

Министерство образования и науки Российской Федерации  
ФГАОУ ВПО «УрФУ имени Президента России Б.Н. Ельцина»  
Институт радиоэлектроники и информационных технологий  
Кафедра информационных технологий

Оценка работы \_\_\_\_\_  
Руководитель от УрФУ \_\_\_\_\_

РАЗРАБОТКА ПО ДЛЯ АВТОМАТИЗАЦИИ  
КОНФИГУРИРОВАНИЯ СЕТЕВОГО ОБОРУДОВАНИЯ CISCO  
Отчет по преддипломной практике

Руководители от предприятия

Трофимов С.П.

Студент

Черетаев И.В.

Специальность

Информатика и вычислительная  
техника

Группа

РИ-420207

Екатеринбург, 2016

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ . . . . .	4
1. Обзор существующих решений . . . . .	6
1.1 Cisco Packet Tracer . . . . .	6
1.2 GNS3 . . . . .	7
1.3 Boson NetSim . . . . .	8
1.4 Cisco IOU . . . . .	9
2. Выбор инструментальной системы разработки ПО . . . . .	12
2.1 Code::Blocks . . . . .	12
2.2 Microsoft Visual Studio Express . . . . .	12
2.3 Qt Creator . . . . .	13
3. Техническое задание . . . . .	14
3.1 Общие сведения . . . . .	14
3.1.1 Полное наименование системы и ее условное обозначение . . . . .	14
3.1.2 Краткая характеристика области применения . . .	14
3.1.3 Перечень документов, на основании которых создается система . . . . .	14
3.1.4 Перечень документов, на основании которых устанавливается порядок оформления и предъявления результатов работ . . . . .	14
3.2 Назначение разработки . . . . .	14
3.2.1 Функциональное назначение . . . . .	14
3.2.2 Эксплуатационное назначение . . . . .	15
3.3 Требования к программному средству . . . . .	15
3.3.1 Требования к функциям, выполняемым системой .	15
3.3.2 Требования к надежности функционирования и безопасности . . . . .	15
3.3.3 Требования к информационной и программной совместимости . . . . .	15
3.3.4 Требования к исходным кодам и языкам программирования . . . . .	16
3.3.5 Специальные требования . . . . .	16

3.4	Стадии и этапы разработки . . . . .	16
3.4.1	Стадии разработки . . . . .	16
3.4.2	Этапы разработки . . . . .	16
3.4.3	Содержание работ по этапам . . . . .	16
3.5	Порядок защиты и контроля . . . . .	17
3.6	Требования к программной докуменации . . . . .	17
3.6.1	Предварительный состав программной документации	17
3.7	Источники разработки . . . . .	17
ЗАКЛЮЧЕНИЕ . . . . .		18
Список использованных источников . . . . .		18

## ВВЕДЕНИЕ

Во второй половине прошлого века в связи с «рождением» первых вычислительных сетей произошла очередная научно-техническая революция. Появилась возможность начать использование рассредоточенной обработки данных, обширно использовать для автоматизации различных видов деятельности новые технологии. В наши дни даже маленький ребенок имеет представление о компьютерных сетях, и, в то же время, наладка сетевого оборудования является довольно кропотливым делом, которое не терпит небрежного отношения.

В наше время происходит активное применение сетевых решений во многих сферах деятельности. В условиях производства, на различных предприятиях, в офисах компаний, различных фирмах и учреждениях отдельно стоящий, не подключенный к сети компьютер является большой редкостью. Можно смело заявлять, что через какое-то время большая часть компьютеров будет включено в те или иные сети. Если же подобной сети в учреждении нет или она плохо развита, нет сомнений, что ей придется развиваться.

Основным моментом в данном вопросе является технический прогресс, постоянно обеспечивающий нас новыми возможностями, но накладывающим на технических специалистов высокие требования, в частности – необходимость повышенной адаптивности. Несмотря на это, для специалистов наладка сетевого оборудования является собой одно из основных направлений, где качество безоговорочно удерживает пальму первенства по важности.

Одним из представителей компаний-производителей сетевого оборудования является Cisco Systems. Cisco Systems — компания-производитель сетевого оборудования, основанная в 1984. Сначала компания производила маршрутизаторы, но затем значительно расширила ассортимент своей продукции. В настоящее время она производит коммутаторы, маршрутизаторы, IP-телефоны, программное обеспечение для своего оборудования.

