

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Тольяттинский государственный университет»

Институт Математики, физики и информационных технологий  
(наименование института полностью)

Кафедра «Прикладная математика и информатика»  
(наименование)

02.03.03 Математическое обеспечение и администрирование информационных систем  
(код и наименование направления подготовки, специальности)

WEB-дизайн и мультимедиа  
(направленность (профиль)/специализация)

## **ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)**

на тему «Разработка ПО для системы учета оборудования и материалов (на примере фармацевтической компании “Озон”)»

Обучающийся

И.С. Грузков

(И.О. Фамилия)

(личная подпись)

Руководитель

М.А. Тренина

(ученая степень, звание, И.О. Фамилия)

Консультант

канд. филол. наук, доцент М.М. Бажутина

(ученая степень, звание, И.О. Фамилия)

## Аннотация

Тема работы: «Разработка ПО для системы учета оборудования и материалов (на примере фармацевтической компании “Озон”)».

Бакалаврскую работу выполнил: Грузков Илья Сергеевич.

Актуальность работы заключается в создании системы мониторинга и учета оборудования вне локальной системы.

Целью данной бакалаврской работы является создание информационной системы учета оборудования на примере коммутаторов D-Link и принтеров KYOCERA, а также их мониторинг в сети, мониторинга расходных материалов необходимых для использования в изготовлении на производстве.

Задачи бакалаврской работы:

- изучение актуальности решения задачи;
- изучение существующих методов реализации системы мониторинга учета;
- проектирование и структурирование базы данных для ресурсов для внедрения в систему;
- реализация системы.

В первой главе бакалаврской работы рассматриваются системы и их инструменты, которые необходимы для решения поставленной задачи.

Во второй главе бакалаврской работы разрабатывается система учета и мониторинга оборудования в системе Zabbix

В третьей главе бакалаврской работы создается макет мониторинга расходных материалов.

Бакалаврская работа состоит из 46 страниц, включая 30 рисунков, 1 приложения и списка используемой литературы, состоящий из 25 источников (в том числе 5 на иностранном языке).

## **Annotation**

The title of the bachelor's thesis «Development of software for the system of accounting for raw materials and materials (exported by the pharmaceutical company “Ozon”)».

Bachelor's work completed: Ilya Sergeevich Gruzkov.

The relevance of the work is regulated by the creation of a system for monitoring and recording equipment outside the surveillance system.

The purpose of this bachelor's work is to create an information system for accounting for equipment using the example of D-Link switches and KYOCERA printers, as well as their monitoring in the network, monitoring consumables necessary for use in manufacturing in production.

Tasks of bachelor's work:

- studying the relevance of solving the problem;
- study of existing methods for implementing the accounting monitoring system;
- designing and structuring a database for resources to be introduced into the system;
- system implementation.

The first chapter of the bachelor's work deals with systems and their tools that are necessary to solve the problem.

In the second chapter of the bachelor's work, a system for accounting and monitoring equipment in the Zabbix system is being developed.

In the third chapter of the bachelor's work, a layout for monitoring consumables is created.

Bachelor's thesis consists of 46 pages, including 30 figures, 1 appendix and a list of references, consisting of 25 sources (including 5 in a foreign language).

## Оглавление

Введение.....	5
Глава 1 Описание фармацевтической компании ООО «Озон», постановка задачи, обнаружение проблемы изучение систем под поставленную задачу..7	
1.1 Общие сведения ООО «Озон».....	7
1.2 Постановка задачи на разработку системы учета и ее актуальность.....	8
1.3 Система SCADA и ее возможности.....	9
1.4 Zabbix - система мониторинга и ее возможности.....	11
1.5 Протокол SNMP, как решение мониторинга.....	14
1.6 База данных информации управления – MIB.....	16
1.7 Выбор для использования базы данных.....	18
1.8 MySQL база данных, для решения дополнительной задачи.....	22
Глава 2 Разработка системы учета и мониторинга оборудования в системе Zabbix.....	25
2.1 Создание мониторинга коммутаторов D-Link.....	25
2.2 Подключение D-Link и KYOCERA в Zabbix.....	26
2.3 Подключение операционной системы в Zabbix.....	32
Глава 3 Создание MVP - проекта по графическому отображению состояния материальных запасов.....	36
Заключение.....	42
Список используемой литературы.....	43
Приложение А Код создания базы данных в MySQL Workbench.....	46

## Введение

История появления первых учетных систем насчитывает много веков. Системы управления учетными данными прошли огромный исторический этап, от заполнения простыми ручными способами до современных и знакомых нам систем в компьютере. На сегодня, масштаб работы, который может выполнить система учета, в разы превосходит возможности прошлых десятков лет и продолжает модернизироваться таким образом, что вмешательство человека уже не заметно, все что нужно, это выбрать нужный пункт и посмотреть всю необходимую и нужную информацию.

Новая рыночная экономика для любого предприятия, заставляет постоянно модернизировать и адаптировать систему учета. Изменчивость рыночной экономики диктует новые правила для систем учета, появляется необходимость в быстром анализе и изменении данных в больших количествах за короткий промежуток времени. Также не стоит забывать об оптимизации и перестройки внутренней микроэкономики на фоне изменений.

Актуальность работы заключается в создании системы мониторинга и учета оборудования вне локальной системы.

Целью бакалаврской работы является создание информационной системы учета оборудования (компьютеров, принтеров и т.п.), а также мониторинг вне локального носителя и создание MVP проекта мониторинга расходных материалов необходимых для использования в изготовлении.

Задачи бакалаврской работы:

- изучение актуальности решения задачи;
- изучение существующих методов реализации системы мониторинга учета;
- проектирование и структурирование базы данных для ресурсов для внедрения в систему;
- реализация системы.

В первой главе бакалаврской работы описывается фармацевтическая компания «Озон», рассматриваются системы и их инструменты, которые необходимы для решения поставленной задачи, рассматривается система SCADA и Zabbix, описывается протокол SNMP, изучается база данных информации управления MIB, а также выбирается база данных для поставленной задачи, описывается MySQL.

Во второй главе бакалаврской работы разрабатывается система учета и мониторинга оборудования в системе Zabbix на примере коммутаторов D-Link и принтеров KYOCERA.

В третьей главе бакалаврской работы создается макет мониторинга расходных материалов, а также графическое отображение состояний емкости материальных запасов.

# **Глава 1 Описание фармацевтической компании ООО «Озон», постановка задачи, обнаружение проблемы изучение систем под поставленную задачу**

## **1.1 Общие сведения ООО «Озон»**

«Озон» – это одна из ведущих фармацевтической компаний России по разработке и производству современных лекарственных средств и выпуску лекарственных препаратов, большую часть которых составляют дженерики (generiki) – аналоги патентованных готовых лекарственных средств (ГЛС). Good Manufacturing Practice (GMP) стандарты дают возможность создавать препараты наилучшего качества, с доступностью всем слоям населения, за счет доступной цены. Все производство полностью соответствует требованию стандарту GMP и разрабатывается на новом, современном оборудовании известных мировых производителей таких как: Block West Holding; Gea; System Air; Marchesini Group S.p.A.; Viani; Gs; Coster и др.

Объемы компании по производству упаковок в год составляют приблизительно 500млн., а продуктовый портфель составляет более 300 названий препаратов ГЛС. Каждый год количество, объемы и появление новых препаратов, выпускаемых в компании, увеличивается.

Компания имеет следующие характеристики:

Сокращенное наименование: ООО «Озон».

Юридический адрес: 445351, Самарская обл., город Жигулевск, ул. Песочная, д.11

Реквизиты:

ИНН: 6345002063

КПП: 6608500001

ОГРН: 102630321634

Тел./факс: (84862) 3 -41 -09, 2 -90 -00

e-mail: [ozon@ozon-pharm.ru](mailto:ozon@ozon-pharm.ru).

На сегодня компания активно занимается выпуском твердых форм готовых лекарственных средств. Участие компании ООО «Озон» на больших международных выставках фармацевтики привлекает и открывает новые рынки сбыта, а также увеличивает количество продаваемых лекарств и продвигает разработанные лекарства на рынок [19].

## **1.2 Постановка задачи на разработку системы учета и ее актуальность**

Каждая корпорация стремится развивать свой бизнес, находить новых спонсоров, развивать свою продукцию. Сегодня, каждая уважающая корпорация стремится цифровизировать и оптимизировать свои бизнес процессы [7]. Сложно представить, как за несколько десятков лет от письменного учета, учет превратился в сложные, большие и очень удобные для пользователя системы. Теперь, чтобы посчитать или занести новое оборудование в систему учета, достаточно нажать две клавиши. Но каждый год несмотря на это системы улучшаются, дорабатываются и изменяются.

Важной задачей стоит контроль и мониторинг данных. Сложные системы хранятся на локальных серверах и их гибкость напрямую зависит от надлежащего ухода и умений администрирования. Еще несколько лет назад трудно было представить систему не локальную, а облачную. В своей работе мне и стоит задача по созданию системы сбора информации, через сеть. Система должна собирать информацию о пользователях не на локальных машинах, а через сеть. С помощью шаблонов по моделям оборудования, а также при использовании протокола SNMP для всего оборудования в сети и их ОС, можно просматривать всю полученную информацию. Основное оборудование для использования – это коммутаторы D-Link, принтера KYOCERA, операционные системы Linux.



Корпоративная сеть распределенная, включает множество хостов сервисов и точек отказа требующих постоянного мониторинга технических службы, своевременность обнаружение проблемы является залогом высокой корпоративной доступности [2]. С целью оперативного выявления проблемы необходимо использовать мониторинг системы, который считается неотъемлемой частью бизнес процесса.

Дополнительной задачей стоит создание макета мониторинг состояния применяемой в работе расходных материалов.

### **1.3 Система SCADA и ее возможности**

Система SCADA используется для мониторинга и контроля большого количества удаленных объектов или одного, но территориально распределенного объекта. Такими объектами обычно считаются: газопроводы, нефтепроводы, распределительные подстанции и т.п. [6].

«SCADA – от англ. Supervisory Control And Data Acquisition (диспетчерское управление и сбор данных) – это программный пакет, предназначенный для разработки или обеспечения работы в реальном времени систем сбора, обработки, отображения и архивирования информации об объекте мониторинга или управления. SCADA может являться частью АСУ ТП, АСКУЭ, системы экологического мониторинга, научного эксперимента, автоматизации здания и т. д.» [6].

Основной задачей данной системы считается сбор информации об объектах далеко расположенных друг от друга, а также отслеживание необходимой информации в едином диспетчерском центре. Помимо этого, вся информация хранится в системе долгосрочно путем архивации данных. Диспетчер может не только наблюдать за объектом, но и управлять, реагируя на различные ситуации [11]. Система SCADA способна обмениваться данными с устройствами связи с объектами (контроллерами); вести базу данных с технологической информацией; обеспечивает

возможность связи с внешними приложениями (электронные таблицы, СУБД); сигнализировать об неисправностях; генерировать и подготавливать отчеты о ходе технологического процесса; а также обрабатывать информацию и все это в реальном времени. На рисунке 1 представлена структура системы SCADA.

