Oracle® Fusion Middleware

Release Notes for Oracle Coherence



26 (26.1.0.0.0)
G56836-01
November 2026



Oracle Fusion Middleware Release Notes for Oracle Coherence, 26 (26.1.0.0.0)

G56836-01

Copyright © 2008, 2026, Oracle and/or its affiliates.

Primary Author: Oracle Corporation

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Contents

Preface

Audience i
Documentation Accessibility i
Diversity and Inclusion i
Related Documents i
Conventions ii

1 Introduction

Latest Release Information 1
Purpose of this Document 1
System Requirements and Specifications 1
Certification Information 2
Product Documentation 2
Oracle Support 2
Licensing Information 2
Downloading and Applying Required Patches 2

2 What's New in this Release

New Features 1
Breaking Changes 4
- Java SE 21 Minimum 4
- Jakarta EE 11 Compatibility 4
- Security Defaults and Upgrade Preparation 4
- Java Clients Connecting to Older Servers 6
- Daemon Pool Configuration and Read Quorum Changes 7
- Continuous Aggregation During Rolling Upgrade 7
- Persistence and Transaction Compatibility 7
- Input, Stored-Value and Native Library Compatibility 8
- OpenTracing 8

Deprecated Features 8

Copyright © 2008, 2026, Oracle and/or its affiliates.

3 Known Issues and Workarounds

4 Bugs Fixed and Enhancements in This Release

Copyright © 2008, 2026, Oracle and/or its affiliates.

Preface

Release Notes for Oracle Coherence summarizes the release information related to new and updated features, known issues and their workarounds, deprecated and removed functionality, and more.

This preface includes the following topics:

- Audience
- Documentation Accessibility
- Diversity and Inclusion
- Related Documents
- Conventions

Audience

This document is intended for users of Oracle Coherence.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Related Documents

For more information, see the following documents in the Oracle Coherence documentation set:

Copyright © 2008, 2026, Oracle and/or its affiliates.

- *Administering Oracle Coherence*
- *Administering HTTP Session Management with Oracle Coherence*Web*
- *Developing Applications with Oracle Coherence*
- *Developing Oracle Coherence Applications for Oracle WebLogic Server*
- *Developing Remote Clients for Oracle Coherence*
- *Installing Oracle Coherence*
- *Integrating Oracle Coherence*
- *Managing Oracle Coherence*
- *Securing Oracle Coherence*
- *Java API Reference for Oracle Coherence*
- *.NET API Reference for Oracle Coherence*
- *C++ API Reference for Oracle Coherence*
- *REST API for Managing Oracle Coherence*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Copyright © 2008, 2026, Oracle and/or its affiliates.

1

Introduction

You can use the Oracle Coherence Release Notes to learn about important production information such as Coherence certifications, support, and licensing.
These Release Notes describe Oracle Coherence 26.1.0.0.0, including new capabilities, changes that require application or configuration updates, and selected fixes. Review the requirements and breaking changes before installing the product or starting upgraded members and clients.

Use the certification information for the exact Java, operating system, application server, and client combinations in your deployment. Availability of optional modules depends on the distribution and its supported configuration.

This chapter provides release, certification, documentation, support, licensing, and patch information.

- Latest Release Information
- Purpose of this Document
- System Requirements and Specifications
- Certification Information
- Product Documentation
- Oracle Support
- Licensing Information
- Downloading and Applying Required Patches

Latest Release Information

This document is accurate at the time of publication. Oracle will update the release notes periodically after the software release. You can access the latest information and additions to these release notes at Oracle Help Center.

Purpose of this Document

This document contains the release information for Oracle Coherence.

Oracle recommends you review its contents before installing, or working with the product.

System Requirements and Specifications

Java SE 21 is the minimum Java runtime for this release. Java SE 25 is also supported. Use the certification information for the exact operating system, JDK and application-server combination; upgrading Java alone does not qualify every integration.

Oracle Coherence follows the Fusion Middleware system requirements and certifications for production environments. For more information, see `https://www.oracle.com/middleware/technologies/fusion-certification.html`.

Copyright © 2008, 2026, Oracle and/or its affiliates.

For system requirements for installations of development environments, visit:

- Coherence for Java — *Installing Oracle Coherence for Java*.
- C++ Client — *Installing the C++ Client Distribution*.
- .Net Client — *Installing the .NET Client Distribution*.

Certification Information

To see versions of platforms and related software for which Oracle Coherence is certified and supported, go to `https://www.oracle.com/middleware/technologies/fusion-certification.html`.

Product Documentation

For complete documentation on Oracle Coherence, go to Oracle Help Center.

Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support .

Licensing Information

Detailed information regarding license compliance for Oracle Fusion Middleware is available at Licensing Information.

Downloading and Applying Required Patches

To download and install the latest software patch:

1. Log in to `My Oracle Support` to download the latest software patches.
2. Click the **Patches & Updates** tab.
3. Under the Patch Search tab, select **Product or Family (Advanced Search)**, and select the **Include all patches in a product family** check box.
4. Enter **Oracle Coherence** as the product, select the platform and release, and click **Search**.