Огромное сообщество сетевых администраторов, IT-специалистов, сту-

дентов не всегда хочет привлекать квалифицированных специалистов Cisco для того, чтобы быстро запустить какой-либо проект на оборудовании Cisco. Они обычно нуждаются в простом, работающем решении, позволяющее им не посвящать остаток жизни изучению всех тонкостей сетевых технологий, маршрутизации, информационной безопасности, беспроводной связи и т.д.

Соответственно, необходимо решение, которое быстро и легко поможет настроить оборудование Cisco и решить данные проблемы.

## 1. ОБЗОР СУЩЕСТВУЮЩИХ РЕШЕНИЙ

Рассмотрим несколько приложений, решающих задачу автоматизации конфигурирования сетевого оборудования Cisco.

Сначала несколько слов по терминологии.

Симуляторы — реализуют статичное множество команд, но, только пользователь выходит за рамки возможного, выдают сообщение об ошибке. Классическим примером является Cisco Packet Tracer.

Эмуляторы — дают возможность выполнять команды образов настоящих устройств, порой без заметных урезаний функциональности. Эмулятором, к примеру, является GNS3/Dynamips.

### 1.1. Cisco Packet Tracer

Начнем обзор с Packet Tracer (см рис. 1) — официального выпускаемого Cisco программного обеспечения. Данный продукт доступен в версиях как для Windows, так и для Linux, бесплатно для учащихся Сетевой Академии Cisco.

Вся настройка осуществляется при помощи логической диаграммы сети, для моделирования доступен большой набор оборудования, выпускаемого Cisco и не только (маршрутизаторы, коммутаторы, персональные компьютеры и многое другое).

Достоинства Packet Tracer — дружелюбность и логичность интерфейса. Также, в нем удобно проверять функционирование различных сетевых сервисов, например DHCP или DNS. И одно из самых интересных преимуществ — это возможность перейти в режим симуляции и увидеть все действия с пакетами с использованием замедления времени.

Однако, стоит заметить, что реализованная функциональность устройств ограничена и не предоставляет всех возможностей реального оборудования. Большим минусом является тот факт, что некоторые команды не поддерживает. Почти всё, что не входит в программу изучения CCNA, на нем собрать не получится.

