

Planning and Analyzing an efficient mesh in an European backbone network using OPNET SP Guru

Alberto Arto Peñas

Dept. of Network Management and Capacity Planning, Telefónica de España
Madrid, Spain

E-mail: alberto.artopenas@telefonica.es

Abstract

Due to a hierarchical Two-Level topology of Telefonica's backbone, both traffic destined to peering routers and traffic between clients in the same AS must be routed through both levels. This results in an important increase in transmission speeds, higher costs, lower bandwidth efficiency and makes redimensioning a more frequent task. And as the traffic between clients represents an important percentage of total traffic, a partial mesh between first-level routers might be a good solution. In this technical paper, an efficient mesh is looked for taking into account, when configuring the IGP metrics, that new links must only carry internal traffic even if a link connected to the upper level fails. With a FLAN, the results will be evaluated in terms of bandwidth efficiency and reduction of costs and Failure Analysis capability will tell us about the correct behaviour of traffic against failure. Finally we will review new opportunities such as backup TE LSPs in the mesh based on class-of-service criteria

1. Introduction

Some principles that apply when designing IP networks are well-known: hierarchy, modular infrastructure construction, limiting scope of failure and healing of infrastructure faults using dynamic routing with fast convergence. Since long distance backbone links tend to cost more, a hierarchical topology is usually chosen. In this way, all the traffic to different border routers is aggregated in an upper level within the backbone so that an important number of long distance links can be saved up ([1], [2]). So does Telefonica's backbone network: as shown in Figure 1 the POPs are connected to the backbone using a first aggregation level, which consists of nine pairs of routers geographically distributed. None of these pairs are connected to each other. Over this level, a second aggregation level exists in Madrid and Barcelona. As long as most of the traffic is destined to external networks, this topology works out rather well and the cost efficiency is high. But when both total customer traffic and percentage of traffic inside the own AS (between POPs) regarding total traffic increase rapidly (up to 40%), the benefit of taking all the traffic up to the second aggregation level starts to be questionable and an efficient partial mesh could be desirable. Because the higher the transmission speeds become, the lower the bandwidth efficiency since a slight increase in traffic might cause an important leap in transmission resources/speeds that it could not be assumable due to unavailability of the transmission speed or a long response time. For these reasons, the deployment of a partial mesh would reduce the traffic between both levels up to a 40% as well as the number of hops between two first level routers (up to 2 hops in case of a first level router from Madrid and one from Barcelona) by means of lower speed links (an idea similar to *divide and conquer*).

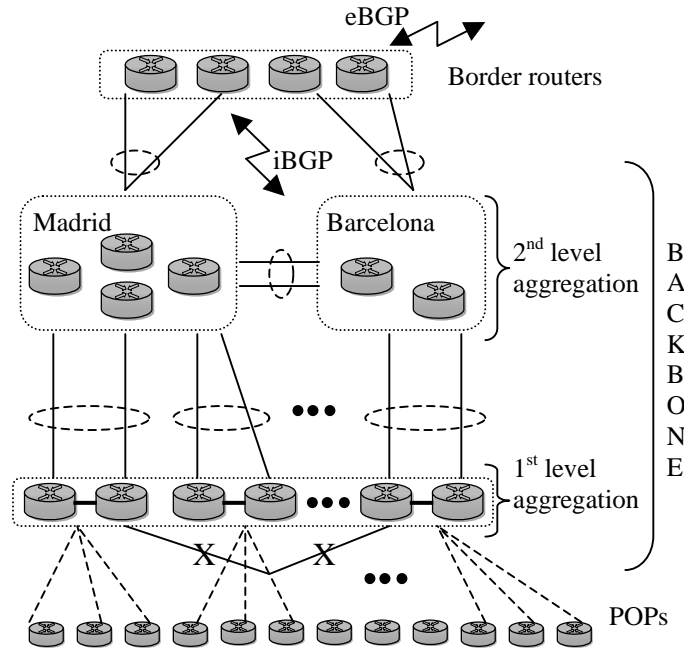
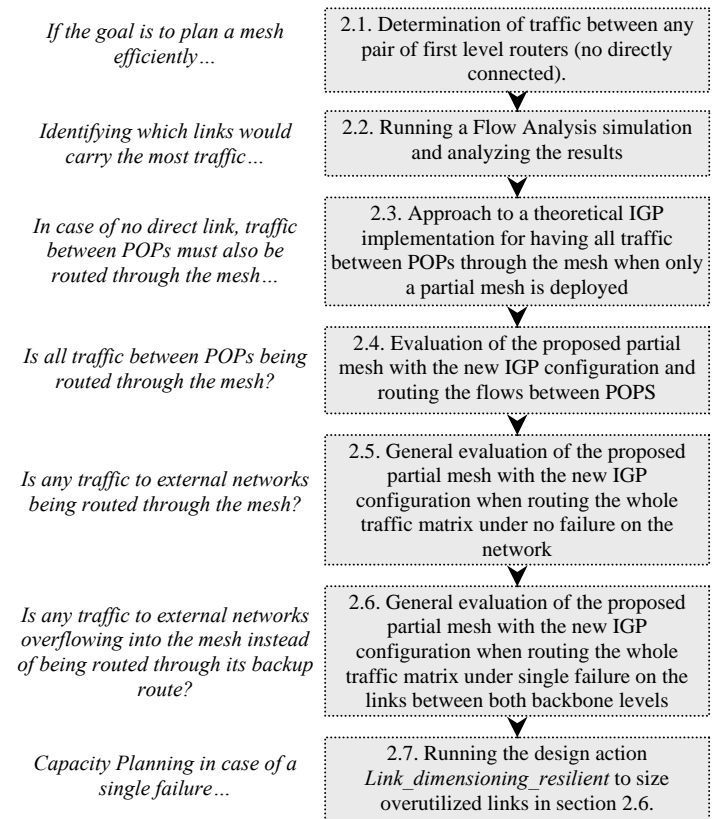


