

# Climate Data Management System (CDMS) Functional and Non-functional Requirements

# World Meteorological Organization

Date: 2026-01-06

Version: 0.1.0

Document status: DRAFT

Document location: <https://wmo-im.github.io/et-im/guides/wmo-cdms-requirements-DRAFT.html>

WMO publication location: TBD

Expert Team on Information Management

Standing Committee on Information Management and Technology (SC-IMT)<sup>[1]</sup>

Commission for Observation, Infrastructure and Information Systems (INFCOM)<sup>[2]</sup>

Copyright © 2025 World Meteorological Organization (WMO)

# Table of Contents

|   |    |
|---|----|
| 1. Scope  | 4  |
| 1.1. Introduction                                     | 4  |
| 1.1.1. Purpose and Relationship to WMO-No. 1131       | 4  |
| 1.1.2. System Context and Boundaries                  | 4  |
| 1.2. System Architecture                              | 6  |
| 1.2.1. System Context (Level 1)                       | 6  |
| 1.2.2. Container Architecture (Level 2)               | 6  |
| 1.3. Document Structure and Organization              | 8  |
| 2. Conformance  | 10 |
| 3. References   | 11 |
| 4. Terms and definitions                              | 12 |
| 5. Conventions  | 13 |
| 5.1. Identifiers                                      | 13 |
| 5.1.1. Requirements URIs                              | 13 |
| 5.1.2. Recommendation URIs                            | 13 |
| 5.1.3. Conformance Test URIs                          | 13 |
| 5.2. Use of IETF RFC 2119 Keywords                    | 13 |
| 5.3. Typographical and Notational Conventions         | 14 |
| 6. Functional Requirements                            | 15 |
| 6.1. Requirements Class: Core                         | 15 |
| 6.1.1. Overview                                       | 15 |
| 6.1.2. Justification and Regulatory Basis             | 15 |
| 6.1.3. Requirements                                   | 15 |
| 6.2. Requirements Class: Observations Quality Control | 18 |
| 6.2.1. Overview                                       | 18 |
| 6.2.2. Justification and Regulatory Basis             | 19 |
| 6.2.3. Requirements                                   | 19 |
| 6.3. Requirements Class: Data Ingestion               | 22 |
| 6.3.1. Overview                                       | 22 |
| 6.3.2. Justification and Regulatory Basis             | 22 |
| 6.3.3. Requirements                                   | 22 |
| 6.4. Requirements Class: Data API                     | 26 |
| 6.4.1. Overview                                       | 27 |
| 6.4.2. Justification and Regulatory Basis             | 27 |
| 6.4.3. Requirements                                   | 27 |
| 6.5. Requirements Class: Homogenisation               | 29 |
| 6.5.1. Overview                                       | 29 |
| 6.5.2. Justification and Regulatory Basis             | 30 |
| 6.5.3. Requirements                                   | 30 |
| 6.6. Requirements Class: Data Rescue                  | 31 |
| 6.6.1. Overview                                       | 31 |
| 6.6.2. Justification and Regulatory Basis             | 31 |
| 6.6.3. Requirements                                   | 32 |
| 6.7. Requirements Class: Metadata Management          | 33 |

|  |    |
|--|----|
| 6.7.1. Overview  | 33 |
| 6.7.2. Justification and Regulatory Basis                      | 33 |
| 6.7.3. Requirements  | 33 |
| 6.8. Requirements Class: User Interface                        | 36 |
| 6.8.1. Overview  | 36 |
| 6.8.2. Justification and Regulatory Basis                      | 36 |
| 6.8.3. Requirements  | 36 |
| 6.9. Requirements Class: Climate Products                      | 37 |
| 6.10. Requirements Class: Processing Automation and Scheduling | 37 |
| 6.10.1. Overview   | 37 |
| 6.10.2. Justification and Regulatory Basis                     | 37 |
| 6.10.3. Requirements   | 37 |
| 7. Non-functional Requirements                                 | 38 |
| 7.1. Requirements Class: Performance and Scalability           | 38 |
| 7.1.1. Overview  | 38 |
| 7.1.2. Justification and Regulatory Basis                      | 38 |
| 7.1.3. Requirements  | 38 |
| 7.2. Requirements Class: Reliability and Availability          | 38 |
| 7.2.1. Overview  | 38 |
| 7.2.2. Justification and Regulatory Basis                      | 39 |
| 7.2.3. Requirements  | 39 |
| 7.3. Requirements Class: Security                              | 40 |
| 7.3.1. Overview  | 41 |
| 7.3.2. Justification and Regulatory Basis                      | 41 |
| 7.3.3. Requirements  | 41 |
| 7.4. Requirements Class: Usability                             | 44 |
| 7.4.1. Overview  | 44 |
| 7.4.2. Justification and Regulatory Basis                      | 44 |
| 7.4.3. Requirements  | 44 |
| 7.5. Requirements Class: Maintainability                       | 45 |
| 7.5.1. Overview  | 45 |
| 7.5.2. Justification and Regulatory Basis                      | 45 |
| 7.5.3. Requirements  | 46 |
| 7.6. Requirements Class: Interoperability                      | 47 |
| 7.6.1. Overview  | 48 |
| 7.6.2. Justification and Regulatory Basis                      | 48 |
| 7.6.3. Requirements  | 48 |
| 7.7. Requirements Class: Data Governance                       | 49 |
| 7.7.1. Overview  | 49 |
| 7.7.2. Justification and Regulatory Basis                      | 49 |
| 7.7.3. Requirements  | 49 |

# Chapter 1. Scope

## 1.1. Introduction

### 1.1.1. Purpose and Relationship to WMO-No. 1131

This document serves as a companion specification to the *Climate Data Management System Specifications* (WMO-No. 1131, 2025 Edition), published by the World Meteorological Organization (WMO). WMO-No. 1131 provides a foundational framework for Climate Data Management Systems (CDMS), detailing essential components, governance principles, and operational guidelines using classifications of "Required," "Recommended," or "Optional."

This specification translates the framework described in WMO-No. 1131 into concrete functional and non-functional requirements expressed as testable conformance classes following the OGC Modular Specification approach. This specification aims to:

- **Assist Developers and Integrators:** Provide practical, testable requirements and detailed technical considerations for the design, development, and integration of robust and compliant CDMS components and systems.
- **Empower Users and Assessors:** Establish a comprehensive set of conformance criteria and common understanding to objectively assess the capabilities and compliance of existing or proposed CDMS implementations against WMO-No. 1131 specifications.
- **Enable Interoperability:** Define standardized interfaces, data formats, and protocols to facilitate integration between CDMS implementations and external systems within the climate data ecosystem.

### 1.1.2. System Context and Boundaries

[Figure 1](#) illustrates the architectural landscape and positioning of a Climate Data Management System within the broader weather, climate, and other environmental data value chain (hereafter, climate data for conciseness). Climate data typically flows from field sensors and observing stations through national Data Collection and Processing Systems (DCPS), and subsequently to various downstream applications and services.

**Key downstream systems include:**

- WIS 2.0 nodes for international data exchange
- Operational meteorological systems (Numerical Weather Prediction, data assimilation)
- Climate Data Management Systems for quality control, homogenization, and archival
- Regional Climate Centres for regional climate monitoring and services
- Long-term preservation archives and public-facing data portals

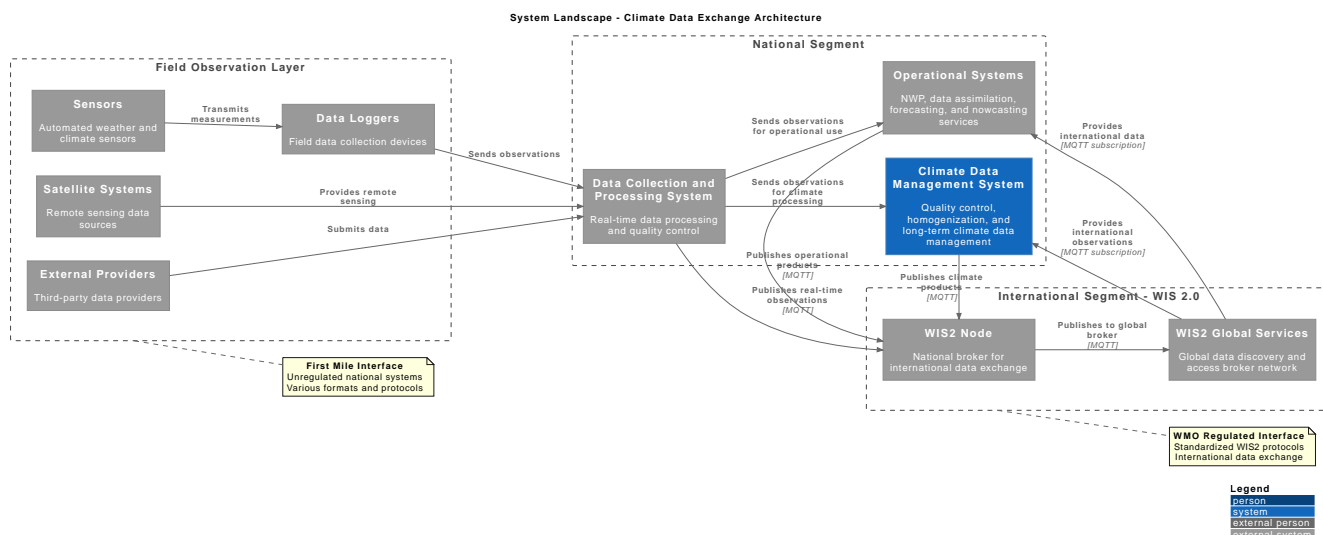


Figure 1. System landscape showing the context of Climate Data Management Systems within the data exchange architecture

### Scope Boundaries:

While some operational implementations may integrate multiple components of the data flow—for example, combining DCPS and CDMS functionality within a single system—this specification focuses exclusively on the Climate Data Management System component and its specific responsibilities.

### The following are explicitly OUT OF SCOPE for this specification:

- Field observation collection (sensors, data loggers, first-mile infrastructure)
- Real-time data collection and processing systems (DCPS)
- Time-critical data exchange via WIS 2.0 for operational meteorology
- Numerical Weather Prediction and forecasting systems
- General-purpose meteorological databases not specifically designed for climate data
- Operational quality control for real-time applications

### The following are IN SCOPE:

- Quality control procedures specific to climate data requirements with emphasis on long-term consistency
- Detection and correction of inhomogeneities in climate time series (homogenization)
- Climate data rescue, including digitization of historical paper records
- Station metadata management with emphasis on long-term consistency and provenance
- Archival and preservation of climate datasets with appropriate documentation
- Publication and distribution of quality-controlled climate data products
- Integration interfaces with WIS 2.0, national networks, and international archives
- Data lineage tracking and provenance management

This specification recognizes that CDMS implementations may exist within national meteorological services (National Segment) and contribute to regional and international climate services (Regional/International Segments) as shown in Figure 1. Requirements are designed to support both contexts while maintaining flexibility for local operational arrangements.

## 1.2. System Architecture

### 1.2.1. System Context (Level 1)

Figure 2 illustrates the system context for a Climate Data Management System, showing the primary users, the CDMS boundary, and its interactions with external systems. The diagram identifies key integration points including:

- **Data sources:** National observation networks and WIS 2.0 Global Services (via national WIS2 Node).
- **Users:** Climate scientists, data managers, and system operators
- **Downstream systems:** Long-term archives, climate data stores, and climate analysis platforms
- **Support systems:** Identity management and operations monitoring

While these external systems and their interactions are shown for context, their internal specifications are outside the scope of this document. This specification focuses exclusively on the requirements for the CDMS component itself and its standardized interfaces with these external systems.

