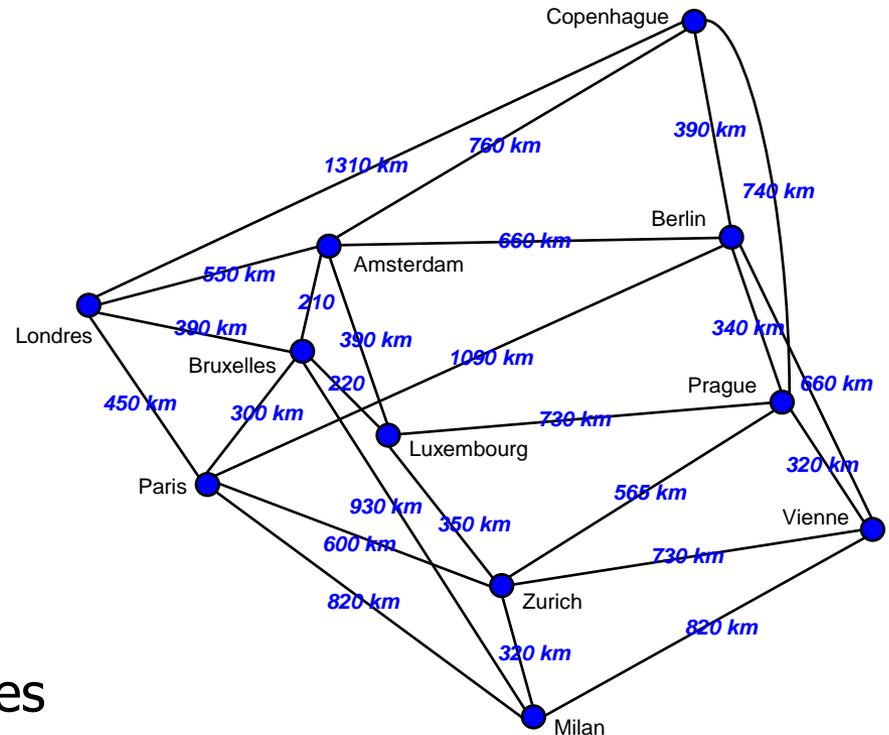


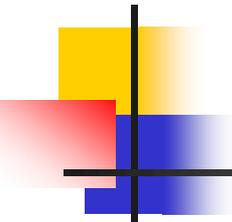
Outil de dimensionnement de réseau optique WDM: PORTO

RNRT PORTO: Alcatel CIT, France
Télécom R&D et Mascotte
(UNSA/CNRS/INRIA)

Réseau dorsaux optiques

- Réseau *Backbone*
- Transporte des « tuyaux » STM-n
STM-16 : 2,5 Gbit/s
- Topologie arbitraire
Réseaux maillés
- Routage et groupage des demandes:
Coût des équipements des nœuds du réseau
(fibre déjà posée)



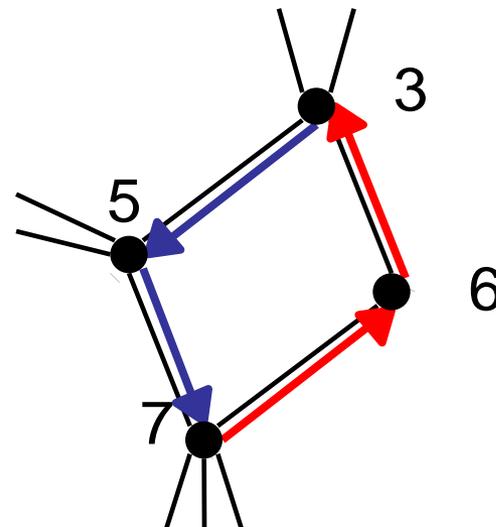
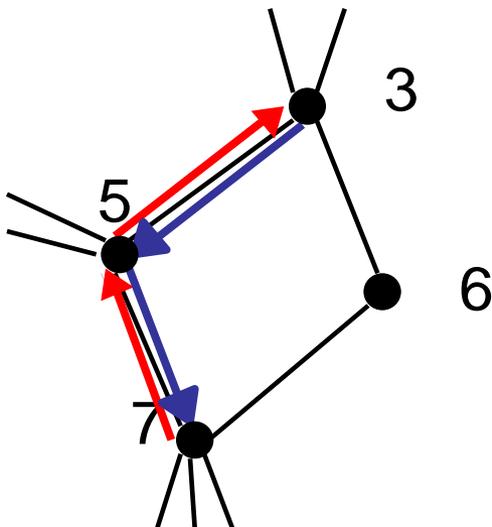
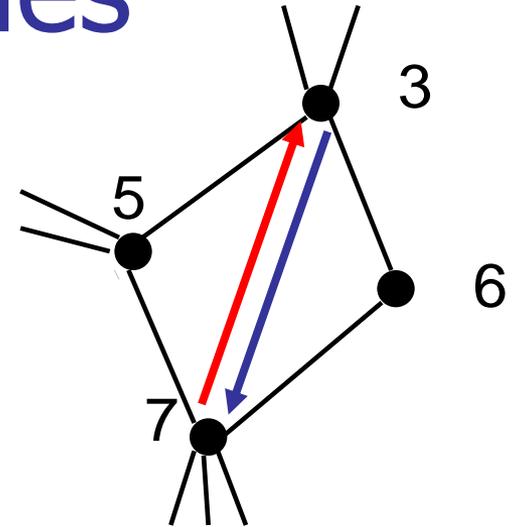


Routage des demandes

- Entrée du problème
 - Graphe orienté $G=(V,E)$
capacités sur les arcs (nombre de λ), longueur des fibres en km
 - Matrice de demandes $D [d_{i,j}]$
 $d_{i,j}$ est le nombre de λ demandées entre i et j
- Résultat : routage physique de D
Un ou plusieurs chemins pour chaque $d_{i,j}$ respectant les contraintes de capacité dans G

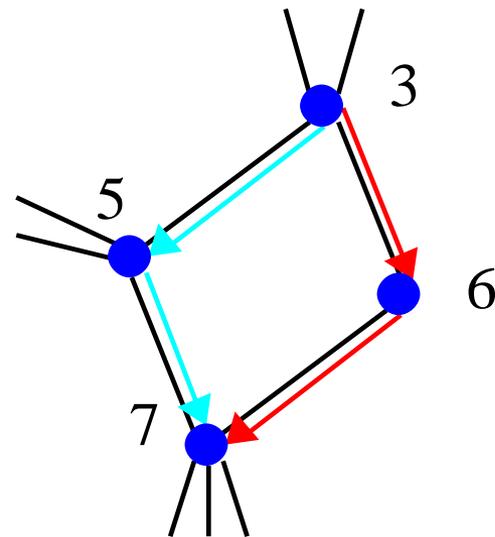
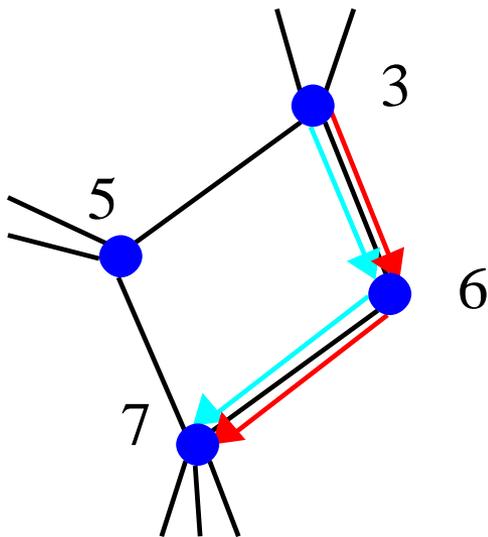
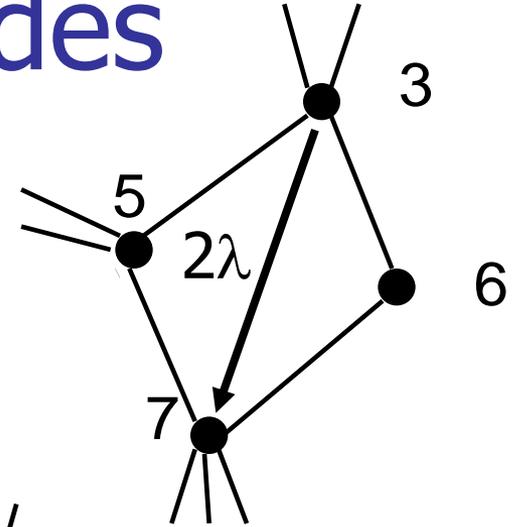
Routage des demandes

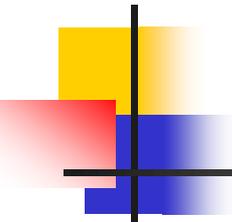
- routage *symétrique*
- routage *non symétrique*



Routage des demandes

- *Mono routage*
- *Multiroutage*





Routage et capacité fixées

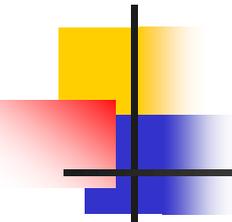
$$\forall d \in D, \forall e \in E, \lambda(e, d) \leq e_{size}$$

$$\forall d \in D, \forall v \in V$$

$$\sum_{e \in E, e \in v^-} \lambda(e, d) - \sum_{e \in E, e \in v^+} \lambda(e, d) = AddDrop(v, d)$$

$$\forall d \in D, \forall e \in E, \lambda(e, d) \in N$$

$$Obj : Min \left(\begin{array}{c} \sum_{e \in E, d \in D} \lambda(e, d) \end{array} \right)$$



Politiques de sécurisation

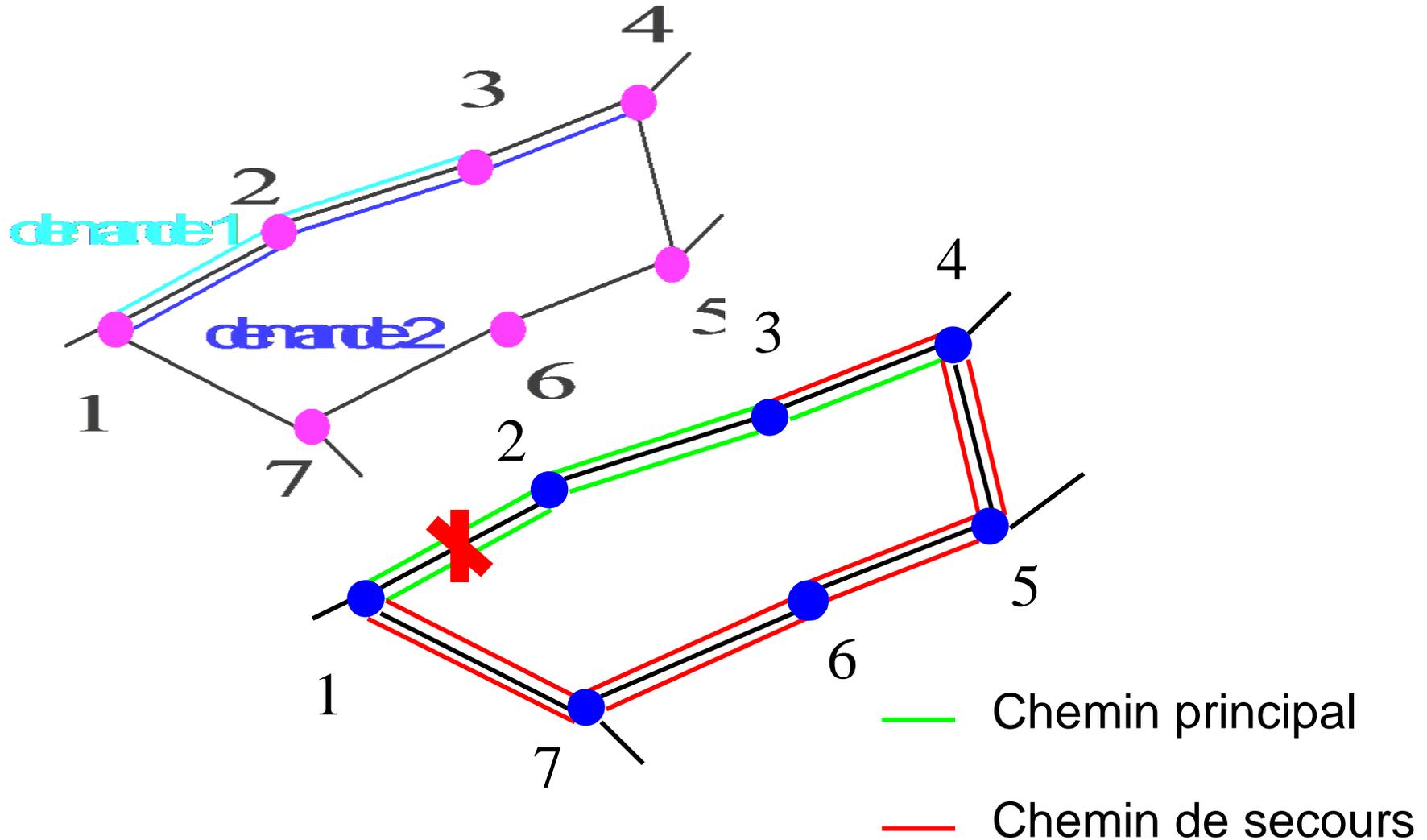
- Protection

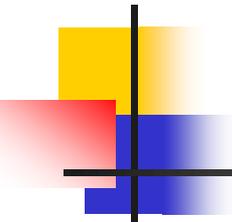
- 1+1
- 1 : 1 (« *un pour un* ») Flot de capacité 2
- M : N (« *M pour N* »)