   A list of currently available patches for Oracle Coherence is returned.
5. Select the required patch and click **Download**.

   You can check the `README` file in the patch distribution for up-to-date information on the software fixes provided by the patch.

Copyright © 2008, 2026, Oracle and/or its affiliates.

2

What’s New in this Release

Learn about the features, enhancements, and changes made to Oracle Coherence. Oracle updates the release notes periodically after the software release. This document is accurate at the time of publication.

This chapter includes the following sections:

- New Features
- Breaking Changes
- Deprecated Features

New Features

This section describes new and improved capabilities in Oracle Coherence 26.1.0.0.0.

Java SE and Jakarta EE Support for 26.1.0.0.0

- **Java SE 25 support.** This release adds support for Java SE 25 and continues to support Java SE 21. Java SE 21 is the minimum Java version; Java SE 17 is no longer supported. Upgrade Java on members, clients, build tools, and operational utilities before deploying the new libraries.
- **Jakarta EE 11 compatibility.** This release adds support for Jakarta EE 11. Check the certified application-server and Jakarta integration combinations for the modules that you deploy. See Jakarta EE 11 Compatibility.

Security Profiles and Application Policy

The default security profile is `hardened`. It enforces compatibility-sensitive serialization, named executable-class, input, and administrative policies. The separate runtime property `coherence.mode` continues to control runtime and license behavior.

Applications can generate per-artifact `META-INF/coherence/security-config.xml` resources from annotations, including `@Remote.Allowed`, `@Remote.Executable`, and portable-type metadata. Maven and Gradle tooling support policy generation. Generated configuration makes the application's accepted types and executable classes explicit; it does not authenticate a caller or grant business permissions.

Additional controls cover gRPC authentication, transport and serializer selection; REST resources and input; management, metrics, Reporter and health administration; serializer-owned resource limits; and optional RAG administration and imports. Configure authentication, authorization and TLS for each exposed service. See Securing Oracle Coherence and Security Defaults and Upgrade Preparation.

Unified Journal Persistence

Journal persistence adds an append-oriented persistence engine with checkpointing, recovery, compaction, and migration from Berkeley DB (BDB). Choose the mode that fits the backing-map and recovery requirements.

Copyright © 2008, 2026, Oracle and/or its affiliates.

Table 2-1 Journal Persistence Modes

Mode	Use
Side-channel persistence	Use journal persistence with an existing backing map as an explicitly configured alternative to BDB.
Unified primary backing map	Use `<journal-scheme/>` so the journal-backed map and persistence share the primary data representation.
Journal backup map	Use `<backup-storage><type>journal</type></backup-storage>` for journal-backed backup storage. This can be combined with the supported primary modes.

BDB remains the default persistence environment unless journal persistence is explicitly selected. Journal observability includes management statistics, checkpoint operations and Java Flight Recorder events. Follow the supported snapshot-import or rolling-migration procedure when changing persistence formats; see Persistence and Transaction Compatibility.

Virtual Daemon Pools and Platform Pool Improvements

A service can use a virtual daemon pool to execute eligible work on Java virtual threads. Select `<daemon-pool>virtual</daemon-pool>` on that service's scheme. The platform daemon pool remains the default. Virtual execution is particularly useful when service work blocks, but it does not remove contention, downstream limits, or the need to test the application's workload.

Virtual pools expose task, backlog, pool-type and admission statistics through the Service MBean. An optional `<task-limit>` provides an aggregate admission ceiling. Platform-thread sizing elements are not substitutes for virtual-pool admission controls.

Platform daemon pools also improve dynamic resizing and cross-queue worker wakeup. Resizing responds to throughput, utilization and sustained blocking within configured bounds; pools without an explicit maximum derive a safety bound from JVM resources. Idle workers can more promptly steal queued work without changing the default work-slot arrangement.

Review Daemon Pool Configuration and Read Quorum Changes before reusing older virtual-thread configuration or read-quorum settings.

Continuous Aggregation

Continuous Aggregation maintains partition-local partial results as cache entries change. The `InvocableMap.addAggregator` API returns a `ContinuousAggregator<K, V, R>` handle that combines those partials when queried, reducing repeated full scans for suitable workloads. When maintained state is building or cannot provide an answer, Coherence scans the affected partition before returning the result.

Supported families include count, numeric sums and averages, minimum and maximum, distinct values, reducers, Top-N, groups, and supported composites. Custom aggregators require the continuous-maintenance contract. Retained state, mutation cost and result cardinality should be considered when choosing a registration.

Handles also support key-affinity projections, and entry processors can obtain partition-local registered results. Non-transforming `ContinuousQueryCache` views compose their view filter with the registration filter; transforming views do not support this registration. Definitions and eligible maintenance state participate in persistence and ownership changes. StorageManager MBean metrics expose registrations, readiness, fallbacks and rebuilds.

