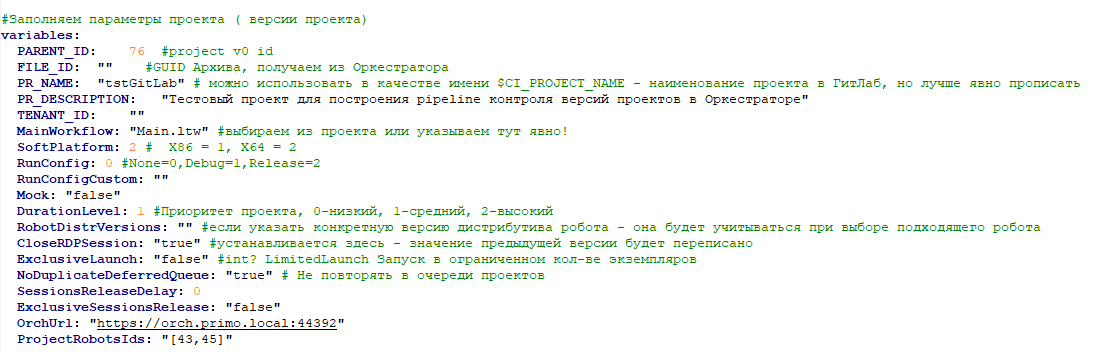
# Контроль версий проектов Оркестратора (на примере GitLab)

Данный документ рассматривает вариант CI/CD пайплайна (далее - пайплайн) версий проектов робота на примере GitLab - системы управления репозиториями кода на git. Предполагается наличие у заказчика развернутого сервера GitLab, с которого доступен по http сервер Оркестратора. Помимо имеющихся возможностей загрузки версий проектов в UI Оркестратора или из Primo.Studio, добавляется возможность автоматической загрузки версий из репозитория.

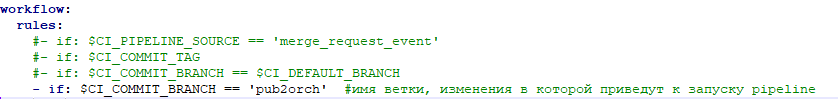
# Предварительная настройка пайплайна

Для каждого проекта робота необходимо завести проект в GitLab, в каждом проекте необходимо создать/определить релизную ветку, из которой версии проекта будут публиковаться в Оркестратор. В релизную ветку заливается проект робота и файл с расширением yml (далее - scripts), который содержит скрипты пайплайна. В файле scripts необходимо обязательно заполнить переменные проекта для корректной загрузки версии в Оркестратор. Ниже приведен список переменных:

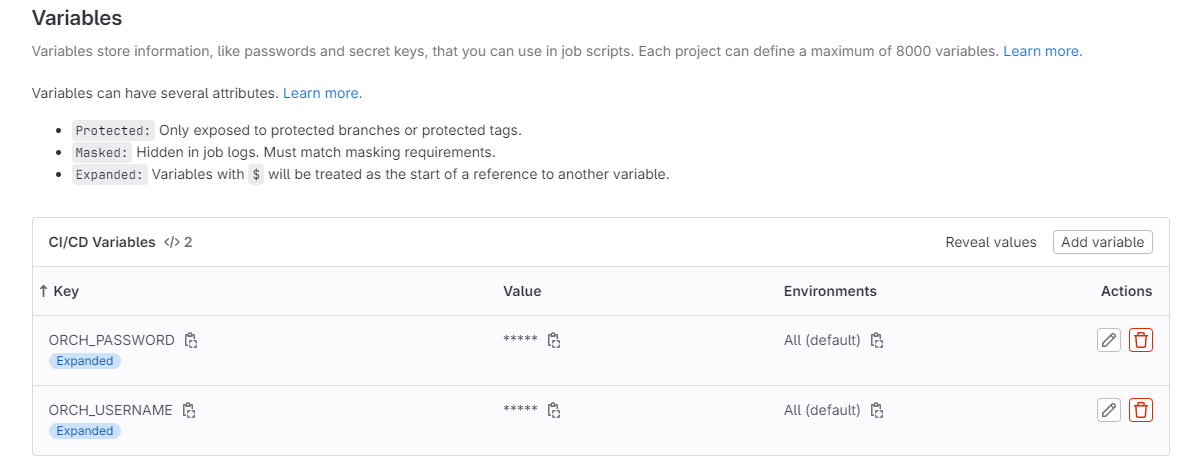


Основная часть переменных взяты из UI Оркестратора и являются параметрами проекта. Отдельно стоит обратить внимание на переменную PARENT\_ID, в которую необходимо записать идентификатор нулевой версии проекта (v0) из Оркестратора, если версия v0 удалена, то необходимо записать идентификатор минимальной версии. Узнать идентификатор версии можно в Оркестраторе, он отобразится в адресной строке браузера при переходе на форму редактирования соответствующей версии проекта, например, отобразится <https://orch.primo.local:44392/projects/all/76>, значит идентификатор версии проекта равен 76. Таким образом для каждого проекта в репозитории необходимо предварительно залить версию v0 в UI Оркестратора или из Primo.Studio чтобы получить PARENT\_ID.

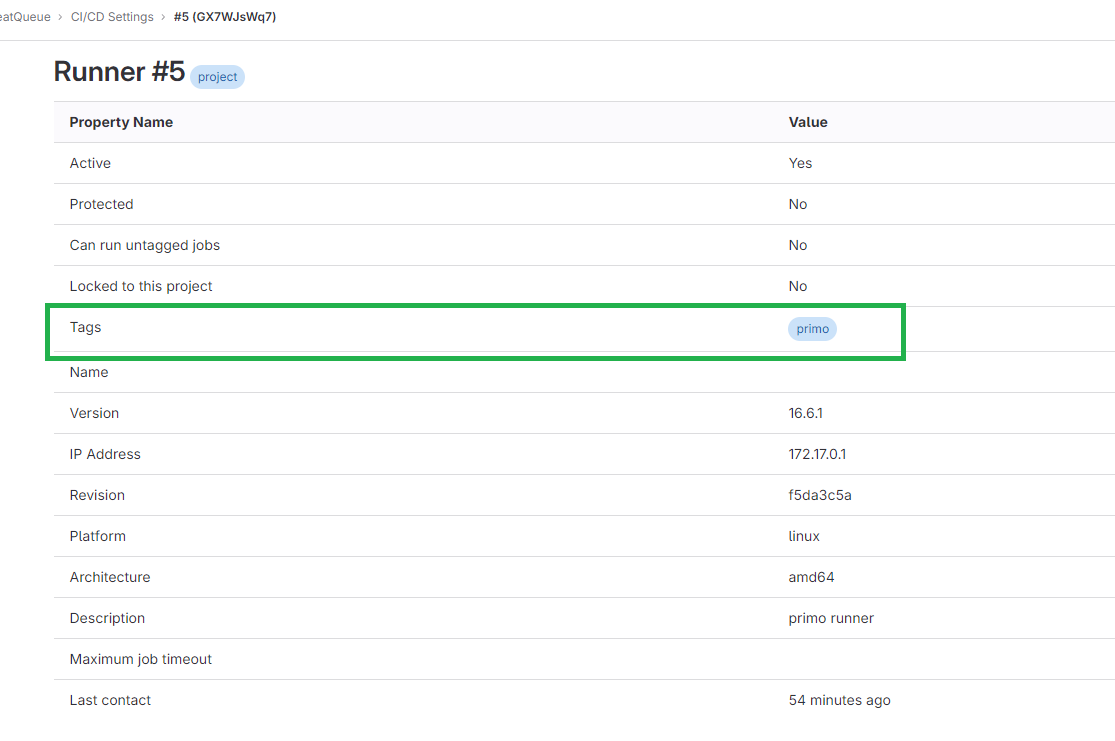
Также в файле scripts необходимо указать имя релизной ветки и действие, на которое будет запускаться пайплайн. Это могут такие действие создание тега, завершение запроса на слияние и другие. По умолчанию выбрано любое изменение в релизной ветке.



В переменных проекта необходимо добавить учетную запись Оркестратора с правами на редактирование проектов. ORCH\_PASSWORD и ORCH\_USERNAME будут использоваться в запросах к Оркестратору.