- Restauration

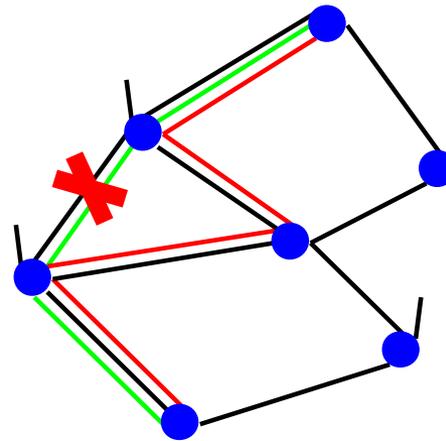
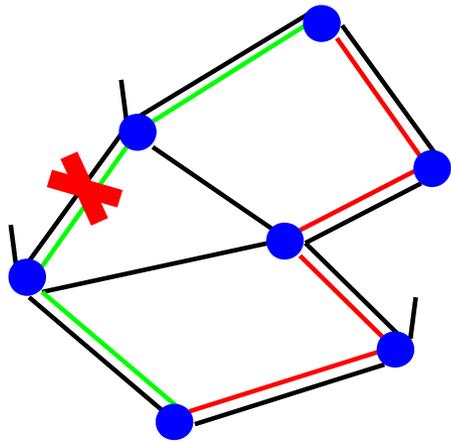
- Restauration de chemin
- Restauration de section