Copyright © 2008, 2026, Oracle and/or its affiliates.

In-cluster, Extend and gRPC Java clients can use the feature. Every storage-enabled member of the cache service must support commercial Coherence 26.1.0.0.0 or the corresponding compatible Community Edition capability. Extend requires NamedCache protocol version 13; gRPC requires NamedCache protocol version 2. See Continuous Aggregation During Rolling Upgrade.

Partition-Streamed Queries and Adaptive Aggregation

Eligible partitioned queries and parallel streaming aggregations use bounded service worker lanes for partition processing. This improves use of available execution capacity while preserving index consistency, authorization, query statistics, timeouts, ownership checks and cleanup.

Explicit `BY_MEMBER` and `BY_PARTITION` aggregator characteristics remain execution contracts. Unqualified parallel aggregators can choose member or partition granularity according to pool capacity and load. These changes do not add a stronger cross-partition snapshot guarantee.

Extend Request Processing

Extend proxies can automatically grow connection-affine selector and decoding pipelines according to connection density and measured pressure. New connections are assigned across pipelines while preserving connection ownership and ordering.

The TCP acceptor's `<pipeline-count>` configuration distinguishes automatic operation, a single legacy pipeline and an explicitly fixed topology. ConnectionManager MBean attributes expose configured, active and limiting pipeline counts. Java Extend initiators also use larger stream buffers to reduce socket and framing overhead without a wire-protocol change.

Operational Diagnostics and Native Images

Journal and FlashJournal diagnostics add visibility into writer, congestion, compaction and file lifecycle behavior. Federation reports include additional transport backlog and journal insertion metrics. These measurements help distinguish execution, network and storage constraints.

The Coherence GraalVM native-image feature supplies Coherence-specific native-image registration support. Use the native-image version, dependencies, configuration and platform supported by the chosen distribution; Java library support alone does not establish native-image certification.

Distributed tracing uses OpenTelemetry. OpenTracing support has been removed; see OpenTracing.

Selected Library Updates

The release updates its third-party dependency baselines. Review application-supplied libraries and optional-module dependencies when upgrading.

Table 2-2 Selected Dependency Versions

Library	Release baseline
Netty	4.1.139.Final
OpenTelemetry	1.64.0
Jersey	4.0.2
Jackson	2.21.5
Log4j 2	2.25.4
gRPC	1.75.0
Protocol Buffers	4.31.1

Copyright © 2008, 2026, Oracle and/or its affiliates.

These are release dependency versions, not a statement of the minimum compatible version for every integration. In particular, supply and configure the appropriate OpenTelemetry runtime for the deployment.

Breaking Changes

Learn about updates in Coherence 26 (26.1.0.0.0) that introduce potentially incompatible changes between Coherence releases.

- Java SE 21 Minimum
- Jakarta EE 11 Compatibility
- Security Defaults and Upgrade Preparation
- Java Clients Connecting to Older Servers
- Daemon Pool Configuration and Read Quorum Changes
- Continuous Aggregation During Rolling Upgrade
- Persistence and Transaction Compatibility
- Input, Stored-Value and Native Library Compatibility
- OpenTracing

Java SE 21 Minimum

Java SE 17 and older runtimes cannot run this release. Upgrade the JDK used by servers, Java clients, command-line tools, build jobs and container images to a supported Java SE 21 or Java SE 25 combination before replacing Coherence libraries. Confirm the complete operating-system and application-server combination using Certification Information.

Jakarta EE 11 Compatibility

Coherence 26.1.0.0.0 adds Jakarta EE 11 compatibility. Align the application's servlet, REST, injection and related integration dependencies with the supported module and container combination. Do not mix incompatible Java EE and Jakarta API variants on the same application class path.

Use the target release's compatibility and certification information to determine the supported application-server versions and migration requirements. See Jakarta EE Compatibility in *Installing Oracle Coherence* and Certification Information.

Security Defaults and Upgrade Preparation

Starting with Coherence 26.1.0.0.0, omitting the security-mode property selects `hardened`. Runtime `coherence.mode=prod` is independent. A blank or unrecognized security-profile value fails configuration resolution.

Copyright © 2008, 2026, Oracle and/or its affiliates.