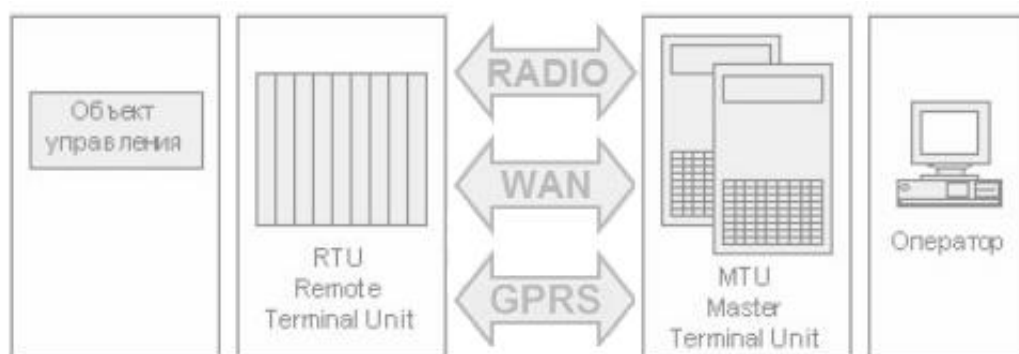


Рисунок 1 – Структура системы SCADA

На рисунке 1 видно, что:

- каждая SCADA-система состоит из трех компонентов: удалённый терминал (RTU – Remote Terminal Unit / Удаленный терминал), диспетчерский пункт управления (MTU – Master Terminal Unit / Главный терминал) и коммуникационную систему (CS – Communication System);
- удаленный терминал подключается напрямую к любому контролируемому объекту, и дает возможность управления в режиме реального времени. Терминал может быть любым, как обычным, примитивным датчиком, так, так и специализированным вычислительным комплексом;
- диспетчерский пункт управления осуществляет обработку данных и управление высокого уровня. Он обеспечивает человеко-

машинный интерфейс. Главный терминал может быть представлен компьютером и подключенным к нему принтером и т.п.;

- коммуникационная система используется для передачи данных с удаленного терминала на главный терминал и обратно. В качестве коммуникационной системы могут использоваться выделенные линии, радиосети, аналоговые телефонные линии. Зачастую устройства подключаются к нескольким сетям для обеспечения надёжности передачи данных.

Система SCADA отличный инструмент для отображения графиков, создания интерактивных схем производств, например, котельные и т.п., но система не может осуществлять мониторинг сетевых устройств, который стоит основной задачей [15]. Для этого действия необходим дополнительный набор инструментов помимо SCADA для вывода информации прямо в сетевой мониторинг.

#### **1.4 Zabbix - система мониторинга и ее возможности**

Zabbix—это система автоматизированного мониторинга за состоянием хостов сети, серверного и иного оборудования с записью истории изменений их состояний для дальнейшего разбора причин отказов оборудования. Открытое решение Zabbix было разработано для мониторинга персональных компьютеров, серверных узлов, web-серверов, СУБД (баз данных). А также других объектов локальной вычислительной сети (ЛВС— компьютерная сеть, покрывающая обычно относительно небольшую территорию или небольшую группу зданий) [10]. Zabbix появился в 1998 году, как внутренний проект в латвийском банке, 7 апреля 2001 года система была выпущена публично под лицензией General Public License [1].

В системе Zabbix есть:

- сервер Zabbix;

- web-интерфейс для управления мониторинга;
- агенты, которые можно установить на операционной систем Windows Linux или Unix;
- сервисы для работы с IPMI, SNMP, сетевые утилиты и HTTP запросы.

Количества компьютеров на крупных предприятиях может исчисляться сотнями, при этом техника может быть географически удалена друг от друга находясь не только в разных зданиях, но и в разных городах, и даже в разных странах. Несмотря на это, от системного администратора требуется регулярный контроль всей инфраструктуры. К самым простым параметрам за которыми необходимо регулярно следит это оперативная информация; о критическом состоянии ресурсов; проверка свободного дискового пространства на сервере или на клиентских компьютерах; ошибки журнала системы; состоянии служб на сервере; список программ, установленных на ПК; доступность узла, а также высокая загрузка процессора.

Система Zabbix готова для работы с большим количеством объектов информационной системы. Сервер Zabbix можно устанавливать на виртуальную машину, что решает огромное количество проблем, связанных с хранением данных и по мере необходимости увеличивать необходимый ресурс (ЦП, ОЗУ или диск) [4]. Система может работать с филиалами и аккумулировать всю информацию на центральном сервере, для этого используется прокси-сервер, который взаимодействует с объектами в удаленном офисе, а затем передает информацию на центральный сервер, а также он влияет на автономный агрегат данных [3].

Получение данных с узла или объекта, например, СУБД, web-сайт, принтер и т.п. можно выполнять различными способами, самый простой и распространенный вариант – это использование агента. При использовании агента на сервере пишется команда, которая будет выполняться на узле, на ПК будет выводиться результат выполнения данной команды и загружаться

на сервер. Для выполнения сложных многостраничных команд на клиенте в конфигурационный файл добавляется ключ и команда. Команда выполняется по запросу от сервера, а сервер в качестве запроса передает ключ, затем агент запускает команду, а сама команда возвращает результат работы, после агент отправляет полученную информацию на сервер в удобном для пользователя виде. Использование агента нужно, когда получение данных, которые не могут быть сформированы в одной команде, так, например, можно отслеживать количество активных сеансов.

Zabbix позволяет использовать SNMP протокол (Simple Network Management Protocol) для управления сетевыми устройствами в IP сетях. Данный протокол часто можно найти в маршрутизаторах, в коммутаторах, в принтерах, в серверах, в системном программном обеспечении и в специализированных контроллерах [23]. Использование HTTP запросов дает возможность изучить и просмотреть проверку работы web-сайтов, узнать доступность, а также отследить контент на странице. Этот функционал можно использовать для получения количества ресурса на картридже принтера, или получение состояния баланса у вашего интернет провайдера.

Как пример можно контролировать датчики температуры в серверной. Список всех доступных решений в Zabbix представлен на рисунке 2.

```
zabbix-agent  
zabbix-agent2  
zabbix-agent2-dbgSYM  
zabbix-agent-dbgSYM  
zabbix-apache-conf  
zabbix-cli  
zabbix-frontend-php  
zabbix-get  
zabbix-get-dbgSYM  
zabbix-java-gateway  
zabbix-js  
zabbix-js-dbgSYM  
zabbix-nginx-conf  
zabbix-proxy-mysql  
zabbix-proxy-mysql-dbgSYM  
zabbix-proxy-pgsql  
zabbix-proxy-pgsql-dbgSYM  
zabbix-proxy-sqlite3  
zabbix-proxy-sqlite3-dbgSYM  
zabbix-release  
zabbix-sender  
zabbix-sender-dbgSYM  
zabbix-server-mysql  
zabbix-server-mysql-dbgSYM  
zabbix-server-pgsql  
zabbix-server-pgsql-dbgSYM  
zabbix-sql-scripts  
zabbix-web-service
```

Рисунок 2 – Список всех доступных решений в Zabbix

Таким образом, система Zabbix позволяет мониторить и отслеживать любое необходимое устройство, от которого можно получать информацию только находясь лично в офисе.

### **1.5 Протокол SNMP, как решение мониторинга**

Simple Network Management Protocol (SNMP)– это протокол прикладного уровня, он делает возможным обмен данными между сетевыми устройствами.

«SNMP означает простой протокол управления сетью, он существует с 1988 года, поэтому он довольно хорошо используется. Протокол был разработан, чтобы позволить администраторам проводить мониторинг, контролировать производительность сети и изменять конфигурацию подключенных устройств, а также отслеживать текущее состояние сетевого оборудования, а также удаленно изменять настройки и конфигурации на оборудование» [4].

SNMP протокол был первоначально разработан для коммутаторов и маршрутизаторов, с тех пор он был распространен на широкий спектр других устройств, таких как компьютеры с Windows и Linux, принтеры и другие источники бесперебойного питания, а также многие другие вещи.

Протокол SNMP задумывался, как инструмент для управления администратором интернета, но гибкая архитектура SNMP (рисунок 2) предоставила возможность мониторинга всех сетевых устройств, а также дала возможность управления с одной консоли [9]. Это и способствовало распространению протокола. Архитектура протокола SNMP представлена на рисунке 3.

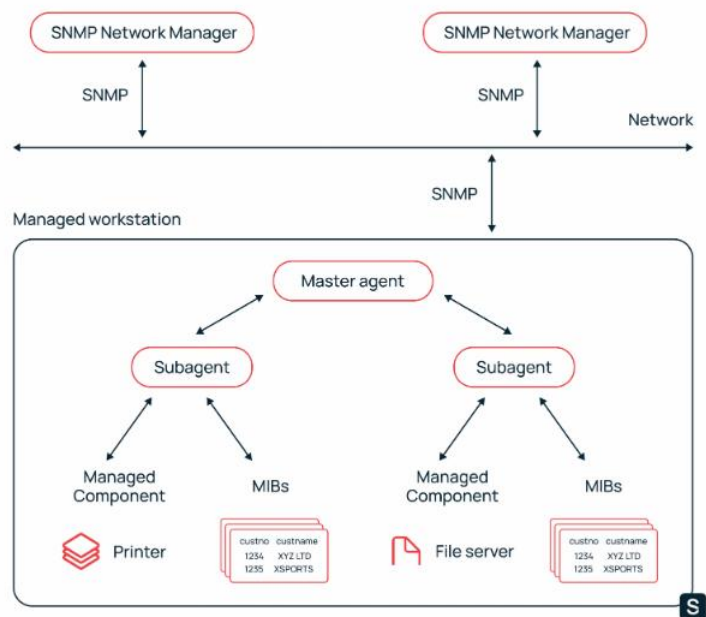


Рисунок 3 – Архитектура протокола SNMP

Данный протокол может быть интегрирован в сеть любого размера, чем больше сеть, тем больше можно получить возможности протокола. Протокол дает возможность смотреть, контролировать и управлять узлами единым интерфейсом, в котором используется пакет команд и автоматическое оповещение.

Диагностика системы в разы упрощается, при знании протокола SNMP, который предназначен для управления и наблюдения за любым устройством в сети. Так, протокол способствует исключению ввода команды администратором вручную. Администратор может автоматизировать сбор статистики с ключевых узлов, таких, как коммутатор, маршрутизатор, ПК, принтер и другие устройства, которые поддерживают данный протокол [17].