Restauration de chemin



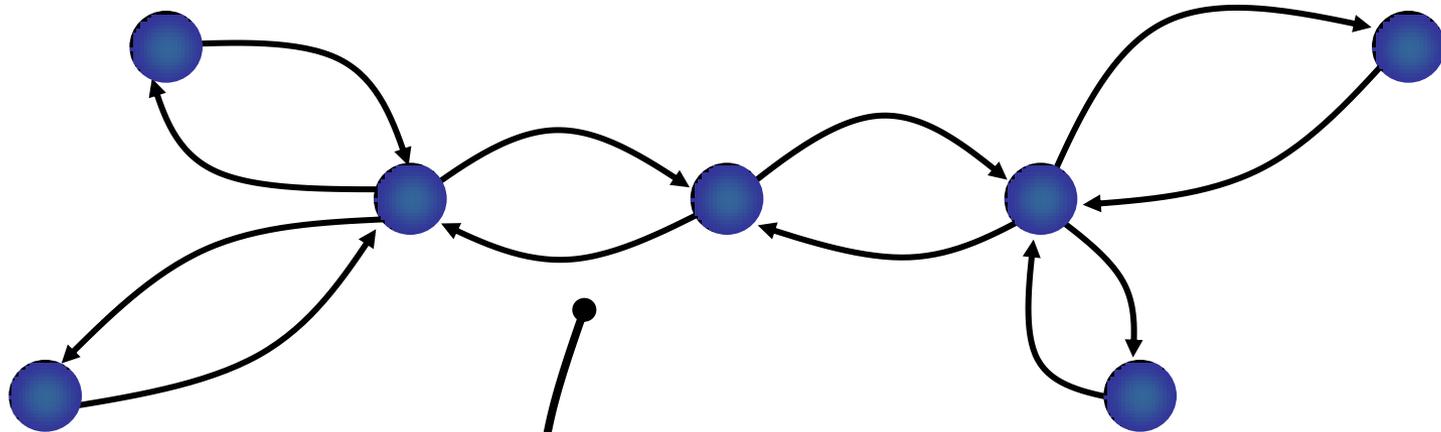


Restauration de section



-  Chemin principal
-  Chemin de secours

Groupage: modèle WDM



Réseau $G=(V,E)$

$e \in E$



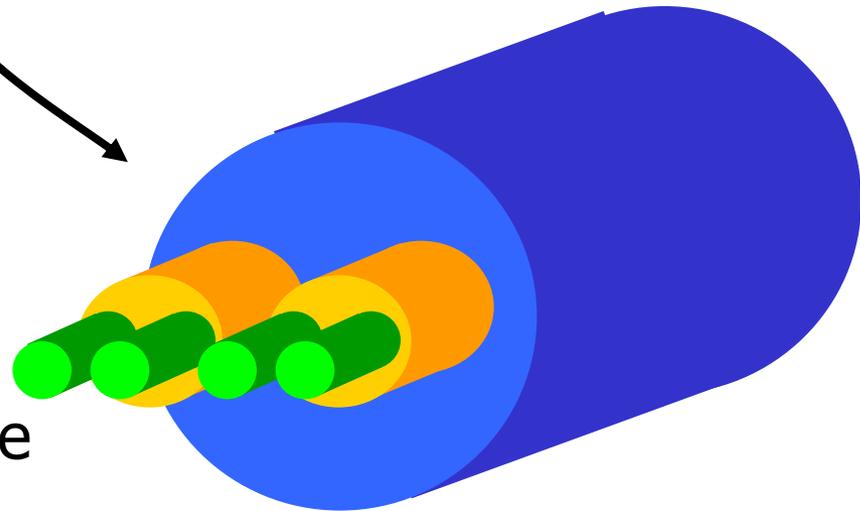
Fibre



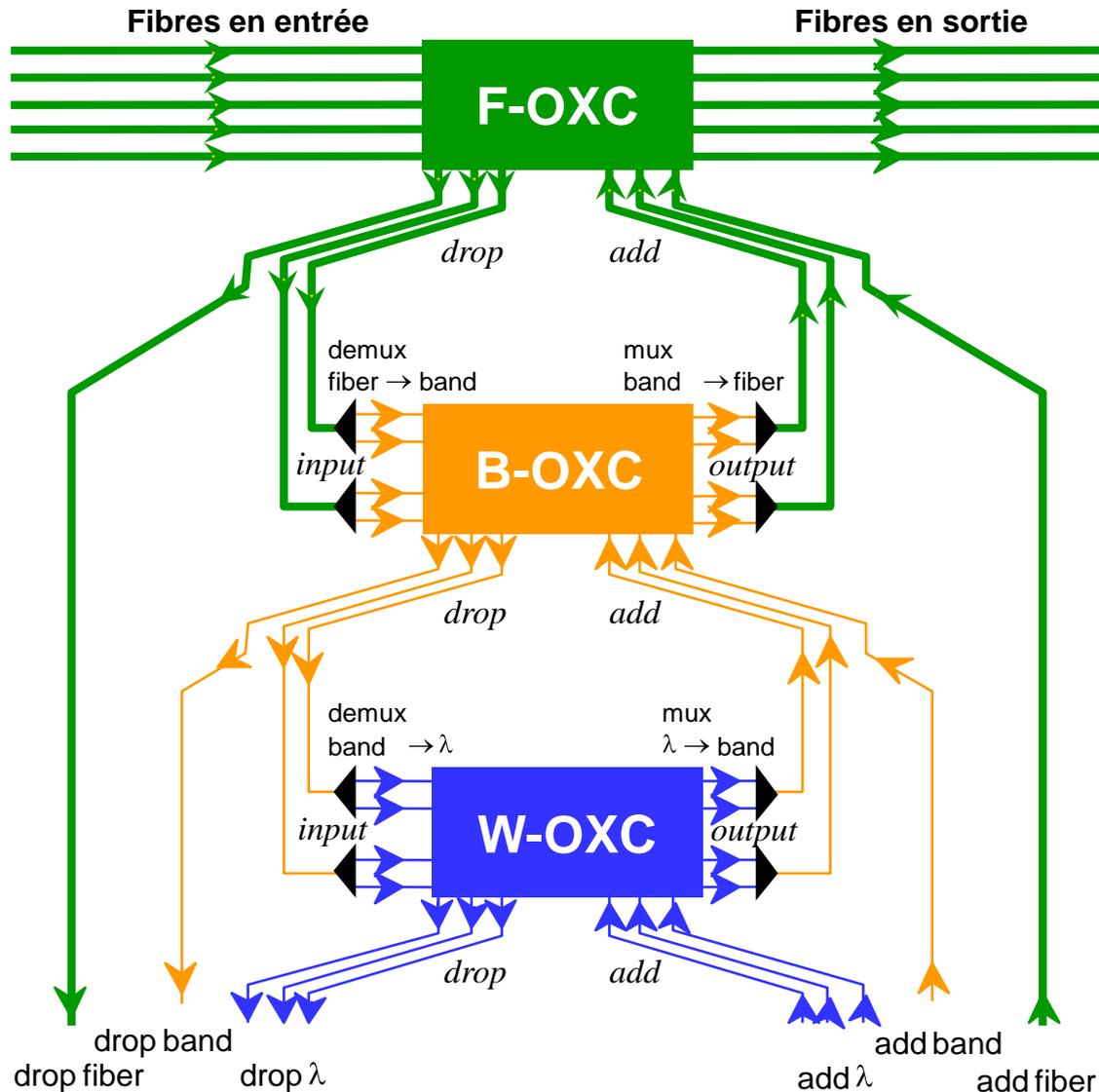
Bande



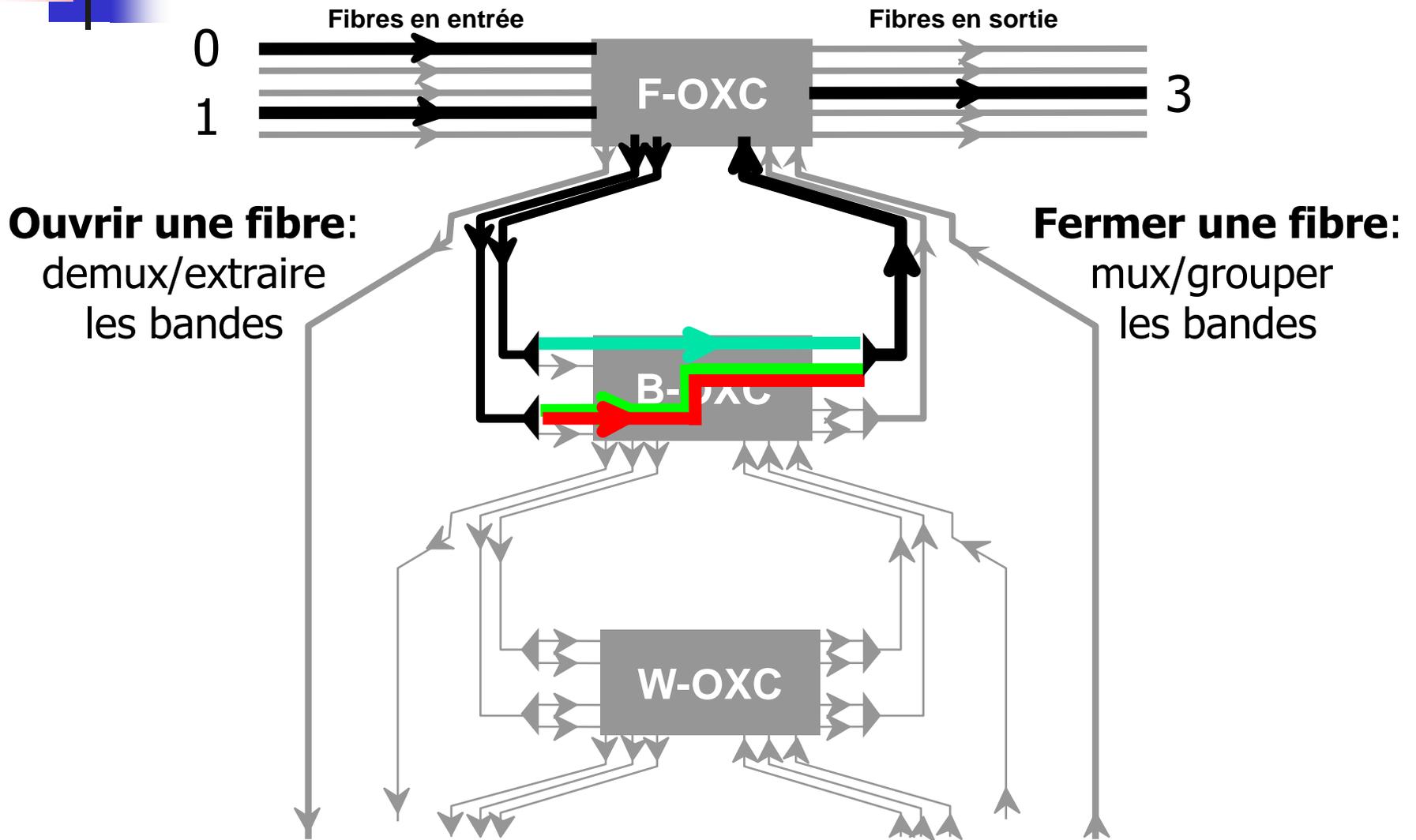
Longueur d'onde



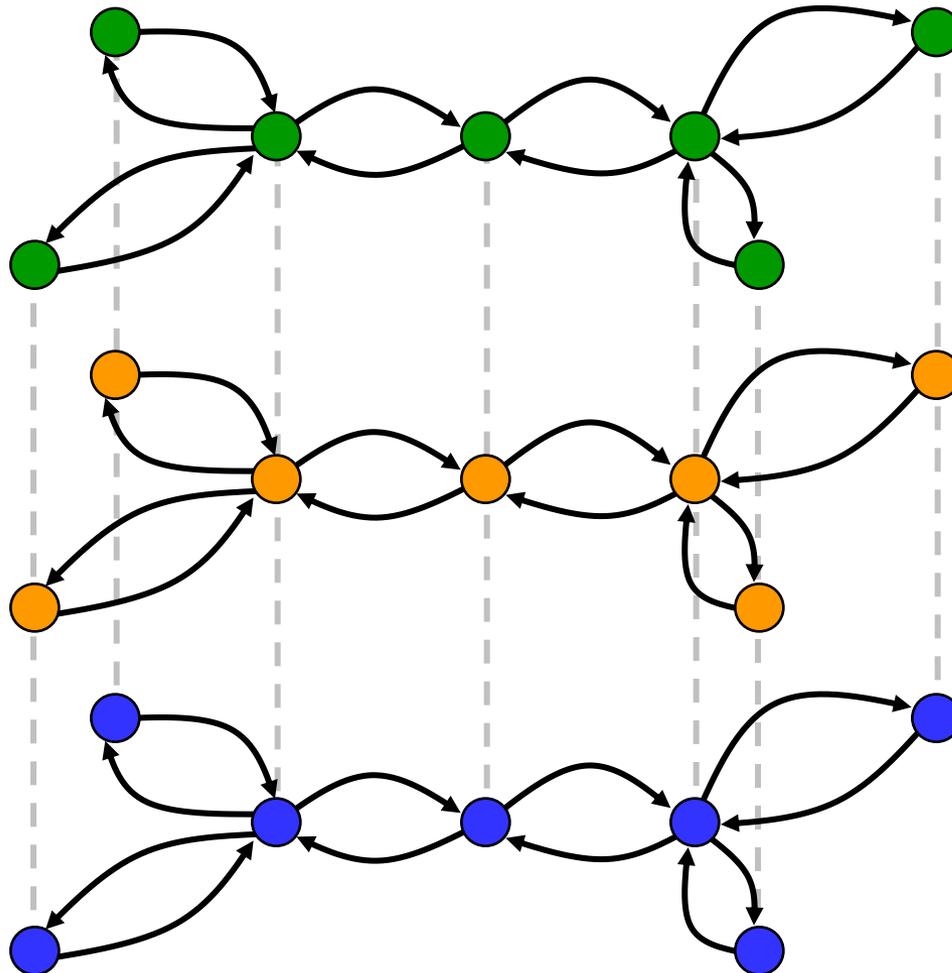
Modèle fonctionnel des nœuds



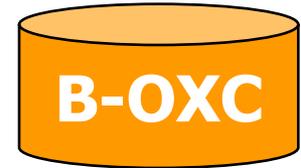
Modèle fonctionnel des noeuds



Réseau en couche



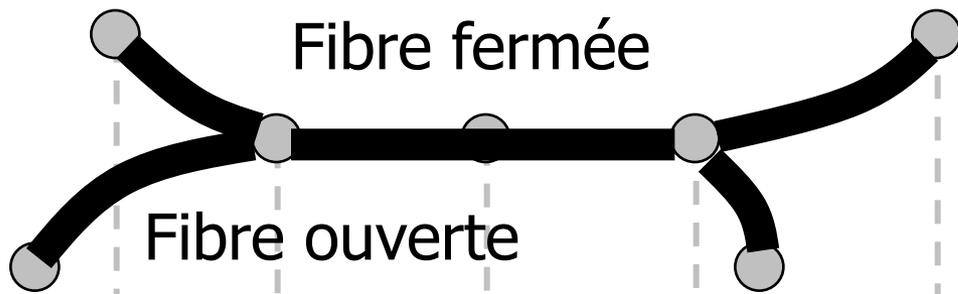
G: rés. physique



D: lightpaths

Réseau en couche

Coût des nœuds:
fonction du degré
des OXC

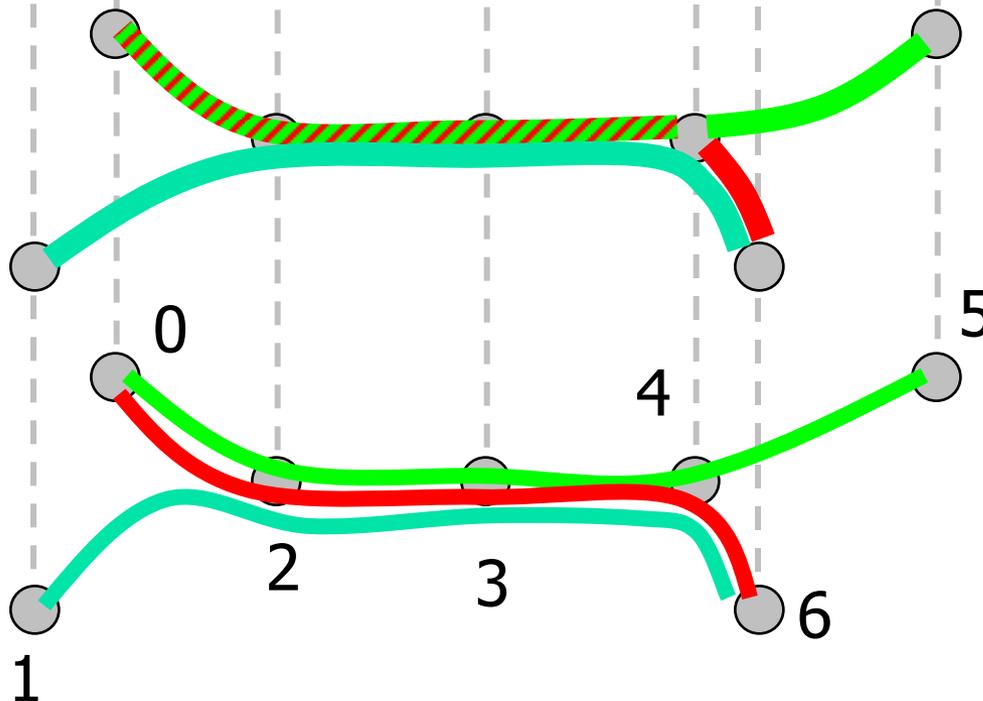


G: rés. physique



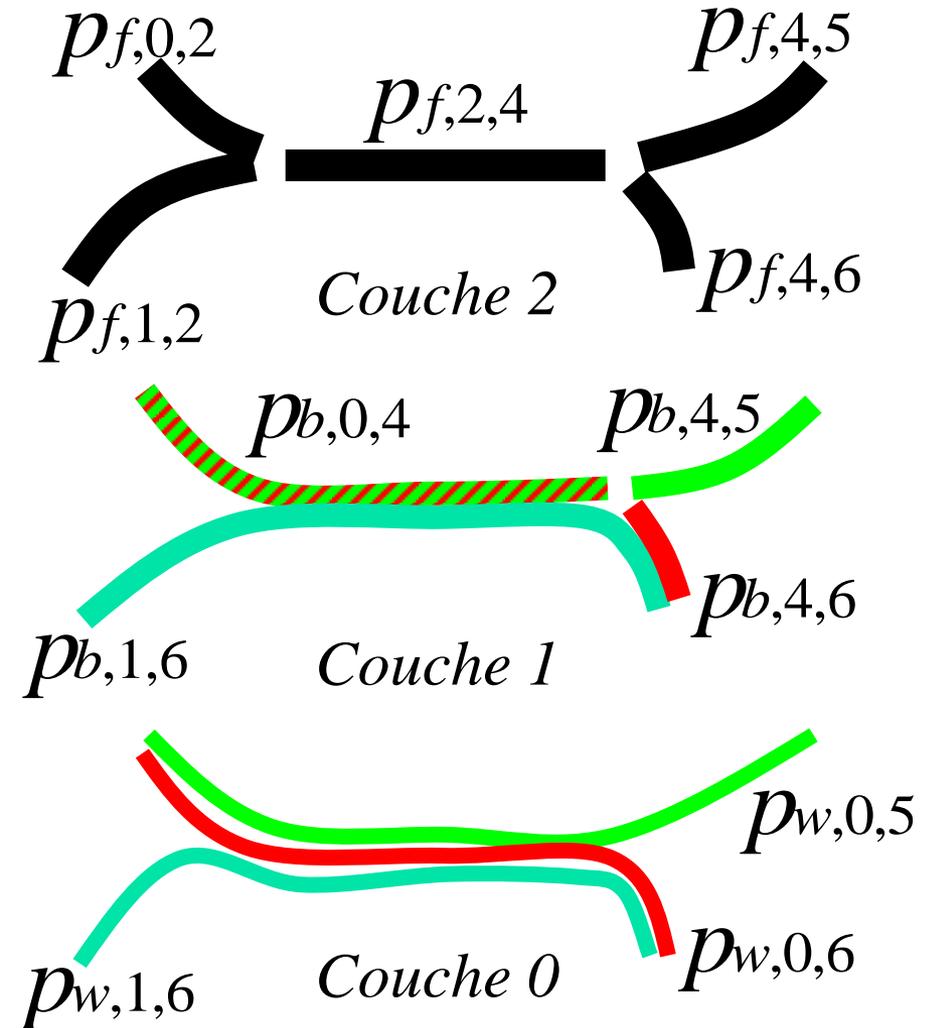
D: lightpaths

Capacités:
 $W=1$
 $B=2$
 $F=2$



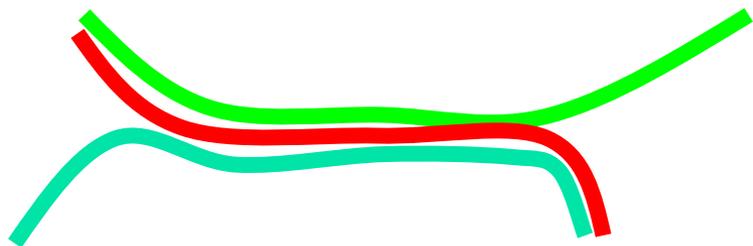
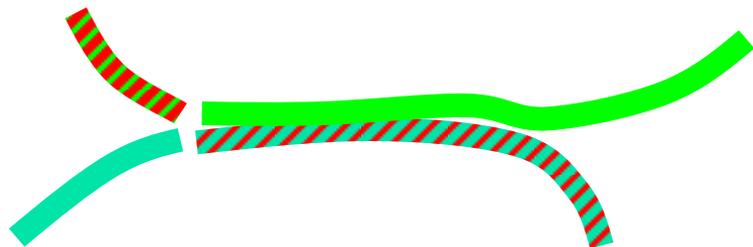
Définition de tubes

- Un chemin contigu au sein d'une même couche
- Définition réursive
 - Un tube de la couche i est une suite de tubes de la couche $i+1$
 - Exemple

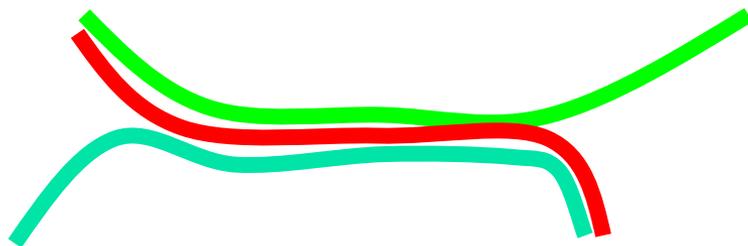
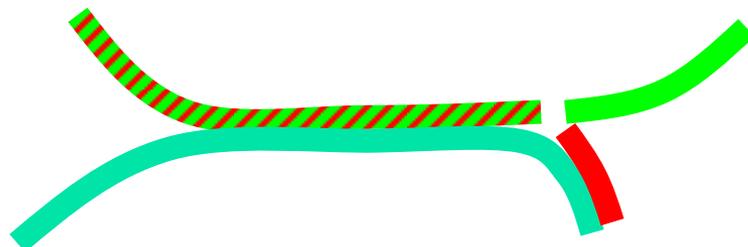


