA: Human Factors and Ergonomics Society.

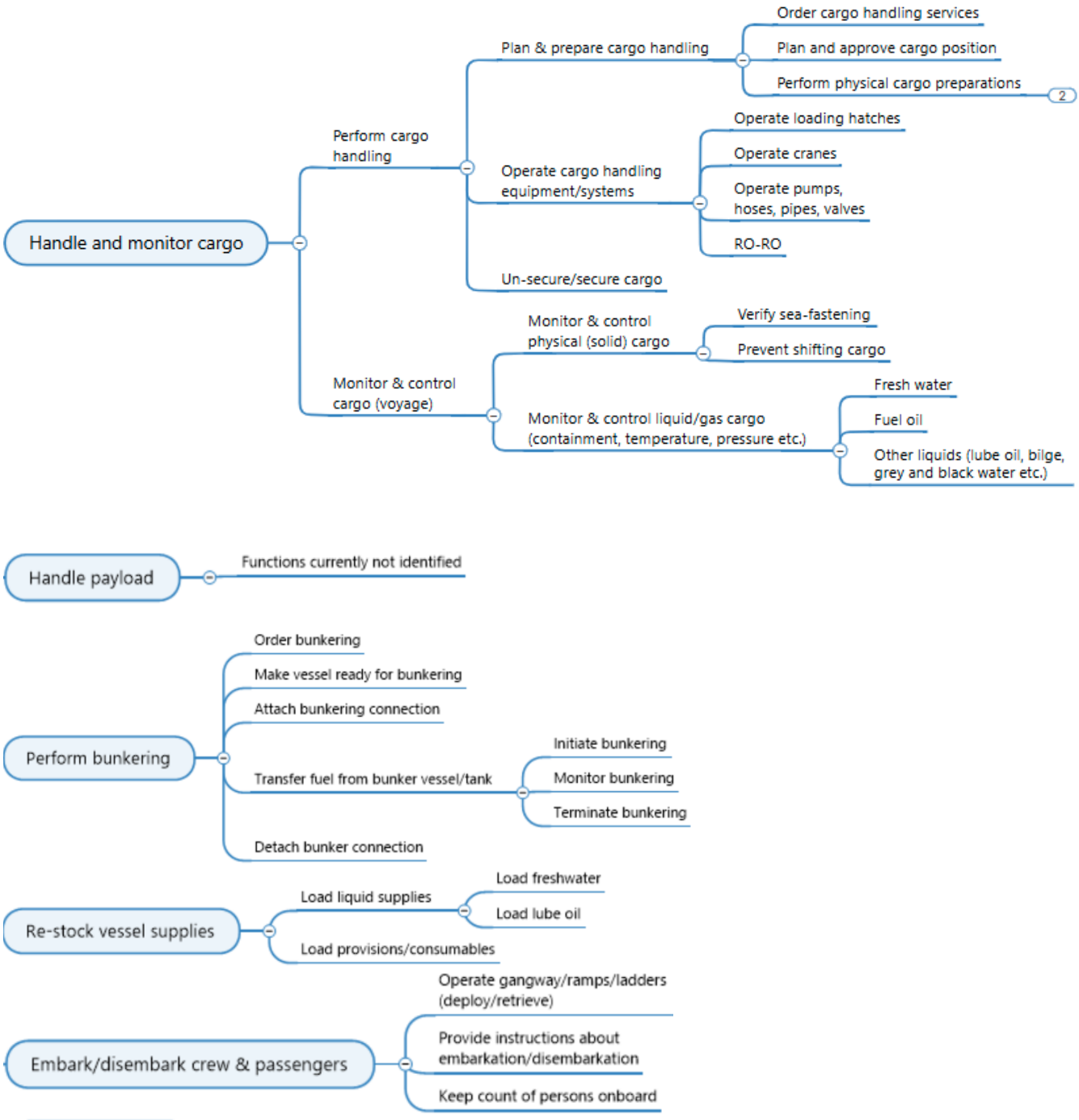
APPENDIXES

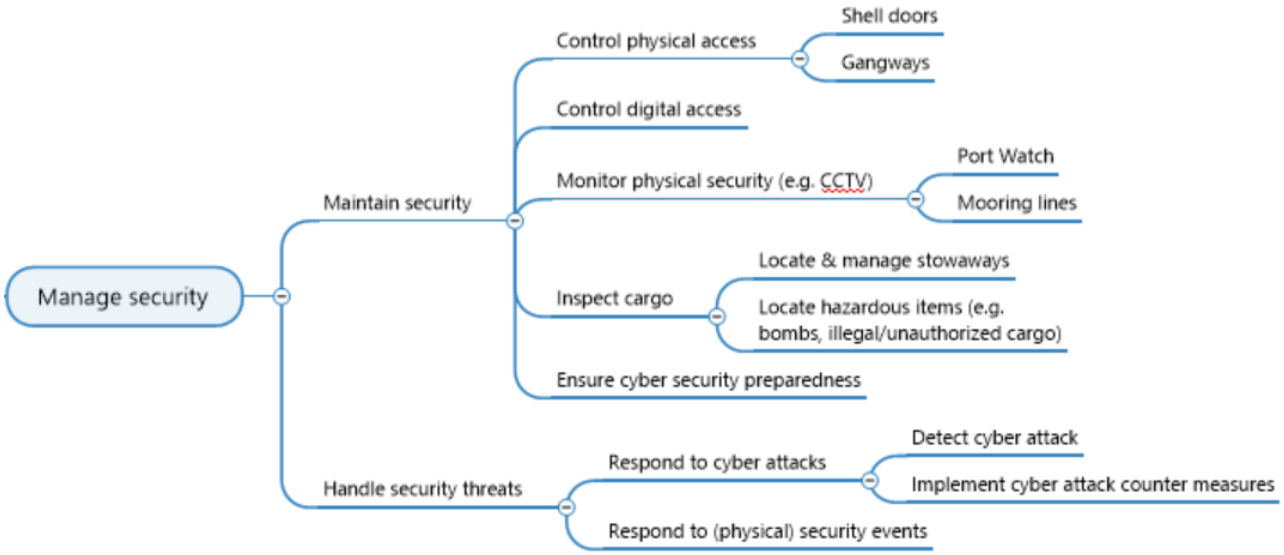
Appendix A RBAT MISSION MODEL

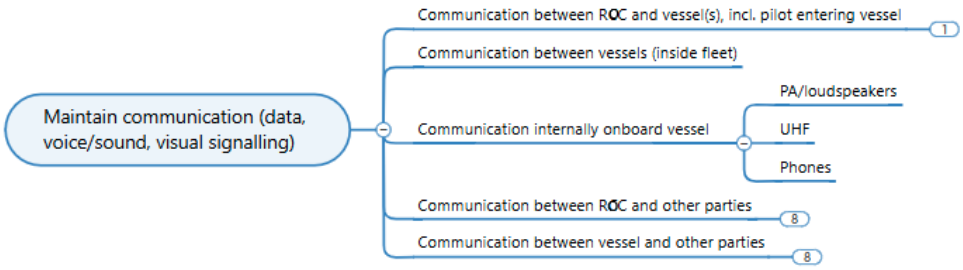


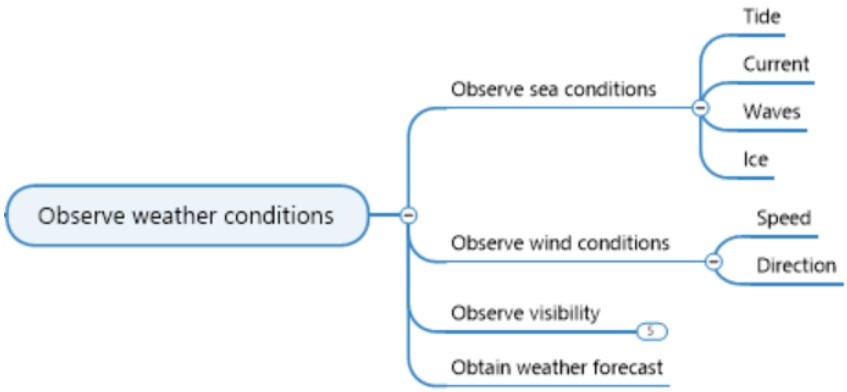

Inspection, maintenance & repair
Perform planned maintenance
3
Perform corrective maintenance (repairs)
Perform/support inspections
6
Waterborne operations
Perform towing
Perform piloting
Perform installation of structures
3
Perform well interventions
Provide support
Provide supplies
4
Perform surveys
Perform anchor handling
Perform ice breaking
Perform search & Rescue

