

КРАТКОЕ ОПИСАНИЕ ПРОГРАМНОГО  
ОБЕСПЕЧЕНИЯ  
VI.QUBE  
METACONTROL

# ОГЛАВЛЕНИЕ

|  |    |
|--|----|
| ВВЕДЕНИЕ.....  | 3  |
| ГЛОССАРИЙ.....   | 3  |
| 1. ФУНКЦИОНАЛЬНЫЕ ВОЗМОЖНОСТИ METACONTROL.....               | 5  |
| 1.1. Структура MetaControl .....                             | 5  |
| 1.2. Преимущества использования компонента MetaControl ..... | 5  |
| 2. ТАБЛИЦЫ METACONTROL.....                                  | 6  |
| 2.1. Настраиваемые таблицы MetaControl .....                 | 8  |
| 2.2. Таблица логов MetaControl .....                         | 9  |
| 2.3. Таблицы дескрипторы MetaControl.....                    | 10 |
| 2.4. Процедуры MetaControl.....                              | 11 |
| 2.4.1. Назначение процедур и функций MetaControl .....       | 12 |
| 2.4.2. Процесс работы процедур .....                         | 12 |

## ВВЕДЕНИЕ

MetaControl – компонент, предназначенный для информирования пользователей путем рассылки уведомлений о произведенных контролях. Могут производиться рассылки о самом факте контроля, о количестве расхождений, список определенных данных или же полноценный файл с данными.

В документе приведено описание компонента и принципы работы с ним. Рассмотрены примеры загрузки данных с помощью компонента из разных источников.

Изучение данного документа позволит понять принцип работы компонента.

## ГЛОССАРИЙ

|    |                             |  |
|----|-----------------------------|--|
| 1. | MetaControl<br>BI.Qube      | - Инструмент, предназначенный для информирования пользователей путем рассылки уведомлений о произведенных контролях.   |
| 2. | MetaOrchestrator<br>BI.Qube | - Инструмент управления процессами.  |
| 3. | Датасет                     | Обработанный и структурированный массив данных   |
| 4. | Процедура                   | Объект базы данных, представляющий собой набор SQL-инструкций, который компилируется один раз и хранится на сервере  |
| 5. | Представление               | Виртуальная таблица, содержимое которой определяется запросом  |
| 6. | Функция                     | Подпрограмма, которая принимает параметры, выполняет действие, например сложное вычисление, и возвращает результат этого действия в виде значения. Возвращаемое значение может быть либо единичным скалярным значением, либо результирующим набором. |
| 7. | Постконтроль                | Набор функций, процедур или представлений, которые вызываются после выполнения ETL процесса.   |
| 8. | smtp-сервер                 | Сетевой протокол, который отвечает за отправку писем   |

## 2. ЦЕЛИ И НАЗНАЧЕНИЕ METACONTROL

Цель компонента MetaControl заключается в своевременном уведомлении пользователей о статусе событий, происходящих с данными на основе правил, создаваемых пользователями.

Уведомления MetaControl отправляются путем рассылки о произведенных контролях с использованием почтового сервера и telegram-канала.

Компонент MetaControl входит в состав системы VI.Qube и может эксплуатироваться как отдельный компонент, так и в составе системы, так и под управлением компонента MetaOrchestrator, в такой конфигурации использование компонента является наиболее эффективной.

# 1. ФУНКЦИОНАЛЬНЫЕ ВОЗМОЖНОСТИ METACONTROL

## 1.1. Структура MetaControl

Для работы MetaControl необходимо создать правило контроля данных в одном из следующих форматов:

- Процедура (P);
- Функция (F);
- Представление (V).

На основе полученных данных будет произведено формирование датасета – тела сообщения о произведенном контроле. После чего с настроенного smtp-сервера произойдет рассылка писем на указанные адреса, в необходимом формате.

Ряд форматов, поддерживаемых компонентом:

- HTML – вся информация будет содержаться в самом письме;
- PDF – набор данных будет загружен в pdf формате и прикреплен к сообщению в telegram-канале или почтовому сообщению.