Обмен информацией по протоколу SNMP может осуществляться с помощью опроса или с помощью ловушек. SNMP— это стандартный протокол связи в системах мониторинга сети. SNMP основан на модели менеджера и агента. Менеджер – это центральный узел информации, где

собираются все виды данных, а затем передаются в разные удаленные места (агентам). Агентов может быть много, работа агента состоит в том, чтобы собирать различные фрагменты информации из их местоположения, они также могут сообщать информацию, которую они просто получают внутри себя [22]. Например, небольшие системные сигналы тревоги, и задача состоит в том, чтобы сообщить менеджеру, для просмотра всей информации в центральном информационном центре. Точный способ, которым менеджер SNMP и агенты общаются друг с другом, традиционно осуществляется через асинхронные сообщения-ловушки SNMP, но некоторые системы могут вместо этого использовать обмен сообщениями с опросом.

«SNMP агент— это часть программного обеспечения, которое поставляется вместе с сетевым устройством (маршрутизатором, коммутатором, сервером, Wi-Fi и т.д.). При включении и настройке выполняет всю тяжелую работу для менеджера, путем компиляции и хранения всех данных обо всех устройствах в базу данных (MIB). Эта база данных правильно структурирована, чтобы программное обеспечение менеджера могло легко опросить информацию и даже отправить информацию менеджеру, если произошла ошибка» [4].

## **1.6 База данных информации управления – MIB**

«MIB (Management Information Base)— это структурированный текстовый файл или несколько файлов, которые содержат информацию о всех объектах устройства, другими словами это иерархическая база данных информации управления, которая хранит в себе информацию о всех объектах устройства» [15].

Объектом может быть какая-нибудь настройка или параметры системы. У каждого объекта есть свой набор полей, таких как тип данных, доступность, статус, текстовое название настройки [24]. Также объект



может содержать другие объекты. Каждый тип устройства имеет свою таблицу, например, в принтере содержится информация о картриджах и их состоянии; коммутатор содержит в себе данные о проходящем трафике.

Каждый MIB имеет свой стандарт, определяемый различными RFC документами (документ из серии пронумерованных информационных документов Интернета), содержащих технические спецификации и стандарты, широко применяемые во всемирной сети. Название «Request for Comments» ещё можно перевести как «заявка (запрос) на отзывы» или «тема для обсуждения». В настоящее время первичной публикацией документов RFC занимается IETF под эгидой открытой организации общество интернет (англ. Internet Society, ISOC). Правами на RFC обладает именно общество интернет). Также существует большое количество MIB от самих производителей оборудования, которые дополняют стандартные данные и находятся на сайтах производителя, где их можно бесплатно использовать. Данные дополнения используются для описания особенных параметров модели устройства [5]. Также MIB можно составить самому, для личного успешного использования.

Файлы MIB – это набор вопросов, которые менеджер может задать агенту. Агент будет собирать эти данные локально и сохранять внутри себя, как определено в MIB. Диспетчер SNMP должен знать нужные ему стандарты и часто задаваемые вопросы для каждого типа агента. За счет MIB менеджер понимает, какую информацию он запрашивал у агента. Упрощенный пример работы MIB представлен на рисунке 4.

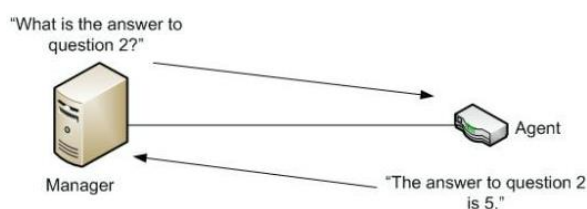


Рисунок 4 – Упрощенный пример работы MIB

«Каждый объект в MIB имеет свой уникальный цифровой адрес OID (Object IDentificator – числовой идентификатор объекта в дереве MIB) и имя Object Name (имя объекта, уникальная константа для всего MIB, однозначно соответствующая определённому OID.)» [15]. SNMP менеджер, используя OID, способен считывать или устанавливать значение объекта. Например, адрес объекта (OID) содержащего наименование системы: 1.3.6.1.2.1.1.5, а его имя (object name): sysName. Так как всё общение между SNMP агентом устройства и SNMP менеджером (системой наблюдения или администратором) происходит через OID, то понимать, что они описывают, надо знать [14]. Имя объекта играет ту же роль в SNMP, что и DNS (компьютерная распределённая система для получения информации о доменах) имя в IP сетях – более наглядное описательное представление набора чисел. В разных MIB оно может представлять разные OID, хотя, те что описаны в RFC, по идее должны быть уникальными для всех.

Простыми словами OID – это простой номер, составленный MIB, объектом интереса и экземпляром. Каждый идентификатор является уникальным для устройства, и при запросе будет предоставлена информация о том, что было запрошено OID. Скалярный и табличный, это типы OID. Скалярный – представляется одним объектом, как пример это может быть имя поставщика устройства, оно может быть только одно, именно так и будет выглядеть скалярный тип. Табличный – уже представляется несколькими результатами, как пример любой процессор он может показывать 4 различных значения использованного ЦП [8].

## **1.7 Выбор для использования базы данных**

Выбрать базу данных для решения задачи, из множества различных СУБД сложно, популярным решением в выборе базы данных служит СУБД от MySQL, но стоит рассмотреть разновидности баз данных их типы, и возможности для описанной задачи [16].

СУБД которые чаще всего используется для кэширования называются ключ-значение. Ключ-значение представлен на рисунке 5.

key:	value
user_id:	f5badc33-5bd7-4b65-a737-b5304675f476
color:	blue
repetitions:	3
text:	hello world
data:	{ ... }

Рисунок 5 – Ключ-значение, база данных

Реляционные базы данных применяются в СУБД для решения Online Transaction Processing (OLTP), стоит отметить основной признак для выбора данной базы – это повышенная нормализация данных, к дополнительным признакам можно отнести способность обработки большого количества транзакций. Данный вид СУБД представлен на рисунке 6.

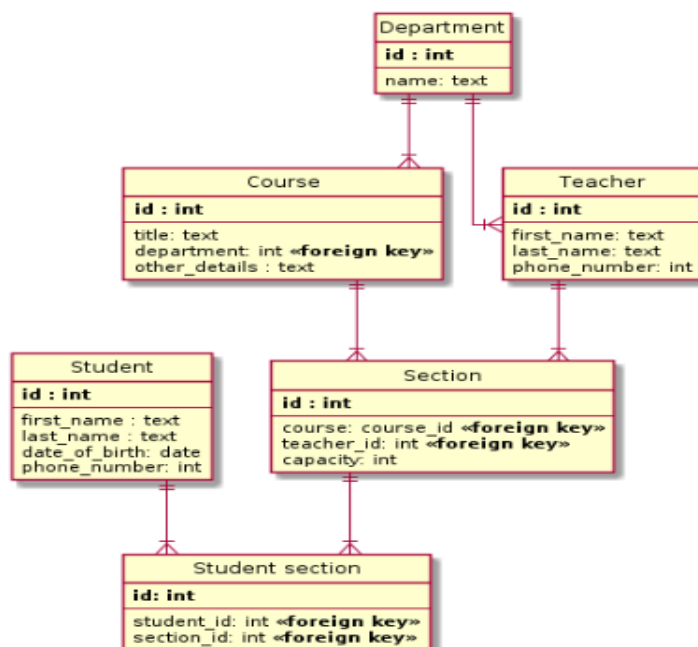


Рисунок 6 – Реляционная, база данных

СУБД, которые применяются для хранения в одном, но разной структурой называются документные. Данный вид СУБД представлен на рисунке 7.

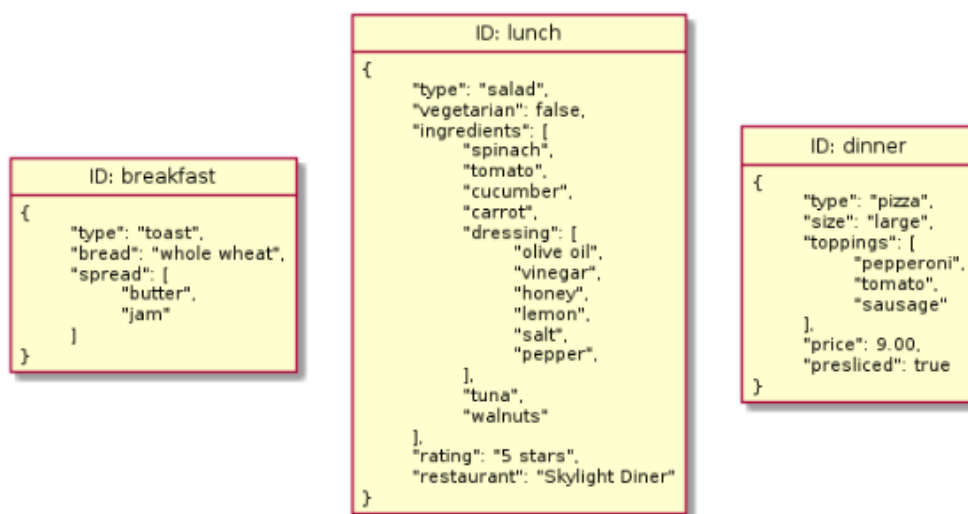


Рисунок 7 – Документная, база данных

Графовые СУБД очень сильно похожи на social networks, другими словами они реализуют оценочную систему [12]. Данный вид СУБД представлен на рисунке 8.

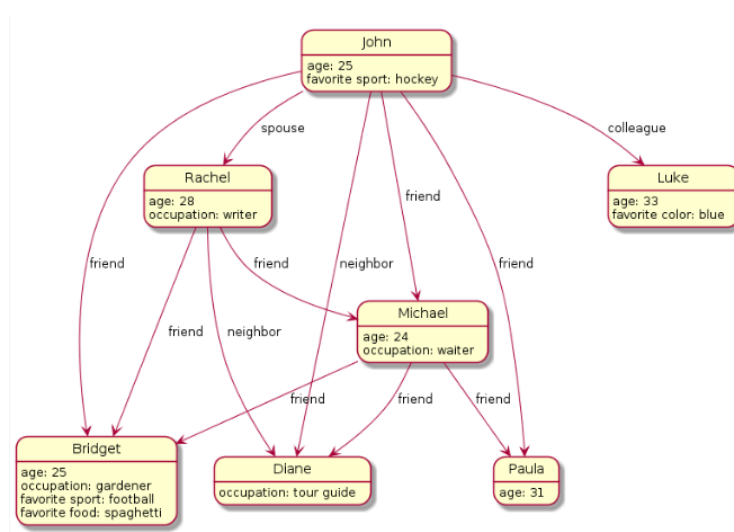


Рисунок 8 – Графовая, база данных

Для построения хранилищ, и создания выборки применяют СУБД колоночные. Основной признак для выбора данной базы служит огромное число строк превышавшие десятки [12]. Данный вид СУБД представлен на рисунке 9.