Figure 1: Backbone's architecture diagram

2. Background, simulation methodology and results.



The remainder of the paper is organized according to the methodology represented in the previous flowchart.

2.1. Determination of traffic between any pair of first level routers (no directly connected)

As all the defined and imported flows are between POPs or between POPs and border routers, there is no way to know beforehand how much traffic is routed between 2 first level routers (they are neither source nor destination of the flows). It will depend on routing tables. So a *Flow Analysis* (FLAN) simulation is required. Having all the flows routed could allow us to know the traffic matrix between routers in the first aggregation level. However, after looking up the output reports from FLAN module, a report in which all the routes taken by all the flows during the FLAN run are parsed together with the traffic routed per each route hasn't been found. The required information would be something like that (sc-source, dt-destination, BR-backbone router):

```
sc1 → BRx → BRy → BRz → ... → dt1 → traffic11
sc1 → BRx' → BRm' → BRz' → ... → dt1 → traffic11,
sc1 → BRx → BRy → BRw → ... → dt2 → traffic12
sc1 → BRx' → BRn' → BRz' → ... → dt2 → traffic12,
sc2 → BRx' → BRy' → BRz' → ... → dt1 → traffic21
....
```

Having this information, it would be easy to process it to obtain the traffic which is looked for (submitted suggestion 75758).

Anyway, a workaround has been thought. It lies in setting up a full mesh of links between all first level routers and running a FLAN simulation to have all traffic between POPs routed through them. Having 9 pairs of routers in this level, the configuration of 144 links (2 x 9 x 2 x 8 / 2) will be required. For this goal, a Model Assistant file will be used. This is available from the Topology menu: Topology → Model Assistant → Save Current Topology to File. Using this option after drawing a new link manually, the link information can be exported to a file and the format in turn can be copied to create other links and then apply them using the Topology → Model Assistant menu options. It's important to note that an existing port must be specified in the Model Assistant file (if the port doesn't exist, SP Guru doesn't create it automatically). For that reason, firstly, we have to find

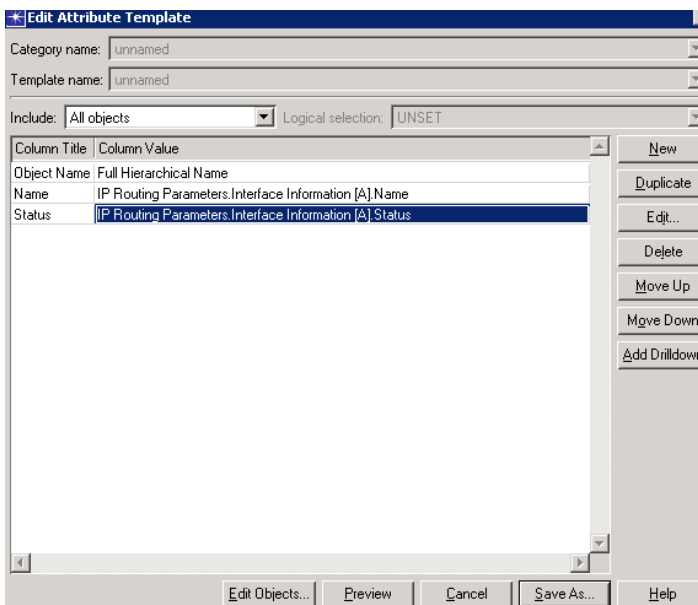


Figure 2: Editing an attribute template

and identify 16 free ports per first-level router and, otherwise, create them. To identify these free ports, we:

1. Select the 18 first-level routers.
2. Use Edit → Edit Attribute Template and, for each selected router, show the following attributes:
 - a. Object Name: Full Hierarchical Name.
 - b. Name: IP Routing Parameters. Interface Information [A].Name
 - c. Status: IP Routing Parameters. Interface Information [A].Status.