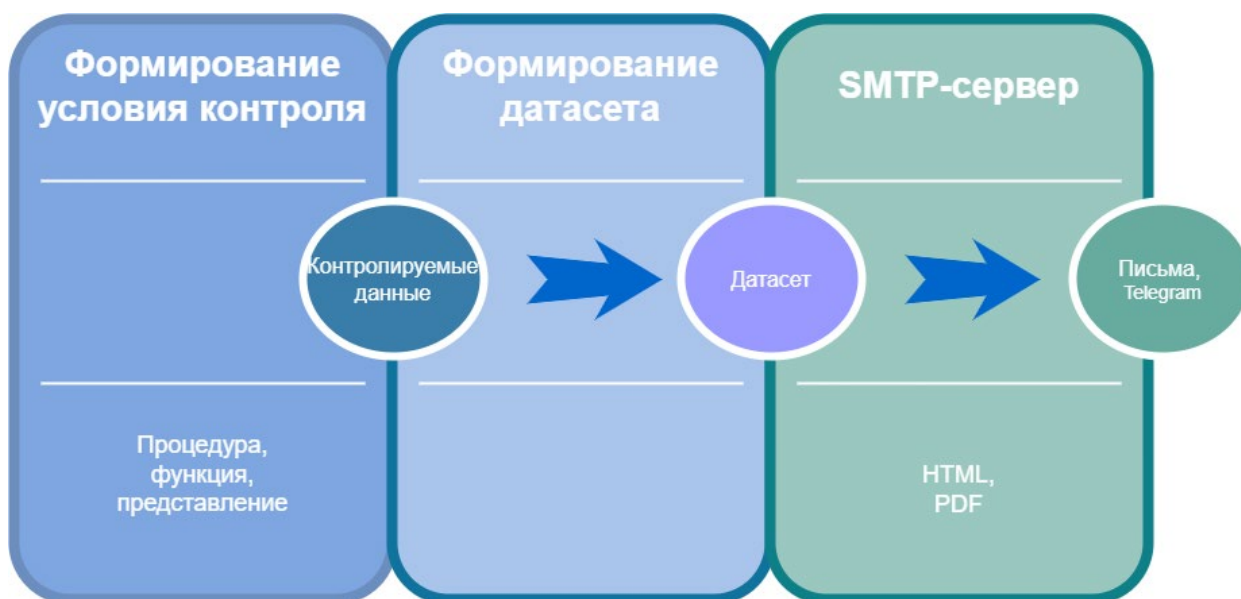


Рисунок 1. Структура MetaControl

## 1.2. Преимущества использования компонента MetaControl

1. MetaControl имеет умное формирование писем, поэтому несколько контролей он объединит в одно письмо и отправит на указанные адреса.

Например, есть контроли 1p, 2p, 3p с соответствующими группами адресов 1m, 2m, 3m. В группах находятся пересекающиеся адреса:

1p рассылка на 1m = (mail\_1@sample.com)

2p рассылка на 2m = (mail\_1@sample.com, mail\_2@sample.com)

3p рассылка на 3m = (mail\_1@sample.com, mail\_2@sample.com, mail\_3@sample.com).

2. MetaControl определит какие контроли необходимо разослать на адреса, после чего сгруппирует контроли по конечным адресам, тем самым каждый пользователь получит лишь ОДНО письмо содержащее все контроли, в которые попадает его адрес:

mail\_1@sample.com – Письмо с контролями: 1p, 2p, 3p

mail\_2@sample.com – Письмо с контролями: 2p, 3p

mail\_3@sample.com – Письмо с контролями: 3p

3. MetaControl имеет возможность задавать параметры в источниках данных, что добавляет динамики в выполнение постконтролей.