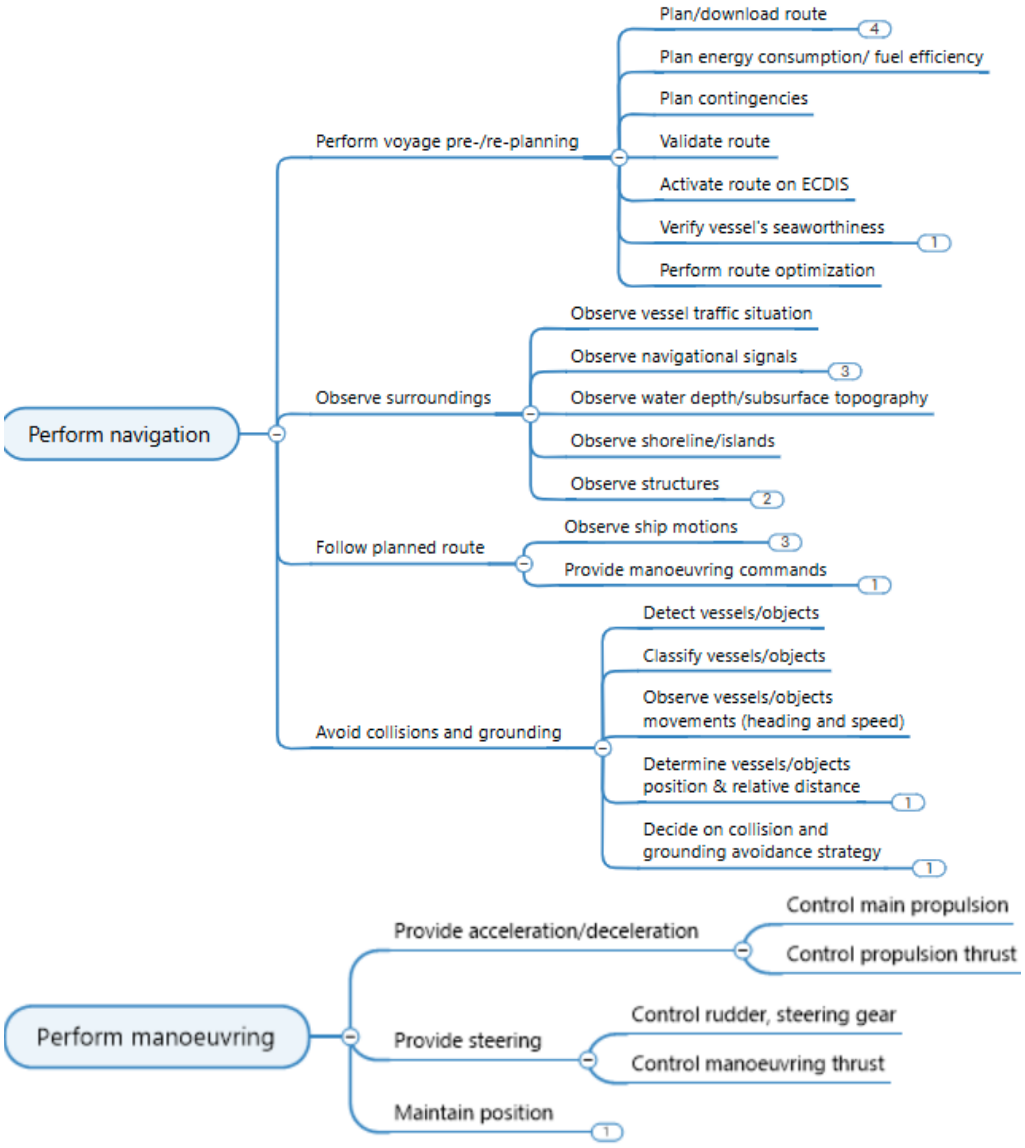
Appendix B RBAT FUNCTION TREE

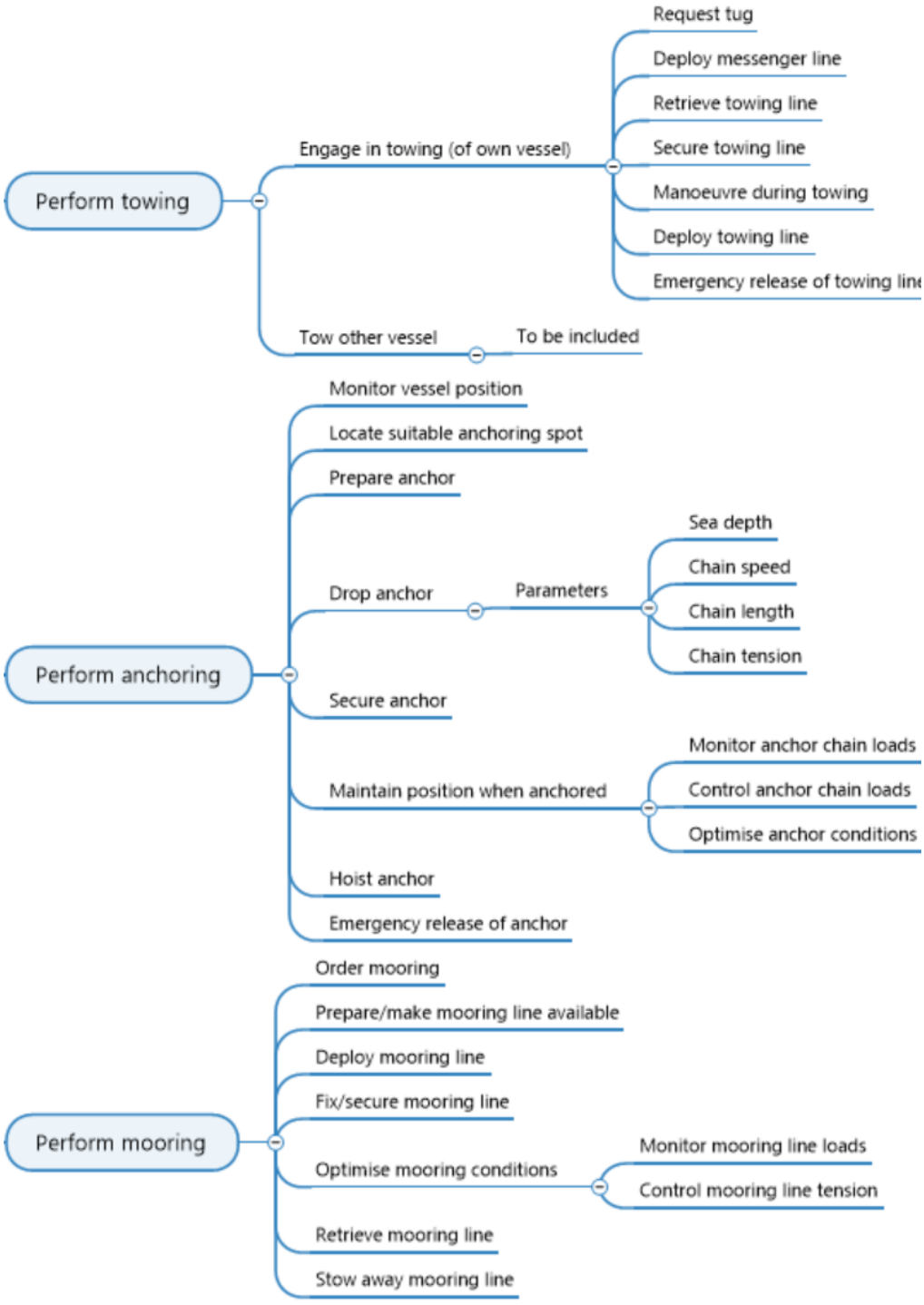


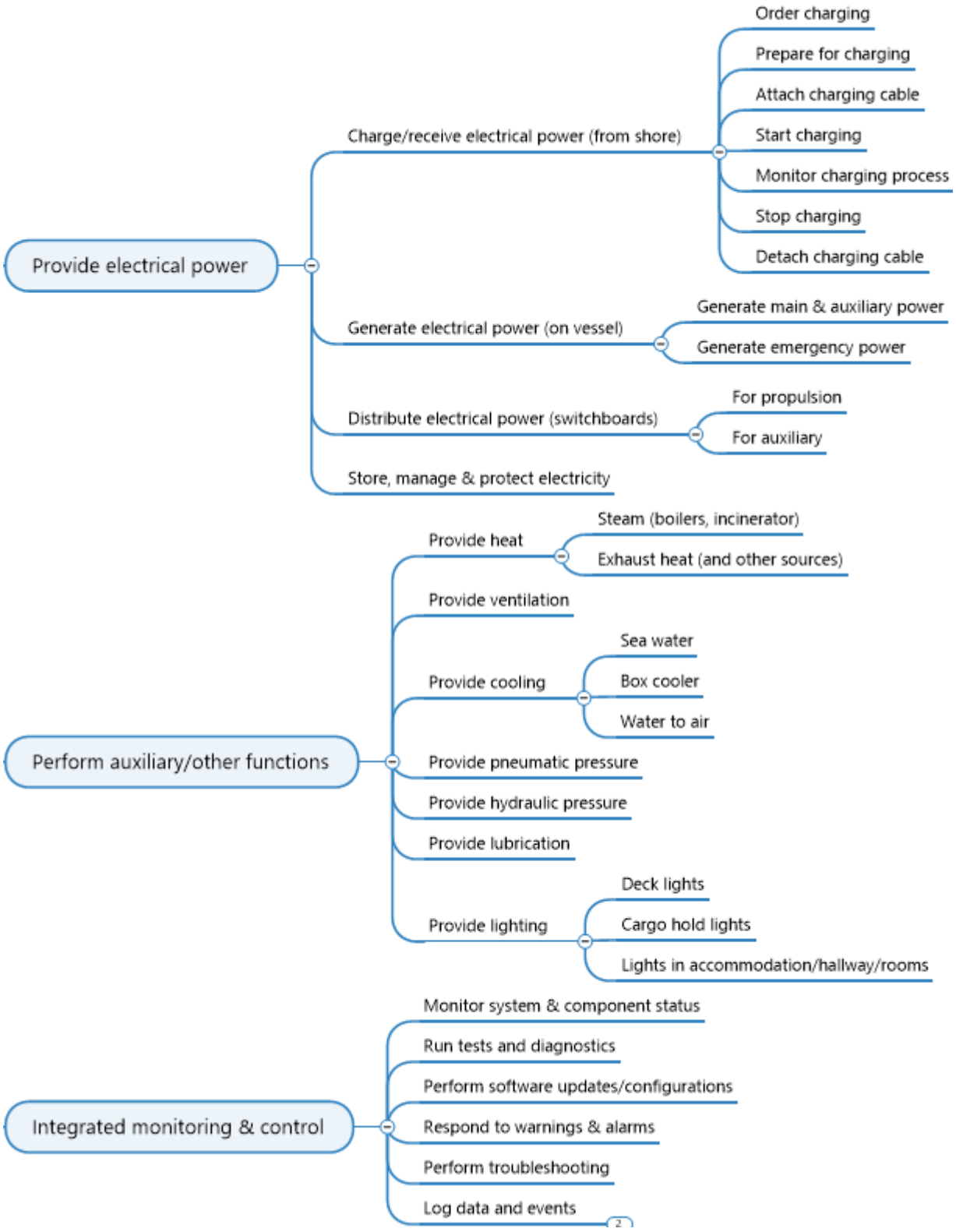

Manage security
Maintain security
Control physical access
Shell doors
Gangways
Control digital access
Monitor physical security (e.g. CCTV)
Port Watch
Mooring lines
Inspect cargo
Locate & manage stowaways
Locate hazardous items (e.g. bombs, illegal/unauthorized cargo)
Ensure cyber security preparedness
Handle security threats
Respond to cyber attacks
Detect cyber attack
Implement cyber attack counter measures
Respond to (physical) security events