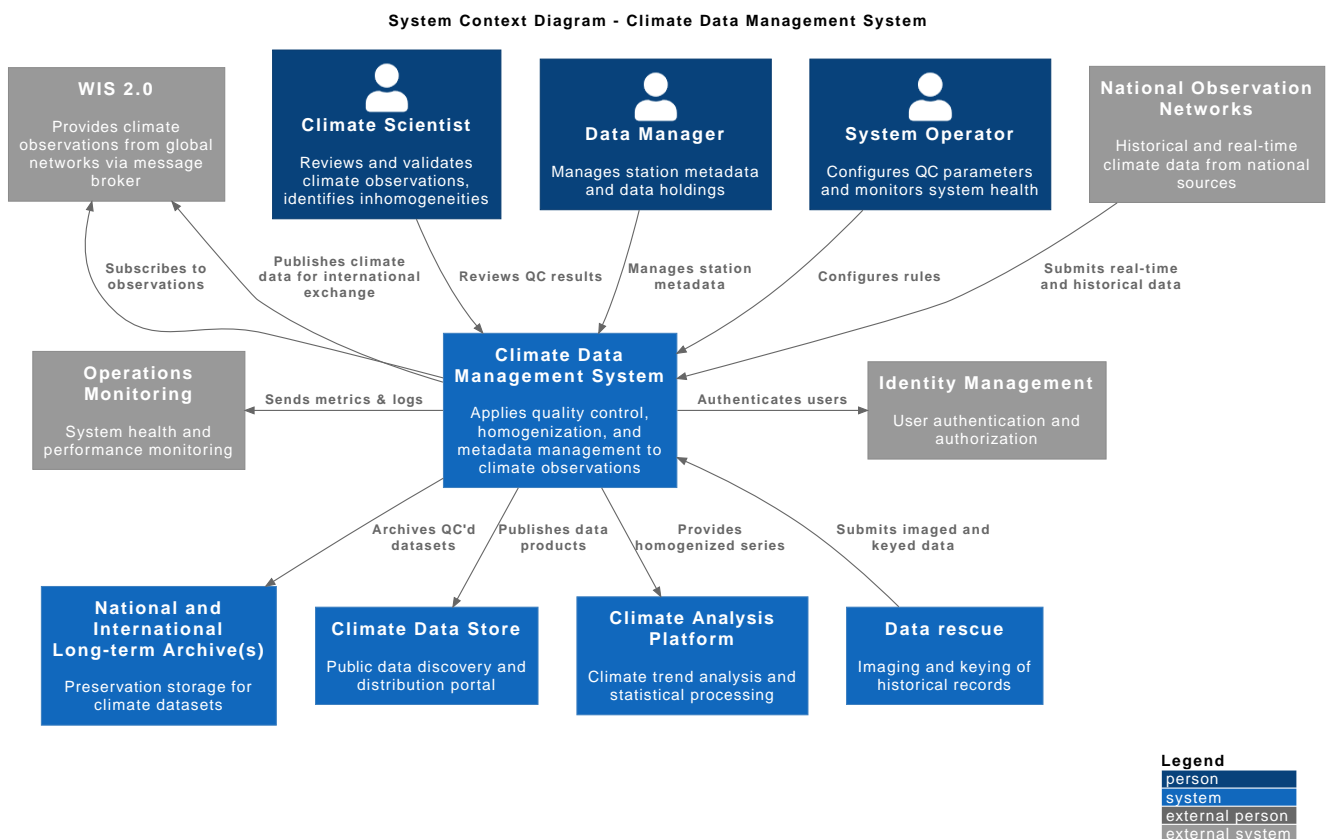


Figure 2. CDMS system context diagram showing users and external system interfaces

DB comment: Do we need to add in DCPS / note that the DCPS would sit between national networks / WIS2 nodes and the CDMS?.

### 1.2.2. Container Architecture (Level 2)

Figure 3 provides a high-level view of the major functional components and data stores within the Climate Data Management System. This diagram is intentionally technology-agnostic, allowing implementers flexibility in architectural and technology choices while conforming to the functional requirements of this specification.

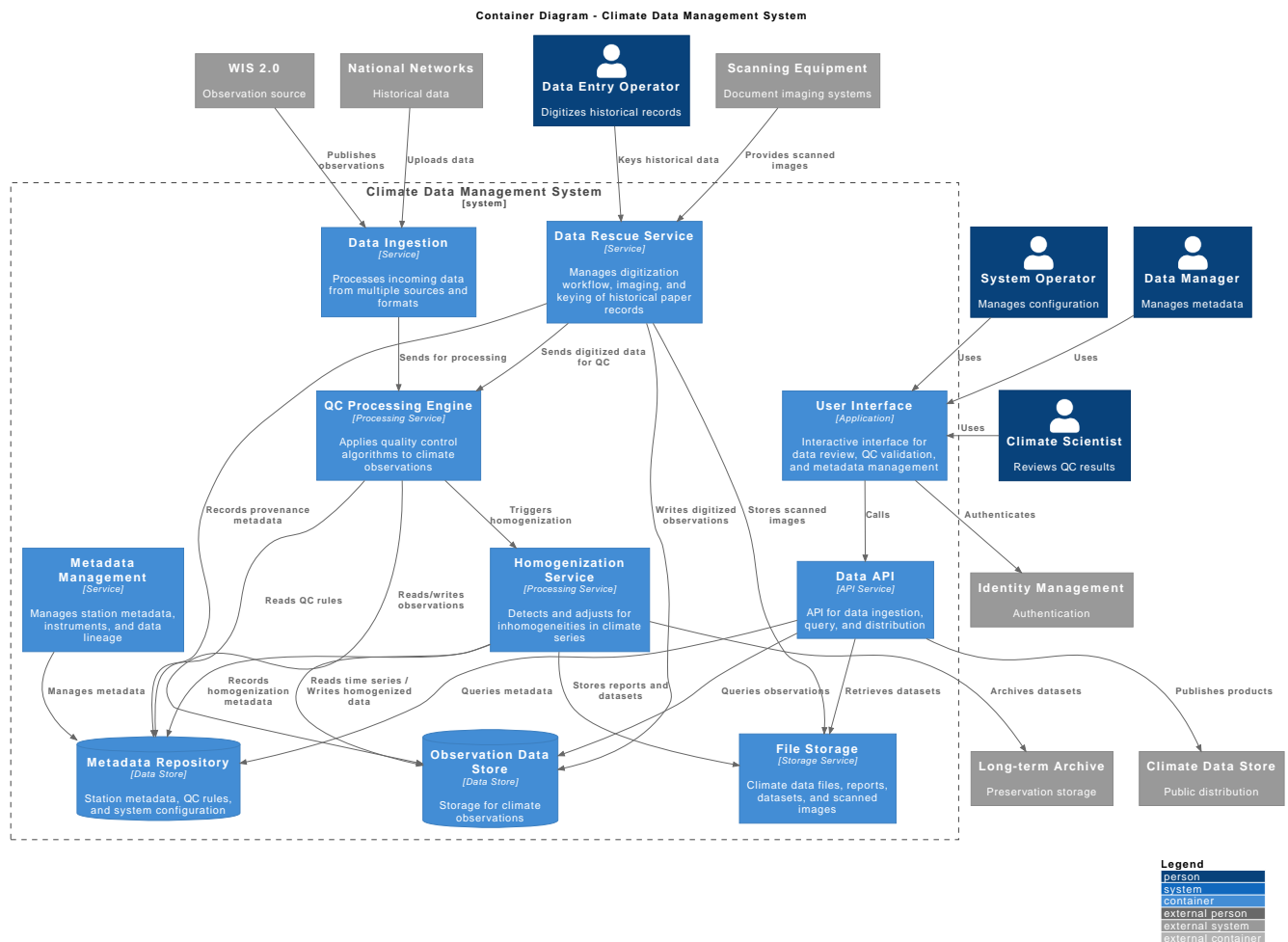


Figure 3. Container diagram showing major functional components and data stores

### Key Containers:

- **User Interface:** Provides interactive capabilities for climate scientists, data managers, and system operators to review quality-controlled data, validate observations, manage station metadata, and configure system parameters. This may be implemented as a web application, desktop application, or a combination of both.
- **Data API:** Exposes standardized interfaces for data ingestion from external sources, data query and retrieval by users and downstream systems, and publication of climate data products. The API should support international standards for climate data exchange including OGC API standards.
- **Data Ingestion:** Processes incoming observations from multiple sources including the national WIS2 Node (receiving both national and international data), national observation networks, and historical data submissions. Handles multiple data formats and protocols, performing initial validation and normalization.
- **QC Processing Engine:** Applies quality control algorithms to climate observations including range checks, temporal consistency checks, spatial consistency checks, and climatological validation. Implements the quality control requirements classes defined in this specification.
- **Homogenization Service:** Performs statistical analysis to detect and adjust for non-climatic discontinuities in observation time series. Produces both the adjusted observation data and comprehensive documentation of the homogenization process including change point detection and adjustment factors.
- **Metadata Management:** Maintains comprehensive station metadata including location information, instrumentation history, observing practices, and data lineage. Supports the traceability requirements essential for climate data and ensures compliance with international metadata standards.



- **Data Rescue Service:** Manages the digitization workflow for historical climate records, including coordination of document imaging, data entry from paper records, and quality assurance of digitized observations. Preserves both the scanned images and the keyed data with full provenance information. Digitized observations flow through the same QC Processing Engine as contemporary observations to ensure consistent quality standards.

#### Data Stores:

- **Observation Data Store:** Persistent storage for climate observations with their associated quality flags and processing history. Implementation may utilize time-series optimized databases, spatial databases, or other suitable technologies that support efficient temporal and spatial queries.
- **Metadata Repository:** Stores station metadata, quality control rules and thresholds, system configuration, homogenization parameters, and data lineage information. Maintains historical records of all metadata changes to support long-term climate analysis.
- **File Storage:** Manages climate-standard file formats (such as NetCDF-CF), quality control reports, homogenization analysis outputs, digitized document images, and archived datasets. Supports versioning and long-term preservation requirements.

#### External System Integration:

The system integrates with external systems at well-defined conformance boundaries where standardized protocols and data formats apply:

- **WIS 2.0:** International data exchange using MQTT messaging and standardized data formats for both consuming international observations and publishing national climate products
- **National Observation Networks:** Historical and real-time climate data submission via file transfer or API
- **Long-term Preservation Archives:** Climate dataset archival using standard formats (e.g., NetCDF-CF, GRIB2) and protocols (OPeNDAP, OGC APIs)
- **Climate Data Stores:** Public data discovery and distribution via OGC API - EDR, STAC catalogs, or equivalent standards
- **Identity Management Systems:** Enterprise authentication and authorization using standard protocols (LDAP, OAuth 2.0, SAML)

This architectural decomposition supports the modular requirements structure of this specification, where different containers may be subject to different requirements classes based on their functional responsibilities.

## 1.3. Document Structure and Organization

To facilitate clear understanding and practical application, this document is structured into the following key sections:

- **Functional Requirements:** Details what a CDMS shall do, providing specific, granular requirements for core functionalities such as data ingestion and validation, transformation and storage, metadata management, quality control, data rescue, processing, and user interface interactions. These requirements operationalize the high-level functional components outlined in WMO-No. 1131.
- **Non-Functional Requirements:** Defines how well a CDMS shall perform, covering essential attributes like performance, scalability, reliability, maintainability, usability, interoperability, data governance, and comprehensive security measures. These requirements ensure that a CDMS not only fulfills its functional duties but also operates effectively, securely, and sustainably in various environments.
- **Test Specifications:** Introduces the Abstract Test Suite (ATS), which provides high-level,

implementation-agnostic test cases directly linked to the functional and non-functional requirements. It serves as a blueprint for verifying compliance and assessing the capabilities of CDMS implementations against the criteria defined herein and by WMO-No. 1131.

- **Appendix:** Provides supplementary information, including a glossary of terms, a list of acronyms used throughout the document, references to related standards and specifications, and any other relevant supporting material that aids in the understanding and application of these specifications.

Each requirement within the functional and non-functional sections is designed to be clear, measurable, and directly applicable, serving as a valuable reference for both system development and compliance assessment activities in the context of WMO-No. 1131.