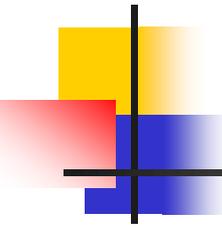
Exemples de groupage

Groupage (b)



Groupage (a)





Problème

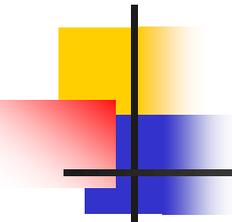
- **Entrée:**

- Ensemble de tubes munis de coûts, candidats pour être utilisés dans la couche $i+1$
- Ensemble de demandes: tubes de la couche i

- **Sortie:**

Un ensemble *min-cost* de tubes de la couche $i+1$ qui transportent les tubes de I sous contraintes de capacités

- Défini sur seulement deux couches: on peut itérer le processus
- Modèle simplifié mais qui permet une résolution efficace sur des problèmes de taille réelle

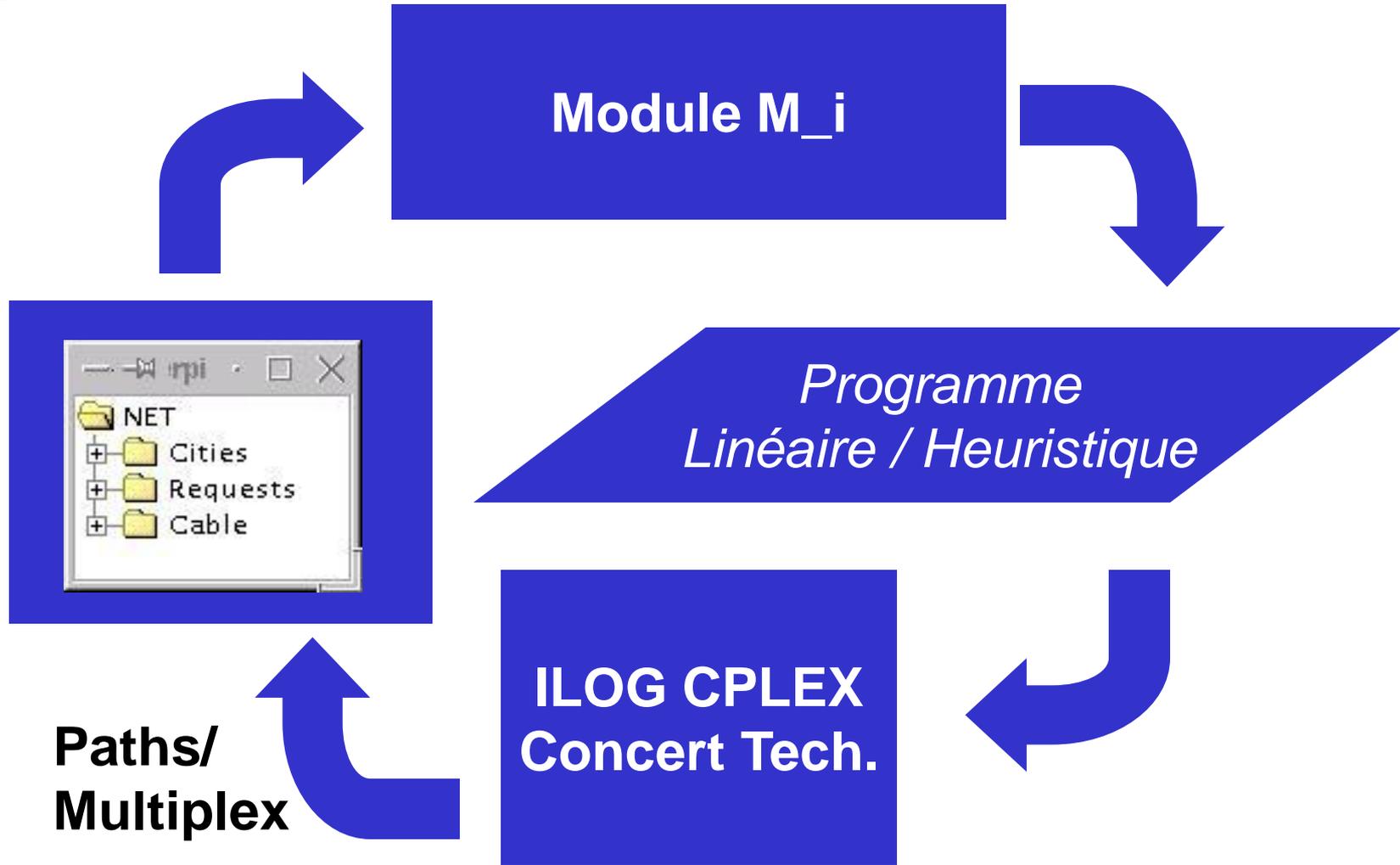


Problème global

Réaliser les meilleurs routages ET groupages possible pour minimiser le coût des équipements dans les nœuds:
Degré des W-OXC, B-OXC, F-OXC

Problème trop dur: découpage en deux phases que l'on peut itérer

Modules de Routage et Groupage



MakeFrame [default.lst]

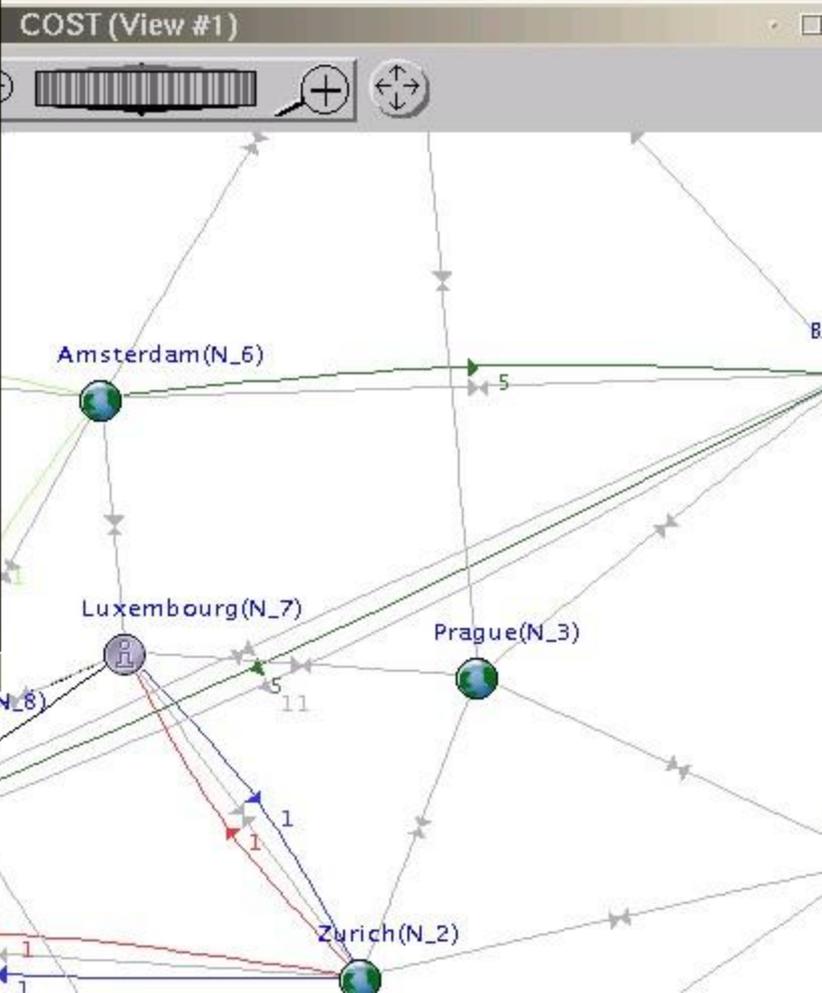
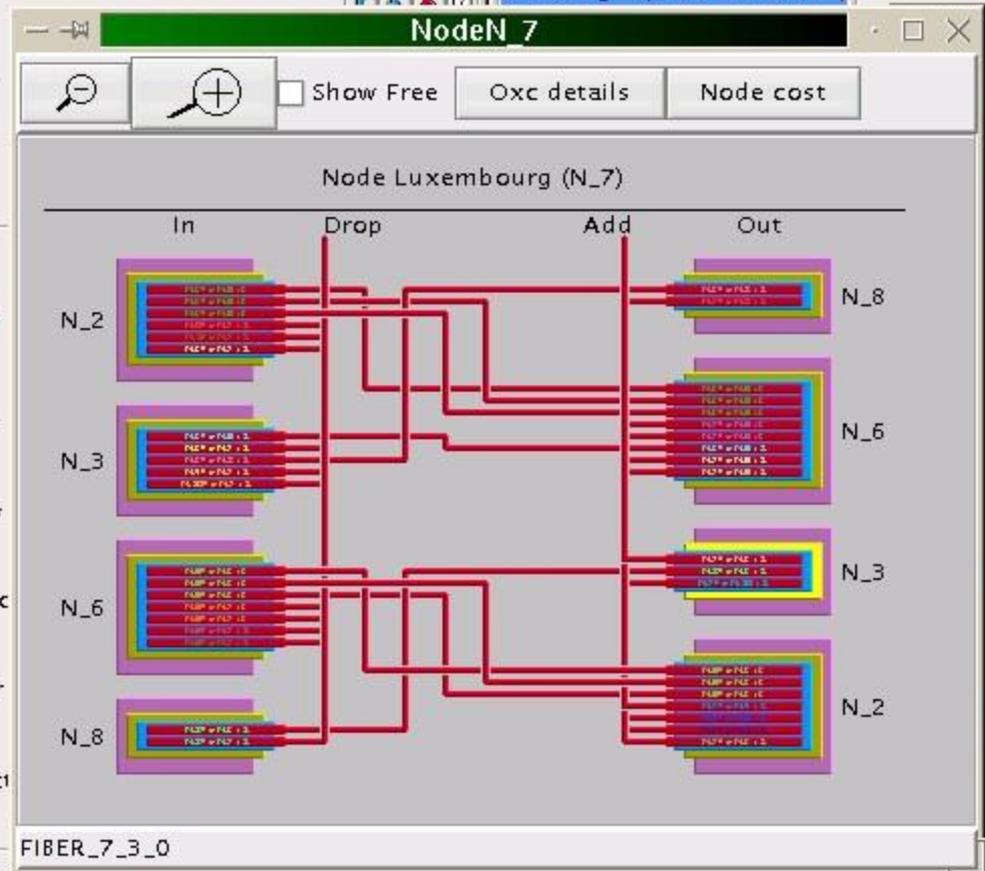
[Starting plugin] : rc
 Changing order: demand
 order.
 Changing objective fun
 Routing 110 requests.
 Solving Cplex linear

PORTO COST

File View Tools Edit

Step: + 0 -

N_0=>N_5 :11	Set Path	Main	Protection
>N_0 :11	Set Path	Main	Protection
>N_6 :5	Set Path	Main	Protection
>N_0 :5	Set Path	Main	Protection



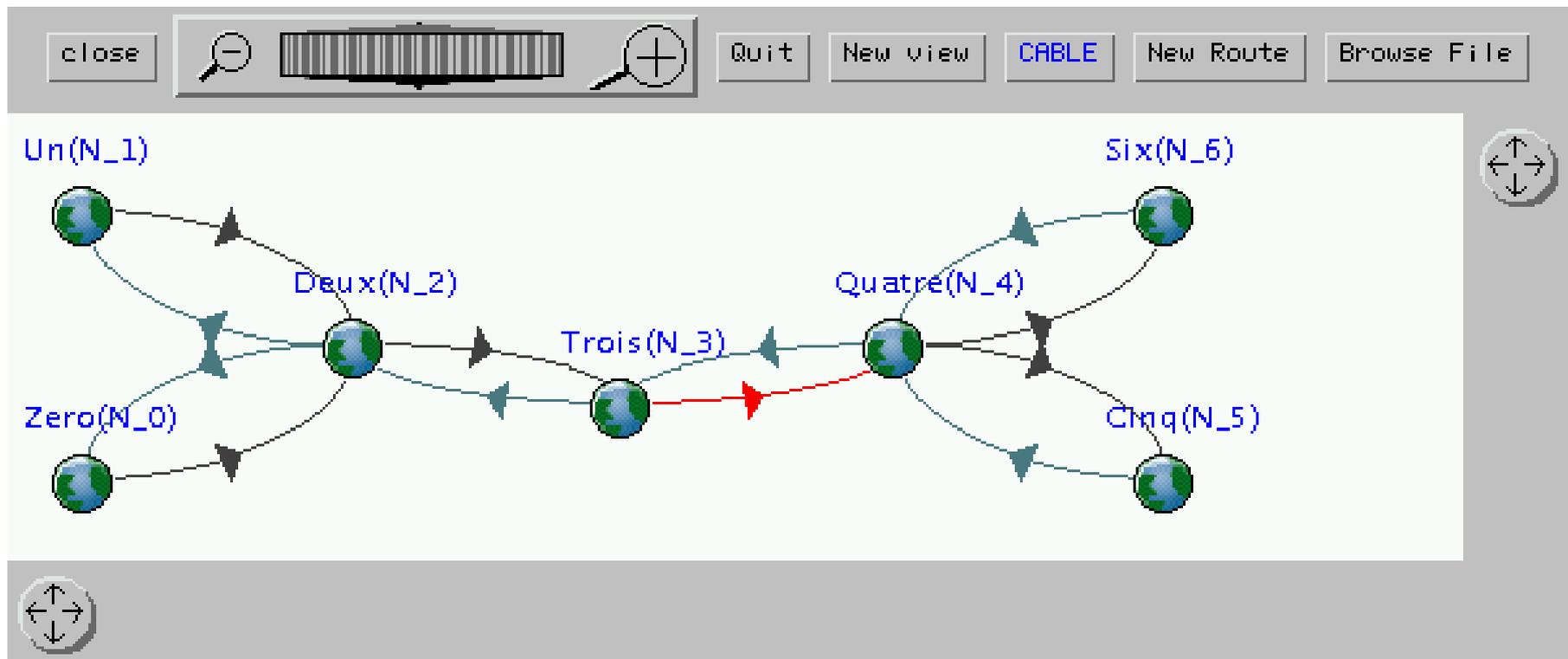
coming by size

closing factor 0

Démonstration de l'outil

Présentation d'un exemple simple:

Deux requêtes: N_1 à N_6 (10λ) et N_0 à N_5 (22λ).



