Peix al cove: $\mathbf{c}_{\mathbf{ov}}^{\mathrm{peix}}\mathbf{e}$

Producció Encapsulada i Incremental de Xarxes de demostració

(Afegit Lingüístic)

Construcció Opcional de Valors i Etiquetes

Josep M. Merenciano
Departament de Llenguatges i Sistems Informàtics
Universitat Politècnica de Catalunya

12 de juliol de 1999

Resum

El conjunt de macros $\mathbf{c}_{\mathbf{ov}}^{\mathrm{peix}}\mathbf{e}$ amplia el paquet $\mathbf{E_LX}$ de construcció de xarxes de demostració en dos àmbits principals: el $^{\mathrm{peix}}$ que introdueix el concepte de bloc i aporta les eines necessàries per definir xarxes a partir dels blocs; i el $\mathbf{c}_{\mathbf{ov}}\mathbf{e}$ que dóna versaltilitat a l'ús de les etiquetes.

The $\mathbf{c}_{\mathbf{ov}}^{\mathrm{peix}}\mathbf{e}$ macro set enhances $\mathbf{E_LX}$ in two aspects: $^{\mathrm{peix}}$ introduces the concept of a bloc and the tools to build up proofnets with blocs; $\mathbf{c}_{\mathbf{ov}}\mathbf{e}$ gives us more flexibility in using labels.

Taula de continguts

1 El peix i el cove **3**
1.1 El $\mathbf{c}_{\mathbf{ov}}\mathbf{e}$ 3
1.2 El $^{\mathrm{peix}}$ 3

2 Construcció d'un nucli **4**
2.1 Informació continguda en un nucli 4
2.2 Permutació del nus 6
2.3 Categoria sintàctica 7

2.4 Tipus semàntic 9
2.5 Etiqueta semàntica 11
2.6 Variables de l'abstracció 11
2.7 Ordre prosòdic 12
2.8 Lligam 13
2.9 Numeració dels nodes d'una xarxa 14
2.10 Exemple 15

3 Composició de xarxes **17**
3.1 El concepte de bloc 17
3.2 Construcció de blocs 17
3.3 Construcció de xarxes compostes 18
3.4 Exemples 21

4 Comandes d'alt nivell **25**

5 Nous nusos **25**

A El nom sí el fa la cosa **26**
A.1 Sobre l'evolució de les espècies 26
A.2 No tot són flors i violes 27

B Modificacions a E$_{\mathbf{L}}$X **28**
B.1 Desbrossat i modificacions 28
B.2 El nou sistema de posicionament 31

C Macros externes **31**
C.1 Conversió de `lengths` a `commands` 32
C.2 Dibuix de línies entre dos punts qualssevol 34
C.3 Gestió de piles amb LaTeX 34

D El codi de c ov peix e **39**

1 El peix i el cove

El conjunt de macros peix es composa de dos mòduls independents entre sí: el peix i el $\mathbf{c_{ov}e}$.

1.1 El $\mathbf{c_{ov}e}$

El $\mathbf{c_{ov}e}$ introdueix el concepte de *nucli*: nus enriquit amb multitut d'etiquetes. Pròpiament es podria veure un nucli com una variant dels nusos estiquetats d'$\mathbf{E_LX}$, i per tant corresponent al nivell nuclear (N=0) de $\mathbf{E_LX}$. Dues raons ens han portat però a implementar-ho com una capa sobre $\mathbf{E_LX}$, que anomenem N=3:

1. *La integritat d'*$\mathbf{E_LX}$. Es tracta d'afegir funcionalitats a $\mathbf{E_LX}$ sense haver-ne de canviar el seu nucli.
2. *Semàntica.* Les etiquetes definides en els nusos d'$\mathbf{E_LX}$ eren purs contenidors sintàctics, i era responsabilitat de l'usuari el situar-hi continguts coherents. En els nuclis del $\mathbf{c_{ov}e}$ cada etiqueta se li ha preassignat una interpretació semàntica; això permet que internament el tractament de cada etiqueta pugui tenir les característiques més adients a la seva semàntica. Per exemple, les etiquetes de numeració es poden generar automàticament; així com algunes de les etiquetes de les categories sintàctiques.

Els nuclis de $\mathbf{c_{ov}e}$ parteixen dels nusos etiquetats d'$\mathbf{E_LX}$; un cop definit un nucli amb tota la seva informació associada hom pot activar o desactivar a voluntat l'aparició de les etiquetes sobre el nus; és a dir, es distingueix entre la càrrega informativa associada a cada nus (el nucli), i quina d'aquesta informació es presenta en cada cas particular.

1.2 El peix

En $\mathbf{E_LX}$ tot nus ha d'anar dins d'una xarxa; això dificulta la construcció de xarxes a partir d'altres xarxes prèviament definides. El peix introdueix el concepte de bloc, com a mecanisme per construir xarxes a partir de sub–xarxes prèviament definides, i això de manera força transparent a l'úsuari: un bloc pot constituir una xarxa per sí mateix, o pot formar part d'una xarxa juntament amb d'altres blocs.

De fet l'aportació bàsica del peix és l'entorn `bloc`, que crida l'entorn **`xarxa`** només quan aquest bloc no s'usa en composició amb d'altres per formar una sola xarxa. El resultat és que ara hom disposa de nusos (o nuclis) força més complexos per a la construcció de xarxes: els blocs definits per l'usuari.

Si l'usuari defineix els blocs seguint unes pautes estàndard, llavors el mode d'*autoposicionament* del peix permet situar automàticament cada bloc un al costat de l'altre sense que es produeixi solapament

Com a comentari al marge cal notar l'existència d'una problemàtica de sobreeiximent de la pila de paràmetres del LaTeX, que apareix quan hom menys s'ho espera. L'apèndix A, pàgina 26, en dóna alguns detalls.

2 Construcció d'un nucli

2.1 Informació continguda en un nucli **4**
2.2 Permutació del nus **6**
2.2.1 Tipus de permutació 6
2.2.2 Activació de la permutació automàtica 7
2.3 Categoria sintàctica **7**
2.3.1 Contenidors de la categoria sintàctica 7
2.3.2 Activació de la categoria sintàctica 8
2.3.3 Construcció de la categoria sintàctica 8
2.4 Tipus semàntic **9**
2.4.1 Contenidors del tipus semàntic 9
2.4.2 Activació del tipus semàntic 9
2.4.3 Construcció del tipus semàntic 10
2.5 Etiqueta semàntica **11**
2.5.1 Contenidor de l'etiqueta semàntica 11
2.5.2 Activació de l'etiqueta semàntica 11
2.6 Variables de l'abstracció **11**
2.6.1 Contenidors de les variables de l'abstracció 11
2.6.2 Activació de les variables de l'abstracció 11
2.6.3 Construcció de les variables de l'abstracció 12
2.7 Ordre prosòdic **12**
2.7.1 Contenidors de l'ordre proòdic 12
2.7.2 Activació de l'ordre proòdic 12
2.7.3 Construcció de l'ordre proòdic 13
2.8 Lligam **13**
2.8.1 Contenidors del lligam 13
2.8.2 Activació del lligam 13
2.8.3 Construcció del lligam 13
2.9 Numeració dels nodes d'una xarxa **14**
2.9.1 Contenidors de la numeració 14
2.9.2 Activació de la numeració 14
2.9.3 Construcció de la numeració 15
2.9.4 Ús de la numeració 15
2.10 Exemple **15**

2.1 Informació continguda en un nucli

El nucli d'una xarxa és un dels nusos definits, amb informació addicional. La taula 1 mostra la informació continguda en un nucli; la figura 1 mostra amb un exemple la ubicació de cadascuna d'aquestes informacions; la taula 2 mostra la comanda `\nucli` i els seus paràmetres. El paràmetre "nus de partida" correspon al nus que hom vol que constitueixi el nucli; si la permutació automàtica està activada llavors, internament, en funció de la connectiva es decideix si el nus del nucli és el passat per paràmetre, o bé el corresponent permutat.

Taula 1 Informació present en els nuclis d'una xarxa

Tipus d'informació	Posició	Activació
Categoria sintàctica	Sobre les *arestes*: a l'exterior en les potes superiors; a la dreta en la inferior	`\calEtSint`(3) `\mostrarEtSint`(0) `\NOmostrarEtSint`(0)
Tipus semàntic	Lateralment en els *vèrtexs* superiors del nucli, a l'exterior	`\calTipSem`(2) `\mostrarTipSem`(0) `\NOmostrarTipSem`(0)
Etiqueta semàntica	Sota el *vèrtex* inferior	`\mostrarEtSem`(0) `\NOmostrarEtSem`(0)
Variables de l'abstracció	A l'inici de l'*aresta* argument; a l'exterior	`\mostrarVar`(0) `\NOmostrarVar`(0)
Ordre prosòdic	Entre les dues potes superiors	`\mostrarOrdre`(0) `\NOmostrarOrdre`(0)
Lligam	Entre les dues potes superiors	`\mostrarLligam`(0)
Numeració dels nodes	En els *vèrtexs*: a sota en les potes superiors del nucli; lateralment, a la dreta i un xic per sota el vèrtex, a la pota inferior	`\mostrarNumeracio`(0) `\NOmostrarNumeracio`(0)

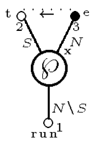


Figura 1: Situació de la informació en un nucli

Taula 2 Paràmetres de la comanda `\nucli`

Núm	Descripció	Valors possibles	
1 (opcional)	Permutació addicional	en blanc	no fa res
		qualsevol	un cop calculat el nus, el gira (permuta)
2, 3	Coordenades de xarxa		
4	Nus de partida	Nus prèviament definit	
5	Connectiva	`sobre` `sota`	
6	Categoria sintàctica de l'argumet		
7	Categoria sintàctica del resultat		
8	Parentització	Veure la taula 9	
9	Etiqueta semàntica		

La informació present en un nucli s'obté dels arguments de la comanda `\nucli` i dels valors que contenen certes comandes, que anomenem *contenidors*: es tracta de valors que caldria passar per paràmetre però que per mor de la limitació del nombre màxim de paràmetres que pot tenir una comanda, o bé per tractar-se de valors bastant estables, es gestionen com a variables globals (o paràmetres ficticis d'entorn). La majoria de contenidors es gestionen internament; externament alguns es defineixen a través dels paràmetres de la comanda `\nucli`; d'altres mitjançant comandes específiques que permeten definir els valors d'un grup de contenidors, o que dónen la informació necessària per a poder-ne construir els valors correctes.

Els *commutadors* permeten seleccionar quines de les informacions presents en un nucli es visualitzen, independentment de si s'han passat els valors per paràmetre (o contenidors) o no. Això permet definir un nucli amb tota la seva informació, i en funció de les necessitats s'activen o desactiven les visualitzacions necessàries.

L'efecte dels commutadors pot ser global o local. L'efecte d'àmbit global actúa com un paràmetre d'entorn: mentre no es digui el contrari tots els nuclis que segueixen es comportaran de la mateixa manera. L'efecte d'àmbit local permet canviar el comportament d'un nucli sense necessitar de a continuació recuperar el valor antic del commutador.

2.2 Permutació del nus

2.2.1 Tipus de permutació

Hi ha dos tipus de permutació dels nusos: l'automàtica i la manual. La permutació automàtica es pot activar i desactivar a conveniència; la necessitat o no de permutar es decideix en funció de la connectiva. La figura 2 mostra les diferents combinacions definides.

La permutació manual s'indica amb el paràmetre opcional de la comanda `\nucli` (taula 2); és independent de la permutació automàtica, i s'aplica sempre després d'aquella. La figura 3 mostra les diferents combinacions de la figura anterior, amb una permutació extra generada manualment.

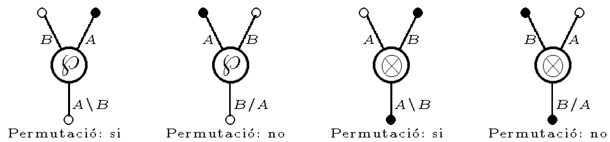


Figura 2: Permutació automática dels nusos

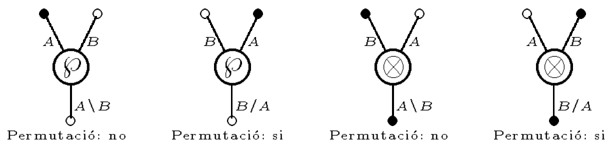


Figura 3: Permutació automática dels nusos amb permutació manual afegida

2.2.2 Activació de la permutació automàtica

Internament el control del càlcul automàtic de la permutació està controlat pel commutador de la taula 3. L'activació i desactivació externa de la permutació automàtica dels nusos es fa amb les comandes de la taula 4.

