БИПЛЕКС-BPLEX

Руководство администратора

Содержание

[1. Общие положения 3](#_Toc68698253)

[1.1. Наименование системы 3](#_Toc68698254)

[1.2. Область применения 3](#_Toc68698255)

[1.3. Уровень подготовки администратора 3](#_Toc68698256)

[2. Назначение 4](#_Toc68698257)

[2.1. Назначение системы 4](#_Toc68698258)

[2.2. Краткое описание возможностей 4](#_Toc68698259)

[3. Структура системы 5](#_Toc68698260)

[4. Условия применения 6](#_Toc68698261)

[4.1. Требования к оборудованию и программному обеспечению 6](#_Toc68698262)

[4.2. Параметры конфигураций 8](#_Toc68698263)

[5. Разграничение прав 8](#_Toc68698264)

[5.1. Роли 8](#_Toc68698265)

[5.2. Настройки безопасности 8](#_Toc68698266)

[6. Установка Системы 8](#_Toc68698267)

[6.1. Процесс запуска программного обеспечения 8](#_Toc68698268)

[6.2. Проверка работоспособности Системы 8](#_Toc68698269)

[7. Настройка интеграционного взаимодействия с внешними системами 8](#_Toc68698270)

[8. Эксплуатация Системы 8](#_Toc68698271)

[8.1. Управление пользователями Системы 8](#_Toc68698272)

[8.2. Мониторинг Системы 8](#_Toc68698273)

[8.3. Работа с логами 8](#_Toc68698274)

[9. Установка обновлений 8](#_Toc68698275)

[10. Порядок восстановления Системы 8](#_Toc68698276)

[Список обозначений и сокращений 9](#_Toc68698277)

# Общие положения

## Наименование системы

Полное наименование системы: «БИПЛЕКС-BPLEX».

Сокращенное называние: Система, BPLEX.

## Область применения

BPLEX используется в системах, ключевые возможности которых лежат в области поиска оптимальных вариантов и принятия к реализации максимально эффективных сценариев.

## Уровень подготовки администратора

Пользователи группы Администратор должны обладать навыками конфигурирования СУБД PostgreSQL, серверных операционных систем семейства Linux, общесистемного ПО, настройки сетевых аппаратных и программных средств, а также ознакомиться с настоящим руководствами администратора.

# Назначение

## Назначение системы

«БИПЛЕКС-BPLEX» применяется для решения задач линейной оптимизации, распределения и запуска задач, контроля очереди расчетов, создания моделей в стиле максимально приближенном к «математическому» описанию задачи.

## Краткое описание возможностей

«БИПЛЕКС-BPLEX» является системой управляемого исполнения математических моделей, позволяющей выполнять настраиваемые последовательности операций, в том числе математических моделей, в заданной последовательности в распределенной среде вычислений.

Математические модели могут быть идемпотентными (с идентичными по типу наборами данных на входе и выходе) и трансформационными (с разными по типу наборами данных). Помимо основных данных, модели могут принимать на вход наборы ограничений, параметров и формировать на выходе опциональные наборы данных.

Кроме того, Система оперирует вспомогательными механизмами: модулями загрузки, выгрузки, преобразования данных, условий, условных и безусловных циклов.

При создании моделей возможно связывать информационные потоки между ними (с учетом типов данных), а также настраивать параметры работы конкретных модулей.

Создание и настройка последовательностей осуществляется с помощью наглядного web-интерфейса в пользовательском интерфейсе системы.

Система открыта к добавлению в нее новых моделей, разработанных в соответствии с SDK.

BPLEX реализует ролевую модель доступа к ее использованию и управлению на принципах Explicit Deny.

# Структура системы

Система реализована в трехзвенной архитектуре технических средств, в соответствии с концепцией построения клиент-серверных информационных систем.

BPLEX состоит из пользовательского интерфейса, API-интерфейса, среды исполнения, системы версионирования и журналирования, системы аутентификации и авторизации, системы верификации, SDK. Среда исполнения, в свою очередь, состоит из системы управления, системы балансировки и исполняющих модулей.