As it is shown in Figure 2. And then click on Preview and, on the new window, File → Export → To Spreadsheet. So, a list with all the interfaces and their status (active/shutdown) has been created for all the first-level routers so that 16 shutdown interfaces per router will be able to be filtered and selected to define the new links in the Model Assistant File. After choosing the shutdown interfaces and matching them, the new links are included in the ma file. The selected link type is different to be able to select them easily by 'Select Similar links' option. Example for a link:

```
start_header_block
    Model Assistant Version:    1.1
end_header_block

start_link_specifications
    "top.Imported Network.routerA[POS6/0] <-> top.Imported
    Network.routerB[POS0/1]":    4.4736e+007;PPP_DS3_int;routerB
    <-> routerA
end_link_specifications
```

After applying the ma file by Topology → Model Assistant → Apply File..., next step is going to be the basic configuration of the interfaces, i.e.: status (Active), address (Auto Assigned), Subnet Mask (Auto Assigned), Routing Protocol(s) (IS-IS) and Metrics Level-2. Again, the full hierarchical Name of each attribute is obtained by means of Edit → Edit Attribute Template...:

```
IP Routing Parameters.Interface Information [A].Name
IP Routing Parameters.Interface Information [A].Status
IP Routing Parameters.Interface Information [A].Address
IP Routing Parameters.Interface Information [A].Subnet Mask
IP Routing Parameters.Interface Information [A].Routing Protocol(s)
IS-IS Parameters.Interface Information [B].Metrics.Level-2
```

Where A and B are the indexes within the (Interface Information) table (equal the row number inside the attributes window) and are the only ones used to apply values to attributes when using a Model Assistant file (Name is an attribute and can not be used to index an interface). However there's no an automatic way to get a list with the index for each interface name so that it could be process it to identify A and B and create the previous lines. Consequently the configuration has been done by hand directly on the attributes window. The selected IS-IS metric for the links in the mesh had been 15 so that traffic between POPs always preferred the route through the mesh to the route through the second level and traffic from Barcelona area to external networks in Madrid were routed through the links between the second-level routers instead of through the mesh. But, after running a first simulation and checking that the traffic through the mesh was higher than the traffic between POPs (known beforehand) due to some cross traffic from

Barcelona to Madrid to external peers, the easiest solution was chosen: to filter the flows between POPs among all the flows (easily in a spreadsheet) and generate a new .tr2 file (instead of having to think the suitable metrics out). The final step has been importing it using Traffic → Import Traffic Flows → From Text (.tr1, .tr2) Files...(as shown in Figure 3):

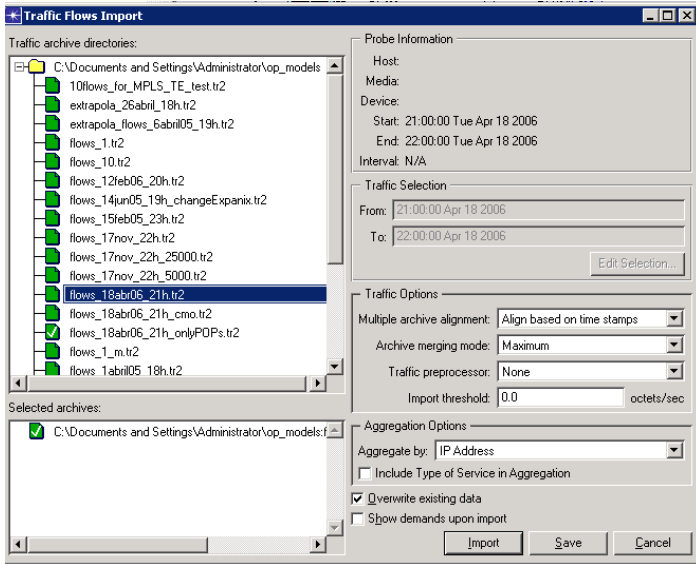


Figure 3: Importing Traffic Flows

2.2. Running a Flow Analysis simulation and analyzing the results

Two steps:

1. Flow Analysis → Configure/Run Flow Analysis...
2. Flow Analysis → Results → View Reports and selecting *Flow Analysis Tables* tab: Global Tables → Performance → Link Utilization → Show. In the new window, File → Export → To Spreadsheet...

