

# ЛАБОРАТОРНАЯ РАБОТА 8

---

## АСЕ и анализ работы приложений

---

### Цель

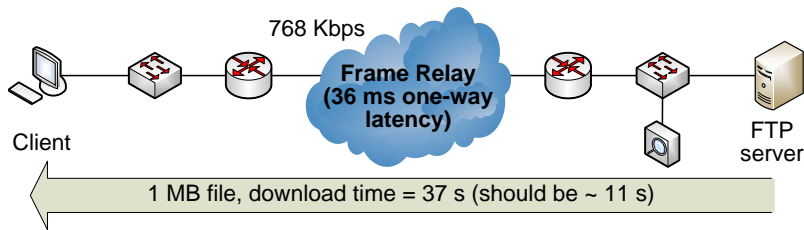
- Знакомство с OPNET Application Characterization Environment (ACE)
- Овладение навыками поиска 'узких мест' в работе приложения
- Анализ влияния характеристик сети на работу приложения

### Методические указания

#### Файл трассировки и AppDoctor:

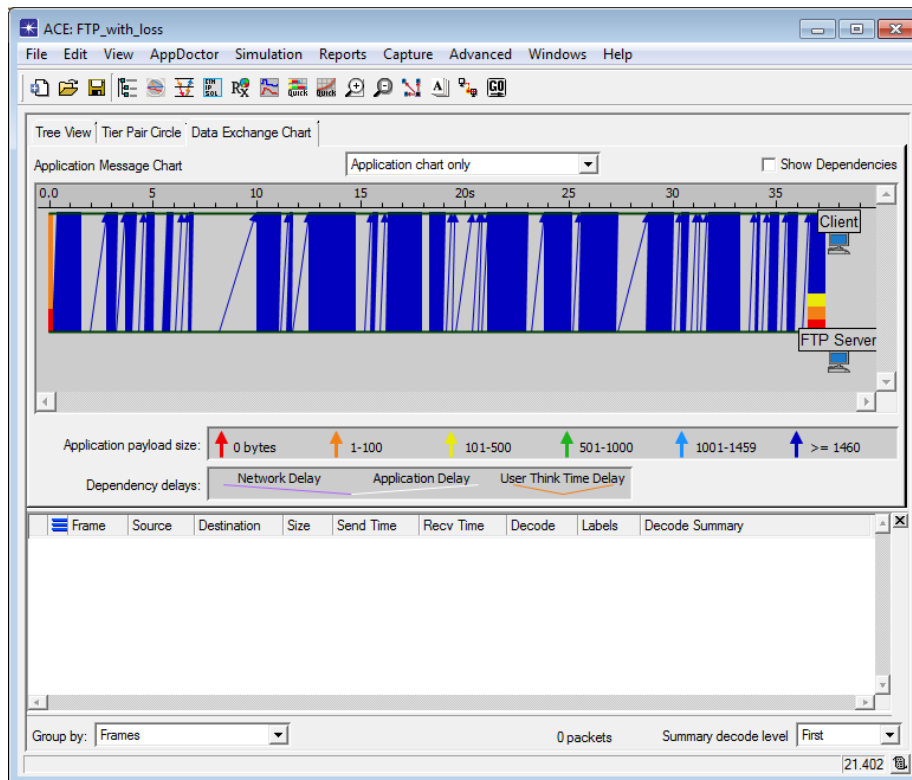
1. Запустите **OPNET Modeler 14.5**  $\Rightarrow$  в меню **File** выберите **Open...**
2. В выпадающем меню **Files of type** выберите **ACE Files (\*.atc.m)**  
 $\Rightarrow$  в выпадающем меню **Look in** перейдите в  
**C:/Program Files/OPNET/14.5.A/models/std/tutorial\_req/module**  
 $\Rightarrow$  выберите **FTP\_with\_loss.atc**  $\Rightarrow$  нажмите **Open**
  - Файл трассировки **FTP\_with\_loss** был получен на реальной сети, аналогичной нижепредставленной
  - Передача файла объемом 1 МБ по протоколу FTP осуществляется от сервера к клиенту, скорость доступа которого составляет 768 кбит/с, задержка на участке сети Frame Relay равна 36 мс
  - Загрузка файла занимает 37 секунд вместо расчетных 11 ( $8 * 1024 * 1024 / 768 * 1024 \approx 10.7$  с)

- Задача: найти причину большой задержки и исследовать влияние различных сетевых характеристик (скорость передачи данных, потери пакетов) на работу приложения

