

Laboratorium 3

Temat: Badanie wydajności ruchu grupowego: różne struktury, różne metody przełączania multicastowego, oraz metody routingu multicastowego.

SPIS TREŚCI

1. Wstęp.....	3
1.1. Co to jest transmisja multicast?	3
1.2. IP Multicast.....	3
1.3. Protokół IGMP	5
1.4. Protokoły routingu multicast	5
Protokół PIM	5
Protokół DVMRP	6
Protokół MOSPF	7
2. OPnet.	8
2.1. OPnet - opis programu.....	8
2.2. Przygotowanie programu do pracy.....	8
2.3. Budowa sieci opartej na magistrali	12
2.4. Budowanie sieci opartej na topologii drzewa.	18

1. Wstęp . Pojęcie multicast oraz elementy związane z tym sposobem transmisji danych.

1.1. Co to jest transmisja multicast?

Multicast to sposób dystrybucji informacji, w którym istnieje więcej niż jeden odbiorca. Multicast różni się od unicast zasadą działania i wynikającą stąd efektywnością. W transmisji multicastowej po każdym łączu sieciowym dystrybuowana informacja jest przekazywana jednokrotnie, podczas gdy w unicastowej dystrybucji informacji do n odbiorców po niektórych łączach biorących udział w transmisji komunikat może być w najgorszym razie przesyłany nawet n razy. Wynika to z tego, że w transmisji unicastowej każdy komunikat ma przypisany dokładnie jeden adres docelowy, który identyfikuje jednoznacznie jeden węzeł sieci. Tak więc nawet jeśli dany komunikat po drodze do dwóch różnych węzłów sieci wykorzystuje to samo łącze, wysłane muszą zostać dwa niezależne komunikaty (o tej samej treści i innym adresie docelowym). W transmisji multicastowej unika się wielokrotnego wysyłania tego samego komunikatu do wielu nadawców (na przykład dzięki adresowaniu grupowemu, tzn. posługiwaniu się adresami, które nie identyfikują pojedynczych węzłów sieci, lecz ich grupy).

Największe oszczędności łącza multicast oferuje tam gdzie rozmiary komunikatów są największe, czyli na przykład w transmisjach telekonferencyjnych.

1.2. IP Multicast.

IP Multicast jest metodą transmisji typu multicast w sieciach opartych na protokole IP. Korzysta z zarezerwowanej do tego celu klasy adresów "D" - adresy 224.0.0.0 do 239.255.255.255 (adresy 224.0.0.0 do 224.0.0.255 zarezerwowane są na wewnętrzne funkcje protokołu). Komputery nieobsługujące multicasu powinny ignorować te pakiety, natomiast routery nie multicastowe nie powinny przekazywać ich, i zasadniczo taka jest zasada w działaniu tego protokołu dzisiaj. Adresy te mają inne znaczenie niż w przypadku unicastu - nie identyfikują one żadnego konkretnego hosta, wyznaczają natomiast swego

rodzaju grupę hostów a właściwie (jako że może to być grupa zmienna w czasie) swego rodzaju odpowiednik kanału telewizyjnego, na którym coś jest nadawane (i kto chce ten odbiera). Odbiorca musi jedynie dostroić się do częstotliwości reprezentowanej przez te adresy, czyli przyłączyć swój komputer tak, by mógł być identyfikowany przez adres IP mieszczący się w zakresie danej grupy. Również pole TTL (time to live) ma tutaj swego rodzaju specjalne znaczenie (z wykorzystaniem tego pola tworzy się regiony, np. poza sieć lokalną i okolice nie powinien wyjść pakiet o $TTL < 32$, poza miasto o $TTL < 64$ a poza Europę o $TTL < 128$).

Poważną przeszkodą - uniemożliwiającą powszechne korzystanie z zalet multicastingu jest brak obsługi adresów klasy D przez routery sieciowe oraz starsze oprogramowanie TCP/IP. Praktyczne jego wykorzystanie ogranicza się obecnie do pojedynczych sieci lokalnych lub grup takich sieci komunikujących się ze sobą za pośrednictwem routerów obsługujących tryb multicast.

Aby obejść obecne ograniczenia niekiedy korzysta się z rozwiązań pośrednich budując w Internecie tzw. tunele (połączenia wewnątrz połączeń - między komputerami wykorzystującymi odpowiednie oprogramowanie). Ich zadaniem jest opakowanie pakietów multicast w zwykłe pakiety unicast i dostarczenie ich do miejsca, gdzie po rozpakowaniu mogą być dalej rozpowszechniane jako pakiety multicast. Metoda tuneli wykorzystana została do zbudowania w Internecie wirtualnej sieci MBONE, pozwalającej na przeprowadzanie transmisji multicastowych o zasięgu globalnym. Router obsługujący Multicast gromadzi informacje od hostów (w tym innych routerów), do których przekazuje ruch o tym, jakimi multicastowymi numerami IP są zainteresowane (informacja taka jest wysyłana na multicastowy adres rozgłoszeniowy w momencie wyrażenia chęci rozpoczęcia pobierania jakiegoś "kanału", router dokonuje też odpytywania, korzystając z tego samego multicastowego adresu rozgłoszeniowego - 224.0.0.1; Za działania te odpowiedzialny jest protokół **IGMP** omówiony niżej) i przekazują tylko te pakiety. Możliwe też jest rozbijanie transmisji multicast na unicastowe przez jakiś serwer w sieci lokalnej (np. dla hostów klienckich nie obsługujących multicasu, transmisja jest przekazywana unicastowo, co powoduje znaczne obciążenie tego routera bądź serwera w sieci lokalnej). Niestety Multicast nie działa w obecnym Internecie w skali globalnej gdyż bardzo wiele routerów go nie

obsługuje. Nadzieją wydaje się być IPv6 gdzie w odmiennej formie multicast jest od początku wpisany w standard.

1.3. Protokół IGMP

Protokół IGMP (*Internet Group - Membership Protocol*) jest podstawowym komponentem multicastu IP. Do tworzenia grup multicast protokół IGMP wykorzystuje adresy IP klasy D. Protokół IGMP został opisany w dokumencie RFC 1112. Host identyfikuje przynależność do grupy wysyłając komunikaty IGMP. Routery pod dyktando protokołu IGMP nad słuchują komunikatów IGMP i regularnie wysyłają zapytania w celu rozróżnienia, które grupy w poszczególnych sieciach LAN są aktywne, a które nie.

Niebagatelna role w działaniu protokołów multicast odgrywają **drzewa dystrybucyjne (MDT)**, które określają ścieżkę, po której transmitowane są pakiety od źródła do odbiorców.

1.4. Protokoły routingu multicast

Do identyfikacji grup multicast i zestawianie tras do nich można użyć jednego z wielu protokołów routingu. Może to być **PIM** (*Protocol-Independent Multicast*), **DVMRP** (*Distance-Vector Multicast Routing Protocol*) i **MOSPF** (*Multicast Open Shortest Path Protocol*).

Protokół PIM

W zależności od rodzaju ruchu protokół PIM pracuje w dwu trybach, dense (skupionym) i sparse (rzadkim).

Tryb dense używa algorytmu RPF (*Reverse Path Flooding*), który jest podobny do protokołu DVMRP. Istnieją jednak różnice, na przykład protokół PIM pracujący w trybie dense, w odróżnieniu od protokołu DVMRP, nie wymaga specjalnego protokołu unicast, może pracować z dowolnym protokołem tego typu używanym w sieci.

