

**AXIOM JDK**

# **Axiom ActiveMQ**

## **Инструкция по эксплуатации**

Axiom JDK | Май 2026

Copyright © 2019-2026 Все права защищены АО "АКСИОМ" (АКСИОМ)

Программное обеспечение АКСИОМ содержит программное обеспечение с открытым исходным кодом. Дополнительная информация о коде сторонних разработчиков доступна на сайте [https://axiomjdk.ru/third\\_party\\_licenses](https://axiomjdk.ru/third_party_licenses). Для дополнительной информации о том, как получить копию исходного кода, можно обратиться по адресу [info@axiomjdk.ru](mailto:info@axiomjdk.ru).

ДАННАЯ ИНФОРМАЦИЯ МОЖЕТ ИЗМЕНЯТЬСЯ БЕЗ ПРЕДВАРИТЕЛЬНОГО УВЕДОМЛЕНИЯ. АКСИОМ ПРЕДОСТАВЛЯЕТ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ "КАК ЕСТЬ" БЕЗ КАКИХ-ЛИБО ГАРАНТИЙ, АКСИОМ ПРЯМО ОТКАЗЫВАЕТСЯ ОТ ВСЕХ ПОДРАЗУМЕВАЕМЫХ ГАРАНТИЙ, ВКЛЮЧАЯ, НО НЕ ОГРАНИЧИВАЯСЬ ПОДРАЗУМЕВАЕМЫМИ ГАРАНТИЯМИ ТОВАРНОЙ ПРИГОДНОСТИ И ПРИГОДНОСТИ ДЛЯ ОПРЕДЕЛЕННОЙ ЦЕЛИ.

АКСИОМ НИ ПРИ КАКИХ ОБСТОЯТЕЛЬСТВАХ НЕ НЕСЕТ ОТВЕТСТВЕННОСТИ ЗА ЛЮБЫЕ КОСВЕННЫЕ, СЛУЧАЙНЫЕ, СПЕЦИАЛЬНЫЕ, ШТРАФНЫЕ ИЛИ КОСВЕННЫЕ УБЫТКИ, ИЛИ УБЫТКИ ОТ ПОТЕРИ ПРИБЫЛИ, ДОХОДА, ДАННЫХ ИЛИ ИСПОЛЬЗОВАНИЯ ДАННЫХ, ПОНЕСЕННЫЕ ВАМИ ИЛИ ЛЮБОЙ ТРЕТЬЕЙ СТОРОНОЙ, БУДЬ ТО В РЕЗУЛЬТАТЕ ДЕЙСТВИЯ ДОГОВОРА ИЛИ ДЕЛИКТА, ДАЖЕ ЕСЛИ АКСИОМ БЫЛО ПРЕДУПРЕЖДЕНО О ВОЗМОЖНОСТИ ТАКИХ УБЫТКОВ.

Использование любого программного продукта АКСИОМ регулируется соответствующим лицензионным соглашением, которое никоим образом не изменяется условиями данного уведомления. Программные продукты и фирменные наименования: Axiom JDK, Axiom JDK Pro, Axiom Runtime Container Pro, Axiom Linux, Libercat, Libercat Certified и АКСИОМ принадлежат АКСИОМ и их использование допускается только с разрешения правообладателя.

Товарный знак Linux® используется в соответствии с сублицензией от Linux Foundation, эксклюзивного лицензиата Линуса Торвальдса, владельца знака на всемирной основе. Java и OpenJDK являются товарными знаками или зарегистрированными товарными знаками компании Oracle и/или ее аффилированных лиц. Другие торговые марки являются собственностью их соответствующих владельцев и используются только в целях идентификации.

# Содержание

1. Введение .....	4
1.1. Что такое Axiom ActiveMQ? .....	4
1.2. Основные возможности Фреймворка .....	4
1.3. Некоторые преимущества Фреймворка .....	4
2. Начало работы с Фреймворком "ActiveMQ" .....	6
2.1. Системные требования .....	6
2.2. Настройка среды разработки .....	6
2.3. Установка JDK .....	6
2.4. Установка ActiveMQ .....	7
2.5. Установка Apache Ant .....	7
2.5.1. Инструменты сборки .....	8
2.6. Редактирование и запуск приложения ActiveMQ .....	11
2.7. Запуск Axiom ActiveMQ .....	12
2.8. Проверка состояния ActiveMQ .....	12
2.9. Остановка ActiveMQ .....	12
2.10. Проверка веб-консоли .....	12
2.11. Защищенная веб-консоль .....	12

# 1. Введение

Данный документ дает базовую информацию для работы с программным изделием Axiom ActiveMQ.