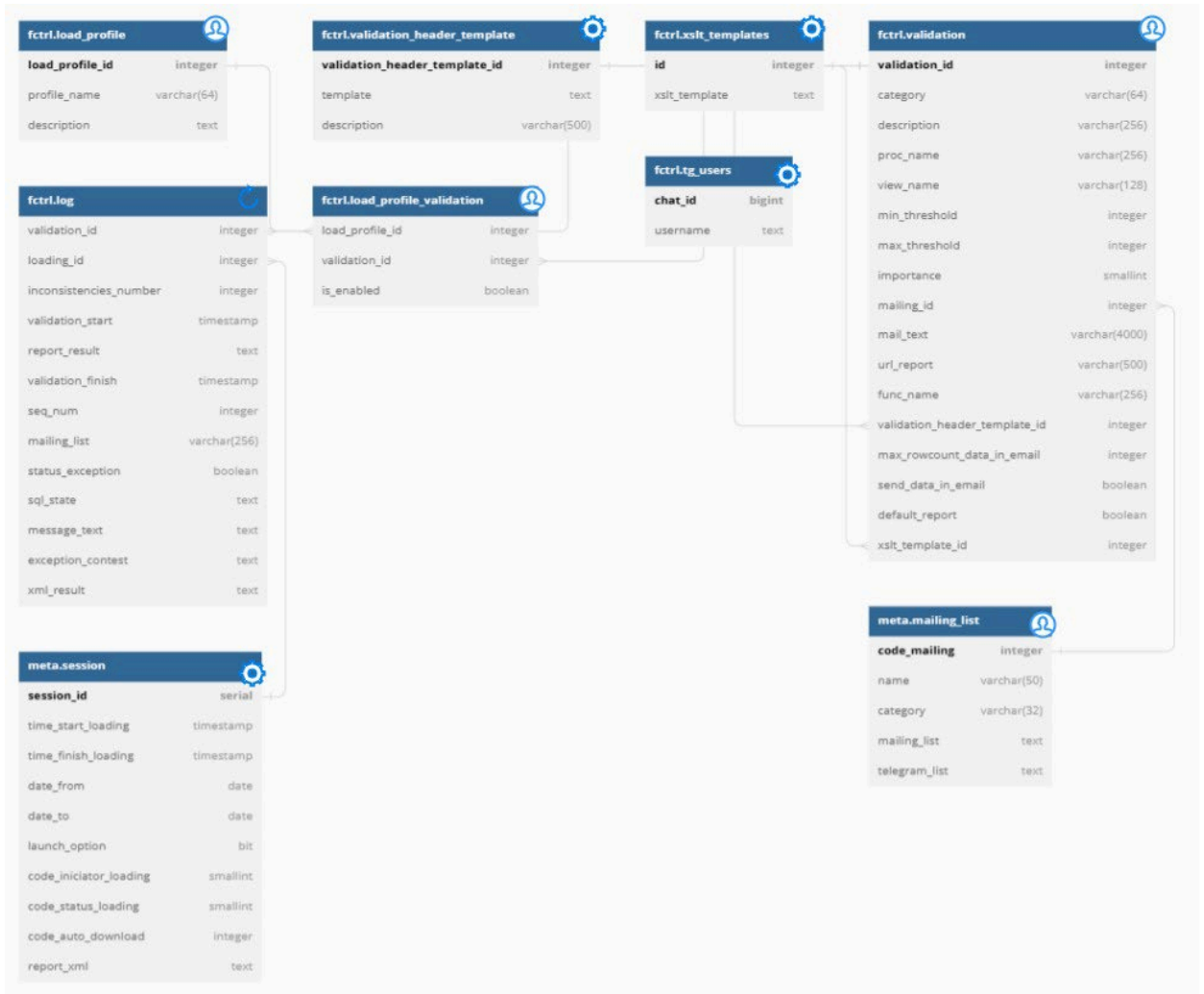
## 2. ТАБЛИЦЫ METACONTROL

Таблицы MetaControl делятся на три категории:

*Настроечные таблицы* – данные вносит пользователь в соответствии с правилами заполнения.

*Таблицы логов (заполняются автоматически)* – данные вносит система по итогам выполнения очередной сессии.

*Таблицы дескрипторы (заполнены разработчиком)* – данные внесены предварительно в соответствии с реализованным функционалом.






Настрочные таблицы   
 Таблицы логов   
 Таблицы дескрипторы 

Рисунок 2. Структура таблиц MetaControl

## 2.1. Настраиваемые таблицы MetaControl

1. «fctrl.validation» – таблица, содержащая список контролей, их название, описание.