Taula 3	Commutador (intern) pel control de la permutació automàtica dels nusos
`\calPermutacioAutomatica`	
Valor dels commutador:	
si	Permutació automàtica
qualsevol altre valor	No es fa la permutació automàtica

Taula 4	Activació externa de la permutació automàtica dels nusos	
`\permutacioAutomatica(0)`	Activa la permutació automàtica dels nusos	Àmbit global
`\NOpermutacioAutomatica(0)`	Desactiva la permutació automàtica dels nusos	Àmbit global

2.3 Categoria sintàctica

2.3.1 Contenidors de la categoria sintàctica

Cada nus o nucli admet tres etiquetes sinàctiques: a la pota inferior del nus, a la pota esquerra, i a la pota dreta; els contenidors d'aquestes etiquetes són, respectivament, `\etSota`, `\etEsq` i `\etDreta`.

2.3.2 Activació de la categoria sintàctica

Internament l'aparició de cada etiqueta està controlada pels commutadors de la taula 5, que s'usen només en el cas que el meta–commutador de la taula 6 estigui activat. L'activació externa de l'aparició de les etiquetes es fa amb la comanda `\calEtSint` (taula 7), que rep per paràmetre el valor a assignar als tres commutadors, amb l'ordre sota, esquerra i dreta. L'activació (i desactivació) externa de la demanda d'usar (o no) els valors definits sobre els commutadors de les etiquetes sintàctiques es fa amb les comandes de la taula 8.

Taula 5 Commutadors (interns) pel control de la visualització de les categories sintàctiques

`\calEtSota`	Etiqueta de la pota inferior dels nusos
`\calEtEsq`	Etiqueta de la pota esquerra dels nusos
`\calEtDreta`	Etiqueta de la pota dreta dels nusos

Valor dels commutadors:

en blanc	no apareix l'etiqueta
qualsevol valor	apareix l'etiqueta

Taula 6 (Meta-)Commutador (intern) pel control de la visualització de les categories sintàctiques

`\calMostrarEtSint`	Indica si les etiquetes de les categories sintàctiques s'han de visualitzar en funció de les seves activacions o no

Valor dels commutadors:

en blanc	Tota etiqueta es considera desactivada
qualsevol valor	S'usen les activacions definides

Taula 7 Activació externa de l'aparició de les categories sintàctiques

`\calEtSint(3)`	Assigna els paràmetres als commutadors interns en l'ordre de la taula 5	Àmbit global

Taula 8 Activació i desactivació externa de l'ús de les activacions sobre les etiquetes de les categories sintàctiques

`\mostrarEtSint(0)`	Activa l'ús de les activacions definides	Àmbit global
`\NOmostrarEtSint(0)`	Desactiva l'ús de les activacions definides	Àmbit global

2.3.3 Construcció de la categoria sintàctica

Les etiquetes sintàctiques es passen com a paràmetre de la comanda `\nucli` (taula 2). L'usuari indica la connectiva sintàctica (`sobre` o `sota`), la categoria sintàctica de l'argument, i la categoria sintàctica del resultat; la categoria del functor es calcula internament, i es parentitza segons el valor del paràmetre corresponent de la mateixa comanda `\nucli` (taula 9).

Taula 9 Ús del paràmetre de parentització en la construcció de l'etiqueta sintàctica de la pota inferior

Valor	Acció	Exemple	Exemple
00	No es parentitza	A/B	B\A
01	Es parentitza el resultat	(A)/B	B\(A)
10	Es parentitza l'argument	A/(B)	(B)\A
11	Es parentitzen el resultat i l'argument	(A)/(B)	(B)\(A)

En el cas que el nucli correspongui a una categoria sintàctica simple, és a dir, en el cas que tinguem una categoria sintàctica atòmica amb tipus elevat, voldrem que la pota inferior del nucli tingui per etiqueta la categoria sintàctica corrresponent, i que les potes superiors tinguin només els tipus semàntics. En aquest cas l'etiqueta calculada internament per la pota de sota deixa de tenir sentit. Per obtenir el comportament desitjat n'hi ha prou amb donar el valor buit a una de les dues categories (argument o resultat); llavors l'etiqueta construïda per a la pota inferior coincideix amb el valor de la categoria sintàctica no buida que hom ha passat com a argument de `\nucli`. (Nota: llavors, la connectiva que cal passar com a argument de `\nucli` només serveix per definir la permutació o no del nus bàsic usat).

NOTA: Cal tenir present que els arguments de `\calEtSint` fan referència als commutador de l'etiqueta inferior, esquerra i dreta, respectivament, independentment de si en aquesta posició hi ha un argument o un resultat. En canvi, en la construcció del tipus semàntic els valors indicats amb `\etTipSem` indiquen el tipus de l'argument i del resultat, independentment de la situació d'aquests a la pota dreta o esquerra del nus. Anàlogament, els arguments de `\calTipSem` fan referència al commutador de l'etiqueta de l'argument, i al commutador de l'etiqueta del resultat, respectivament.

2.4 Tipus semàntic

2.4.1 Contenidors del tipus semàntic

Cada nucli admet dues etiquetes pel tipus semàntic: a la pota esquerra, i a la pota dreta; els contenidors d'aquestes etiquetes són, respectivament, `\etEsqTip` i `\etDretaTip`.

2.4.2 Activació del tipus semàntic

Internament l'aparició de cada etiqueta està controlada pels commutadors de la taula 10, que s'usen només en el cas que el meta–commutador de la taula 11 estigui activat. L'activació externa de l'aparició de les etiquetes es fa amb la comanda `\calTipSem` (taula 12), que rep per paràmetre el valor a assignar als dos commutadors, amb l'ordre esquerra i dreta. L'activació (i desactivació) externa de la demanda d'usar (o no) els valors definits sobre els commutadors de les etiquetes del tipus semàntic es fa amb les comandes de la taula 13.

Taula 10	Commutadors (interns) pel control de la visualització dels tipus semàntics
`\calEtEsqTip`	Etiqueta de la pota esquerra dels nusos
`\calEtDretaTip`	Etiqueta de la pota dreta dels nusos

Valor dels commutadors:

en blanc	no apareix l'etiqueta
qualsevol valor	apareix l'etiqueta

Taula 11	(Meta-)Commutador (intern) pel control de la visualització del tipus semàntic
`\calMostrarTipSem`	Indica si les etiquetes dels tipus semàntics s'han de visualitzar en funció de les seves activacions o no

Valor dels commutadors:

en blanc	Tota etiqueta es considera desactivada
qualsevol valor	S'usen les activacions definides

Taula 12	Activació externa de l'aparició dels tipus semàntics	
`\calTipSem(2)`	Assigna els paràmetres als commutadors interns, en l'ordre de la taula 10	Àmbit local

Taula 13	Activació i desactivació externa de l'ús de les activacions sobre les etiquetes del tipus semàntic	
`\mostrarTipSem(0)`	Activa l'ús de les activacions definides	Àmbit global
`\NOmostrarTipSem(0)`	Desactiva l'ús de les activacions definides	Àmbit global

2.4.3 Construcció del tipus semàntic

Els tipus semàntics s'inicialitzen amb la comanda `\etTipSem`, on els paràmetres indiquen el tipus semàntic de l'argument i el del resultat, respectivament. Aquesta inicialització és local: cada cop que s'usen el valors indicats, s'esborren els contenidors corresponent.

NOTA: Cal tenir present que en la construcció del tipus semàntic els valors indicats amb `\etTipSem` indiquen el tipus de l'argument i del resultat, independentment de la situació d'aquests a la pota dreta o esquerra del nus. Anàlogament, els arguments de `\calTipSem` fan referència al commutador de l'etiqueta de l'argument i al commutador de l'etiqueta del resultat, respectivament. En canvi, els arguments de `\calEtSint` fan referència als commutador de l'etiqueta inferior, esquerra i dreta , respectivament, independentment de si en aquesta posició hi ha un argument o un resultat.

2.5 Etiqueta semàntica

2.5.1 Contenidor de l'etiqueta semàntica

L'etiqueta semàntica és sempre una constant passada com a paràmetre de la comanda `\nucli` (taula 2), i per tant no necessita de cap contenidor. Ara bé, si el valor del paràmetre es passa mitjançant el contenidor `\etSem` llavors es pot controlar la seva aparició. (De fet, aquest, juntament amb `\varSem`, són els únics contenidors que l'usuari ha de manipular directament).

2.5.2 Activació de l'etiqueta semàntica

Internament l'esborrat o no del contenidor `\etSem` just abans de dibuixar un nucli, està controlat pel commutador de la taula 14. L'activació externa d'aquest esborrat es fa amb la comanda `\NOmostrarEtSem` (sense paràmetres), i la seva desactivació amb `\mostrarEtSem` (taula 15).

Taula 14	Commutador (intern) pel control de la visualització de l'etiqueta semàntica
`\calMostrarEtSem`	
Valor del commutador:	
en blanc	Abans de visualitzar l'etiqueta semàntica s'esborra `\etSem`
qualsevol valor	No s'esborra el valor de `\etSem`

Taula 15	Activació i desactivació externa de l'aparició de les etiquetes semàntiques passades a través de `\etSem`	
`\mostrarEtSem(0)`	En el moment de visualitzar l'etiqueta semàntica no s'esborra el contingut de `\etSem`	Àmbit global
`\NOmostrarEtSem(0)`	Abans de visualitzar l'etiqueta semàntica s'esborra el contingut de `\etSem`	Àmbit global

2.6 Variables de l'abstracció

2.6.1 Contenidors de les variables de l'abstracció

En la lectura del λ–terme associat a una xarxa, els inputs dels nuclis $\wp$ s'han de llegir com a variables lligades a l'abstracció representada pel $\wp$ en qüestió. El nom de les variables lligades apareix a l'inici de la pota input ($\bullet$) dels nuclis $\wp$. El nom de la variable que cal visualitzar es manté en el contenidor `\varSem`.

2.6.2 Activació de les variables de l'abstracció

Internament l'aparició de les variables de l'abstracció està controlada pel commutador de la taula 16. L'activació externa de l'aparició de les variables de l'abstracció es fa amb la comanda `\mostrarVar` (sense paràmetres), i la desactivació amb `\NOmostrarVar` (taula 17).

Taula 16	Commutador (intern) pel control de la visualització de les variables lligades dels λ–termes	
`\calMostrarVar`		
	Valor del commutador:	
	en blanc	no apareix el nom de la variable
	qualsevol valor	apareix el nom de la variable

Taula 17	Activació i desactivació externa de l'aparició dels noms de les variables lligades dels λ–termes	
`\mostrarVar(0)`	Activa la visualització del nom de les variables lligades.	Àmbit global
`\NOmostrarVar(0)`	Desactiva la visualització del nom de les variables lligades.	Àmbit global

2.6.3 Construcció de les variables de l'abstracció

El valor del nom de la variable que es vol visualitzar s'ha de situar dins `\varSem`. (De fet, aquest, juntament amb `\etSem`, són els únics contenidors que l'usuari ha de manipular).

Posicionament de l'etiqueta. Les variables de l'abstracció no tenen una posició pròpia definida dins dels nusos; per això es visualitzen com a subíndex de l'etiqueta corresponent a la categoria sintàctica. El valor del desplaçament vertical de l'etiqueta de la variable de l'abstracció en relació a la de la categoria sintàctica es manté en el paràmetre d'entorn `\posVarSem` (és un `length`).

2.7 Ordre prosòdic

2.7.1 Contenidors de l'ordre proòdic

La posició dels arguments sobre la recta prosòdica s'indica amb una sageta des de la pota del nus que correspon a l'argument, a la pota que correspon al resultat; la taula 18 en dóna el significat. El contenidor del valor a visualitzar és `\ordreProsodic`.