Analyzing the results, because of how the network is built and how the traffic is balanced, we realize that the traffics from a router to both routers in a different first level pair are practically the same and that both links can be put together into only one, having all the traffic to that pair routed through it, as long as all the routers are connected to all the pairs. A possibility consists of connecting the first router of each pair to the rest of first routers and the second router of each pair to the rest of second routers. At this point, the full mesh would consist of 72 links. When representing the cumulative distribution curve of traffic through the mesh depending on the number of links in the mesh (Figure 4), it can be seen that the cumulative distribution is quite close to the perfect distribution line (GINI coefficient [4] = 1.15) which means that traffic is quite well-spread out between the 9 pairs of routers. Specifically, to have 50% traffic routed through the mesh, 28 links are needed.

Therefore a partial mesh based on traffic criteria is not as decisive as it might seem and, on the other hand, transmission criteria become determining factors. So, a mixed solution has been considered: only the links which carry out the transmission requirements have been taken into account. In this case, pursuing again a 50% traffic directly routed between two first level routers, the selection has been widened up to 36 links. However, as we will see in next section, we will force the remaining 50%

traffic to be also routed through the mesh by means of IGP configuration.

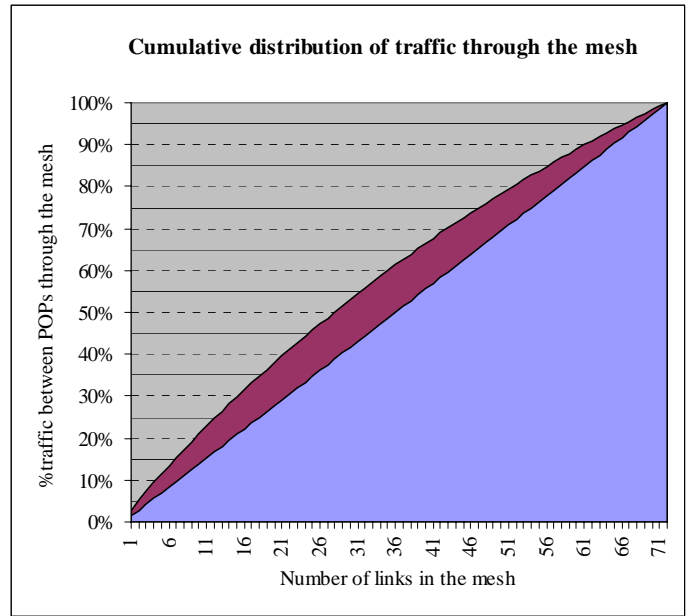


Figure 4: Traffic versus number of links in the mesh

2.3. Approach to a theoretical IGP implementation for having all traffic between POPs through the mesh when only a partial mesh is deployed

Without using MPLS Traffic Engineering (TE), the goal lies in forcing all the flows between POPs to be routed through the mesh by means of appropriate IGP metrics so that a two-hop route along the mesh is preferred over a route through the second aggregation level (a wide enough mesh is supposed so that it always exists a route no longer than two hops between the routers in the mesh). The used IGP protocol is IS-IS and the metric, by default, 10. First of all, let's define three different values for the metric/weight of the following links (as shown in Figure 5):

- X: let X be the metric for the existing links between the routers

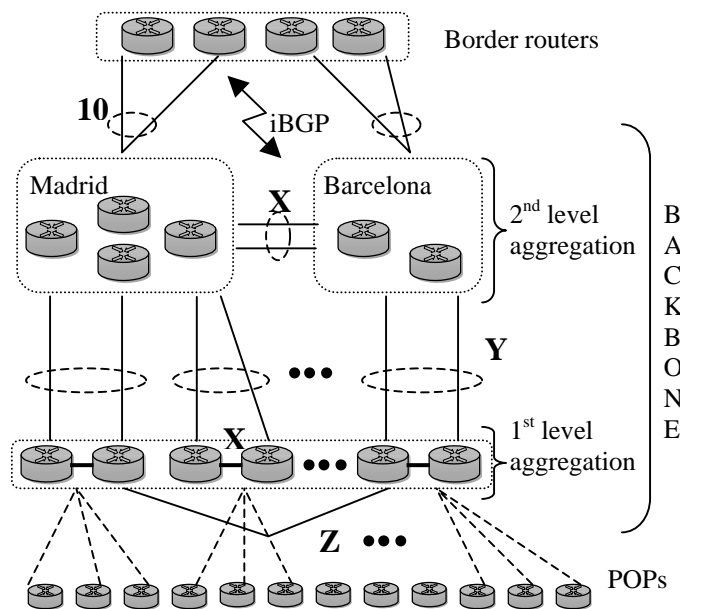


Figure 5: Set of metrics to configure

of each pair in the first aggregation level and the links between routers of the second aggregation level.

- Y: let Y be the metric for the links between both levels.
- Z: let Z be the metric for the new links in the mesh.