Tryb sparse jest przeznaczony dla intersieci ze stosunkowo niewielką liczbą sieci LAN, ale wieloma strumieniami danych. Definiuje punkty spotkań, które później są używane jako punkty rejestracji w celu zapewnienia właściwego routingu pakietów.

Jeśli nadawca chce przesłać dane, to pierwszy router, licząc od źródła, wysyła dane do punktu spotkania. Gdy odbiorca chce odebrać dane, ostatni router od strony odbiorcy rejestruje się w punkcie spotkania. Po wykonaniu tych czynności strumień danych może przepłynąć od nadawcy do punktu spotkania i do odbiorcy. Routery uczestniczące w połączeniu optymalizują ścieżkę i automatycznie eliminują niepotrzebne skoki, nawet w punkcie spotkania.

Protokół DVMRP

DVMRP (*Distance - Vector Multicast Routing Protocol*) wykorzystuje technikę RPF i jest używany jako podstawowy protokół dla internetowego szkieletu multicast MBONE (*multicast backbone*).

Protokół DVMRP został zdefiniowany w dokumencie RFT 1075 i ma pewne wady. Protokół DVRMP ma złą opinię zwłaszcza z powodu kiepskiej skalowalności sieci, wynikającej z reffloodingu, szczególnie w wersjach, w których nie zaimplementowano oczyszczania. Taki płaski mechanizm routingu unicast protokołu DVMRP wpływa na jego zdolności skalowania.

Działanie RPF polega na tym, że router w momencie otrzymania pakietu wysyła jego kopie do wszystkich ścieżek z wyjątkiem zwrotnej do źródła. Jeśli do routera jest przyłączona sieć LAN, która nie chce przyjąć określonej grupy multicast, to router w celu zatrzymania strumienia danych wysyła do źródła komunikat czyszczący.

W operacjach RPF protokołu DVMRP są używane techniki reffloodingu i adresacji *unicast*. Podczas reffloodingu routery DVMRP okresowo zalewają przyłączoną sieć w celu osiągnięcia nowego hosta. Mechanizm floodingu używa algorytmu, który bierze pod uwagę częstotliwość floodingu i wymagany czas dla nowej grupy *multicast* do przyjęcia strumienia danych. Technika *unicast* DVMRP jest używana do określenia, który interfejs prowadzi z powrotem do źródła strumienia danych. Choć technika ta nie występuje poza protokołem DVMRP, to podobna jest do użytej w protokole RIP, opartej na zliczaniu skoków.

Środowisko unicast DVMRP pozwala na użycie innych ścieżek niż używanych w ruchu *multicast*.

Protokół MOSPF

MOSPF (*Multicast Open Shortest Path First*) jest rozszerzeniem protokołu OSPF. Wykorzystuje protokół routingu unicast, który wymaga zdobycia przez każdy router w sieci informacji o wszystkich dostępnych łączach.

Router protokołu MOSPF wylicza trasy ze źródła do wszystkich możliwych członków grupy dla określonej grupy multicast. Informacja o nich jest przechowywana w tablicach stanu łącz protokołu OSPF. Tak ustalone trasy dla każdej pary źródło- grupa *multicast* przechowuje do czasu wystąpienia zmian w topologii sieci.

MOSPF może pracować tylko w sieciach, które używają protokołu OSPF. Na podstawie analizy jego licznych implementacji można stwierdzić, że:

- > najlepiej się sprawdza w środowisku, w którym jest stosunkowo mało aktywnych par źródło-grupa
- > może obsłużyć znaczące pasmo pomiędzy routerem a hostem w środowisku, które jest niestabilne lub ma wiele aktywnych par źródło - grupa.

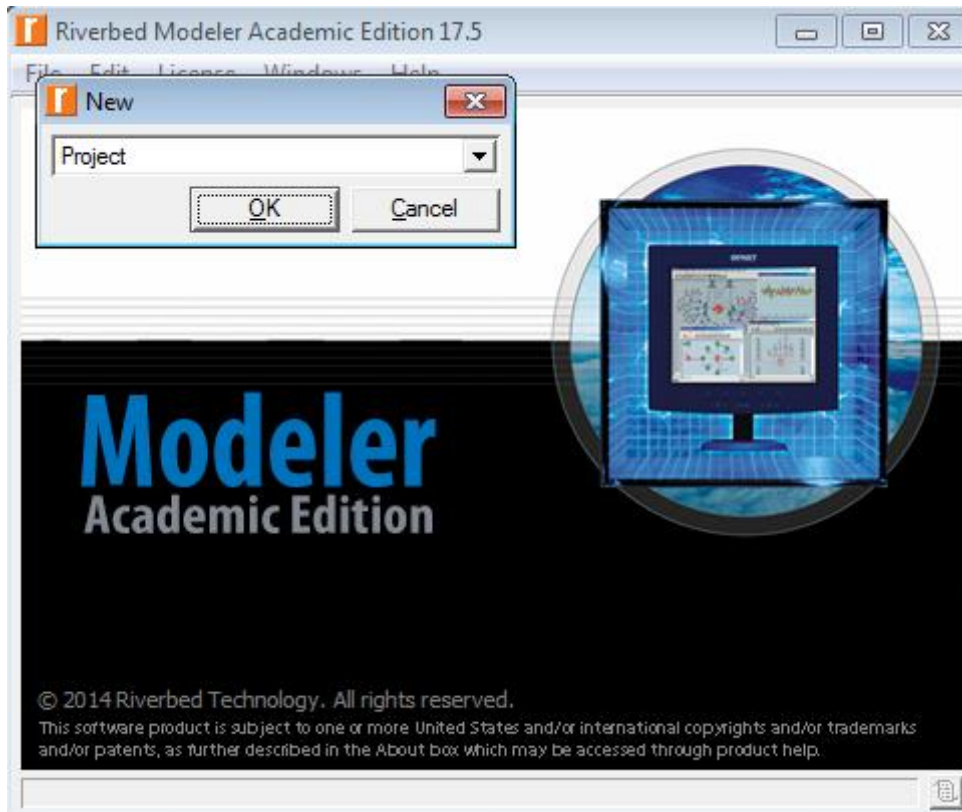
2. OPnet. Opis programu, przedstawienie możliwości oraz przygotowanie przykładowych struktur pomagających zobrazować ruch multicast o raz jego analizę.