Taula 18	Significat de la indicació de l'ordre prosòdic
$\rightarrow$	L'argument és per la dreta
$\leftarrow$	L'argument és per l'esquerra

2.7.2 Activació de l'ordre proòdic

Internament l'aparició de les variables de l'abstracció està controlada pel commutador de la taula 19. L'activació externa de l'aparició de l'ordre prosòdic es fa amb la comanda `\mostrarOrdre` (sense paràmetres), i la desactivació amb `\NOmostrarOrdre` (taula 20).

Taula 19	Commutador (intern) pel control de la visualització de l'ordre prosòdic
`\calMostrarOrdre`	
	<u>Valor del commutador:</u> en blanc — no apareix l'ordre qualsevol valor — apareix l'ordre

Taula 20	Activació i desactivació externa de l'aparició de l'ordre prosòdic	
`\mostrarOrdre(0)`	Activa la visualització de l'ordre prosòdic	Àmbit global
`\NOmostrarOrdre(0)`	Desactiva la visualització de l'ordre prosòdic	Àmbit global

2.7.3 Construcció de l'ordre proòdic

Es construeix automàticament en funció del nus, la connectiva i la permutació.

2.8 Lligam

2.8.1 Contenidors del lligam

Alguns nusos apareixen per mor del tipus semàntic, però no corresponen a cap comportament sintàctic; és el cas de les categories sintàctiques que no corresponen a tipus semàntics bàsics. En aquests casos les potes superiors del nus en qüestió van unides amb un lligam ($\cdots$).

El lligam és un valor constant contingut a `\ctLligam`; el valor que realment es visualitza (la constant o res) es manté a `\lligam`.

2.8.2 Activació del lligam

L'activació és local al proper nucli usat, i per tant no ens cal cap contenidor. L'activació externa de l'aparició del lligam es fa amb la comanda `\mostrarLligam` (sense paràmetres); la desactivació és automàtica (taula 21).

Taula 21	Activació externa de l'aparició del lligam (la desactivació és automàtica)	
`\mostrarLligam(0)`	Activa la visualització del lligam	Àmbit local

2.8.3 Construcció del lligam

És un valor constant; per tant només s'ha d'activar o desactivar.

2.9 Numeració dels nodes d'una xarxa

2.9.1 Contenidors de la numeració

La numeració dels nodes és automàtica. Els nodes d'un nucli es numeren en l'ordre sota, esquerra, dreta; els nuclis d'una xarxa es numeren en l'ordre d'inclusió dels nuclis dins la xarxa. L'únic contenidor és el `counter` anomenat `comptadorXarxa`, que es reinicialitza cada cop que s'obre un entorn `xarxa`.

2.9.2 Activació de la numeració

Internament la numeració de cada pota d'un nus està controlada pels commutadors de la taula 22; els tres primers ténen un àmbit local; els tres darrers el ténen global. Tant uns com els altres només s'usen en el cas que el meta–commutador de la taula 23 estigui activat. L'activació externa de la numeració es fa amb les comandes `\numeracio` i `\numeracioPerOmissio`, que reben per paràmetre el valor dels commutaddors interns, en l'ordre sota, esquerra i dreta; la primera comanda afecta els commutadors locals, la segona els globals (taula 24). L'activació (i desactivació) de la demanda d'usar (o no) els commutadors de la numeració automàtica es fa amb les comandes de la taula 25.

Taula 22 Commutadors (interns) pel control de la numeració dels nodes

`\comptSota`	Numeració del node de la pota inferior	Local
`\comptEsq`	Numeració del node de la pota esquerra	Local
`\comptDreta`	Numeració del node de la pota dreta	Local
`\comptSotaPerOmissio`	Numeració del node de la pota inferior	Global
`\comptEsqPerOmissio`	Numeració del node de la pota esquerra	Global
`\comptDretaPerOmissio`	Numeració del node de la pota dreta	Global

<u>Valor del commutador:</u>

en blanc	numeracio automàtica
qualsevol valor	apareix en lloc de la numeració automàtica

Taula 23 (Meta-)Commutador (intern) pel control de la visualització de la numeració

`\calMostrarNumeracio`	Indica si la numeració dels vèrtexs s'ha de visualitzar en funció de les seves activacions o no

<u>Valor dels commutadors:</u>

en blanc	No apareix cap vèrtex numerat
qualsevol valor	S'usen les activacions definides

Taula 24 Activació de la numeració dels nodes

`\numeracio(3)`	Assigna els paràmetres als commutadors interns d'àmbit local, segons l'ordre de la taula 22	Local
`\numeracioPerOmissio(3)`	Assigna els paràmetres als commutadors interns d'àmbit global, segons l'ordre de la taula 22	Global

Taula 25 Activació i desactivació externa de l'ús de les activacions sobre la numeració dels vèrtexs

`\mostrarNumeracio(0)`	Activa l'ús de les activacions definides	Àmbit global
`\NOmostrarNumeracio(0)`	Desactiva l'ús de les activacions definides	Àmbit global

2.9.3 Construcció de la numeració

Per a determinar la visualització de la numeració s'usen els commutadors locals. En cas que el valor del commutador estigui en blanc, es genera una etiqueta usant el comptador que fa de contenidor; altrament s'usa el valor del commutador com a etiqueta a visualitzar. Així si hom vol suprimir la numeració cal inicialitzar el commutador a un valor no visible, diferent del blanc o buit (el més pràctic és donar-li el valor {} que s'ha de donar com {{}}).

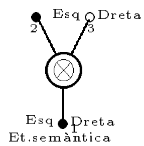
Un cop usat el commutador local per generar l'etiqueta de numeració, aquest es reinicialitza al valor del respectiu commutador global.

2.9.4 Ús de la numeració

Cal tenir present que els nodes es numeren sobre els nuclis, no sobre la xarxa. Això fa que un mateix vèrtex compartit per dos nuclis rebi dos valor numèrics (un situat a l'esquerra, i l'altre a la dreta). És responsabilitat de l'usuari desactivar localment les numeracions per tal que els nodes susceptibles de rebre etiqueta de numeració només en rebin una.

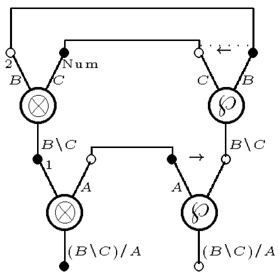
Per altra banda, la decissió de disseny de situar la numeració lateralment als vèrtexs, però amb un petit desplaçament vertical, permet que amb l'ús de `\rodoneta` hom pugui afegir altres etiquetes laterals sense que es generi solapament. Per exemple:

```
\mostrarNumeracio
\numeracioPerOmissio{}{}{}
\begin{xarxa}{0}{0}
   \nucli{0}{0}{\nustimes}{sobre}%
             {N}{S}{}{Et.sem\'antica}
   \rodoneta{}{0}{0}{Esq}{Dreta}
   \rodoneta{}{1}{1}{Esq}{Dreta}
\end{xarxa}
```



2.10 Exemple

Mostrem a continuació un exemple de construcció d'una xarxa usant les possibilitats del **c$_{\mathbf{ov}}$e** . La primera comanda deineix els commutadors de la manera indicada a la taula 32, pàgina 25.




```
\xarxaEstandard


\numeracioPerOmissio{{}}{{}}{{}}
\renewcommand{\distanciaEntrePortes}{2.5}
\begin{xarxa}{5}{3}
   \posicioAbs{5}{0}%
   %
   \calEtSint{x}{}{x}
   \nucli{0}{0}{\nustimes}{sobre}{A}{B\sota C}{01}{}%
   %
   \calEtSint{x}{x}{x}%
   \numeracio{}{}{Num}%
   \nucli{0}{1}{\nustimes}{sota}{B}{C}{}{}%
   %
   \renewcommand{\varSem}{x}%
   \calEtSint{x}{x}{}%
   \mostrarOrdre%
   \nucli{\distanciaEntrePortes}{0}{\nuspar}{sobre}{A}%
           {B\sota C}{01}{}%
   %
   {\suma{1}{\distanciaEntrePortes}%
    \edef\distanciaEntrePortes{\resultat}%
    \calEtSint{x}{x}{x}%
    \renewcommand{\varSem}{y}%
    \mostrarLligam%
    \nucli{\distanciaEntrePortes}{1}{\nuspar}{sota}{B}{C}{}{}%
    %
    \axioma{0}{1}{e}{\distanciaEntrePortes}{1}{d}{1}{}%
    \axioma{0}{1}{d}{\distanciaEntrePortes}{1}{e}{0}{}%
   }%
   %
  \axioma{0}{0}{d}{\distanciaEntrePortes}{0}{e}{0}{}%
\end{xarxa}
```

3 Composició de xarxes

3.1 El concepte de bloc **17**
3.2 Construcció de blocs **17**
3.2.1 L'esquelet d'un bloc 17
3.2.2 L'entorn `bloc` 18
3.3 Construcció de xarxes compostes **18**
3.3.1 Xarxes de blocs i xarxes de xarxes 20
3.3.2 Posicionament dins d'una xarxa composta 20
3.4 Exemples **21**

3.1 El concepte de bloc

Per tal de poder construir xarxxes incrementalment a partir de xarxes prèviament definides, introduïm el concepte de bloc. Un *bloc* és una comanda que construeix una xarxa, i que té les següents característiques:

1. Defineix les mesures envolvents de la xarxa que construeix
2. Pot aparèixer aïllat o formant part d'una altra xarxa
3. En el cas d'aparèixer dins d'una altra xarxa modifica el punt de referència dins de la xarxa contenidora només si l'autoposicionament està activat (veure la secció 3.3); en cas contrari deixa el punt de referència inalterat.

És responsabilitat de l'usuari programar les comandes de construcció de blocs.

La següent secció explica com programar les comandes de construcció de blocs; la secció 3.3 descriu com usar els blocs definits per a la construcció de xarxes compostes.

3.2 Construcció de blocs

3.2.1 L'esquelet d'un bloc

Les mesures envolvents d'un bloc s'han de guardar a `\xh` i `\xv`, usant la comanda de definició `\xdef`. Per tal que el bloc pugui aparèixer tant aïllat com dins d'una xarxa, i que en aquest cas el punt de referència de la xarxa contenidora es gestioni convenientment, cal definir-lo amb l'entorn `bloc`. La taula 27 mostra l'estructura general d'un bloc.

Taula 26 Mesures que ha de definir tota comanda de definició d'un bloc

`\xh`	Mesura horitzontal del rectangle envolvent
`\xv`	Mesura horitzontal del rectangle envolvent

Totes les mesures són amb coordenades de xarxa

Taula 27 Estructura de les comandes de definició de blocs

Inicialització de les mesures del bloc (taula 26)
`\begin{bloc}`
    Construcció de la sub-xarxa o bloc
`\end{bloc}`

3.2.2 L'entorn `bloc`

Expliquem aquí el funcionament intern de l'entorn `bloc`, amb el benentès que aquest coneixement és del tot innecessari al programador de comandes constructores de blocs.

L'entorn `bloc` té les següents funcionalitats bàsiques:

1. Mantenir el punt de referència, amb independència dels canvis que es facin dins l'entorn, per facilitar la constrrucció de la xarxa. Per fer-ho utilitza la pila `posicions`: en el moment d'entrar a l'entorn s'empila la posició de referència, i es desempila al sortir-ne. Les funcions encarregades de la gestió de la pila són `\guardaPosicio` i `\recuperaPosicio`.
2. Construir una xarxa contenidora només quan la composició de xarxes (taula 28) estigui desactivada. En aquest cas, les mesures de la xarxa són les mesures envolvents del bloc (`\xh` i `\xv`)
3. Situar el bloc dins la xarxa contenidora actual, i modificar el punt de referència d'aquesta quan calgui. Si el commutador `\autoPosicionament` està desactivat (taula 28), cada bloc es situa en la posició de referència activa dins la xarxa contenidora; aquest posicionament no modifica la posició de referència. Si el commutador `\autoPosicionament` està activat, el bloc es situa en la posició de referència activa dins la xarxa conenidora; i es canvia automàticament aquesta referència per tal que el següent bloc no es superposi a l'actual. Internament el que fem és que cada bloc, un cop dibuixat, canvia el punt de referència amb un desplaçament horitzontal igual a la seva magnitut més el valor de `\distMin` (expressat amb unitats de xarxa).