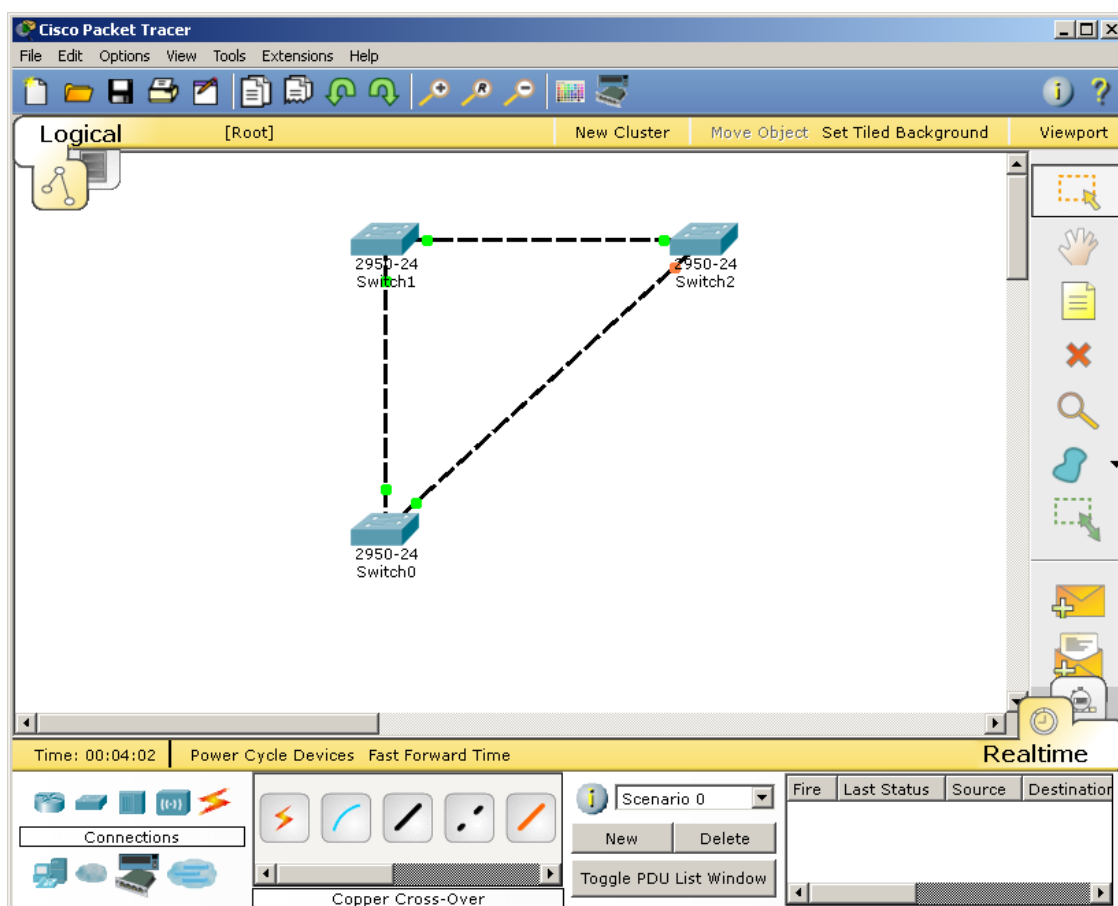


Рисунок 1. Cisco Packet Tracer.

## 1.2. GNS3

Следующий продукт в обзоре — GNS3, представляющий собой графический интерфейс, написанный на Qt, для эмулятора dynamips (см. рис. 2). Является свободно распространяемым проектом и доступен под следующими операционными системами: Linux, Windows и Mac OS X.

В то же время, большинство его функций, призванных улучшить производительность, работают только под Linux, 64 битная версия так же только для Linux.

Это эмулятор, который работает с оригинальными прошивками IOS. Это означает, что для использования GNS3, вам необходимо иметь в наличии реальные образы.

Так же имеется несколько других недостатков:

- ограниченное число платформ;
- невозможность полноценно использовать коммутаторы Catalyst;
- уменьшение производительности при использовании огромного ко-