## 1.1. Что такое Axiom ActiveMQ?

Axiom ActiveMQ – это брокер сообщений с открытым исходным кодом, написанный на Java. Он полностью соответствует стандартам JMS 1.1. Разработка и поддержка осуществляются Apache Software Foundation, лицензия – Apache. Он обеспечивает высокую доступность, масштабируемость, надежность, производительность и безопасность для корпоративных приложений обмена сообщениями.

JMS – это спецификация, позволяющая разрабатывать системы, основанные на обмене сообщениями. ActiveMQ выступает в качестве брокера сообщений, который находится между приложениями и позволяет им обмениваться данными асинхронным и надежным способом.

## 1.2. Основные возможности Фреймворка

- Работающий брокерский сервер - возможность проверить очереди, топики, подключения и т. д.
- Консоль администратора - возможность использовать консоль администратора для управления очередями, топиками, подписчиками, подключениями, сетью и т. д.
- Приложение-производитель - возможность отправлять сообщения в очередь ActiveMQ.
- Приложение-потребитель - возможность получать сообщения из очереди ActiveMQ.

## 1.3. Некоторые преимущества Фреймворка

ActiveMQ разработан для обеспечения высокой доступности, масштабируемости, надежности, производительности и безопасности корпоративных приложений обмена сообщениями. Ниже перечислены некоторые из основных особенностей ActiveMQ.

- Соответствие стандарту JMS – ActiveMQ полностью соответствует стандарту JMS 1.1. Спецификация JMS предоставляет стандартный механизм для синхронной или асинхронной доставки сообщений, однократной доставки сообщений, обеспечения надежности сообщений для подписчиков и т. д.

- Возможности подключения – ActiveMQ поддерживает HTTP/S, многоадресную рассылку, SSL, STOMP, TCP, UDP, XMPP, предоставляя таким образом широкий спектр вариантов подключения и позволяя различным системам обмениваться данными, используя выбранные ими протоколы.
- Подключаемая архитектура – ActiveMQ позволяет выбирать механизм сохранения данных, а также предоставляет возможности для настройки безопасности аутентификации и авторизации в соответствии с потребностями приложения.
- Многоплатформенность – ActiveMQ предоставляет клиентские API для многих популярных языков, таких как Java, C, C++, .NET, Perl, PHP, Python, Ruby и др. Брокер ActiveMQ будет работать в JVM, но клиенты могут быть написаны на любом из поддерживаемых языков.
- Кластер брокеров ActiveMQ позволяет подготовить сеть брокеров для масштабируемости и может поддерживать различные типы топологий.
- Богатый функционал – ActiveMQ предоставляет множество расширенных возможностей как для брокера, так и для клиентов, а также поддерживает Apache Camel.
- Простой интерфейс администрирования – консоль администрирования ActiveMQ проста в использовании, но при этом предоставляет множество мощных функций администрирования.

## 2. Начало работы с Фреймворком "ActiveMQ"

### 2.1. Системные требования

Требования к программному обеспечению для работы с Фреймворком ActiveMQ:

- Среда разработки приложений на языке Java. Например, IntelliJ IDEA, NetBeans, Eclipse.
- Java SE версии 1.6 или выше.
- Apache ActiveMQ.
- СУБД для работы с базами данных.
- Драйверы JDBC для выбранной СУБД.
- Редактор HTML-кода.
- Сервер приложений Axiom Libercat для запуска веб-приложений.

Поддерживаются следующие инструменты сборки:

- Maven
- Gradle

### 2.2. Настройка среды разработки

Установка производится в системах Unix/Linux/MacOSX/macOS.

### 2.3. Установка JDK

Требуется JDK версии 1.6 или выше. Скачать можно по ссылке <https://axiomjdk.ru/downloads/axiom-jdk> . После установки JDK вы можете проверить правильность её настройки, открыв Терминал и введя следующую команду.

```
$ java -version
```

Вы увидите следующий результат.

```
openjdk version "1.8.0_492"  
OpenJDK Runtime Environment (build 1.8.0_492-b09)  
OpenJDK 64-Bit Server VM (build 25.492-b09, mixed mode)
```

Результат будет различаться в зависимости от установленной версии. Важно отметить, что в приведенном выше примере установлена Axiom JDK Pro, и ее версия – 1.8.

## 2.4. Установка ActiveMQ

Установить и загрузить ActiveMQ можно с веб-сайта по ссылке <https://download.axiomjdk.ru/activemq/5.19.2/activemq-5.19.2-1-noarch-linux.deb> . Для дальнейшей работы нужно получить стабильную версию.

Для дистрибутивов Unix/Linux/Cygwin загрузите `apache-activemq-5.19.2-bin.tar.gz` или последнюю версию. После загрузки распакуйте файл.

Пример пути

```
/Users/test_user/Documents/apache-activemq-5
```

## 2.5. Установка Apache Ant.

Apache Ant – это библиотека Java и инструмент командной строки, помогающий в разработке программного обеспечения.