Исполняющими модулями могут быть, в частности линейные оптимизаторы (CPLEX, Gurobi, Sparx и т.п.).

****

**Рисунок 1.** Логическая схема расположения компонент Системы в потенциальной среде развертывания.

|  |  |
| --- | --- |
| UI | Публичный и административный интерфейсы системы |
| API Gateway | Интерфейс взаимодействия с BPLEX для UI и внешних систем |
| Validator | Служба проверки консистентности модели |
| Storage | Подсистема хранения моделей, служебной информации, промежуточных результатов вычислений, состояний |
| Scheduler | Служба настройки расписания запуска |
| Controller | Механизм управления ресурсами для параллельного исполнения моделей и их элементов, в том числе за счет динамического создания новых Executor-ов и управления выделяемыми им ресурсами |
| Interpreter | Служба интерпретации модели для конкретных Executor-ов |
| Executor | Прикладной модуль линейной оптимизации |
| Reader | Служба загрузки данных из внешних источников |
| Writer | Служба передачи данных во внешние источники |
| ESB | Шина данных для межсервисного взаимодействия |
| Permissions | Служба управления правами доступа |
| Logging | Подсистема журналирования |
| Monitoring | Подсистема мониторинга |
| Version control | Подсистема версионирования моделей |

# Условия применения

## Требования к оборудованию и программному обеспечению

BPLEX должен быть развернут на базе виртуальных машин с операционными системами Linux.

Ниже приведены рекомендуемые параметры конфигурации аппаратной части компонентов системы. Для всех перечисленных аппаратных компонентов и их устройств должны быть установлены драйверы производителя (или сертифицированные совместимые).

Таблица 4. Требования к комплексу технических средств

| Узел | | Технические характеристики | Общесистемное программное обеспечение – операционная система (ОС) | Специальное программное обеспечение |
| --- | --- | --- | --- | --- |
| ПК оператора | | Процессор: Intel Pentium/Celeron 1800 МГц и выше;  ОЗУ: не менее 1 Гб;  HDD: 1 Гб и более | ОС:  Windows XP (SP3) и выше, Linux, MAC OS, Android v6 и выше | Веб-браузер:  Google Chrome |
| Управляющий сервер виртуальной среды (оркестрация контейнеров) | Kubernetes Master | Процессор: 2 х Intel/AMD x64;  ОЗУ: не менее 4 Gb;  HDD: не менее 32 Gb | ОС:  Linux семейства CentOS 7.4 | Kubernetes |
| Kubernetes Slave | Процессор: 16 х Intel/AMD x64;  ОЗУ: не менее 64 Gb;  HDD: не менее 64 Gb | ОС:  Linux семейства CentOS 7.4 |
| Сервер приложений виртуальной среды | | Процессор: 4х Intel/AMD x64;  ОЗУ: не менее 16 Gb;  HDD: не менее 64 Gb | ОС:  Linux семейства CentOS 7.4 | Kubernetes |
| Сервер БД | | Процессор: 16 х Intel/AMD x64;  ОЗУ: не менее 32 Gb;  HDD: не менее 200 Gb | ОС:  Linux семейства CentOS 7.4 | СУБД:  PostgreSQL, MS SQL, Mongo |
| Сервер обмена сообщениями  (основная шина) | | Процессор: 8 х Intel/AMD x64;  ОЗУ: не менее 8 Gb;  HDD: не менее 40 Gb | ОС:  Linux семейства CentOS 7.4 | Kafka |
| Сервер журналирования (логов) | | Процессор: 4 х Intel/AMD x64;  ОЗУ: не менее 8 Gb;  HDD: не менее 200 Gb | ОС:  Linux семейства CentOS 7.4 | Elastic Search (сбор), Kibana (отображение) |
| Сервер мониторинга (логов) | | Процессор: 4 х Intel/AMD x64;  ОЗУ: не менее 8 Gb;  HDD: не менее 200 Gb | ОС:  Linux семейства CentOS 7.4 | Prometeus (сбор), Graphana (отображение) |

Дополнительной специализированное ПО:

* Контроль версий – Gitlab/Gitlab Registry;
* Права доступа – Basic/Kerberos/LDAP;
* Контейнеры – Docker;
* Среда проектирования и программные языки – Quarkus, Java, Python.