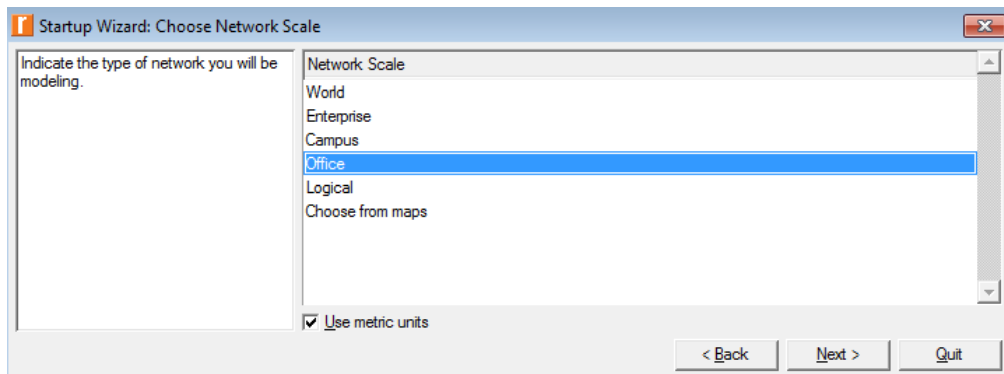
2.1. OPnet - opis programu

Program OPnet jest to zaawansowany program do symulacji budowy i działania sieci komputerowych pozwalający na śledzenie ruchu i zjawisk zachodzących w symulowanej sieci. Program zapewnia obserwacje w wielu płaszczyznach zarówno z wykorzystaniem narzędzi graficznych jak i z możliwością wyprowadzenia danych na zewnątrz programy i analizę ich w innym środowisku. Do naszej pracy wykorzystaliśmy wersje akademicką dostępną bezpłatnie ze strony producentapod adresem http://www.opnet.com/services/university/itguru_academic_edition.html. Pobranie wersji instalacyjnej oraz uruchomienie programu dostępne jest po uprzednim zarejestrowaniu się.

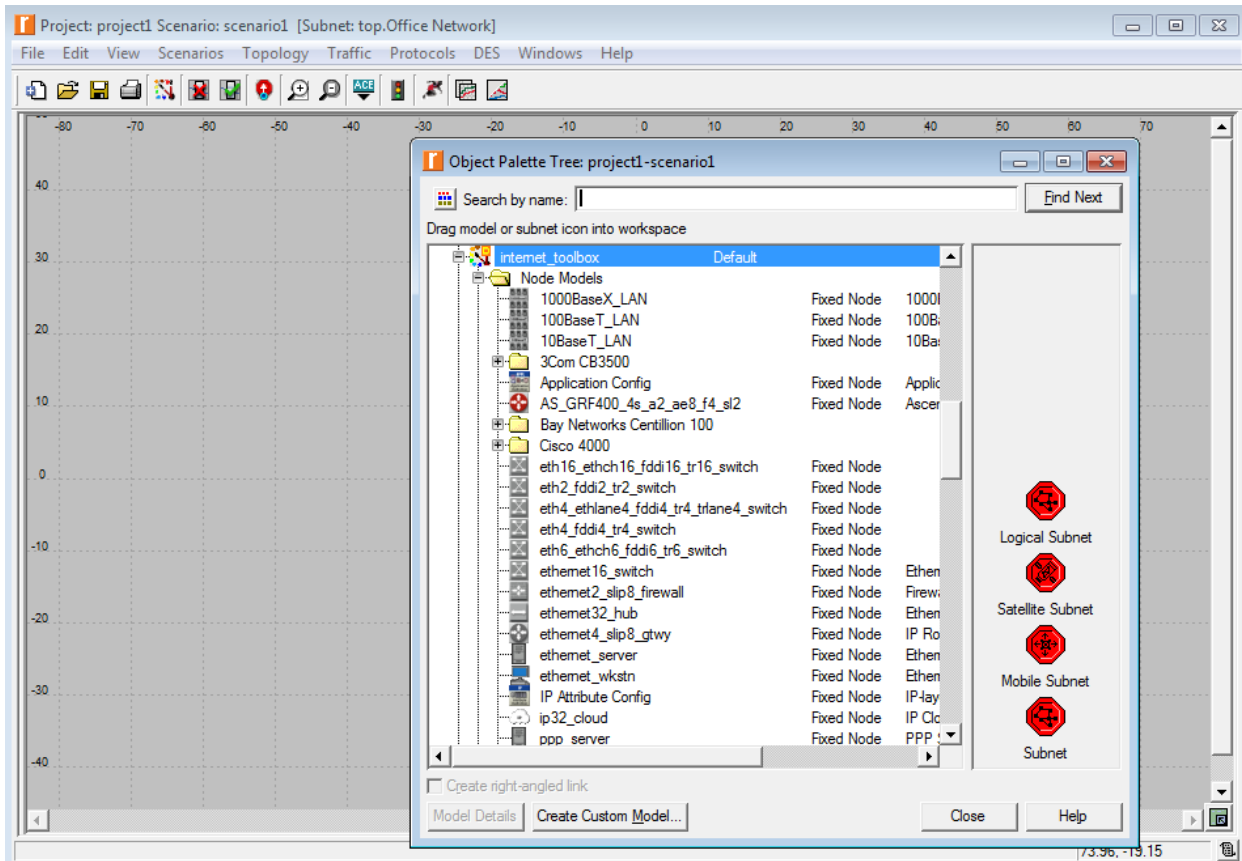
2.2. Przygotowanie programu do pracy

Po uruchomieniu należy utworzyć nowy projekt a następnie wybrać jego nazwę oraz zainicjować jego topologie. My na początek wybieramy pusty scenariusz oraz skalę sieci ustalamy na Office.





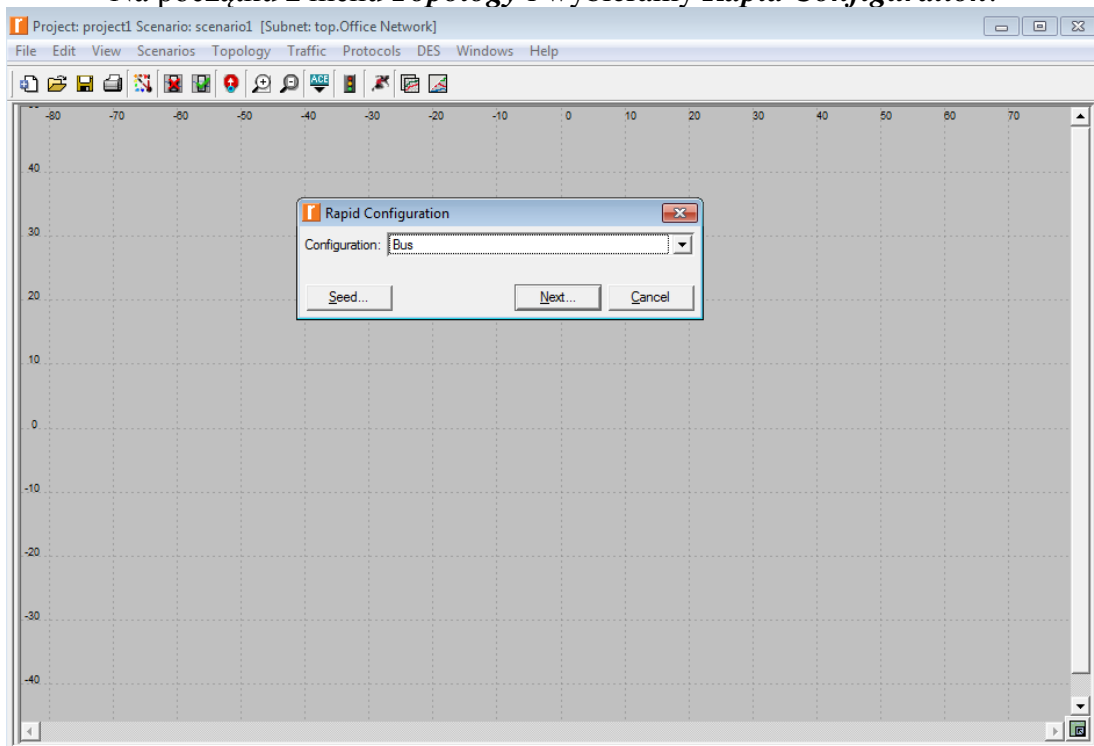
Po wybraniu w następnych oknach jeszcze wielkości sieci oraz pomijając jakieś konkretne rozwiązania sieciowe którejs z firm produkujących urządzenia sieciowe otrzymujemy gotowe okno do pracy, czyli tworzenia sieci, które powinno na tym etapie wyglądać mniej więcej tak:



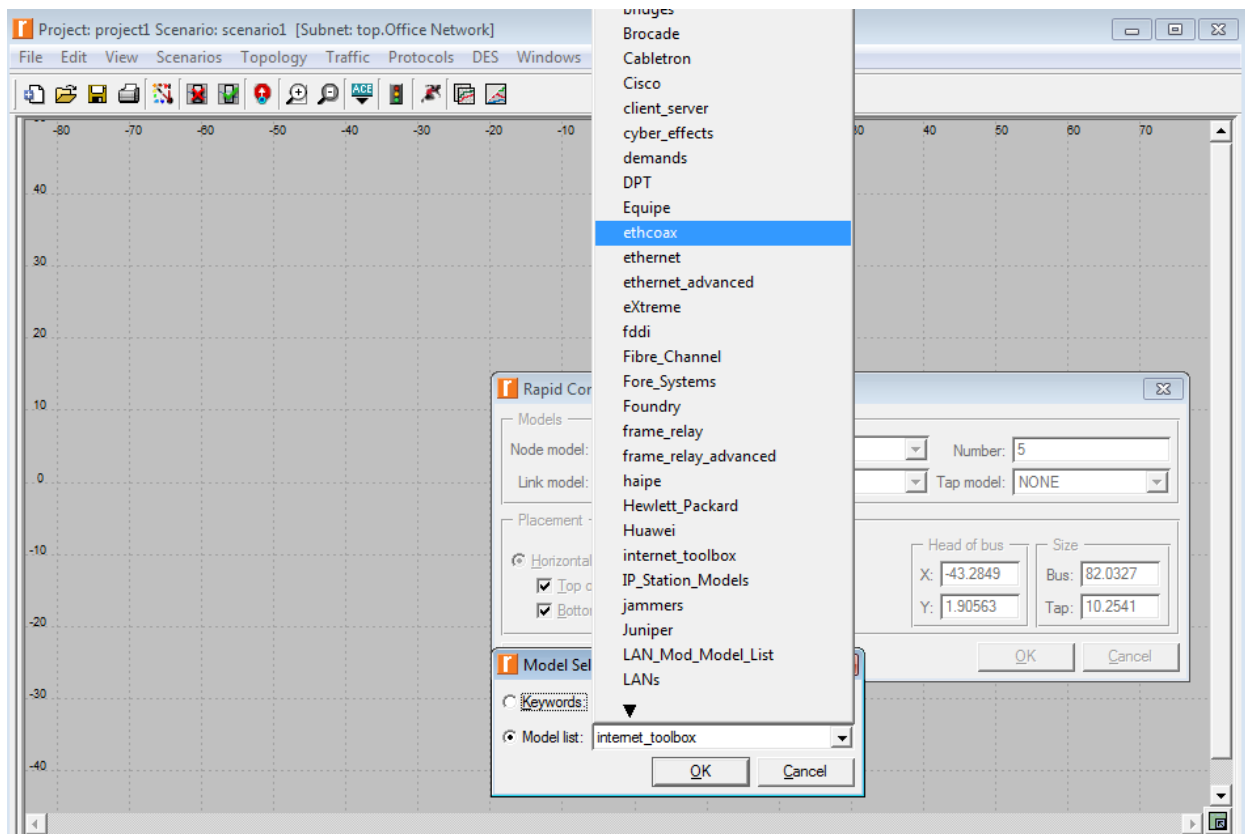
W tym oto oknie możemy przystąpić do budowania naszej sieci, która będziemy chcieli zasymulować i przetestować. Naszą budowę rozpoczęliśmy od budowania sieci opartej na magistrali. Pokażemy krok po kroku jak taka sieć zbudować i jak uruchomić w niej ruch, który będzie symulował ruch multicast. Pokażemy również jak stworzyć animacje ruchu w tej sieci oraz jak wygenerować wykresy pokazujące przepływ danych. Następnie zbudujemy jeszcze drugą sieć by pokazać jak zbudować sieci oparte o inne urządzenia, np. switche i huby a budowę pokazemy na podstawie topologii drzewa.

2.3. Budowa sieci opartej na magistrali

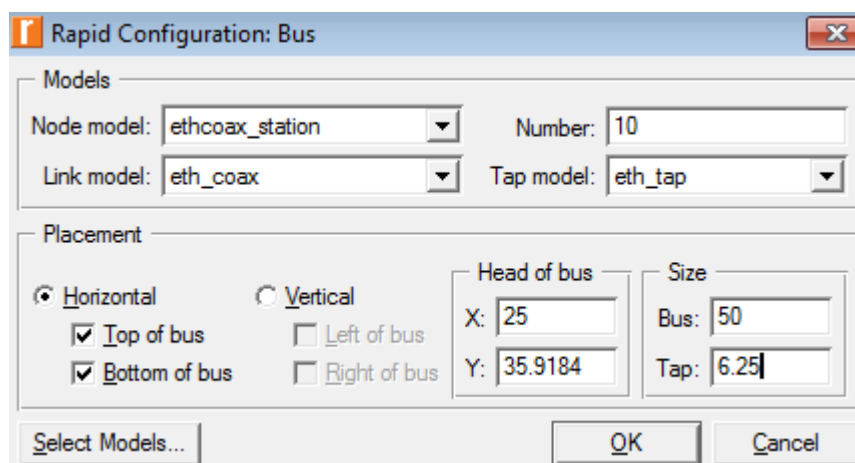
Na początku z menu *Topology* i wybieramy *Rapid Configuration*.