[1] <https://community.wmo.int/governance/commission-membership/commission-observation-infrastructures-and-information-systems-infcom/commission-infrastructure-officers/infcom-management-group/standing-committee-information-management-and-technology-sc-imt>

[2] <https://community.wmo.int/governance/commission-membership/infcom>

# Chapter 2. Conformance

This specification defines the following conformance classes that align with the major functional containers shown in [Figure 3](#):

- **Core Data Management** (Required): Basic observation storage, retrieval, and metadata management capabilities
- **Quality Control** (Required): Climate-specific QC procedures including range, temporal, spatial, and climatological checks
- **Data Ingestion** (Required): Multi-format data import, validation, and normalization
- **Data API** (Required): Standards-based data query, retrieval, and distribution interfaces
- **Homogenization** (Recommended): Statistical detection and adjustment of inhomogeneities in climate time series
- **Data Rescue** (Optional): Digitization workflow for historical paper records including imaging and data entry
- **Metadata Management** (Required): Comprehensive station metadata with full history and provenance tracking

Each conformance class is described in detail in subsequent sections, with corresponding test specifications provided in [\[test-specifications\]](#).

## NOTE

The classification of conformance classes as Requirement, Recommendation and Permission aligns with the Required, Recommended and Optional classifications respectively from WMO-No. 1131. Required classes must be implemented for a system to be considered conformant. Recommended classes should be implemented where operationally feasible. Optional classes may be implemented based on specific organizational needs.

## Chapter 3. References

The following documents are referred to in this specification:

<To follow>

# Chapter 4. Terms and definitions

For the purposes of this document, the terms and definitions given in WMO-No. 1131 and the following apply:

<To Follow>

# Chapter 5. Conventions

For the purposes of this document, the conventions described in this clause apply.

## 5.1. Identifiers

The normative provisions (Requirements and Conformance Tests) in this module are denoted by URIs which follow a hierarchical structure to ensure unique identification and traceability.

The base URI for this specification is: `spec-name/[version]/[module-name]`

All requirements and conformance tests that appear in this document are denoted by **partial URIs** which are relative to this base.

The structure for partial URIs is defined as follows:

### 5.1.1. Requirements URIs

Requirements are identified by a URI fragment that includes the Requirements Class name and a unique identifier for the specific requirement.

- **Format:** `/req/[requirements-class-name]/[requirement-id]`
- **Example:** `/req/quality-control/range-check`

### 5.1.2. Recommendation URIs

Recommendations are identified by a URI fragment that includes the Recommendations Class name and a unique identifier for the specific requirement.

- **Format:** `/rec/[recommendation-class-name]/[recommendation-id]`
- **Example:** `/rec/homogenisation/detection`

DB comment: need to check the syntax.

### 5.1.3. Conformance Test URIs

Conformance Tests are identified by a URI fragment that includes the Conformance Class name and a unique identifier for the test.

- **Format:** `/conf/[conformance-class-name]/[test-id]`
- **Example:** `/conf/quality-control/range-check`

## 5.2. Use of IETF RFC 2119 Keywords

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in IETF RFC 2119.

The keywords indicate the level of strictness required for compliance with a normative provision:

- **MUST/REQUIRED/SHALL:** These keywords mean that the definition is an **absolute requirement** of the specification.
- **MUST NOT/SHALL NOT:** These phrases mean that the definition is an **absolute prohibition** of the specification.
- **SHOULD/RECOMMENDED:** These keywords mean that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
- **SHOULD NOT/NOT RECOMMENDED:** These phrases mean that there may exist valid reasons in particular circumstances when the behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before proceeding.
- **MAY/OPTIONAL:** These keywords mean that an item is truly optional.

## 5.3. Typographical and Notational Conventions

The following typographical conventions are used throughout this document to distinguish different types of content:

- **Monospace Text:** Used for source code examples, file names, directory names, machine-readable data structures (e.g., JSON keys, XML tags), URIs, and literal text that appears in a program or command line.
- **Italics:** Used for titles of documents, variable placeholders within code or URI templates, and for emphasis.
- **[Bracketed] Text:** Used for placeholders in this template (e.g., `[spec-name]`, `[module-name]`) that must be replaced with specific content during implementation.
- **[NOTE], [TIP], [IMPORTANT]:** Used to highlight informative text blocks that provide context, advice, or warnings that are **not** part of the normative requirements.

# Chapter 6. Functional Requirements

## 6.1. Requirements Class: Core

|                    |                 |
|--------------------|-----------------|
| Requirements class | /req/core       |
| Target type        | Software system |
| Dependency         | None            |
| Conformance test   | /conf/core      |

### 6.1.1. Overview

The Core requirements class defines the requirements for basic observation storage, retrieval, and metadata management capabilities. This provides the foundational data management capabilities for a Climate Data Management System and support for long-term climate data stewardship.

### 6.1.2. Justification and Regulatory Basis

WMO-No. 1131 identifies Core Data Management as a Required component across all CDMS implementations.

The Manual on High-quality Global Data Management Framework for Climate (WMO-No. 1238) requires Members to "ensure that Entities establish and document data management capabilities and practices for their climate data" (3.1.1). The framework's principles mandate "preservation and management of all data and metadata that are used for, or may potentially be useful for, climate studies" (1.1.1(3)). Core capabilities support the FAIR data principles requiring data to be Findable, Accessible, Interoperable and Reusable (3.2).

### 6.1.3. Requirements

#### 6.1.3.1. Observation Data Model

| Requirement 1 |  |
|---------------|--|
| Identifier    | /req/core/data-model   |
| Statement     | <p>The system SHALL implement a data model capable of storing environmental data from across the Earth system domains and disciplines. The data model SHALL include:</p> <ul style="list-style-type: none"><li>• Station and sensor metadata</li><li>• Observed element/variable</li><li>• Observation value/result and associated units</li><li>• Quality flag status</li><li>• Observation timestamp, including time zone support</li><li>• Geographic location of the observation</li><li>• Provenance metadata</li></ul> |

**NOTE** | A draft data model <link to wccdm-obs>



#### 6.1.3.2. Reference data

- To follow

#### 6.1.3.3. Observation Storage

| Requirement 2 |  |
|---------------|--|
| Identifier    | /req/core/observation-storage  |
| Statement     | The system SHALL provide persistent storage for observation records. |

#### 6.1.3.4. Observation Retrieval

| Requirement 3 |   |
|---------------|---|
| Identifier    | /req/core/observation-retrieval   |
| Statement     | <p>The system SHALL support retrieval of observations filtered by:</p> <ul style="list-style-type: none"><li>• Station identifier or group of stations</li><li>• Geographic bounding box or polygon</li><li>• Time period (start and end timestamps)</li><li>• Observed element/variable</li><li>• Quality control status</li></ul> |

#### 6.1.3.5. Station and Sensor Metadata Storage

| Requirement 4 |   |
|---------------|---|
| Identifier    | /req/core/metadata-storage  |
| Statement     | The system SHALL provide persistent storage for station and sensor metadata as specified in /req/metadata-management/wigos. |

#### 6.1.3.6. Station and Sensor Metadata Retrieval

| Requirement 5 |  |
|---------------|--|
| Identifier    | /req/core/metadata-retrieval   |
| Statement     | <p>The system SHALL support retrieval of station and sensor metadata filtered by:</p> <ul style="list-style-type: none"><li>• Station identifier (WIGOS Station Identifier, national identifier, or legacy identifier)</li><li>• Geographic extent (bounding box, polygon, or proximity search)</li><li>• Administrative boundaries (country, region, organization)</li><li>• Station classification or network affiliation</li><li>• Observed element capability</li><li>• Operational status (active, historical, planned)</li><li>• Temporal validity (metadata state at a specified point in time)</li></ul> |

#### 6.1.3.7. Station and Sensor Metadata Versioning

| Requirement 6 |  |
|---------------|--|
| Identifier    | /req/core/metadata-versioning  |
| Statement     | <p>The system SHALL maintain temporal versioning of all station and sensor metadata such that:</p> <ul style="list-style-type: none"><li>• All changes to metadata are recorded with effective date and time</li><li>• Historical states of metadata can be reconstructed for any point in time</li><li>• The system can return metadata valid at the time of any given observation</li><li>• Metadata changes do not overwrite or destroy previous versions</li></ul> |

#### 6.1.3.8. Provenance Metadata Storage

| Requirement 7 |   |
|---------------|---|
| Identifier    | /req/core/provenance-storage  |
| Statement     | <p>The system SHALL store provenance metadata for all observation data documenting the complete data lineage including:</p> <ul style="list-style-type: none"><li>• Original data source identification (network, station, transmission method)</li><li>• Acquisition timestamp (when data entered the system)</li><li>• Ingestion processing details (format conversion, unit normalization, coordinate transformation)</li><li>• Quality control processing history (quality control procedures applied, flags assigned, timestamps)</li><li>• Manual interventions (edits, corrections, deletions with operator identification and justification)</li><li>• Derived data lineage (source observations, algorithms, parameters used in derivation)</li><li>• Homogenization adjustments (break points detected, adjustment factors applied, reference series used)</li><li>• Data rescue provenance (source document reference, image identifier, keying method, verification status)</li></ul> |

#### 6.1.3.9. Provenance Metadata Retrieval

| Requirement 8 |                                |
|---------------|--------------------------------|
| Identifier    | /req/core/provenance-retrieval |

|           |   |
|-----------|---|
| Statement | <p>The system SHALL support retrieval of provenance metadata including:</p> <ul style="list-style-type: none"> <li>• Complete processing history for any observation or set of observations</li> <li>• Identification of all transformations applied to data from acquisition to current state</li> <li>• Audit trail queries by operator, time period, or processing type</li> <li>• Traceability from derived products back to original source observations</li> <li>• Export of provenance information in machine-readable format (e.g., W3C PROV-O, ISO 19115 lineage)</li> </ul> |
|-----------|---|

#### 6.1.3.10. Provenance Metadata Immutability

| Requirement 9 |   |
|---------------|---|
| Identifier    | /req/core/provenance-immutability   |
| Statement     | <p>The system SHALL ensure provenance records are immutable such that:</p> <ul style="list-style-type: none"> <li>• Provenance entries cannot be modified or deleted after creation</li> <li>• Original (raw) observation values are preserved even when corrections are applied</li> <li>• A complete audit trail exists for all data modifications</li> </ul> |

#### 6.1.3.11. Batch-processing

| Requirement 10 |  |
|----------------|--|
| Identifier     | /req/core/batch-processing   |
| Statement      | <p>The system SHALL support batch operations for efficiency:</p> <ul style="list-style-type: none"> <li>• Bulk station metadata updates via file upload</li> <li>• Batch quality control flag modifications with justification</li> <li>• Bulk data export for multiple stations and time ranges</li> <li>• Progress indicators for long-running operations</li> <li>• Ability to cancel or pause long-running batch operations</li> </ul> |

## 6.2. Requirements Class: Observations Quality Control

|                    |                       |
|--------------------|-----------------------|
| Requirements class | /req/quality-control  |
| Target type        | Software system       |
| Dependency         | /req/core             |
| Conformance test   | /conf/quality-control |

### 6.2.1. Overview

The Observations Quality Control requirements class defines requirements to ensure observation data meets climate quality standards through systematic automated checks that detect erroneous, suspicious, or

inconsistent values while maintaining complete audit trails of all quality control decisions. Quality control is essential for establishing trustworthy climate records suitable for long-term trend analysis and climate monitoring.

## 6.2.2. Justification and Regulatory Basis

WMO-No. 1131 classifies Observations Quality Control as Required (Section 5.3), emphasizing that "all data have been checked for accuracy, and that a record is maintained of what checks were performed, the outcome of the checks and when the checks took place."

WMO-No. 1238 mandates that "Entities apply quality control to all climate datasets under their management and record the result of that quality control in the metadata" (3.3.2.1).

### NOTE

The *Guidelines on Surface Station Data Quality Control and Quality Assurance for Climate Applications* (WMO-No. 1269) provide specific methodologies referenced in both documents.