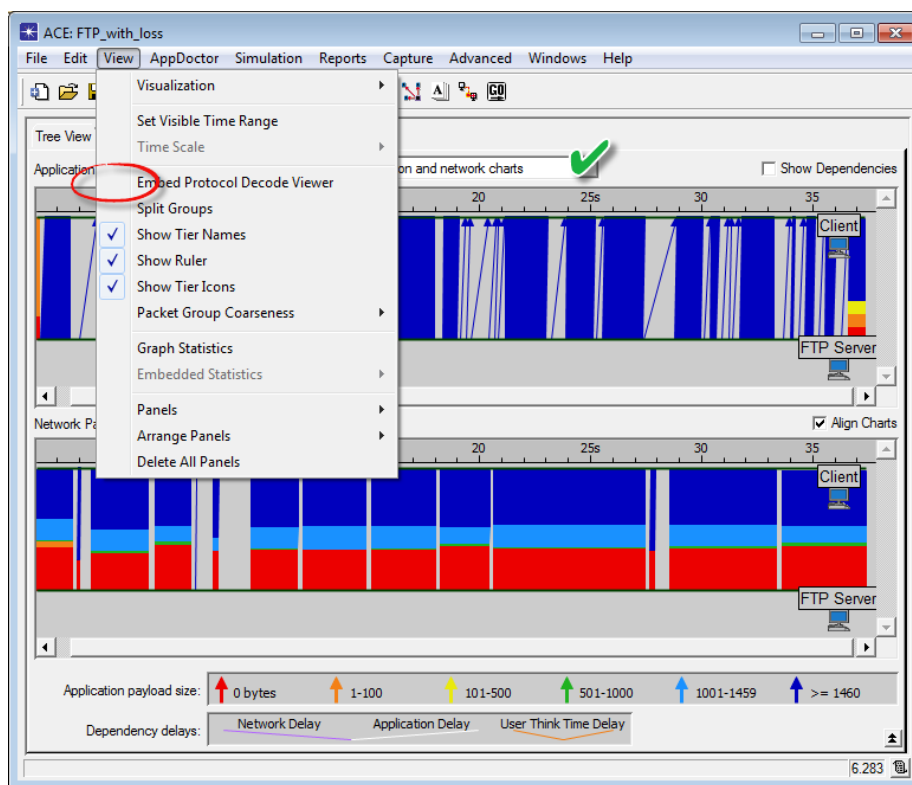


### 3. В открывшемся окне **Getting Started Analyzing an Application Transaction** выберите **Data Exchange Chart**

- В результате в окне **ACE** будет в графическом виде представлен обмен пакетами между FTP сервером и клиентом

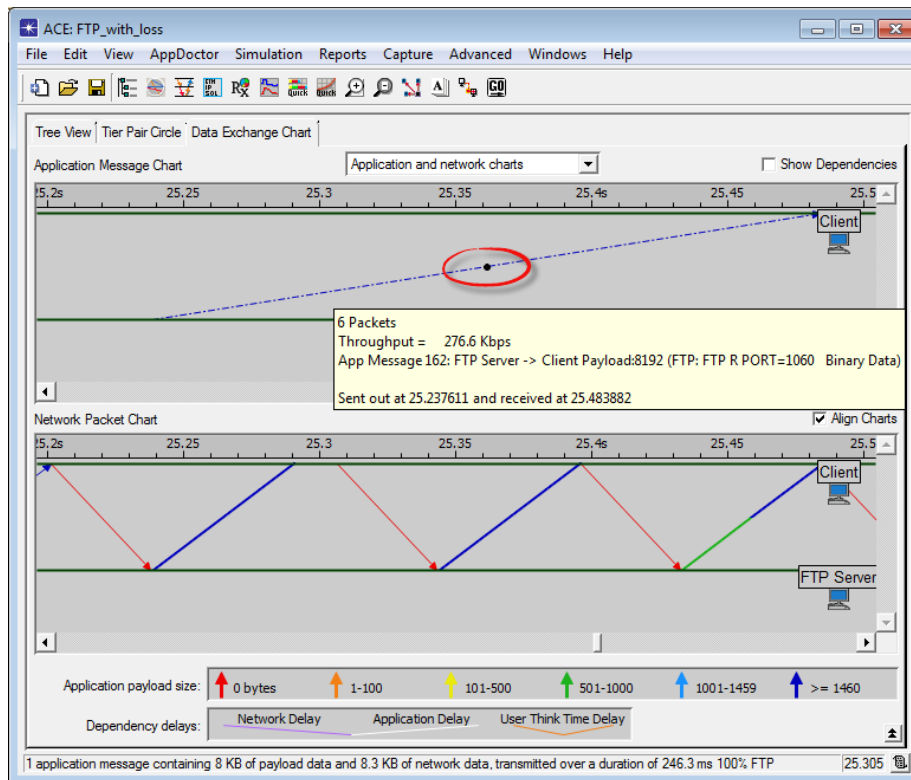


4. В меню **View** уберите галочку напротив **Embed Protocol Decode Viewer**
5. В выпадающем меню выберите **Application and network charts**
  - Обратите внимание, что **Application chart only** показывает лишь обмен данными прикладного уровня между клиентом и сервером
  - В свою очередь, **Network chart only**, помимо данных прикладного уровня, также отображает передачу служебной информации протоколов нижних уровней
  - Эти протоколы осуществляют сегментацию передаваемых данных, добавляют заголовки, осуществляют квитиование для защиты от ошибок и т.п.
  - Все это может оказывать влияние на качество предоставления услуги



