

Getting Started with Infinispan

Table of Contents

1. Get Up and Running with Infinispan	1
1.1. Software Requirements	1
1.2. JDK	1
1.3. Maven	1
1.4. Getting Infinispan	1
1.5. Infinispan Tutorials	1
2. Creating your own Infinispan project	3
2.1. Maven Archetypes	3
2.1.1. Starting a new project	3
2.1.2. Playing with your new project	3
2.1.3. On the command line... ..	3
2.1.4. Writing a test case for Infinispan	3
2.1.5. On the command line... ..	4
2.1.6. Versions	4
2.1.7. Source Code	5

Chapter 1. Get Up and Running with Infinispan

It's easy to get started with Infinispan. All you need is Java, dependency management tooling, and either the Infinispan server or core libraries.

1.1. Software Requirements

To run Infinispan, you'll need:

- Java 1.8 JDK at a minimum. Infinispan recommends JDK 11 but also works with JDK 14.
- Maven 3.2 or later. Infinispan also works with Gradle and Ivy.

1.2. JDK

Choose your Java runtime and follow their installation instructions. For example, you could choose:

- [OpenJDK](#)
- [Oracle Java SE](#)

1.3. Maven

Run `mvn --version` to check your current Maven version. If you have Maven 3.2 or later, you are ready to go.

If you need to install Maven, visit <https://maven.apache.org/install.html>.

1.4. Getting Infinispan

To start using Infinispan, you can:

- Download the Infinispan server distribution. Visit the [Download Infinispan](#) page to get a stable or development version.
- Pull the [Infinispan server image](#).
- Add the `infinispan-core` artifact to your `pom.xml` and use Infinispan directly in your project. See [Embedding Infinispan](#) for more information.

1.5. Infinispan Tutorials

Infinispan provides a set of tutorials that demonstrate various capabilities in a few easy steps.

Clone the tutorial repository as follows:

```
$ git clone git@github.com:infinispan/infinispan-simple-tutorials.git
```

Chapter 2. Creating your own Infinispan project

2.1. Maven Archetypes

Infinispan currently has 2 separate Maven [archetypes](#) you can use to create a skeleton project and get started using Infinispan. This is an easy way to get started using Infinispan as the archetype generates sample code, a sample Maven pom.xml with necessary dependencies, etc.



You don't need to have any experience with or knowledge of Maven's Archetypes to use this! Just follow the simple steps below.

2.1.1. Starting a new project

Use the newproject-archetype project. The simple command below will get you started:

```
$ mvn archetype:generate \
  -DarchetypeGroupId=org.infinispan.archetypes \
  -DarchetypeArtifactId=newproject-archetype \
  -DarchetypeVersion=1.0.23 \
  -DarchetypeRepository=http://repository.jboss.org/nexus/content/groups/public
```

You will be prompted for a few things, including the *artifactId*, *groupId* and *version* of your new project. And that's it - you're ready to go!

2.1.2. Playing with your new project

The skeleton project ships with a sample application class, interacting with Infinispan. You should open this new project in your IDE - most good IDEs such as IntelliJ and Eclipse allow you to import Maven projects, see [this guide](#) and [this guide](#). Once you open your project in your IDE, you should examine the generated classes and read through the comments.

2.1.3. On the command line...

Try running

```
$ mvn install -Prun verify
```

in your newly generated project! This runs the main() method in the generated application class.

2.1.4. Writing a test case for Infinispan

This archetype is useful if you wish to contribute a test to the Infinispan project and helps you get set up to use Infinispan's testing harness and related tools. Use

```
$ mvn archetype:generate \
  -DarchetypeGroupId=org.infinispan.archetypes \
  -DarchetypeArtifactId=testcase-archetype \
  -DarchetypeVersion=1.0.23 \
  -DarchetypeRepository=http://repository.jboss.org/nexus/content/groups/public
```

As above, this will prompt you for project details and again as above, you should open this project in your IDE. Once you have done so, you will see some sample tests written for Infinispan making use of Infinispan's test harness and testing tools along with extensive comments and links for further reading.

2.1.5. On the command line...

Try running

```
$ mvn test
```

in your newly generated project to run your tests.

The generated project has a few different profiles you can use as well, using Maven's -P flag. E.g.,

```
$ mvn test -Pudp
```

Available profiles

The profiles available in the generated sample project are:

- `udp`: use UDP for network communications rather than TCP
- `tcp`: use TCP for network communications rather than UDP
- `jbosstm`: Use the embedded [JBoss Transaction Manager](#) rather than Infinispan's embedded transaction manager

Contributing tests back to Infinispan

If you have written a functional, unit or stress test for Infinispan and want to contribute this back to Infinispan, your best bet is to [fork the Infinispan sources on GitHub](#) . The test you would have prototyped and tested in an isolated project created using this archetype can be simply dropped in to Infinispan's test suite. Make your changes, add your test, prove that it fails even on Infinispan's upstream source tree and issue a [pull request](#) .

2.1.6. Versions

The archetypes generate poms with dependencies to specific versions of Infinispan. You should edit these generated poms by hand to point to other versions of Infinispan that you are interested in.

2.1.7. Source Code

The source code used to generate these archetypes are [on GitHub](#) . If you wish to enhance and contribute back to the project, fork away!