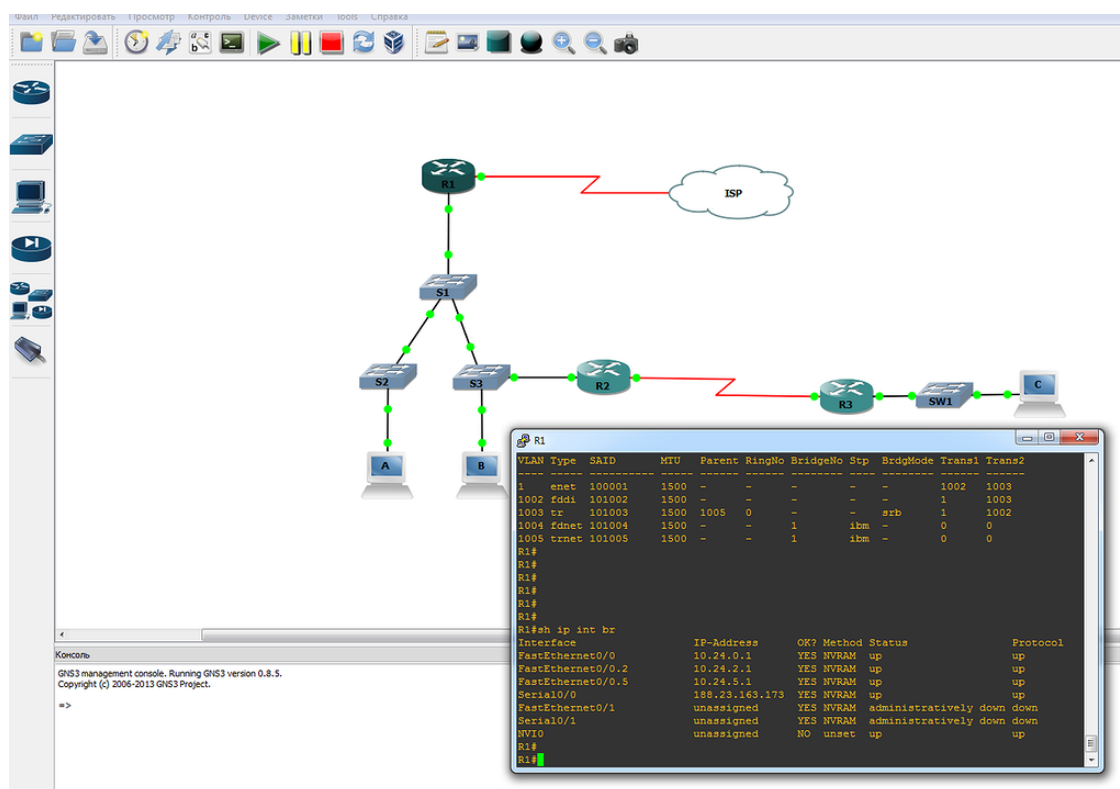


Рисунок 2. GNS3.

личества устройств.

### 1.3. Boson NetSim

Доступен в версии только для Windows, цена колеблется от 179CCN A349 за CCNP.

Выступает в роли набора лабораторных работ, объединенных по темам экзамена. Интерфейс разделен на нескольких разделов: постановка задачи, карта сети, в левой части находится список доступных лабораторных работ (см. рис. 3).

По завершении работы, можно получить результат и узнать все ли было сделано.

Присутствует возможность создания собственных топологий, с некоторыми ограничениями.

Ключевые возможности Boson NetSim:

- поддержка 42 маршрутизаторов, 6 коммутаторов;
- симуляция сетевого трафика при помощи технологии виртуальных пакетов;
- два режима просмотра: telnet или подключение по консоли;



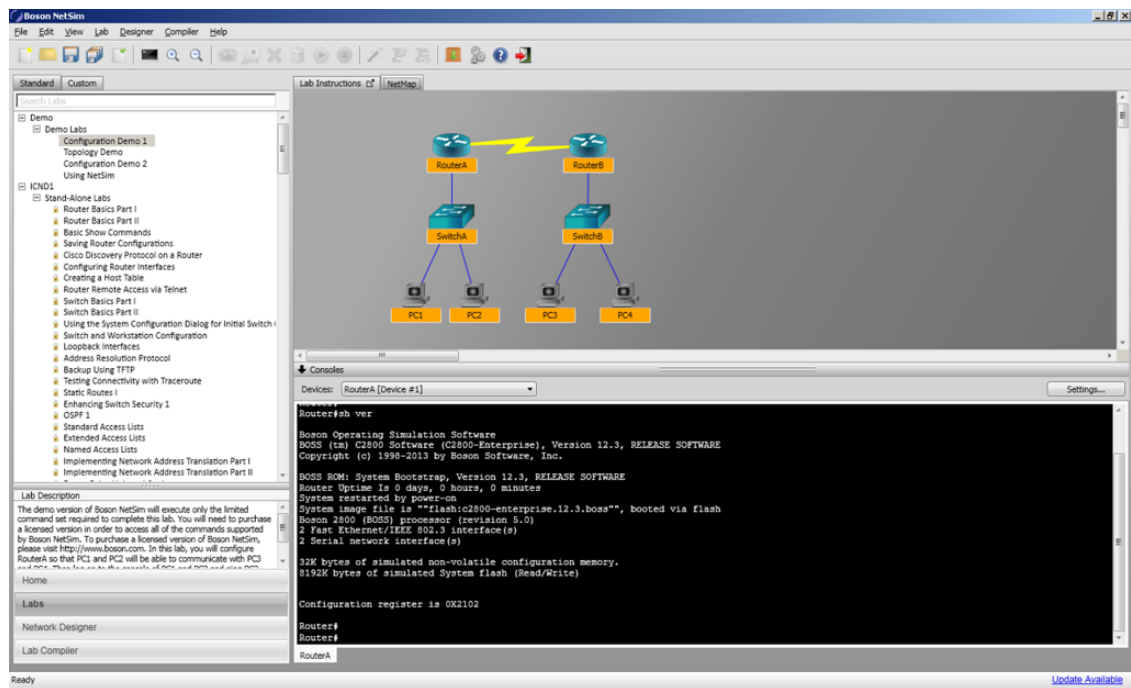


Рисунок 3. Boson NetSim.