6. Чтобы увидеть передачу отдельных сообщений, в меню **View** выберите **Set Visible Time Range** ⇒ установите значение параметра **Start Time** равным **25.2**, а значение параметра **End Time** – равным **25.5** ⇒ нажмите **OK**

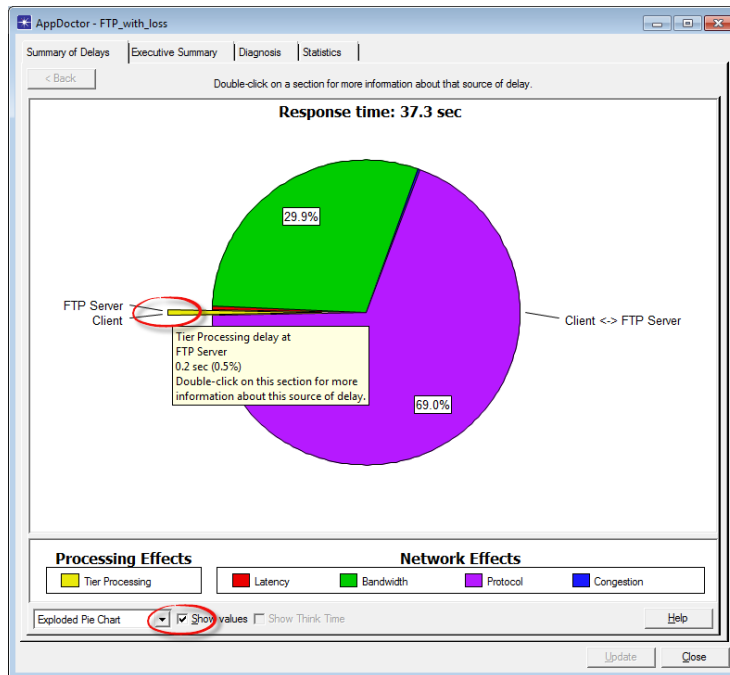
- В результате на **Application Message Chart** будет представлена передача одного сообщения от FTP сервера к клиенту
- Чтобы увидеть объем переданных данных прикладного уровня, наведите курсор на соответствующую стрелку (FTP Server -> Client Payload показан равным 8192 байт)
- Согласно **Network Packet Chart**, в ходе пересылки этих данных был передан ряд пакетов: больших (обозначены синим и зеленым цветами) от сервера к клиенту и маленьких (обозначены красным цветом) от клиента к серверу
- Эти маленькие пакеты не содержат данных прикладного уровня (Application payload size равен 0 байт) и являются ACK-сегментами протокола TCP



7. Чтобы увидеть структурный состав задержки доставки данных, в меню **AppDoctor** выберите **Summary of Delays (AppDoctor Analysis)** ⇒ в открывшемся окне поставьте галочку в поле **Show Values**
  - Обратите внимание, что источником основной составляющей задержки является работа протокола(ов) (69.0%)

- Низкая скорость передачи данных (768 кбит/с по условию задачи) находится на втором месте по значимости в общей структуре задержки (29.9%)
- Задержка, вносимая работой самого FTP сервера, в данном случае крайне мала (0.5%)