Table 2-3 Security Upgrade Actions

Area	Changed behavior	Preparation
Global profile	An unset `coherence.security.mode` selects `hardened`.	Prepare the application and configuration before first startup. If a temporary migration phase is needed, explicitly set `-Dcoherence.security.mode=compatibility` on the affected upgraded process before starting it.
Application types and named executables	Hardened checks class allowlists and named remote executable classification at the applicable boundary.	Generate and deploy security configuration on receivers. Classify processors, aggregators, filters, extractors, tasks and other remote executable classes, including nested state and custom lambda targets.
Extend identity and service exposure	The default hardened asserter rejects non-null wire identity tokens; InvocationService and Concurrent Extend proxy defaults are disabled.	Configure a validating identity transformer/asserter chain. Enable an intended remote proxy explicitly with `coherence.invocation.enabled=true` or `coherence.concurrent.extend.enabled=true`, with authentication and authorization prepared.
Management and metrics HTTP	The shipped hardened authentication default is `basic`. Explicit compatibility uses `none`; configured overrides remain possible.	Prepare management clients and scrapers. Set the intended authentication and TLS configuration consistently across endpoint members before upgrade.
Serializer container limits	Hardened defaults are 64 MiB of container bytes, 262144 elements and 131072 map entries at the documented serializer-owned checks.	Test legitimate largest payloads. Configure a reviewed bound where required; these limits are not a universal process allocation budget.
REST and JSON	Hardened REST requires declared resources and approved input. Hardened JSON class metadata requires exact configured aliases.	Declare resources, enable direct queries only where intended, configure expression/type aliases, and verify the consuming marshaller. Package aliases do not satisfy the hardened JSON class-metadata requirement.
gRPC	Authentication and secure transport are explicitly configured. Hardened policy refuses Basic credentials over an insecure network transport and constrains serializer names and diagnostic exposure.	Configure the proxy and client together. Prepare custom serializer allowlists and TLS providers; choosing hardened does not itself provision TLS or an authentication scheme.

Copyright © 2008, 2026, Oracle and/or its affiliates.

Table 2-3 (Cont.) Security Upgrade Actions

Area	Changed behavior	Preparation
TLS and signed identity	Hardened hostname verification and access-controller signing policy are stricter. TLS references retain the intended DNS or numeric identity.	Use certificates whose subject alternative names match the actual endpoint identity. Prepare modern signing keys and trust; test real endpoint names and reconnections.
Reporter and JMX	Hardened policy restricts administrative targets, Reporter sources/output paths and dynamic RMI stub publication.	Approve required report sources and output roots. Configure static JMX publication, including `coherence.management.remote.registryport` when applicable, and verify management tooling.

The current dynamic remote-code default is `allow` in both security profiles. To refuse the paths governed by the shared dynamic gate, explicitly set `coherence.remote.dynamic.unauthenticated=deny`. This includes DYNAMIC lambda bytecode, reflective `MethodInvocationProcessor`, script execution and dynamic remote installs. Allowance does not bypass bytecode, lambda-target, script host-access or caller checks. Some dynamic install routes can be allowed without a named executable-class entry; do not infer dynamic default-deny from the hardened profile.

The default POF context requires registered type identifiers. Use `SafeConfigurablePofContext` only as an explicitly selected compatibility context where needed; `SerializableSerializer` is not registered by the default product POF configuration. Confirm serializers, POF registrations and generated security resources together on each receiver.

JMXMP binds to loopback by default and requires a configured authenticator. Before connector creation it uses an existing JVM-wide object-input filter or installs and verifies its constrained fallback. Review that process-wide effect with other libraries rather than assuming an absent preconfigured filter always prevents startup.

Health mutators remain separately disabled by default. Explicit enablement requires supported authentication. Storage authorization remains an explicitly configured application policy. The hardened profile does not supply business roles, keys, credentials or TLS identities.

If compatibility is used temporarily, collect available shadow diagnostics and test the actual workload under hardened policy before completing migration. A class-policy shadow event is not proof of the final result on every dynamic route, and it does not measure all deserialization or lambda checks. See Securing Oracle Coherence.

Java Clients Connecting to Older Servers

A 26.1 Java client using `compute` or `replaceAll` operations against 14.1.2 or 15.1.1 servers must explicitly select `coherence.security.mode=compatibility` on the client before Coherence initializes. The hardened client can select newer named processor representations that those older servers do not recognize.

Verify the supported client/server and serializer combinations for the deployment, including failover and all APIs in use. Treat client interoperability and the security profile of upgraded 26.1 servers as separate deployment settings.

Copyright © 2008, 2026, Oracle and/or its affiliates.

Daemon Pool Configuration and Read Quorum Changes

The former `<virtual-threads-enabled>` operational element and `coherence.virtualthreads.enabled` properties, including service-specific variants, are removed. The old switch did not provide the new virtual daemon pool. Remove it and select virtual execution per service with `<daemon-pool>virtual</daemon-pool>`. XML containing the removed element fails the updated schema; the removed properties are no longer read.

When selecting a virtual pool, replace incompatible platform-thread sizing settings with the supported virtual-pool admission configuration. The default service pool remains platform-based.

`get` and `getAll` on read-through-capable or sliding-expiry caches are now treated as write-shaped operations for quorum and execution routing because they can commit replicated state. Deployments that configured only read quorum can now see these operations subject to write quorum. Review service quorum settings and custom backing-map classification; do not assume every API named as a read is mutation-free.