Internament aquestes tasques les realitzen `\fesComposicio` que s'executa a l'obrir el bloc; i `\preparaNovaComposicio` que s'executa al tancar el bloc. Cal tenir present que per mor de l'ús de la pila `posicions` per mantenir el punt de referència, els blocs defineixen un entorn tancat, de tal manera que les definicions que es realitzin dins d'un bloc no en traspassarán el seu àmbit, a no ser que fem la definició global. (Per un estudi acurat del funcionament de les piles, vegueu l'apèndix C.3).

3.3 Construcció de xarxes compostes

La taula 28 defineix els commutadors interns que s'usen en la composició de les xarxes; la taula 29 mostra les comandes externes d'activació i desactivació d'aquests commutadors; la taula 30 mostra els paràmetres d'entorn i el seu valor per omissió.

<table>
<tr><th colspan="3">Taula 28 Commutadors (interns) usats en la composició de blocs</th></tr>
<tr><th>Commutador</th><th>Descripció</th><th>Per omissió</th></tr>
<tr><td><code>\composicioXarxes</code></td><td>Indica si els blocs predefinits s'usen en mode de composició o no</td><td><code>si</code></td></tr>
<tr><td colspan="3">El valor <code>no</code> fa que cada bloc predefinit es construeixi en un entorn <code>xarxa</code> propi; qualsevol altre valor permet col.locar diferents blocs predefinits dins d'un mateix entorn <code>xarxa</code> extern al bloc. En el cas d'haver de construir una xarxa, s'usen <code>\xh</code> i <code>\xv</code> com a mesures d'aquesta.</td></tr>
<tr><td><code>\autoPosicionament</code></td><td>Indica si els blocs predefinits es posicionen automàticament dins la xarxa</td><td><code>si</code></td></tr>
<tr><td colspan="3">Si el valor de <code>\autoPosicionament</code> és <code>si</code> es “reserva” l'espai horitzontal d'un bloc (la seva mesura <code>\xh</code>, més el desplaçament mínim <code>\distMin</code> per evitar solapaments) a base de modificar el punt de referència actual amb un desplaçament horitzontal relatiu igual a la quantitat d'espai a “reservar”.</td></tr>
</table>

Taula 29 Activació i desactivació externa dels commutadors per a la composició de blocs

`\composar`	Activa la composició de blocs: els blocs no obren per si mateixos cap entorn xarxa.	Àmbit global
`\NOcomposar`	Desactiva la composició de blocs: cada bloc obre un entorn xarxa amb les mesures del bloc.	Àmbit global
`\autoPosicionar`	Activa l'autoposicionament: en el posicionament dels blocs cada bloc canvia el punt de referència actual fent un desplaçament horitzontal relatiu igual a la seva magnitut horitzontal més el valor de `\distMin`	Àmbit global
`\NOautoPosicionar`	Desactiva l'autoposicionament	Àmbit global

Taula 30 Paràmetres d'entorn per a la composició de blocs i xarxes

`\distMin`	Desplaçament horitzontal, en coordenades de xarxa, que s'afegeix a la magnitut horitzontal de cada bloc per redefinir les coordenades de referència quan l'autoposicionament està actiu (taula 29)	Per omissió: 0.5

3.3.1 Xarxes de blocs i xarxes de xarxes

En principi (a part de la pila del LaTeX) res no impedeix de construir una xarxa dins d'una altra. Malgrat tot hi ha un parell de problemes que cal tenir en compte:

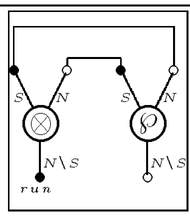
1. Les coordenades de cada xarxa són independents. Això fa que sigui difícil, per exemple, establir un lligam de tall entre dues xarxes
2. Cada entorn `xarxa` genera un petit espaiat. Com a conseqüència, la construcció d'una xarxa complexa a partir de xarxes més simples té la dificultat de fer coincidir exactament els elements

La construcció de xarxes a partir dels blocs resol ambdós problemes: hom usa un sol sistema de coordenades, i l'únic entorn `xarxa` present és el corresponent a la xarxa contenidora.

Un problema apart és el de les mesures lògiques d'un bloc. Si usem un bloc com a xarxa (hem usat la desactivació `\NOcomposar`) llavors la caixa LaTeX resultant té la mesura adient, i es pot, per exemple, emmarcar. Si usem el bloc dins d'una xarxa contenidora (hem usat l'activació `\composar`) llavors la mesura de la caixa és zero. A efectes pràctics això significa que si volem emmarcar un bloc llavors o bé aquest s'usa tal qual amb la composició desactivada; o bé s'ha d'emmarcar la xarxa envolvent (figura 1).

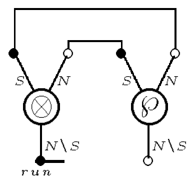
Figura 1 Exemple: un bloc com una xarxa, una xarxa d'un bloc

```
\NOcomposar
\frame{\xrun}
```



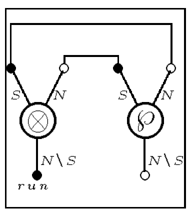
La xarxa es construeix internament, i té les mesures adients

```
\composar
\begin{xarxa}{2}{0.5}
   \frame{\xrun}
\end{xarxa}
```



El bloc no genera cap xarxa; la caixa envolvent no és la correcta

```
\composar
\frame{
   \begin{xarxa}{2}{0.5}
\xrun
\end{xarxa}
}
```



Creem manualment una xarxa contenidora amb les mesures adients, i emmarquem aquesta

3.3.2 Posicionament dins d'una xarxa composta

Les comandes **ELX** `\posicioAbs` i `\posicioRel` permeten situar on més convingui els blocs dins d'una xarxa; les coordenades indicades absolutes fan referència a les coordenades globals de la xarxa; les relatives, al punt de referència actual: cal tenir present peró que si s'ha usat l'activació `\autoPosicionar` cada bloc modifica el punt de referència en funció de la seva magnitut horitzontal i el valor de `\distMin`.

La comanda `\x` (taula 31) permet situar un bloc en la coordenada especificada sense que es modifiqui el punt de referència; equival a usar el bloc amb la desactivació temporal de l'autoposicionament, de tal manera que els següents blocs es posicionen (en el cas de tenir l'autoposicionament activat) en funció dels blocs anteriors que no s'han emprat amb `\x`. Si el paràmetre opcional conté qualsevol valor diferent de {} llavors el posicionament és relatiu a les coordenades actuals; altrament el posicionament és absolut (aquesta és l'opció per omissió).

Taula 31 Comanda de posicionament dels blocs

`\x`([rel] x,y)	Posiciona un bloc en la posició `(x,y)` sense canviar el punt de referència.	Per omissió: posicionament absolut

Valor del paràmetre opcional:

en blanc	Les coordenades són absolutes
qualsevol valor	Les coordenades són relatives

3.4 Exemples

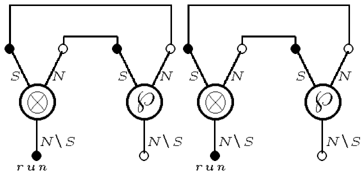
La figura 2 mostra un exemple de construcció d'un bloc. La figura 3 mostra com usar el bloc anterior per construir una xarxa. Les figures 4 i 5 mostren les comandes de posicionament dels blocs dins d'una xarxa composta: les úniques diferències entre elles és la presència o no de l'autoposicionament automàtic.

Figura 2 Exemple de construcció d'un bloc

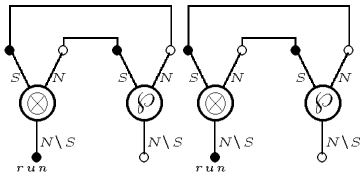
```
\newcommand{\xxdef}[7]
{\setcounter{comptadorXarxa}{0}%
 \renewcommand{\A}{#3}%
 \renewcommand{\B}{#4}%
 \renewcommand{\C}{#5}%
 \renewcommand{\K}{#6}%
 \renewcommand{\etSem}{#7}%
 \ifthenelse{\equal{#2}{}}%
     {\renewcommand{\distanciaEntrePortes}{#1}}%
     {\renewcommand{\distanciaEntrePortes}{#2}}%
 %
}

\newcommand{\xrun}[1][{}]
{\xxdef{2}{#1}{N}{S}{x}{run}{$\abstraccio{x}\aplicacio{\K}{x}$}%
 \renewcommand{\xh}{\distanciaEntrePortes}%
 \renewcommand{\xv}{0.5}%
 %
 \begin{bloc}
       \calEtSint{x}{x}{x}%
       \renewcommand{\varSem}{x}%
       \numeracio{}{}{}%
       \nucli{0}{0}{\nustimes}{sota}{\A}{\B}{}{$\K$}%
       \numeracio{}{}{}%
       \nucli{\distanciaEntrePortes}{0}{\nuspar}{sota}{\A}{\B}{}{\etSem}%
       \axioma{0}{0}{e}{\distanciaEntrePortes}{0}{d}{1}{}%
       \axioma{0}{0}{d}{\distanciaEntrePortes}{0}{e}{0}{}%
  \end{bloc}
}
```

Figura 3 Exemple de composició de xarxes

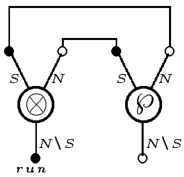
```
\NOcomposar%
\NOautoPosicionar%
\xrun
\xrun
```



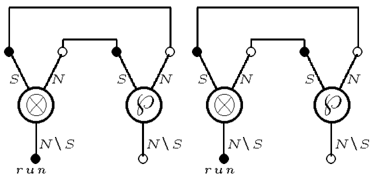
```
\NOcomposar%
\autoPosicionar%
\xrun
\xrun
```



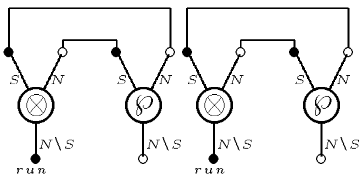
```
\composar%
\NOautoPosicionar%
\begin{xarxa}{6}{1}
   \xrun
   \xrun
\end{xarxa}
```



```
\composar%
\autoPosicionar%
\begin{xarxa}{6}{1}
   \xrun
   \xrun
\end{xarxa}
```



```
\NOcomposar%
\autoposicionar%
\begin{xarxa}{6}{1}
  \xrun
  \xrun
\end{xarxa}
```



Els dos primers exemples mostren com no cal usar explícitament l'entorn **xarxa** si estem en no composició; i com en aquest cas l'autoposicionat no té cap mena d'efecte Els darrers tres exemples mostren el resultat d'usar explícitament l'entorn **xarxa**, tant amb composició com sense. És important d'observar com els espaiats són lleugerament diferents en el cas de composar o no.

Figura 4 Ús de les comandes de posicionament en les xarxes compostes: amb autoposicionament automàtic

```
\composar
\autoPosicionar % <--
\begin{xarxa}{5}{5}
   \xrun%
   \x{\xrun}{0}{4}%
   \posicioRel{0}{2}%
   \xrun%
   \xrun%
   \posicioAbs{0}{0}%
   \xrun%
   \xrun%
   \x[x]{\xrun}{0}{3}%
   \xrun%
 \end{xarxa}
```

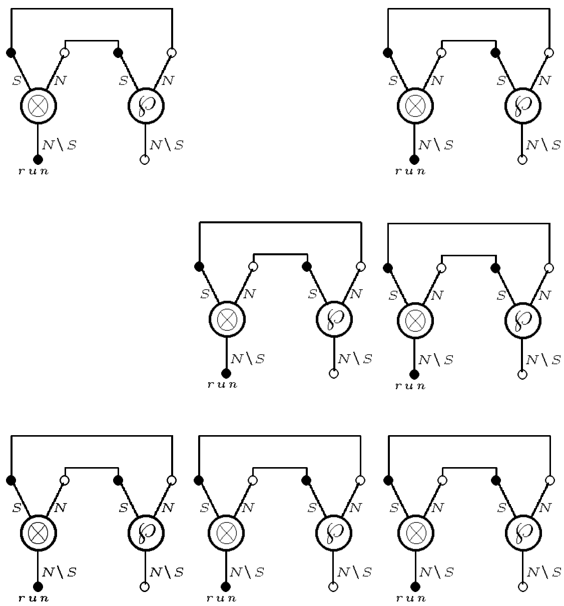
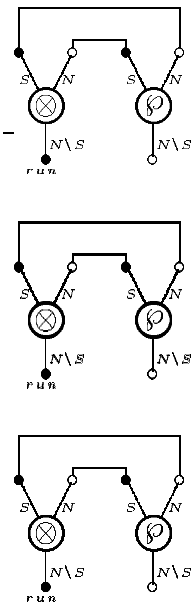


Figura 5 Ús de les comandes de posicionament en les xarxes compostes: sense autoposicionament automàtic

```
\composar
\NOautoPosicionar     % <--
\begin{xarxa}{5}{5}
   \xrun%
   \x{\xrun}{0}{4}%
   \posicioRel{0}{2}%
   \xrun%
   \xrun%
   \posicioAbs{0}{0}%
   \xrun%
   \xrun%
   \x[x]{\xrun}{0}{3}%
   \xrun%
 \end{xarxa}
```



4 Comandes d'alt nivell

L'entorn `xarxaDual` genera un entorn on les definicions dels nusos bàsics s'han canviat entre sí, de tal manera que a l'incloure un bloc dins d'aquest entorn s'obté el bloc dual.