8. Закройте окно **AppDoctor**



9. Для более детального анализа в меню **AppDoctor** выберите **Diagnosis**

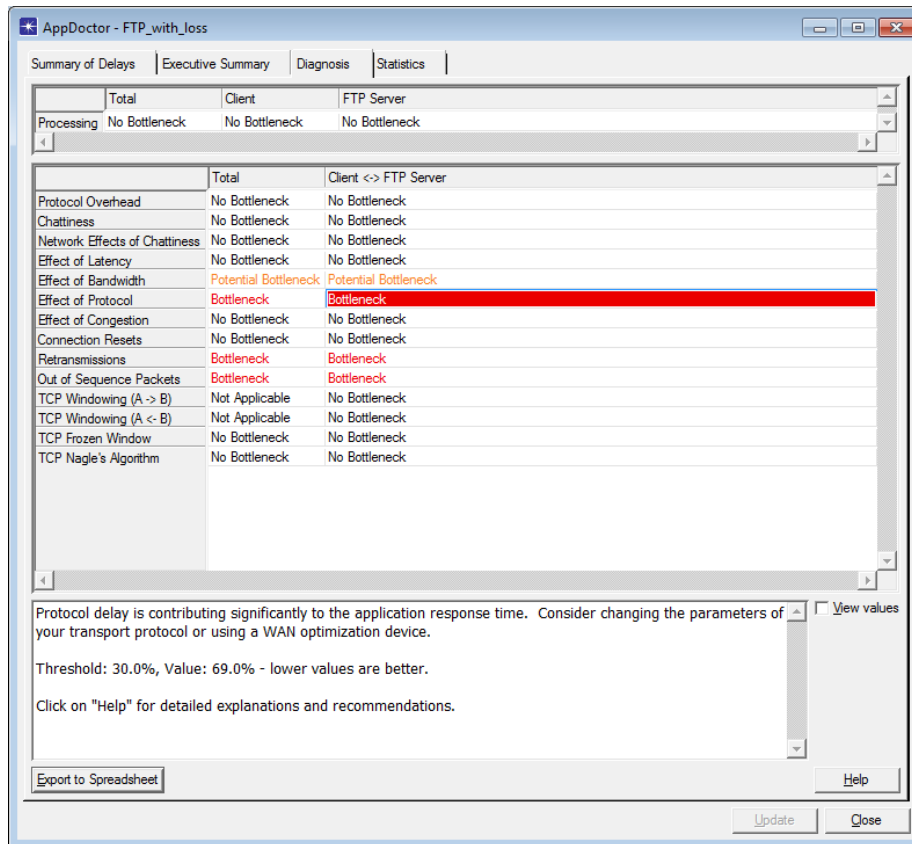
- Красным цветом выделены 'узкие места' (bottlenecks)
- Ниже в окне приведены рекомендации по их устранению и дополнительные пояснения

10. Закройте окно **AppDoctor**

11. Чтобы посмотреть статистику по файлу трассировки, в меню **AppDoctor** выберите **Statistics**

- Обратите внимание, что из 1281 пакетов (Network Packets) 52 пакета (Retransmissions) были переданы повторно, т.е. доля повторных передач составляет примерно 4%

12. Закройте окно **AppDoctor**



13. Чтобы посмотреть статистику в графическом виде, в меню **View** выберите **Graph Statistics**
14. В открывшемся окне выберите **2 метрики**: **Network Throughput (Kbits/sec): Client to FTP Server** и **Network Throughput (Kbits/sec): FTP Server to Client** ⇒ нажмите **Show**
15. Затем выберите **другие 2 метрики**: **Retransmissions: Client to FTP Server** и **Retransmissions: FTP Server to Client** ⇒ установите значение параметра **Bucket width (msec)** равным **100** ⇒ нажмите **Show**

### Определение оптимального размера окна приемника TCP:

1. В окне **Graph Statistics** выберите новую метрику: **TCP In-Flight Data (bytes) FTP Server to Client** ⇒ установите значение параметра **Bucket width (msec)** равным **1000** ⇒ нажмите **Show**

AppDoctor - FTP\_with\_loss

Summary of Delays | Executive Summary | Diagnosis | Statistics

	Total	Client	FTP Server
User Think Time (sec)	0.000000	0.000000	Not Applicable
Effect of Processing (sec)	0.181852	0.000119	0.181733
Effect of Network (sec)	37.098266	Not Applicable	Not Applicable
Parallel Effects (sec)	0.000000	Not Applicable	Not Applicable

	Total	Client <-> FTP Server
Duration (sec)	37.280119	37.280119
Response Time (sec)	37.280119	37.280119
Application Turns	4	4
Application Messages	241	241
Application Data (bytes)	1,057,043	1,057,043
Average Application Message (bytes)	4,386.07	4,386.07
Network Packets	1,281	1,281
Network Data (bytes)	1,201,409	1,201,409
Average Network Packet (bytes)	937.87	937.87
Latency (ms)	Not Applicable	36.00
Effect of Latency (sec)	0.144000	0.144000
Bandwidth (Kbps)	Not Applicable	768.000
Effect of Bandwidth (sec)	11.146865	11.146865
Effect of Protocol (sec)	25.739874	25.739874
Effect of Congestion (sec)	0.067527	0.067527
Max Application Bytes Per Turn (A -> B)	Not Applicable	23
Max Application Bytes Per Turn (A <- B)	Not Applicable	1,056,891
Max Unacknowledged Data (A -> B) (bytes)	Not Applicable	10
Max Unacknowledged Data (A <- B) (bytes)	Not Applicable	8,192
Retransmissions	52	52
Out of Sequence Packets	41	41
Connection Resets	0	0
TCP Frozen Window (sec)	0.000000	0.000000
TCP Nagle's Algorithm (sec)	0.000000	0.000000
TCP Triple-Duplicate ACK Loss Indications	34	34