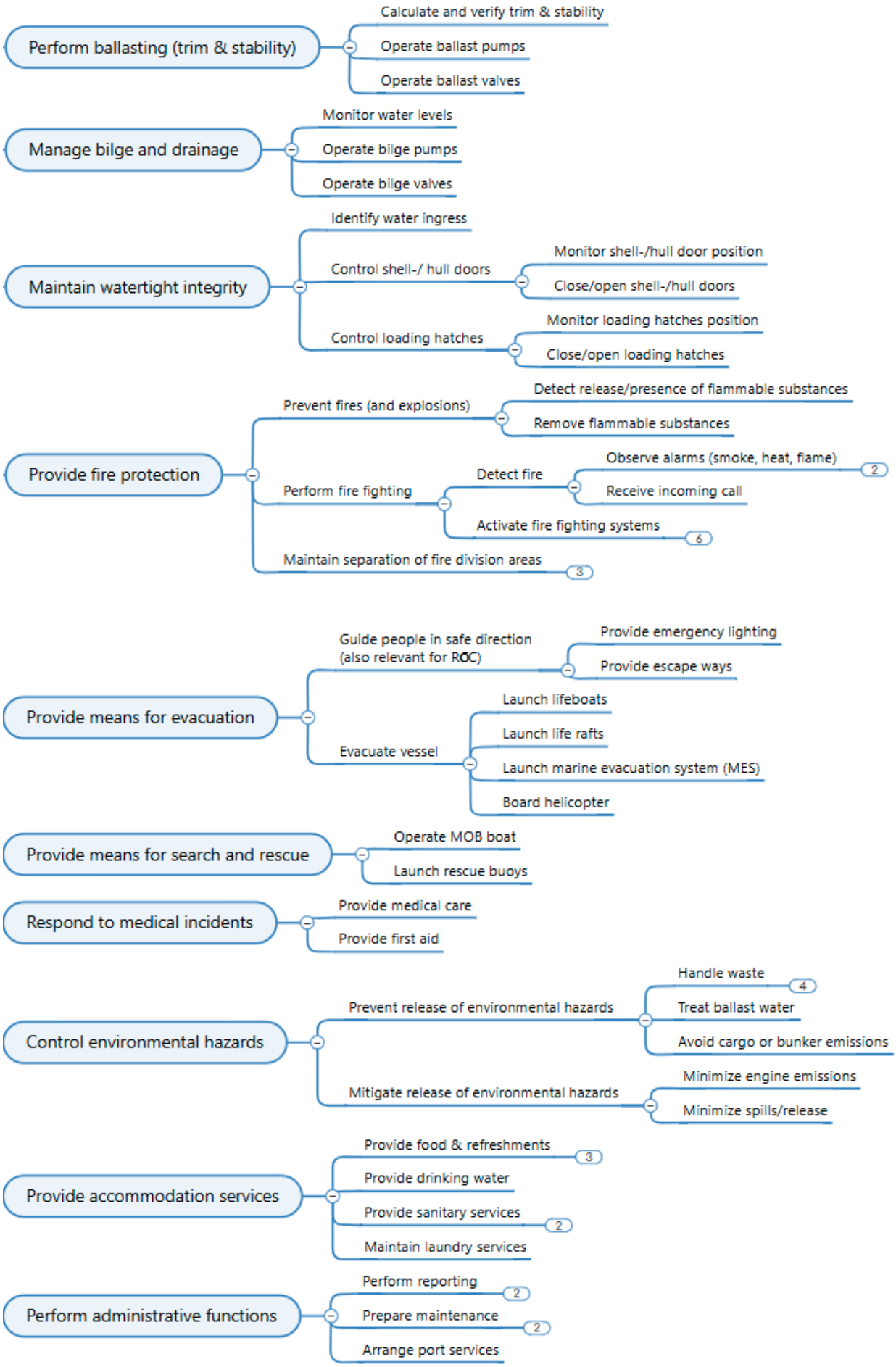

Maintain communication (data, voice/sound, visual signalling)
Communication between ROC and vessel(s), incl. pilot entering vessel
1
Communication between vessels (inside fleet)
Communication internally onboard vessel
PA/loudspeakers
UHF
Phones
Communication between ROC and other parties
8
Communication between vessel and other parties
8


Observe weather conditions
Observe sea conditions
Tide
Current
Waves
Ice
Observe wind conditions
Speed
Direction
Observe visibility
5
Obtain weather forecast


Perform navigation
Perform voyage pre-/re-planning
Plan/download route
4
Plan energy consumption/ fuel efficiency
Plan contingencies
Validate route
Activate route on ECDIS
Verify vessel's seaworthiness
1
Perform route optimization
Observe surroundings
Observe vessel traffic situation
Observe navigational signals
3
Observe water depth/subsurface topography
Observe shoreline/islands
Observe structures
2
Follow planned route
Observe ship motions
3
Provide manoeuvring commands
1
Avoid collisions and grounding
Detect vessels/objects
Classify vessels/objects
Observe vessels/objects movements (heading and speed)
Determine vessels/objects position & relative distance
1
Decide on collision and grounding avoidance strategy
1
Perform manoeuvring
Provide acceleration/deceleration
Control main propulsion
Control propulsion thrust
Provide steering
Control rudder, steering gear
Control manoeuvring thrust
Maintain position
1