Для сборки и тестирования примеров, входящих в состав ActiveMQ, требуется Apache Ant. Скачать данную библиотеку можно по ссылке <https://ant.apache.org/bindownload.cgi> .

Для работы Apache Ant версии 1.9.x требуется Java 5, а для версии 1.10.x – Java 8.

Для дистрибутивов Unix/Linux/Mac загрузите `apache-ant-1.10.1-bin.tar.bz2` или последнюю версию. После загрузки распакуйте файл.

Пример пути

```
/Users/test_user/Documents/apache-ant-1.10.1
```

Наконец, обязательно настройте и добавьте переменную среды `$ANT_HOME` в переменную среды `$PATH`.

Откройте терминал и введите следующую команду.

```
$ cd
```

Теперь, используя команду `open.bash_profile`

```
$ vi .bash_profile
```

Теперь добавьте в файл следующие строки. Нажмите клавишу, чтобы перейти в режим вставки.

```
#Apache Ant
export ANT_HOME=/Users/test_user/Documents/apache-ant-1.10.1
#Export to PATH
export PATH=$ANT_HOME/bin:$PATH
```

Переменная `PATH` может содержать и другие значения. Будьте осторожны при внесении изменений.

После этого нажмите ESC клавишу, введите текст `:wq` и нажмите клавишу `Enter`.

Теперь проверьте, правильно ли установлен Ant, введя в терминале следующую команду.

```
$ ant -version
Apache Ant(TM) version 1.10.1 compiled on February 2 2017
```

Версия Ant может отличаться в зависимости от того, что вы скачали.

### 2.5.1. Инструменты сборки

Перед установкой инструментов сборки необходимо убедиться, что на компьютере установлена JDK. Чтобы убедиться в этом, выполните команду `Java -version` в терминале или командной строке.



#### Примечание:

Для автоматической установки инструментов сборки с помощью пакетных менеджеров или магазина приложений, обратитесь к документации вашей ОС или среды разработки.

#### Maven

Чтобы автоматизировать процесс скачивания необходимых компонентов Maven, выполните следующее:

1. Создайте проект Spring Boot на странице [Spring Initializr](#) и укажите Maven в разделе Project. Укажите другие настройки проекта, если необходимо.

2. Нажмите **Generate**, чтобы загрузить проект в виде ZIP-файла.
3. Распакуйте ZIP-файл созданного проекта Spring Boot в нужное место на локальном диске.
4. В распакованном проекте найдите файл `mvnw.sh` или `mvnw.cmd` в зависимости от используемой ОС. Запустите файл и все необходимые компоненты и зависимости для Maven будут закачаны и установлены автоматически.

Если вам не подходит автоматическая загрузка и установка компонентов, установите Maven вручную, как показано ниже.

Чтобы установить Maven вручную, выполните следующие шаги:

1. Скачайте архив со сборщиком для своей операционной системы с официального сайта [Apache Maven](#).
2. Распакуйте архив. Лучше создать для этого отдельную папку, так как путь к ней понадобится позже.
3. Отредактируйте переменные среды. На разных операционных системах этот процесс различается.

Для **Windows**:

- a. Нажмите правой кнопкой мыши на «Этот компьютер» и выберите пункт меню «Свойства».
- b. Выберите «Дополнительные параметры системы» → «Дополнительно» → «Переменные среды».
- c. В окне «Переменные среды» найдите переменную Path и нажмите на кнопку «Изменить».
- d. В открывшемся окне нажмите кнопку «Создать» и укажите полный путь до папки bin из распакованного архива Maven.
- e. Проверьте настройку переменных среды. Для этого откройте командную строку и выполните команду `mvn -v`. Если Maven установлен правильно, то появится информация о его версии.

Для Linux и macOS:

- a. Откройте в текстовом редакторе файл `~/.bashrc` или `~/.bash_profile` в Linux или `.zshrc` в macOS.  
  
Если файла нет, создайте его и впишите: `export PATH="<path_to_maven>:$PATH"`.  
Вместо `path_to_maven` укажите путь к файлу.
- b. Проверьте настройку переменных среды. Для этого запустите терминал и выполните

команду `mvn -n`. Должно появиться сообщение с версией Maven.

## Gradle

Чтобы автоматизировать процесс скачивания необходимых компонентов Gradle, выполните следующее:

1. Создайте проект Spring Boot на странице [Spring Initializr](#) и укажите Gradle в разделе Project. Укажите другие настройки проекта, если необходимо.
2. Нажмите **Generate**, чтобы загрузить проект в виде ZIP-файла.
3. Распакуйте ZIP-файл созданного проекта Spring Boot в нужное место на локальном диске.
4. В распакованном проекте найдите файл `gradlew.sh` или `gradlew.bat` в зависимости от используемой ОС. Запустите файл и все необходимые компоненты и зависимости для Maven будут закачаны и установлены автоматически.