Export to Spreadsheet | Help | Update | Close

Graph Statistics - FTP\_with\_loss

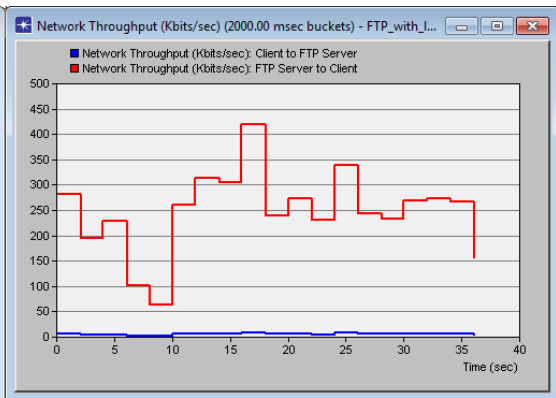
Tier Pair 1: Client <-> FTP Server

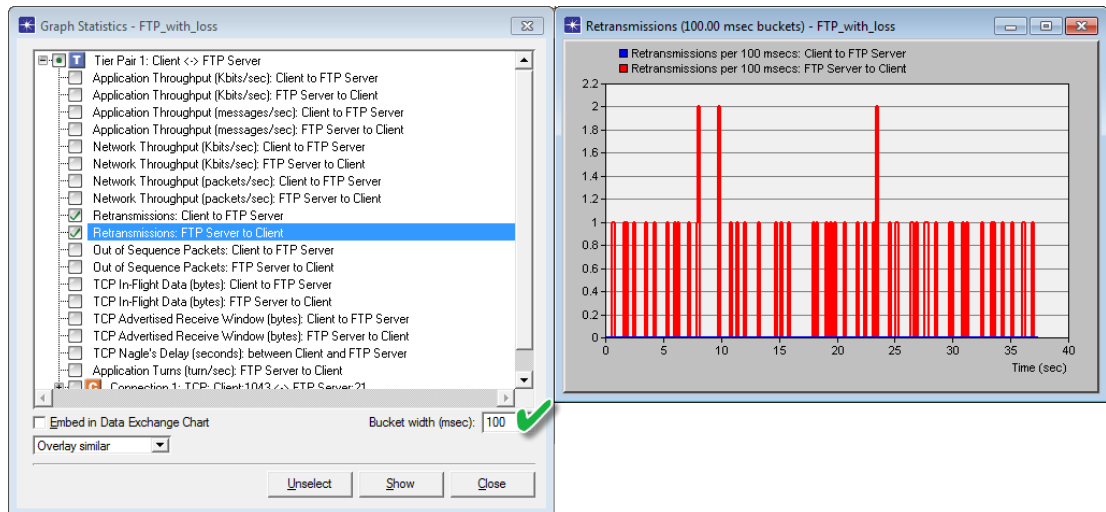
- Application Throughput (Kbits/sec): Client to FTP Server
- Application Throughput (Kbits/sec): FTP Server to Client
- Application Throughput (messages/sec): Client to FTP Server
- Application Throughput (messages/sec): FTP Server to Client
- Network Throughput (Kbits/sec): Client to FTP Server
- Network Throughput (Kbits/sec): FTP Server to Client
- Network Throughput (packets/sec): Client to FTP Server
- Network Throughput (packets/sec): FTP Server to Client
- Retransmissions: Client to FTP Server
- Retransmissions: FTP Server to Client
- Out of Sequence Packets: Client to FTP Server
- Out of Sequence Packets: FTP Server to Client
- TCP In-Flight Data (bytes): Client to FTP Server
- TCP In-Flight Data (bytes): FTP Server to Client
- TCP Advertised Receive Window (bytes): Client to FTP Server
- TCP Advertised Receive Window (bytes): FTP Server to Client
- TCP Nagle's Delay (seconds): between Client and FTP Server
- Application Turns (turn/sec): FTP Server to Client
- Connection 1: TCP: Client:1043 <-> FTP Server:21