- возможность создания собственных лабораторий.

#### 1.4. Cisco IOU

Последний в этом списке это Cisco IOU (Cisco IOS on UNIX) — проприетарное ПО, которое официально не распространяется (см. рис. 4).

Существует мнение, что у Cisco есть возможность отследить и идентифицировать того, кто использует IOU. Известный факт состоит в том, что центр технической поддержки Cisco использует именно данных продукт.

Сначала был доступен только под Solaris, но затем был портирован и на Linux. В комплекте идут две основные части — l2iou и l3iou. Первая часть эмулирует канальный уровень и коммутаторы, вторая же — сетевой уровень и маршрутизаторы.

Настройка проводится путем редактирования конфигурационных файлов, недавно для него стал доступен также графический интерфейс.

Интерфейс довольно интуитивен, и позволяет выполнять почти все действия.

Включение на моделирование топологии, указанной на рис. 5, приводит примерно к 20% загрузке процессора.

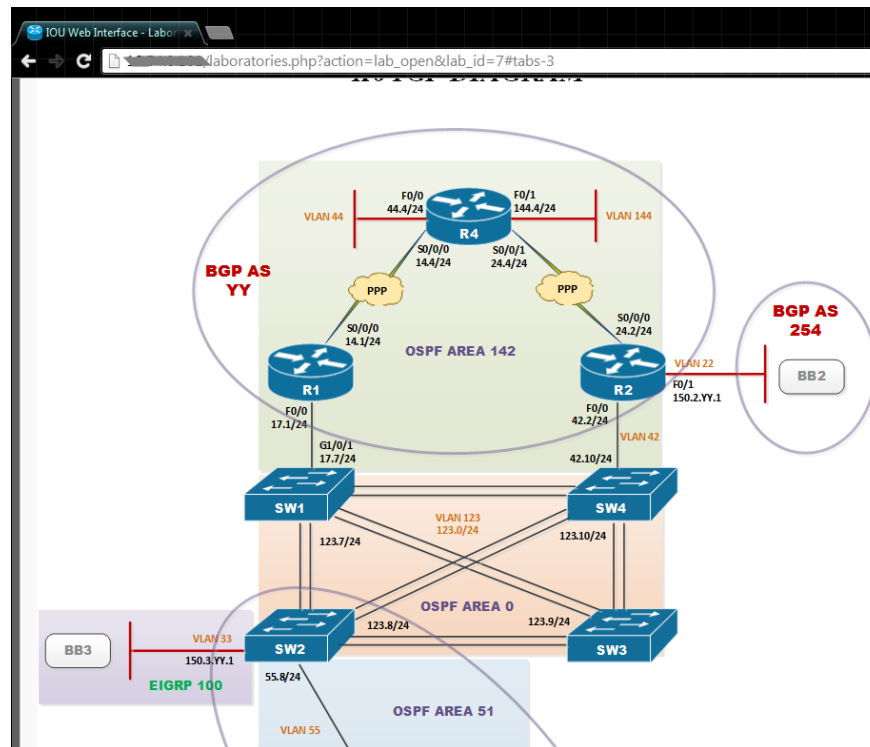


Рисунок 4. Cisco IOU.

