

**ФЕДЕРАЛЬНОЕ АГЕНСТВО СВЯЗИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ТЕЛЕКОММУНИКАЦИЙ ИМ. ПРОФ. М. А. БОНЧ-БРУЕВИЧА» (СПбГУТ)  
ФАКУЛЬТЕТ ИНСТИТУТ НЕПРЕРЫВНОГО ОБРАЗОВАНИЯ (ИНО)**

## **АНАЛИТИЧЕСКИЙ ПРАКТИКУМ №2**

по дисциплине:

### **«Разработка имитационных моделей инфокоммуникационных сетей и систем»**

тема: Коммутируемые локальные сети

Выполнил студент:

Рыжкова Дарья Анатольевна

IV курса группы ПИБ-11з

*(09.03.04 - Программная инженерия)*

студенческий билет № **1905218**

Санкт-Петербург, 2025

## Цель практикума

Исследование технологий объединения локальных сетей и сравнительный анализ эффективности различных архитектур коммутируемых сетей.

## Порядок выполнения практикума

Перед созданием новой модели сети необходимо добавить новый проект (**project**) и сценарий (**scenario**). Проект – это группа зависимых сценариев, каждый из которых описывает различные детали сети. Проект может содержать множество сценариев.

### Создание нового проекта:

1. Запуск **Riverbed Modeler Academic Edition**, в меню **File** выбрать **New...**
2. Выбрать **Project**  $\Rightarrow$  **OK**  $\Rightarrow$  озаглавить проект как *<номер\_студенческого>\_SwitchedLAN*  $\Rightarrow$  *1905218\_SwitchedLAN*, а сценарий – как *OnlyHub*  $\Rightarrow$  **OK**.

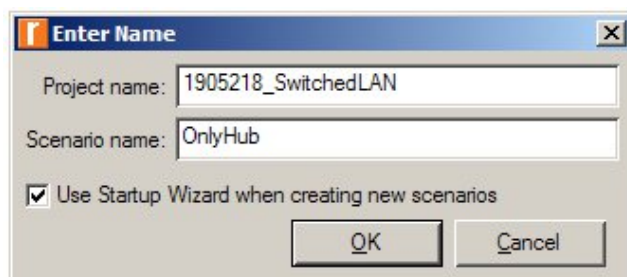


Рисунок 1. Создание проекта *1905218\_SwitchedLAN* и сценария *OnlyHub*

После создания нового сценария для добавления нужно использовать начальный **<мастер запуска> (Startup Wizard)**. Параметры мастера позволяют: определить начальную топологию сети, масштаб и размер сети; выбрать карту для фона сети (план здания, города); сопоставить палитру компонентов (базу ресурсов сети) сценарию.

3. В окне **Startup Wizard: Initial Topology** выбрать **Create empty scenario**  $\Rightarrow$  нажать **Next**  $\Rightarrow$  в списке **Network Scale** выбрать *Office*  $\Rightarrow$  **Next**.
4. В окне **Startup Wizard: Specify Size** в выпадающем меню **Units** выбрать *Meters*  $\Rightarrow$  дважды нажать **Next**  $\Rightarrow$  **Finish**.

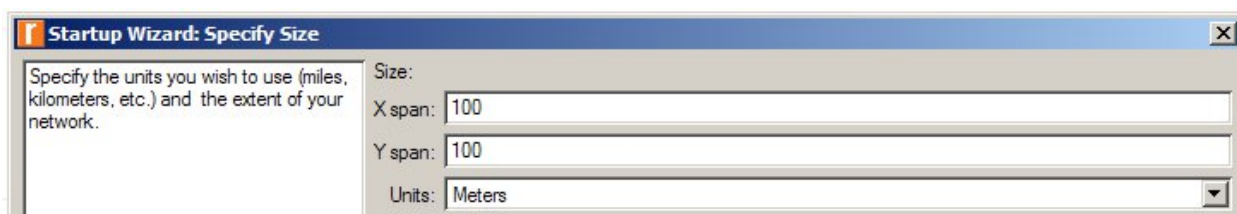


Рисунок 2. Выбор масштаба сети (диалоговое окно *Choose Network Scale*). Выбираем Office – сеть масштаба офиса. Выбрана опция *Use metric units* – использование метрической системы. Определение размера рабочей области 100x100м (по умолчанию). Диалоговое окно *Select Technologies* для выбора используемых технологий в данной лабораторной работе пропускаем со значениями, выставленными по умолчанию

## Создание сети Ethernet с топологией 'звезда' (Star)

Модель сети создается с помощью редактора, используя узлы (nodes) и каналы связи (links) из базы ресурсов (object palette). Узел сети – это реальный объект сети, способный передавать и принимать информацию. Канал связи представляет собой среду передачи данных, между узлами, которая может быть электрическим или оптоволоконным кабелем. Все эти объекты доступны в базе ресурсов (окне с изображениями узлов и связей).

Для создания топологии сети можно использовать один из трех методов или их комбинацию. Первый метод предполагает импорт существующей топологии. Второй — ручная расстановка узлов и связей в рабочей области. Третий метод, который мы будем использовать, — это метод «быстрого создания топологии» (**Rapid Configuration**). Этот метод позволяет быстро создать сеть, выбрав тип топологии, узлов и каналов связи, что упрощает процесс проектирования сети.

Метод Rapid Configuration создает сеть за одно действие, после выбора топологии сети, типов узлов и типов связей между узлами.

1. Закрывать открывшееся окно базы ресурсов **Object Palette**.
2. В меню **Topology** выбрать **Rapid Configuration...**
3. В выпадающем меню **Configuration** выбрать *Star* ⇒ **Next**.

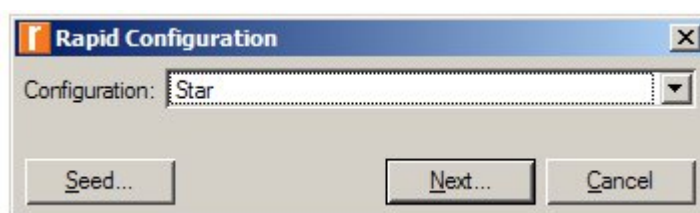


Рисунок 3. Выбор типа топологии сети

4. В открывшемся окне **Rapid Configuration: Star** нажать **Select Models...**  
 ⇒ в выпадающем меню **Model List** выбрать *ethernet* ⇒ OK.
5. В окне **Rapid Configuration: Star** установить 7 значений в соответствии с методическими указаниями ⇒ OK.

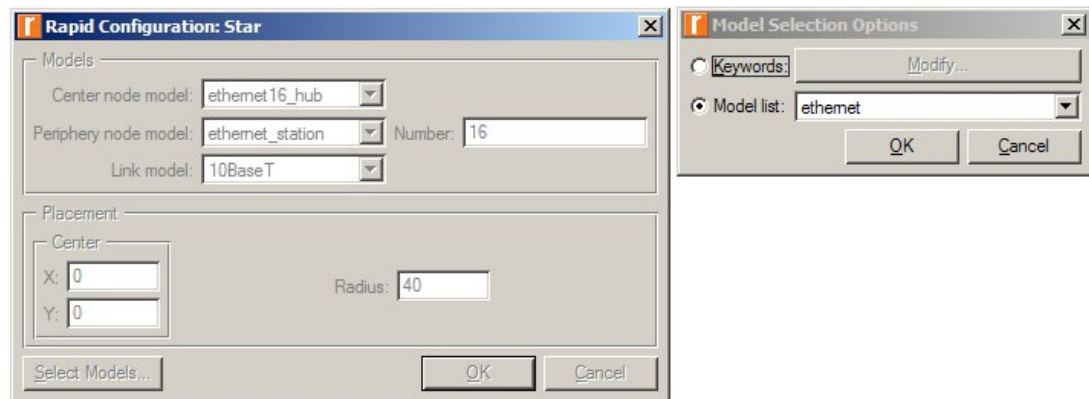


Рисунок 4. Создание сети Ethernet топологии «звезда» с 16 узлами, ее позиционирование в рабочей области и тип соединения с хабом