Continuous Aggregation During Rolling Upgrade

Continuous Aggregation registration, removal and handle queries require all storage-enabled members of the cache service to advertise the compatible capability. These operations fail with `UnsupportedOperationException` while older storage members remain during upgrade or downgrade. Ordinary cache operations and on-demand aggregation remain available.

Complete the storage-member upgrade and use compatible clients and proxies before registering or querying maintained results. A compatible transport connection alone does not establish feature support. Retain the defining classes and their security configuration for persistence recovery. Transforming `ContinuousQueryCache` views do not support registration.

Persistence and Transaction Compatibility

Changing from BDB to journal persistence is a data-format migration. Use the supported BDB snapshot-import or rolling-restart procedure and keep a tested backup and rollback plan. BDB remains the default unless journal persistence is explicitly configured; do not point an arbitrary new store at old active files and assume format compatibility.

Transaction exception and XID bridges accept narrow supported type sets. The POF XID writer normalizes XIDs to the supported representation; tightened readers reject arbitrary legacy custom serializable XID implementations. Complete or explicitly test prepared XA work that depends on such implementations before mixed-version traffic can reach the new reader.

Transaction message bindings use typed POF fields 4 through 9. New readers do not materialize legacy binding field 2, so a legacy-only sender can yield empty bindings. Coordinate binding-dependent transaction traffic during rolling upgrade. Supported exception graphs are bounded, including a 16384-element bridge array ceiling; translate or trim unsupported state before transmission.

Persisted topic subscriber executable metadata is rechecked on replay. Hardened policy defaults to rejecting executable-policy drift; `coherence.topics.persisted.policy-drift=warn-allow` is a deliberate migration option. Keep the metadata's classes and executable policy available on recovery members.

Copyright © 2008, 2026, Oracle and/or its affiliates.

Input, Stored-Value and Native Library Compatibility

Malformed, overflowing or truncated serialized containers and native index inputs are rejected at the documented boundaries. Memcached-origin stored values no longer permit arbitrary serialized object terminal forms through binary pass-through; the supported scalar/binary forms and the narrow configured-serializer interoperability path remain available.

Use the HNSW native binary supplied for the matching Java module, operating system and architecture. The macOS x86_64 HNSW library is no longer supplied. Do not copy an older native library into a newer installation to bypass input validation or platform availability.

Optional RAG deployments must prepare supported authentication, administrator roles and explicit import/model policy. Hardened model download rejects an empty model allowlist. HTTP imports use the approved direct transport and validate redirect destinations; selected proxies are refused. Local imports require supported secure directory traversal and protected roots. Consult the security guide for the exact optional-module controls.

OpenTracing

OpenTracing integration is removed in Coherence 26.1.0.0.0. Replace OpenTracing dependencies and initialization with the supported OpenTelemetry integration.

Coherence does not supply a tracing implementation for the application's runtime. Provide the supported OpenTelemetry dependencies and configure initialization and export for the deployment. Do not treat the old OpenTracing libraries as a compatibility switch.

See Distributed Tracing in *Developing Applications with Oracle Coherence*.

Deprecated Features

Deprecated features are distinct from features removed in this release. Deprecated features may still be available but subject to their documented restrictions.

Table 2-4 Removed Features and Replacements

Removed item	Replacement or action
Java SE 17 runtime support	Use a supported Java SE 21 or Java SE 25 combination. See Java SE 21 Minimum.
OpenTracing integration	Use OpenTelemetry. See OpenTracing.
`<virtual-threads-enabled>` and `coherence.virtualthreads.enabled`	Use the per-service `<daemon-pool>` configuration. See Daemon Pool Configuration and Read Quorum Changes.

Review the deprecation notices in the Java API reference for individual APIs and the certification information for supported runtime and container combinations. Do not interpret a removed runtime or integration as a still-supported deprecated option.

Copyright © 2008, 2026, Oracle and/or its affiliates.

3

Known Issues and Workarounds

Learn about any known issues at the time of release.
Review the latest Release Notes, the README supplied with the installed patch, and My Oracle Support for issues and workarounds that apply to the exact release, platform and configuration.

When reporting an issue, include the Coherence and Java versions, operating system, enabled optional modules, client/server topology, relevant configuration and a reproducible description. For a rolling upgrade, include the versions and selected security profiles of both clients and members.

See Latest Release Information , Oracle Support and Downloading and Applying Required Patches.

Planned migration restrictions are described in Breaking Changes.

Copyright © 2008, 2026, Oracle and/or its affiliates.

4

Bugs Fixed and Enhancements in This Release

Learn about the bugs fixed and enhancements in this release.
This section lists bugs fixed and enhancements in Oracle Coherence 26.1.0.0.0, grouped by area. Some fixes may also be available in maintenance releases. Consult the README included with the patch installation.

Security policy and compatibility changes are described in Breaking Changes. Review those preparation steps even if the workload did not encounter one of the issues listed here.