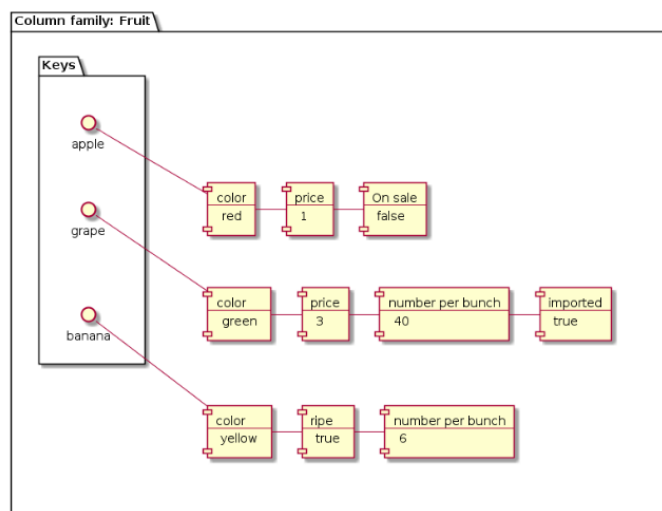


Рисунок 9 – Колонная, база данных

Стоит отметить, что часть крупных производителей создают несколько типов СУБД, это может быть отдельный отличный продукт, либо он используется во внутренней реализации. Как пример, можно рассмотреть крупного производителя СУБД Oracle, в его арсенале огромное количество СУБД, начиная реляционной СУБД, заканчивая продуктом Oracle NoSQL Database, которая может использовать и документной, и колоночной, и как ключ-значение [13].

Для решение поставленных задач, применение реляционной базы данных подходит больше, чем другой тип, к реляционным базам данных относятся MySQL, Oracle, Microsoft SQL Server, PostgreSQL. Система Zabbix дает возможность активно использовать две основные базы данных – это MySQL и PostgreSQL, стоит отметить, что для сравнения можно рассмотреть и другие интересные базы данных, но к сожалению другие базы

данных не поддерживаются в системе Zabbix. Из данного выбора, использование MySQL будет удобнее.

### **1.8 MySQL база данных, для решения дополнительной задачи**

Если для решения основной задачи база данных не используется, так как все необходимое, а также база данных уже интегрирована в систему Zabbix, то для решения дополнительной задачи, нам необходима локальная база данных [20]. Это связано с тем, что сама инсталляция Zabbix располагается на нескольких физических и виртуальных серверах. Также разделяется на следующие направления:

- на первой находится сам сервер Zabbix;
- на втором сервере или виртуальной машине сама база данных.

Стоит отметить, что используется реплицирующая дополнительная база данных по типу Master-Slave и все запросы для аналитики осуществляются только с использованием Slave, это означает, что production или master-db не используется для select запросов, таким образом идет снижение нагрузки на предоставление запросов, делается работа быстрее, а также снижаются различные очереди [21].

MySQL, основное преимущество – это бесплатное использование и скачивание. Данная система способна эффективно функционировать и взаимодействовать с сайтами интернета и веб-приложениями. При этом она проста в освоении, что лишь увеличивает ее популярность. Эта база данных является наиболее приспособленной для применения в среде web – СУБД (системой управления базами данных). Не секрет, что для исполнения приложений клиента на большинстве хостинг-площадок провайдеры предоставляют небольшое количество ресурсов (как вычислительных, так и дисковых). Поэтому для данного применения необходима высокоэффективная СУБД, обладающая при этом высокой надежностью (большинство web-приложений и сайтов должны работать в режиме 24/7).

Большая часть популярности база данных приобрела за счет множества руководств в интернете, а также наличием систем, которые в корне упрощают использование данной базы, это плагины и расширения.

Из основных плюсов данной базы данных стоит отметить:

- масштабируемость, которая позволяет работать и с маленькими, и с большими объемами данных;
- разнообразие доступных инструментов, обеспечивает обширным функционалом;
- способность обслуживания нескольких запросов (многопоточность);
- наличие удобного интерфейса для любого пользователя;
- встроенная система безопасности.

Сферы применения MySQL:

- при распределённых операциях, когда функционала SQLite (другая популярная система) не хватает;
- когда требуется обеспечить высокий уровень безопасности, что MySQL делает с успехом;
- для работы с интернет-страницами и веб-приложениями, поскольку MySQL является наиболее удобной СУБД для этой сферы применения;
- при работе со специфическим проектом, где функционал MySQL дает оптимальный результат.

Для удобства решения необходимой задачи можно использовать MySQL Workbench – это инструмент для визуального проектирования базы данных, включающий в себя моделирование, проектирование, а также способный использовать базу данных, как единое бесшовное окружение баз данных MySQL. Удобный дизайн для администратора, который позволяет визуально проектировать, моделировать, а также генерировать и управлять базами данных. Сложные модели ER могут занимать очень много времени,

но инструмент MySQL Workbench позволяет упростить возможность создания прямого или обратного проектирования и т.п.

Визуальные инструменты дают возможность создавать, выполнять и оптимизировать SQL-запросы. Интегрированный редактор SQL имеет в себе авто заполнение, выделение синтаксиса, а также имеет в себе буфер обмена, который записывает историю выполнения действий. Браузер объектов обеспечивает мгновенный доступ к схеме и объектам базы данных.

«MySQL Workbench – инструмент для визуального проектирования баз данных, интегрирующий проектирование, моделирование, создание и эксплуатацию БД в единое бесшовное окружение для системы баз данных MySQL» [13].

По сути MySQL Workbench это набор инструментов, в котором все важные показатели необходимые администратору возможно рассмотреть в панели мониторинга, так у администратора в одном месте собраны все необходимые данные для работы в базе данных.

Вывод по первой главе.

Были поставлены задачи на разработку системы учета. Рассмотрены системы Zabbix и SCADA для решения поставленных задач, в результате выбрана системы Zabbix. Изучен протокол SNMP и база данных информации управления MIB, также подобрана база данных MySQL Workbench.



## **Глава 2 Разработка системы учета и мониторинга оборудования в системе Zabbix**

### **2.1 Создание мониторинга коммутаторов D-Link**

Как уже было написано в первой главе, нужно создать систему в сети для сбора и мониторинга данных оборудования, основное оборудование – это коммутаторы D-Link, принтера KYOCERA, а также операционные системы Windows и Linux.

Прежде чем приступить к созданию системы коммутаторов D-Link, следует понять разницу между управляемыми и неуправляемыми коммутаторами.

Управляемые коммутаторы могут предоставить все функции необходимые для сети. Это происходит за счет разнообразия функций (VLAN– виртуальная локальная сеть, предназначенная для логического разделения устройств разных назначений, т.е. разделяем серверное оборудование от рабочих станций, телефонов и IP камер, к примеру, CLI-Command Line Interface, SNMP и т.д.) [25]. Существует два типа управляемых коммутаторов:

- это может быть управляемые коммутаторы используемые на уровне ядра в сети, они используются в крупных и сложных центрах обработки данных;
- а также есть «легко управляемые» коммутаторы, которые называются интеллектуальными коммутаторами. Они созданы для удовлетворения рынка с меньшими запросами, то есть не нуждаются во всех функциях и имеют меньший бюджет. Эти коммутаторы созданы с частью от возможностей управляемого коммутатора, за меньшую стоимость, что дает альтернативное оптимальное использование необходимое организации.

Неуправляемые коммутаторы – это уже более простые в техническом плане коммутаторы, которые работают по системе подключи и используй. После подключения коммутатора пользователю необходимо подождать, после чего без необходимости какой-либо настройки, пользователь может начать работу с коммутатором. Использование неуправляемого коммутатора, как простого настольного, для решения необходимости его использования в малом количестве портов внутри дома или в малых конференц-зала.

Использования в системе Zabbix можно только управляемые коммутаторы, так как неуправляемые коммутаторы не имеют в себе SNMP и получить из них какую-либо информацию, нельзя.

## **2.2 Подключение D-Link и KYOCERA в Zabbix**

Чтобы подключить коммутатор, принтер, и т.п. в систему Zabbix существует несколько вариантов, самые популярные это описание MIB отдельной модели устройства, или использование уже готового шаблона внутри Zabbix.

Описание MIB достаточно долгий процесс, так как необходимо получить информацию от производителя касаясь устройства занести ее в Zabbix. Произвести анализ значений MIB с использованием ПО, например, у D-Link есть своя D-View, занести значения надо в Zabbix Templates по Object Name. На рисунке 10 представлено описание MIB.

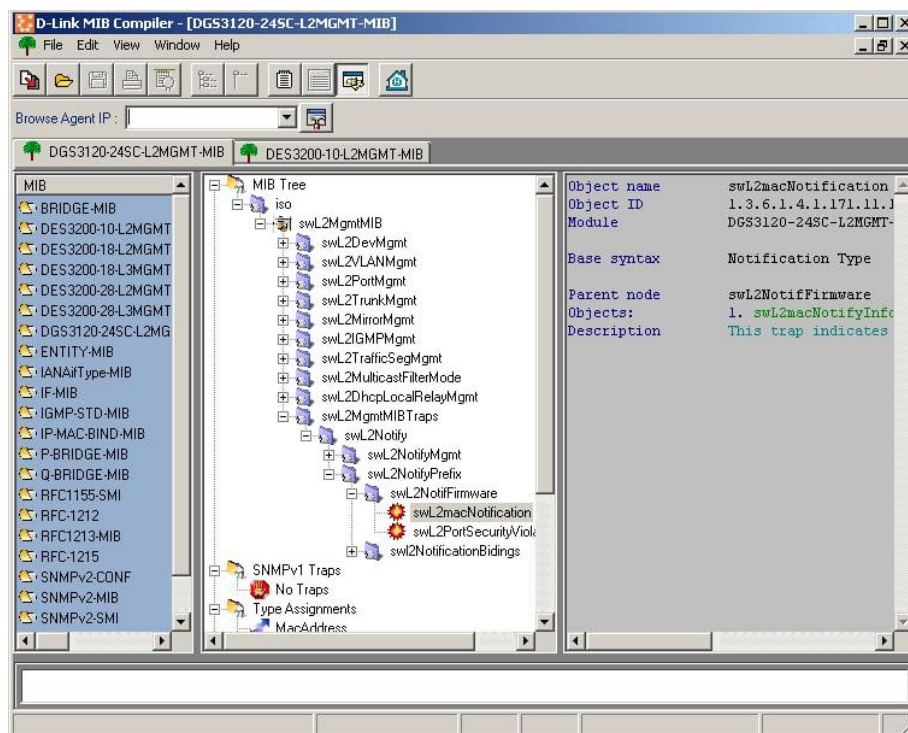


Рисунок 10 – Описание MIB

Все становится проще, при использовании шаблона, которых в репозиториях сообщества Zabbix уже достаточно много и использование MIB описание имеет смысл для каких-то редких моделей редкого применения.

Для использования шаблона и подключению D-Link заходим по IP в D-Link, на которой представлена web-версия интерфейса, управляемого коммутатора. Она представлена на рисунке 11.