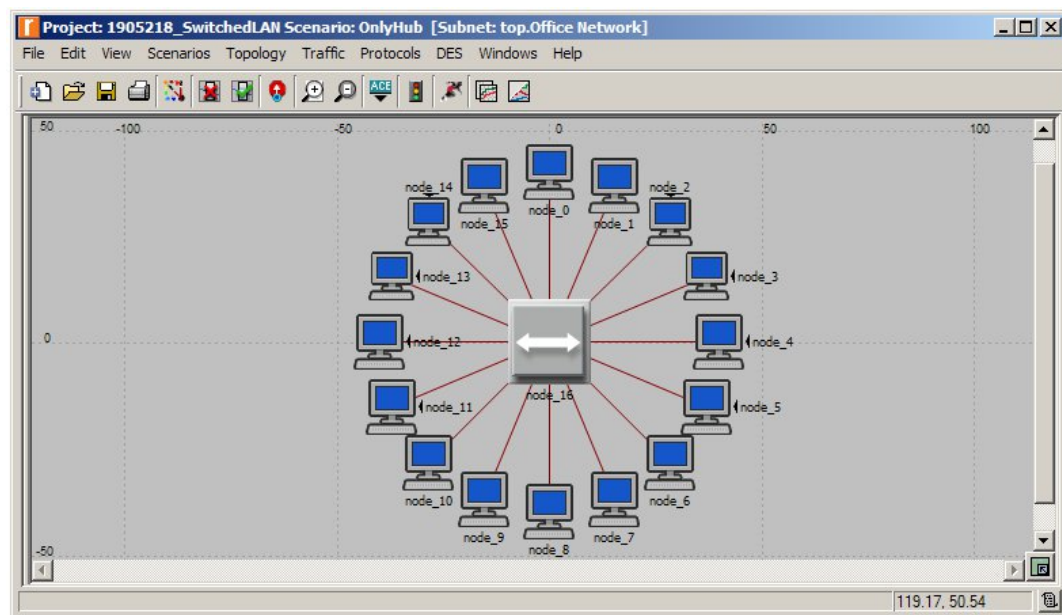


Рисунок 5. Получившаяся сеть после настройки конфигурации Rapid Configuration

6. ПКМ на хабе ⇒ в появившемся меню выбрать **Edit Attributes**.
7. В открывшемся окне установить значение параметра **name** как *Hub1*  
 ⇒ OK.

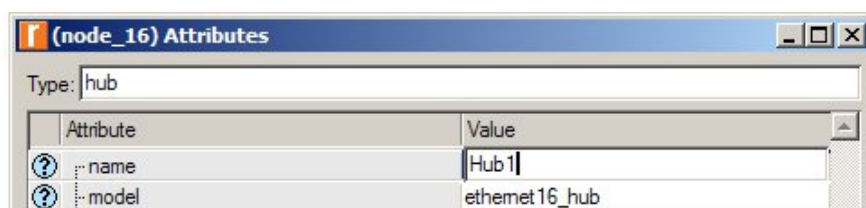


Рисунок 6. Окно настройки атрибутов хаба сети. Изменение имени хаба, данного ему по умолчанию при настройке сети, node\_16, на Hub1

## 8. Сохранить проект.

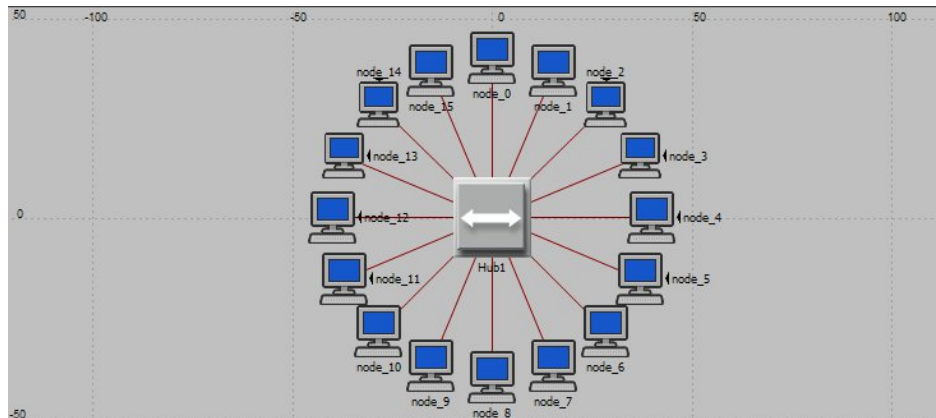


Рисунок 7. Получившаяся сеть после настройки параметра *name* для хаба

## Генерация трафика

1. ПКМ на любой из 16 станций (node\_0, node\_1, ..., node\_15)  $\Rightarrow$  выбрать **Select Similar Nodes**, чтобы выделить разом все станции в сети. ПКМ на любой из выделенных 16 станций  $\Rightarrow$  в появившемся меню выбрать **Edit Attributes**.
2. В открывшемся окне поставить галочку в поле **Apply to selected objects**, чтобы избежать необходимости конфигурировать каждую станцию в отдельности.
3. Открыть иерархическое меню **Traffic Generation Parameters** и **Packet Generation Arguments**  $\Rightarrow$  установить следующие 4 значения в соответствии с методическими указаниями  $\Rightarrow$  **OK**  $\Rightarrow$  сохранить проект.

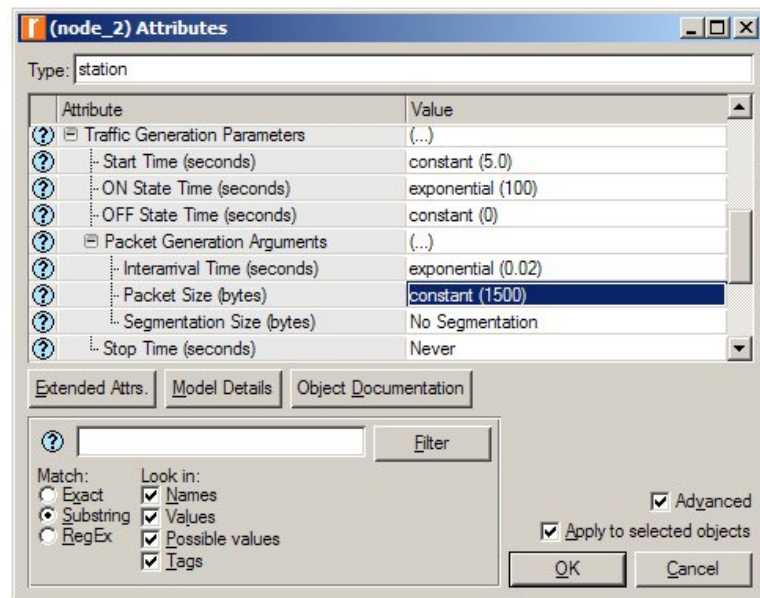


Рисунок 8. Настройка параметров Traffic Generation Parameters для узлов сети

## Сбор статистики

1. ПКМ где-либо на рабочей области (но не на одном из элементов сети)  $\Rightarrow$  в появившемся меню выбрать **Choose Individual DES Statistics**  $\Rightarrow$  открыть иерархическое меню **Global Statistics**.
2. Выбрать следующие 4 метрики в соответствии с методическими указаниями  $\Rightarrow$  **ОК**.