Perform towing
Engage in towing (of own vessel)
Request tug
Deploy messenger line
Retrieve towing line
Secure towing line
Manoeuvre during towing
Deploy towing line
Emergency release of towing line
Tow other vessel
To be included
Perform anchoring
Monitor vessel position
Locate suitable anchoring spot
Prepare anchor
Drop anchor
Parameters
Sea depth
Chain speed
Chain length
Chain tension
Secure anchor
Maintain position when anchored
Monitor anchor chain loads
Control anchor chain loads
Optimise anchor conditions
Hoist anchor
Emergency release of anchor
Perform mooring
Order mooring
Prepare/make mooring line available
Deploy mooring line
Fix/secure mooring line
Optimise mooring conditions
Monitor mooring line loads
Control mooring line tension
Retrieve mooring line
Stow away mooring line


Provide electrical power
Charge/receive electrical power (from shore)
Order charging
Prepare for charging
Attach charging cable
Start charging
Monitor charging process
Stop charging
Detach charging cable
Generate electrical power (on vessel)
Generate main & auxiliary power
Generate emergency power
Distribute electrical power (switchboards)
For propulsion
For auxiliary
Store, manage & protect electricity
Perform auxiliary/other functions
Provide heat
Steam (boilers, incinerator)
Exhaust heat (and other sources)
Provide ventilation
Provide cooling
Sea water
Box cooler
Water to air
Provide pneumatic pressure
Provide hydraulic pressure
Provide lubrication
Provide lighting
Deck lights
Cargo hold lights
Lights in accommodation/hallway/rooms
Integrated monitoring & control
Monitor system & component status
Run tests and diagnostics
Perform software updates/configurations
Respond to warnings & alarms
Perform troubleshooting
Log data and events


Perform ballasting (trim & stability)
Calculate and verify trim & stability
Operate ballast pumps
Operate ballast valves
Manage bilge and drainage
Monitor water levels
Operate bilge pumps
Operate bilge valves
Maintain watertight integrity
Identify water ingress
Control shell-/ hull doors
Monitor shell-/hull door position
Close/open shell-/hull doors
Control loading hatches
Monitor loading hatches position
Close/open loading hatches
Provide fire protection
Prevent fires (and explosions)
Detect release/presence of flammable substances
Remove flammable substances
Perform fire fighting
Detect fire
Observe alarms (smoke, heat, flame)
2
Receive incoming call
Activate fire fighting systems
6
Maintain separation of fire division areas
3
Provide means for evacuation
Guide people in safe direction (also relevant for ROC)
Provide emergency lighting
Provide escape ways
Evacuate vessel
Launch lifeboats
Launch life rafts
Launch marine evacuation system (MES)
Board helicopter
Provide means for search and rescue
Operate MOB boat
Launch rescue buoys
Respond to medical incidents
Provide medical care
Provide first aid
Control environmental hazards
Prevent release of environmental hazards
Handle waste
4
Treat ballast water
Avoid cargo or bunker emissions
Mitigate release of environmental hazards
Minimize engine emissions
Minimize spills/release
Provide accommodation services
Provide food & refreshments
3
Provide drinking water
Provide sanitary services
2
Maintain laundry services
Perform administrative functions
Perform reporting
2
Prepare maintenance
2
Arrange port services

Appendix C LIST OF VERBS

Information acquisition	Information analysis	Decision making	Action implementation
Access	Calculate	Command	Acknowledge
Detect	Classify	Conclude	Activate
Hear	Compare	Determine	Alert
Observe	Consider	Generate	Align
Read	Define	Plan	Announce
Receive	Identify	Select	Approve
Record	Integrate		Attach
Registrate	Interpret		Attain
Review	Organize		Brief
Scan	Predict		Close
Sense	Prioritize		Communicate
	Trend		Compute
	Verify		Configure
Action implementation cont.			
Continue	Extinguish	Monitor	Reset
Control	Fasten	Open	Respond
Coordinate	Fill	Operate	Secure
Cycle	Follow	Order	Stabilize
Deactivate	Guard	Perform	Start
Debrief	Illuminate	Position	Steer
Decelerate	Increase	Prepare	Stop
Decrease	Initialize	Pressurize	Stow
Depressurize	Initiate	Prevent	Test
Detach	Inspect	Proceed	Transmit
Deviate	Intercept	Program	Trim
Discharge	Interrogation	Provide	Tune
Eliminate	Isolate	Recover	Turn
Enter	Load	Remove	Unfasten
Evacuate	Maintain	Repeat	Unload
Exit	Manoeuvre	Report	Unsecure
Extend	Modify	Request	Update

Appendix D CAUSAL FACTORS

A modern system may be subject to many different types of failures. Failures can be classified as:

- Random (hardware) failures,
- Systematic failures,
- Systemic failures,
- Operator failures,
- Failures due to environmental causes
- Failures due to deliberate actions.