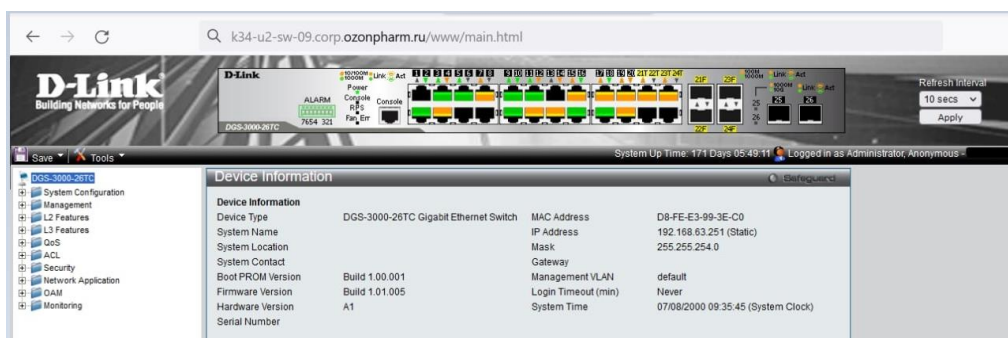


Рисунок 11 – Web-версия интерфейса

Здесь необходимо подключить параметры SNMP протокола для его дальнейшего использования в системе Zabbix, а также нужно задавать SNMP community, что будет служить идентификатором. Это выполняется для подключения SNMP протокола в Zabbix. Включение SNMP протокола представлено на рисунке 12.

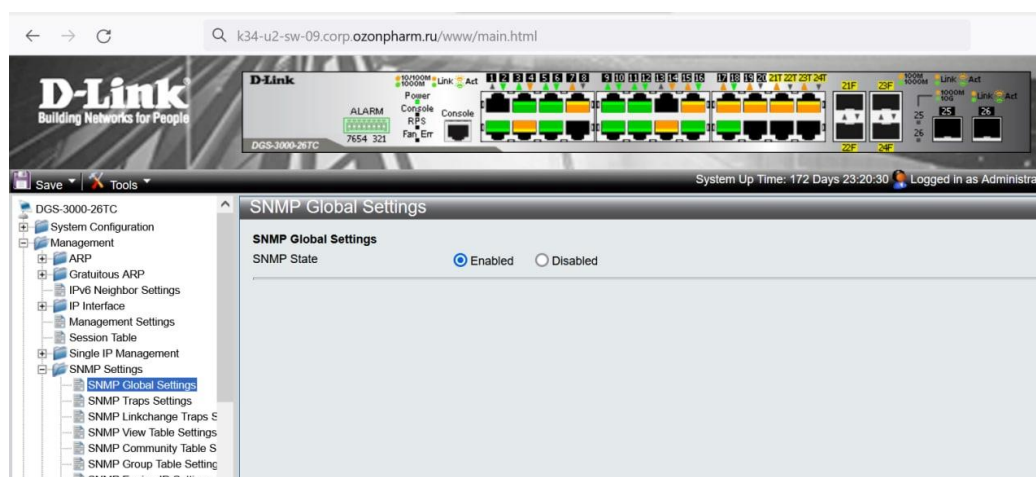


Рисунок 12 – Включение SNMP протокола

Переходим с необходимыми данными в систему Zabbix и в разделе host создаем готовый host. Правильно будет сопоставить DNS имена, вместо IP-адреса. В Интернете бесчисленное количество физических устройств (серверов, компьютеров, планшетов и т.п.), связанных между собой в сеть. Любой сайт в интернете по факту находится на физическом устройстве. Каждое устройство имеет свой уникальный номер – IP-адрес вида 123.123.123.123.

Чтобы попасть на сайт, нужно знать IP-адрес устройства, на котором расположен этот сайт. Поэтому для удобства работы в Интернете в 80-х годах была создана система доменных имен – DNS (Domain Name System). Смысл её в том, что каждому цифровому IP-адресу присваивается понятное буквенное имя (домен). Например, IP-адресу сервера 194.58.116.30 соответствует домен ozonfarm. Когда вводится в браузере доменное имя,

сервера DNS автоматически преобразуют его в IP-адрес. Домен за доли секунды переводится в IP-адрес DNS-системой, и можно осуществить переход на нужный сайт. На рисунке 13 представлен созданный host.

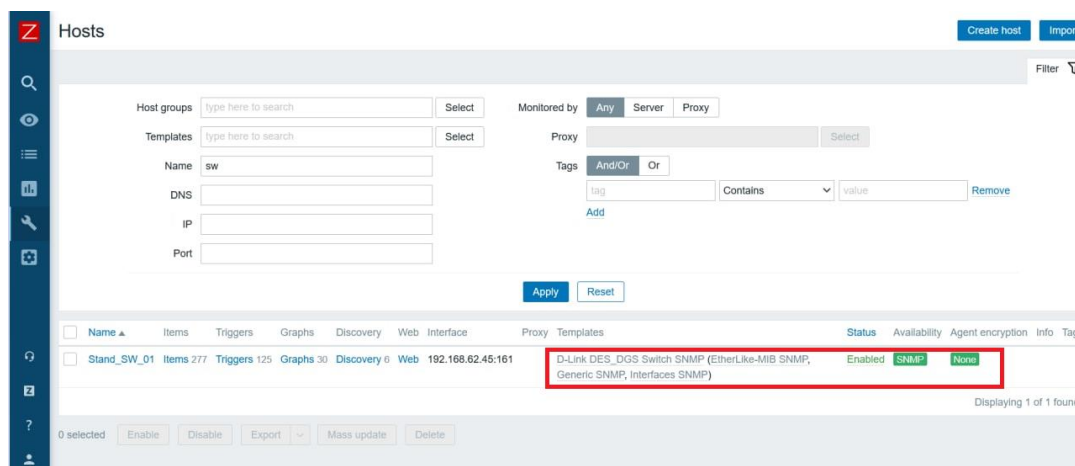


Рисунок 13 – Созданный host

Выбираем заранее заготовленный шаблон. Также на рисунке 12 и рисунке 13, надпись SNMP выделенная зеленым, что означает контроль доступности SNMP, если выделение красным цветом, это означало бы отсутствие подключения доступности SNMP. Добавление заготовленного шаблона представлено на рисунке 14.

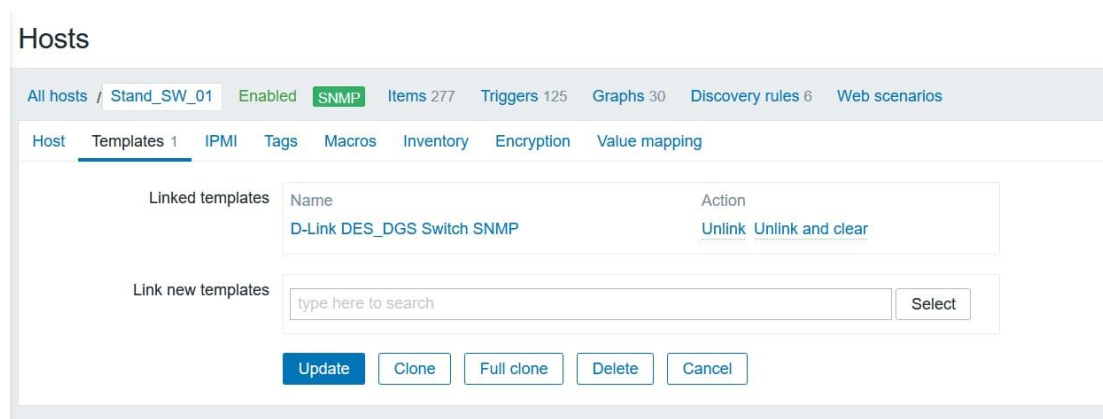


Рисунок 14 – Добавление заготовленного шаблона

После успешного добавления хоста первое время значение позиций, находящихся на мониторинге может быть равным нескольким единицам, содержащих только базовую информацию, свойственную всем видам оборудования. К примеру – контроль ICMP доступности, потери пакетов до удаленного хоста, но через какое-то время, спустя время от старта процесса автообнаружения (discovery) в список items будет расширен на автообнаруженные позиции, согласно правилу обнаружения [18]. После получения интерфейсов, доступных для мониторинга для части из них будут сопоставлены triggers то есть условия корректной работы, определяющие нормальное состояние для того или иного интерфейса. Первоначальное обнаружение интерфейсов и сопоставление триггеров представлено на рисунке 15.

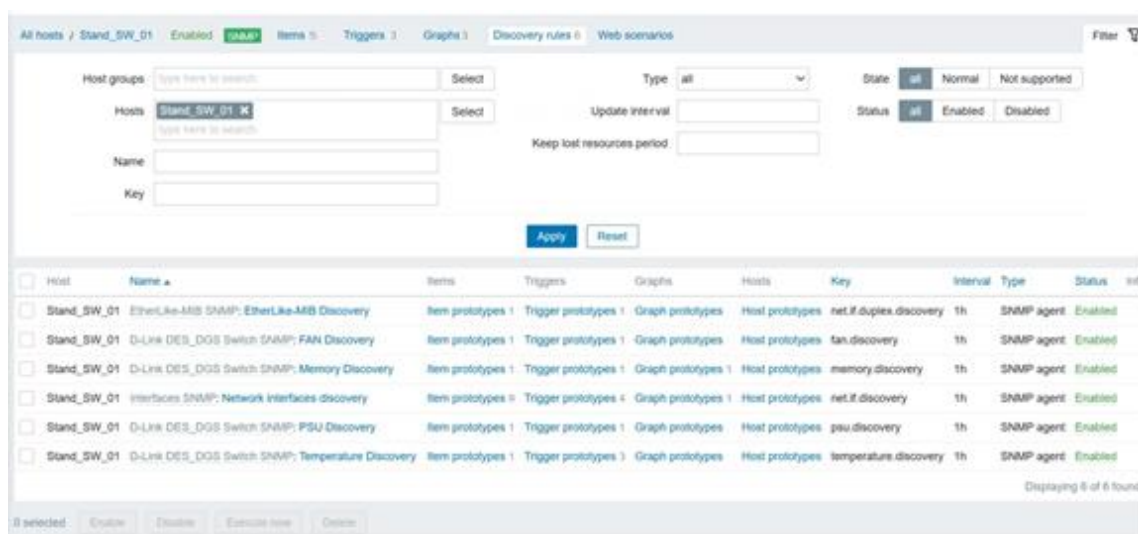


Рисунок 15 – Первоначальное обнаружение интерфейсов и сопоставление триггеров

Проделанные действия создают необходимые нам данные для мониторинга, остается их удобно и красиво оформить, для описания информации, которую нам передает коммутатор D-Link. Чтобы создать удобную для нас информацию надо прописать необходимые данные для коммутатора. Мониторинг D-Link представлен на рисунке 16.