La comanda `\xarxaEstandard` estableix els valors dels commutadors i dels contenidors de la manera que mostra la taula 32.

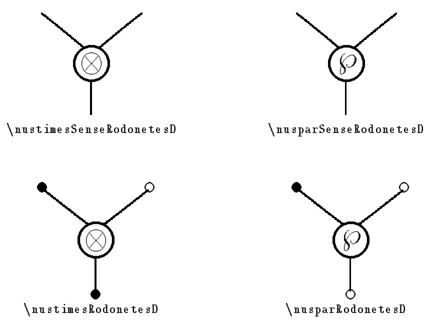
Taula 32 Definició d'una xarxa estàndard usant `\xarxaEstandard`

```
%--- Els nusos han de contenir etiquetes
  \ambEtiquetes%
%-- commutadors
  \permutacioAutomatica
  \NOmostrarOrdre%
  \NOmostrarVar%
  \mostrarEtSem%
  \numeracioPerOmissio{{}}{{}}{{}}%  No numerem
%-- meta-commutadors
  \mostrarEtSint%
  \mostrarTipSem%
  \mostrarNumeracio%
%-- Contenidors de manipulacio externa
  \renewcommand{\varSem}{x}%
  \renewcommand{\etSem}{}%
```

5 Nous nusos

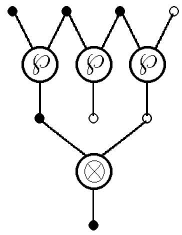
S'han creat nusos que tenen una amplada exactament el *doble* que els nusos normals. Per fer-ho s'ha tingut que variar el codi de `\defnus` (veure l'apartat B.1, pàgina 28). Per no saturar la memòria s'han definit només nusos dobles encerclats, amb i sense rodonetes. Si l'usuari necessita d'altres modalitats pot pendre la definició d'aquest nusos com a patró. La taula 33 mostra els nous nusos definits; la figura 34 mostra un exemple d'ús.

Taula 33 Nous nusos definits



Taula 34 Exemple d'ús dels nous nusos definits

```
\nusosAmbRodonetes
\senseEtiquetes
\begin{xarxa}{4}{1}
\nus{\nuspar}{0}{1}
\nus{\nuspar}{1}{1}
\nus{\nuspar}{2}{1}
\nus{\nustimesD}{0.5}{0}
\end{xarxa}
```



A El nom sí el fa la cosa

A.1 Sobre l'evolució de les espècies

Inicialment havíem pensat en una comanda de composició o adjunció de xarxes: es tractava d'enviar-li per paràmetre un parell de xarxes i les indicacions de quin punt era el punt compartit, i que ens retornés la xarxa resultant. D'alguna manera la idea era que els nusos definits amb **E$_{\mathbf{L}}$X** constituïen els casos bàsics de la definició recursiva de les xarxes, i que a partir d'ells i d'una funció de composició es podien obtenir noves xarxes, que a tots els efectes actuarien com si de nous nusos bàsics es tractés.

Si bé la idea semblava simple i elegant, la desesperació ens arribà en forma d'impertinències contínues del LaTeX:

```
! TeX capacity exceeded, sorry [parameter stack size=60].
```

Vam redissenyar múltiples vegades totes les funcions, les vam implementar de diversess maneres, i malgrat tot, de forma inesperada i quan hom menys s'ho esperava, el LaTeX decidia tornar a fer acte de protesta. Malgrat l'intent d'endinsar-nos en els foscos avencs de la gestió de memòria del LaTeX no vam aconseguir res més que autèntiques dosis de maldecap i desesperació.

Arribats a aquest punt vam decidir canviar d'estratègia: ens oblidàvem de veure cada xarxa com un tot, i passàvem a considerar-la com un conjunt de nusos. Les conseqüències d'aquest plantejament foren:

1. La gestió de la visualització o no de les etiquetes, així com la generació d'aquestes deixava de ser un procés automàtic per esdevenir, en el millor dels casos, semi-automàtic. Alguns casos ens ho aclariran.
   - Numeració dels vèrtexs. Si la numeració és global a tota la xarxa (cas del primer enfoc) internament es pot evitar donar dues etiquetes sobre un mateix vèrtex. Si la numeració és independent per cada nus de la xarxa (segon enfoc) llavors aquest control s'ha de fer de forma externa, bloquejant la numeració dels vèrtexs ja numerats.

- Categoria sintàctica. Amb l'enfoc inicial, l'ús de la composició permetia detectar quan dues arestes han de tenir la mateixa etiqueta, i per tant internament es podia decidir quina d'ambdues arestes era sobre la que es visualitzava l'etiqueta. Amb el nou enfoc aquesta decissió s'ha de fer externament, bloquejant l'aparició de les etiquetes en aquelles arestes on la informació resultaria redundant.

La conseqüència immediata de tot plegat és la necessitat de comandes per activar o bloquejar la visualització de les etiquetes definides sobre els nusos. Aquesta és la gènesi del $\mathbf{c_{ov}e}$.

2. Ara tenim un sol entorn `xarxa` que engloba tots els nusos (o nuclis) que formen la xarxa. Les subxarxes no poden ser, per tant, xarxes. D'aquí la necessitat de l'entorn `bloc`. Aquesta és la gènesi del peix.
3. Sorprenentment, i després de rebre unes quantes impertinències més del nostre millor enemic, l'exclamació ha aparegut sense ni adonar-nos-en a la nostra boca: ja està, ja tenim el peix al cove!! La cosa ha fet el nom: la resta, un xic d'enginyeria inversa.

A.2 No tot són flors i violes

Ja tenim el peix al cove: $\mathbf{c_{ov}e}$ (peix)! Malgrat tot encara tenim certs problemes: o el peix està encara massa viu, o el cove és massa petit. Si bé $\mathbf{c_{ov}e}$ (peix) resol força bé els objectius marcats, en la construcció de blocs grans encara apareixen problemes de memòria. Aquests però no semblen seguir una pauta més o menys clara, i apareixen en el moment més impensat. Després de múltiples proves només podem aventurar alguns consells:

- La inserció d'una línia en blanc a voltes soluciona el problema. Tot fa indicar que el LaTeX té un buffer d'expansió de paràgrafs, de tal manera que si el paràgraf resultant és massa gran es produeix un sobreeiximent; en aquest cas, la partició del bloc en dos paràgrafs soluciona el problema. Malgrat tot aquesta solució no és universal: hem trobat casos on la memòria peta independentment de la magnitud del paràgraf.
- L'ús de blocs com a arguments de funcions no és molt recomenable. Aquesta sembla ser la semàntica de l'esgarrip del LaTeX: la pila dels paràmetres té sobreeximent, per excés de crides. Novament tampoc aquesta solució és universal: hem trobat casos de blocs gairebé idèntics on un es pot passar com a argument, i l'altre no. El més sorprenent del cas és que l'ús d'un bloc o un altre (malgrat la seva quasi identitat) ens fa passar d'un valor de pila de 47 a un valor de pila de 63. Incomprensible.

Un altre problema és la introducció dels nous nusos: ara el nombre de macros és excessivament gran i el LaTeX es pot quedar fàcilment sense memòriaa principal. L'ús de la versió 3.1415 del TeX, enlloc de la 3.14159, soluciona el problema (la nova versió redueix la memòria disponible a la meitat).

B Modificacions a ELX

En la construcció de coveix e hem intentat mantenir intacta la integritat d'**ELX**. Malgrat tot ens hem vist obligats a fer alguna petita modificació, i hem aprofitat per suprimir alguna petita brossa d'**ELX**, i per millorar sensiblement tot el sistema de posicionament.

B.1 Desbrossat i modificacions

N=0

- Per evitar que les comandes al ser usades dins d'un `picture` introdueixin espaiats indesitjables, s'ha afegit `%` al final de cada línia de codi. Aquest canvi s'ha fet extensiu també a tots els nivells d'**ELX** (N=1 i 2)
- Els valors numèrics explícits usats en les comandes de dibuix s'han substituït per les constants de la taula 35

Taula 35 Constants pels valors numèrica

`\cta`	0.29289322	$1-\sqrt{2}/2 = 1-$`\ctb`
`\ctb`	0.70710678	$\sqrt{2}/2$
`\ctc`	0.79289322	`\cta`+0.5
`\ctd`	1.4142136	$\sqrt{2}$

- A la comanda `\dibuixaAxiomaEt` s'ha afegit el codi de la taula 36 per tal que l'enllaçat entre premisses i axiomes o nusos sigui el correcte

Taula 36 Afegit a l codi de la comanda `\dibuixaAxiomaEt`

```
\ifthenelse{\equal{#4}{c}}
    {\addtolength{\ypa}{-\cta\midaActualRodoneta}}{}%
\ifthenelse{\equal{#7}{c}}%
    {\addtolength{\ypb}{-\cta\midaActualRodoneta}}%
```

- A la comanda `\defnus` s'ha canviar la primera aparició de `\addtolength{\ypa}` per `\settolength{\ypa}`. El resultat és el mateix, però abans usàvem la inicialització implícita del LaTeX, i ara usem inicialització explícita.
- S'ha redefinit de nou el dibuix de les potes superiors a laa comanda `\defnus`. Ara es pot definir amb paràmetres fictícis la relació entre la projecció horitzontal de les potes superiors dels nusos, i la mida de la pota inferior (abans la relació era 1/1) . La taula 37 mostra els nous paràmetres fictícis, i la seva funcionalitat. El paràmetre fictici `\ampladaRod` permet crear nodes per ser usats com a arguments de la comanda `\defnusRodonetes`, de tal manera que el nus amb rodonetes obtingut tingui l'amplada justa desitjada.

Taula 37 Paràmetres ficticis nous per a `\defnus`

`\ampladaRel`	Proporció entre la projecció horitzontal de cada pota superior i la mida de la pota inferior	`command`
`\ampladaRod`	Indica la magnitut horitzontal dels símbols que es situaran després a les potes. El resultat és que la proporció exacta demanada s'obté després de situar el símbol al nus generat.	`length`

La nova versió de `\defnus` requereix poder dibuixar línies amb qualsevol angle; per això usa la macro `\linia` , que al seu torn usa la comanda `\qbeziers`, enlloc de `\line`, per a dibuixar les línies (veure l'apartat C.2, pàgina 34).

La taula 38 mostra la part del codi de `\defnus` que ha variat.