NET

Cities

Requests

Cable

CABLE CABLE_0_2 length=100

CABLE CABLE_1_2 length=100

FIBER F_1_2_0 InputState:open Prev:null OutputState:open Next:null

BAND B_1_2_0_0 InputState:close Prev:B_1_1_0_0 OutputState:close Next:B_2_3_0_3

LAMBDA L_1_2_0_0_0 N_1=>N_6

LAMBDA L_1_2_0_0_1 N_1=>N_6

LAMBDA L_1_2_0_0_2 N_1=>N_6

LAMBDA L_1_2_0_0_3 N_1=>N_6

LAMBDA L_1_2_0_0_4 N_1=>N_6

LAMBDA L_1_2_0_0_5 N_1=>N_6

LAMBDA L_1_2_0_0_6 N_1=>N_6

LAMBDA L_1_2_0_0_7 N_1=>N_6

BAND B_1_2_0_1 InputState:open Prev:null OutputState:open Next:null

LAMBDA L_1_2_0_1_0 N_1=>N_6 Prev:L_1_1_0_0_0 Next:L_2_3_0_2_6

LAMBDA L_1_2_0_1_1 N_1=>N_6 Prev:L_1_1_0_0_0 Next:L_2_3_0_2_7

LAMBDA L_1_2_0_1_2

LAMBDA L_1_2_0_1_3

LAMBDA L_1_2_0_1_4

LAMBDA L_1_2_0_1_5

LAMBDA L_1_2_0_1_6

LAMBDA L_1_2_0_1_7

BAND B_1_2_0_2

BAND B_1_2_0_3

FIBER F_1_2_1

CABLE CABLE_2_0 length=100

CABLE CABLE_2_1 length=100

CABLE CABLE_2_3 length=100

CABLE CABLE_3_2 length=100

CABLE CABLE_3_4 length=100

CABLE CABLE_4_3 length=100

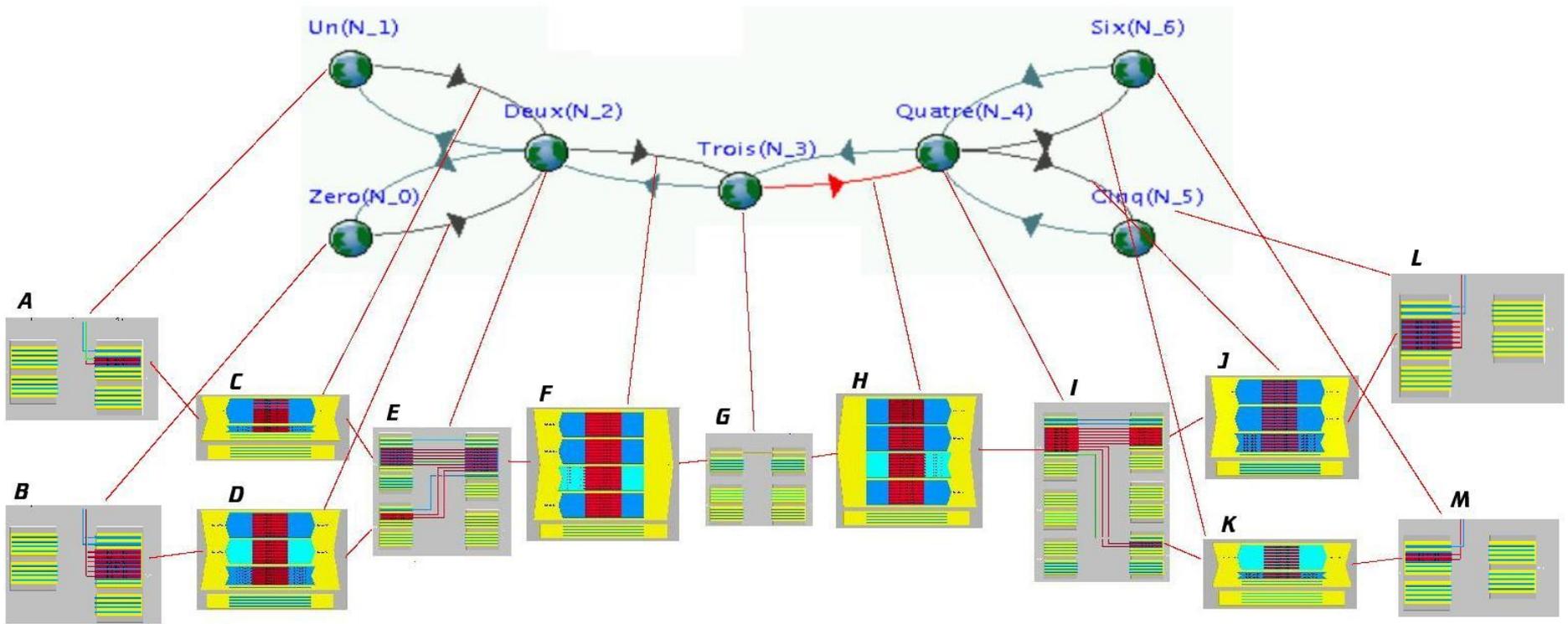
CABLE CABLE_4_5 length=100

CABLE CABLE_4_6 length=100

CABLE CABLE_5_4 length=100

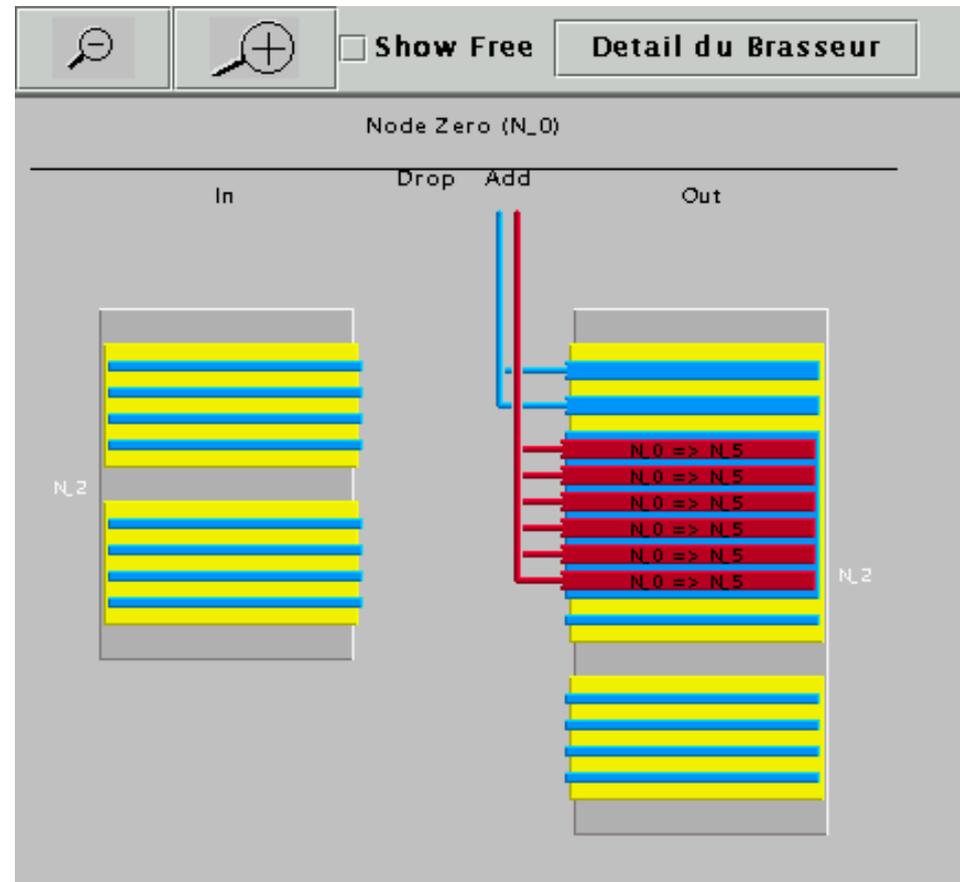
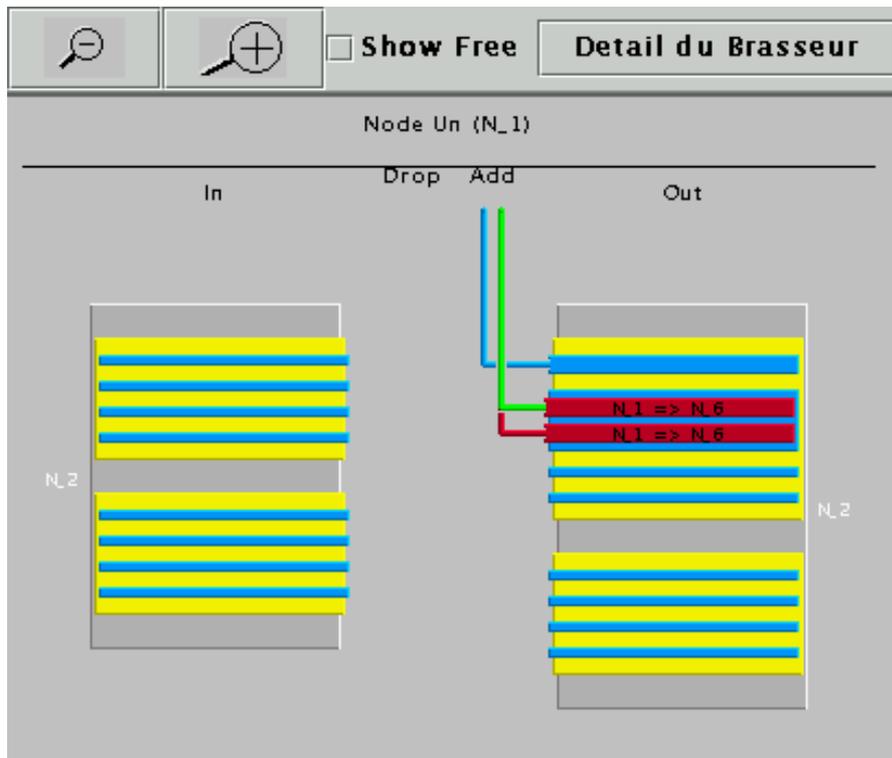
CABLE CABLE_6_4 length=100