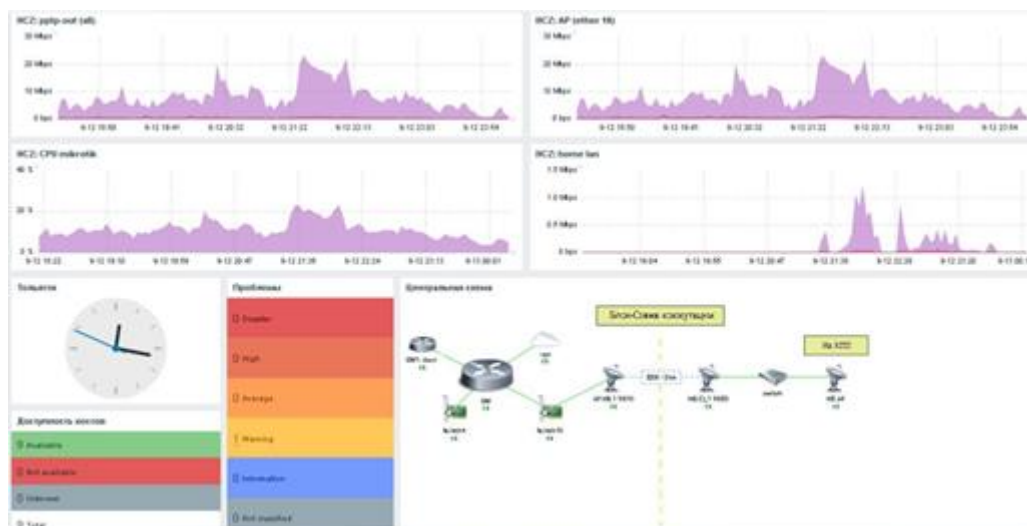


Рисунок 16 – Мониторинг D-Link

На графиках изображен трафик, проходящий через определенный интерфейс.

Внизу часов на рисунке 16, показаны данные, сколько всего хостов в данной хост группы (объединения устройств). Справа проблемы, которых может быть много, например, падение интерфейса; повышенный ping; потеря пакета; недоступность, контрольных узлов и т.п. Эти данные основаны на триггерах. Используемые в триггерах выражения являются очень гибкими. Их можно использовать для создания сложных логических тестов, учитывая статистику по мониторингу.

Простое полезное выражение может выглядеть примерно так:

{<сервер>:<ключ>.<функция>(<параметр>)}<оператор><константа>

Функции триггеров позволяют ссылаться на собранные значения, текущее время и другие факторы. На рисунке 17 представлена блок-схема коммутации.



Рисунок 17 – Блок-схема коммутации

Она представляет схему ядра коммутации, центральный маршрутизатор, на который приходит вышестоящий интернет провайдер, поверх которого поднимается VPN тоннель до головного офиса. В ядре коммутации выходит локальный порт, а также один уходит на радио мост дальнего КПП на посту охраны. К этому посту идет радиоканал с двух сторон передающий точки доступа, на посту расположен коммутатор и точка доступа.

Таким же подобным способом создаем систему необходимую по принтерам KYOCERA, уже не выводя его данные в удобный функционал, так как от принтеров активный мониторинг не требуется, есть лишь необходимость контроля количества картриджей в принтере, а эти данные агент передает удобно.

### 2.3 Подключение операционной системы в Zabbix

Рассматривать подключение будем на примере Linux. Так как, в целом основное подключение на Windows и другую ОС аналогично.

Обновляем репозиторий на Linux до последней версии либо необходимые нам Zabbix – agent, после установки агента необходимо будет



провести его конфигурацию, указав адрес сервера, откуда будет поступать запрос. Аналогично действиям по коммутатору.

Агент может быть двух типов активный или пассивный, указываем понятное имя host name для сервера. Переходим в Zabbix Hosts, подключаем агента и задаем нужное нам имя IP. Листинг конфигурационного файла Zabbix агента представлен на рисунке 18.

```
144 ##### Active checks related
145
146 ### Option: ServerActive
147 # List of comma delimited IP:port (or DNS name:port) pairs of Zabbix
148 # If port is not specified, default port is used.
149 # IPv6 addresses must be enclosed in square brackets if port for that
150 # If port is not specified, square brackets for IPv6 addresses are op
151 # If this parameter is not specified, active checks are disabled.
152 # Example: ServerActive=127.0.0.1:20051,zabbix.domain,[:,1]:30051,::1
153 #
154 # Mandatory: no
155 # Default:
156 # ServerActive=
157
158 ServerActive=zabbix.corp.ozonpharm.ru
159
160 ### Option: Hostname
161 # List of comma delimited unique, case sensitive hostnames.
162 # Required for active checks and must match hostnames as configured o
163 # Value is acquired from HostnameItem if undefined.
164 #
165 # Mandatory: no
166 # Default:
167 # Hostname=
168
169 Hostname=SRV_PC_Info
170
```

Рисунок 18 – Листинг конфигурационного файла Zabbix агента

Zabbix агент не активен, меняем IP, чтобы посмотреть подключение, можно подключиться к данному серверу через WinSCP, чтобы посмотреть лог этого сервера, здесь видно, что в файле идет ошибка. Листинг конфигурационного файла Zabbix агента представлен на рисунке 19.

```

1 15031:20220508:201245.564 Starting Zabbix Agent [Zabbix server]. Zabbix 5.4.12 (revision 9bd5a418b8).
2 15031:20220508:201245.564 **** Enabled features ****
3 15031:20220508:201245.564 IPv6 support:      YES
4 15031:20220508:201245.564 TLS support:      YES
5 15031:20220508:201245.564 *****
6 15031:20220508:201245.564 using configuration file: /etc/zabbix/zabbix_agentd.conf
7 15031:20220508:201245.564 agent #0 started [main process]
8 15036:20220508:201245.568 agent #5 started [active checks #1]
9 15034:20220508:201245.569 agent #3 started [listener #2]
10 15036:20220508:201245.571 active check configuration update from [127.0.0.1:10051] started to fail (cannot connect to [[127.0.0.1]:10051]:
11 15032:20220508:201245.571 agent #1 started [collector]
12 15035:20220508:201245.571 agent #4 started [listener #3]
13 15033:20220508:201245.572 agent #2 started [listener #1]
14 15031:20220508:201516.615 Got signal [signal:15 (SIGTERM),sender_pid:15091,sender_uid:107,reason:0]. Exiting ...

```

## Рисунок 19 – Листинг конфигурационного файла Zabbix агента

Запускаем в ручном режиме процесс авто-обнаружений позиций, находящихся на мониторинге, согласно описанного в шаблоне правила автообнаружения (discovery). Этот процесс мог начаться автоматически, либо в течение времени, на автоматический запуск, указанный в шаблоне к данному узлу, а также начаться ранее указанного времени в случае отсутствия очереди на опрос в Zabbix сервере. При первом добавление хоста на мониторинг к нему автоматически добавятся свойственные данному объекту мониторинга относящиеся к базовым понятиям. К примеру, добавиться шаблон ping запроса, сопоставятся триггеры отсутствия на данный запрос ответа, с повышенным временем ответа. Процесс успешного автообнаружения можно отследить путем наглядного увеличения поля items. Схема постройки. Изменяется карта, заготовка в виде текста, выбираем что можем заменить, создается запрос для вывода на карту.

На выходе отображаются данные, которые приходят в систему мониторинга, данные о состоянии операционной системы. Данные мониторинга представлены на рисунке 20.

Interface ens192: Interface type	05/10/2022 04:09:1...	Ethernet (1)
Interface ens192: Operational status	05/10/2022 09:42:1...	up (1)
Interface ens192: Outbound packets discarded	05/10/2022 09:42:0...	0
Interface ens192: Outbound packets with errors	05/10/2022 09:42:0...	0
Interface ens224: Bits received	05/10/2022 09:42:0...	11.34 Kbps -10.46 Kbps
Interface ens224: Bits sent	05/10/2022 09:42:0...	13.08 Kbps -3.6 Kbps
Interface ens224: Inbound packets discarded	05/10/2022 09:41:5...	0
Interface ens224: Inbound packets with errors	05/10/2022 09:41:5...	0
Interface ens224: Interface type	05/10/2022 04:09:1...	Ethernet (1)
Interface ens224: Operational status	05/10/2022 09:42:1...	up (1)
Interface ens224: Outbound packets discarded	05/10/2022 09:42:0...	0
Interface ens224: Outbound packets with errors	05/10/2022 09:42:0...	0
Interface tun0: Bits sent	05/10/2022 09:42:0...	12.22 Kbps -1.82 Kbps
Maximum number of open file descriptors	05/10/2022 04:07:4...	9.2234E+18
Version of Zabbix agent running	05/10/2022 04:07:4...	5.4.12
Zabbix agent availability	05/10/2022 09:42:2...	available (1)
Zabbix agent ping	05/10/2022 09:42:4...	Up (1)

Рисунок 20 – Данные мониторинга.

Стоит отметить, что одна и та же операционная система, либо виртуальная машина, имеющая схожую конфигурацию, могут иметь разное количество items находящихся на мониторинге, что обусловлено разной физической особенностью и количеством запущенных процессов на данных машинах или хостах.

Вывод по второй главе.

По изученным существующие методы реализации системы мониторинга учета, была выбрана система Zabbix в которой создавалась система учета и мониторинга оборудования. В систему мониторинга были добавлены коммутаторы D-Link, принтера KYOCERA, а также операционные системы Linux.

## Глава 3 Создание MVP - проекта по графическому отображению состояния материальных запасов

«Minimal Viable Product (минимально жизнеспособный продукт) – тестовая версия товара, услуги или сервиса с минимальным набором функций (иногда даже одной), которая несет ценность для конечного потребителя»[5].

MVP создают для тестирования гипотез и проверки жизнеспособности задуманного продукта, насколько он будет ценным и востребованным на рынке.

Результаты тестирования минимально жизнеспособного продукта и обратная связь от целевой аудитории помогают понять, стоит ли развивать проект дальше, какие изменения следует внести в стратегию, а что оставить в первоначальном виде.

В данной задаче использование базы данных идет локально, для этого устанавливаем MySQL, и создаем собственную базу данных, разработанный код находится в приложении А. Создание БД в MySQL Workbench представлено на рисунке 21.

