Introduction

The aim of this guide is to assist you migrating an existing application using any version 6.0.x of Hibernate Search to the latest of the 6.1.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.

To Hibernate Search 5 users

Be aware that a lot of APIs have changed since Hibernate Search 5, some only because of a package change, others because of more fundamental changes (like moving away from using Lucene types in Hibernate Search APIs).

When migrating from Hibernate Search 5, you are encouraged to migrate first to Hibernate Search 6.0 using the 6.0 migration guide, and only then to later versions (which will be significantly easier).
Requirements

Hibernate Search 6.1.8.Final now requires using Hibernate ORM versions from the 5.6.x family.
Data format and schema changes

Indexes created with Hibernate Search 6.0 can be read from and written to with Hibernate Search 6.1.8.Final.
Configuration changes

The configuration properties are backward-compatible with Hibernate Search 6.0.

Some configuration properties API have been deprecated, and will be removed in the next major version:

- `hibernate.search.automatic_indexing.strategy`: use `hibernate.search.automatic_indexing.enabled` instead, passing either `true` or `false`. 
API changes

The API is backward-compatible with Hibernate Search 6.0.

Parts of the API have been deprecated, and will be removed in the next major version:

- `FromDocumentFieldValueConverter`: implement `FromDocumentValueConverter` instead.
- `ToDocumentFieldValueConverter`: implement `ToDocumentValueConverter` instead.
- `ElasticsearchBackendSettings.Defaults#MULTITENANCY_STRATEGY`, `LuceneBackendSettings.Defaults#MULTITENANCY_STRATEGY`: the default for these properties is now dynamic. If the multi-tenancy is enabled in the mapper, the default is `MultiTenancyStrategyName#DISCRIMINATOR`; Otherwise, the default is still `MultiTenancyStrategyName#NONE`. 
SPI changes

The SPI is mostly backward-compatible with Hibernate Search 6.0.

Below are the most notable SPI changes:

- `org.hibernate.search.backend.elasticsearch.client.spi.ElasticsearchHttpClientConfigurer` moved to `org.hibernate.search.backend.elasticsearch.client.ElasticsearchHttpClientConfigurer` and is now API.
- `org.hibernate.search.backend.elasticsearch.client.spi.ElasticsearchHttpClientConfigurationContext` moved to `org.hibernate.search.backend.elasticsearch.client.ElasticsearchHttpClientConfigurationContext` and is now API.
- `org.hibernate.search.engine.common.timing.spi.Deadline` moved to `org.hibernate.search.engine.common.timing.Deadline` and is now API.
- `URLEncodedString#fromJsonString` was removed.
- `FieldPaths#absolutize(String, String, String)` was removed.
- `IndexManagerImplementor#createIndexingPlan` no longer expects an `EntityReferenceFactory` parameter, but `IndexIndexingPlan#executeAndReport` does.
- API changes around `ToDocumentFieldValueConverter`/`FromDocumentFieldValueConverter` led to more SPI changes. See https://github.com/hibernate/hibernate-search/pull/2611.
- The document model SPI for backend implementations changed significantly:
  - `org.hibernate.search.engine.backend.document.model.dsl.spi.IndexSchemaResolution`
ootNodeBuilder is now org.hibernate.search.engine.backend.document.model.dsl.spi.IndexRootBuilder.

- Implementations should rely on abstract classes provided as part of the SPI, for example org.hibernate.search.engine.backend.document.model.spi.AbstractIndexModel.

- Etc. See https://github.com/hibernate/hibernate-search/pull/2591 for an example of how to migrate backend implementations.

- The entry points of the search DSL SPI for backend implementations changed significantly. See https://github.com/hibernate/hibernate-search/pull/2591 and https://github.com/hibernate/hibernate-search/pull/2592 for an example of how to migrate backend implementations.

Parts of the SPI have been deprecated, and will be removed in the next major version:

- ElasticsearchAwsCredentialsProvider: implement ElasticsearchAwsCredentialsProvider instead.

- PojoAdditionalMetadataCollectorTypeNode#markAsEntity(String, org.hibernate.search.mapper.pojo.model.path.spi.PojoPathsDefinition): use PojoAdditionalMetadataCollectorTypeNode#markAsEntity(String, org.hibernate.search.mapper.pojo.model.path.spi.PojoPathDefinitionProvider) instead.
Behavior changes

No behavior changes to report.