## 6.2.3. Requirements

### 6.2.3.1. Range Checks

| Requirement 11 |  |
|----------------|--|
| Identifier     | /req/quality-control/range-checks  |
| Statement      | <p>The system SHALL perform range validation checks, including:</p> <ul style="list-style-type: none"> <li>• Gross error limits (physically impossible values)</li> <li>• Climatological based range checks</li> <li>• Sensor-based range checks (measurements within observable range for sensor)</li> <li>• Configurable thresholds per element and station</li> </ul> |

### 6.2.3.2. Temporal Consistency Checks

| Requirement 12 |  |
|----------------|--|
| Identifier     | /req/quality-control/temporal-checks   |
| Statement      | <p>The system SHALL perform temporal consistency checks, including:</p> <ul style="list-style-type: none"> <li>• Flatline checks - detection of repeated identical values</li> <li>• Rate of change checks - validation of temporal gradients</li> <li>• Spike checks - detection of anomalous rapid changes followed by rapid reversals</li> <li>• Internal consistency - validation of relationships between related elements (e.g., <math>T_{\min} \rightarrow T_{\text{avg}} \rightarrow T_{\max}</math>)</li> </ul> |

### 6.2.3.3. Spatial Consistency Checks

| Requirement 13 |                                     |
|----------------|-------------------------------------|
| Identifier     | /req/quality-control/spatial-checks |

|           |  |
|-----------|--|
| Statement | <p>The system SHALL perform spatial consistency checks including:</p> <ul style="list-style-type: none"> <li>• Comparison with neighbouring stations</li> <li>• Spatial regression tests</li> <li>• Buddy checks using configurable distance and elevation thresholds</li> </ul> |
|-----------|--|

#### 6.2.3.4. Internal Consistency Checks

| Requirement 14 |  |
|----------------|--|
| Identifier     | /req/quality-control/internal-consistency  |
| Statement      | The system SHALL perform internal consistency checks to validate physical relationships between simultaneously observed elements as specified in WMO-No. 1269. |

| Requirement 15 |  |
|----------------|--|
| Identifier     | /req/quality-control/temperature-relationships   |
| Statement      | <p>The following checks SHALL be applied</p> <ul style="list-style-type: none"> <li>• Wet-bulb temperature (<math>T_w</math>) SHALL be less than or equal to dry-bulb temperature (<math>T_d</math>): <math>T_w \leq T_d</math></li> <li>• Dew-point temperature (<math>T_{dp}</math>) SHALL be less than or equal to dry-bulb temperature: <math>T_{dp} \leq T_d</math></li> <li>• Dew-point temperature SHALL be less than or equal to wet-bulb temperature: <math>T_{dp} \leq T_w</math></li> <li>• Minimum temperature (<math>T_{min}</math>) SHALL be less than or equal to maximum temperature (<math>T_{max}</math>): <math>T_{min} \leq T_{max}</math></li> <li>• Mean temperature SHALL fall within or near the range bounded by <math>T_{min}</math> and <math>T_{max}</math></li> </ul> |

| Requirement 16 |  |
|----------------|--|
| Identifier     | /req/quality-control/humidity-relationships  |
| Statement      | <p>The following checks SHALL be applied</p> <ul style="list-style-type: none"> <li>• When relative humidity equals 100%, dry-bulb, wet-bulb, and dew-point temperatures SHALL be equal within measurement uncertainty</li> <li>• Relative humidity SHALL be between 0% and 100% inclusive</li> <li>• Vapour pressure SHALL not exceed saturation vapour pressure at the observed temperature</li> </ul> |

| Requirement 17 |   |
|----------------|---|
| Identifier     | /req/quality-control/pressure-relationships   |
| Statement      | <p>The following checks SHALL be applied</p> <ul style="list-style-type: none"> <li>• Station-level pressure and mean sea-level pressure SHALL be consistent given station elevation</li> <li>• Pressure tendency SHALL be consistent with consecutive pressure observations</li> </ul> |

| Requirement 18 |   |
|----------------|---|
| Identifier     | /req/quality-control/precipitation-relationships  |
| Statement      | <p>The following checks SHALL be applied</p> <ul style="list-style-type: none"> <li>• When precipitation amount is greater than zero, present weather codes SHALL indicate precipitation occurrence</li> <li>• Precipitation type (rain, snow, mixed) SHALL be consistent with observed temperature</li> <li>• Snow depth changes SHALL be consistent with reported snowfall and temperature</li> </ul> |

| Requirement 19 |   |
|----------------|---|
| Identifier     | /req/quality-control/wind-relationships   |
| Statement      | <p>The following checks SHALL be applied</p> <ul style="list-style-type: none"> <li>• Wind gust speed SHALL be greater than or equal to mean wind speed</li> <li>• Maximum wind speed SHALL be greater than or equal to mean wind speed for the period</li> </ul> |

| Requirement 20 |  |
|----------------|--|
| Identifier     | /req/quality-control/sunshine-relationships  |
| Statement      | <p>The following checks SHALL be applied</p> <ul style="list-style-type: none"> <li>• Sunshine duration SHALL not exceed the astronomically possible duration for the date and location</li> <li>• Global radiation SHALL not exceed the extraterrestrial radiation for the date and location</li> <li>• Diffuse radiation SHALL not exceed global radiation</li> <li>• When sunshine duration is zero, direct solar radiation SHALL be zero or near-zero</li> </ul> |

| Requirement 21 |  |
|----------------|--|
| Identifier     | /req/quality-control/cloud-relationships   |
| Statement      | <p>The following checks SHALL be applied</p> <ul style="list-style-type: none"> <li>• When visibility is reported as very low (e.g., fog conditions), present weather codes SHALL be consistent</li> <li>• Cloud base height observations SHALL be consistent with reported cloud types and amounts</li> </ul> |

#### 6.2.3.5. Quality Flag Management

| Requirement 22 |                            |
|----------------|----------------------------|
| Identifier     | /req/quality-control/flags |

|           |   |
|-----------|---|
| Statement | <p>The system SHALL maintain quality flags and:</p> <ul style="list-style-type: none"> <li>• Support multiple quality control levels (automatic, manual, final)</li> <li>• Preserve quality control history and audit trail</li> <li>• Allow manual override with justification</li> <li>• Record which specific quality control check(s) triggered each flag and the reason for failure</li> </ul> |
|-----------|---|

#### 6.2.3.6. Quality Control Configuration Management

| Requirement 23 |  |
|----------------|--|
| Identifier     | /req/quality-control/configuration   |
| Statement      | <p>The system SHALL support configuration of quality control parameters, including:</p> <ul style="list-style-type: none"> <li>• Station-specific and element-specific thresholds</li> <li>• Seasonal variation of thresholds where climatologically appropriate</li> <li>• Enabling/disabling of individual checks per station or element</li> <li>• Version control of quality control rule configurations with effective dates</li> </ul> |

## 6.3. Requirements Class: Data Ingestion

|                    |                      |
|--------------------|----------------------|
| Requirements class | /req/data-ingestion  |
| Target type        | Software system      |
| Dependency         | /req/core            |
| Conformance test   | /conf/data-ingestion |

### 6.3.1. Overview

The Data Ingestion requirements class defines requirements for the reliable ingestion of climate observation data from diverse sources and formats, ensuring data quality through validation and normalization while maintaining complete provenance of source data characteristics. Data ingestion serves as the entry point for observations into the climate record, requiring careful handling to preserve data integrity.

### 6.3.2. Justification and Regulatory Basis

WMO-No. 1131 identifies Data Ingest as a Required capability (Section 5.1.1), stating systems must handle "WMO messages, vector, raster array, ASCII and other formats" with "business rules, status logging, automated self-recovery and transformation."

WMO-No. 1238 requires Members ensure "Entities encode data, and associated metadata, intended for international exchange following the standards specified in the Manual on Codes (WMO-No. 306)" (3.2.2.1).

### 6.3.3. Requirements

### 6.3.3.1. Automated Data Collection

| Requirement 24 |  |
|----------------|--|
| Identifier     | /req/data-ingestion/automated-collection   |
| Statement      |  |
| A              | <p>The system SHALL support automated data collection from external sources including:</p> <ul style="list-style-type: none"><li>• <b>Scheduled polling:</b> Configurable polling of external data sources at defined intervals (hourly, daily, monthly)</li><li>• <b>Real-time subscription:</b> MQTT-based subscription to WIS 2.0 message brokers for international data exchange</li><li>• <b>File system monitoring:</b> Automatic detection and ingestion of new files in monitored directories</li><li>• <b>Protocol support:</b> FTP, SFTP, HTTP/HTTPS for automated file retrieval</li><li>• <b>API integration:</b> RESTful API calls to external observation systems with authentication management</li></ul> |
| B              | <p>For each automated collection source, the system SHALL maintain:</p> <ul style="list-style-type: none"><li>• Collection schedule or subscription configuration</li><li>• Connection credentials and authentication parameters (stored securely)</li><li>• Last successful collection timestamp</li><li>• Collection attempt history and status</li><li>• Error logs for failed collection attempts</li></ul>  |
| C              | <p>The system SHALL implement resilient collection mechanisms:</p> <ul style="list-style-type: none"><li>• Retry logic with configurable retry count and backoff intervals</li><li>• Detection of duplicate data from multiple collection attempts</li><li>• Graceful handling of temporarily unavailable sources</li><li>• Alert generation on persistent collection failures</li><li>• Automatic recovery and resumption after system restart</li></ul>  |

#### NOTE

Automated collection sources should be configurable without system restart. The system should support pausing and resuming collection from specific sources for maintenance purposes.

### 6.3.3.2. Collection Monitoring

| Requirement 25 |   |
|----------------|---|
| Identifier     | /req/data-ingestion/collection-monitoring |



|           |  |
|-----------|--|
| Statement | <p>The system SHALL provide monitoring capabilities for automated data collection:</p> <ul style="list-style-type: none"> <li>• Dashboard showing status of all configured collection sources</li> <li>• Statistics on collection success/failure rates per source</li> <li>• Data volume metrics (records/observations collected per time period)</li> <li>• Latency metrics (time between observation and ingestion)</li> <li>• Alert generation when expected data does not arrive within defined time windows</li> <li>• Collection log export for auditing and troubleshooting</li> </ul> |
|-----------|--|

#### 6.3.3.3. Manual Data Acquisition

| Requirement 26 |   |
|----------------|---|
| Identifier     | /req/data-ingestion/manual-acquisition  |
| Statement      | <p>The system SHALL support manual data submission including:</p> <ul style="list-style-type: none"> <li>• Web-based file upload interface with drag-and-drop support</li> <li>• Batch upload of multiple files</li> <li>• Upload of compressed archives (ZIP, TAR.GZ)</li> <li>• Manual submission via API with authentication</li> <li>• Upload progress indication for large files</li> <li>• Validation feedback during or immediately after upload</li> <li>• Confirmation of successful ingestion with record counts</li> </ul> |

#### 6.3.3.4. Format Support

| Requirement 27 |   |
|----------------|---|
| Identifier     | /req/data-ingestion/formats   |
| Statement      | <p>The system SHALL support the ingestion of climate observation data in the following formats:</p> <ul style="list-style-type: none"> <li>• <b>CSV and delimited text files:</b> With configurable field mapping, delimiter specification (comma, tab, semicolon, pipe), header row handling, and support for common date/time formats</li> <li>• <b>BUFR (Binary Universal Form for the Representation of meteorological data):</b> WMO standard binary format as specified in WMO Manual on Codes (WMO-No. 306), Volume I.2</li> <li>• <b>WMO text-based formats and code forms:</b> Including FM-12 SYNOP and other code forms as specified in WMO Manual on Codes</li> </ul> |

| Recommendation 1 |                             |
|------------------|-----------------------------|
| Identifier       | /rec/data-ingestion/formats |

|           |   |
|-----------|---|
| Statement | <p>The system <b>SHOULD</b> support the following additional formats:</p> <ul style="list-style-type: none"> <li>• <b>NetCDF</b> following CF (Climate and Forecast) conventions version 1.6 or later</li> <li>• <b>JSON</b> for observation data exchange</li> <li>• <b>XML</b> formats used in national or regional climate networks</li> </ul> |
|-----------|---|

#### NOTE

Support for additional historical or legacy formats may be required based on data rescue activities and national archive requirements. Implementers should document all supported formats and any format-specific constraints in their conformance documentation.

