

# Integrity Checking (CPP-003)

<b>CPP-Identifier</b>	CPP-003
<b>CPP-Label</b>	Integrity Checking
<b>Author</b>	Johan Kylander
<b>Contributors</b>	Bertrand Caron
<b>Evaluators</b>	Maria Benauer, Felix Burger, Laura Molloy
<b>Date of edition completed</b>	29.08.2025
<b>Change history</b>	<b>Comments</b>
Version 1.0 - 29.08.2025	Milestone version

# 1. Description of the CPP

The TDA supports periodic integrity checking, reporting any damaged or missing *Files*.

## Inputs and outputs

Input(s)	
Data	<i>Information package</i>
Metadata	<i>Fixity metadata</i>
	<i>Storage management information</i>
Documentation / guidance	Storage management policy - Integrity checking
	Storage management policy - Checksum algorithms
Output(s)	
Metadata	<i>Fixity metadata</i>
	<i>Provenance metadata</i>

## Definition and scope

Integrity checking is a periodically performed process where a checksum is calculated for a target *Information Package* and compared to the existing stored checksum (as calculated in CPP-001 **Checksum Generation and Recording**). The goal of integrity checking is to confirm that a target *Information Package* has remained unaltered across its life cycle. A TDA must perform and document periodic checks, and the frequency of the checks should be defined in its policy as part of the **Risk Mitigation** (CPP-012) approach.

Integrity checking is closely related to the process of **Checksum Validation** (CPP-002). Whereas Checksum Validation is tied to **Ingest** (CPP-029), **Enabling Access** (CPP-025), or **Replication** (CPP-011) (i.e. processes where *Files* are transferred or new copies are created), Integrity Checking is related to continuous risk management. Integrity checking aims to mitigate bit rot and provides evidence for trustworthy preservation by maintaining a continuous audit trail verifying that a *File* has remained unchanged and authentic over time.

Periodic integrity checks are performed separately on all accessible copies of a target *Information Package* (for example, off-line copies in a dark archive are usually excluded from periodic integrity checks). Copies on different storage media might be subjected to different intervals of checks. The results of the integrity checks, including *Fixity Metadata*, should be documented as preservation actions.

If integrity checks discover problems in the integrity of the target *Information Packages*, this information must be clearly documented in a digital archive's system, so that the broken *Information Packages* can be restored from valid copies (see CPP-004 **Data Corruption Management**).

## Process description

### Trigger event(s)

Trigger event	CPP-identifier
Frequency of integrity checks defined in a digital archives policy	CPP-012 (Risk Mitigation)
Suspicion of an error triggering an integrity check on an ad hoc basis	/

### Step-by-step description

No	Supplier	Input	Steps	Output	Customer
1	CPP-012 (Risk Mitigation)	Storage management policy - Integrity checking	Gather a batch of targets to check and their corresponding <i>Fixity metadata</i> (e.g. <i>Information Packages</i> whose last-checked timestamp is older than the specified checking frequency)	<i>AIPs</i>	
		<i>Fixity metadata</i>			
2		<i>Storage Management information</i>	For each <i>AIP</i> in the selected batch (steps 2a to 2e):		
2A	CPP-001 (Checksum Generation and Recording)	<i>Fixity metadata</i>	Gather the <i>AIP</i> 's fixity metadata	<i>Fixity metadata</i>	

2B		<i>Fixity metadata</i> (algorithms)	Calculate the checksum of the <i>AIP</i> from the specified <i>File</i> path	<i>Fixity metadata</i>	
2C		<i>Fixity metadata</i>	Compare the calculated checksum with the stored checksum	Checksums match: proceed to next step	
				Alert that any of the checksums does not match: <ul style="list-style-type: none"> <li>• mark broken <i>AIP</i> for repair</li> <li>• Proceed to next step</li> </ul>	CPP-004 (Data Corruption Management)
2D			Store the new integrity checking event to the <i>AIP</i>	<i>Provenance metadata</i>	
2E			Update the timestamp of the integrity check	<i>Fixity metadata</i> (timestamp)	
3			Document the event and its timestamp	<i>Provenance metadata</i>	

## Rationale(s)<sup>1</sup> and worst case(s)

Rationale	Impact of inaction or failure of the process
Periodic integrity checks on all copies	Data can get corrupted and degenerate (i.e. the chain of custody is not safeguarded, and the authenticity of <i>IPs</i> may be destroyed)

## 2. Dependencies and relationships with other CPPs

### Dependencies

CPP-ID	CPP-Title	Relationship description
CPP-001	Checksum Generation and Recording	CPP-001 is responsible for creating checksums that are used in integrity checking.
CPP-012	Risk Mitigation	The frequency and target of periodic integrity checks (CPP-003) is defined by an institutional storage management policy as part of risk mitigation (CPP-012).