Embed in Data Exchange Chart | Bucket width (msec): 2000

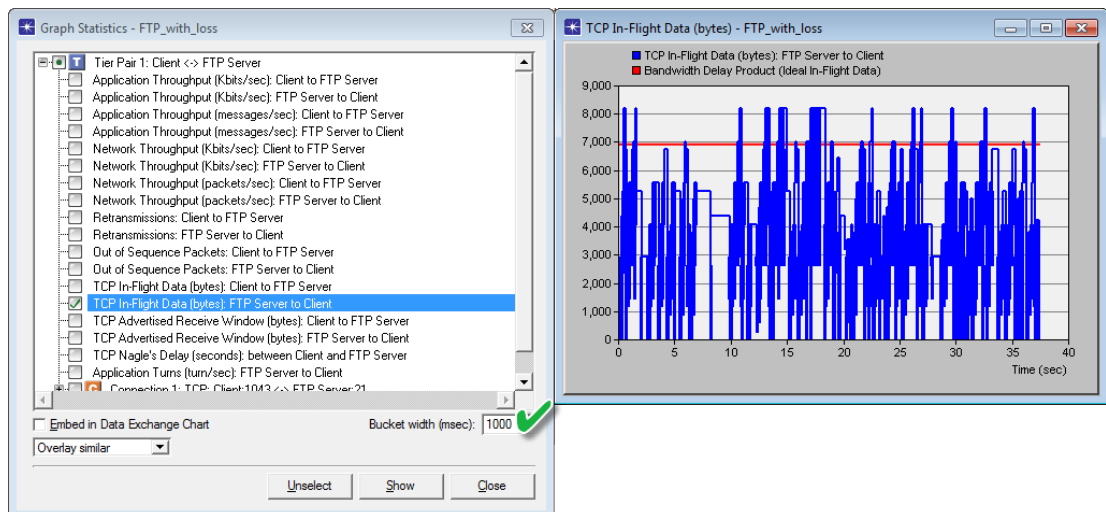
Overlay similar

Unselect | Show | Close





2. Согласно расчетам программы, оптимальный размер окна приемника TCP примерно равен 7,000 байт (красная линия **Bandwidth Delay Product (Ideal In-Flight Data)** на графике)



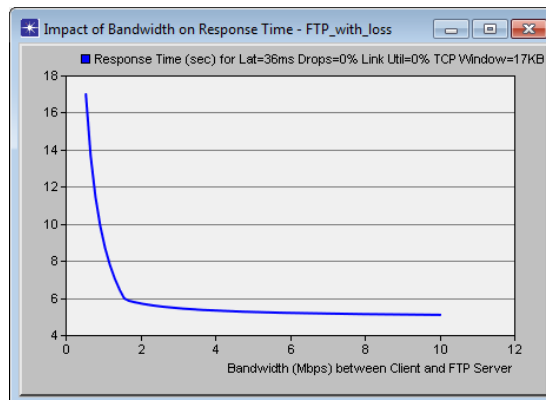
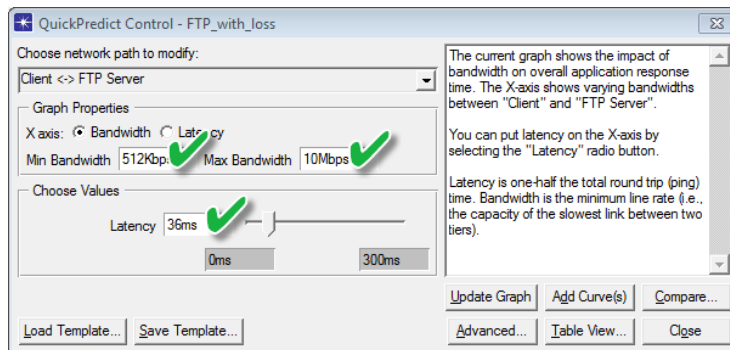
### Анализ влияния пропускной способности сети:

1. Чтобы рассмотреть, как пропускная способность сети влияет на качество обслуживания, в меню **Simulation** выберите **QuickPredict**
2. В окне **QuickPredict Control** установите значение параметра **Latency**