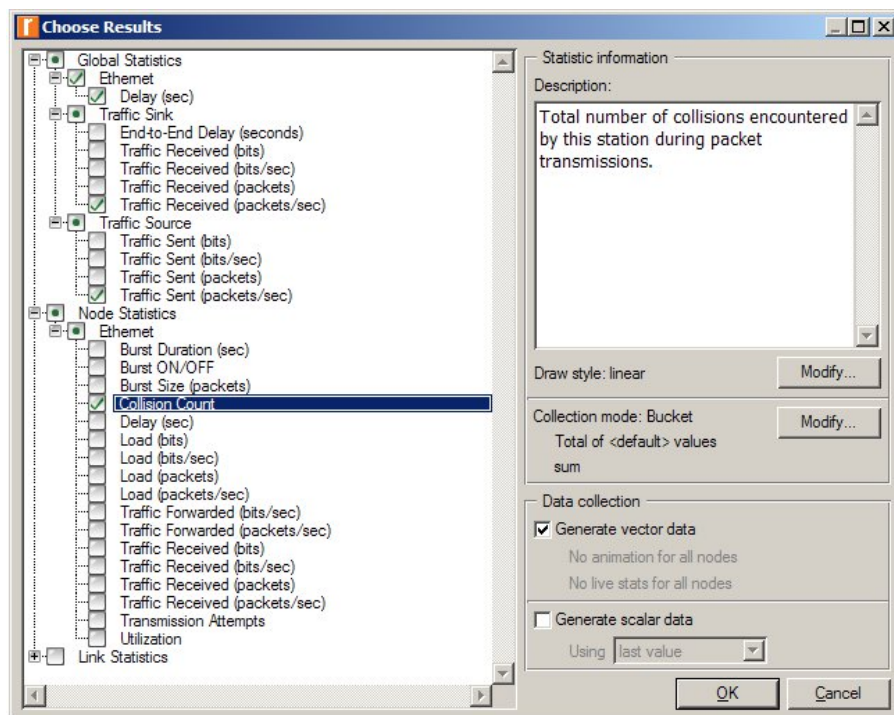


Рисунок 9. Выбор метрик для сбора статистики



## Настройка параметров моделирования

1. На панели инструментов нажать кнопку **Configure/Run Discrete Event Simulation (DES)**.
2. В открывшемся окне **Configure/Run DES** установить значение параметра **Duration** равным *2 minute(s)*  $\Rightarrow$  для запуска моделирования нажать **Run**.

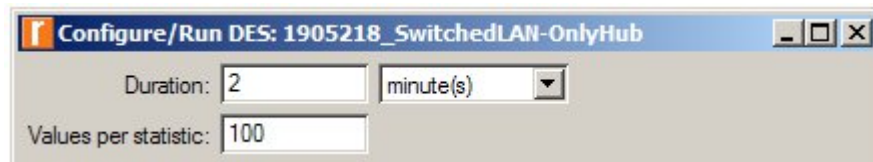


Рисунок 10. Установка времени проведения имитационного моделирования для сценария *OnlyHub*

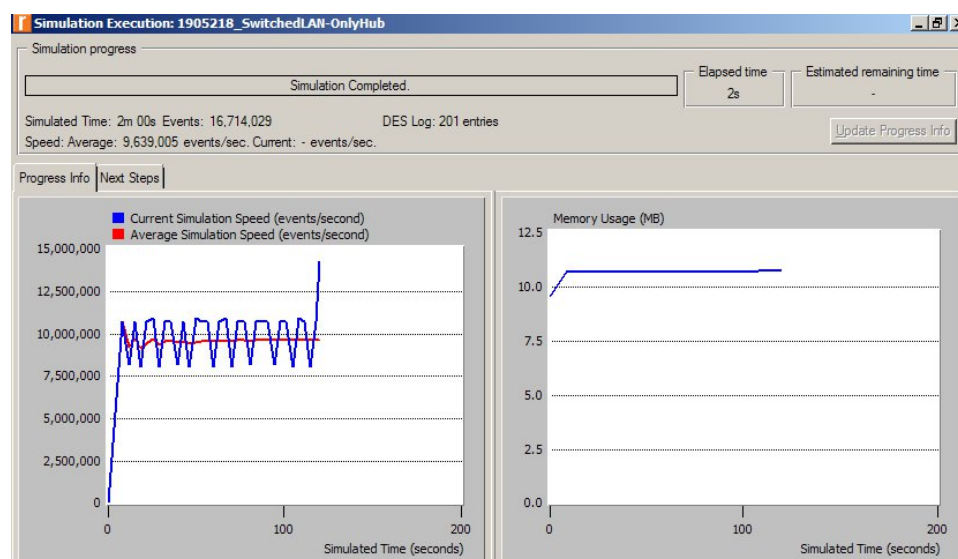


Рисунок 11. Результат выполнения прогона. Вкладка *Progress Info*. *Elapsed time* – длительность прогона в секундах (реальное время). *Simulated time* – модельное время.

3. По окончании нажать **Close** и сохранить проект.

## Создание копии сценария

Во время сравнения сетей необходимо сохранить базовую сеть как один сценарий, а экспериментальную сеть создать в другой сценарии. Необходимо скопировать существующий сценарий и уже в копии можно делать изменения топологии.

1. Создать копию сценария **OnlyHub** и озаглавить ее как *HubAndSwitch*.
2. На панели инструментов нажать кнопку **Open Object Palette**.



Рисунок 12. Панель инструментов. Кнопка *Open Object Palette*

3. В открывшемся окне **Object Palette** в иерархическом меню **Shared Object Palettes** выбрать *ethernet*.
4. Добавить на рабочую область проекта *ethernet16\_hub* и *ethernet16\_switch*.
  - ☐ Для добавления перетащить мышкой соответствующий объект из окна **Object Palette** на рабочую область.
  - ☐ Щелчок ЛКМ добавит еще 1 копию объекта, щелчок ПКМ прекратит копирование.

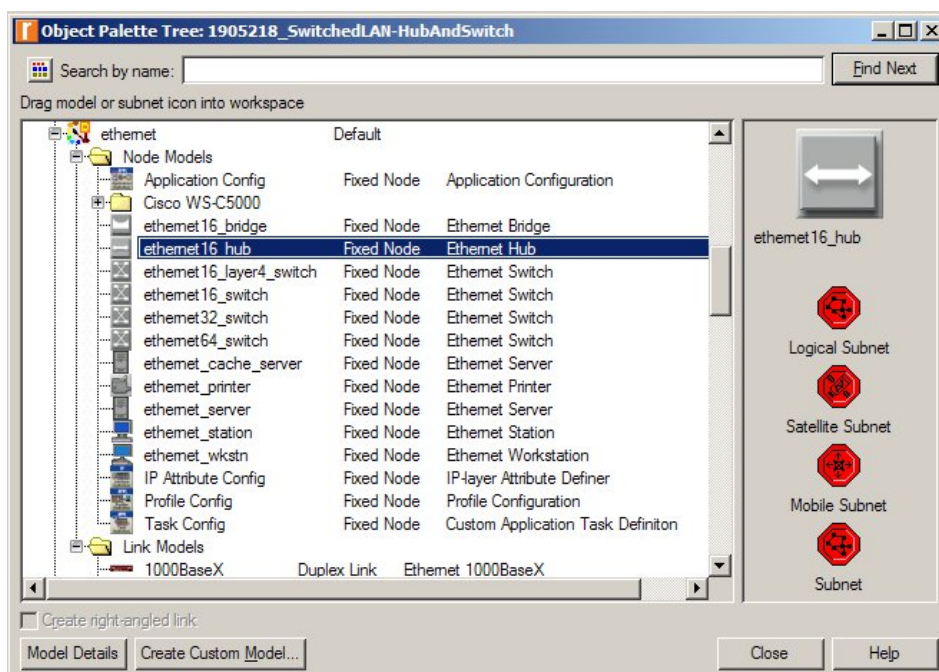


Рисунок 13. Вид окна *Object Palette*

5. ПКМ на *новом хабе*  $\Rightarrow$  в появившемся меню выбрать **Edit Attributes**.
6. В открывшемся окне установить значение параметра **name** как *Hub2*  $\Rightarrow$  нажать **ОК**.