Vues détaillées dans PORTO



Nœuds N_1 et N_0

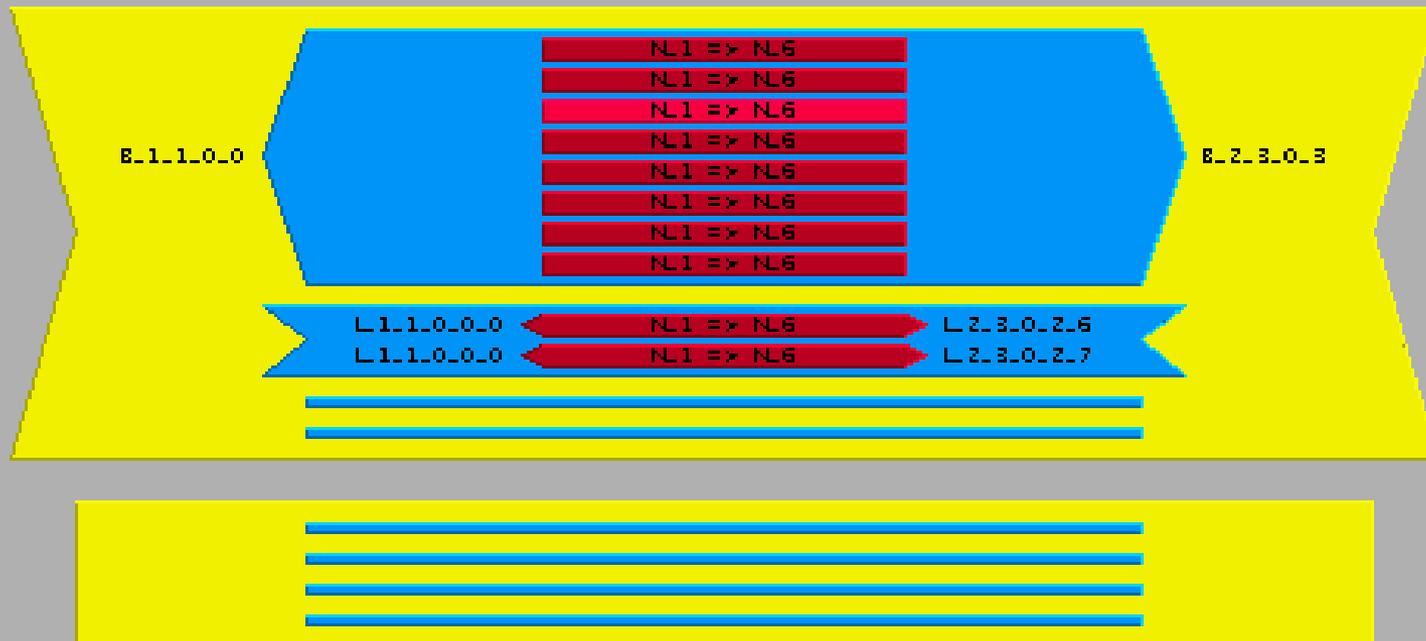
- 1 fibre (jaune) : 4 bandes
- 1 bande (bleue) : 8 λ (rouges)



Câble de N_1 à N_2

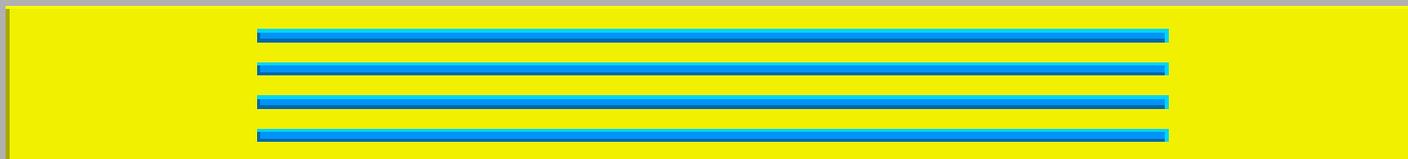
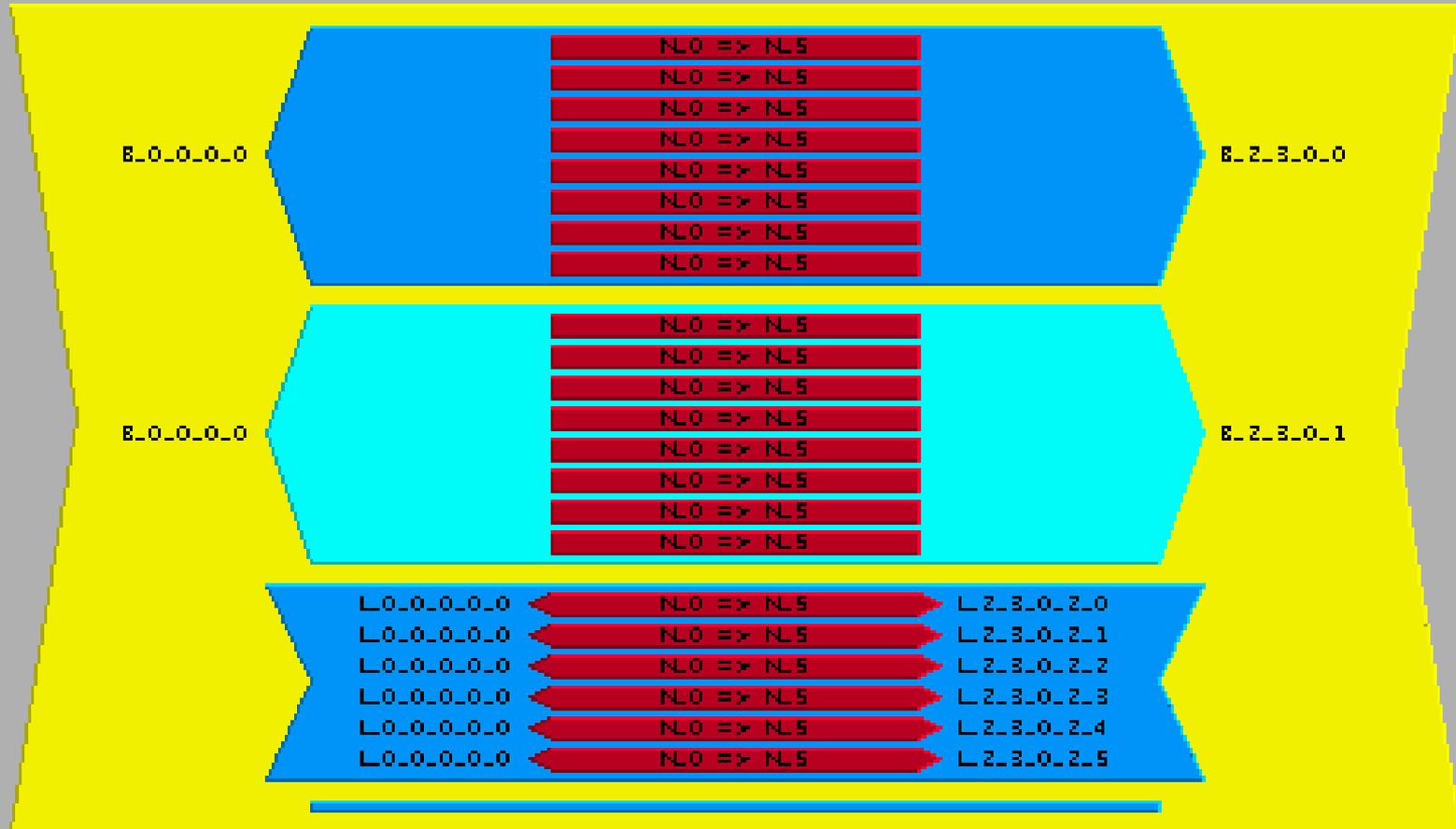


Show Free



L_1_2_0_0_2 : [In Open] [Out Open]

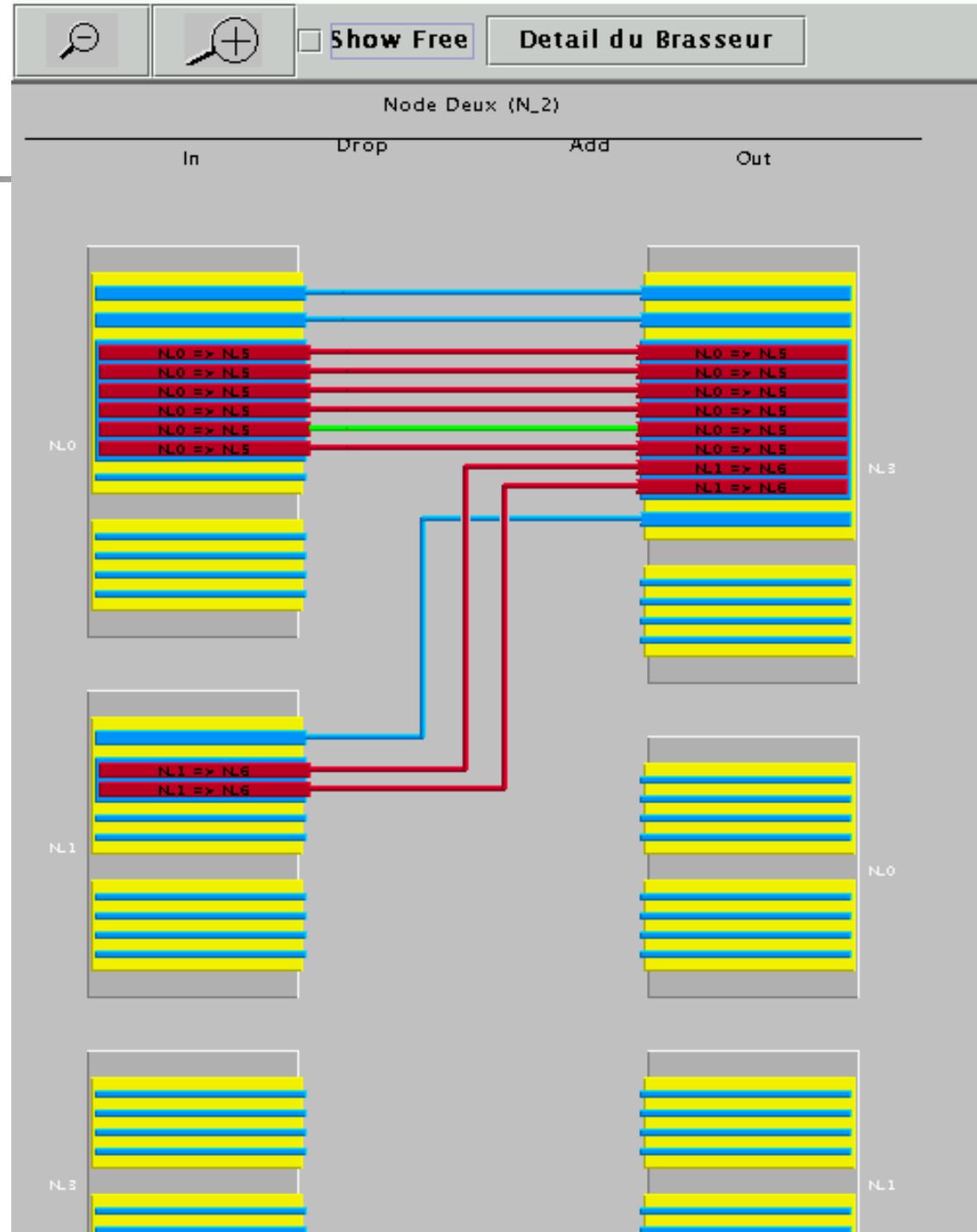
Câble de N_0 à N_2



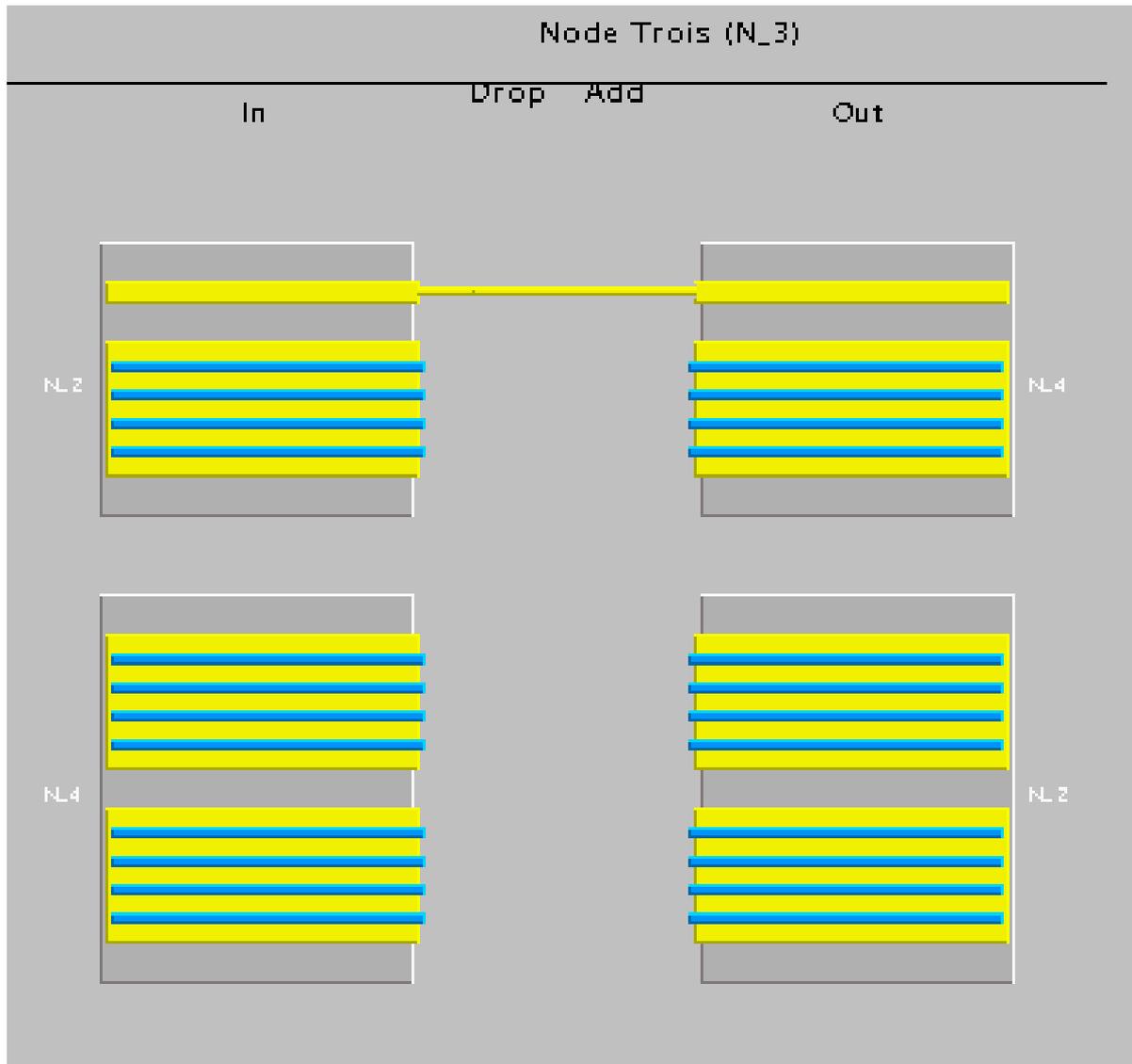
N_2

✓ 6 λ de la requête N_0 \rightarrow N_5 sont groupées dans une même bande avec 2 λ de la requête N_1 \rightarrow N_6