When proposing these new IS-IS metrics, two criteria have been kept in mind:

1. Defining the least number of different IS-IS metrics due to fast convergence, stability and simplicity motivations. And 3 is the least number to have a compatible system of inequations.
2. Assuming the same link weight across all links in the same level or between both levels since symmetry and homogeneity in the whole network is fundamental to have a basic understanding about how flows are routed.

Secondly, the conditions that X, Y and Z have to fulfill are the following:

Operation under no failure

a) All the flows between POPs must be routed though the mesh. Condition 1: highest distance through the mesh < lowest distance though 2nd level → $Z+Z < Y+Y \rightarrow Z < Y$

b) All the flows to external networks must be routed through the 2nd level. Condition 2: longest default route (cross-traffic from Barcelona to Madrid) < route through the mesh → $Y+X+10 < Z+Y+10 \rightarrow Z > X$

Operation under single failure in a link connecting both levels

- a) Flows between POPs: nothing changes.
- b) Flows to external networks (important to note that traffic must be re-routed through the router in the same pair) must be routed through the 2nd level. Condition 3: longest route (from Barcelona to a specific iBGP peer in Madrid) < shortest route through the mesh → $X+Y+X+X+10 < Z+Y+10 \rightarrow 3X < Z$ (the 3rd route is because of iBGP configuration).

Operation under single failure in a link belonging to the mesh

a) Flows between POPs (important to note that traffic must be re-routed through the route in the same pair) must not be routed through the 2nd level. Condition 4: longest route through the mesh using the router in the same pair < shortest route though 2nd level → $X+Z+Z < Y+Y \rightarrow X+2Z < 2Y$

Resulting conditions

$$Z < Y$$

~~Z > X~~ (the following condition is more restrictive)

$$3X < Z$$

$$X + 2Z < 2Y$$

To facilitate fast convergence and guarantee stability of the network ([3]), X will not change and still equal 10 (since links with metric X are the most numerous). So, the conditions become:

$$Z < Y$$

$$30 < Z$$

$$2Z + 10 < 2Y$$

and, for example:

$$Z = 35$$

$$Y = 45$$

2.4. Evaluation of the proposed partial mesh with the new IGP configuration and routing only the flows between POPs

The goal of this simulation is going to be to check that all the traffic between POPs is being routed through the mesh by means of a second hop in case of a direct link doesn't exist. Firstly, the excess of links in the mesh has been deleted (144-36=108 links). Secondly, the IS-IS level-2 metric has been set to 35 in the new

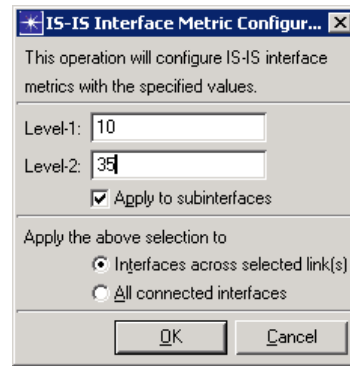


Figure 6: Changing IS-IS metrics on interfaces

links. Easily, right-clicking on one of these links and 'Select similar links' (as the selected link type is different from the rest). After that, Protocols → IS-IS → Configure Interface Metrics... (see Figure 6) and applied a level-2 metric of 35 to the interfaces across selected links.

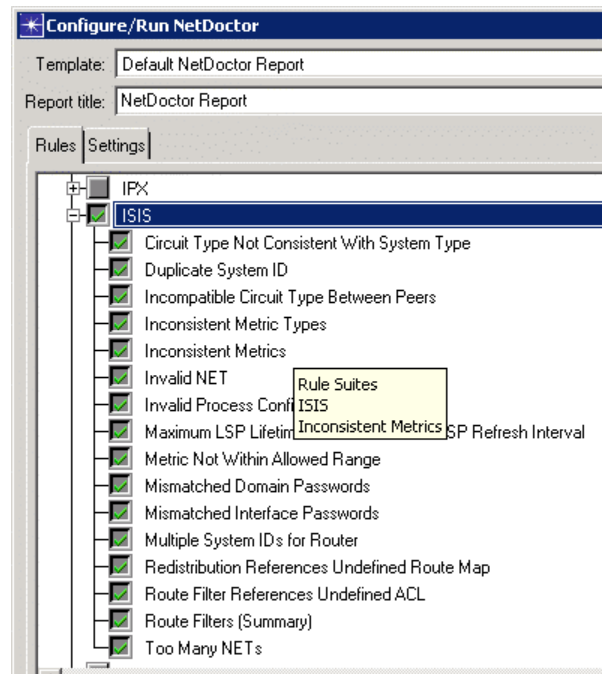


Figure 7: Validating the new metrics with NetDoctor

Next, the metric for the links between both levels must be changed from 10 to 45. We operate the same as previously. Finally, a NetDoctor simulation has been run to check the correct configuration of the new metrics across the network. Only the ISIS rule suite (as shown in Figure 7) has been selected to make the simulation faster. Neither errors nor warnings have been found so that we have proceeded to run a FLAN using the flows between POPs.