# Разграничение прав

## Роли

Управление доступом к объектам системы осуществляется на основе ролевой модели. В Системе предусмотрена роль «Пользователь», дающая возможность создания и управления сущностями Системы.

## Настройки безопасности

Настройки безопасности Системы задаются на уровне Kubernetes.

# Установка Системы

Для развертывания BPLEX необходимо выполнить следующие действия в установленном порядке.

## Процесс запуска программного обеспечения

Развёртывание бэкенда

Чтобы подготовить к работе и запустить бэкенд системы BPLEX (проект https://gitlab.dellin.ru/OMOiM\_developers/bplex-system/), используя сервер или рабочую станцию с ОС Linux, необходимо:

Установить и настроить JDK 11

Установить и настроить Docker

Установить и настроить клиентское ПО Kubernetes

Скачать исходники проекта (командой git clone https://gitlab.dellin.ru/OMOiM\_developers/bplex-system.git или иным способом).

В файле *resources/application.properties* указать актуальные параметры доступа к БД и другие привязанные к развёртыванию.

Собрать файл jar командой *./gradlew quarkusBuild compileTestKotlin*

Собрать и опубликовать Docker-образ:

*docker build -t* *Название\_образа* *-f src/main/docker/Dockerfile* .

*docker push Название\_образа*

Подготовить и развернуть конфигурацию на кластере Kubernetes.

Типовой шаблон конфигурации находится в файле *src/main/kubernetes/deployment.template.yaml* проекта. В него необходимо подставить требуемые значения параметров для новой конфигурации и сохранить файл с названием *deployment.yaml*

Развернуть конфигурацию командой *kubectl replace --force --cascade -f deployment.yaml*

Развёрнутый бэкенд имеет UI, сгенерированный с использованием Swagger, расположенный по URL *http://web-test.bplex.sandbox.bia-tech.ru/q/swagger-ui/* (здесь и далее домен в указанном URL характерен для демо-стенда). Рабочее API соответствует содержимому этой страницы.

Развёртывание фронтенда

Чтобы подготовить к работе и запустить фронтенд системы BPLEX (проект *https://gitlab.dellin.ru/OMOiM\_developers/bplex-ui/*), используя сервер или рабочую станцию с ОС Linux, необходимо:

Установить и настроить Docker.

Установить и настроить клиентское ПО Kubernetes.

Скачать исходники проекта (командой *git clone https://gitlab.dellin.ru/OMOiM\_developers/bplex-ui.git* или иным способом)

В файле *src/api.ts* указать актуальный адрес бэкенда (в соответствии с Ingress или Service).

Собрать и опубликовать Docker-образ:

*docker build -t Название\_образа* .

*docker push Название\_образа*

Подготовить и развернуть конфигурацию на кластере Kubernetes.

Пример конфигурации находится в файле *kubernetes/deployment.json* проекта. При необходимости можно изменить некоторые параметры.

Развернуть конфигурацию командой *kubectl rollout restart -f ./kubernetes/deployment.json*

## Проверка работоспособности Системы

Общее состояния системы отображается на /q/health

# Настройка интеграционного взаимодействия с внешними системами

Не требуется.

# Эксплуатация Системы

## Управление пользователями Системы

Управление пользователями системы BPLEX осуществляется встроенными механизмами *ingress nginx kubernetes*,

## Мониторинг Системы

Вывод метрик Системы осуществляется на */q/metrics*. Общее состояния системы отображается на */q/health*

## Работа с логами

Работа с логами выполняется встроенными механизмами Kubernetes.