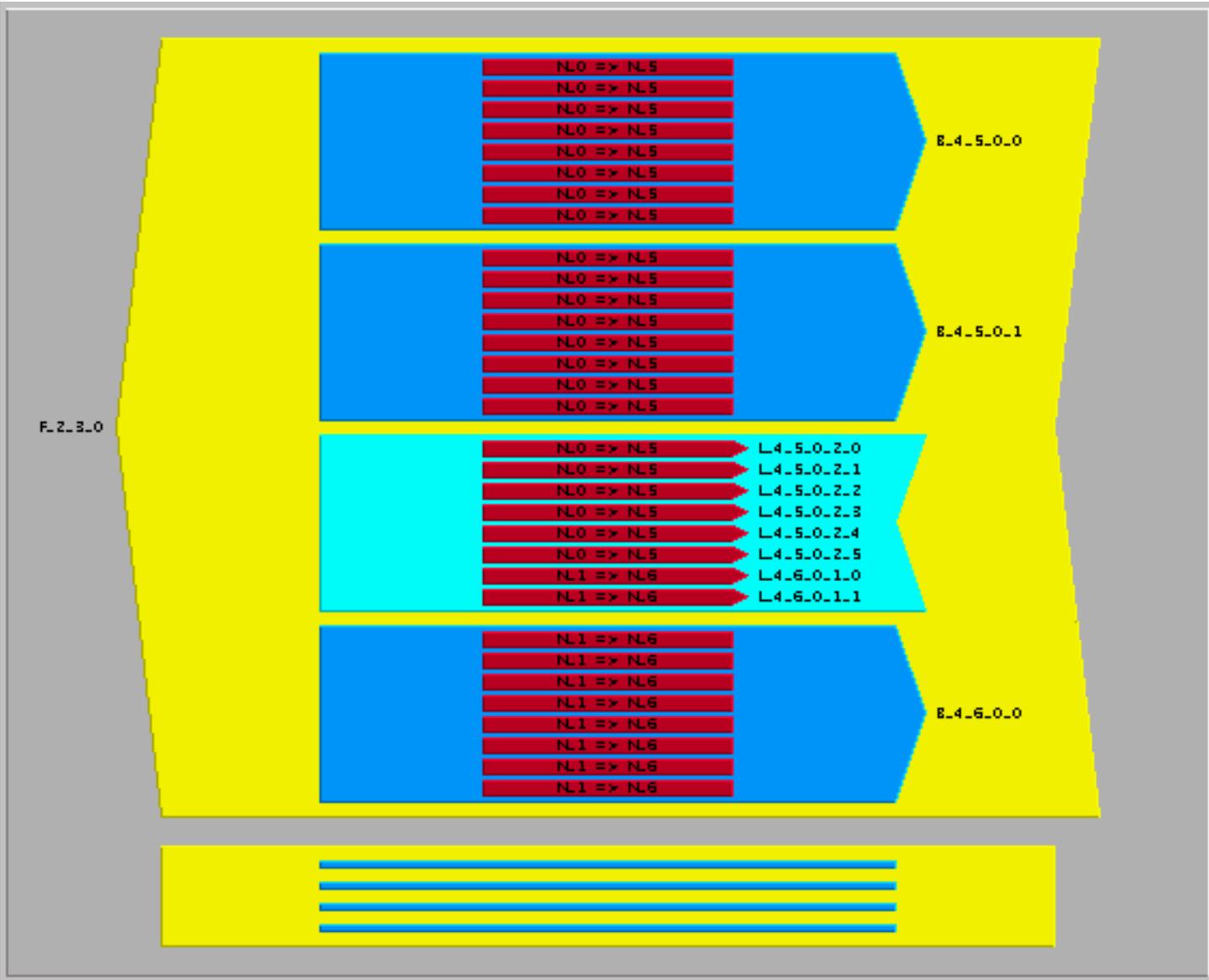
✓ 1 bande de la requête N_1 \rightarrow N_6 est groupée dans la même fibre avec les 3 bandes de la requête N_0 \rightarrow N_5



N_3 : niveau F-OXC



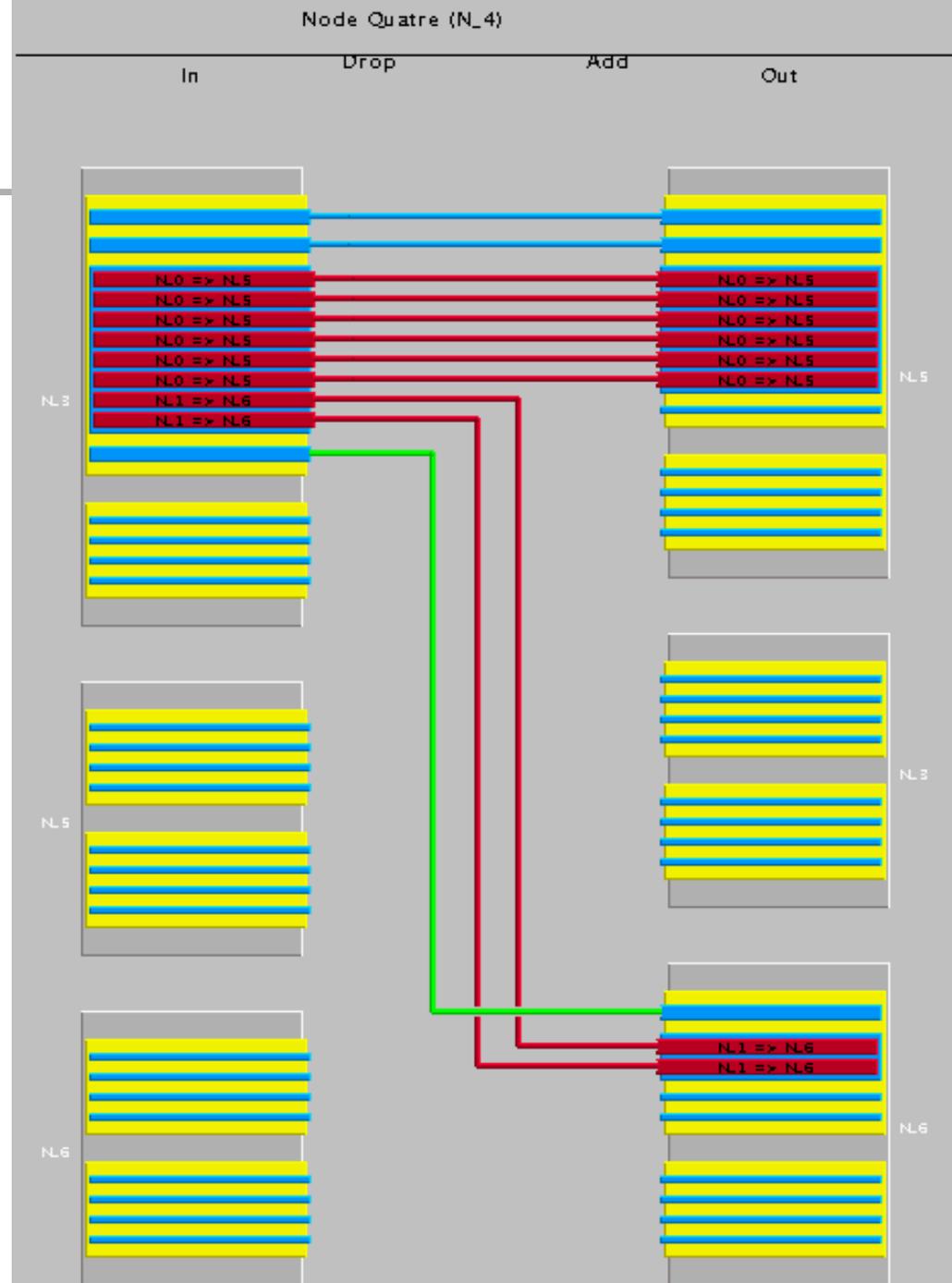
Câble de N_3 à N_4



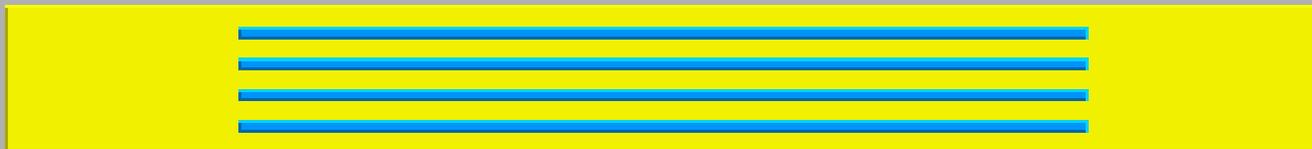
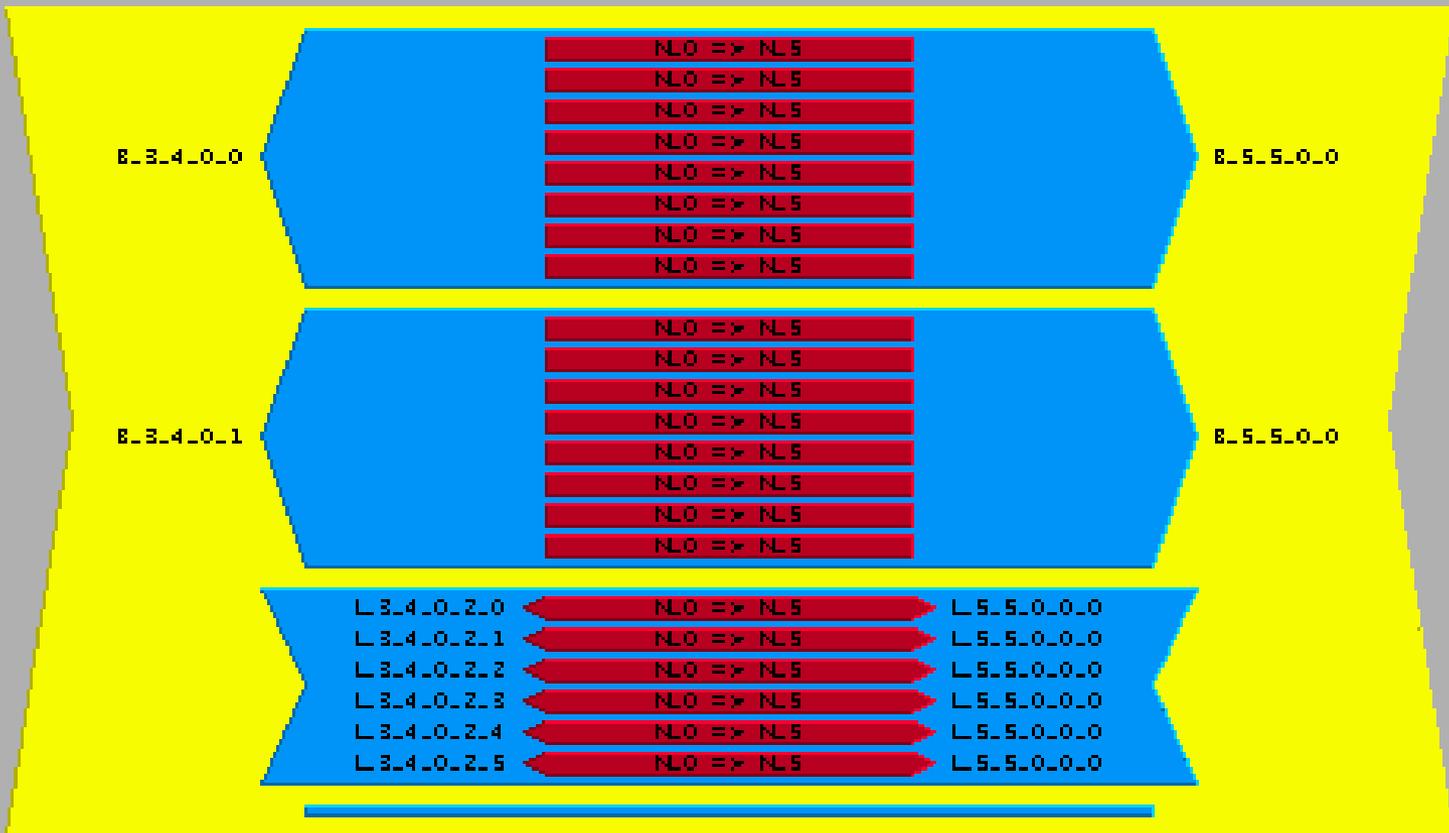
N_4