Analyzing the results, the following facts have been found:

- None of the traffic has been routed through the links between both levels. All the traffic has been routed through the mesh. The configured ISIS metrics work properly for traffic between POPs.
- Traffic through the mesh is 48,96% higher than when having a full mesh. This value is also correct since, when deploying the 36-link partial mesh, only 51,04% of traffic goes through direct links between first level router whereas the remaining 48,96% needs a second hop. So, given X as the initial traffic though the

full mesh, the expected growth in the traffic through the deployed partial mesh would be:

$$\Delta = \frac{0.5109X + 2 \cdot 0.4896X}{X} - 1 = 48,96\%$$

2.5. General evaluation of the proposed partial mesh with the new IGP configuration when routing the whole traffic matrix under no failure on the network

The new test is going to be to check that flows to external networks aren't routed though the mesh. The verification will consist of comparing the new traffic in the mesh with the one obtained in section 2.4. Again, after running Flow Analysis, Flow Analysis → Results → View Reports and Export to Spreadsheet. The traffic in each link has been added up and the total equals the traffic obtained in section 2.4. exactly. So, all the traffic to external network is being routed as before the deployment of the mesh.

2.6. General evaluation of the proposed partial mesh with the new IGP configuration when routing the whole traffic matrix under single failure on the links between both backbone levels.

In this case, the goal of the simulation is to make sure that traffic directed to external networks isn't routed through the mesh when a link, which connects both levels, fails. The traffic must be redirected to the router in the same pair and this one will route it through its connecting link(s) to the upper level. The amount of traffic to external network doubles the traffic between POPs and, because of that, the capacity planning between both backbone levels is based on STM-64 whereas, in the mesh, on STM-16. That's why it's necessary to prevent traffic to external networks from overflowing into the mesh.

Before running a Failure Impact Analysis, all Failure Candidates must be failed in the scenario. So, we select all of them and, right-clicking, 'Fail this link'. Immediately afterwards, the Failure Impact Analysis is run with the following options set (as shown in Figure 8):

- Failure candidates: Failed objects in scenario
- Failure scenario: Iterate over single failures

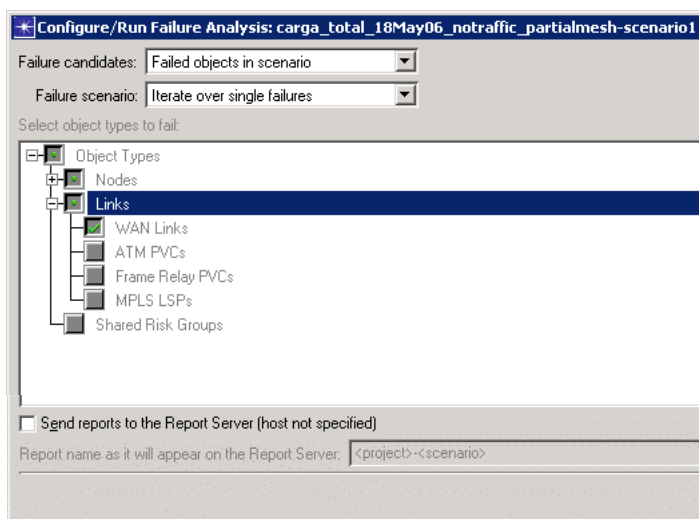


Figure 8: Configuring a Failure Analysis

In the following lines, the beginning of the Flow Analysis log is shown:

```

Product: FlowAnalysis
Version: 4.0.1241
Build Date: Apr 23 2005 23:43:29

== Failure Analysis Summary ==

Start Time: 21:00:00.000 Apr 18 2006
End Time: 21:10:00.000 Apr 18 2006
Duration: 0/0/0 00:10:00
Interval Size: 300 sec
Number of Intervals: 2
Report On: 0 (Simultaneous Peak)
Reporting Time: 21:00:00.000 Apr 18 2006
Intensity Factor: 1.000000
Flow Analysis Mode: Head-end Reroute

Failure Candidates: Failed Objects
Failure Scenario: Single
Number of Runs: 36

Failed Object: Imported Network.router1 / POS3/0 (A_B_C_D) <->
router2 / POS4/0 (A_B_C_D)

Impacted objects by shared risk groups: 1
Imported Network.router1 / POS3/0 (A_B_C_D) <-> router2 / POS4/0
(A_B_C_D)

Performance Analyzer Results:
Measure                                     Value
WAN Link - Number of overutilized links      7
WAN Link - Maximum Utilization (%)           187
WAN Link - Total Consumed BW (bps)           8.19204e+011
WAN Link - BW Efficiency (%)                 28.5
LAN - Maximum LAN Utilization (%)            0.000000
Demand - Total Active Demands                21386
Demand - Failed-Unroutable Demands           0

*Real IP Addresses have been replaced by the string A_B_C_D