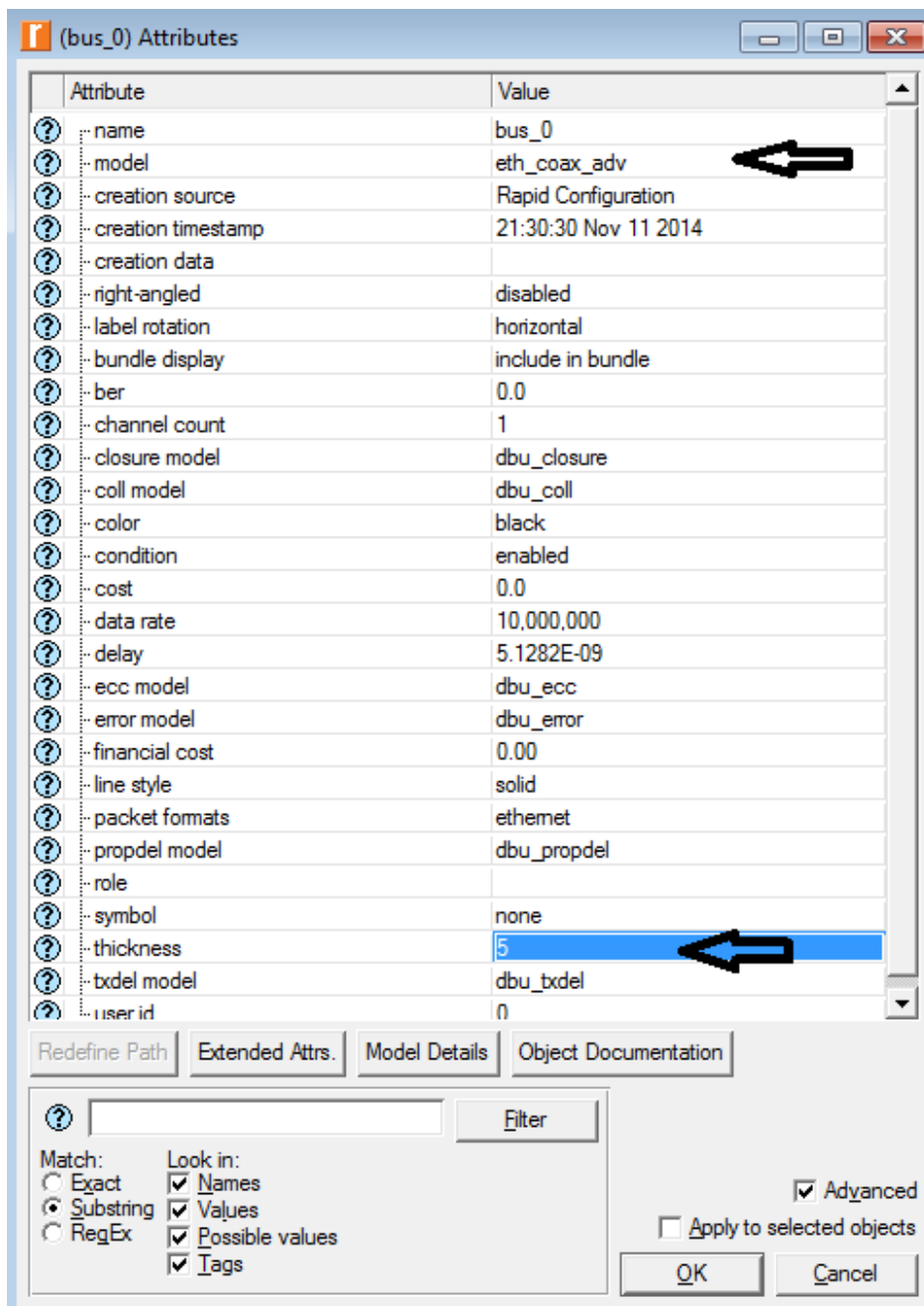
Klikamy *OK* a następnie w oknie, które się pojawi wybieramy przycisk *Select Models* a w nim wzbieramy **ethcoax** czyli kabel koncentryczny.



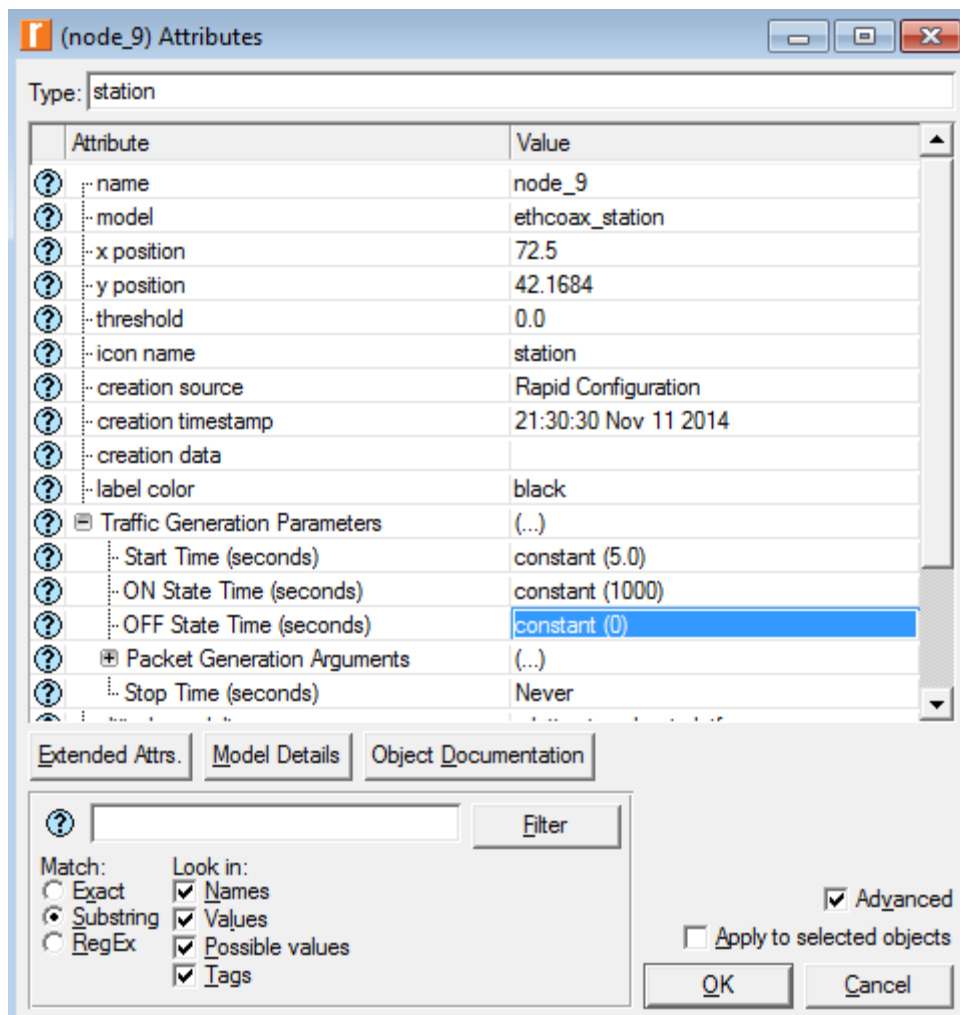
Powracamy do poprzedniego okna i wypełniamy wszystkie z pól elementami zgodnymi z magistralą tak jak poniżej:



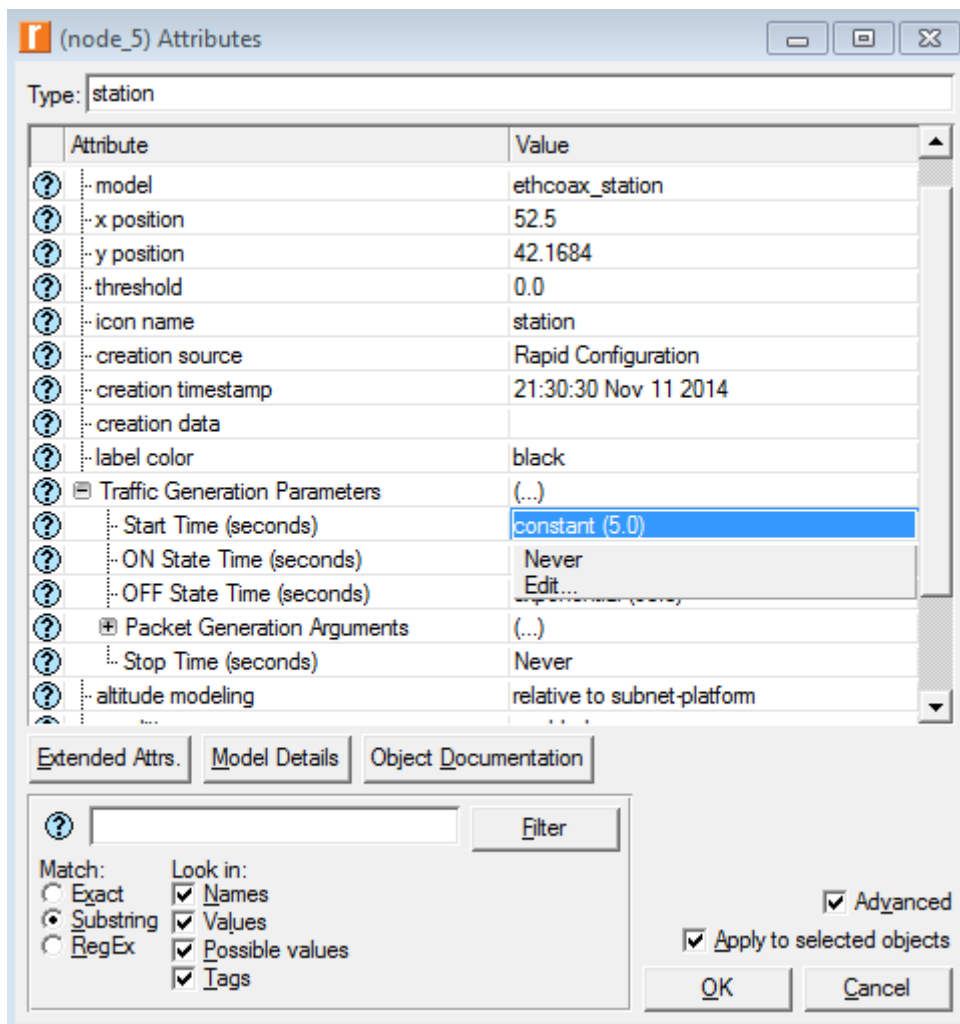
Po wciśnięciu *OK*. pojawia się nam sieć, którą będziemy badać. Od razu przechodzimy do konfiguracji magistrali, wciskamy prawym przyciskiem myszy na linie symbolizującą ją a następnie wybieramy opcje *Advanced Edit Attributes* a następnie okno modyfikujemy jak na rysunku:



Ostatnią czynnością jest przygotowanie ruchu w sieci. Wybieramy jedną ze stacji, która będzie pełniła rolę stacji nadającej dane, pozostałe będą te dane tylko odbierać. Klikamy prawym przyciskiem na pierwszej ze stacji, następnie wybieramy opcję *Edit Attributes* aby w oknie, które się pojawi rozwinąć *Traffic Generation Parameters*. W tym oknie zmieniamy opcje *ON State Time* na *constant=1000* *OFF State Time* na *constant=0*, od tego momentu stacja ta będzie stale generować ruch, co będzie odpowiadało wysyłaniu danych przez serwer bądź transmisji strumieniowej, która jest głównym typem transmisji multicast a następnie klikamy *OK*.



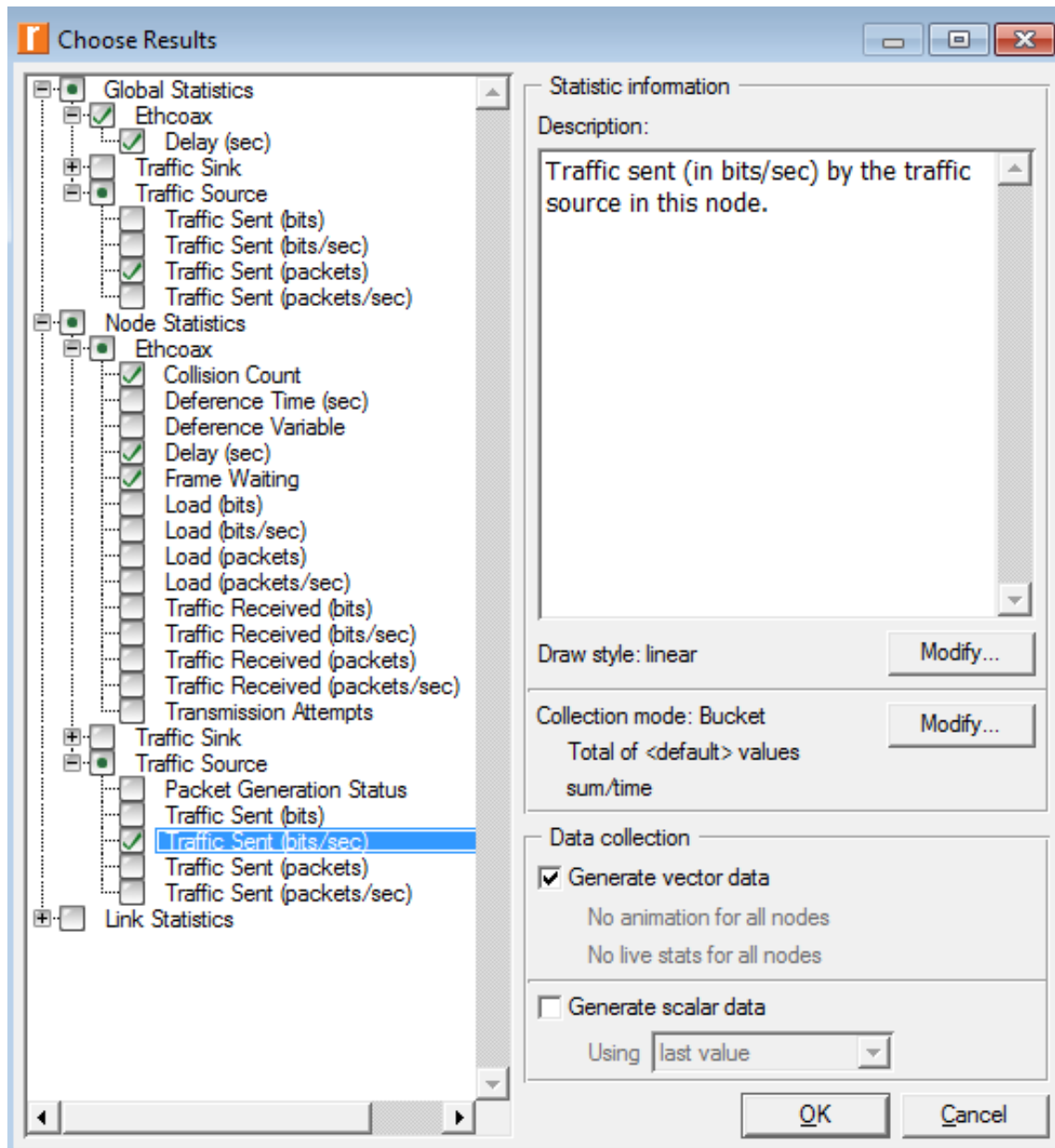
Zaznaczając zaś wszystkie pozostałe stacje wybieramy ponownie opcję *Edit Attributes* aby w oknie, które się pojawi rozwinąć *Traffic Generation Parameters* lecz teraz zmieniamy tylko opcje *Start Time*. Klikamy na nią i z okna wybieramy opcje *Never* na pozycji *Special Value*. Po zmianie tej opcji i zaznaczeniu jeszcze opcji *Apply Changes to Selected Objects* w oknie następnym decydujemy o tym ze te stacje nie będą nadawać żadnych danych.