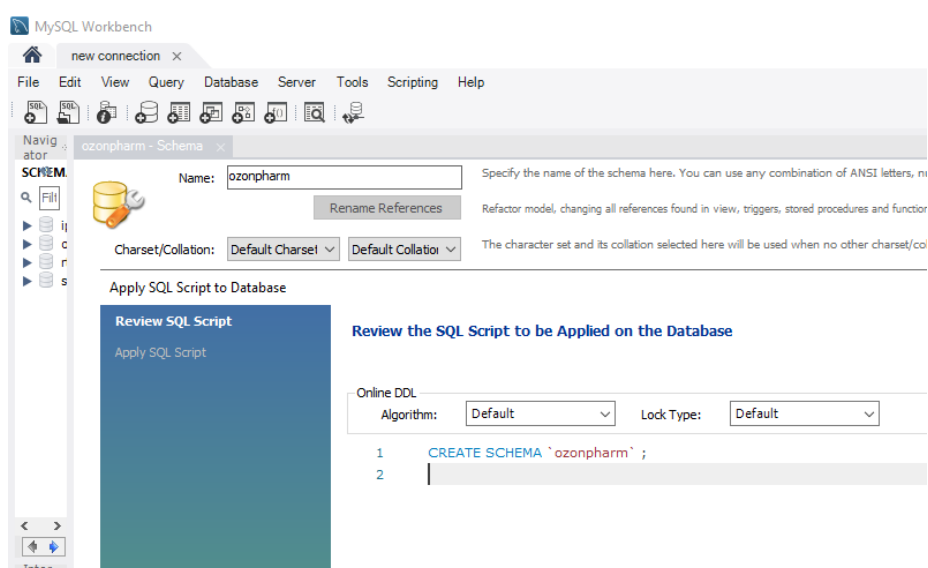


Рисунок 21 – Создание БД в MySQL Workbench

Наша задача немного не стандартна для Zabbix, но ее интегрирование не проблема, существует система управления производством, которая в свою очередь имеет интерфейсы управления, с помощью API интерфейса добываем информацию в систему мониторинга. Поэтому для удобства схемы применяем свою картинку резервуара. Доступные изображения представлены на рисунке 22.



Рисунок 22 – Доступные изображения

Создаем Zabbix-agents.

Var1, Var2 – это скрипты внешних проверок для MySQL, которые осуществляют соединение с базой данных разработанный код находится в приложении А.

Активируем агентов, при нормальном запуске, они будут активированы (Рисунок 23).

```
root@pcinfo:~# systemctl restart zabbix-agent.service
root@pcinfo:~# systemctl status zabbix-agent.service
● zabbix-agent.service - Zabbix Agent
   Loaded: loaded (/lib/systemd/system/zabbix-agent.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2022-05-01 11:46:49 MSK; 2s ago
     Process: 15172 ExecStart=/usr/sbin/zabbix_agentd -c $CONFFILE (code=exited, status=0/SUCCESS)
    Main PID: 15174 (zabbix_agentd)
      Tasks: 6 (limit: 4701)
     Memory: 4.0M
    CGroup: /system.slice/zabbix-agent.service
           └─15174 /usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agentd.conf
             └─15175 /usr/sbin/zabbix_agentd: collector [idle 1 sec]
               └─15176 /usr/sbin/zabbix_agentd: listener #1 [waiting for connection]
                 └─15177 /usr/sbin/zabbix_agentd: listener #2 [waiting for connection]
                   └─15178 /usr/sbin/zabbix_agentd: listener #3 [waiting for connection]
                     └─15179 /usr/sbin/zabbix_agentd: active checks #1 [idle 1 sec]

May 01 11:46:49 pcinfo systemd[1]: Starting Zabbix Agent...
May 01 11:46:49 pcinfo systemd[1]: zabbix-agent.service: Can't open PID file /run/zabbix/zabbix_agentd.pid:
May 01 11:46:49 pcinfo systemd[1]: Started Zabbix Agent.
lines 1-18/18 (END)
```

Рисунок 23 – Успешный запуск

Осуществляется установка Zabbix агента на виртуальную Linux машину, для выполнения роли связующего звена между внешней системы управления производством и системы мониторинга.

При отсутствии запущенного агента на удаленном сервере идет ошибка (Рисунок 24).

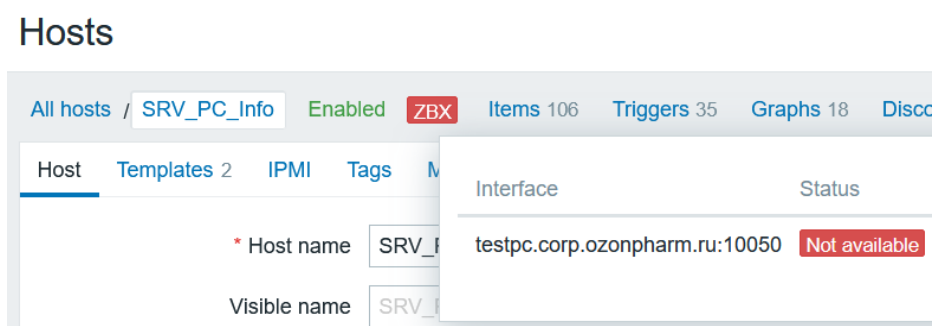


Рисунок 24 – Ошибка удаленного сервера

Создаем необходимый item, для импорта данных (Рисунок 25).

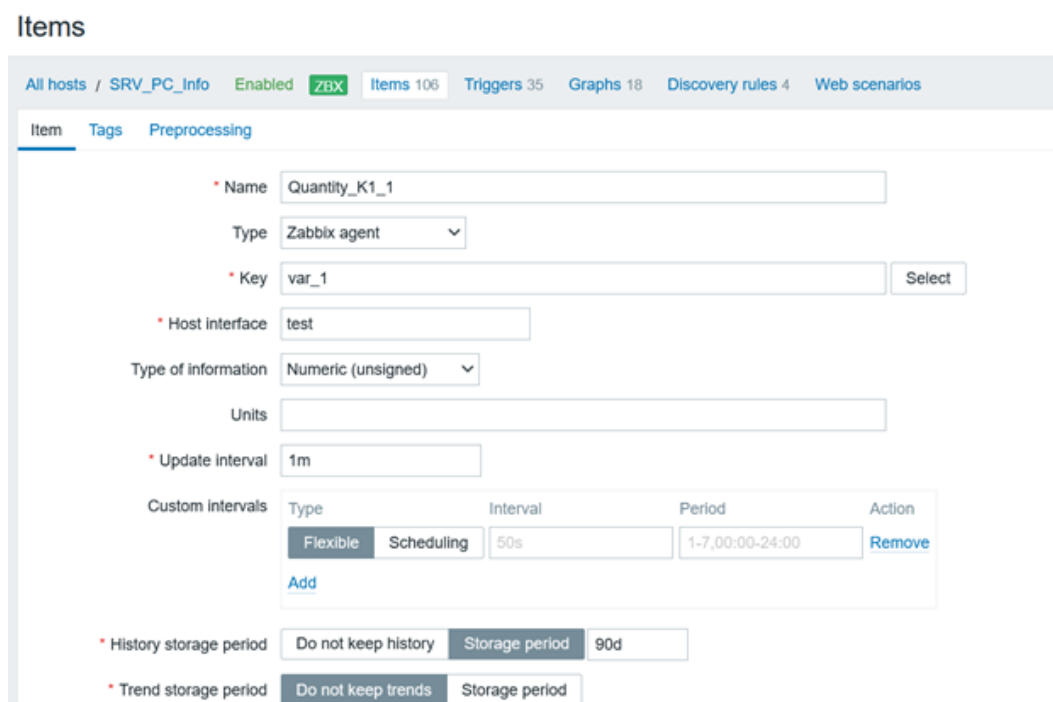


Рисунок 25 – Item

Одинаковые значения просто копируем, изменяя их свойства, и создаем запрос на импорт значения.

Проводим проверку корректности прохождения данных (Рисунок 26 – 27).

The screenshot shows the 'Latest data' section of a monitoring tool. It includes search filters for 'Host groups', 'Hosts' (with 'SRV\_PC\_Info' selected), and 'Name' (with 'Quantity\_K1' entered). There are also 'Tags' and 'Show details' options. Below the filters is a table with the following data:

<input type="checkbox"/>	Host	Name ▲	Last check	Last value	Change
<input type="checkbox"/>	SRV_PC_Info	Quantity_K1_1 ?	05/01/2022 11:59:4...	643	
<input type="checkbox"/>	SRV_PC_Info	Quantity_K1_2 ?	05/01/2022 11:59:4...	643	

Рисунок 26 – Проверка

This screenshot shows the same monitoring interface as Figure 26, but with a sidebar menu on the left. The 'Monitoring' menu item is highlighted with a red box, and the 'Latest data' sub-item is also highlighted with a red box. The search filters and data table are identical to the previous figure.

Рисунок 27 – Мониторинг данных

Создаем триггеры, которые необходимы нам для создания данных, которые будут предупреждать и обозначать объем содержимого. Их создание представлено на рисунке 28.

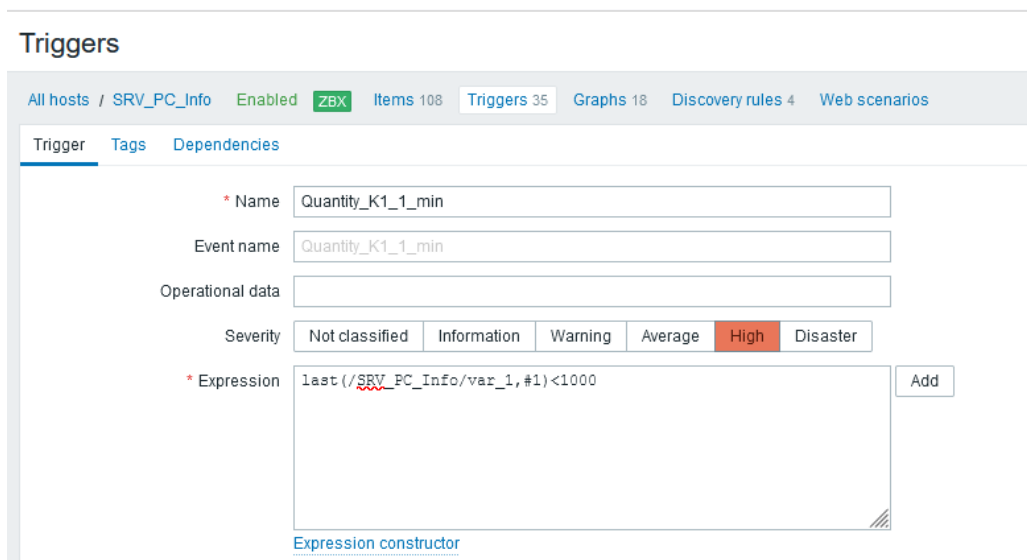


Рисунок 28 – Триггер на min

Добавляем триггеры на схему, чтобы alert выводился прямо на карту (Рисунок 29).