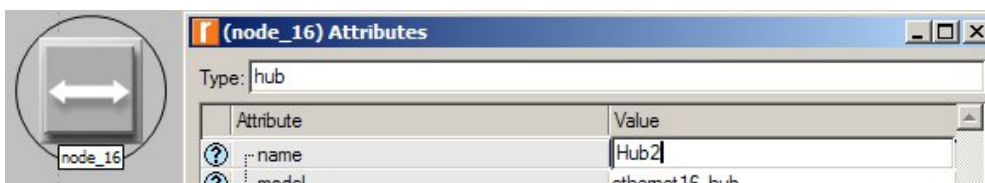


Рисунок 14. Окно настройки атрибутов второго хаба сети. Изменение имени хаба, данного ему по умолчанию при настройке сети, *node\_16*, на *Hub2*

7. ПКМ на *коммутаторе*  $\Rightarrow$  в появившемся меню выбрать **Edit Attributes**.



8. В открывшемся окне установить значение параметра **name** как *Switch*  $\Rightarrow$  нажать **ОК**.

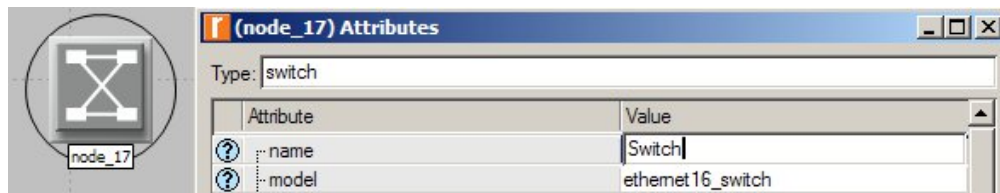


Рисунок 15. Окно настройки атрибутов второго коммутатора. Изменение имени коммутатора, данного ему по умолчанию при настройке сети, node\_17, на Switch

9. Сконфигурировать сеть в соответствии с методическими указаниями по выполнению лабораторной работы:
- ☐ Чтобы удалить соединительную линию, выделите ее  $\Rightarrow$  нажмите на клавиатуре **Delete**.
  - ☐ Чтобы добавить соединительную линию, перетащите мышкой *10BaseT* из окна **Object Palette** на рабочую область  $\Rightarrow$  поочередно щелкните ЛКМ на соединяемых объектах.

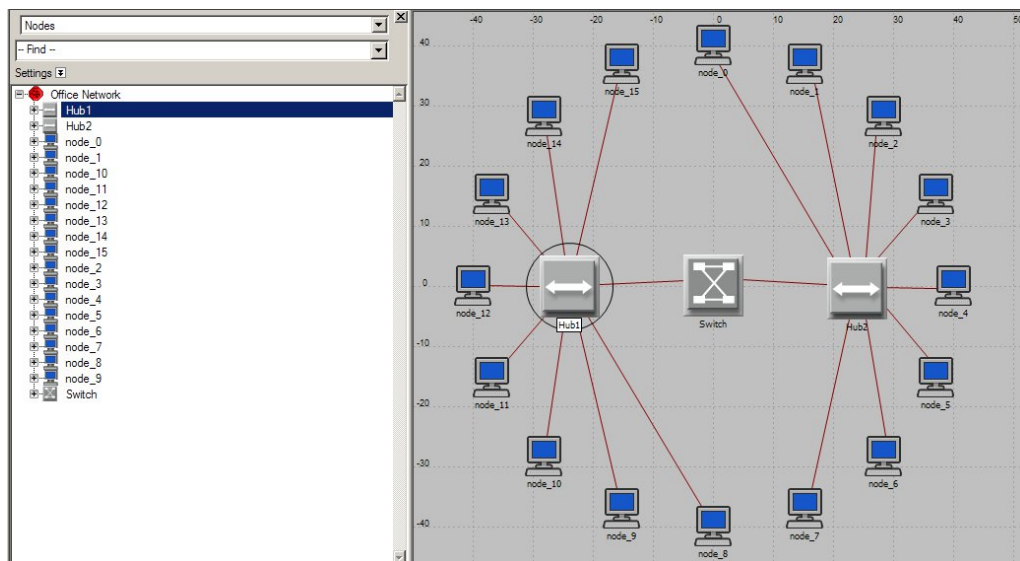


Рисунок 16. Вид измененной сети с добавлением хаба Hub2 и коммутатора Switch в сценарии HubAndSwitch

10. Запустить имитационное моделирование для данного сценария.  
11. По окончании сохранить проект.

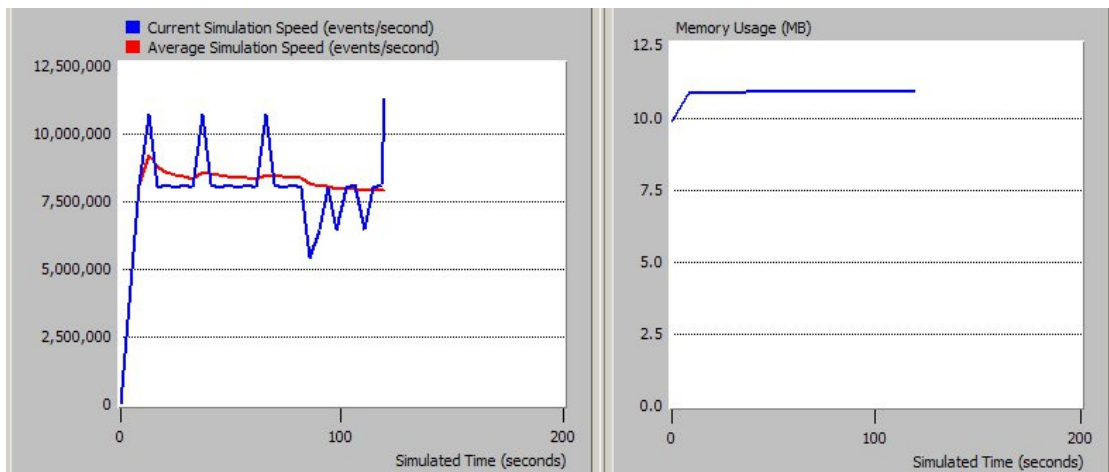


Рисунок 17. Результат выполнения прогона сценария HubAndSwitch. Вкладка Progress Info

## Просмотр результатов моделирования

1. В меню **DES** выбрать **Results**  $\Rightarrow$  выберите **Compare Results...**
2. В открывшемся окне **Results Browser** выбрать оба сценария  $\Rightarrow$  в выпадающем меню выбрать **Overlaid Statistics** и **As Is**.
3. Получившиеся в результате графики должны быть аналогичны приведенным в методических указаниях. Результаты могут слегка отличаться в зависимости от расположения узлов и расстояния между ними.
4. Открыть иерархическое меню **Global Statistics**.
5. Открыть иерархическое меню **Ethernet**  $\Rightarrow$  выбрать *Delay (sec)*  $\Rightarrow$  **Show**.

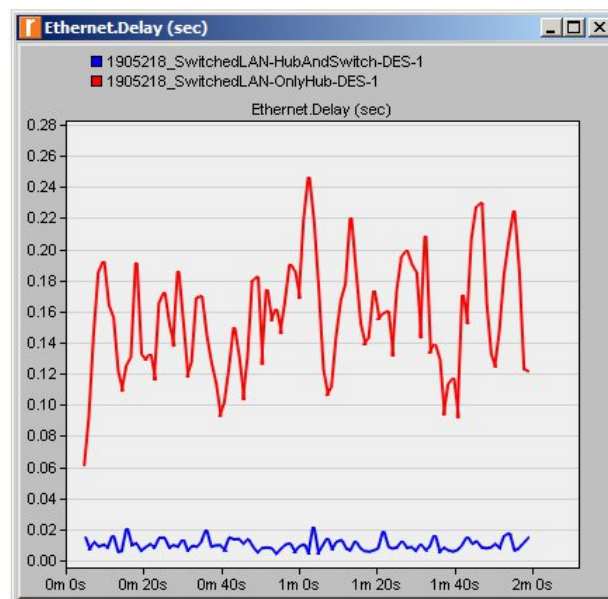


Рисунок 18. График для метрики Delay (sec) для сценариев OnlyHub и HubAndSwitch

HubAndSwitch – гибридная сеть с хабом и коммутатором. Средняя задержка: 0.01-0.02 сек. Задержка в десяток раз меньше, чем в OnlyHub. Кривая более стабильная, без резких скачков.