### 6.3.3.5. Format Validation

| Requirement 28 |  |
|----------------|--|
| Identifier     | /req/data-ingestion/validation   |
| Statement      |  |
| A              | <p>The system <b>SHALL</b> validate ingested data for:</p> <ul style="list-style-type: none"> <li>• Format compliance</li> <li>• Required field presence (station ID, timestamp, observed element, value)</li> <li>• Data type correctness (numeric values, valid date/time formats)</li> <li>• Referential integrity (station IDs exist in metadata repository)</li> <li>• Value range plausibility (preliminary sanity checks before full QC)</li> </ul> |
| B              | <p>Validation failures <b>SHALL</b> result in:</p> <ul style="list-style-type: none"> <li>• Rejection of the invalid record or file</li> <li>• Generation of detailed error reports identifying specific validation failures</li> <li>• Logging of validation errors for system monitoring and diagnostics</li> </ul>  |

### 6.3.3.6. Data Normalisation

| Requirement 29 |   |
|----------------|---|
| Identifier     | /req/data-ingestion/normalization   |
| Statement      |   |
| A              | <p>The system <b>SHALL</b> normalize ingested data from all supported formats into a common internal representation that includes:</p> <ul style="list-style-type: none"> <li>• Observation value with standardized unit of measure</li> <li>• Observation timestamp normalized to UTC</li> <li>• Station identifier (preferentially WIGOS Station Identifier)</li> <li>• Observed element code using WMO BUFR/GRIB parameter codes or equivalent</li> <li>• Source format identifier</li> <li>• Ingestion timestamp and processing metadata</li> </ul> |

|   |  |
|---|--|
| B | <p>The system SHALL perform unit conversions where necessary, maintaining:</p> <ul style="list-style-type: none"> <li>• Original value and unit in provenance metadata</li> <li>• Conversion algorithm and factors used</li> <li>• Target standard units as specified in WIGOS metadata standards</li> </ul> |
|---|--|

#### 6.3.3.7. Duplicate Detection

| Requirement 30 |  |
|----------------|--|
| Identifier     | /req/data-ingestion/duplicate-detection  |
| Statement      |  |
| A              | <p>The system SHALL detect and handle duplicate observations based on:</p> <ul style="list-style-type: none"> <li>• Station identifier</li> <li>• Sensor identifier (where available)</li> <li>• Observation timestamp</li> <li>• Observed element</li> </ul>  |
| B              | <p>When duplicates are detected, the system SHALL:</p> <ul style="list-style-type: none"> <li>• Flag duplicate records for review</li> <li>• Apply configurable duplicate resolution strategies (keep first, keep last, manual review)</li> <li>• Log duplicate detection events</li> <li>• Preserve information about all duplicate instances in provenance metadata</li> </ul> |

#### 6.3.3.8. Character Encoding

| Requirement 31 |   |
|----------------|---|
| Identifier     | /req/data-ingestion/character-encoding  |
| A              | <p>The system SHALL handle international character sets:</p> <ul style="list-style-type: none"> <li>• UTF-8 as default character encoding for all text data</li> <li>• Support for legacy encodings (ASCII, ISO-8859-1) during data ingestion</li> <li>• Proper handling of diacritics and non-Latin scripts in station names and metadata</li> <li>• Character encoding declaration in all text-based exports</li> </ul> |

#### 6.3.3.9. Calculation of derived parameters

- To follow

## 6.4. Requirements Class: Data API

|                    |               |
|--------------------|---------------|
| Requirements class | /req/data-api |
|--------------------|---------------|

|                  |                 |
|------------------|-----------------|
| Target type      | Software system |
| Dependency       | /req/core       |
| Conformance test | /conf/data-api  |

### 6.4.1. Overview

Provide standardized programmatic access to climate observation data and metadata, enabling interoperability with external systems and supporting diverse user communities through well-documented APIs. Standard APIs facilitate integration with WIS 2.0, Regional Climate Centres, and international archives.

### 6.4.2. Justification and Regulatory Basis

WMO-No. 1131 classifies Climate Data Delivery Services as Required (Chapter 8), emphasizing "open spatial standards" and "WMO formats" for data exchange.

WMO-No. 1238 requires that "data, and associated metadata, are registered in the WMO Information System Global Discovery Catalogue" (3.2.1.1c) and that systems support "API interoperability" using standards like OGC API family (Features, Records, Environmental Data Retrieval, Processes). The framework principle that "datasets designated to be exchanged internationally through the WMO Information System" (1.1.1(4)) requires standardized API access.

### 6.4.3. Requirements

#### 6.4.3.1. RESTful API

| Requirement 32 |   |
|----------------|---|
| Identifier     | /req/data-api/rest  |
| Statement      | <p>The system SHALL provide a RESTful API that:</p> <ul style="list-style-type: none"> <li>• Uses HTTP/HTTPS protocols</li> <li>• Implements standard HTTP methods (GET, POST, PUT, DELETE) where appropriate</li> <li>• Returns data in standard formats (JSON, XML, CSV, NetCDF)</li> <li>• Provides comprehensive API documentation</li> <li>• Supports content negotiation based on Accept headers</li> </ul> |

#### 6.4.3.2. Standards Compliance

| Recommendation 2 |   |
|------------------|---|
| Identifier       | /rec/data-api/standards   |
| A                | <p>The system SHOULD implement one or more of the following standard APIs:</p> <ul style="list-style-type: none"> <li>• OGC API - Environmental Data Retrieval (EDR)</li> <li>• OGC API - Features</li> <li>• OGC SensorThings API</li> </ul> |

**NOTE**

Implementation of standardized APIs enhances interoperability with the broader climate and meteorological data ecosystem. The choice of which standards to implement should be based on user community needs and integration requirements.

**6.4.3.3. Query Capabilities**

| Requirement 33 |  |
|----------------|--|
| Identifier     | /req/data-api/query  |
| Statement      | <p>The system SHALL support API queries with the following parameters:</p> <ul style="list-style-type: none"><li>• Spatial extent (bounding box, point, polygon)</li><li>• Temporal extent (time range, specific timestamp)</li><li>• Station identifiers (single or multiple)</li><li>• Observed elements/variables</li><li>• Quality control status</li><li>• Data aggregation options (raw, hourly, daily, monthly means)</li></ul> |

**6.4.3.4. Authentication and Authorization**

| Requirement 34 |   |
|----------------|---|
| Identifier     | /req/data-api/auth  |
| Statement      | Authentication and authorization SHALL comply with /req/security/authentication and /req/security/authorization |

**6.4.3.5. API Versioning**

| Requirement 35 |   |
|----------------|---|
| Identifier     | /req/data-api/versioning  |
| Statement      | <p>The system SHALL implement API versioning to:</p> <ul style="list-style-type: none"><li>• Maintain backward compatibility with existing clients</li><li>• Clearly indicate API version in endpoints or headers</li><li>• Provide migration guidance for deprecated API versions</li><li>• Support multiple API versions concurrently during transition periods</li></ul> |

**6.4.3.6. Integration Interfaces**

| Requirement 36 |                           |
|----------------|---------------------------|
| Identifier     | /req/data-api/integration |

|           |  |
|-----------|--|
| Statement | <p>The system SHALL provide integration mechanisms for:</p> <ul style="list-style-type: none"> <li>• WIS 2.0 message broker subscription and publication</li> <li>• Webhook notifications for data availability or quality control events</li> <li>• File-based integration using watched directories or object storage</li> <li>• Database-level integration using standard connection protocols (ODBC, JDBC)</li> <li>• Message queue integration for asynchronous processing</li> </ul> |
|-----------|--|

#### 6.4.3.7. Data Format Interoperability

| Requirement 37 |   |
|----------------|---|
| Identifier     | /req/data-api/data-formats  |
| Statement      |   |
| A              | <p>The system SHALL export data in interoperable formats:</p> <ul style="list-style-type: none"> <li>• NetCDF following CF conventions for gridded data and discrete sampling geometries.</li> <li>• CSV with RFC 4180 compliance and documented column definitions</li> <li>• JSON following consistent schema conventions</li> <li>• GeoJSON for observations with spatial context</li> <li>• BUFR for exchange with WMO systems</li> </ul> |
| B              | <p>Each export format SHALL include:</p> <ul style="list-style-type: none"> <li>• Comprehensive metadata describing the data structure</li> <li>• Units of measure for all variables</li> <li>• Quality flag definitions and values</li> <li>• Temporal reference system (UTC or clearly specified)</li> <li>• Spatial reference system (coordinate system, datum)</li> </ul>   |

## 6.5. Requirements Class: Homogenisation

|                    |                                 |
|--------------------|---------------------------------|
| Requirements class | /req/homogenisation             |
| Target type        | Software system                 |
| Dependency         | /req/core, /req/quality-control |
| Conformance test   | /conf/homogenisation            |

### 6.5.1. Overview

Detect and correct non-climatic discontinuities in long-term observation series caused by changes in instrumentation, observation practices, station location, or surrounding environment, ensuring temporal consistency essential for climate trend analysis. Homogenisation is critical for producing reliable climate change assessments.

## 6.5.2. Justification and Regulatory Basis

WMO-No. 1131 classifies Data Homogenisation as Recommended (Section 6.1.3), recognizing that homogenized datasets "aim to ensure that the only variability remaining in the time series is that resulting from actual climate variability" (4.5.1.3).

WMO-No. 1238 requires that "Entities record any homogenisation or reprocessing carried out on the data as part of the provenance metadata" (3.3.2.5). The Guidelines on Homogenisation (WMO-No. 1245) provide detailed methodologies. GCOS Monitoring Principle 1 requires assessing "impact of new systems or changes to existing systems," making homogenisation essential for maintaining long-term climate records.

## 6.5.3. Requirements

### 6.5.3.1. Break Point Detection

| Recommendation 3 |  |
|------------------|--|
| Identifier       | /rec/homogenisation/detection  |
| Statement        | <p>The system SHOULD:</p> <ul style="list-style-type: none"><li>• Implement statistical methods for detecting inhomogeneities.</li><li>• Support:<ul style="list-style-type: none"><li>- Relative homogenisation using reference series from neighboring stations</li><li>- Absolute homogenisation methods where appropriate</li><li>- Detection of multiple break points in a single series</li><li>- Configurable significance levels for break point detection</li></ul></li></ul> |

### 6.5.3.2. Adjustment Calculation

| Recommendation 4 |   |
|------------------|---|
| Identifier       | /rec/homogenisation/adjustment  |
| Statement        | <p>The system SHOULD calculate adjustments for detected inhomogeneities:</p> <ul style="list-style-type: none"><li>• Compute adjustment factors based on the magnitude of detected breaks</li><li>• Apply adjustments that preserve long-term means and variability characteristics</li><li>• Support different adjustment methods (additive, multiplicative, or hybrid)</li><li>• Validate adjusted series for statistical consistency</li></ul> |

### 6.5.3.3. Homogenisation Documentation

| Requirement 38 |                                   |
|----------------|-----------------------------------|
| Identifier     | /req/homogenisation/documentation |

|           |  |
|-----------|--|
| Statement | <p>The system SHALL document all homogenisation processes applied including:</p> <ul style="list-style-type: none"> <li>• Break points detected with dates and statistical confidence levels</li> <li>• Adjustment factors applied to each segment</li> <li>• Reference series used for relative homogenisation</li> <li>• Statistical test results and significance values</li> <li>• Comparison of original and homogenized series</li> <li>• Metadata about known station changes that correspond to detected breaks</li> </ul> |
|-----------|--|

#### 6.5.3.4. Homogenized Data Management

| Requirement 39 |  |
|----------------|--|
| Identifier     | /req/homogenisation/data-management  |
| Statement      | <p>The system SHALL manage homogenized data such that:</p> <ul style="list-style-type: none"> <li>• Original (raw) observations are preserved unchanged</li> <li>• Homogenized values are stored as derived data with clear lineage</li> <li>• Both original and homogenized series are accessible to users</li> <li>• Provenance metadata clearly identifies homogenisation processing</li> <li>• Homogenisation can be revised or reprocessed without data loss</li> </ul> |

## 6.6. Requirements Class: Data Rescue

|                    |                                |
|--------------------|--------------------------------|
| Requirements class | /req/data-rescue               |
| Target type        | Software system                |
| Dependency         | /req/core, /req/data-ingestion |
| Conformance test   | /conf/data-rescue              |

### 6.6.1. Overview

Facilitate the recovery of historical climate observations from non-digital sources, ensuring rescued data is integrated into the climate record with complete provenance and appropriate quality control. Data rescue extends the temporal coverage of climate records, essential for understanding long-term climate variability and change.

### 6.6.2. Justification and Regulatory Basis

WMO-No. 1131 classifies Data Rescue capabilities as Recommended (Section 5.2), covering "imaging and digitization, monitoring, and data entry."