Cache Services, Queries, and Indexes

Table 4-1 Cache Services, Queries, and Indexes Bugs Fixed and Enhancements

Bug Number	Description
40160353	Fixed an issue where distributed grouping operations using Java serialization could fail when their results contained sorted sets or `Optional` values.
40128933	Fixed an issue where `AsyncNamedCache` operations could throw a `NullPointerException` after a failed cache restart, masking the original error and preventing subsequent recovery.
40072370	Coherence distributed queries and streaming aggregators now use bounded, daemon-pool-coordinated partition execution instead of member-local `ForkJoinPool` fan-out and blocking joins. Eligible queries stream partition responses progressively, while streaming aggregators preserve one member-level response and choose partition or member execution according to explicit `BY_PARTITION`/`BY_MEMBER` contracts or current platform-daemon-pool pressure. This improves indexed-query throughput and substantially reduces aggregation latency below the service saturation knee without adding overload pressure when the service is busy, while preserving protocol compatibility and existing fallback behavior.
39942912	Fixed an issue where `SimpleMapIndex` could overcount or undercount its memory footprint after repeated updates to collection-valued index entries. Footprint accounting now remains accurate as inverse mappings change or internal collections expand.
39635940	Fixed an issue where interrupted cache operations could leave partition ownership processing stuck, causing repeated partition transfer attempts and potentially requiring affected members to be restarted.
39542054	Fixed an issue where a cache configured with write-behind remove could report a negative size after entries were removed.
39089627	Fixed an issue where partition migration may be unable to complete in some rare circumstances.
39023805	Improved range filters performance via direct forward index evaluation.
39023803	Fixed ArrayFilter memory leak that could cause OutOfMemoryError.
38914458	Fixed parallel query execution issue that could cause some partitions to be silently dropped from the query results.
38748319	Fixed an issue where a partitioned cache service could terminate with a `ClassCastException` involving `MultiBufferWriteBuffer` and `BufferSequence`.
38582916	Fixed an issue where an index may not be created for one or more partitions after recovery from persistence.

Copyright © 2008, 2026, Oracle and/or its affiliates.

Table 4-1 (Cont.) Cache Services, Queries, and Indexes Bugs Fixed and Enhancements

Bug Number	Description
38506522	Fixed a performance regression due to excessive calls to construct DoAsAction instances.
38231092	Fixed an issue where the PartitionedCache partition assignment strategy may try to lower the backup strength below NODE_SAFE resulting in a log message reporting an IllegalStateException.
37388347	Fixed an issue where a view-scheme without a back-scheme did not pick up default configuration values for the embedded distributed-scheme, for example the default serializer could be used instead of the configured default serializer.

Coherence Concurrent and Daemon Pools

Table 4-2 Coherence Concurrent and Daemon Pools Bugs Fixed and Enhancements

Bug Number	Description
40128719	Fixed an issue where cancelled Coherence Executor tasks could leave their futures incomplete, causing result retrieval to time out or report an incorrect exception. Cancelled futures now complete promptly and correctly report cancellation.
39955738	Platform daemon pools now adapt worker concurrency to workload behavior. CPU-bound services remain near the available processor count, while blocking and I/O-bound services can grow beyond it when measured throughput demonstrates a benefit. Ineffective growth is automatically reversed, improving throughput and latency under blocking workloads without adding unnecessary workers to CPU-bound services.
39820969	Fixed an issue where TCP proxy worker pools configured without a maximum thread count could grow without bound under load, potentially exhausting native thread resources and destabilizing cluster members. Unbounded daemon pools now share a JVM-wide worker budget while honoring configured minimum and explicit maximum thread counts.
39363219	Coherence Concurrent now enables Extend heartbeats by default for concurrent proxy/client connections. This improves detection of failed or abruptly disconnected Extend clients and helps release client-owned concurrent resources, such as remote locks, more promptly. The defaults can be overridden with `coherence.concurrent.extend.heartbeat.interval` and `coherence.concurrent.extend.heartbeat.timeout`.
39256499	Improved the legacy Coherence DaemonPool load-balancing behavior for skewed workloads by waking idle workers pool-wide so they can steal work from busy queues. This reduces backlog and tail latency without changing the default work-slot configuration or collapsing all work onto a single queue.
33748696	Coherence daemon pool dynamic resizing has been improved to react more quickly to sudden increases in queued work. The resize algorithm now uses throughput and utilization signals to grow worker pools more aggressively during bursts, while shrinking linearly and with cooldown to avoid oscillation once load subsides. For platform-thread daemon pools with no explicit thread-count-max, Coherence now derives a safe effective maximum from JVM heap and thread stack sizing, with a built-in ceiling, to prevent accidental unbounded thread growth under heavy backlog conditions. Explicitly configured finite daemon pool limits continue to be honored.

Copyright © 2008, 2026, Oracle and/or its affiliates.

Coherence*Web

Table 4-3 Coherence*Web Bugs Fixed and Enhancements

Bug Number	Description
39751078	Fixed an issue where Coherence*Web session reaper tasks could repeatedly fail after a shared session cache was released.
39668710	Fixed a sticky-session registry race during parallel Coherence*Web application deployment.