```

It can be seen that the number of runs has been 36, one per failed object. A brief summary for each simulation appears in the log, reporting information about the number of overutilized links, maximum utilization, etc. To know specifically the overutilized links in each case: Flow Analysis → Results → View Reports and, then, Global Tables → Analyses → Failure Impact – Overutilized links. In this report, all links overutilized when failing each selected object are shown. After exporting it to spreadsheet, none of the links in the mesh has been found, which means that traffic to external networks aren't overflowing into the mesh.

2.7. Running the design action Link_dimensioning_resilient to size overutilized links in section 2.6.

The last step has been to run the design action *Link_dimensioning_resilient* to size existing links in the IP network such that their bandwidth accommodates traffic flow in the normal case and each defined failure case.

First of all, the failure candidates must be selected by means of the option Edit → Select Objects... and, in the new window, the selection of the following attribute criteria (link's condition equals 'disabled') (see figure 9):

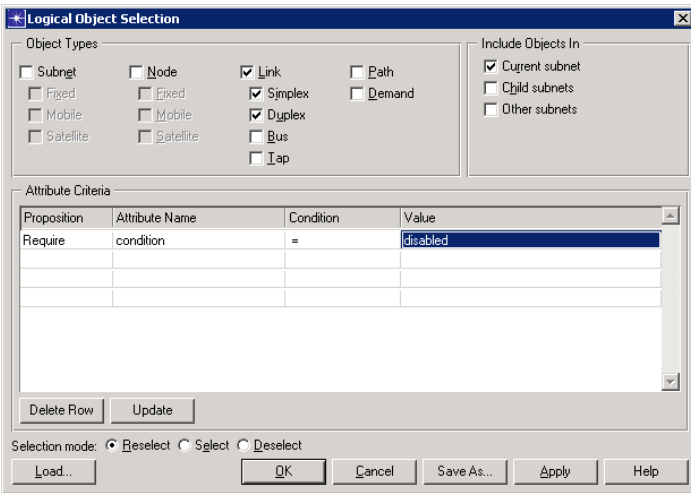


Figure 9: Selecting the failed objects in section 2.6.

Now, after finding the design action in the Configure/Run Design Action window and clicking on 'Edit Attributes', the following window pops up (Figure 10):

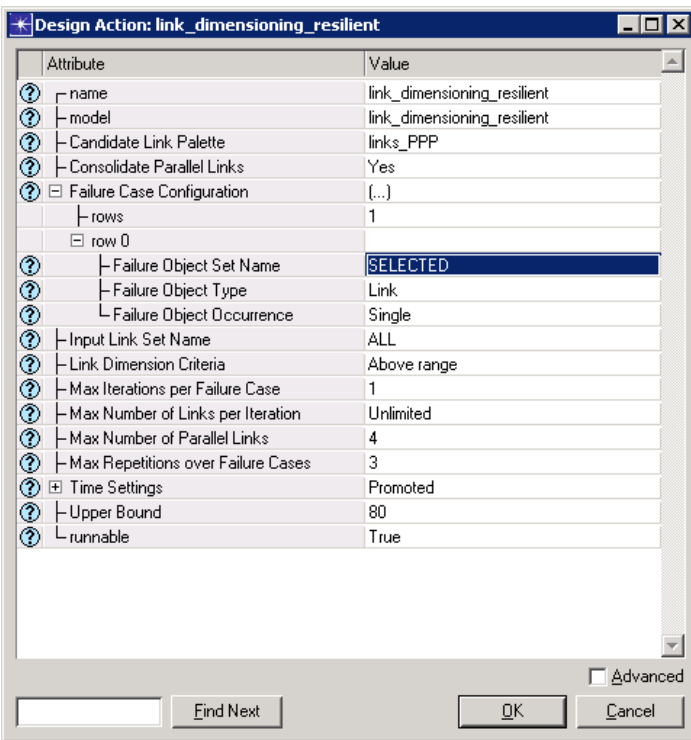


Figure 10: Configuring attributes for link_dimensioning_resilient design action

Where the attribute *Failure Object Set Name* is set to SELECTED and the *Max Number of Parallel Links* equals 4 (so that when this number is reached, the bandwidth of each link is increased instead of adding new links).

