



Hibernate Validator 9.1.0.Final - Jakarta Validation Reference Implementation

Migration Guide from 9.0

2025-11-07

Table of Contents

Introduction 1

Requirements..... 2

Artifacts 3

Configuration 4

API..... 5

SPI..... 6

Behavior 7

Introduction

The aim of this guide is to assist you migrating an existing application using any version 9.0.x of Hibernate Validator to the latest of the 9.1.x series.



If you think something is missing or something does not work, please [contact us](#).

If you're looking to migrate from an earlier version, you should migrate step-by-step, from one minor version to the next, following the migration guide of [each version](#).



As this release includes significant changes to the processed bean tracking, path implementation and related areas, we encourage users to give it a try and report any findings, especially if there are non-trivial validation scenarios.

Requirements

The requirements of Hibernate Validator 9.1 are the same as those of Hibernate Validator 9.0.

Artifacts

The coordinates of Maven artifacts in Hibernate Validator 9.1 are the same as in Hibernate Validator 9.0.

Configuration

The configuration properties in Hibernate Validator 9.1, in general, are backward-compatible with Hibernate Validator 9.0.

API

The [API](#) in Hibernate Validator 9.1 is, in general, backward-compatible with Hibernate Validator 9.0.

SPI

The [SPI](#) in Hibernate Validator 9.1 is, in general, backward-compatible with Hibernate Validator 9.0.

Behavior

The behavior of Hibernate Validator 9.1 is, in general, backward-compatible with Hibernate Validator 9.0.

Keep in mind that the internal representation of the `Path` and `Node` became mutable during validation process, so do not rely on these being immutable in your traversable resolvers. The `Path` and `Node` representations within the `ConstraintViolation` remains immutable.