Dependencies and Packaging

Table 4-4 Dependencies and Packaging Bugs Fixed and Enhancements

Bug Number	Description
40257416	Updated ASM to version 9.10.1.
40257412	Updated ONNX Runtime to version 1.25.0.
40257363	Updated GraalVM component library to version 25.0.5.0.2.
40197619	Updated ClassGraph to version 4.8.199.
40135638	Updated Netty to version 4.1.139.Final.
40047180	Updated Gradle to version 9.7.1.
39765709	Updated Jackson to version 2.21.5.
39765472	Updated OpenTelemetry to version 1.64.0.
39463975	Removed the macOS x86_64 HNSW native library from the coherence-hnsw module.
39311129	Fixed an issue where the Coherence POF Maven Plugin patch artifacts could reference SNAPSHOT versions in their Maven metadata, which could cause repository synchronization or publication failures.
39298054	Updated Log4J 2 to version 2.25.4.
39120403	Updated Jersey to version 4.0.2.
38643440	Reduced the size of Coherence container images.

Federation

Table 4-5 Federation Bugs Fixed and Enhancements

Bug Number	Description
40087001	Fixed an issue where federation could become stuck while resuming from the STOPPED state on a cluster member that owns no partitions.
40071524	Fixed an issue where federation could enter the ERROR state when an internal cache request timed out under load. Federation now remains active and retries after a delay.
39792340	Added support in MessageBus for distinguishing peers that advertise the same endpoint, enabling federation between independent clusters that reuse hostnames and listener ports. This feature is currently enabled only for federation connections.
39602396	Added transport backlog, connection, retry/error, replication progress, and interceptor metrics to the built-in federation destination and origin reports, along with service and participant identifiers for easier filtering and diagnosis.

Copyright © 2008, 2026, Oracle and/or its affiliates.

Table 4-5 (Cont.) Federation Bugs Fixed and Enhancements

Bug Number	Description
39531288	Federation destination monitoring now reports the number of journal records, cache entries, and serialized journal-record bytes inserted into the local JournalRecord cache for each outbound destination. These counters are exposed as destination MBean attributes, Coherence metrics, and federation destination report columns to help diagnose journal backlog and outbound replication volume before records are sent.
39235738	Fixed an issue in federation where a service context update message could block a SelectionService thread.
39162248	Fixed an issue in federation where a DestinationController worker thread could be abandoned after a remote participant connection timeout. The controller now stays guarded and can continue the existing disconnect/retry recovery flow after slow endpoint resolution or connection timeouts.
39071060	Added a log message and enhanced the federation destination MBean ErrorDescription to include additional diagnostic state when journalcache-highunits is reached or when the Elastic Data journal is full.
39070998	Added originating member information to the federation error description MBean attribute.
38296786	Added the remote participant name and network connection information to key federation log messages where this information was missing.
38187194	Federation now resumes sending data sooner when a destination is recovering from a backlog, instead of waiting for all queued work at the destination to finish. This can improve federation throughput during periods of heavy replication load by reducing repeated pauses in data transfer.
38172539	Added federation destination MBean and management over REST operation disconnect() which disconnects this member from the remote participant and triggers a reconnect.

Management, Metrics, and Tracing

Table 4-6 Management, Metrics, and Tracing Bugs Fixed and Enhancements

Bug Number	Description
40109926	Fixed an issue where Coherence Reporter performed redundant MBean queries for wildcard column joins, which could cause slow report execution in environments with many MBeans. Reporter now reuses MBean name resolutions within each report execution to reduce query overhead.
39842031	Fixed an issue where Coherence MicroProfile Metrics cache gauges for front and back tiers used inconsistent tag sets.
39663704	Fixed a startup deadlock between Coherence MP Metrics and Helidon 4 metrics initialization.
39081492	Fixed an issue where the Management over REST cache endpoint could report an inflated cache size for near caches
38884104	Added support for enabling or disabling the health check service and endpoint via the coherence.health.http.enabled system property.
38660649	Fixed an issue where a ClassCastException may be thrown when initializing Coherence Management in some rare scenarios.
38294912	OpenTracing support, deprecated since Coherence 14.1.2, has been removed.

Copyright © 2008, 2026, Oracle and/or its affiliates.