Note that these categories overlap to some extent, yet they are useful as a guide to identify a wide range of failures that may pose risk.

Random hardware failures are linked to the physical properties of components. The term random is used because the exact moment a specific component will fail is unknown and does not imply that the failure happens arbitrarily. Typical failure rates for a large group of the same component can be predicted through analysis of statistics from field experience, and this makes it possible to perform Quantitative Risk Analysis (QRA) that takes into account the probability of failure for the different components in a system.

The degradation mechanisms that lead to random failures can to some extent be controlled by adjusting how components are designed produced, transported, installed, operated, and maintained. Thus, the failure rates for specific components will partly depend on the quality, operational and maintenance regimes applied. In this regard, it is important to be aware that generic failure rates for specific type of components consider all employed quality regimes equal, which is a simplification that represents an uncertainty in the calculations. Furthermore, it should be noted that the failure rates used in QRA typically excludes the run-in and wear-out periods, and therefore failures experienced in usage inside of these periods may be considered systematic failure events rather than random.

Systematic failure events are the consequence of inadequate work processes and may be introduced at all stages in the system lifecycle. Some examples are incomplete risk analysis, inadequate development of barrier strategies, incomplete requirement specifications, weaknesses in software design, programming errors, quality problems in hardware production, and inadequate planning of maintenance. It is difficult to quantify the probability of systematic failure events as they typically will be present in a system from day one, or introduced through modification, but be hidden until specific circumstances occur. This makes it difficult to compare the risks associated with different systems quantitatively, and necessitates broader risk descriptions if a comparison is to be made.

A **systemic failure** is an event which occurs even if no individual component in the system has failed. This may be caused e.g., by overlooked dependencies among the technical, operational, human, and organisational elements of systems, specifications that are based on inadequate understanding of physical processes, or unexpected inputs for which no specific response has been specified. Increasing system complexity may increase the risk of systemic failures, and this is particularly relevant for systems containing software functions. It can be related to intricate dependencies and feed-back mechanisms among system components leading to nonlinear and unpredictable system behaviour. Lack of knowledge and understanding of interactions in a system increase the risk of systemic failures as it makes it difficult to implement robust barrier strategies to prevent them. Choice of simple solutions with few interacting or interdependent elements may reduce the risk of systemic failures and make systems more robust.

Operator failures occur when an operator fails to perform appropriate actions or performs an inappropriate action. The ability of an operator to perform appropriate actions and avoid inappropriate actions depends on the availability and quality of information to act on, the availability of sufficient time to act, and possession of knowledge of how to act. Therefore, the underlying causes of an operator failure may be systematic or systemic failures that involve technical, operational and organisational elements. In particular, operator failures may be dependent on system designs, operational procedures, training of the operator, and assumptions made in the risk treatment strategy. The latter includes availability of measures that realistically can be used to mitigate the risk under relevant operational conditions.

Failures due to environmental causes are caused by physical processes having negative influence on the control system. Some examples are lightning strike, water ingress, fire, electrostatic discharge from personnel, sensors covered by salt, and electromagnetic interference affecting communications. What is considered the environment depends on the boundaries of the system being analysed. E.g., loss of

cooling in a control room may in some risk analyses be seen as an environmental cause, but not if the cooling system is a part of the system being analysed.

Failures due to deliberate actions may be caused for example by hacking, data viruses, physical sabotage, deliberate jamming of radio signals, GPS spoofing (false signals).

Regarding evaluation of possible mitigations, it should be considered that a systemic failure reflects inadequate identification of relevant requirements. Thus, systemic failure may be seen as a form of systematic failure introduced in the requirement specification phase. Mitigation of a failure scenario caused by inadequate requirements typically requires some level of functional diversity between the control functions affected by the failure and the mitigating measure.

In general, all software failures are systematic or systemic in nature, although the occurrence of the input conditions revealing the weakness in the software may in some cases may be perceived as being random-like in nature. Local detection mechanisms, e.g., range checking and plausibility checks may be used to detect some of these. Other failures can only be detected at higher levels in the system that have a broader overview of the system state and the current operational mode, e.g., by comparing output from different controllers in functionally diverse subsystems, or through operator observation of system behaviour.

It will not always be possible to test a system under all relevant use scenarios, and it may even be that the test scenarios that are feasible to check are not realistic. In addition, for software functions within a system, the number of possible input combinations and possible execution paths typically prevents exhaustive testing even when using a simulated environment. This means that testing typically can only demonstrate the presence of conditions that can lead to failures and not their absence. A cautionary approach is therefore warranted to make systems robust to unforeseen conditions that it may experience. This may include fall-back solutions and use of safety margins considering worst-case scenarios.

It will in many cases not be possible to implement detection for all types of systematic/systemic failures. E.g., incomplete analysis of systems, operations, interfaces, and risks may lead to omissions in specifications evading all detection mechanisms. For safety-critical systems, there must either be an efficient fallback chain, or it must be possible to argue that activities associated with analyses, development, verification, and validation have reduced the likelihood of systematic and systemic failures to a tolerable level.