Taula 38 (Fragment de) El nou codi de `\defnus`

```
     % Calcul de les mesures per les potes superiors
        % Extrem x
            \setlength{\lxtemp}{\inversatangAngle\ltemp}%
            \setlength{\lxtemp}{\ampladaRel\lxtemp}%
            \setlength{\xtemp}{\cosinusAngle\mh}%
            \addtolength{\lxtemp}{\ampladaRel\xtemp}%
            \addtolength{\lxtemp}{-\xtemp}%
            % Control de la rodoneta
                \setlength{\ampladaRod}{\ctb\ampladaRod}%
                \setlength{\xtemp}{\ampladaRel\ampladaRod}%
                \addtolength{\xtemp}{-\ampladaRod}%
                \addtolength{\lxtemp}{0.5\xtemp}%
             \treurePt[2]{\lxtemp}%
            \edef\cxtemp{\resultat}%
         % Extrem y
            \setlength{\lytemp}{\ltemp}%
             \treurePt[2]{\lytemp}%
             \edef\cytemp{\resultat}%
   % Pota dreta
      \ifthenelse{\equal{#7}{}}{}%
        {\setlength{\unitlength}{\ypa}%
         \put(0,1){%
           \setlength{\unitlength}{\cosinusAngle\mh}%
           \put(1,0){\linia{0}{0}{\cxtemp}{\cytemp}}}}%
  % Pota esquerra
      \ifthenelse{\equal{#6}{}}{}%
         {\setlength{\unitlength}{\ypa}%
          \put(0,1){%
            \setlength{\unitlength}{-\cosinusAngle\mh}%
            \put(1,0){\linia{0}{0}{-\cxtemp}{\cytemp}}}}%
```

N=1

- Les comandes `\posicionarObjecte` i `\rodoneta` ara ténen dues etiquetes laterals: esquerra i dreta. La taula 39 en mostra el codi.

Taula 39 El nou codi de `\posicionarObjecte` i `\rodoneta`

```
    \newcommand{\posicionarObjecte}[6][{}]
{\coordenadesEnPts{#3}{#4}{\dhtemp}{\dvtemp}%
 \gotoXY{\dhtemp}{\dvtemp}{\makebox(0,0){#2}}%
%
 % Etiqueta de sota
   \addtolength{\dvtemp}{-0.5\midaVertObjecte}%
   \addtolength{\dvtemp}{-\distEtiqEntrada}%
   \tiny%
   \gotoXY{\dhtemp}{\dvtemp}{\makebox(0,0)[t]{#1}}%
%
 % Etiquetes laterals
   \addtolength{\dvtemp}{\distEtiqEntrada}%
   \addtolength{\dvtemp}{\midaVertObjecte}%   Recuperem l'alcada
  {%
 % Etiqueta lateral esquerra
   \addtolength{\dhtemp}{-\distHorEtiqAuxEntrada}%
  \gotoXY{\dhtemp}{\dvtemp}{\makebox(0,0)[r]{#5}}%
 }%
 {%
 % Etiqueta lateral dreta
   \addtolength{\dhtemp}{\distHorEtiqAuxEntrada}%
   \addtolength{\dhtemp}{\midaHoritObjecte}%
   \gotoXY{\dhtemp}{\dvtemp}{\makebox(0,0)[l]{#6}}%
 }%
   \normalsize%
}

\newcommand{\rodoneta}[6][{}]
  {\setlength{\midaHoritObjecte}{\midaRodoneta}%
   \setlength{\midaVertObjecte}{\midaRodoneta}%
   \setlength{\unitlength}{-0.29289322\midaRodoneta}%
   \posicionarObjecte[#1]{\put(1,0){#2}}{#3}{#4}{#5}{#6}%
  }
```

- En la comanda `\posicioRel` cal fer globals les assignacions sobre les coordenades, pel correcte posicionament dels blocs. N'hi ha prou amb afegir `\global` a les darreres dues línies del codi.

N=2

- L'entorn `xarxa` posiciona a la posicio `(0,0)` i inicialitza a zero el `comptadorXarxa`
- Per tal de millorar l'eficiència en espai en les comandes de personalització de l'entorn s'han canviat els `\renewcommand` per `\edef`
- S'ha modificat convenientment `\nusosEncerclats` per tal que les comandes de redefinició de l'entorn que engloba contemplin els nous nusos definits

B.2 El nou sistema de posicionament

La possibilitat de poder gestionar una pila de posicions ha permès simplificar tot el mecanisme: s'han suprimit coordenades internes, i s'ha alleugerit la gestió de les coordenades. En concret:

N=0

- S'han suprimit les comandes `\resitua` i `\resituaB`
- Les úniques coordenades que es mantenen són `\coorX` i `\coorY`
- S'ha modificat `\coordenadesEnPtsRel` tal com mostra la taula 40

Taula 40 El nou codi de `\coordenadesEnPtsRel`

```
\newcommand{\coordenadesEnPtsRel}[4]
    {\coordenadesEnPtsAbs{#1}{#2}{\xxtemp}{\yytemp}%
     \setlength{#3}{\coorX}%
     \setlength{#4}{\coorY}%
     \addtolength{#3}{\xxtemp}%
     \addtolength{#4}{\yytemp}%
```

A part d'aquestes modificacions, el peix aporta les comandes `\guardaPosicio` i `\recuperaPosicio`. La comanda `\guardaPosicio` es preocupa d'empilar les coordenades `\coorX` i `\coorY`; la comanda `\recuperaPosicio` reinicialitza aquestes coordenades amb els valors de la pila `posicions` . Cal tenir present que hem suposat que ambdues comandes `\guardaPosicio` i `\recuperaPosicio` aniran sempre aparellades (de fet això és el que fa l'entorn `bloc`), i per tant no cal l'ús de la comanda `\buidaPila` (veure l'apèndix C.3). En cas que no sigui així és responsabilitat de l'usuari assegurar-se que la pila està buida abans de tancar l'entorn `xarxa`, i d'inicialitzar-la (amb `\iniPila`) abans d'obrir una nova `xarxa`.

C Macros externes

En la construcció de covepeix s'han usat les macros de gestió de piles; en la definició dels nusos d'amplada doble s'ha usat la macro `\linia`, que al su torn usa `\treurePt`.

C.1 Conversió de `lengths` a `commands`

En LaTeX si assignem una longitut (`length`) a un comptador (`counter`), es fa una conversió interna $1pt = 65536$. Amb aquesta informació podem convertir `lengths` a `commands` usant el codi de la taula 41. En el cas que hom hagi inicialitzat el `length` amb punts, el resultat de `\treurePtBasic` equival a haver suprimit les unitats.

Taula 41 Conversió de les `length` expressades en punts a `counter`, és a dir, supressió de les unitats `pt`

```
\newcommand{\unitatDeConversio}%
      {0.00001525878}  % 1/65536
\newcounter{resultat}{}
\newlength{\ltemp}

\newcommand{\treurePtBasic}[1]
  {\setlength{\ltemp}{#1}
   \setlength{\ltemp}{\unitatDeConversio\ltemp}
   \global\setcounter{resultat}{\ltemp}
  }
```

La taula 42 mostra el codi de la comanda `\treurePt`, anàloga a `\treurePtBasic` però amb la possibilitat d'indicar el nombre de decimals amb el que es vol el resultat, que ara es retorna com una cadena ASCII dins de `\resultat`.

Taula 42 Conversió dels `length` expressats en punts a `command` sense les unitats, amb el nombre de decimals indicats, i amb arrodoniment

```
\newcommand{\treurePt}[2][\precissioPerOmissio]
 {% n.mp pt  -->  nm pt --> nm
    \setcounter{ci}{#1}
    \setlength{\lb}{#2}
    \whiledo{\value{ci} > 0}
       {\setlength{\lb}{10\lb}
        \addtocounter{ci}{-1}}

      % Tractament de l'arrodoniment
        \ifthenelse{\equal{\calArrodonir}{no}}
            {}{\addtolength{\lb}{0.5pt}}

 % n'm --> n'.m -> n'   % No uso el resultat inicial perque
                        % l'arrodoniment pot haver-lo fetcanviar
   \setcounter{ci}{#1}
   \setlength{\la}{\lb}
   \whiledo{\value{ci} > 0}
      {\setlength{\la}{0.1\la}
       \addtocounter{ci}{-1}}
   \treurePtBasic{\la}
   \setcounter{cres}{\value{resultat}}

  % n' --> n'0...0 pt --> n'0..0   (#0 =|m|=prec)
    \setcounter{ci}{#1}
    \treurePtBasic{\la}
    \setlength{\la}{\value{resultat} pt}
    \whiledo{\value{ci} > 0}
      {\setlength{\la}{10\la}
       \addtocounter{ci}{-1}}
    \treurePtBasic{\la}
    \setcounter{ca}{\value{resultat}}

  % n pt, nm pt --> m pt --> m
   \setlength{\lc}{\lb}
   \addtolength{\lc}{-\la}
   \treurePtBasic{\lc}

  % composicio del resultat
    \ifthenelse{\equal{#1}{0}}
         {\edef\resultat{\arabic{cres}}}
         {\edef\resultat\arabic{cres}.\arabic{resultat}}}
 }
```

C.2 Dibuix de línies entre dos punts qualssevol

En l'entorn `picture` la comanda `\line` permet dibuixar línies; el problema és que l'angle de dibuix s'ha d'escollir dins d'un conjunt prefixat força limitat. Per unir dos punts qualssevol, llavors, cal usar la comanda `\qbeziers`, i dibuixar una corba de Béziers sense curvatura. Per tal de poder usar coordenades diferents, calculem el `\factor` amb la comanda `\treurePt`. La taula 43 mostra el codi.

Taula 43 Línia recta entre dos punts qualssevol

```
\newcounter{clxm}
\newcounter{clym}
\newlength{\migx}
\newlength{\migy}
\newcommand{\unitat}{pt}
\newcommand{\factor }{1}%

\newcommand{\linia}[5][1 pt]
 {% ----- Calculem el factor
     \treurePt{#1}%
     \edef\factor{\resultat}%
  % ----- Calculem el punt mig
     \setlength{\migx}{#1 \unitat}%
     \addtolength{\migx}{#3 \unitat}%
     \setlength{\migx}{0.5\migx}%
     %
     \setlength{\migy}{#2 \unitat}%
     \addtolength{\migy}{#4 \unitat}%
     \setlength{\migy}{0.5\migy}%
     %
   % ----- Treiem les unitats
     \setlength{\unitlength}{\factor pt}%
     \treurePt[2]{\migx}%
     \setcounter{clxm}{\value{cres}}%
     \treurePt[2]{\migy}%
     \setcounter{clym}{\value{cres}}%
   % -----  Dinuix de la linia
     \put(0,0){\qbezier(#1,#2)(\value{clxm},\value{clym})(#3,#4)}%
 }
```

Problemes

> A l'usar `\qbeziers` per dibuixar la línia, el traç d'aquesta queda més brut (bast) que si s'usa `\line`

C.3 Gestió de piles amb LaTeX

El LaTeXusa un mecanisme intern de piles per mantenir visibilitats locals dels objectes. Així per construir una pila d'usuari n'hi ha prou en tenir un objecte (per exemple, un `command`) definit en diversos àmbits: un `push` significa guardar el nou valor en l'àmbit actual i obrir un nou àmbit; `pop` significa tancar l'àmbit actual i recuperar el valor de l'objecte pila un cop s'ha tancat l'àmbit. La taula 44 mostra el codi de les macros pertinents.

Si volem controlar l'error en el `pop`, convertim aquest en una funció oculta, i construïm les funcions públiques de la taula 45.

Exemple:

```
\newcommand{\res}{}
\creaPila{enters}
\creaPila{char}
\push{enters}{2}
\push{char}{a}
\push{char}{abc}
\push{char}{xyz}
\push{enters}{3}
\push{enters}{4}
\pop{enters}{\res}
\res
\pop{char}{\res}
\res
\pop{char}{\res}
\res
\pop{char}{\res}
\res
\pop{enters}{\res}
\res
\pop{char}{\res}%

\buidaPila{enters}
\buidaPila{char}
```

4
3
xyz
abc
a
"Error per pila buida"

Taula 44 Macros per a la gestió de piles

```
\newcommand{\nomPila}{}

\newcommand{\creaNomPila}[1]
 { % Donat el nom del comptador de la pila, deixa a \nomPila
   % el nom del contenidor
     \makeatletter
       \renewcommand{\nomPila}{\expandafter{\#1@}}
       \renewcommand{\nomPila}{}
     \makeatother
 }

\newcommand{\creaPila}[1]  % Comptador
  {% Creem el comptador per la pila
     \newcounter{#1}
   % Creem el nom del contenidor de la pila
     \creaNomPila{#1}
   \iniPila{#1}
  }

\newcommand{\iniPila}[1]
  {\setcounter{#1}{1}}

\newcommand{\push}[2]
  {\stepcounter{#1}
   \creaNomPila{#1}
     \makeatletter
          \renewcommand{\nomPila}{#2}
      \makeatother
   \begingroup
   }

\newcommand{\popInt}[2]
 {\creaNomPila
    \makeatletter
       \endgroup
        \edef#2{\nomPila}
   \makeatother
  \addtocounter{#1}{-1}
   }

\newcommand{\buidaPila}[1]
  {\whiledo{\not{\value{#1}=1}}
    {\addtocounter{#1}{-1}
      \endgroup
    }
  }
```

Taula 45 Control de pila buida

```
\newcommand{\pbuida}{}
\newcommand{\errorPila}{``Error per pila buida''}

\newcommand{\pop}[2]
    {\pilabuida{#1}
     \ifthenelse{\equal{\pbuida}{true}}
           {\errorPila}
           {\popInt{#1}{#2}}
    }

 \newcommand{\pilabuida}[1]
    {
      \ifthenelse{\equal{\value{#1}}{1}}
           {\renewcommand{\pbuida}{true}}
          {\renewcommand{\pbuida}{false}}
    }
```

Problemes

1. La gestió de la pila a través dels àmbits de visibilitat del LaTeX (`groups`) significa que sempre que usem una pila dins d'un àmbit, caldrà buidar la pila abans de tancar l'àmbit o entorn envolvent si no volem que es produeixi un error. Per fer-ho necessitem un comptador que ens mantingui l'alçada de la pila

   $\boxed{\Longrightarrow}$ Cal no oblidar-se, doncs, d'usar `\budaPila` cada cop que sortim d'un entorn.