OnlyHub – сеть только с хабами (концентраторами). Средняя задержка: 0.16-0.2 сек. Характерны резкие колебания задержки. Пиковые значения достигают 0.25 сек.

Технические причины различий:

- ❑ Природа работы хабов: Хабы работают на физическом уровне (Layer 1). Создают единый домен коллизий. При одновременной передаче данных происходят коллизии, требующие повторной отправки
- ❑ Принцип работы коммутаторов: Коммутаторы работают на канальном уровне (Layer 2). Создают отдельные домены коллизий для каждого порта. Используют MAC-адреса для адресной доставки пакетов. Поддерживают полнодуплексный режим передачи.

Добавление коммутатора улучшает задержку в 10 раз. Максимальная задержка снижается с 0.25 сек до 0.02 сек. Сеть с коммутатором демонстрирует более предсказуемую задержку. Отсутствуют резкие скачки, характерные для pure-hub сети (сценарий OnlyHub).

Для современных сетей хабы не рекомендуются. Даже частичное использование коммутаторов дает значительный выигрыш в производительности. В смешанных сетях критически важные узлы следует подключать к коммутатору.

6. Открыть иерархическое меню **Traffic Source**  $\Rightarrow$  выбрать *Traffic Sent (packets/sec)*  $\Rightarrow$  нажать **Show**.
7. Открыть иерархическое меню **Traffic Sink**  $\Rightarrow$  выбрать *Traffic Received (packets/sec)*  $\Rightarrow$  нажать **Show**.

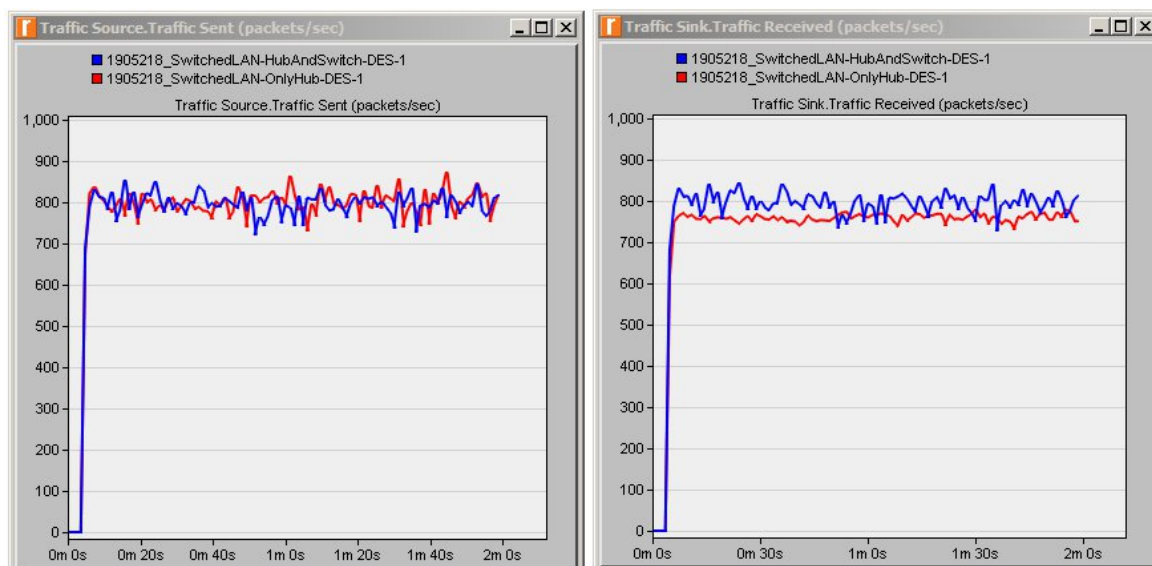


Рисунок 19. Графики метрик *Traffic Sent* и *Traffic Received* (packets/sec) для сценариев *OnlyHub* и *HubAndSwitch*

#### □ Анализ графика **Traffic Sent** (отправленных пакетов)

Оба сценария показывают схожие уровни генерации трафика (~800 packets/sec). Кривые схожи и стабильны во времени. Нет существенной разницы между pure-hub и hybrid-конфигурацией.

Источники трафика работают с максимальной производительностью в обоих случаях. Ограничения на отправку пакетов определяются настройками источников, а не топологией сети.

#### □ Анализ графика **Traffic Received** (полученных пакетов)

**OnlyHub** (только хабы): доставка с колебаниями, ~700 packets/sec. Средняя пропускная способность ниже, чем **Traffic Sent** (~800 packets/sec).

**HubAndSwitch** (хаб+коммутатор): Стабильная доставка на уровне ~800 packets/sec. Кривая почти идентична **Traffic Sent**.

Причины различий:

В сети **OnlyHub**: коллизии приводят к потере пакетов. Необходимость ретрансмиссий (повторная отправка пакета) снижает эффективную пропускную способность. Общий домен коллизий ограничивает производительность.

В сети **HubAndSwitch**: коммутатор изолирует домены коллизий. Полнодуплексная передача на портах коммутатора. Уменьшение коллизий и потерь пакетов

Таким образом, при использовании хабов фактическая пропускная способность сети составляет лишь ~80% от генерируемого трафика.

Добавление коммутатора позволяет реализовать полную пропускную способность источников трафика.

Pure-hub сеть демонстрирует непредсказуемую доставку пакетов. Гибридная конфигурация обеспечивает стабильную и предсказуемую работу

Данные графики наглядно демонстрируют, что ключевая проблема pure-hub сетей – не в генерации трафика, а в его эффективной доставке. Коммутаторы решают эту проблему, обеспечивая практически идеальную передачу данных даже в гибридных конфигурациях.

8. Откройте иерархическое меню **Object Statistics**  $\Rightarrow$  **Office Network**  $\Rightarrow$  **Hub1**  $\Rightarrow$  **Ethernet**  $\Rightarrow$  выбрать *Collision Count*  $\Rightarrow$  нажать **Show**.

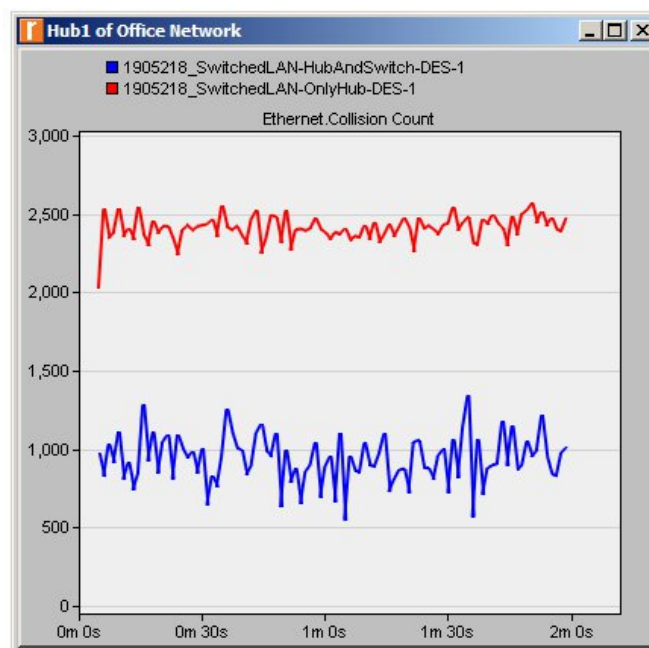


Рисунок 20. График для метрики Collision Count хаба Hub1, присутствующем в обоих сценариях OnlyHub и HubAndSwitch

Красная линия (OnlyHub) стабильно держится около 2,500 коллизий. Синяя линия (HubAndSwitch) колеблется около 1,000 коллизий. В гибридной сети (HubAndSwitch) коллизии уменьшаются, но не исчезают полностью.