WMO-No. 1238 requires Members ensure "Entities develop a data rescue strategy" including "systematic" preservation activities, digitization plans, and resource mobilization (3.5.3.1).

#### NOTE

The Guidelines on Good Practices for Data Rescue (WMO-No. 1182) emphasizes that data rescue is "the ongoing process of identifying and preserving all data, records and climate archives that are at risk of being lost."



## 6.6.3. Requirements

### 6.6.3.1. Document Imaging Management

| Recommendation 5 |   |
|------------------|---|
| Identifier       | /rec/data-rescue/imaging  |
| Statement        | <p>The system SHOULD support document imaging workflows including:</p> <ul style="list-style-type: none"><li>• Storage of scanned images with persistent identifiers</li><li>• Association of images with source document metadata (archive, collection, date range)</li><li>• Support for multiple image formats (JPEG, TIFF, PDF)</li><li>• Image quality metadata (resolution, color depth, scan date)</li><li>• Linkage between images and digitized observations</li></ul> |

### 6.6.3.2. Data Entry Interface

| Recommendation 6 |  |
|------------------|--|
| Identifier       | /req/data-rescue/data-entry  |
| A                | <p>The system SHOULD provide a data entry interface for manual keying that:</p> <ul style="list-style-type: none"><li>• Displays source images alongside data entry forms</li><li>• Supports configurable data entry templates based on historical form layouts</li><li>• Provides real-time validation of entered values</li><li>• Tracks data entry operator and timestamp</li><li>• Supports both single-key and double-key entry workflows for quality assurance</li></ul> |

### 6.6.3.3. Data Rescue Quality Control

| Requirement 40 |  |
|----------------|--|
| Identifier     | /req/data-rescue/quality-control   |
| Statement      | <p>The system SHALL apply quality control to rescued data including:</p> <ul style="list-style-type: none"><li>• Comparison between double-key entries to identify transcription errors</li><li>• Range and consistency checks appropriate to historical observation practices</li><li>• Verification against original images when discrepancies are detected</li><li>• Flagging of uncertain or illegible values</li><li>• Documentation of data rescue quality assurance decisions</li></ul> |

### 6.6.3.4. Data Rescue Provenance

| Requirement 41 |                             |
|----------------|-----------------------------|
| Identifier     | /req/data-rescue/provenance |

|           |   |
|-----------|---|
| Statement | <p>The system SHALL record data rescue provenance including:</p> <ul style="list-style-type: none"> <li>• Source document identification and archive location</li> <li>• Image identifier and scan metadata</li> <li>• Data entry operator(s) and keying date(s)</li> <li>• Keying method (single-key, double-key, automated OCR)</li> <li>• Quality assurance verification status</li> <li>• Known limitations or uncertainties in rescued data</li> </ul> |
|-----------|---|

## 6.7. Requirements Class: Metadata Management

|                    |                           |
|--------------------|---------------------------|
| Requirements class | /req/metadata-management  |
| Target type        | Software system           |
| Dependency         | /req/core                 |
| Conformance test   | /conf/metadata-management |

### 6.7.1. Overview

Maintain detailed, version-controlled station and sensor metadata that supports climate data interpretation, quality control, and homogenization through complete documentation of station history and observing conditions. Comprehensive metadata is essential for establishing data provenance and assessing fitness-for-purpose.

### 6.7.2. Justification and Regulatory Basis

WMO-No. 1131 classifies Climate Metadata management as Required (Section 4.3 and 5.5), encompassing "observations metadata, discovery metadata, and data provenance metadata."

WMO-No. 1238 mandates compliance with "WIGOS Metadata Standard" (3.3.1.3) and requires "Members shall ensure that Entities follow the provisions for the collection, management and publication of observational metadata as specified in the Manual on the WMO Integrated Global Observing System (WMO-No. 1160)" (2.1.2.3). The framework principle requiring "comprehensive records of all modifications and processing" (3.3.1.1) depends on robust metadata management.

### 6.7.3. Requirements

#### 6.7.3.1. WIGOS Metadata Compliance

| Requirement 42 |                                |
|----------------|--------------------------------|
| Identifier     | /req/metadata-management/wigos |

|           |   |
|-----------|---|
| Statement | <p>The system SHALL support metadata fields as specified in WIGOS Metadata Standard (WMO-No. 1195), including:</p> <ul style="list-style-type: none"> <li>• Station identification (WIGOS Station Identifier, national identifiers, historical identifiers)</li> <li>• Station name, description, and alternative names</li> <li>• Geographic location (latitude, longitude, elevation, datum reference)</li> <li>• Administrative information (operating organization, country, region)</li> <li>• Station classification (WMO station type, network affiliations, programme participation)</li> <li>• Sensor and instrument inventory with make, model, serial number, and calibration status</li> <li>• Measurement configuration (observed elements, measurement height/depth, sampling interval)</li> <li>• Siting environment description and exposure classification</li> <li>• Operating schedule and reporting practices</li> <li>• Responsible party and contact information</li> </ul> |
|-----------|---|

#### 6.7.3.2. Station History Documentation

| Requirement 43 |  |
|----------------|--|
| Identifier     | /req/metadata-management/history   |
| Statement      | The system SHALL:  |
| A              | <p>maintain a complete station history, including:</p> <ul style="list-style-type: none"> <li>• Station relocations with old and new coordinates</li> <li>• Instrument changes with make, model, and installation dates</li> <li>• Changes to observation practices or measurement heights</li> <li>• Changes to surrounding environment affecting exposure</li> <li>• Periods of station closure or data gaps</li> <li>• Network affiliation changes</li> </ul> |
| B              | <p>for each entry in the station history include:</p> <ul style="list-style-type: none"> <li>• Effective date and time of the change</li> <li>• Description of the change</li> <li>• Reason for the change where known</li> <li>• Responsible party documenting the change</li> <li>• Supporting documentation references</li> </ul>   |

#### 6.7.3.3. Discovery Metadata

To follow ...

#### 6.7.3.4. Metadata Standards Export

| Requirement 44 |   |
|----------------|---|
| Identifier     | /req/metadata-management/export   |
| Statement      | <p>The system SHALL support export of metadata in standard formats including:</p> <ul style="list-style-type: none"><li>• WIGOS Metadata Representation (WMDR)</li><li>• ISO 19115 geographic metadata</li><li>• WMO Core Metadata Profile v2</li><li>• Station metadata exchange formats used by regional or global programs</li></ul> |

#### 6.7.3.5. Metadata Quality Validation

| Requirement 45 |  |
|----------------|--|
| Identifier     | /req/metadata-management/validation  |
| Statement      | The system SHALL   |
| A              | <p>validate metadata for:</p> <ul style="list-style-type: none"><li>• Required field completeness per WIGOS standards</li><li>• Coordinate system validity and range checks</li><li>• Temporal consistency (effective dates do not overlap)</li><li>• Referential integrity (stations referenced by observations exist)</li><li>• Controlled vocabulary compliance for classification fields</li></ul> |
| B              | <p>generate validation reports identifying:</p> <ul style="list-style-type: none"><li>• Missing required metadata</li><li>• Metadata quality issues or inconsistencies</li><li>• Recommendations for metadata improvement</li></ul>  |

#### 6.7.3.6. Citation and Attribution

| Requirement 46 |  |
|----------------|--|
| Identifier     | /req/metadata-management/citation  |
| Statement      | <p>The system SHALL support proper data citation by:</p> <ul style="list-style-type: none"><li>• Generating unique persistent identifiers (DOIs or equivalents) for datasets</li><li>• Providing recommended citation text for datasets</li><li>• Recording data usage and access patterns for impact assessment</li><li>• Supporting dataset versioning with clear version identifiers</li><li>• Maintaining links between derived products and source datasets</li></ul> |

## 6.8. Requirements Class: User Interface

|                    |                      |
|--------------------|----------------------|
| Requirements class | /req/user-interface  |
| Target type        | Software system      |
| Dependency         | /req/core            |
| Conformance test   | /conf/user-interface |

### 6.8.1. Overview

Provide interactive capabilities for climate scientists, data managers, and operators to visualize, explore, and analyze climate observation data and quality control results. The user interface enables efficient human interaction with complex climate datasets and supports quality assurance workflows.

### 6.8.2. Justification and Regulatory Basis

WMO-No. 1131 identifies the Graphical User Interface for time-series data exploration as Recommended (Section 7.1), supporting "tables and charts, visualization, manage content, integrated search of climate data, and data download."

While WMO-No. 1238 does not mandate specific UI requirements, it emphasizes the need for "climate scientists to review and validate climate observations, identify inhomogeneities" and for "data managers to manage station metadata and data holdings" (Figure 2), which requires appropriate user interfaces. The requirement for "comprehensive records" and "quality control" workflows (3.3.2) implies interactive tools for human oversight.

### 6.8.3. Requirements

#### 6.8.3.1. Visualization Capabilities

| Requirement 47 |  |
|----------------|--|
| Identifier     | /req/user-interface/visualization  |
| Statement      | <p>The system SHALL provide data visualization capabilities including:</p> <ul style="list-style-type: none"><li>• Time series plots for observation values over configurable time ranges</li><li>• Simultaneous display of original and quality-controlled or homogenized values</li><li>• Station location maps with filtering and selection capabilities</li><li>• Ability to export visualizations in common image formats (PNG, SVG, PDF)</li><li>• Interactive exploration of quality control flags and metadata</li></ul> |

#### 6.8.3.2. Tabular data

To follow...

#### 6.8.3.3. Integrated search

To follow ...

#### 6.8.3.4. Data entry / forms

To follow ...

## 6.9. Requirements Class: Climate Products

|                    |                        |
|--------------------|------------------------|
| Requirements class | /req/climate-products  |
| Target type        | Software system        |
| Dependency         |                        |
| Conformance test   | /conf/climate-products |

- Climate products
- Climate indices

## 6.10. Requirements Class: Processing Automation and Scheduling

|                     |                  |
|---------------------|------------------|
| Requirements class: | /req/automation  |
| Target type:        | Software system  |
| Dependency:         | /req/core        |
| Conformance test:   | /conf/automation |

### 6.10.1. Overview

Define requirements for automated execution of routine climate data processing tasks, scheduled operations, and workflow orchestration to ensure timely data processing without manual intervention.

### 6.10.2. Justification and Regulatory Basis

WMO-No. 1238 requires timely generation of core climate products (2.2.1) and systematic data processing. WMO-No. 1131 emphasizes automated processing workflows.

### 6.10.3. Requirements

to follow ...

# Chapter 7. Non-functional Requirements

This section defines the quality attributes and constraints that specify how well the Climate Data Management System performs its functions. These requirements address system characteristics essential for operational deployment, long-term sustainability, and user satisfaction.

## 7.1. Requirements Class: Performance and Scalability

|                    |                   |
|--------------------|-------------------|
| Requirements class | /req/performance  |
| Target type        | Software system   |
| Dependency         | /req/core         |
| Conformance test   | /conf/performance |

### 7.1.1. Overview

Ensure the system can handle operational data volumes, concurrent users, and query loads efficiently while maintaining acceptable response times and supporting future growth. Performance requirements ensure the CDMS remains viable as data volumes increase and user communities expand over decades of operation.

### 7.1.2. Justification and Regulatory Basis

WMO-No. 1131 emphasizes that CDMSs must handle "big data" characterized by "large volumes of data in a variety of formats that present challenges for effective data management" (Section 4.1). The specification notes systems must manage "observation datasets spanning minimum 100 years of historical data" with scalability for "considerable amounts of derived data" (Section 4.5).

WMO-No. 1238 requires systems support "duplicate repositories for the duration of their retention periods" (3.4.2.2) and handle data from multiple Earth system domains (1.1.2), implying substantial performance requirements. The framework principle of ensuring "adequate uniformity and standardization" (Introduction, para 5e) requires consistent performance across implementations.

### 7.1.3. Requirements

To follow ...

## 7.2. Requirements Class: Reliability and Availability

|                    |                   |
|--------------------|-------------------|
| Requirements class | /req/reliability  |
| Target type        | Software system   |
| Dependency         | /req/core         |
| Conformance test   | /conf/reliability |

### 7.2.1. Overview

Ensure the system operates dependably with minimal unplanned downtime, maintains data integrity under all conditions, and recovers gracefully from failures. Reliability is essential for climate data, which serves as an irreplaceable scientific resource requiring protection for future generations.