2. Desempilar en una pila significa tancar un entorn. Pel LaTeX tancar l'entorn significa desfer totes les assignacions que es pugin considerar locals. $\boxed{\Longrightarrow}$ Això significa que *tota definició feta entre un* `push` *i un* `pop` *només té vigència entre aquest* `push` *i aquest* `pop`. Heus ací un exemple:

```
\newcommand{\prova}{Valor inicial}
\creaPila{prova}
\prova\\
\push{prova}{}
\renewcommand{\prova}{Nou valor}
\prova\\
\pop{prova}{\resultat}
\prova\\
```

Valor inicial Nou valor Valor inicial

D El codi de c$^{peix}_{ov}$e

```
% ELX N=3 (Peix al Cove)          elxN3.tex
%
% Josep M. Merenciano
% Juny 1999
% ----------------------------------------------




% ==================================================
% ==================================================
% ==================================================
%            COMPOSICIO
% ==================================================
% ==================================================
% ==================================================
%
%
% DESCRIPCIO GLOBAL
%
% ==================================================
%
%
% -------------- Comandes ------------------------
%
%   \x[re](x,y,definicio)   Situa un objecte en la posicio inidcada
%
%
% --------------- Entorn de composicio -------------
%
%    bloc
%
%        Ell mateix obre l'entorn xarxa si es necessari
%
%
% --------------- Definicio de l'entorn ----------------
%

% ==================================================


% ==================================================
% DEFINICIO DE L'ENTORN
% ==================================================

%  \composicioXarxes
%       no: Cada bloc predefinit constitueix una xarxa independent
```

```
%        x : Els blocs predefinits es situen tots en una mateixa xarxa
%            envolvent
%
%  \autoposicionament
%        * si : afegeix \distMin entre dos blocs consecutius
%          x  : juxtaposa els blocs segons la seva magnitut
%               horitzontal
%
%   x== Qualsevol altre valor dels especificats
%   * Valor per omissio

\newcommand{\composicioXarxes}{}
\newcommand{\autoPosicionament}{}

\newcommand{\composar}
   {\renewcommand{\composicioXarxes}{x}}
\newcommand{\NOcomposar}
   {\renewcommand{\composicioXarxes}{no}}


\newcommand{\autoPosicionar}
   {\renewcommand{\autoPosicionament}{x}}
\newcommand{\NOautoPosicionar}
   {\renewcommand{\autoPosicionament}{no}}



% ---------------------------------------------------
%       ENTORN PER OMISSIO
% ---------------------------------------------------
       \composar
       \autoPosicionar

     % --- Separacio entre blocs que afegeix l'\autoposicionament
           \newcommand{\distMin}{0.5}

% ---------------------------------------------------
%       Valors intern de l'entorn
% ---------------------------------------------------

% ---------- Es poden usar externament ----------

     % --- Mesures de la caixa envolvent de cada unitat de xarxa (els
     %     anomenats blocs)
           \newcommand{\xh}{0}
           \newcommand{\xv}{0}

% ===================================================
%      Comandes  EXTERNES per a la composicio
% ===================================================
```

```
\newenvironment{bloc}
   {\fesComposicio}
   {\preparaNovaComposicio}

% ---------------------------------------------------------------------------
% \x situa dins d'una xarxa les definicions parcials. Aixo
% permet composar les xarxes amb facilitat
%
% \x[rel](x,y,definicio)
%
%        rel=  {}   posicionament Absolut
%              x    posicionament relatiu
% ---------------------------------------------------------------------------

% Desplacem el punt d'ancoratge segons el que ens diu el darrer bloc usat
\newcommand{\x}[4][{}]%
{% --- Canviem localment l'autoposicionament
 {\NOautoPosicionar%
  \begin{bloc}%
    \ifthenelse{\equal{#1}{}}%
       {\posicioAbs{#3}{#4}}%
       {\posicioRel{#3}{#4}}%
     #2%
  \end{bloc}%
 }%
}%

% ---------------------------------------------------
%          COMANDES INTERNES PER A LA COMPOSICIO
% ----------------------------------------------------
%     \fesComposicio   : S'executa just despres de definir les
%                        mesures del bloc, i just abans de
%                        contruir-lo. Crea una xarxa, si cal,
%                        i es posiciona en el punt d'ancoratge de la
%                        xarxa global.
%     \preparaNovaComposicio : S'executa un cop construit tot el
%                        bloc. Recupera les coordenades anteriors
%                        a la construccio del bloc, i modifica
%                        convenientment els parametres de la xarxa
%                        global.
% ----------------------------------------------------

\newcommand{\fesComposicio}
  {\ifthenelse{\equal{\composicioXarxes}{no}}%
      {\begin{xarxa}{\xh}{\xv}}%
      {}%
    \guardaPosicio% <<<<<-------
  }
```

```
\newcommand{\preparaNovaComposicio}
 {\recuperaPosicio%  <<<<<<-------
  %
   \ifthenelse{\not\equal{\autoPosicionament}{no}}%
        {\posicioRel{\distMin}{0}%
         \posicioRel{\xh}{0}%
         \posicioRel{1}{0}%
        }%
        {}%
   \ifthenelse{\equal{\composicioXarxes}{no}}%
       {\end{xarxa}}%
       {}%
 }


% ---------------------------------------------------
%  \guardaPosicio       Situa en la pila de posicions
%                       les coordenades actuals
%  \recuperaPosicio     Desempila una posicio, i la
%                       converteix en les coordenades actuals
% ---------------------------------------------------

\creaPila{posicions}
\newcommand{\posicioTemp}{}

\newcommand{\recuperaPosicio}
   {\pop{posicions}{\posicioTemp}%
    \global\setlength{\coorX}{\posicioTemp   pt}%
    \pop{posicions}{\posicioTemp}%
    \global\setlength{\coorY}{\posicioTemp pt}%
}

\newcommand{\guardaPosicio}
  {\treurePt[0]{\coorY}%
    \push{posicions}{\resultat}%
    \treurePt[0]{\coorX}%
   \ifthenelse{\coorX<0}
     {\push{posicions}{-\resultat}}%
     {\push{posicions}{\resultat}}%
  }





% ===================================================
% ===================================================
% ===================================================
```

```
% ==================================================
%
%    NUCLI de les xarxes
%
% ==================================================

% ==================================================
% ==================================================
% ==================================================


      \newcommand{\distanciaEntrePortes}{2}

% ==================================================
%  CONTENIDORS dels objectes del nucli  (nus)
% ==================================================

% ----- Permutacio automatica -----

      \newcommand{\calPermutacioAutomatica}{}

% ----- Etiqueta sintactica -----
%
%  Apareix en les arestes

\newcommand{\etEsq}{}
\newcommand{\etDreta}{}
\newcommand{\etSota}{}

%  -----  Tipus  semantics -----
%
%  Apareix en vertexs, lateralment

\newcommand{\etEsqTip}{}
\newcommand{\etDretaTip}{}

%  -----  Etiqueta  semantica (lambda terme) -----
%
%  Apareix sota  dels vertexs coresponents a conclusions obertes
%  Es passa per parametre; no te contenidor propi
%  L'us de \etSem com a contenidor permet controlar la seva aparicio

\newcommand{\etSem}{}

%  ---- Variables semantiques -----
%
%  Apareixen com a subindex de les etiquetes sintactiques a la pota
%  argument dels nodes par. No tenen contenidor propi

   \newcommand{\varSem}{}
```

```
        % Posicio de subindex on apareixen

             \newcommand{\posVarSem}{3pt}

% -----  Ordre prosodic  -----
%
% Apareix com a simbol central del nus. Es pot combinar amb el lligam

       \newcommand{\ordreProsodic}{}

       % Els valors que pot contenir sn:

            \newcommand{\ordreEsqDreta}    % --->
                {{\setlength{\unitlength}{\distHorit}%
                  \addtolength{\unitlength}{-2\midaRodoneta}%
               \addtolength{\unitlength}{-\distHorEtiqAuxEntrada}%
                 \raisebox{-3pt}{\makebox[\unitlength]{$\rightarrow$}}%
                 }}%

             \newcommand{\ordreDretaEsq}   % <---
                {{\setlength{\unitlength}{\distHorit}%
                  \addtolength{\unitlength}{-2\midaRodoneta}%
                  \addtolength{\unitlength}{-\distHorEtiqAuxEntrada}%
                  \raisebox{-3pt}{\makebox[\unitlength]{$\leftarrow$}}%
                 }}%


% -----  Lligam entre les potes  -----
%
% Apareix com a simbol central del nus. Es pot combinar em l'ordre

    \newcommand{\lligam}{}

    \newcommand{\ctLligam}
       {{\setlength{\unitlength}{\distHorit}%
                  \addtolength{\unitlength}{-2\midaRodoneta}%
                  \addtolength{\unitlength}{-\distHorEtiqAuxEntrada}%
                  \raisebox{3pt}{\makebox[\unitlength]{$\ldots\ldots$}}%
        }}%

% -----  Numeracio dels nodes ------
%
%    Apareix a sota de les rodonetes en les potes superiors, i al seu
%    costat esquerre en les inferiors

    \newcounter{comptadorXarxa}
    \setcounter{comptadorXarxa}{0}

% ==================================================
```

```
%
%    COMMUTADORS de control
%
% ==================================================

% ----- Permutacio automatica dels nuclis ------

      \newcommand{\permutacioAtuomatica}{si}

% ----- Aparicio de l'etiqueta sintactica -----

       \newcommand{\calEtEsq}{}
       \newcommand{\calEtDreta}{}
       \newcommand{\calEtSota}{}

  %  Meta--commutador
         \newcommand{\calMostrarEtSint}{x}

% ----- Aparicio del tipus semantic  -----

       \newcommand{\calEtEsqTip}{}
       \newcommand{\calEtDretaTip}{}

  %  Meta--commutador
         \newcommand{\calMostrarTipSem}{x}

% ----- Aparicio de les etiquetes semantiques (no esborrat de \etSem   -----

       \newcommand{\calMostrarEtSem}{}

% ----- Aparicio de l'ordre prosodic -----

\newcommand{\calMostrarOrdre}{}

% ----- Aparicio de les variables semantiques -----

\newcommand{\calMostrarVar}{}

% ---- Calcul automatic del numero de pota
%
%   Qualsevol valor s'interpreta com el valor a visualitzar; en blanc
%   es calcula automaticament. Els valors per omissio son els
%   que s'apliquen sempre, excepte que haguem usat \numeracio