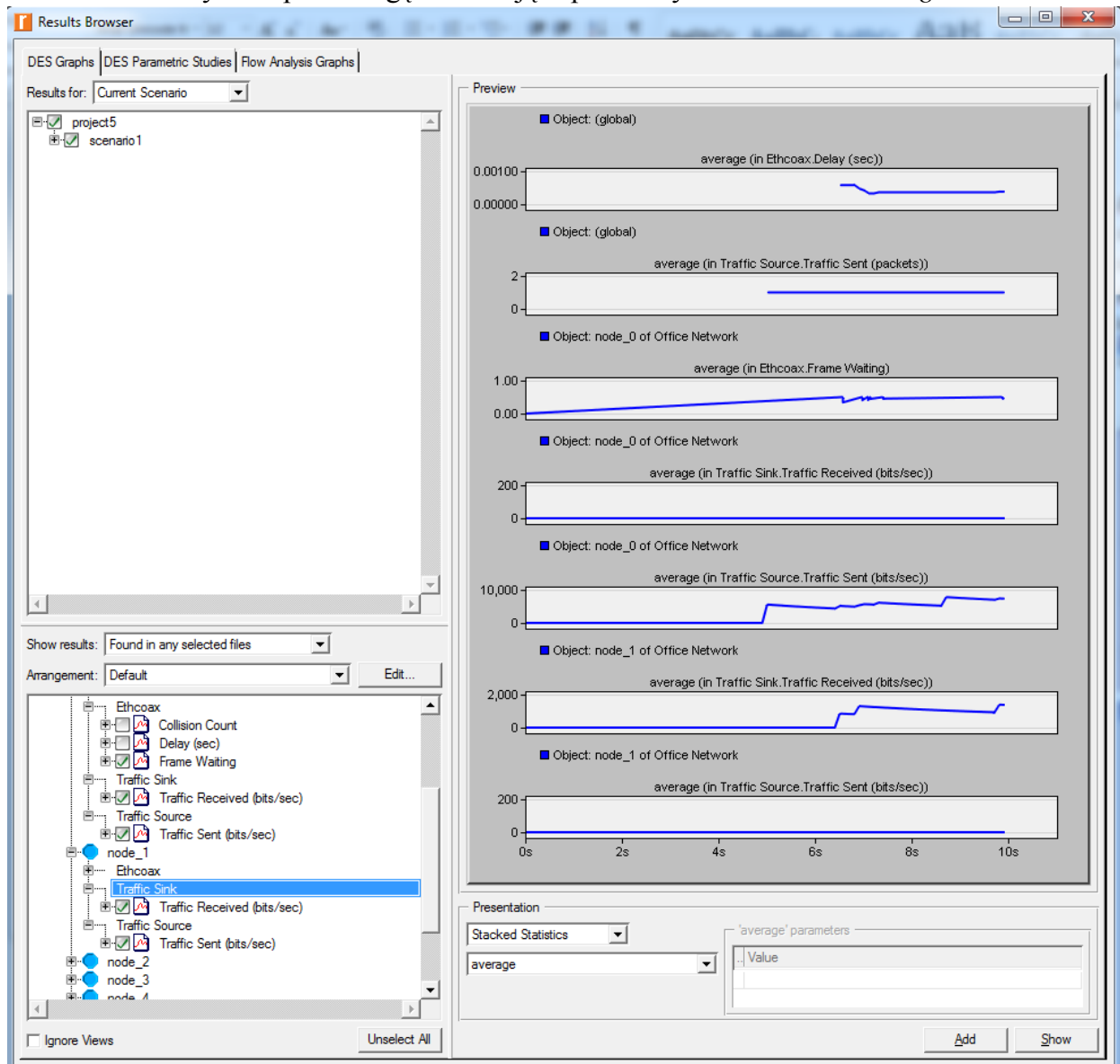
W ten sposób, przygotowaliśmy sieć do symulacji.

Symulacja:

Aby poprawnie przeprowadzić symulacje najpierw z menu **DES** wybieramy opcję **Choose Individual Statistic**. Tam z rozwijanych opcji powinniśmy wybrać opcje pod kątem których chcemy przetestować konfigurację. My proponujemy do wybrania takie jak: ilość danych wysłanych, pobranych (w pakietach bądź bitach), opóźnienie, kolizje oraz oczekujące ramki (Frame Waiting). W menu **DES** zaznaczamy także opcję **Record Animation For Subnet** co pozwoli nam na odtworzenie animacji drogi oraz kolejności przepływu danych w symulowanej sieci po przeprowadzeniu procesu.



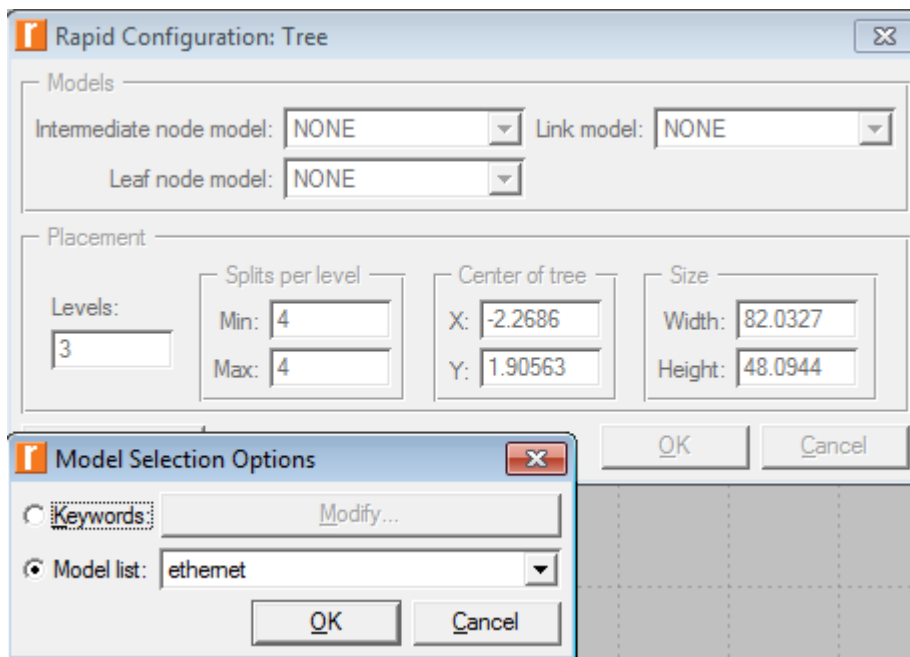
Aby uruchomić symulację wybieramy z tego samego menu opcję **Configure Discrete Event Simulation** ustalamy tam czas pracy sieci, jaki ma zostać zasymulowany a następnie klikamy **Run**. Po zakończeniu symulacji klikamy **Close** a następnie, aby przejrzeć wyniki symulacji wybieramy z menu **Results** opcje **View Results** oraz opcje **Play Animation**. W oknie **View Results** wybieramy te wykresy, które chcemy brać pod uwagę zmieniając sposób wyświetlania na *average*.



