

**AXIOM JDK**

# **Axiom ActiveMQ**

## **Руководство пользователя**

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 .....	5
1.1. Что такое Axiom ActiveMQ? .....	5
1.2. Основные возможности ActiveMQ .....	5
2. Начало работы с "ActiveMQ" .....	7
2.1. Системные требования .....	7
2.2. Настройка среды разработки .....	7
2.3. Установка JDK .....	7
2.4. Установка ActiveMQ .....	8
2.5. Установка Apache Ant .....	8
2.5.1. Установка JDK .....	9
2.5.2. Инструменты сборки .....	9
2.6. Редактирование и запуск приложения ActiveMQ .....	12
2.7. Запуск ActiveMQ .....	12
3. Основные концепции ActiveMQ .....	15
3.1. Apache ActiveMQ – работающий брокер-сервер .....	15
3.2. Apache ActiveMQ – консоль администратора .....	15

3.3. Apache ActiveMQ – приложение-производитель .....	15
3.3.1. Создать проект .....	15
3.4. Apache ActiveMQ – приложение-потребитель .....	16
3.4.1. Создать проект .....	16
3.5. Apache ActiveMQ – приложение-потребитель .....	16
3.5.1. Создать проект .....	16
3.6. Apache ActiveMQ – приложение-подписчик .....	17
3.6.1. Создать проект .....	17

# 1. Введение

Настоящий документ содержит начальные сведения по эксплуатации программного обеспечения 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 ActiveMQ

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 "ActiveMQ"

### 2.1. Системные требования

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

Требования к программному обеспечению для работы с ActiveMQ:

- Требуется Java SE версии 1.6 или выше.
- Установить и загрузить ActiveMQ можно с веб-сайта Apache. Для дальнейшей работы нужно получить стабильную версию.
- Для сборки и тестирования примеров, входящих в состав ActiveMQ, требуется Apache Ant.

Поддерживаются следующие инструменты сборки:

- Maven
- Gradle

### 2.2. Настройка среды разработки

### 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 ESCклавишу, введите текст :wqи нажмите клавишу Enter.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. Установка JDK

Мы рекомендуем использовать Axiom JDK Pro вместе с проектами использующими Axiom ActiveMQ.

Вы можете установить Axiom JDK Pro на компьютеры под управлением Microsoft Windows, Linux и macOS. Тип установки, который вы выберете, зависит от ваших требований и платформы, которую вы используете.

Вы можете загрузить Axiom JDK Pro либо из [Центра загрузок Axiom JDK Pro](#), либо по ссылке на [портале поддержки](#). Доступ к этому portalу предоставляется клиентам с активным договором поддержки.

Подробную информацию по установке Axiom JDK Pro вы можете получить из Руководства по установке соответствующей версии на [странице центра загрузок](#).

### 2.5.2. 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 ActiveMQ на странице [ActiveMQ 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 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 ActiveMQ 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.

## 2.6. Редактирование и запуск приложения ActiveMQ

Важные файлы/каталоги:

- **activemq-all-5.19.2.jar** - Этот JAR-файл содержит все компоненты ActiveMQ.
- **контейнер** - В этой директории находятся все исполняемые файлы.
- **conf** - В этом каталоге находятся файлы конфигурации ActiveMQ.
- **данные** - Эта директория предназначена для хранения файлов журналов и данных сообщений.
- **пример** - В этой директории находятся несколько примеров использования ActiveMQ.
- **библиотека** - В этой директории находятся все библиотеки, необходимые для ActiveMQ.

## 2.7. Запуск ActiveMQ

Перейдите в каталог ActiveMQ и введите следующую команду.