В сценарии HubAndSwitch мы имеем коммутатор и два хаба, которые служат неким мостом между двумя группами узлов (node0-node7 и node8-node15), разделяя сеть на две равные части (т.к. генерируемый трафик от узлов исходит с одинаковыми параметрами), при этом метрика была снята только для Hub1, поэтому мы и получили уменьшение коллизий в сравнении со сценарием OnlyHub примерно в 2 раза, что, в принципе, и следовало ожидать. Все остальные коллизии возникают при передаче пакетов в группах между собой (между node0-node7, сконцентрированными у хаба Hub2).



График подтверждает, что даже частичное применение коммутаторов даёт значительное улучшение, но для кардинального решения проблемы требуется полный отказ от хабов в сети.

9. Щелкните ПКМ где-либо на открывшемся графике (но не на области вокруг)  $\Rightarrow$  выбрать **Add Statistic**  $\Rightarrow$  в открывшемся окне **Results Browser** открыть иерархические меню **Object Statistics**  $\Rightarrow$  **OfficeNetwork**  $\Rightarrow$  **Hub2**  $\Rightarrow$  **Ethernet**  $\Rightarrow$  выбрать *Collision Count*  $\Rightarrow$  нажать **Add**  $\Rightarrow$  закрыть окно **Results Browser**.

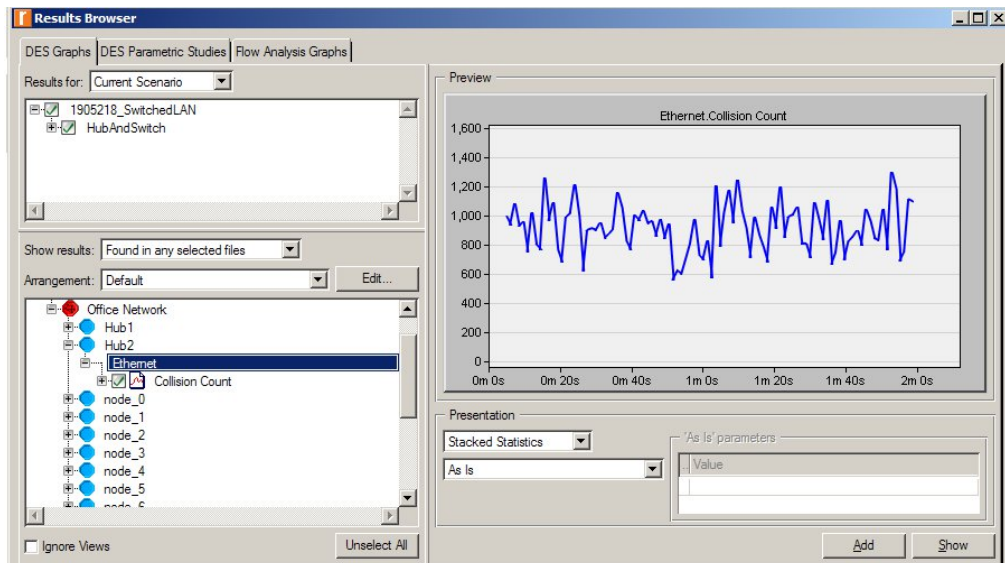


Рисунок 21. Добавление метрики Collision Count хаба Hub2 сценария HubAndSwitch

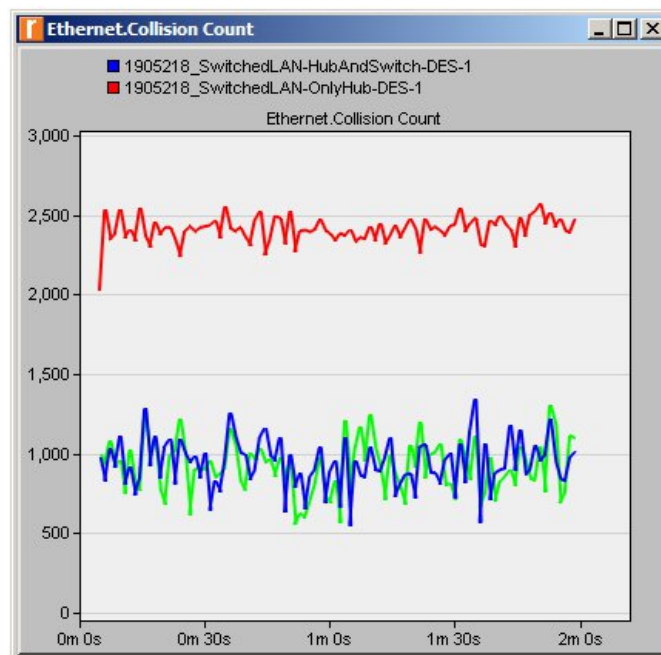


Рисунок 22. Получившийся в результате добавления график для метрики Collision Count хабов Hub1 и Hub2

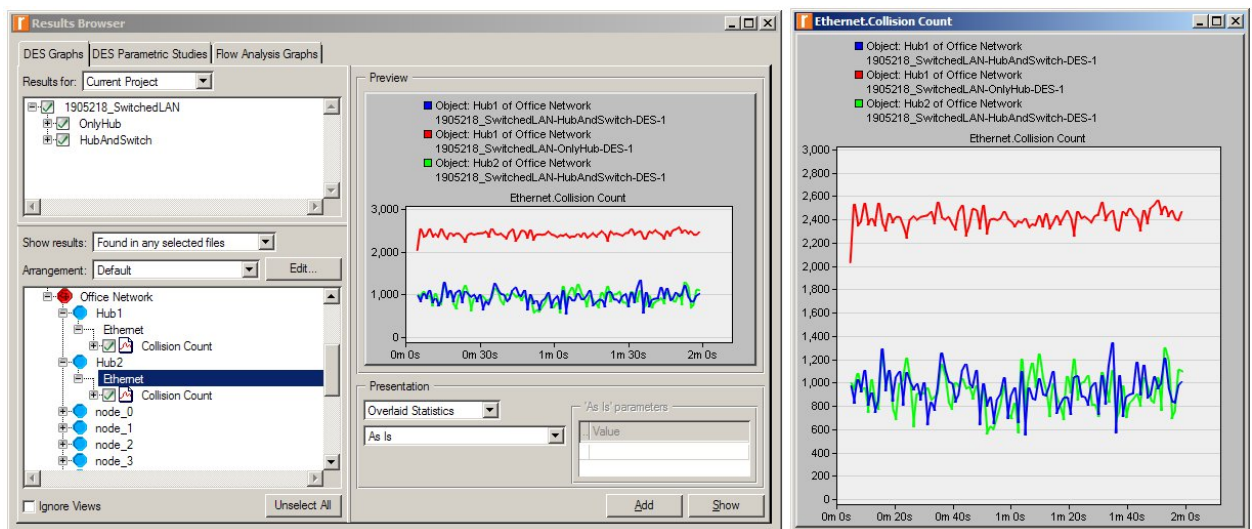


Рисунок 23. График с изначально выбранными метриками Collision Count хабов Hub1 и Hub2. В данном случае также указывается легенда для зеленой линии метрики Collision Count хаба Hub2

## Задания для самостоятельного выполнения

1. Проанализируйте полученные графики. Как и почему использование коммутатора влияет на работу сети (см. метрики **Traffic Received**, **Delay** и **Collision Count**)?

Коммутатор создает отдельные домены коллизий для каждого порта, изолируя трафик между подключенными устройствами. Коллизии сохраняются только между устройствами, оставшимися на хабе.

Снижение задержки (Delay) ввиду отсутствия повторных передач из-за коллизий.

Технические преимущества коммутатора:

- ☐ Сегментация домена коллизий. Каждый порт — отдельный домен (в хабе — один на все устройства).
- ☐ MAC-адресная маршрутизация. Передача пакетов только целевому устройству (хаб дублирует трафик на все порты).
- ☐ Полнодуплексный режим — одновременная передача/приём без коллизий (хабы работают в полудуплексе).

2. Статистика по числу коллизий была получена для хабов. Можно ли получить аналогичную статистику для коммутатора?