Networking and Coherence*Extend

Table 4-7 Networking and Coherence*Extend Bugs Fixed and Enhancements

Bug Number	Description
40088069	Fixed an issue where multiple services sharing an address-provider configuration could interfere with each other's address selection, causing connection failures.
40061955	Fixed an issue where cluster shutdown could return before the NameService had fully stopped, potentially causing a subsequent cluster startup in the same JVM to use stale configuration.
39968019	Fixed an issue where the Coherence cache configuration schema rejected supported `incoming-buffer-pool` and `outgoing-buffer-pool` settings for TCP acceptors.
39955740	Coherence Extend TCP proxies now process independent client connections through connection-affine pipelines instead of serializing all request decoding on a single acceptor thread. Each connection retains strict request ordering, while multiple connections are decoded concurrently, substantially improving proxy throughput and latency under load. Pipeline concurrency is selected automatically based on available processors and can be overridden using the new `<connection-pipeline-count>` TCP acceptor setting.
39955264	Improved Coherence*Extend client performance by increasing the default TCP initiator input and output stream buffers from 5 bytes to 8 KiB. The larger buffers reduce socket system calls and protocol-framing overhead, improving throughput and CPU efficiency without changing the Extend wire protocol, configuration, or connection behavior. This change adds approximately 16 KiB of heap usage per physical Extend client connection.
39765047	Fixed an issue where Message Bus connections could disconnect or fail to recover when socket migration overlapped with active traffic or simultaneous peer migration.
39634552	Fixed an issue where a SelectionService thread could CPU spin indefinitely in rare cases, potentially causing MessageBus ack timeouts, connection migrations, or service disruption under load.
39042323	Eliminated lock contention on the MessageBus connection during send() operation
38668667	Fixed an issue where a thread for a TLS enabled proxy service may be stuck in a CPU busy loop.

Persistence and Elastic Data

Table 4-8 Persistence and Elastic Data Bugs Fixed and Enhancements

Bug Number	Description
39760736	Added FlashJournal observability for write and force operations, congestion, compaction, and journal-file lifecycle activity. The new statistics are available through JMX, Management over REST, Coherence Metrics, and Reporter output.
39272594	A new journal-based persistence engine is available as an alternative to Berkeley DB (BDB) JE. Three configuration modes are supported, ranging from a drop-in BDB replacement to a fully unified backing-map and persistence model that eliminates the double-write penalty of the existing architecture. A JMX MBean and JFR event surface expose extent state, compaction progress, and per-phase recovery timings.
38981329	Fixed an issue where an Elastic Data daemon thread may be terminated by the guardian in some rare circumstances.
38981216	Elevated to ERROR level the log message that indicates that the ElasticData CollecterDaemon has been abnormally terminated. The CollectorDaemon is responsible for deleting files which are no longer being used by Elastic Data's FlashJournal.

Copyright © 2008, 2026, Oracle and/or its affiliates.

Security and Serialization

Table 4-9 Security and Serialization Bugs Fixed and Enhancements

Bug Number	Description
40184124	Security hardened mode is now enabled by default. Applications that require compatibility mode can explicitly enable it by setting `coherence.security.mode=compatibility`.
40033112	Fixed an issue where a request payload rejected during deserialization could cause parsing to continue from an invalid stream position, resulting in stream corruption and cache-service termination. Coherence now stops processing the affected request and returns the original deserialization failure to the caller, preserving cache-service availability.
39345752	Coherence adds a security hardening framework for serialization and remote execution paths. Hardened mode is enabled by default when `coherence.security.mode` is unset. In hardened mode, Coherence applies stricter validation for remote payloads, dynamic lambdas, POF user types, and remote executable objects. The framework includes generated `META-INF/coherence/security-config.xml` allowlists, `@Remote.Allowed`, `@Remote.Executable`, Maven and Gradle security-config generation, stricter POF type handling, dynamic class and lambda bytecode validation, and route-aware serialization gate telemetry. Applications that require compatibility mode can explicitly enable it by setting `coherence.security.mode=compatibility` before the first upgraded startup. The `coherence.security.mode` setting is separate from the existing `coherence.mode` dev/prod semantics.

Topics

Table 4-10 Topics Bugs Fixed and Enhancements

Bug Number	Description
40114150	Fixed an issue where PagedTopic subscribers could stop receiving messages and lose channel allocations after overlapping cluster member restarts. Subscribers now ignore stale channel-allocation notifications and automatically recover pending receive requests after reconnecting.
39945811	Fixed an issue where POF-serialized PagedTopic subscription metadata could grow continuously after repeated subscription updates. The accumulated duplicate data could increase processing and memory usage and eventually cause a StreamCorruptedException during deserialization. Subscription metadata now remains stable across repeated serialization while preserving data written by future versions.
39555883	Fixed an issue where PagedTopic subscriber initialization could time out after 30 seconds even when a longer subscriber timeout was configured. This could cause intermittent failures in topic-backed workflows such as Coherence RAG document ingestion under slower startup or loaded environments.
39327816	Fixed several paged topic subscriber lifecycle races involving channel reassignment, disconnect, close, and ownership loss. These fixes prevent subscribers from resuming at incorrect positions or skipping uncommitted topic messages during subscriber churn.
39255287	Fixed an issue in paged topics where closing a subscriber could hang indefinitely while waiting for subscriber cleanup to complete. Subscriber close now respects the configured close timeout and will no longer block forever if cleanup state does not converge.

Copyright © 2008, 2026, Oracle and/or its affiliates.