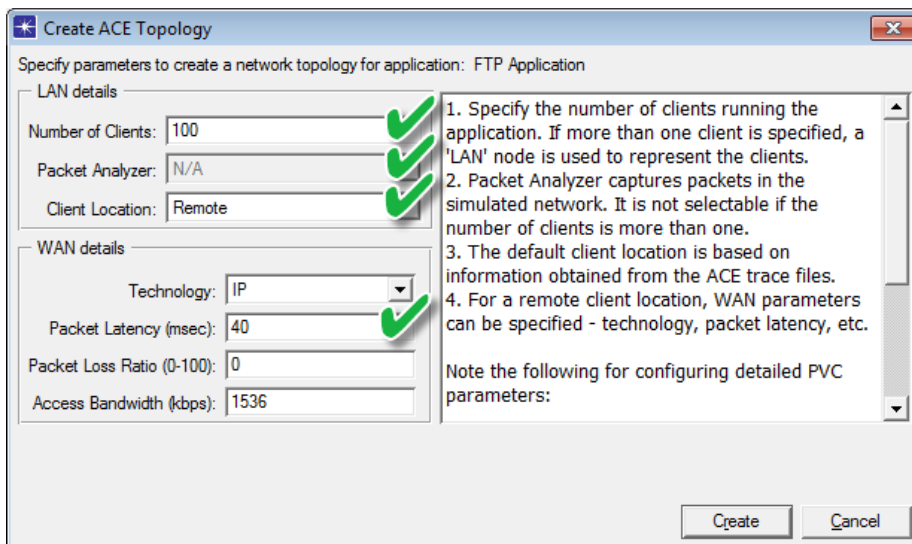
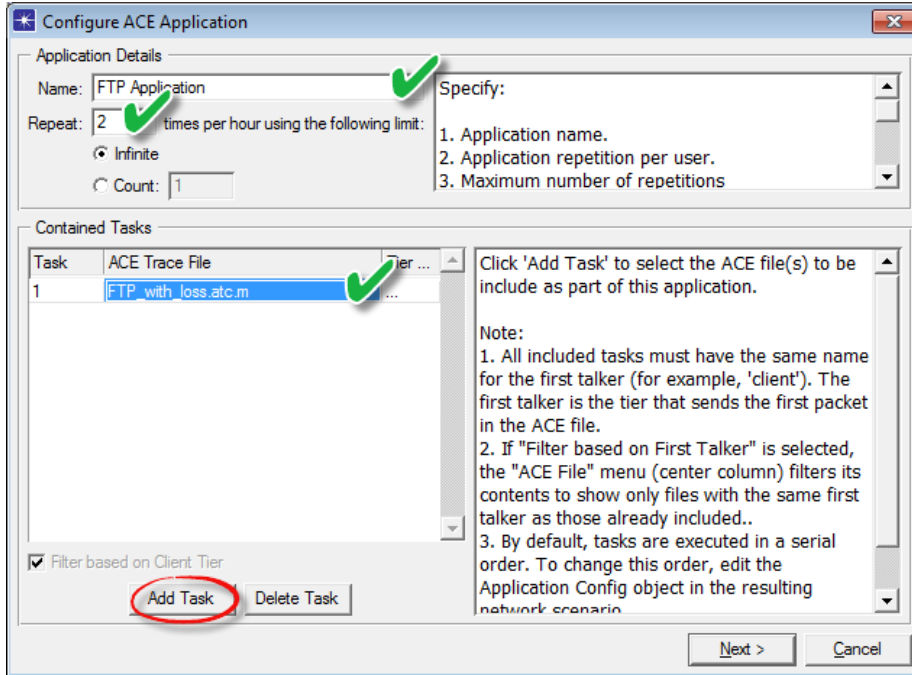
равным **36ms**, а значения параметров **Min Bandwidth** и **Max Bandwidth** равными **512Kbps** и **10Mbps**, соответственно  $\implies$  нажмите **Update Graph**

- Получившийся в результате график должен быть аналогичен нижеприведенному
- Закройте окно с графиком, окно **QuickPredict Control** и окно **ACE**



### Предиктивный анализ:

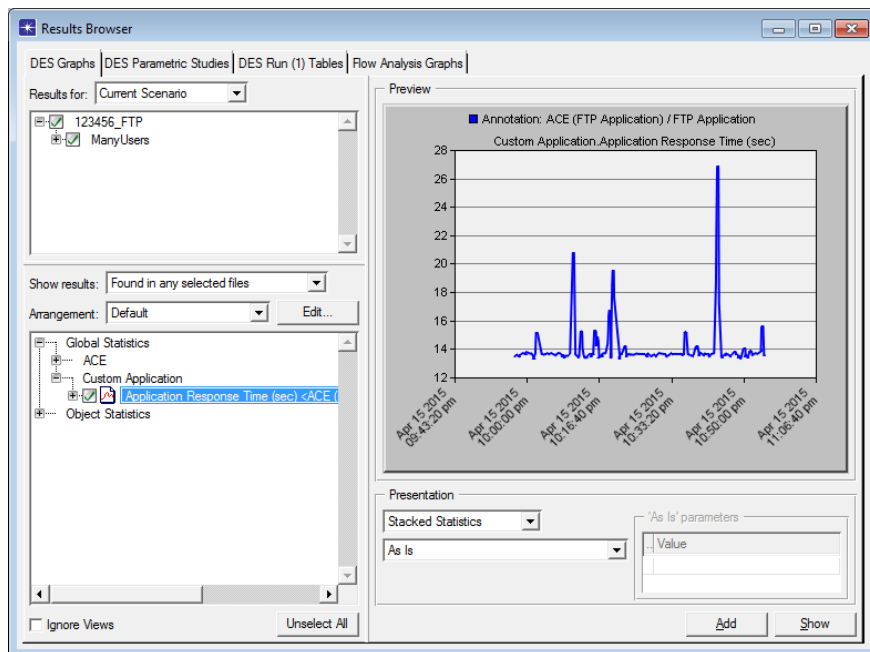
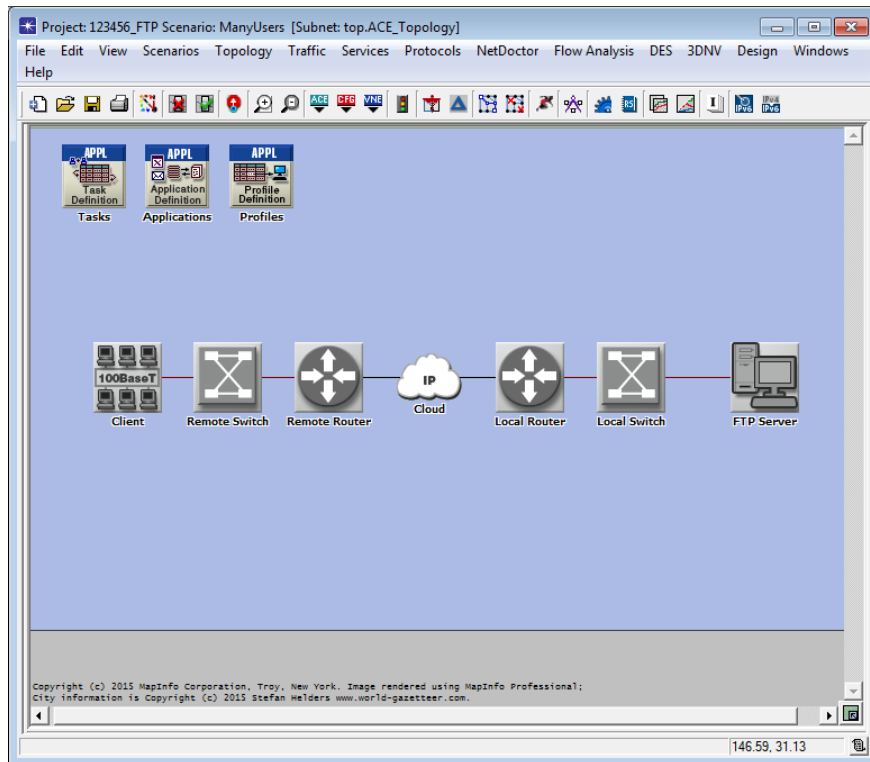
- OPNET Application Characterization Environment (ACE) позволяет провести предиктивный анализ и ответить, например, на вопрос о работе приложения FTP при одновременном обращении к нему 100 пользователей по IP-сети
- Для этого в окне **OPNET Modeler 14.5** в меню **File** выберите **New...**  
 $\implies$  выберите **Project**  $\implies$  нажмите **OK**  $\implies$  озаглавьте проект как **номер\_вашего\_студенческого\_FTP**, а сценарий – как **ManyUsers**  
 $\implies$  нажмите **OK**