В сценарии HubAndSwitch коллизии фиксировались только на хабе (Hub1), так как коммутатор (Switch) физически не участвует в коллизиях, и, следовательно, устройства, подключённые к коммутатору, обмениваются данными без коллизий.

Ответ: Нет, классические коммутаторы (L2) не формируют статистику коллизий, так как принципиально исключают их возникновение в своей работе. Однако для диагностики можно анализировать смежные метрики, например, количества отброшенных кадров при попытке отправки.

3. Создайте копию сценария **OnlyHub** и озаглавьте ее как *OnlySwitch*. В созданной копии заменить хаб коммутатором.

- ☐ Чтобы произвести данную замену, щелкните ПКМ на узле *Hub1*  $\Rightarrow$  в появившемся меню выберите **Edit Attributes**. В открывшемся окне установите значение параметра **model** как *ethernet16\_switch*. В том же окне установите значение параметра **name** как *Switch*  $\Rightarrow$  **OK**.

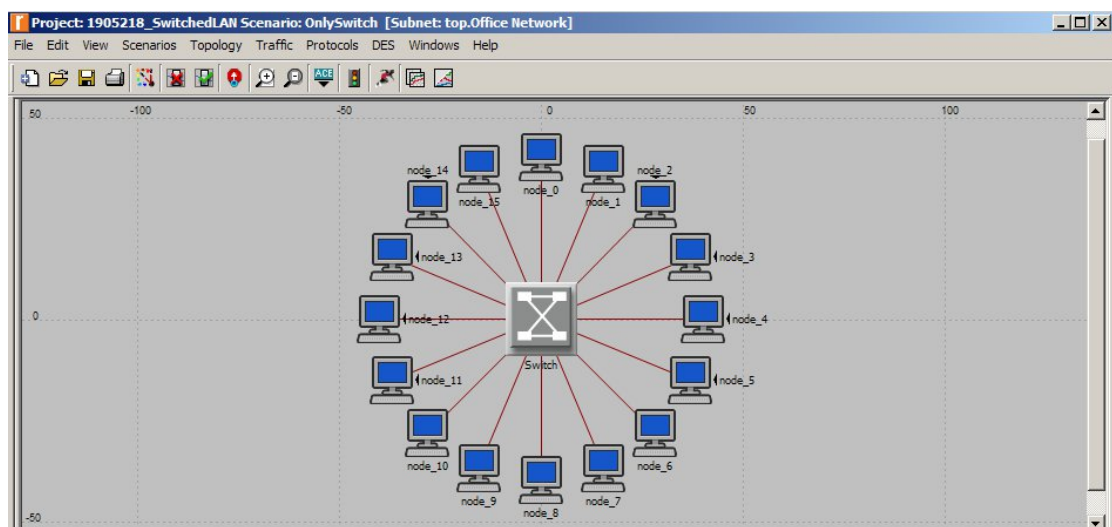


Рисунок 24. Получившаяся сеть после замены хаба коммутатором в сценарии *OnlySwitch*

4. Создать копию сценария **HubAndSwitch**, использовавшегося в данной лабораторной работе, и озаглавить ее как *SwitchAndSwitch*.

- ☐ В созданной копии удалить имеющийся коммутатор *Switch*, а хабы заменить коммутаторами, соединив их посредством **10BaseT**. Озаглавьте новые коммутаторы как *Switch1* и *Switch2*.
- 5. Запустить имитационное моделирование для сценариев *OnlySwitch* и *SwitchAndSwitch*.

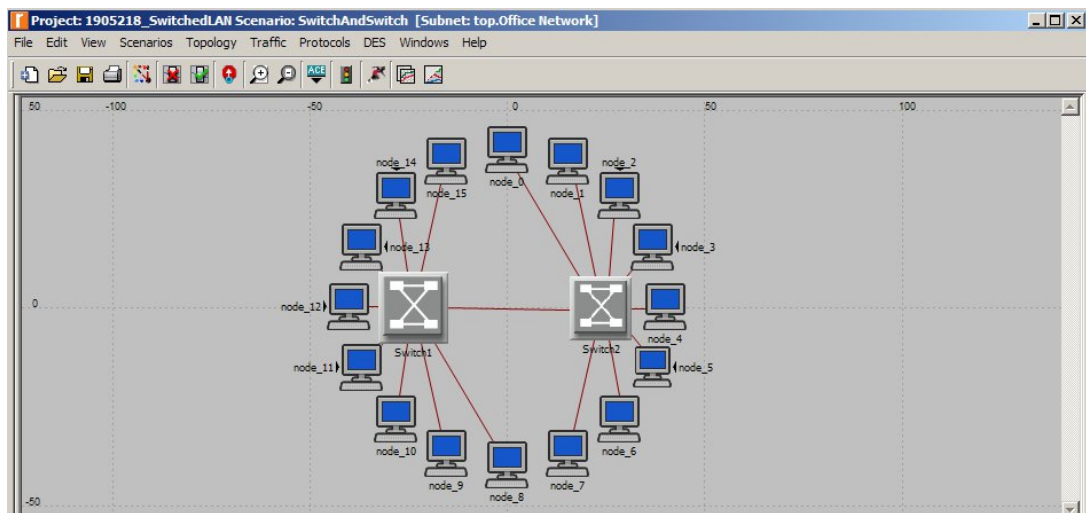


Рисунок 25. Получившаяся сеть с двумя коммутаторами в сценарии SwitchAndSwitch

6. Сравнить эти 2 сценария по метрикам **Delay**, **Traffic Sent** и **Traffic Received**.

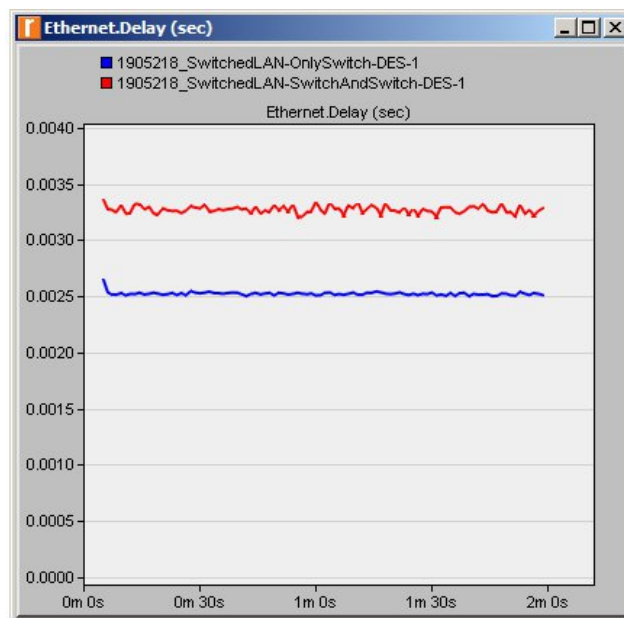


Рисунок 26. График для метрики Delay (sec) для сценариев OnlySwitch и SwitchAndSwitch

Оба сценария демонстрируют крайне низкие значения задержки (в пределах 0.0025-0.0035 сек). Сценарий OnlySwitch с одним коммутатором показывает незначительно меньшую задержку ( $\approx 0.0005$  сек разницы). Задержка остается стабильной на протяжении всего времени моделирования (2 минуты).

Добавление второго коммутатора незначительно улучшает задержку в сети. Обе конфигурации обеспечивают отличное время отклика. Для офисных сетей такого масштаба разница практически незаметна.