**Поля таблицы:**

| Имя столбца                     | Тип данных | Источник      | Назначение  |
|---------------------------------|------------|---------------|---|
| validation_id                   | int        | Автоинкремент | Уникальный ID проверки  |
| category                        | text       | Вручную       | Категория проверки. Контроли группируются в письмах в рамках одной категории.   |
| description                     | text       | Вручную       | Описание проверки   |
| proc_name                       | text       | Вручную       | Наименование процедуры для произведения контроля  |
| view_name                       | text       | Вручную       | Наименование представления для произведения контроля  |
| func_name                       | text       | Вручную       | Наименование функции для произведения контроля  |
| min_threshold,<br>max_threshold | int        | Вручную       | Границы допустимого кол-ва строк (если число записей в таблице или процедуре не входит в диапазон [min; max] то происходит рассылка               |
| importance                      | boolean    | Вручную       | Степень важности контроля (пометка «важно» в письме)  |
| mailing_id                      | int        | Вручную       | ID из meta.mailing_list(email/telegram тех, кому нужна рассылка)  |
| mail_text                       | text       | Вручную       | Дополнительный текст письма   |
| url_report                      | text       | Вручную       | URL на отчет (если есть) по проверке  |
| send_data_in_email              | boolean    | Вручную       | Вкл/откл отправки всех данных из проверки   |
| max_rowcount_data_in_email      | int        | Вручную       | Ограничение максимального кол-ва строк данных в письме  |
| validation_header_template_id   | int        | Вручную       | ID шаблона из fctrl.validation_header_template  |
| default_report                  | boolean    | Вручную       | Пометка стандартного отчета.  |
| xslt_template_id                | int        | Вручную       | Позволяет задавать формат xslt шаблона, что обеспечивает определенный шаблон отображения данных у пользователя. ID из таблицы fctrl.xslt template |



## 2. «meta.mailing\_list» – справочник адресов для рассылки.

### Поля таблицы:

| Имя столбца   | Тип данных | Источник      | Назначение  |
|---------------|------------|---------------|---|
| code mailing  | int        | Автоинкремент | Уникальный ID рассылки  |
| name          | text       | Вручную       | Наименование набора адресов   |
| category      | text       | Вручную       | Категория рассылки  |
| mailing_list  | text       | Вручную       | Список адресов для рассылки. В один список может входить множество адресов, задаваемых через запятую. Например: sample@it.ru, exmpl@it.ru   |
| telegram_list | text       | Вручную       | Список telegram-аккаунтов для рассылки. В один список может входить множество адресов, задаваемых через запятую. Например: exmpl, testexmpl |

## 3. «fctrl.load\_profile\_validation» – связующая таблица, в которой настраиваются отношения профиль-контроль, также имеет возможность отключения контроля.

### Поля таблицы:

| Имя столбца     | Тип данных | Источник | Назначение                        |
|-----------------|------------|----------|-----------------------------------|
| load_profile_id | int        | Вручную  | Уникальный ID профиля             |
| validation_id   | int        | Вручную  | Код контроля                      |
| is_enabled      | boolean    | Вручную  | Вкл/откл проверки из постконтроля |

## 4. «fctrl.load\_profile» – таблица с описанием профилей.

### Поля таблицы:

| Имя столбца     | Тип данных | Источник      | Назначение            |
|-----------------|------------|---------------|-----------------------|
| load_profile_id | int        | Автоинкремент | Уникальный ID профиля |
| profile_name    | text       | Вручную       | Наименование профиля  |
| description     | text       | Вручную       | Описание профиля      |

## 2.2. Таблица логов MetaControl

«fctrl.log» – таблица, определяющая функциональные особенности работы компонента.

### Поля таблицы:

| Имя столбца   | Тип данных | Источник      | Назначение             |
|---------------|------------|---------------|------------------------|
| validation_id | int        | Автоматически | Уникальный ID проверки |
| session_id    | int        | Автоматически | Код загрузки           |

|                        |           |               |  |
|------------------------|-----------|---------------|--|
| inconsistencies_number | int       | Автоматически | Количество расхождений данных на хранилище с эталоном  |
| validation_start       | timestamp | Автоматически | Время начала контроля  |
| report_result          | text      | Автоматически | Ссылка на отчет  |
| validation_finish      | timestamp | Автоматически | Время окончания контроля   |
| seq_num                | int       | Автоматически | Номер последовательности контроля (актуален при наличии нескольких контролей)                      |
| mailing_list           | int       | Автоматически | ID в таблице fctrl.mailing_list, которая содержит либо список email-адресов или telegram-аккаунты. |
| status_exception       | boolean   | Автоматически | Отчет об ошибке.   |
| sql_state              | text      | Автоматически | Номер ошибки. Заполняется в случае значения поля «status_exception» «true».                        |
| message_text           | text      | Автоматически | Описание ошибки. Заполняется в случае значения поля «status_exception» «true».                     |
| exception_context      | text      | Автоматически | Контекст кода. Заполняется в случае значения поля «status_exception» «true».                       |
| xml_result             | text      | Автоматически | XML-код, в который оборачивается набор данных  |

## 2.3. Таблицы дескрипторы MetaControl

1. «meta.session» – таблица с описанием сессии загрузки.