3. В окне **Startup Wizard: Initial Topology** выберите **Import from ACE**  $\implies$  нажмите **Next**
4. В открывшемся окне **Configure ACE Application** установите следующие **3 значения**  $\implies$  нажмите **Next**
5. В открывшемся окне **Create ACE Topology** установите следующие **4 значения**  $\implies$  нажмите **Create**  $\implies$  сохраните проект
6. В результате будет создана топология, аналогичная нижеприведенной
  - Объекты **Tasks**, **Applications** и **Profiles** окажутся сконфигурированы согласно файлу трассировки и установленным ранее значениям параметров
7. На панели инструментов нажмите кнопку **Configure/Run Discrete Event Simulation (DES)**
8. Запустите имитационное моделирование, используя настройки по умолчанию
9. По окончании сохраните проект
10. В меню **DES** выберите **Results**  $\implies$  выберите **View Results...**  $\implies$  откройте иерархическое меню **Global Statistics** и **Custom Application**  $\implies$  выберите **Application Response Time (sec)**
11. Получившийся в результате график должны быть аналогичен нижеприведенному

## Задания для самостоятельного выполнения

1. Объясните, почему пакеты, передаваемые от клиента к FTP серверу, имеют, в основном, малый размер?
2. Используя данные из окна **AppDoctor Summary of Delays**, объясните, какое влияние на время загрузки по протоколу FTP окажет:
  - Апгрейд сервера
  - Апгрейд сети
  - Апгрейд протокола(ов)
3. Объясните, как повторные передачи пакетов влияют на время передачи файла по протоколу FTP?
4. Какой из данных протоколов отвечает за повторную передачу пакетов: IP, TCP или FTP?



5. На ранее полученном графике **Network Throughput (Kbits/sec)**, скорость передачи от FTP сервера к клиенту составляет в среднем около 250 кбит/с с пиком до 400 кбит/с. В то же время номинальная скорость равна 768 кбит/с. Объясните, почему фактическая скорость передачи данных оказывается ниже номинальной?
6. Определение оптимального размера окна приемника TCP в байтах осуществляется по формуле:

$$\frac{2 \times \text{Propagation Delay (seconds)} \times \text{Transmission Speed (bits/second)}}{8 \text{ (bits/byte)}}$$

Используя данные из окна **AppDoctor Statistics**, проверьте расчетное значение на графике **TCP In-Flight Data (bytes)**, примерно равное 7,000 байт

7. Проанализируйте ранее полученный график **Impact of Bandwidth on Response Time**. Почему, начиная с какого-то момента, увеличение пропускной способности сети не сокращает время отклика приложения?

## К защите

1. По результатам работы представить отчет, содержащий:
  - Графики, полученные в ходе имитационного моделирования
  - Выводы по результатам моделирования
2. Знать основы построения инфокоммуникационных систем и сетей