## 7.2.2. Justification and Regulatory Basis

WMO-No. 1131 states that "the most important stakeholders for climate data are future generations, and the reason to implement and manage CDMSs now is to protect the integrity of climate data for their future use" (Section 2.3). The specification's core theme is to "ensure that the data are securely and transparently managed" (Section 2.3).

WMO-No. 1238 mandates that "Members shall ensure that Entities develop and maintain a business continuity plan to mitigate risks associated with disruption of operations" (3.4.1.1) and requires "routine backup, and procedures for timely restoration of their databases" (3.4.1.2). The requirement to "retain their original and irreproducible climate records, and associated metadata, permanently" (3.4.1.4) establishes reliability as a foundational requirement.

## 7.2.3. Requirements

### 7.2.3.1. System Availability

| Requirement 48 |   |
|----------------|---|
| Identifier     | /req/reliability/availability   |
| Statement      | <p>The system SHALL provide:</p> <ul style="list-style-type: none"><li>• Minimum 99.5% uptime for data query and retrieval operations (excluding scheduled maintenance)</li><li>• Minimum 99.0% uptime for data ingestion operations</li><li>• Scheduled maintenance windows clearly communicated minimum 7 days in advance</li><li>• Maximum of 4 hours per month for scheduled maintenance downtime</li></ul> |

#### NOTE

Availability is calculated as:  $(\text{Total time} - \text{Downtime}) / \text{Total time} \times 100\%$ , measured monthly. Uptime excludes scheduled maintenance windows but includes all unplanned outages.

### 7.2.3.2. Data Integrity

| Requirement 49 |   |
|----------------|---|
| Identifier     | /req/reliability/data-integrity   |
| Statement      | <p>The system SHALL ensure data integrity through:</p> <ul style="list-style-type: none"><li>• ACID-compliant transactions for all data create, update, and delete operations</li><li>• Referential integrity constraints preventing orphaned observations (observations without valid station references)</li><li>• Checksums or cryptographic hash verification for file-based data transfers</li><li>• Write-ahead logging or equivalent mechanisms to prevent data loss during system failures</li><li>• Periodic integrity checks to detect and report data corruption</li></ul> |



### 7.2.3.3. Backup and Recovery

| Requirement 50 |   |
|----------------|---|
| Identifier     | /req/reliability/backup-recovery  |
| Statement      | <p>The system SHALL implement backup and recovery capabilities with:</p> <ul style="list-style-type: none"><li>• Automated daily incremental backups of all observation and metadata</li><li>• Automated weekly full backups retained for minimum 3 months</li><li>• Point-in-time recovery capability to restore system state to any point within the last 30 days</li><li>• Recovery Time Objective (RTO) of 4 hours for complete system restoration</li><li>• Recovery Point Objective (RPO) of 1 hour maximum data loss</li></ul> |

### 7.2.3.4. Fault Tolerance

| Requirement 51 |   |
|----------------|---|
| Identifier     | /req/reliability/fault-tolerance  |
| Statement      | <p>The system SHALL handle failures gracefully by:</p> <ul style="list-style-type: none"><li>• Continuing to serve read requests when write operations are unavailable</li><li>• Queuing ingestion requests during temporary processing unavailability</li><li>• Providing meaningful error messages rather than generic failures</li><li>• Logging all error conditions with sufficient detail for diagnosis</li><li>• Automatically retrying failed operations with exponential backoff</li></ul> |

### 7.2.3.5. Data Validation on Retrieval

| Requirement 52 |  |
|----------------|--|
| Identifier     | /req/reliability/retrieval-validation  |
| Statement      | <p>The system SHALL validate data consistency when retrieving stored data:</p> <ul style="list-style-type: none"><li>• Verify checksums for file-based retrievals</li><li>• Detect and report any data corruption to system administrators</li><li>• Provide alternative access to backup copies when primary data is corrupted</li><li>• Maintain audit logs of data integrity violations</li></ul> |

## 7.3. Requirements Class: Security

|                    |                 |
|--------------------|-----------------|
| Requirements class | /req/security   |
| Target type        | Software system |
| Dependency         | /req/core       |
| Conformance test   | /conf/security  |

### 7.3.1. Overview

Protect climate data assets, ensure appropriate access controls, maintain confidentiality where required, and provide comprehensive audit trails of security-relevant events. Security safeguards the integrity and trustworthiness of the climate record while enabling appropriate access for legitimate users.

### 7.3.2. Justification and Regulatory Basis

WMO-No. 1131 requires "controlled access to data and systems" as a Required governance component (Section 3.2.1.1), specifying that "staff who have write access to data stored within a database only be able to change data under software control" with comprehensive audit trails. The specification mandates "each successful change to observations data be audited to ensure that the change, the details of the operator who made the change and the time of the change have all been recorded" (Section 3.2.1.1).

WMO-No. 1238 requires "Members shall ensure Entities implement rigorous authorization processes for modification or deletion of data or metadata" (3.4.4.1) and mandates "regularly review systems for the physical- and cyber-security of the climate record" (3.4.4.2).

### 7.3.3. Requirements

#### 7.3.3.1. Authentication

| Requirement 53 |  |
|----------------|--|
| Identifier     | /req/security/authentication   |
| Statement      | <p>The system SHALL implement authentication mechanisms that:</p> <ul style="list-style-type: none"><li>• Support integration with enterprise identity management systems (LDAP, Active Directory, OAuth 2.0, SAML 2.0)</li><li>• Enforce strong password policies (minimum length, complexity, expiration) for local accounts</li><li>• Support multi-factor authentication (MFA) for administrative accounts</li><li>• Implement account lockout after configurable number of failed login attempts</li><li>• Maintain secure session management with automatic timeout after inactivity</li></ul> |

#### 7.3.3.2. Authorization and Access Control

| Requirement 54 |                             |
|----------------|-----------------------------|
| Identifier     | /req/security/authorization |

|           |   |
|-----------|---|
| Statement | <p>The system SHALL implement role-based access control (RBAC) with:</p> <ul style="list-style-type: none"> <li>• Separation of privileges for read-only users, data entry operators, quality control analysts, and system administrators</li> <li>• Station-level or network-level data access restrictions where operationally required</li> <li>• Support for data access policies that restrict access to embargoed or sensitive data</li> <li>• Principle of least privilege - users granted minimum permissions necessary for their role</li> <li>• Ability to temporarily elevate privileges with approval workflow for exceptional circumstances</li> </ul> |
|-----------|---|

#### 7.3.3.3. Data Encryption

| Requirement 55 |   |
|----------------|---|
| Identifier     | /req/security/encryption  |
| Statement      | <p>The system SHALL implement encryption for:</p> <ul style="list-style-type: none"> <li>• All data in transit using TLS 1.2 or higher for HTTPS connections</li> <li>• All API communications using TLS 1.2 or higher</li> <li>• Sensitive data at rest including user credentials, API keys, and personal information</li> <li>• Database connections using encrypted channels</li> <li>• Backup files stored on external media or cloud storage</li> </ul> |

#### 7.3.3.4. Audit Logging

| Requirement 56 |  |
|----------------|--|
| Identifier     | /req/security/audit-logging  |
| Statement      | The system SHALL:  |
| A              | <p>maintain comprehensive audit logs recording:</p> <ul style="list-style-type: none"> <li>• All user authentication attempts (successful and failed)</li> <li>• All data modifications (create, update, delete) with user identification</li> <li>• All administrative operations (configuration changes, user management, permission changes)</li> <li>• All quality control flag overrides with justification</li> <li>• All data export operations including user, timestamp, and data selection criteria</li> <li>• All API access patterns including endpoint, user, timestamp, and response status</li> </ul> |

|   |  |
|---|--|
| B | <p>Audit logs SHALL:</p> <ul style="list-style-type: none"> <li>• Be tamper-evident or stored in write-once media</li> <li>• Be retained for minimum 7 years</li> <li>• Be searchable and exportable for security incident investigation</li> <li>• Include sufficient detail to reconstruct the sequence of events</li> </ul> |
|---|--|

### 7.3.3.5. Vulnerability Management

| Requirement 57 |   |
|----------------|---|
| Identifier     | /req/security/vulnerability-management  |
| A              | <p>The system SHALL support security maintenance through:</p> <ul style="list-style-type: none"> <li>• Regular security updates to address known vulnerabilities</li> <li>• Documented process for applying security patches</li> <li>• Vulnerability scanning capability or support for external security assessment tools</li> <li>• Security advisories for known issues affecting the deployment</li> </ul> |

### 7.3.3.6. Data Privacy

| Requirement 58 |   |
|----------------|---|
| Identifier     | /req/security/data-privacy  |
| A              | <p>The system SHALL protect personal information by:</p> <ul style="list-style-type: none"> <li>• Minimizing collection and storage of personally identifiable information (PII)</li> <li>• Providing mechanisms to redact or anonymize personal information where required by data protection regulations</li> <li>• Supporting data subject access requests (GDPR right to access)</li> <li>• Supporting data deletion requests where legally required and operationally feasible</li> <li>• Documenting data retention policies and automated deletion of expired personal data</li> </ul> |

### 7.3.3.7. Provenance Security

| Requirement 59 |   |
|----------------|---|
| Identifier     | /req/security/provenance-security   |
| A              | <p>The system SHALL protect provenance records through:</p> <ul style="list-style-type: none"> <li>• Cryptographic timestamps or equivalent mechanisms to prevent backdating</li> <li>• Tamper-evident storage for provenance audit trails</li> </ul> |

## 7.4. Requirements Class: Usability

|                    |                 |
|--------------------|-----------------|
| Requirements class | /req/usability  |
| Target type        | Software system |
| Dependency         | /req/core       |
| Conformance test   | /conf/usability |

### 7.4.1. Overview

Ensure the system is accessible, intuitive, and efficient for climate scientists, data managers, and operators with varying levels of technical expertise. Usability directly impacts the effectiveness of quality control, data validation, and metadata management workflows that depend on expert human judgment.

### 7.4.2. Justification and Regulatory Basis

WMO-No. 1131 emphasizes that CDMSs must support diverse user roles including "climate scientists, data managers, system operators and data entry operators" (Figure 2 and Figure 3). The specification notes that while "most NMHSs would not manage infrastructure and software required to support" certain advanced components, systems must still be "flexible enough to accommodate national climate monitoring products and other national indices" (Section 4.4.2.3).

WMO-No. 1238 recognizes the need for "climate scientists to review and validate climate observations, identify inhomogeneities" (Figure 2), which requires effective user interfaces. The Competencies requirements (3.6) acknowledge that staff need appropriate training, implying systems should support learnable, efficient workflows.

### 7.4.3. Requirements

#### 7.4.3.1. User Interface Design

| Requirement 60 |   |
|----------------|---|
| Identifier     | /req/usability/ui-design  |
| A              | <p>The system user interface SHALL:</p> <ul style="list-style-type: none"><li>• Provide consistent navigation patterns across all functions</li><li>• Use clear, unambiguous labels and terminology aligned with climate science conventions</li><li>• Provide contextual help and documentation accessible from relevant screens</li><li>• Support keyboard navigation for efficiency</li><li>• Indicate required fields and validation requirements clearly before submission</li><li>• Provide meaningful error messages with guidance on corrective actions</li></ul> |

#### 7.4.3.2. Accessibility

| Recommendation 7 |                              |
|------------------|------------------------------|
| Identifier       | /rec/usability/accessibility |

|   |  |
|---|--|
| A | <p>The system user interface SHOULD conform to:</p> <ul style="list-style-type: none"> <li>• Web Content Accessibility Guidelines (WCAG) 2.1 Level AA where applicable</li> <li>• Support for screen readers and assistive technologies</li> <li>• Sufficient color contrast for visual accessibility</li> <li>• Text scaling without loss of functionality</li> <li>• Keyboard-only operation for all critical functions</li> </ul> |
|---|--|

#### 7.4.3.3. User Feedback and Notifications

| Requirement 61 |   |
|----------------|---|
| Identifier     | /req/usability/notifications  |
| A              | <p>The system SHALL provide notifications for:</p> <ul style="list-style-type: none"> <li>• Completion of long-running operations (ingestion, QC processing, homogenization)</li> <li>• Critical quality control failures requiring manual review</li> <li>• System maintenance schedules</li> <li>• Data embargo or access restriction changes</li> <li>• Configurable notification delivery methods (email, in-application alerts)</li> </ul> |

## 7.5. Requirements Class: Maintainability

|                    |                       |
|--------------------|-----------------------|
| Requirements class | /req/maintainability  |
| Target type        | Software system       |
| Dependency         | /req/core             |
| Conformance test   | /conf/maintainability |

### 7.5.1. Overview

Ensure the system can be updated, configured, and troubleshoot efficiently with minimal disruption to operations. Maintainability is critical for long-term sustainability, as climate data systems must operate reliably over multiple decades with evolving technologies and requirements.