The latter approach may be challenging, e.g., the number of possible combinations of inputs to the system, and the number of possible sequences of input combinations can make it difficult to know whether specifications are complete. Thus, in practice, one often uses a combination where both a fallback chain and a rigorous development process are used to reduce residual risk to a tolerable level.

Since the effectiveness of mitigation measures varies with the type of cause, it is important to consider all failure categories mentioned at the start of this section when performing risk analysis and developing risk treatment strategies. For example, hardware redundancy in combination with voting may be an efficient mitigation against random hardware failures, but it will not be efficient if the cause is systematic or systemic. Furthermore, the use of functional diverse supporting functions may reduce risks related to systematic failures in those functions, but it may not be efficient against systematic failures in the top-level function. Operator intervention through independent means may be efficient against systematic failures in the top-level function, but additional measures may be necessary if the cause is an operator failure, fire and flooding, or deliberate actions like hacking or sabotage.

Appendix E RBAT ACCIDENT MODEL

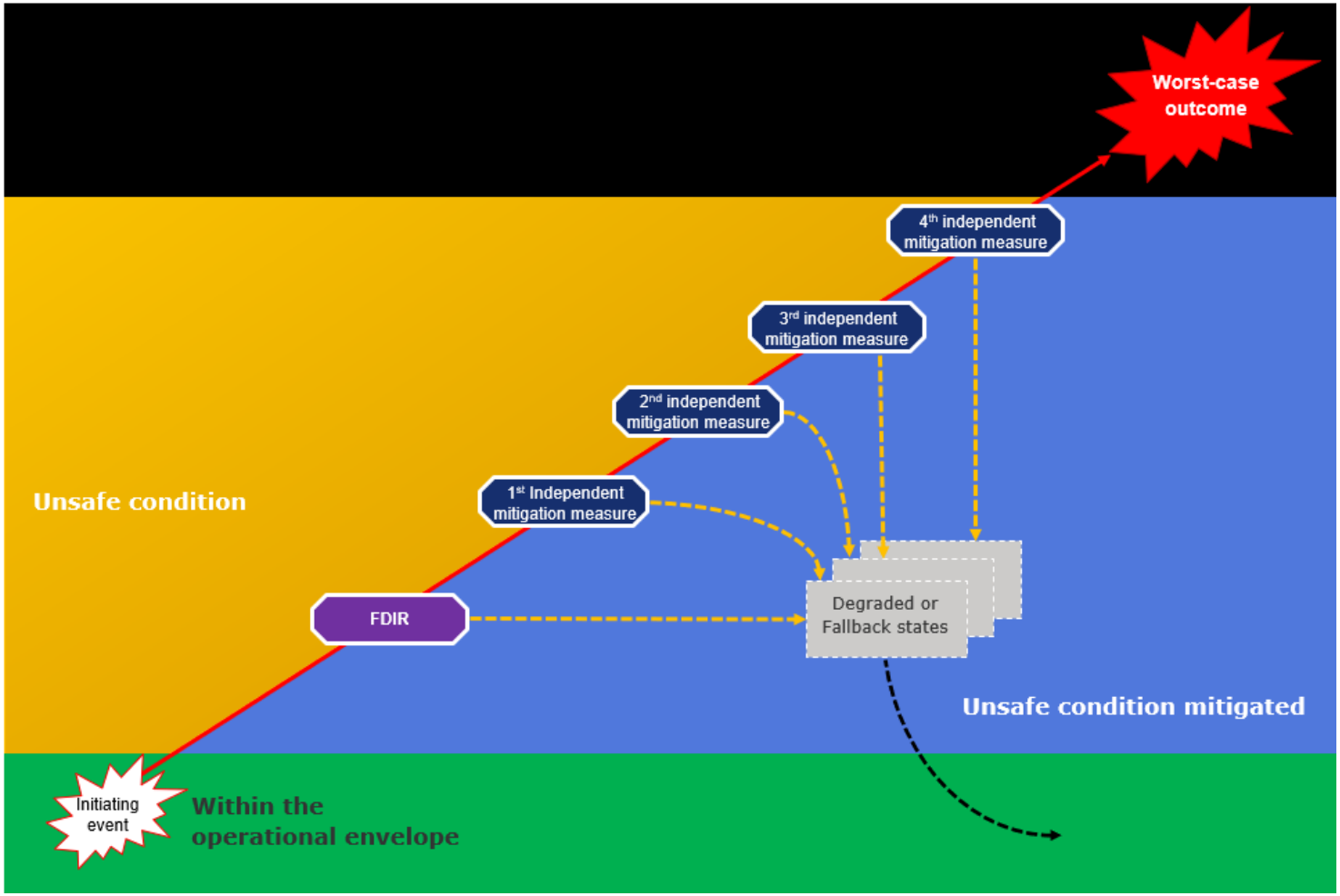


Figure 15 RBAT accident model

European Maritime Safety Agency
Praça Europa 4
1249-206 Lisbon, Portugal
Tel +351 21 1209 200
Fax +351 21 1209 210
emsa.europa.eu