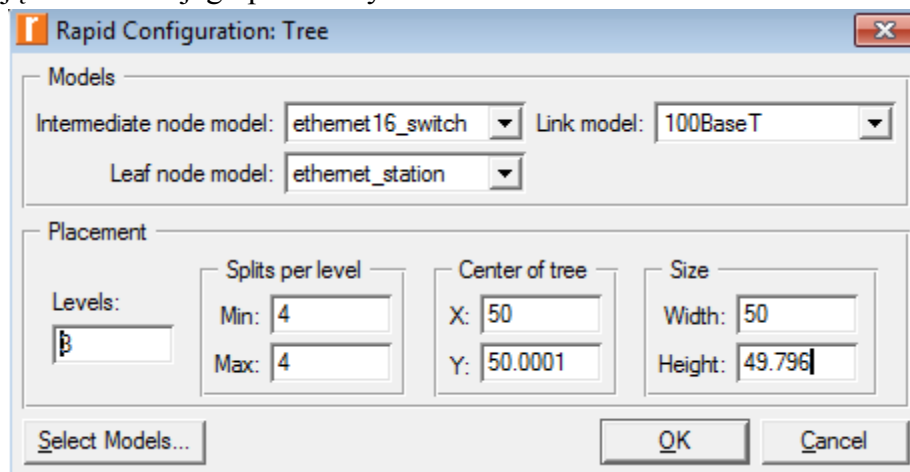
2.4. Budowanie sieci opartej na topologii drzewa.

Do przygotowania sieci o topologii drzewa w oknie **Rapid Configuration** wybieramy opcje **tree**. Postępujemy dokładnie tak samo jak w poprzednim przypadku, jeśli chodzi o

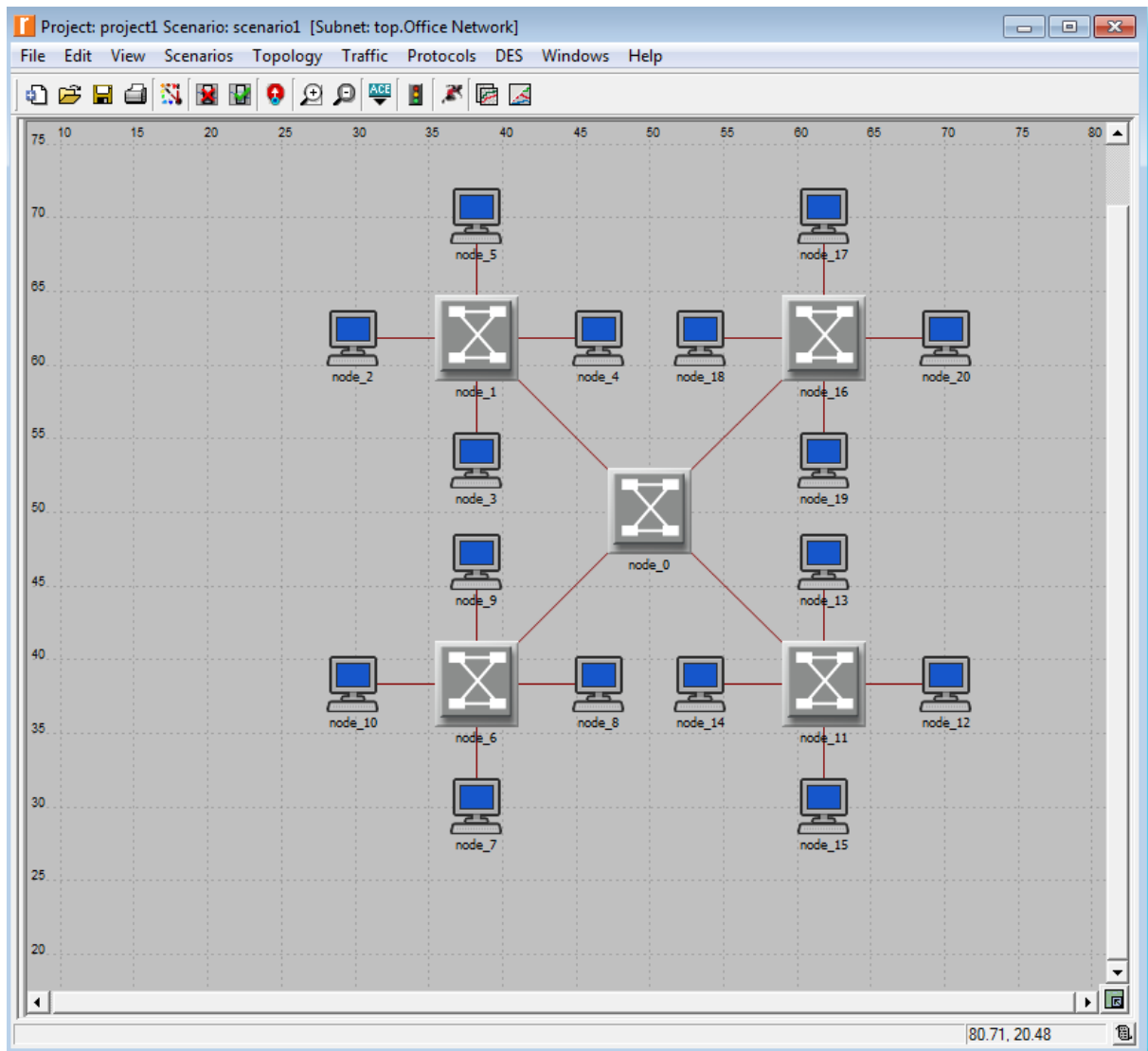
generacje ruchu w sieci. Zasadniczą różnicą jest tylko przygotowanie okna po wybraniu opcji **Rapid Configuration**. Po wybraniu opcji **Select Model** wybieramy *ethernet*.



Następnie okno wypełniamy w następujący sposób, chyba że zamierzamy użyć innych elementów, po pojawieniu się sieci każdy element można edytować osobno zmieniając dowolnie jego parametry.



Po takim przygotowaniu sieci i konfiguracji ruchu sieć wygląda mniej więcej w ten sposób:



Symulacje przeprowadzamy dokładnie w ten sam sposób jak w przypadku magistrali.

3. Zadanie do samodzielnego przygotowania.

Na podstawie przykładów tworzonych sieci opracuj projekt sieci i przetestuj go pod kątem ruchu multicast. Do przygotowania sieci użyj nie tylko elementów pokazanych na przykładach, ale spróbuj użyć innych narzędzi i urządzeń dostępnych w menu.

W trakcie tworzenia projektu wykonaj następujące czynności:

- spróbuj połączyć dwie lub trzy różne topologie wybierając urządzenia, które mogą realnie obsłużyć i stać się mostem pomiędzy dwoma sieciami
- ustal poprzez ile urządzeń musi przejść pakiet aby dotrzeć do najbardziej odległego punktu twojej sieci (możesz wspomóc się animacją lecz pamiętaj jaki wpływ na pakiet w transmisji multicast ma parametr TTL)
- ustal jak przechodzenie poprzez kolejne węzły wpływa na opóźnienia generowane w sieci (wyniki symulacji)
- uzasadnij dlaczego transmisja multicast jest bardziej wydajna niż transmisja unicast w przypadku wideokonferencji.

Wnioski oraz uzasadnienia z przeprowadzonych czynności zamieść w protokole a następnie poprzyj odpowiedniki wykresami wygenerowanymi podczas symulacji.