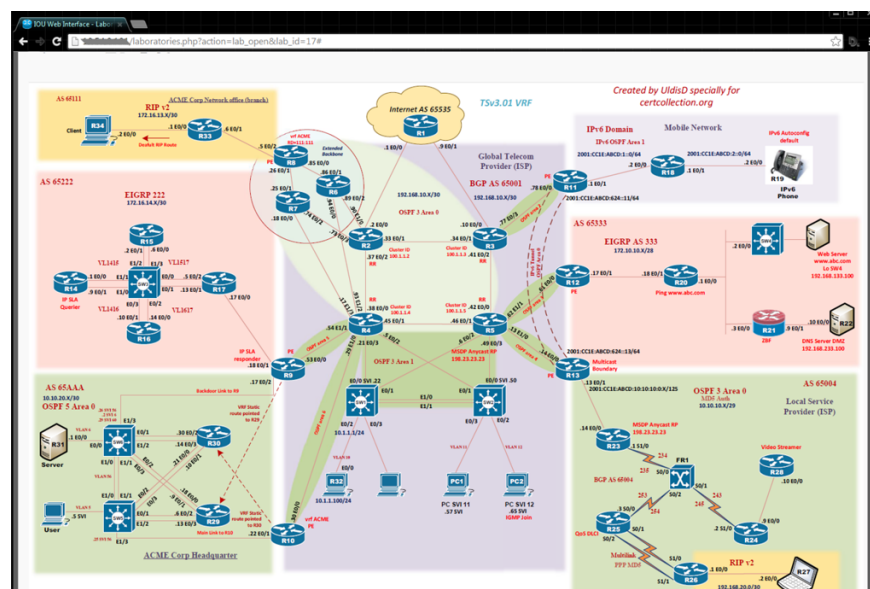


Рисунок 5. Пример топологии.

Надо сказать, данная топология предназначена для подготовки к сдаче экзамена на Cisco Certified Internetwork Expert.

Возможности IOU на самом деле очень широкие. В тоже время, некоторые проблемы на канальном уровне все же имеются. Для некоторых устройств, например, отсутствует возможность жестко выставить дуплексный режим, но это всё мелочи.

Таблица 1 - Сравнение существующих решений.

Наименование показателей	Cisco Packet Tracer	GNS3	Boson NetSim	Cisco IOU
Тип решения	симулятор	эмулятор	симулятор	симулятор
Ценовая политика	доступен студентам сетевой академии Cisco	бесплатен	от 179\$ за CCNA и до 349\$ за CCNP	проприетарен
Версии ОС	Windows, Linux	Linux, Windows и Mac OS X	Windows	Solaris, Linux

II

Общим недостатком представленных программных продуктов можно назвать излишнюю комплексность решения, вызванную попытками смоделировать полное поведение сетевых технологий, что, в свою очередь, вызывает дальнейшие сложности при освоении.

## 2. ВЫБОР ИНСТРУМЕНТАЛЬНОЙ СИСТЕМЫ РАЗРАБОТКИ ПО

В ходе прохождения преддипломной практики были рассмотрены несколько инструментальных сред разработки на C/C++. В связи с тем проект некоммерческий, в списке присутствуют только бесплатные программные продукты.

### 2.1. Code::Blocks

Разработчики Code::Blocks выбрали путь открытой архитектуры, позволив сторонним программистам повышать эффективность программы за счет собственных разработок (плагинов). Об одном из плагинов нужно сказать отдельно - wxSmith, который, по сути, является wxWidgets RAD инструментом, то есть позволяет создавать оконные формы и прочие графические объекты, используя библиотеку wxWidgets (библиотека wxWidgets устанавливается отдельно).

Плюсы Code::Blocks:

- подсветка кода;
- автозавершение кода;
- просмотрщик классов;
- быстрая система сборки (не требуются make-файлы);
- поддержка параллельных сборок.

### 2.2. Microsoft Visual Studio Express

Эта версия Visual Studio представляет собой набор урезанных средств разработки для языков Visual Basic, C# и C++, и обозначается Microsoft как инструментальная среда разработки начального уровня для тех лиц, кто не занимается профессионально программированием (школьников, студентов, любителей и т.д.).

Графический интерфейс и возможность создать оконные приложения присутствует, но возможность воспользоваться наработками компании в области оптимизации и рефакторинга кода почти отсутствует.

### 2.3. Qt Creator

Qt Creator является средой разработки для кроссплатформенного фреймворка Qt. Вследствие этого, нужно указать на следующие его возможности:

- интеграция дизайнера форм Qt и справочной системы Qt
- расширяемость (посредством плагинов)
- поддержка дебагеров GDB (графический фронтенд) и CDB
- подсветка кода с поддержкой нескольких языков и разметок

Таблица 2 - Сравнение IDE.