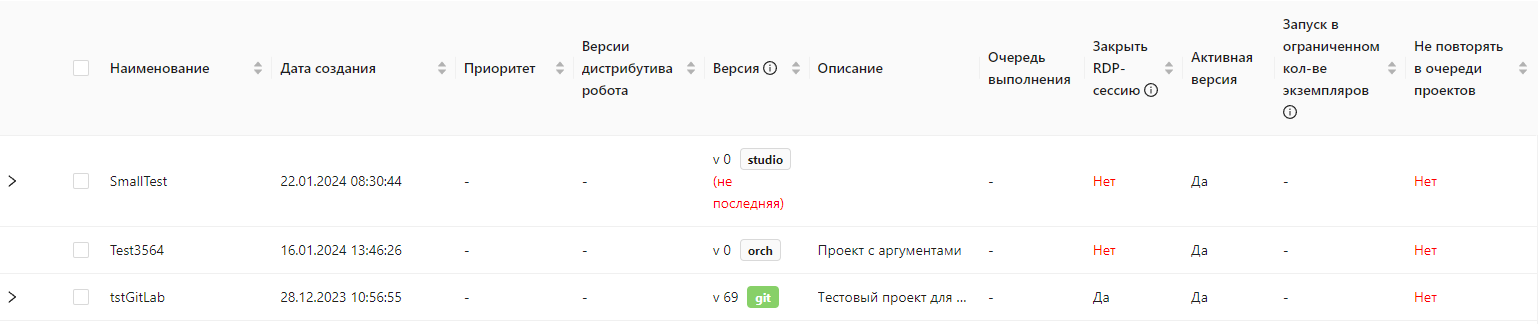


Для запуска пайплайна необходимо добавить Runner (в данном случае используется shell runner Linux) в настройках CI/CD проекта GitLab. В раннере необходимо установить тег, например, primo, который прописан в скриптах пайплайна, чтобы отдельные секции (jobs) скриптов выполнялись только этим раннером. Пример раннера:



# Описание пайплайна

Пайплайн состоит из двух основных секций (jobs), в первой секции происходит подготовка архива проекта. Во второй секции по средствам запросов в API Оркестратора происходит публикация версии проекта. Сначала загружается в Оркестратор полученный в первой секции архив, потом из текущей активной версии проекта определяется список привязанных к проекту роботов, чтобы привязать их к новой версии. Список идентификаторов роботов проекта можно также закрепить в виде переменной ProjectRobotsIds. Новая версия с параметрами публикуется в Оркестраторе. Версии, загруженные из репозитория будут помечаться флагом git, версии, добавленные в UI Оркестратора или опубликованные из Primo.Studio будут помечаться флагами orch и studio соответственно:



Флаг распространяется только на архив проекта, так, если после загрузки версии проекта из репозитория отредактировать проект в Оркестраторе, например, изменить значение галок на форме редактирования или добавить/удалить привязанных роботов, такие действия не приведут к изменению флага на orch, он по-прежнему будет git, так как архив проекта остался прежним. Таким образом в Оркестраторе можно наглядно наблюдать за тем какие версии проектов загружены из репозитория, а значит полностью соответствуют его содержимому, а какие добавлены руками. Автоматическая публикация позволит избежать часть ошибок, которые допускаются при ручном добавлении версии, например, если не привязать роботов к новой версии, то проект, запускаемый по заданию не запустится, если используется стратегия «Только привязанные роботы», если при этом не будет установлена галка «Не повторять в очереди», то проект будет накапливаться в очереди, пока это не станет серьезной проблемой в работе Оркестратора или пока не вмешается оператор. Текущий вариант не содержит автотестов, но они могут быть добавлены в пайплайн при необходимости.

В ближайших версия Оркестратора планируется добавление фильтра по флагу источника публикаций на форме проектов, что позволит быстро находить проекты не синхронизированные или отсутствующие в репозитории.

Таким образом представленный пайплайн позволяет следить за источником публикации версий проектов, автоматизирует публикацию, позволяет избежать часть ошибок, связанных с человеческим фактором, вести единый репозиторий проектов используя git.

Описан самый простой (базовый) пайплайн, при уточнении требований конкретных заказчиков он может быть доработан.