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

Выход

```
INFO: Loading '/Users/test_user/Documents/apache-activemq-5.19.2//bin/env'
INFO: Using java '/usr/bin/java'
INFO: Starting in foreground, this is just for debugging purposes (stop
process by pressing CTRL+C)
INFO: Creating pidfile /Users/test_user/Documents/apache-activemq-
5.19.2//data/activemq.pid
Java Runtime: Axiom JSC 1.8.0_492
/Library/Java/JavaVirtualMachines/jdk1.8.0_102.jdk/Contents/Home/jre
Heap sizes: current=62976k free=60026k max=932352k
JVM args: -Xms64M -Xmx1G -Djava.util.logging.config.file=logging.properties
-Djava.security.auth.login.config=/Users/test_user/Documents/apache-activemq
-5.19.2//conf/login.config -Dcom.sun.management.jmxremote
```

```
-Djava.awt.headless=true -Djava.io.tmpdir=/Users/test_user/Documents/apache-activemq-5.19.2//tmp -Dactivemq.classpath=/Users/test_user/Documents/apache-activemq-5.19.2//conf:/Users/test_user/Documents/apache-activemq-5.19.2//../lib/: -Dactivemq.home=/Users/test_user/Documents/apache-activemq-5.19.2/ -Dactivemq.base=/Users/test_user/Documents/apache-activemq-5.19.2/ -Dactivemq.conf=/Users/test_user/Documents/apache-activemq-5.19.2//conf -Dactivemq.data=/Users/test_user/Documents/apache-activemq-5.19.2//data
Extensions classpath:
[/Users/test_user/Documents/apache-activemq-5.19.2/lib,/Users/test_user/Documents/apache-activemq-5.19.2/lib/camel,/Users/test_user/Documents/apache-activemq-5.19.2/lib/optional,/Users/test_user/Documents/apache-activemq-5.19.2/lib/web,/Users/test_user/Documents/apache-activemq-5.19.2/lib/extra]
ACTIVEMQ_HOME: /Users/test_user/Documents/apache-activemq-5.19.2
ACTIVEMQ_BASE: /Users/test_user/Documents/apache-activemq-5.19.2
ACTIVEMQ_CONF: /Users/test_user/Documents/apache-activemq-5.19.2/conf
ACTIVEMQ_DATA: /Users/test_user/Documents/apache-activemq-5.19.2/data
Loading message broker from: xbean:activemq.xml
INFO | Refreshing org.apache.activemq.xbean.XBeanBrokerFactory$1@28feb3fa: startup date [Tue Feb 14 19:46:22 IST 2017]; root of context hierarchy
INFO | Using Persistence Adapter:
KahaDBPersistenceAdapter[/Users/test_user/Documents/apache-activemq-5.19.2/data/kahadb]
INFO | PListStore:[/Users/test_user/Documents/apache-activemq-5.19.2/data/localhost/tmp_storage] started
INFO | Apache ActiveMQ 5.19.2 (localhost, ID:test_user-MacBook-Pro.local-58262-1487081799213-0:1) is starting
INFO | Listening for connections at: tcp://test_user-MacBook-Pro.local:61616?maximumConnections=1000&wireFormat.maxFrameSize=104857600
INFO | Connector openwire started
INFO | Listening for connections at: amqp://test_user-MacBook-Pro.local:5672?maximumConnections=1000&wireFormat.maxFrameSize=104857600
INFO | Connector amqp started
INFO | Listening for connections at: stomp://test_user-MacBook-Pro.local:61613?maximumConnections=1000&wireFormat.maxFrameSize=104857600
INFO | Connector stomp started
INFO | Listening for connections at: mqtt://test_user-MacBook-Pro.local:1883?maximumConnections=1000&wireFormat.maxFrameSize=104857600
INFO | Connector mqtt started
WARN | ServletContext@o.e.j.s.ServletContextHandler@3e6f3f28{/,null,STARTING} has uncovered http methods for path: /
INFO | Listening for connections at ws://test_user-MacBook-Pro.local:61614?maximumConnections=1000&wireFormat.maxFrameSize=104857600
INFO | Connector ws started
```



```
INFO | Apache ActiveMQ 5.19.2 (localhost, ID:test_user-MacBook-Pro.local-58262-1487081799213-0:1) started
INFO | For help or more information please see: http://activemq.apache.org
INFO | No Spring WebApplicationInitializer types detected on classpath
INFO | ActiveMQ WebConsole available at http://0.0.0.0:8161/
INFO | ActiveMQ Jolokia REST API available at
http://0.0.0.0:8161/api/jolokia/
INFO | Initializing Spring FrameworkServlet 'dispatcher'
INFO | No Spring WebApplicationInitializer types detected on classpath
INFO | jolokia-agent: Using policy access restric
```