Поля таблицы:

| Имя столбца            | Тип данных | Источник      | Назначение  |
|------------------------|------------|---------------|---|
| session_id             | int        | Автоинкремент | Уникальный ID загрузки  |
| time_start_loading     | timestamp  | Автоматически | Время начала загрузки   |
| time_finish_loading    | timestamp  | Автоматически | Время окончания загрузки  |
| date_from              | timestamp  | Автоматически | Период, за который ETL процесс загружает данные («дата с»)        |
| date_to                | timestamp  | Автоматически | Период, за который ETL процесс загружает данные («дата по»)       |
| launch_option          | bit        | Автоматически | Статус действия загрузки  |
| code_initiator_loading | smallint   | Автоматически | Показатель кем была инициирована загрузка (Вручную/автоматически) |

|                     |          |               |  |
|---------------------|----------|---------------|--|
| code_status_loading | smallint | Автоматически | Статус загрузки  |
| code_auto_download  | int      | Автоматически | Если загрузка была инициирована автоматически, показывает её статус(ETL/по расписанию) |
| report_xml          | text     | Автоматически | XML-Отчет  |

2. «fctrl.validation\_header\_template» – таблица, содержащая шаблон для настройки вывода полей при выполнении контроля.

**Поля таблицы:**

| Имя столбца                   | Тип данных | Источник      | Назначение            |
|-------------------------------|------------|---------------|-----------------------|
| validation_header_template_id | int        | Автоинкремент | Уникальный ID шаблона |
| template                      | text       | Вручную       | Шаблон (html формат)  |
| description                   | text       | Вручную       | Описание шаблона      |

3. «fctrl.xslt\_templates» - таблица, содержащая XSLT-шаблоны для формирования html-результата.

| Имя столбца       | Тип данных | Источник      | Назначение                 |
|-------------------|------------|---------------|----------------------------|
| xslt_templates_id | int        | Автоинкремент | Уникальный ID XSLT-шаблона |
| xslt_templates    | text       | Вручную       | XSLT-шаблон                |

4. «fctrl.tg\_users» - таблица, содержащая id и username пользователей, которые инициировали диалог с telegram-ботом

| Имя столбца | Тип данных | Источник      | Назначение                    |
|-------------|------------|---------------|-------------------------------|
| chat_id     | int        | Автоматически | ID чата                       |
| username    | text       | Автоматически | Логин пользователя в Telegram |

## 2.4. Процедуры MetaControl

Основными программными продуктами являются:

1. процедуры
  - a. fctrl.init\_validation,
2. функции
  - a. fctrl.recive\_mailing\_list,
  - b. fctrl.recive\_mailing\_content
  - c. fctrl.recive\_telegram\_list.
  - d. fctrl.get\_data\_as\_xml

### *2.4.1. Назначение процедур и функций MetaControl*

- Процедура «**fctrl.init\_validation**» – получает информацию из настроечной таблицы, вызывает определённые представления функции и процедуры и записывает эти данные в таблицу логов.
  - Функция «**fctrl.recive\_mailing\_content**» – позволяет получить содержание писем для определенных адресов отправки по коду загрузки
  - Функция «**fctrl.recive\_mailing\_list**» – позволяет получить список email-адресов, принадлежащие определенному коду загрузки.
  - Функция «**meta.current\_session\_code**» позволяющая получить текущий код загрузки для записи этой информации в таблицу логов
  - Функция «**meta.etl\_is\_launched**» – проверяет статус etl-процесса.
  - Функция «**meta.sat\_parameters**» – функция, которая вставляет параметры в таблицу «**meta.params**».
  - Функция «**meta.split**» – служит разделителем для данных поля «**mailing\_list**» таблицы «**mailing\_list**».
  - Процедура «**fctrl.get\_data\_as\_xml**» – преобразует в xml-формат набор данных, полученных при выполнении контроля.

### *2.4.2. Процесс работы процедур*

Процедура «**fctrl.init\_validation**» производит обработку записей «**fctrl.validation**» после чего добавляет их результат в таблицу «**fctrl.Log**».

Приложение определяет количество записей попавших в «**fctrl.Log**», в случае ненулевого результата запускается C# скрипт формирования писем путем извлечения данных из «**fctrl.Log**».

Готовые письма рассылаются на указанные для них адреса.