On doit extraire la bande
et les deux λ groupées
en N_2 :

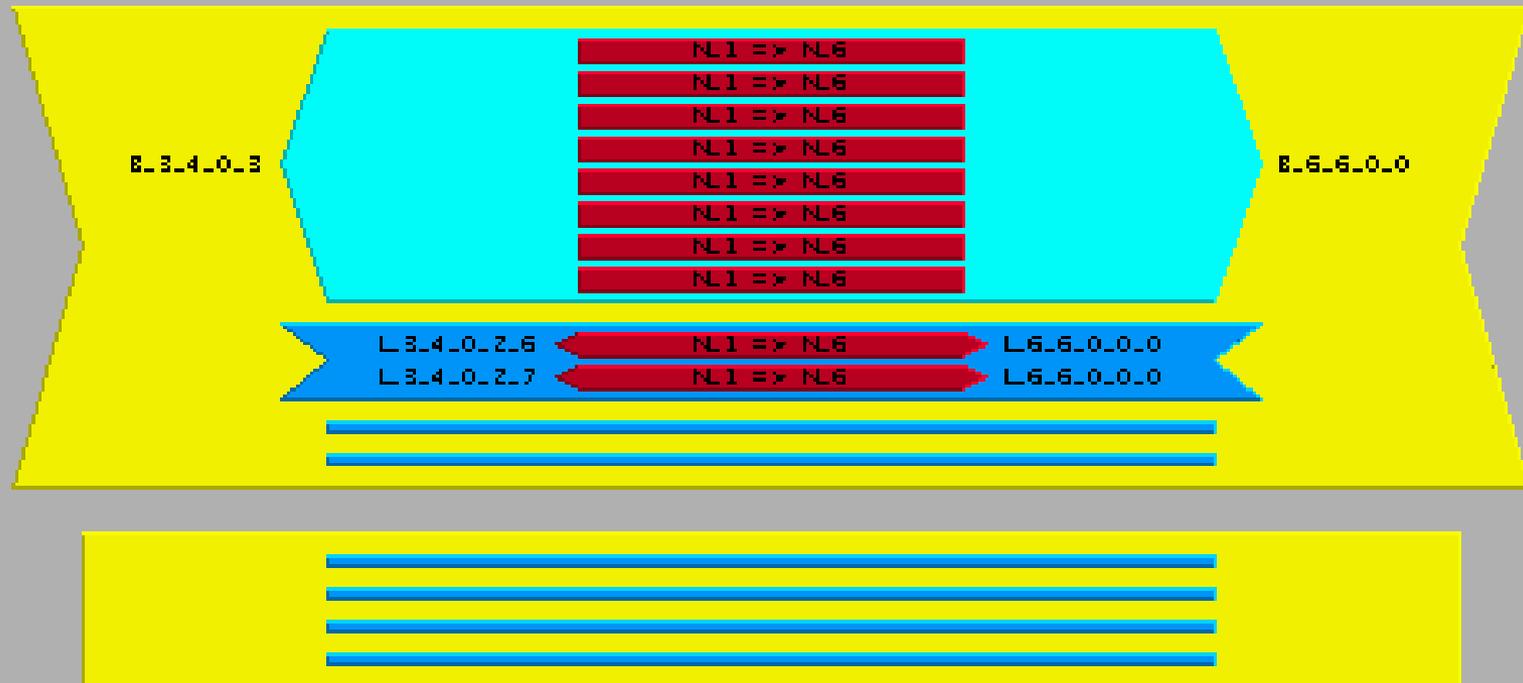
B-OXC et W-OXC



Câble de N_4 à N_5



Câble de N_4 à N_6



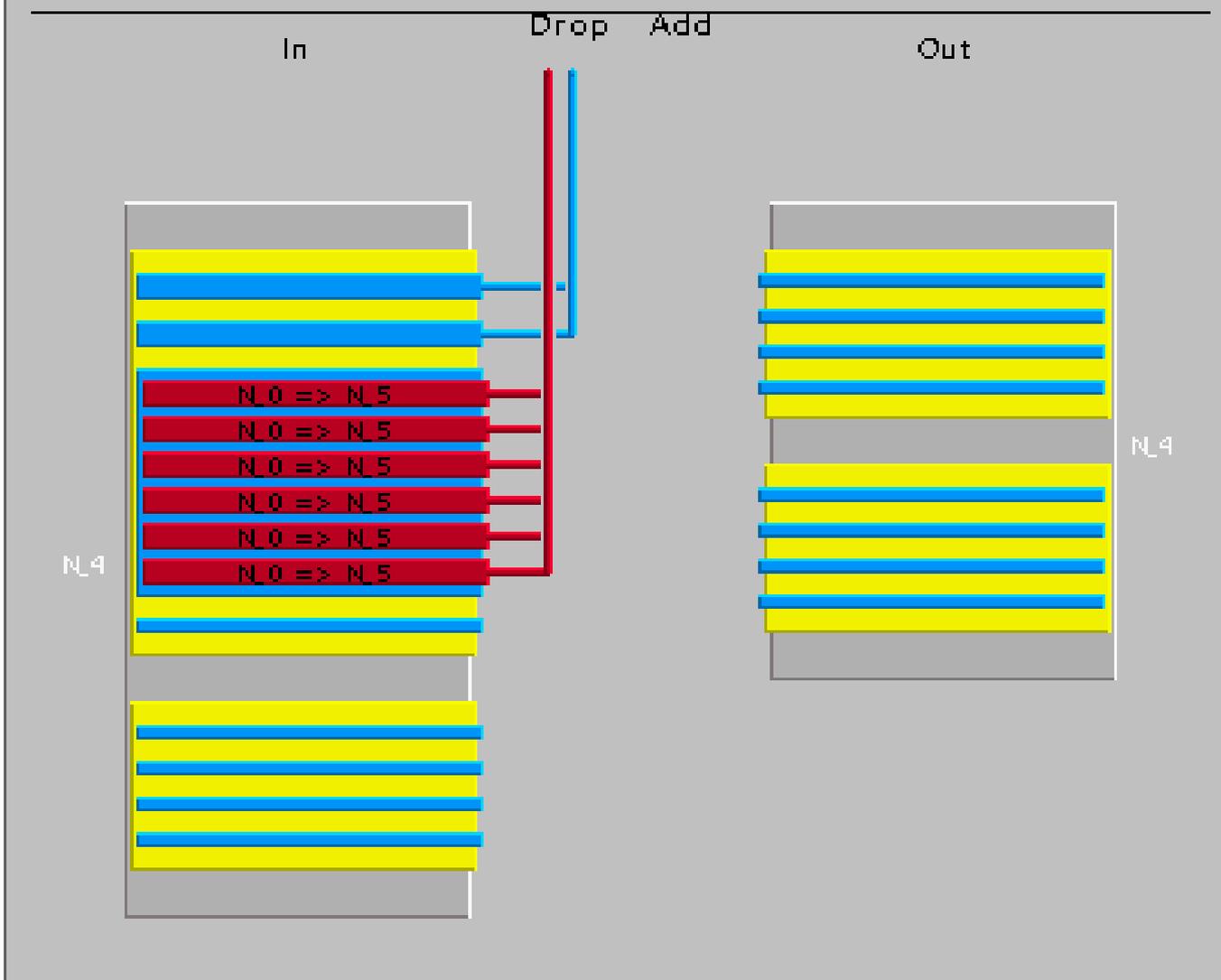
N_5



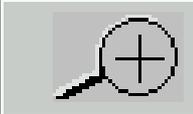
Show Free

Detail du Brasseur

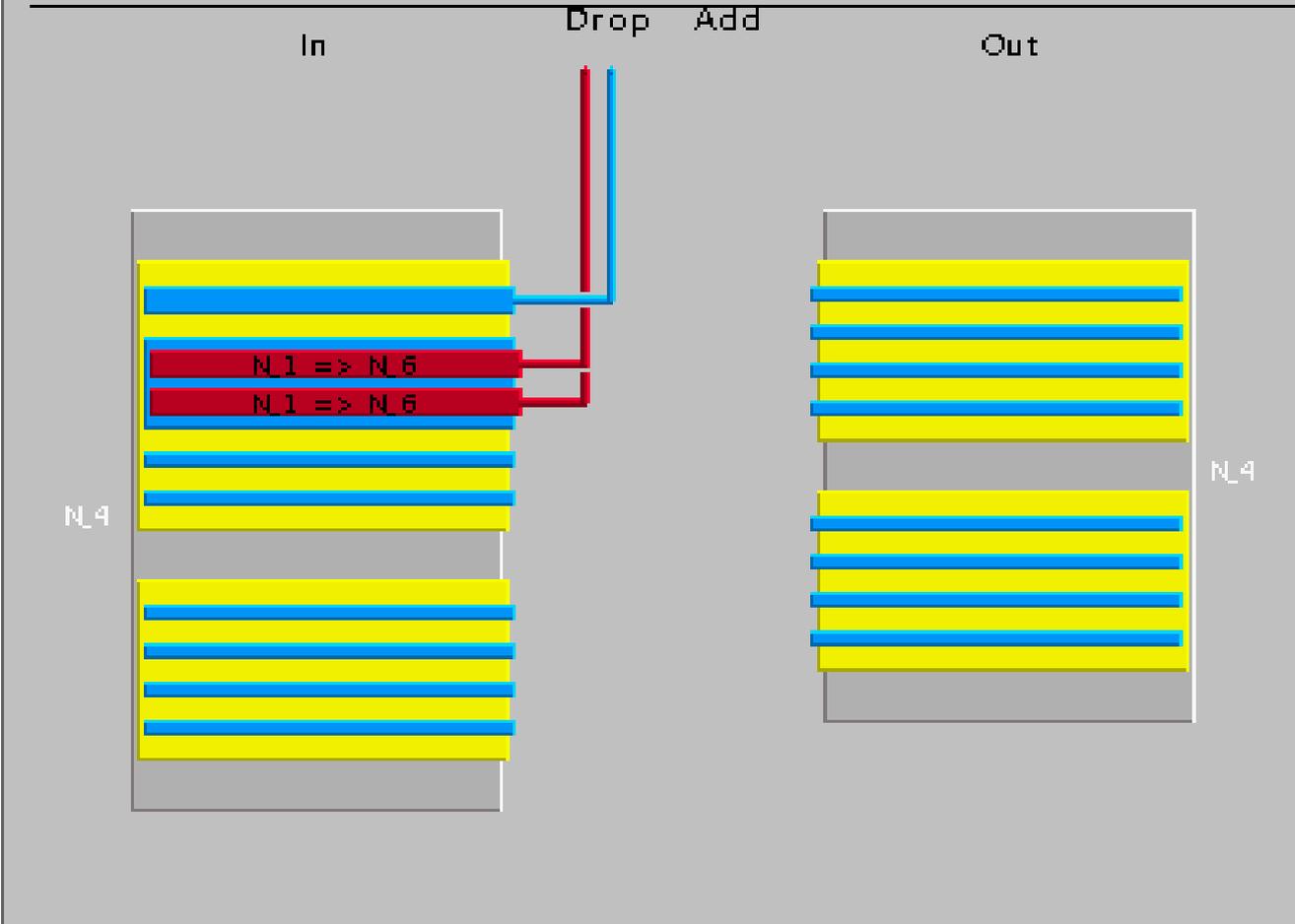
Node Ginq (N_5)

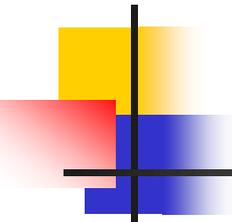


N_6

  Show Free Detail du Brasseur

Node Six (N_6)





Conclusion

- Outils permettant de traiter des réseaux réels (France, Europe, USA)
- Ajout de nouveaux modules
- Perspectives:
 - PORTO-2, Mascopt, ...