Once the design action has been run, the upgraded or added links through the iterations can be seen in the table *Link Change Summary* by clicking on Design→Results→View Output Tables... In this table, the initial and final model, bandwidth, cost and utilization are shown for each link of each bundle. Specifically, in this simulation 33 links were added and other 35 upgraded.

3. Future study and simulations

The idea is to have somehow the most important traffic (enterprises, traffic SLA, etc) protected against pairwise failures, specifically, the ones affecting the existing links between a pair of first level routers and their upper level. Faced with a double/triple failure on these links (when several links depend on a fiber), all the traffic to external networks would try to reach its destination through the mesh. Since this amount of traffic is quite high, the mesh would become useless. So, it would be quite advisable to classify and mark the initial flows depending on their importance so that they can be mapped into different E-LSPs. Thus, after configuring the association flow→LSP, we would be able to run the design action mpls_te to compute primary and secondary route for the LSPs defined in the MPLS network. In an early stage, Traffic Engineering (TE) would be only configured in these cases. It's important to note that the definition of a Shared Risk Group (SRG) per each pair of first level routers would be necessary. Each SRG would consist of all the links between both routers and the upper level. So the primary route would still be the default route and the secondary route would come on stage when the SGR failed. Figure 11 shows all the configurable parameters when running the design action:

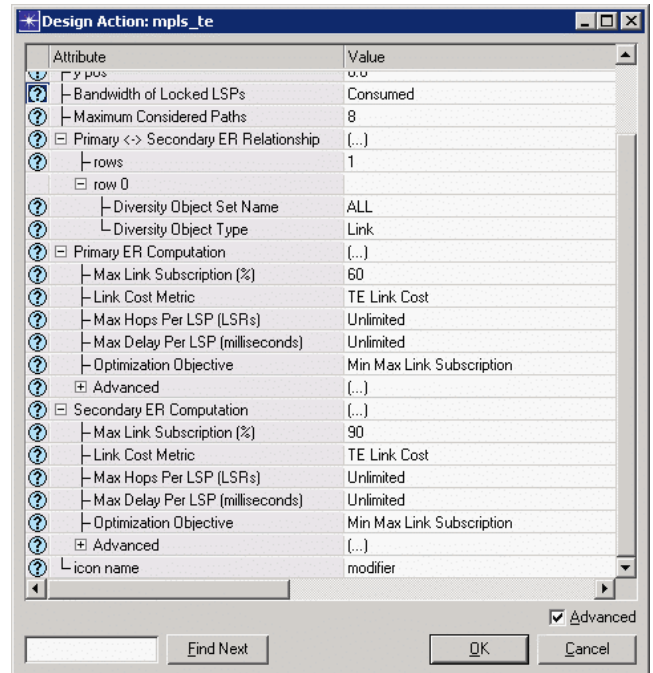


Figure 11: Configuration window of mpls_te design action

4. Conclusion

In this paper we have presented the methodology used to plan and analyze an efficient mesh in Telefonica's backbone by means of SP Guru. Every step has been followed by its corresponding simulation so that the output results have allowed us to supply additional feedback during the whole process. It's important to note the value of this additional feedback and the automation of the analysis. For these reasons, SP Guru has showed itself as a valuable tool and has allowed us to evaluate the bandwidth saving when deploying a mesh and the new bandwidth efficiency on the whole network. Finally, the large amount of processed data has been remarkable.

Regarding the implementation on real network, these results have been the starting point used in the final mesh design. As well as a higher bandwidth efficiency, other goals like a better matching between the sources and destinations of IP traffic and the long distance transmission routes, a higher connectivity and robustness between areas (protection against multiple transmission failures by increasing the possible routes to reach a destination) and a lower latency have been pursued. So the final design has combined a set of different network architectures: not only have direct links between first-aggregation routers been deployed but co-located first- and second-aggregation routers have also been merged and cross-connection between first-level routers and second-aggregation routers has been carry out in some cases. The resulting architecture is not detailed due to confidentiality reasons and will be implemented before the end of 2006. In terms of costs, a 17 million dollar CAPEX reduction is expected in the next two years.

References

- [1] C. Diot, "A Tier-1 IP Backbone Network: Architecture and Performance", Sprint Advanced Technologies Laboratories, 2003.
ipmon.sprint.com/pubs_trs/tutorials/Backbone-Architecture-And-Performance.pdf
- [2] L. Palacios, "Designing Complex IP Networks", Greece, Cisco Expo, February 2005.
http://www.ciscoexpo.gr/2006/downloads/0802/ball2/tdesign_1.pdf
- [3] The Internet Society RFC 2791, "Scalable Routing Design Principles", July 2000. <http://www.ietf.org/rfc/rfc2791.txt>
- [4] Gini coefficient definition:
http://en.wikipedia.org/wiki/Gini_coefficient (accessed 24/07/2006).