### Other relations

Relation	CPP-ID	CPP-Title	Relationship description
Required By	CPP-013	Object Management Reporting	Periodic integrity checking provides reports on the integrity of data and reports corrupted <i>AIPs</i> .
Required By	CPP-016	Metadata Ingest and Management	The timestamp of the <i>AIPs</i> checksum needs to be updated to keep track of the last successful check.
Triggers	CPP-004	Data Corruption Management	In case of broken target <i>Information, Packages</i> must be marked for repair.
Affinity with	CPP-007	Virus Scanning	Both processes aim to ensure the "health" of files. However, Integrity

---

<sup>1</sup> Term derived from PREMIS.

			Checking focuses on detecting technical corruption of <i>Files</i> (e.g. bit rot), whereas virus scanning looks to mitigate human-made risks ( e.g. malicious code).
Not to be confused with	CPP-002	Checksum Validation	Both CPPs can get input from CPP-001, and both calculate a checksum from an <i>Information Package</i> and compare it to a given checksum. The difference is that CPP-002 is done during the ingest or access phases (relating to transfer of content, changes in space), while CPP-003 is done periodically during the preservation of the contents in the archival storage (relating to changes over time). Thus, CPP-002 and CPP-003 are not only triggered by different processes, but also trigger different responses.

### 3. Links to frameworks

#### Certification

Certification framework	Term used in framework to refer to the CPP	Section
CTS	Fixity checks	R14 Storage & Integrity
Nestor Seal	Integrity checks	C15 Integrity: Functions of the archival storage
ISO 16363 <a href="#">Link</a>	Fixity checks	4.4.1.2

#### Other frameworks and reference documents

Reference Document	Term used in framework to refer to the process	Section
OAIS <a href="#">Link</a>	Error checking	4.2.3.4
PREMIS <a href="#">Link</a>	Fixity check	1.5.2, Glossary

## 4. Reference implementations

### Publicly available documentation

Institution	Organisation type	Language	Hyperlink
TIB – Leibniz Information Centre for Science and Technology and University Library, Germany	National library	English	<a href="https://wiki.tib.eu/confluence/spaces/lza/pages/93608391/Preservation+of+data+integrity+as+part+of+the+process+routines">https://wiki.tib.eu/confluence/spaces/lza/pages/93608391/Preservation+of+data+integrity+as+part+of+the+process+routines</a> ; and <a href="https://wiki.tib.eu/confluence/spaces/lza/pages/93608373/Archival+Storage#ArchivalStorage-Integrityassurance">https://wiki.tib.eu/confluence/spaces/lza/pages/93608373/Archival+Storage#ArchivalStorage-Integrityassurance</a>
	Non-commercial digital preservation service		
	Research infrastructure		
	Research performing organisation		
CSC – IT Center for Science Ltd., Finland	Non-commercial digital preservation service	English	<a href="https://digitalpreservation.fi/en/services/quality_reports/2024">https://digitalpreservation.fi/en/services/quality_reports/2024</a>
Archivematica	Digital preservation system	English	<a href="https://www.archivematica.org/en/docs/storage-service-0.23/fixity/#fixity-docs">https://www.archivematica.org/en/docs/storage-service-0.23/fixity/#fixity-docs</a>
AUSSDA - Austrian Social Science Data Archive	Discipline-specific data repository	English	<a href="https://aussda.at/fileadmin/user_upload/p_aussda/Documents/kaczmirek_bischof_2024_preservation_fixity_checks_v1_0-1.pdf">https://aussda.at/fileadmin/user_upload/p_aussda/Documents/kaczmirek_bischof_2024_preservation_fixity_checks_v1_0-1.pdf</a>