# Установка обновлений

Установка обновлений осуществляется через сборку новых Docker образов п. 6.1 и rolling update средствами Kubernetes.

Общее состояния системы отображается на /q/health

# Порядок восстановления Системы

Восстановление Системы осуществляется встроенными средствами Kubernetes. При ошибке восстановления выполняется установка в соответствии с п. 6.1.

# Список обозначений и сокращений

|  |  |
| --- | --- |
| Сокращение | Значение |
| API | И[нтерфейс](http://o-ili-v.ru/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81) программирования приложений, интерфейс прикладного программирования ([англ.](http://o-ili-v.ru/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) Application programming interface) – набор готовых [классов](http://o-ili-v.ru/wiki/%D0%9A%D0%BB%D0%B0%D1%81%D1%81_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)), [процедур](http://o-ili-v.ru/wiki/%D0%9F%D1%80%D0%BE%D1%86%D0%B5%D0%B4%D1%83%D1%80%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)), [функций](http://o-ili-v.ru/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D1%8F_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)), [структур](http://o-ili-v.ru/wiki/%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) и [констант](http://o-ili-v.ru/wiki/%D0%9A%D0%BE%D0%BD%D1%81%D1%82%D0%B0%D0%BD%D1%82%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)), предоставляемых приложением (библиотекой, сервисом) или операционной системой для использования во внешних программных продуктах. Используется программистами при написании всевозможных приложений |
| Basic | Beginner's All-purpose Symbolic Instruction Code (Всецелевой язык программирования для начинающих) |
| CentOS | ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) Community ENTerprise Operating System) – [дистрибутив](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D1%81%D1%82%D1%80%D0%B8%D0%B1%D1%83%D1%82%D0%B8%D0%B2_%D0%BE%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%BE%D0%B9_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D1%8B) [Linux](https://ru.wikipedia.org/wiki/Linux), основанный на коммерческом [Red Hat Enterprise Linux](https://ru.wikipedia.org/wiki/Red_Hat_Enterprise_Linux) компании [Red Hat](https://ru.wikipedia.org/wiki/Red_Hat) и совместимый с ним |
| CPLEX | Пакет программного обеспечения («решатель»), предназначенный для решения задач линейного и квадратичного программирования, в том числе целочисленного программирования |
| Docker | Программное обеспечение для автоматизации развёртывания и управления приложениями в среде [виртуализации на уровне операционной системы](https://ru.wikipedia.org/wiki/%D0%92%D0%B8%D1%80%D1%82%D1%83%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F_%D0%BD%D0%B0_%D1%83%D1%80%D0%BE%D0%B2%D0%BD%D0%B5_%D0%BE%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%BE%D0%B9_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D1%8B) |
| Elastic Search | Высокомасштабируемая распределенная поисковая система полнотекстового поиска и анализа данных, работающая в режиме реального времени |
| Explicit Deny | Приоритет отказа над разрешением |
| Gitlab | Веб-инструмент жизненного цикла DevOps с открытым исходным кодом, представляющий систему управления репозиториями кода для Git с собственной вики, системой отслеживания ошибок, CI/CD пайплайном и другими функциями |
| Gitlab Registry | GitLab Container Registry – это безопасный приватный реестр для образов (images) Docker, разработанный с помощью ПО с открытым кодом |
| Gradle | Система сборки |
| Graphana | Инструмент с открытым исходным кодом для визуализации данных из различных систем сбора статистики |
| Gurobi | Внешний модуль решения задач с параллельными алгоритмами для задач линейного программирования, задач квадратичного программирования и смешанно-целочисленных задач |
| Java | [Строго](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D0%B8_%D1%81%D0%BB%D0%B0%D0%B1%D0%B0%D1%8F_%D1%82%D0%B8%D0%BF%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F) [типизированный](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D0%B0%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D1%82%D0%B8%D0%BF%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F) [объектно-ориентированный язык программирования](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) общего назначения |