Наименование показателей	Code::Blocks	Microsoft Visual Studio Express	Qt Creator
Лицензия	GPL	Freeware	GPL
Кроссплатформенность	Да	Нет	Да
Разработка GUI	Да <sup>1</sup>	Да	Да

Исходя из приведенных выше сведений, было принято решение, что приложение будет написано в среде Qt Creator с использованием кроссплатформенного фреймворка Qt на языке C++. Это позволит сократить время на разработку, избежать написания большого количества кода, посредством использования готовых библиотек. Использование библиотеки Qt также поможет создать удобный интерфейс, отличный от стандартного.

---

<sup>1</sup>С использованием плагина wxSmith

### 3. ТЕХНИЧЕСКОЕ ЗАДАНИЕ

#### 3.1. Общие сведения

##### 3.1.1. Полное наименование системы и ее условное обозначение

Программное обеспечение для автоматизации конфигурирования сетевого оборудования Cisco. Условное обозначение – АКСО[4].

##### 3.1.2. Краткая характеристика области применения

АКСО предназначено для автоматизации конфигурирования сетевого оборудования Cisco путем предоставления пользователю графического интерфейса с последующим получением набора команд, необходимых для внедрения изменений внесенных пользователем.

##### 3.1.3. Перечень документов, на основании которых создается система

- 1) Приказ ректора УрФУ № \_\_\_\_\_ от "\_\_\_" \_\_\_\_\_ г.

##### 3.1.4. Перечень документов, на основании которых устанавливается порядок оформления и предъявления результатов работ

- 1) Методические указания по выполнению выпускной квалификационной работы бакалавра техники и технологий по направлению "Информатика и вычислительная техника"/ сост. А.Б.Николаев. – Москва: МАДИ, 2010. – 17 с.
- 2) Соколов С.С. Рекомендации по оформлению курсовых, выпускных и дипломных проектов (работ). Электронные методические указания - Екатеринбург: Изд. УрФУ, 2010. - 38 с.

#### 3.2. Назначение разработки

##### 3.2.1. Функциональное назначение

Функциональным назначением АКСО является предоставлению пользователю удобных инструментов для облегчения конфигурирования сетевого оборудования от компании Cisco Systems, таких как маршрутизаторы и/или коммутаторы.

### 3.2.2. Эксплуатационное назначение

АКСО предназначено для автоматизации процесса конфигурирования сетевого оборудования с целью получения исходного текста конфигурации либо автоматического применения внесенных с помощью АКСО изменений на сетевом оборудовании.

## 3.3. Требования к программному средству

### 3.3.1. Требования к функциям, выполняемым системой

- 1) Наличие основных моделей маршрутизаторов и коммутаторов.
- 2) Наличие редактора, позволяющего добавлять новые модели (в том числе, на основе уже существующих моделей), изменять и удалять существующие. Редактор должен обеспечивать возможность поиска в списке имеющихся моделей. Так же при помощи данного редактора должны производиться операции визуального изменения конфигурации для выбранной модели сетевого оборудования.
- 3) Наличие возможности экспорта/импорта выбранной конфигурации в виде XML-файла, содержащего все необходимые сведения.
- 4) Наличие возможности сохранения выбранной конфигурации в виде текстового файла, содержащего команды конфигурирования
- 5) Наличие справочную подсистему. Справка должна содержать краткую информацию о системе и ее возможностях, описание действий пользователя и получаемых результатов при работе с программным обеспечением.

### 3.3.2. Требования к надежности функционирования и безопасности

Надёжность системы должна обеспечивать работоспособность в течение всего срока эксплуатации при бесперебойном питании ЭВМ. Программное обеспечение не должно содержать явных логических ошибок и функционировать без сбоев.

### 3.3.3. Требования к информационной и программной совместимости

АКСО должна иметь возможность функционировать под управлением различных операционных систем (Windows, Linux и т.д.).

### 3.3.4. Требования к исходным кодам и языкам программирования

Исходные коды программного средства должны быть реализованы на языке C++. В качестве интегрированной среды разработки программы должна быть использована среда Qt Creator.

### 3.3.5. Специальные требования

Программа должна обеспечивать взаимодействие с пользователем (оператором) посредством графического пользовательского интерфейса.

## 3.4. Стадии и этапы разработки

### 3.4.1. Стадии разработки

Разработка должна быть произведена в три стадии[3]:

- 1) Разработка технического задания;
- 2) Рабочее проектирование;
- 3) Внедрение;

### 3.4.2. Этапы разработки

На стадии рабочего проектирования должны быть выполнены перечисленные ниже этапы работ:

- 1) разработка АКСО;
- 2) разработка программной документации;
- 3) испытания АКСО.