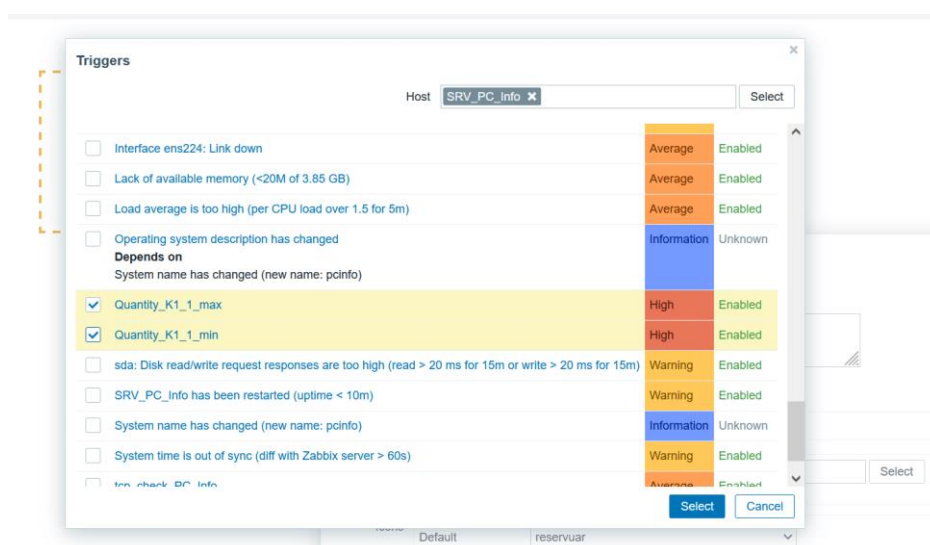


Рисунок 29 – Alert



Готовый результат созданного MVP-проекта представлен на рисунке 30.

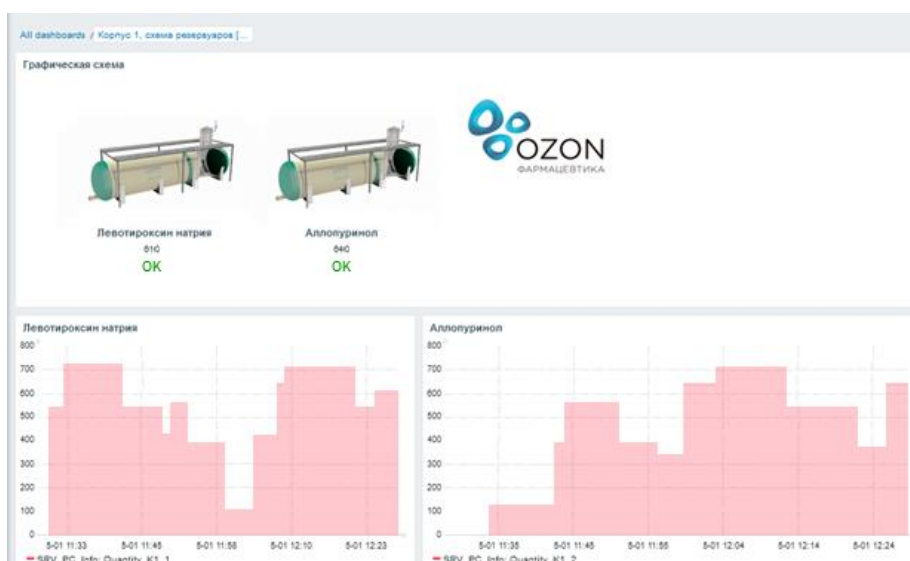


Рисунок 30 – MVP-проект

В конечном виде получен график отслеживания динамики жидкости в резервуаре, а также состояние и объем.

Вывод по третьей главе.

Дополнительной задачей было поставлено создать MVP-проект, для возможности мониторинга расходных материалов в системе. Для решения задачи использовалась система Zabbix. Была создана база данных в MySQL Workbench и интегрирована в систему Zabbix. В самой системе с помощью интерфейса управления и с помощью API интерфейса получилось собрать информацию в систему мониторинга.

## Заключение

Для фармацевтической компании ООО «Озон» требовалось создать информационную систему по учету оборудования вне локального носителя, как решение на производстве, для дистанционного контроля необходимого оборудования.

В работе разработана система для сбора и мониторинга данных оборудования в сети, создан MVP-проект, позволяющий отслеживать расходный материал.

В ходе выполнения данной бакалаврской работы были изучены существующие методы реализации системы мониторинга учета, спроектирована база данных ресурсов для внедрения в систему. В бакалаврской работе на основе проведенного анализа двух систем SCADA и Zabbix, была выбрана система Zabbix которая больше подходит под поставленные задачи, а также поддерживает необходимые протоколы.

Были изучены возможности Zabbix, такие как Zabbix агенты, описание MIB и т.п., для упрощенного способа передачи необходимых данных, изучены возможности протоколов, и применение SNMP протокола в результате разработана система для сбора и мониторинга данных оборудования в сети.

Был создан MVP-проект на основе поставленной дополнительной задачи, который расширил возможности мониторинга данных в системе и позволил отслеживать расходный материал. При создании MVP-проекта в систему Zabbix была подключена база данных MySQL Workbench. С помощью системы управления производством в системе Zabbix, которая в свою очередь имеет интерфейсы управления, а также с помощью API интерфейса была получена информация в систему мониторинга.

## Список используемой литературы

1. Бережной, А.Н. Сохранение данных: теория и практика / А.Н. Бережной. - Москва : ДМК Пресс, 2016. - 317 с. - ISBN 978-5-97060-185-3. - Текст : электронный.
2. Варфоломеева, А. О. Информационные системы предприятия : учебное пособие / А.О. Варфоломеева, А.В. Коряковский, В.П. Романов. — 2-е изд., перераб. и доп. - Москва : ИНФРА-М, 2022. - 330 с. - (Высшее образование: Бакалавриат). — DOI 10.12737/21505. - ISBN 978-5-16-012274-8. - Текст : электронный.
3. Вдовин В. М. Теория систем и системный анализ : учебник для бакалавров / В. М. Вдовин, Л. Е. Суркова, В. А. Валентинов. — 5-е изд., стер. — Москва : Издательско-торговая корпорация «Дашков и К », 2020. - 642 с. - ISBN 978-5-394-03716-0.
4. Далле Вакке, А. Zabbix. Практическое руководство / Андреа Далле Вакке ; пер. с англ. А.Н. Киселева. - Москва : ДМК Пресс, 2017. - 356 с. - ISBN 978-5-97060-462-5. - Текст : электронный.
5. Зиндер, Е.З. Технологии системного проектирования. Учебное пособие. – М.: Центр Информационных Технологий, 2018
6. Кангин, В.В. Разработка SCADA-систем : учеб. пособие / В.В. Кангин, М.В. Кангин, Д.Н. Ямолдинов. - Москва ; Вологда : Инфра-Инженерия, 2019. - 564 с. - ISBN 978-5-9729-0319-1. - Текст : электронный.
7. Капулин, Д. В. Информационная структура предприятия/Капулин Д.В., Кузнецова А.С., Носкова Е.Е. - Краснояр.: СФУ, 2014. - 186 с.: ISBN 978-5-7638-3128-3. - Текст : электронный.
8. Карминский А. М. Применение информационных систем: учебное пособие / А. М. Карминский, Б. В. Черников. - 2-е изд., перераб. и доп. - Москва : ФОРУМ : ИНФРА-М, 2019. - 320 с. - (Высшее образование). - ISBN 978-5-8199-0495-4.

9. Колисниченко, Д. Н. Беспроводная сеть дома и в офисе : практическое руководство / Д. Н. Колисниченко. - Санкт-Петербург : БХВ-Петербург, 2009. - 480 с. - (Самоучитель). - ISBN 978-5-9775-0427-0.
10. Косарев, В. А. Локальные вычислительные сети : учебное пособие / В. А. Косарев, А. А. Игнаткин. - Москва : ИД МИСиС, 2018. - 149 с. - Текст : электронный.
11. Маркарян, Л. В. Компьютерные технологии управления с применением SCADA-системы TRACE MODE 6 : лабораторный практикум / Л. В. Маркарян. - Москва : Изд. Дом НИТУ «МИСиС», 2018. - 104 с. - Текст : электронный.
12. Марки, А. В. СУБД «Ред База Данных». Основы SQL : учебное пособие / А. В. Маркин. - Москва : Ай Пи Ар Медиа, 2022. - 460 с. - ISBN 978-5-4497-1605-7.
13. Прокушев, Я. Е. Базы данных : учебник с практикумом / Я. Е. Прокушев – 2-е изд. – Санкт-Петербург : Интермедия, 2022. - 264 с.
14. Турецкий В. Я. Математика и информатика : учебник / В. Я. Турецкий. - 3-е изд., испр. и доп. - Москва : ИНФРА-М, 2020. - 558 с. - (Высшее образование). - ISBN 978-5-16-005296-0.
15. Федоров, Н.В. Проектирование информационных систем. – М.: МГИУ, 2018. – 287 с.
16. Шустова, Л. И. Базы данных : учебник / Л.И. Шустова, О.В. Тараканов. — Москва : ИНФРА-М, 2021. — 304 с. + Доп. материалы [Электронный ресурс]. — (Высшее образование: Бакалавриат). — DOI 10.12737/11549. - ISBN 978-5-16-010485-0. - Текст : электронный.
17. Юсупов, Р. Х. Основы автоматизированных систем управления технологическими процессами: Учебное пособие / Юсупов Р.Х. - Москва :Инфра-Инженерия, 2018. - 132 с. ISBN 978-5-9729-0229-3. - Текст : электронный.

18. Яковлев В. П. Эконометрика : учебник для бакалавров / В. П. Яковлев. — Москва : Издательско-торговая корпорация «Дашков и К », 2019. - 384 с. - ISBN 978-5-394-02532-7.
19. Официальный сайт ООО «ОЗОН Фармацевтика» [Электронный ресурс]. – Режим доступа: <https://ozonpharm.ru/>
20. Официальный сайт «Zabbix» [Электронный ресурс]. – Режим доступа: <https://www.zabbix.com/documentation/current/ru/manual>
21. Agarwal B.B., Gupta M., Tayal S.P. Software Engineering And Testing: An Introduction – Jones & Bartlett Publishers, 2009. – 515 p. – ISBN-10 1934015555; ISBN-13 978-1934015551.
22. Alpaev Gennadiy. Software Testing Automation Tips – Apress, 2017. – 50 p.
23. Andrea Dalle Vacche, Masrering Zabbix, second edition, 2019. -278 p. – ISBN – 9781785289262.
24. Nathan Liefting, Brian van Baekel. Zabbix 5 IT Infrastructure Monitoring Cookbook: Explore the New Features of Zabbix 5 for Designing, Building, and Maintaining Your Zabbix Setup, 2021. – 346 p. – ISBN-1800202237, 9781800202238.
25. Nathan Liefting, Zabbix 6 IT Infrastructure Monitoring Cookbook: Explore the new features of Zabbix 6 for designing, building, and maintaining your Zabbix setup, 2nd Edition - Simon and Schuster, 2022. – 528 pages. - ISBN-13: 978-180324691, ISBN-10: 180324691X.

## Приложение А

### Код создания базы данных в MySQL Workbench

```
CREATE TABLE `ozonpharm`.`remains` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `product` VARCHAR(45) NULL,  
  `remainscol` VARCHAR(45) NULL,  
  `quantity` VARCHAR(45) NULL,  
  `comment` VARCHAR(45) NULL,  
  `date` DATETIME NULL DEFAULT CURRENT_TIMESTAMP,  
  PRIMARY KEY (`id`),  
  UNIQUE INDEX `id_UNIQUE` (`id` ASC) VISIBLE)  
  COMMENT = 'Table for data from external API';
```