## 3. Основные концепции ActiveMQ

### 3.1. Apache ActiveMQ – работающий брокер-сервер

Перейдите в папку `/Users/test_user/Documents/apache-activemq-5.19.2/bin` и введите следующую команду.

```
activemq start
```

Теперь откройте в браузере `http://127.0.0.1:8161/admin/`. Вам будет предложено ввести учетные данные. Используйте `admin/admin` в качестве имени пользователя/пароля, и загрузится консоль администратора ActiveMQ, где вы сможете проверить очереди, темы, соединения и т. д.

### 3.2. Apache ActiveMQ – консоль администратора

После запуска сервера ActiveMQ вы можете использовать консоль администратора для управления очередями, темами, подписчиками, соединениями, сетью и т. д.

Откройте в браузере `http://127.0.0.1:8161/admin/`. Вам будет предложено ввести учетные данные. Используйте `admin/admin` в качестве имени пользователя/пароля, и загрузится консоль администратора ActiveMQ, где вы сможете проверить очереди, темы, соединения и т. д.

### 3.3. Apache ActiveMQ – приложение-производитель

Создадим приложение-отправитель, которое будет отправлять сообщения в очередь ActiveMQ.

#### 3.3.1. Создать проект

В Eclipse выберите `File & New → Maven Project`. Установите флажок `Create a simple project (skip archetype selection)` и нажмите `Next`.

Введите данные, как показано ниже:

- `groupId – com.tutorialspoint`
- `artifactId – producer`
- `версия – 0.0.1-SNAPSHOT`

- имя – ActiveMQ Producer

Нажмите кнопку «Готово», и будет создан новый проект.

## 3.4. Apache ActiveMQ – приложение-потребитель

Создадим приложение-потребитель, которое будет получать сообщения из очереди ActiveMQ.

### 3.4.1. Создать проект

В Eclipse выберите File & New → Maven Project . Установите флажок Create a simple project (skip archetype selection) и нажмите Next.

Введите данные, как показано ниже:

- groupId – com.tutorialspoint
- artifactId – consumer
- версия – 0.0.1-SNAPSHOT
- имя – потребитель ActiveMQ

Нажмите кнопку «Готово», и будет создан новый проект.

## 3.5. Apache ActiveMQ – приложение-потребитель

Создадим приложение-потребитель, которое будет получать сообщения из очереди ActiveMQ.

### 3.5.1. Создать проект

В Eclipse выберите File & New → Maven Project . Установите флажок Create a simple project (skip archetype selection) и нажмите Next.

Введите данные, как показано ниже:

- groupId – com.tutorialspoint
- artifactId – consumer
- версия – 0.0.1-SNAPSHOT
- имя – потребитель ActiveMQ

Нажмите кнопку «Готово», и будет создан новый проект.

## 3.6. Apache ActiveMQ – приложение-подписчик

Создадим приложение-подписчик, которое будет получать сообщения из темы ActiveMQ.

### 3.6.1. Создать проект

В Eclipse выберите File & New → Maven Project . Установите флажок Create a simple project (skip archetype selection) и нажмите Next.

Введите данные, как показано ниже:

- groupId – com.tutorialspoint
- artifactId – subscriber
- версия – 0.0.1-SNAPSHOT
- имя – Подписчик ActiveMQ

Нажмите кнопку «Готово», и будет создан новый проект.