На стадии внедрения должен быть выполнен этап разработки - подготовка АКСО.

### 3.4.3. Содержание работ по этапам

На этапе разработки АКСО должна быть выполнена работа по программированию (кодированию) и отладке программного обеспечения (АКСО).

На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованием п. 3.6.1 настоящего технического задания.

На этапе испытаний АКСО должны быть выполнены перечисленные ниже виды работ:



- 1) проверка выполнения заданных функций АКСО;
- 2) выявления и устранения недостатков в АКСО и программной документации;
- 3) корректировка АКСО и программной документации по результатам тестирований.

На этапе подготовки АКСО должна быть выполнена работа по подготовке программного средства и программной документации для эксплуатации.

### 3.5. Порядок защиты и контроля

Защита осуществляется перед Государственной аттестационной комиссией (ГАК), утвержденной приказом ректора.

### 3.6. Требования к программной документации

#### 3.6.1. Предварительный состав программной документации

Предварительный состав программной документации должен включать в себя[5]:

- 1) техническое задание;
- 2) текст программы;
- 3) описание программы;
- 4) пояснительную записку[1, 2];
- 5) руководство пользователя.

### 3.7. Источники разработки

- 1) ГОСТ 19.201-78. Техническое задание, требования к содержанию и оформлению.
- 2) ГОСТ Р ИСО/МЭК 9294-93 Информационная технология. Руководство по управлению документированием программного обеспечения.
- 3) ISO/IEC 12207:2008 (ГОСТ Р) Системная и программная инженерия. Процессы жизненного цикла программных средств.
- 4) ISO/IEC 9126:1991 (ГОСТ Р) Информационные технологии. Оценка программного продукта. Характеристики качества и порядок их применения.

## ЗАКЛЮЧЕНИЕ

Актуальность темы дипломной работы определяется быстрым ростом количества компаний, имеющих сложную информационную структуру. Наряду с этим возникает необходимость настройки огромного количества активного сетевого оборудования. Все эти факторы приводят к тому, что администраторам информационной инфраструктуры компании приходится тратить много времени на порой рутинные работы.

Цель дипломной работы состоит в разработке системы для автоматизированного создания конфигурационных файлов оборудования.

Поставленная цель обусловила следующие задачи дипломной работы:

- определить сущности проектируемой системы и архитектуру их взаимодействия;
- реализовать систему в виде кроссплатформенного приложения с графическим интерфейсом;
- провести тестированием в лабораторной среде.

Объектом исследования выступает сетевое оборудование Cisco Systems.

Предметом исследования в дипломной работе являются современные средства коммуникации и их интеграция в методы управления информационной инфраструктурой для повышения таких аспектов информационной безопасности предприятия, как целостность и доступность.

Теоретическая значимость дипломного исследования состоит в развитии и совершенствовании методологии управления оборудованием информационной инфраструктуры предприятия.

Практическая значимость работы определяется тем, что ее результаты позволяют повысить степень эффективности управления информационной инфраструктурой предприятия и снизить связанные с этим операционные расходы при использовании разработанной системы, направленной на повышение уровня таких аспектов администрирования, как быстрота и легкость обслуживания.

Новизна дипломной работы заключается в разработке и реализации кроссплатформенной модели автоматизации конфигурирования оборудования.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] С.С. Соколов. Рекомендации по оформлению курсовых, выпускных и дипломных проектов (работ). УрФУ, Екатеринбург, 2010.
- [2] А.Б. Николаев. Методические указания по выполнению выпускной квалификационной работы бакалавра техники и технологий по направлению «Информатика и вычислительная техника». МАДИ, Москва, 2010.
- [3] ISO/IEC 12207:2008 (ГОСТ Р). Системная и программная инженерия. Процессы жизненного цикла программных средств.
- [4] ГОСТ 19.201-78. Техническое задание, требования к содержанию и оформлению.
- [5] ГОСТ Р ИСО/МЭК 9294-93. Информационная технология. Руководство по управлению документированием программного обеспечения.