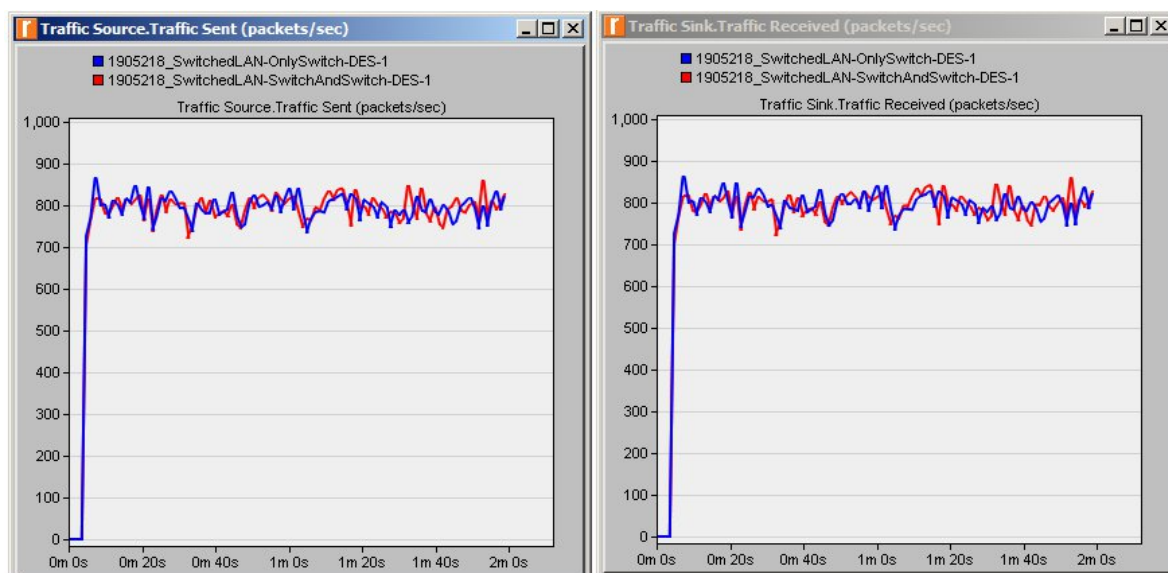


Рисунок 27. Графики метрик *Traffic Sent* и *Traffic Received* (packets/sec) для сценариев *OnlySwitch* и *SwitchAndSwitch*

#### □ Анализ графика отправленных пакетов (*Traffic Sent*, packets/sec)

Оба сценария показывают высокую и стабильную скорость отправки пакетов ( $\approx 700-900$  packets/sec). Нет существенной разницы между конфигурациями. Небольшие колебания ( $\approx 50$  packets/sec) носят случайный характер.

Количество коммутаторов не влияет на способность узлов генерировать трафик. Ограничения по генерации трафика определяются настройками источников, а не топологией. Сети не перегружены в обоих случаях.

#### □ Анализ графика полученных пакетов (*Traffic Received*, packets/sec)

Значения практически идентичны показателям *Traffic Sent* ( $\approx 700-900$  packets/sec). Минимальные потери пакетов (разница между sent/received достаточно мала). *SwitchAndSwitch* показывает чуть более стабильную доставку.

Обе конфигурации обеспечивают эффективную доставку трафика. Добавление второго коммутатора незначительно улучшает надежность доставки. Сети работают без перегрузок и потерь пакетов.

Для текущего масштаба сети (офис) одного коммутатора достаточно. Добавление второго коммутатора не дает существенных преимуществ.

В небольших сетях (до 16 узлов) экономически целесообразнее использовать один коммутатор. Два коммутатора могут быть полезны для:



резервирования (повышение отказоустойчивости), подготовки к расширению сети и для сегментации трафика (если узлы логически разделены). Для сетей с особыми требованиями к надежности стоит рассмотреть избыточную конфигурацию.

Все графики подтверждают, что для данного масштаба сети оба варианта конфигурации являются работоспособными и эффективными. Выбор между ними должен основываться на требованиях к отказоустойчивости и планах по расширению сети.

## Выводы

В рамках аналитического практикума была создана сеть с 16 узлами и хабом (Hub1) с использованием метода Rapid Configuration для быстрой настройки. Были настроены параметры генерации трафика и сбора статистики.

Также были созданы дополнительные сценарии: "HubAndSwitch", "OnlySwitch" и "SwitchAndSwitch". Проведено сравнение работы сети с хабами и коммутаторами.

Сети на хабах демонстрируют:

- ☐ Высокий уровень коллизий (~2500 за 2 мин, ≈21 кол/сек)
- ☐ Большие задержки (0.2–0.25 сек)
- ☐ Потери пакетов

Сети с коммутаторами (даже гибридные):

- ☐ Снижают коллизии
- ☐ Уменьшают задержку в 3–7 раз (до 0.0025–0.0035 сек)
- ☐ Обеспечивают лучшую доставку пакетов

**Технические причины различий.** Хабы (концентраторы) работают в полудуплексном режиме с общим доменом коллизий, используют алгоритм CSMA/CD, что приводит к ретрансмиссиям и потерям. Коммутаторы создают отдельные домены коллизий для каждого порта, поддерживают полнодуплексную передачу без коллизий и буферизуют и адресно направляют трафик (по MAC-адресам).

Интересно отметить, что разница между сетями с хабами (OnlyHub vs HubAndSwitch) коммутаторами (в 10 раз) значительно больше, чем разница между разными конфигурациями коммутаторов (OnlySwitch vs SwitchAndSwitch). Это подтверждает, что переход от hub- к switch-технологии дает наибольший выигрыш в производительности.

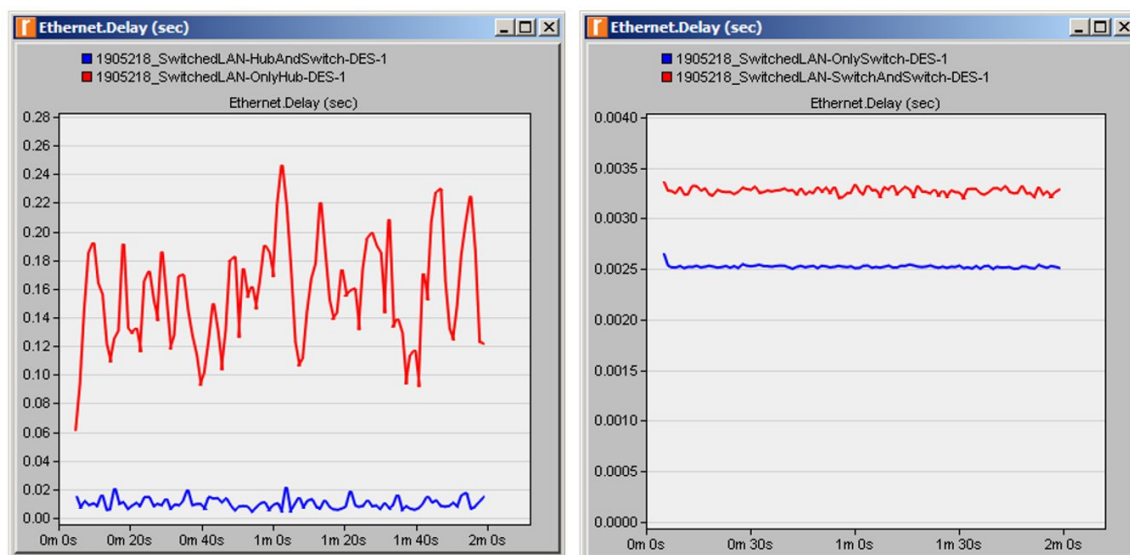


Рисунок 28. Графики для метрики Delay (sec) для сценариев OnlyHub и HubAndSwitch (слева) и для сценариев OnlySwitch и SwitchAndSwitch (справа)

Данный график наглядно демонстрирует фундаментальные преимущества коммутируемых сетей перед сетями на концентраторах.

В конечном итоге, при рассмотрении сценариев с различными конфигурациями и их сравнении были сформированы заключительные выводы, такие, что:

- ☐ Для повышения производительности сети рекомендуется использовать коммутаторы вместо хабов
- ☐ В крупных сетях целесообразно использовать несколько коммутаторов для распределения нагрузки
- ☐ При проектировании сети важно учитывать метрики задержки, количества коллизий и пропускной способности

Работа успешно демонстрирует преимущества коммутируемых сетей перед сетями на хабах и важность правильного выбора сетевого оборудования при проектировании локальных сетей.