Если вам не подходит автоматическая загрузка и установка компонентов, установите Gradle вручную, как показано ниже.

Чтобы установить Gradle, выполните следующие шаги:

1. Скачайте последнюю версию Gradle. Дистрибутив можно загрузить на официальном сайте по адресу [gradle.org](https://gradle.org).
2. Распакуйте дистрибутив. Для Linux и macOS нужно распаковать ZIP-файл в выбранную директорию. В Windows нужно создать новую директорию `C:\Gradle` и распаковать папку `gradle-8.13` из ZIP-архива в созданную директорию `C:\Gradle`.
3. Настройте системную среду. Для Linux и macOS нужно настроить переменную среды PATH, чтобы включить директорию `bin` распакованного дистрибутива. В Windows в File Explorer нажмите правой кнопкой мыши на значок «Этот ПК» (или «Компьютер»), затем выберите «Свойства» – «Дополнительные системные настройки» – «Переменные среды». В разделе «Системные переменные» выберите Path, затем нажмите «Редактировать». Добавьте путь к каталогу `bin` в папке Gradle и сохраните изменения.
4. Проверьте правильность установки Gradle. Для этого откройте консоль или командную строку Windows и выполните команду `gradle -v`, чтобы запустить Gradle и вывести версию.

Если установка прошла успешно, в выводе будет указана версия Gradle и другая информация.

Чтобы использовать Gradle в IntelliJ IDEA, нужно выполнить следующие шаги:

1. Запустите IntelliJ IDEA и создайте новый проект Java. Затем выберите папку, в которой будет храниться проект, и укажите его имя.

2. После создания проекта IntelliJ IDEA предложит настроить Gradle. Выберите опцию «Use auto-import» и укажите путь к установленному Gradle.
3. В корне проекта создайте файл с именем «build.gradle». В нём будут определяться настройки сборки.
4. Отредактируйте файл сборки.
5. После внесения изменений в файл сборки выполните синхронизацию Gradle, чтобы IntelliJ IDEA обновила зависимости и настройки проекта. Для этого щёлкните правой кнопкой мыши на «build.gradle» и выберите опцию «Sync Gradle».

После синхронизации Gradle можно собрать Java-проект, выбрав опцию «Build» в меню IntelliJ IDEA

Чтобы использовать Gradle в Eclipse, нужно установить плагин Gradle Integration Plugin. Используйте следующие шаги:

1. Запустите Eclipse.
2. Нажмите «Help» > «Eclipse Marketplace».
3. Найдите Gradle Integration Plugin.
4. Установите плагин.

После установки плагина нужно настроить расположение Gradle в Eclipse:

1. Запустите Eclipse и перейдите в меню «Windows».
2. Нажмите «Preferences».
3. В окне «Preferences» найдите «Gradle» и выберите его.
4. В разделе «Gradle» найдите опцию «Локальный каталог установки».
5. Введите путь к папке Gradle в системе.

На Windows это обычно место, где распаковывается папка Gradle.

На Mac найти расположение Gradle можно с помощью команды «Brew info gradle» в терминале.

6. Нажмите «Применить» и «Закреть», чтобы сохранить изменения.

## 2.6. Редактирование и запуск приложения ActiveMQ

## 2.7. Запуск Axiom ActiveMQ

Перейдите в каталог ActiveMQ и введите следующую команду.

```
$ cd /Users/test_user/Documents/apache-activemq-5.19.2
$ ./bin/activemq console
```

## 2.8. Проверка состояния ActiveMQ

Для проверки состояния ActiveMQ используется следующая команда.

```
$ ./bin/activemq status
INFO: Loading '/Users/test_user/Documents/apache-activemq-5.19.2//bin/env'
INFO: Using java '/usr/bin/java'
ActiveMQ is running (pid '17572')
```

## 2.9. Остановка ActiveMQ

Для остановки используем + стопили Ctrl+ C.

## 2.10. Проверка веб-консоли

После запуска ActiveMQ мы сможем получить к нему доступ через веб-интерфейс.

На локальном компьютере просто введите следующий URL-адрес в веб-браузере.

```
http://localhost:8161/admin
```

На сервере введите имя вашего домена, а затем точку с запятой :8161/admin.

```
http://www.example.com:8161/admin
```

При запросе имени пользователя и пароля введите admin в оба поля. В случае успешного ввода откроется веб-страница.

## 2.11. Защищенная веб-консоль

По умолчанию для веб-консоли ActiveMQ используются имя пользователя и пароль admin/admin. Чтобы изменить их, выполните следующие действия:



- Установка `authenticate = true`
- Установка имя пользователя/пароль

