

# Платформа ВІV

Документация, содержащая информацию,  
необходимую для установки и эксплуатации  
программного обеспечения

15.06.2023



# Введение

Платформа BIV — это набор инфраструктурных библиотек и сервисов для создания и развития информационных систем для финансовых организаций в Микросервисной Архитектуре.

Платформа BIV разработана на языке Java с использованием фреймворка Quarkus. Данный фреймворк собран из лучших библиотек и стандартов Java и предназначен для создания современных императивных и реактивных Java-приложений в первую очередь для облачной платформы Kubernetes.

## Состав поставки платформы BIV

В состав поставки Платформы BIV входят:

- maven-артефакты платформы.  
Установку артефактов см. «Приложение 1. Скрипт по установке артефактов в репозиторий maven»:
  - Набор библиотек для разработки API:  
crypto-service-1.2.jar, crypto-service-1.2.pom  
localization-1.2.jar, localization-1.2.pom  
service-exception-1.0.jar, service-exception-1.0.pom  
rest-common-2.0.jar, rest-common-2.0.pom
  - Набор библиотек и расширений для расширенного описания модели данных:  
entity-info-1.6.jar, entity-info-1.6.pom  
entity-info-deployment-1.6.jar, entity-info-deployment-1.6.pom  
entity-info-parent-1.6.pom  
service-discovery-api-1.2.jar, service-discovery-api-1.2.pom  
distributed-entity-info-1.8.jar, distributed-entity-info-1.8.pom
  - Библиотека, реализующая машину состояний, основанная на расширенной модели данных:  
state-machine-1.0.jar, state-machine-1.0.pom
  - Расширение для универсального создания объектов по шаблону:  
entity-object-template-1.0.jar, entity-object-template-1.0.pom  
entity-object-template-deployment.jar, entity-object-template-deployment.pom  
entity-object-template-parent-1.0.pom
  - Библиотека для описания и хранения данных НСИ:  
refbook-2.9.jar, refbook-2.9.pom
  - Библиотека для настраиваемого отображения табличных данных:  
entity-data-api-core-1.4.jar, entity-data-api-core-1.4.pom  
entity-data-api-1.4.jar, entity-data-api-1.4.pom  
entity-data-api-all-1.4.pom, entity-data-api-all-1.4.pom  
data-export-core-1.2.jar, data-export-core-1.2.pom  
data-export-1.2.jar, data-export-1.2.pom  
data-export-all-1.2.pom, data-export-all-1.2.

entityjournal-core-2.9.jar, entityjournal-core-2.9.pom  
entityjournal-services-2.9.jar, entityjournal-services-2.9.pom

- Общесистемные сервисы платформы:
  - Сервис формирования печатных форм по шаблону.
- Документация по использованию библиотек и сервисов:
  - Руководство разработчика по использованию фреймворка Quarkus;
  - Руководство разработчика API;
  - Руководство разработчика BIV Platform;
  - Руководство разработчика entity-info;
  - Руководство разработчика service-discovery-api;
  - Руководство разработчика distributed-entity-info;
  - Руководство разработчика entity-object-template;
  - Руководство разработчика state-machine;
  - Руководство разработчика refbook;
  - Руководство разработчика libreferma;
  - Руководство разработчика по использованию концепции микрофронтенд;
  - Руководство разработчика по использованию фреймворка Angular;
  - Руководство разработчика по локализации веб-приложений;
  - Руководство пользователя МСП;

## **Способ поставки платформы BIV**

Платформа может быть поставлена как архив файлов. В архив включены собранные (скомпилированные) компоненты.

«Сервис формирования печатных форм по шаблону» устанавливается специалистом компании BIV.

# Приложение 1. Скрипт по установке артефактов в репозиторий maven

Установка компонентов платформы в репозиторий артефактов maven (например, Nexus Sonatype), развёрнутый у Заказчика, осуществляется при помощи команды «mvn deploy».

```
export DEPLOY_URL=A_ДРЕС_ХРАНИЛИЩА_АРТЕФАКТОВ_MAVEN
export REPO_ID=P_РЕПОЗИТОРИЙ

mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=crypto-service-1.2.jar -DpomFile=crypto-service-1.2.pom
mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=localization-1.2.jar -DpomFile=localization-1.2.pom
mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=service-exception-1.0.jar -DpomFile=service-exception-1.0.pom
mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=rest-common-2.0.jar -DpomFile=rest-common-2.0.pom

mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=entity-info-1.6.jar -DpomFile=entity-info-1.6.pom
mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=entity-info-deployment-1.6.jar -DpomFile=entity-info-deployment-1.6.pom
mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=entity-info-parent.pom -DpomFile=entity-info-parent.pom -Dpackaging=pom
mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=service-discovery-api-1.2.jar -DpomFile=service-discovery-api-1.2.pom
mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=distributed-entity-info-1.8.jar -DpomFile=distributed-entity-info-1.8.pom

mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=state-machine-1.0.jar -DpomFile=state-machine-1.0.pom

mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=entity-object-template-1.0.jar -DpomFile=entity-object-template-1.0.pom
mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=entity-object-template-deployment.jar -DpomFile=entity-object-template-deployment.pom
mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=entity-object-template-parent-1.0.pom -DpomFile=entity-object-template-parent-1.0.pom -Dpackaging=pom

mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=refbook-2.9.jar -DpomFile=refbook-2.9.pom

mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=entity-data-api-core-1.4.jar -DpomFile=entity-data-api-core-1.4.pom
mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=entity-data-api-1.4.jar -DpomFile=entity-data-api-1.4.pom
mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=entity-data-api-all-1.4.jar -DpomFile=entity-data-api-all-1.4.pom
mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=data-export-core-1.2.jar -DpomFile=data-export-core-1.2.pom
mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=data-export-1.2.jar -DpomFile=data-export-1.2.pom
mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=data-export-all-1.2.jar -DpomFile=data-export-all-1.2.pom
mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=entityjournal-core-2.9.jar -DpomFile=entityjournal-core-2.9.pom
mvn deploy:deploy-file -Durl=${DEPLOY_URL} -DrepositoryId=${REPO_ID} -Dfile=entityjournal-servicies-2.9.jar -DpomFile=entityjournal-servicies-2.9.pom
```

Установка компонентов платформы в локальный репозиторий артефактов maven на компьютере разработчика осуществляется при помощи команды «mvn install».