| JDK | JDK (Java SE Development Kit) – набор инструментов разработчика для создания программ на Java. Включает в себя JRE плюс инструменты для разработки, отладки и мониторинга Java приложений |
| JSON | (JavaScript Object Notation) – [текстовый формат](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BA%D1%81%D1%82%D0%BE%D0%B2%D1%8B%D0%B9_%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82) [обмена данными](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D0%BC%D0%B5%D0%BD_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D0%BC%D0%B8), основанный на [JavaScript](https://ru.wikipedia.org/wiki/JavaScript) | |
| Kafka | Диспетчер сообщений на Java платформе |
| Kerberos | Kerberos – сетевой протокол аутентификации, позволяющий передавать данные через незащищённые сети для безопасной идентификации. Ориентирован , в первую очередь , на клиент-серверную модель и обеспечивает взаимную аутентификацию – оба пользователя через сервер подтверждают личности друг друга |
| Kibana | Платформа для анализа и визуализации с открытым исходным кодом, предназначенная для работы с Elasticsearch |
| Kubernetes | Программное обеспечение с открытым [исходным кодом](https://ru.wikipedia.org/wiki/%D0%98%D1%81%D1%85%D0%BE%D0%B4%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BE%D0%B4) |
| LDAP | (англ. Lightweight Directory Access Protocol – «легковесный протокол доступа к каталогам») – протокол прикладного уровня для доступа к службе каталогов X.500, разработанный IETF как облегчённый вариант разработанного ITU-T протокола DAP |
| Mongo | [Документоориентированная](https://ru.wikipedia.org/wiki/%D0%94%D0%BE%D0%BA%D1%83%D0%BC%D0%B5%D0%BD%D1%82%D0%BE%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94) [система управления базами данных](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), не требующая описания схемы таблиц. Считается одним из классических примеров [NoSQL](https://ru.wikipedia.org/wiki/NoSQL)-систем, использует [JSON](https://ru.wikipedia.org/wiki/JSON)-подобные документы и схему базы данных. Написана на языке [C++](https://ru.wikipedia.org/wiki/C%2B%2B). Применяется в веб-разработке, в частности, в рамка [JavaScript](https://ru.wikipedia.org/wiki/JavaScript)-ориентированного стека [MEAN](https://ru.wikipedia.org/wiki/MEAN) |
| MS SQL | СУБД реляционного типа. Для манипуляции данными используется специально разработанный язык Transact-SQL |
| PostgreSQL | [Свободная](https://ru.wikipedia.org/wiki/%D0%A1%D0%B2%D0%BE%D0%B1%D0%BE%D0%B4%D0%BD%D0%BE%D0%B5_%D0%9F%D0%9E) [объектно-реляционная](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D1%80%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94) [СУБД](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) |
| Prometeus | Система мониторинга серверов и программ с открытым исходным кодом |
| Python | Интерпретируемый язык программирования |
| Quarkus | Полная, многоплатформенная среда проектирования, легко адаптируемая к требованиям конкретного проекта |
| SDK | Набор средств разработки, позволяющий специалистам по программному обеспечению создавать приложения для определённого пакета программ, программного обеспечения базовых средств разработки, аппаратной платформы, компьютерной системы, игровых консолей, операционных систем и прочих платформ |
| Sparx | Sparx Enterprise Architect (EA) - инструмент визуального моделирования и проектирования, основанный на OMG UML |
| Swagger | фреймворк для спецификации RESTful API |
| UI | (Unique identifier) – уникальный идентификатор |
| БД | База данных |
| Бэкенд | (англ. back-end) – программно-аппаратная часть сервиса. Бэкенд отвечает за осуществление функционирования внутренней части веб-сайта |
| ОС | Операционная система |
| ПК | Персональный компьютер |
| ПО | Программное обеспечение |
| СУБД | Система управления базами данных |
| Фронтенд | (англ. front-end) – клиентская сторона пользовательского интерфейса к программно-аппаратной части сервиса |