### 7.5.2. Justification and Regulatory Basis

WMO-No. 1131 emphasizes that "one of the steps often overlooked when implementing a CDMS is to ensure that the NMHS has established a solid governance framework to protect the integrity of the climate record" (Section 3), including "processes that ensure adherence to CDMS-related policies" and comprehensive documentation requirements (Section 3.2.2.5). The specification requires "effective IT security and monitoring" (Section 5.2.3) and "managed change" as Required governance (Section 3.2.2.2), noting that "uncontrolled change can result in chaos for users" including "data corruptions or even data loss."

WMO-No. 1238 mandates "Members shall ensure that Entities communicate work involving changes to the data management system and associated software to affected stakeholders in a timely fashion" (3.3.3.3)

and requires comprehensive documentation of "data management processes, including details of roles and responsibilities" (3.3.3.1).

### 7.5.3. Requirements

#### 7.5.3.1. Configuration Management

| Requirement 62 |  |
|----------------|--|
| Identifier     | /req/maintainability/configuration   |
| A              | <p>The system SHALL support configuration management with:</p> <ul style="list-style-type: none"><li>• Externalized configuration files separate from application code</li><li>• Hot-reload capability for quality control threshold updates without system restart</li><li>• Configuration version control with rollback capability</li><li>• Configuration validation before activation to prevent invalid settings</li><li>• Documentation of all configurable parameters with valid value ranges</li></ul> |

#### 7.5.3.2. Logging and Diagnostics

| Requirement 63 |  |
|----------------|--|
| Identifier     | /req/maintainability/logging   |
| A              | <p>The system SHALL provide logging and diagnostic capabilities:</p> <ul style="list-style-type: none"><li>• Configurable log levels (ERROR, WARN, INFO, DEBUG) per component</li><li>• Structured logging with timestamps, component identification, and correlation IDs</li><li>• Log rotation and retention policies to manage disk space</li><li>• Centralized logging capability for distributed deployments</li><li>• Performance metrics logging (response times, throughput, resource utilization)</li></ul> |

#### 7.5.3.3. Health Monitoring

| Requirement 64 |   |
|----------------|---|
| Identifier     | /req/maintainability/health-monitoring  |
| A              | <p>The system SHALL expose health monitoring endpoints providing:</p> <ul style="list-style-type: none"><li>• Overall system health status (healthy, degraded, unhealthy)</li><li>• Component-level health status (database, message queue, external integrations)</li><li>• Key performance indicators (query response times, ingestion rates, error rates)</li><li>• Resource utilization metrics (CPU, memory, disk, network)</li><li>• Integration with standard monitoring tools (Prometheus, Grafana, Nagios, or equivalents)</li></ul> |

#### 7.5.3.4. API Backward Compatibility

| Requirement 65 |   |
|----------------|---|
| Identifier     | /req/maintainability/api-compatibility  |
| A              | <p>The system SHALL maintain API stability through:</p> <ul style="list-style-type: none"><li>• Semantic versioning of API endpoints</li><li>• Support for deprecated API versions for minimum 24 months after deprecation announcement</li><li>• Clear deprecation notices in API documentation and response headers</li><li>• Migration guides for breaking changes</li><li>• Ability to run multiple API versions concurrently during transition periods</li></ul> |

#### 7.5.3.5. Database Schema Evolution

| Requirement 66 |  |
|----------------|--|
| Identifier     | /req/maintainability/schema-evolution  |
| A              | <p>The system SHALL support database schema changes with:</p> <ul style="list-style-type: none"><li>• Version-controlled schema migration scripts</li><li>• Automated or documented migration procedures</li><li>• Ability to test schema migrations in non-production environments</li><li>• Rollback capability for failed migrations</li><li>• Backward compatibility for schema changes where feasible</li></ul> |

#### 7.5.3.6. Documentation

| Requirement 67 |  |
|----------------|--|
| Identifier     | /req/maintainability/documentation   |
| A              | <p>The system SHALL provide comprehensive documentation including:</p> <ul style="list-style-type: none"><li>• System architecture and component descriptions</li><li>• Installation and deployment procedures</li><li>• Configuration reference with parameter descriptions and examples</li><li>• API reference documentation with request/response examples</li><li>• Troubleshooting guides for common issues</li><li>• Backup and recovery procedures</li><li>• Security hardening guidelines</li></ul> |

## 7.6. Requirements Class: Interoperability

|                    |                       |
|--------------------|-----------------------|
| Requirements class | /req/interoperability |
| Target type        | Software system       |



|                  |                          |
|------------------|--------------------------|
| Dependency       | /req/core, /req/data-api |
| Conformance test | /conf/interoperability   |

### 7.6.1. Overview

Ensure the system can exchange data and integrate with external systems using industry-standard protocols, formats, and conventions. Interoperability enables climate data to flow seamlessly between national systems, regional centers, international archives, and the WIS 2.0 infrastructure.

### 7.6.2. Justification and Regulatory Basis

WMO-No. 1131 emphasizes that "open spatial standards are a mechanism that is being increasingly adopted by many organizations and industry sectors around the world" (Section 8.1) and that these standards are "expected to become an increasingly important mechanism for distributing data in future years." The specification notes that standardized delivery "will ensure that data are represented in a form that facilitates data interoperability and are accessible to a wide range of end users from disparate industries" (Section 3.1.4.1).

WMO-No. 1238 requires "Members shall ensure that Entities encode data, and associated metadata, intended for international exchange following the standards specified in the Manual on Codes (WMO-No. 306), the Manual on the WMO Information System (WMO-No. 1060)" (3.2.2.1). The framework principle mandating registration in "the WIS Global Discovery Catalogue" (1.1.1(4)) requires standards-based interoperability.

### 7.6.3. Requirements

#### 7.6.3.1. Standards Compliance

| Requirement 68 |   |
|----------------|---|
| Identifier     | /req/interoperability/standards   |
| A              | <p>The system SHALL comply with relevant international standards:</p> <ul style="list-style-type: none"> <li>• WMO WIGOS Metadata Standard for station metadata</li> <li>• WMO Manual on Codes (WMO-No. 306) for BUFR and code form support</li> <li>• CF Conventions for NetCDF climate data files</li> <li>• ISO 19115 for geographic metadata where applicable</li> <li>• OGC standards (API - Features, API - EDR, or SensorThings API) for data access APIs</li> </ul> |

#### 7.6.3.2. API Interoperability

| Requirement 69 |                           |
|----------------|---------------------------|
| Identifier     | /req/interoperability/api |

|   |   |
|---|---|
| A | <p>The system API SHALL support interoperability through:</p> <ul style="list-style-type: none"> <li>• Content negotiation using standard HTTP Accept headers</li> <li>• Standard HTTP status codes with appropriate semantics</li> <li>• CORS (Cross-Origin Resource Sharing) support for web-based clients</li> <li>• OpenAPI (Swagger) specification or equivalent machine-readable API description</li> <li>• Hypermedia links (HATEOAS) for resource discovery where applicable</li> </ul> |
|---|---|

## 7.7. Requirements Class: Data Governance

|                    |                       |
|--------------------|-----------------------|
| Requirements class | /req/data-governance  |
| Target type        | Software system       |
| Dependency         | /req/core             |
| Conformance test   | /conf/data-governance |

### 7.7.1. Overview

Ensure proper stewardship of climate data through lifecycle management, quality documentation, access policies, and compliance with organizational and regulatory requirements. Data governance provides the policy framework and accountability structures necessary for long-term climate data stewardship.

### 7.7.2. Justification and Regulatory Basis

WMO-No. 1131 dedicates Chapter 3 entirely to "CDMS Governance," stating that "one of the steps often overlooked when implementing a CDMS is to ensure that the NMHS has established a solid governance framework to protect the integrity of the climate record" involving "Policies" and "Governance processes." The specification defines data governance as Required (Section 3.2.1), including "controlled access to data and systems," "approval process for new data types," "approval process to change data," and "IT change approvals."

WMO-No. 1238 requires "Members should ensure that Entities implement and maintain a governance mechanism to oversee the various governance elements, coordinated by a governance body" (3.5.1.1) and mandates policies covering commitments, sustainability, intellectual property, and data delivery (Section 3.1 of WMO-No. 1131). The framework emphasizes that "data stewardship assigns to an entity rights and responsibilities for acquiring and managing climate data and information" (Definitions), requiring formalized governance structures.

### 7.7.3. Requirements

#### 7.7.3.1. Data Lifecycle Management

| Requirement 70 |                                |
|----------------|--------------------------------|
| Identifier     | /req/data-governance/lifecycle |

|           |  |
|-----------|--|
| Statement | <p>The system SHALL support data lifecycle management with:</p> <ul style="list-style-type: none"> <li>• Configurable data retention policies per dataset type or network</li> <li>• Automated archival of historical data to long-term storage</li> <li>• Data deletion workflows with approval requirements for permanent deletion</li> <li>• Tombstone records for deleted data maintaining metadata about what was deleted and why</li> <li>• Migration paths for data format updates or schema changes</li> </ul> |
|-----------|--|

#### 7.7.3.2. Data Quality Metadata

| Requirement 71 |   |
|----------------|---|
| Identifier     | /req/data-governance/quality-metadata   |
| Statement      | <p>The system SHALL maintain data quality metadata including:</p> <ul style="list-style-type: none"> <li>• Completeness metrics (percentage of expected observations received)</li> <li>• Timeliness metrics (delay between observation time and system receipt)</li> <li>• Quality control statistics (pass/fail rates per check type)</li> <li>• Manual intervention frequency and types</li> <li>• Homogenization status and adjustment statistics</li> <li>• Overall fitness-for-purpose assessments</li> </ul> |
| B              | <p>Quality metadata SHALL be:</p> <ul style="list-style-type: none"> <li>• Queryable alongside observation data</li> <li>• Exportable in machine-readable formats</li> <li>• Updated automatically as data quality changes</li> </ul>   |

#### 7.7.3.3. Data Access Policies

| Requirement 72 |   |
|----------------|---|
| Identifier     | /req/data-governance/access-policies  |
| Statement      | <p>The system SHALL enforce data access policies including:</p> <ul style="list-style-type: none"> <li>• Public/restricted/confidential data classification</li> <li>• Embargo periods for recently collected data where required</li> <li>• License and attribution requirements for data usage</li> <li>• Data sharing agreements and restrictions per network or organization</li> <li>• Automated enforcement of access restrictions based on user roles</li> </ul> |

#### 7.7.3.4. Regulatory Compliance

| Requirement 73 |                                 |
|----------------|---------------------------------|
| Identifier     | /req/data-governance/compliance |

|           |  |
|-----------|--|
| Statement | <p>The system SHALL support compliance with relevant regulations:</p> <ul style="list-style-type: none"> <li>• What goes here?</li> <li>• Is this needed?</li> </ul> |
|-----------|--|

#### 7.7.3.5. Data Stewardship Roles

| Requirement 74 |   |
|----------------|---|
| Identifier     | /req/data-governance/stewardship  |
| Statement      | <p>The system SHALL define and support data stewardship roles including:</p> <ul style="list-style-type: none"> <li>• Data custodian responsibilities for dataset maintenance</li> <li>• Data quality manager authority for quality control decisions</li> <li>• Data access coordinator for managing access requests and policies</li> <li>• System administrator separation from data stewardship roles</li> <li>• Clear escalation paths for data governance issues</li> </ul> |