%   per canviar temporalment (una aplicacio) el valor del parametre

    \newcommand{\comptSota}{}
    \newcommand{\comptDreta}{}
    \newcommand{\comptEsq}{}

    \newcommand{\comptSotaPerOmissio}{}
```

```
    \newcommand{\comptDretaPerOmissio}{}
    \newcommand{\comptEsqPerOmissio}{}

    % -- Meta-commutador

        \newcommand{\calMostrarNumeracio}{}

% ==================================================
% PERSONALITZACIO de l'entorn
% ==================================================

% --------------------------------------------------
%
%    Inicialitzacions predefinides de les variasbles d'entorn
%
% --------------------------------------------------

\newcommand{\xarxaEstandard}
{\ambEtiquetes%
 %-- commutadors
   \permutacioAutomatica
   \NOmostrarOrdre%
   \NOmostrarVar%
   \mostrarEtSem%
   \numeracioPerOmissio{{}}{{}}{{}}%  No numerem
 %-- meta-commutadors
   \mostrarEtSint%
   \mostrarTipSem%
   \mostrarNumeracio%
 %-- Contenidors de manipulacio externa
   \renewcommand{\varSem}{x}%
   \renewcommand{\etSem}{}%
}


% ==================================================
%
%      ACTIVADORS EXTERNS  DEL NUCLI
% ==================================================
%
\newcommand{\permutacioAutomatica}{\renewcommand{\calPermutacioAutomatica}{si}}
\newcommand{\NOpermutacioAutomatica}{\renewcommand{\calPermutacioAutomatica}{}}

\newcommand{\calEtSint}[3]
  {\renewcommand{\calEtSota}{#1}%
   \renewcommand{\calEtEsq}{#2}%
   \renewcommand{\calEtDreta}{#3}%
  }

\newcommand{\mostrarEtSint}{\renewcommand{\calMostrarEtSint}{x}}
```

```
\newcommand{\NOmostrarEtSint}{\renewcommand{\calMostrarEtSint}{}}

\newcommand{\calTipSem}[2]
  {\renewcommand{\calEtEsqTip}{#1}%
   \renewcommand{\calEtDretaTip}{#2}%
  }

\newcommand{\mostrarTipSem}{\renewcommand{\calMostrarTipSem}{x}}
\newcommand{\NOmostrarTipSem}{\renewcommand{\calMostrarTipSem}{}}

\newcommand{\mostrarEtSem}{\renewcommand{\calMostrarEtSem}{x}}
\newcommand{\NOmostrarEtSem}{\renewcommand{\calMostrarEtSem}{}}

\newcommand{\mostrarOrdre}{\renewcommand{\calMostrarOrdre}{x}}
\newcommand{\NOmostrarOrdre}{\renewcommand{\calMostrarOrdre}{}}

\newcommand{\mostrarVar}{\renewcommand{\calMostrarVar}{x}}
\newcommand{\NOmostrarVar}{\renewcommand{\calMostrarVar}{}}

\newcommand{\mostrarLligam}{\renewcommand{\lligam}{\ctLligam}}

\newcommand{\mostrarNumeracio}{\renewcommand{\calMostrarNumeracio}{x}}
\newcommand{\NOmostrarNumeracio}{\renewcommand{\calMostrarNumeracio}{}}


% -------------------------------------------------
%
%   \etTipSem(tipus argument, tipus functor)
%
%  L'aparicio del tipus semantic es controla amb \mostrarTipSem
% -------------------------------------------------

\newcommand{\etTipSem}[2]
  {\renewcommand{\etEsqTip}{#1}%
   \renewcommand{\etDretaTip}{#2}%
  }

% -------------------------------------------------
%    \numeracioPerOmissio(numero pota de sota, numero pota esq,
%               numero pota dreta)

%    \numeracio(numero pota de sota, numero pota esq,
%               numero pota dreta)
%
%        numero={}  => numeracio automatica
%
%   Els valors especificats son vigents fins al seu us
%   Llavors es reinicialitzen al valor per omissio, fixada
%   per \numeracioPerOmissio
% -------------------------------------------------
```

```
\newcommand{\numeracio}[3]
 {\renewcommand{\comptSota}{#1}%
  \renewcommand{\comptEsq}{#2}%
  \renewcommand{\comptDreta}{#3}%
 }

\newcommand{\numeracioPerOmissio}[3]
 {\renewcommand{\comptSotaPerOmissio}{#1}%
  \renewcommand{\comptEsqPerOmissio}{#2}%
  \renewcommand{\comptDretaPerOmissio}{#3}%
  \numeracio{#1}{#2}{#3}
 }

% ===================================================
%
%      CONSTRUCCIO DEL NUCLI
%
% ===================================================

% ---------------------------------------------------
%           nucli[permutacio] (x,y, nus (4), connectiva(5)
%                               argument(6), functor(7))
%                               parentitzacio(8), etSemantica(9)
% ---------------------------------------------------

\newcommand{\xuesq}{}
\newcommand{\xudreta}{}
\newcommand{\yu}{}


\newcommand{\nucli}[9][{}]
{\construeixEtiquetes[#8]{#5}{#6}{#7}%
 \decideixNus[#1]{#4}{#5}%  % Pot permutar les etiquetes
 \decideixOrdre%
 \defineixEtiquetes%       % Esborra les que no s'han de visualitzar,
                           % i genera les etiquetes de les variables
 % --- Esborra \etSem
   \ifthenelse{\equal{\calMostrarEtSem}{}}%
        {\renewcommand{\etSem}{}}%
        {}%
 %
 % --- Nus amb les cat sintactiques, l'etiqueta semantica
 %     i les variables
 %
    \nus[#9]{\nusNucli}{#2}{#3}{$\etSota$}{$\etEsq$}{$\etDreta$}%
 %
 % --- Tipus semantics i ordre
 %
   \suma{1}{#3}%
```

```
   \edef\yu{\resultat}%
   \ifthenelse{\not\isodd{#3}}%
       {\suma{#2}{1}%
        \edef\xudreta{\resultat}%
        \renewcommand{\xuesq}{#2}%
       }%
       {\renewcommand{\xudreta}{#2}%
        \resta{#2}{1}%
        \edef\xuesq{\resultat}%
       }%
   \rodoneta{}{\xuesq}{\yu}{\etEsqTip}{\ordreProsodic}%
   \rodoneta{}{\xudreta}{\yu}{\lligam}{\etDretaTip}%
   %
   \renewcommand{\lligam}{}%
 %
 % --- Numeracio dels nodes
 %
   \rodoneta{}{#2}{#3}%  Sota
       {}{\hspace*{-\midaRodoneta}%
          \hspace*{\midaActualRodoneta}%
          \raisebox{-4\midaActualRodoneta}{\fesComptador{\comptSota}}%
         }%
   \rodoneta{}{\xuesq}{\yu}%  Esquerra
       {\raisebox{-6\midaActualRodoneta}{\fesComptador{\comptEsq}}%
        \hspace*{-1.5\midaActualRodoneta}%
        \hspace*{-0.5\midaRodoneta}%
       }{}%
   \rodoneta{}{\xudreta}{\yu}%   Dreta
       {}{\hspace*{-0.25\midaActualRodoneta}%
          \hspace*{-1.5\midaRodoneta}%
          \raisebox{-6\midaActualRodoneta}{\fesComptador{\comptDreta}}%
         }%
 }



% ==================================================
%
%      CONSTRUCCIO dels objectes del nucli
%--------------------------------------------------
%
% Op. privades
% ------------
%
%  \construeixEtiquetes
%          Coonstrueix les etiquetes sintactiques
%
%  \fesComptador
%
%         Genera la numeracio dels nodes
```

```
%
% ==================================================

% --------------------------------------------------
%  \construeixEtiquetes[parentesis](connectiva, argument, functor)
%
% --------------------------------------------------
%   Inicialitza \etEsq, \etDreta, \etSota,
%   \etEsqTip i \etDretaTip  considerant
%   que tenim un \nustimes no permutat.
%
%   La \parametritzacio indica com s'ha de parentitzar al
%   construir l'etiqueta de sota
%
%         * 00: No es parentitza
%           01: Parentitzem el functor
%           10: Parentitzem l'argument
%           11: Parentitzem argument i functor
% --------------------------------------------------

\newcommand{\etArgAux}{}
\newcommand{\etFunctorAux}{}

\newcommand{\construeixEtiquetes}[4][00]
{%---- Parentitzem -------
 \ifthenelse{\equal{#1}{00}\or\equal{#1}{}}%
     {\renewcommand{\etArgAux}{#3}%
      \renewcommand{\etFunctorAux}{#4}%
     }{}%
 \ifthenelse{\equal{#1}{01}}%
     {\renewcommand{\etArgAux}{#3}%
      \renewcommand{\etFunctorAux}{(#4)}%
     }{}%
 \ifthenelse{\equal{#1}{10}}%
     {\renewcommand{\etArgAux}{(#3)}%
      \renewcommand{\etFunctorAux}{#4}%
     }{}%
 \ifthenelse{\equal{#1}{11}}%
     {\renewcommand{\etArgAux}{(#3)}%
      \renewcommand{\etFunctorAux}{(#4)}%
     }{}%
 % ----- Construccio etiquetes -----------
 \ifthenelse{\equal{#2}{sobre}}%
    {\renewcommand{\etEsq}{#4}%
     \renewcommand{\etDreta}{#3}%
     \renewcommand{\etSota}{\etFunctorAux\sobre\etArgAux}%
     %
     \let\qq=\calEtEsqTip%
     \let\calEtEsqTip=\calEtDretaTip%
     \let\calEtDretaTip=\qq%
```

```
     %
     \let\qq=\etEsqTip%
     \let\etEsqTip=\etDretaTip%
     \let\etDretaTip=\qq%
    }%
    {\renewcommand{\etEsq}{#3}%
     \renewcommand{\etDreta}{#4}%
     \renewcommand{\etSota}{\etArgAux\sota\etFunctorAux}%
    }%
 % ------ Redefinicio de les etiquetes en el cas de no
 %        tenir una connectiva
 %
 %            Per raons que se m'escapen la comparacio directa
 %            dona error. Per aixo comparo longituts
 %
    \settowidth{\xtemp}{#3}%
    \ifthenelse{\lengthtest{\xtemp=0pt}}%
              {\renewcommand{\etSota}{#4}}{}%
    \settowidth{\xtemp}{#4}%
    \ifthenelse{\lengthtest{\xtemp=0pt}}%
              {\renewcommand{\etSota}{#3}}{}%
 % ---- Us de la meta-activacio ------
 %
  \ifthenelse{\equal{\calMostrarEtSint}{}}%
      {\renewcommand{\etEsq}{}%
      \renewcommand{\etDreta}{}%
      \renewcommand{\etSota}{}%
      }%
      {}%
}

% --------------------------------------------------
%       \fesComptador(quina pota)
%
%    La pota s'indica amb el commutador de control pertinent
%    (\comptSota, \comptEsq, \comptDreta). Si el commutador
%    no te cap valor s'utilitza un comptador intern (comptadorXarxa);
%    altrament s'usa el valor del commutador
% --------------------------------------------------

\newcommand{\fesComptador}[1]
 {\ifthenelse{\not\equal{\calMostrarNumeracio}{}}%
     {\ifthenelse{\equal{#1}{}}%
        {\stepcounter{comptadorXarxa}%
         \arabic{comptadorXarxa}%
        }%
        {#1}%
     }%
     {}%
  \recuperaComptador{#1}%
```

```
 }

\newcommand{\recuperaComptador}[1]
{\ifthenelse{\equal{#1}{\comptSota}}%
   {\global\let\comptSota=\comptSotaPerOmissio}{}%
 \ifthenelse{\equal{#1}{\comptEsq}}%
   {\global\let\comptEsq=\comptEsqPerOmissio}{}%
 \ifthenelse{\equal{#1}{\comptDreta}}%
   {\global\let\comptDreta=\comptDretaPerOmissio}{}%
}


% =====================================================
% Decissio del nus nuclear i POSICIONAMENT
%       dels objectes
% =====================================================
%
%  Op. publiques:
%        \decideixNus   --->   \nusNucli, etiquetes
%
%  Op. privades:
%        \decideixPermutacioNus   --> \calPermutarNus
%        \seleccioNus             -->  \nusNucli, \calPermutarEtiq
%         \permutacioEtiquetes    -->  etiquetes
% -----------------------------------------------------

      \newcommand{\nusNucli}{}

% ----------  Controls interns ---------------
      \newcommand{\calPermutarNus}{}
      \newcommand{\calPermutarEtiq}{}

% -----------------------------------------------------
%
%   decideixNus [permutacio] (nus de partida(2), connectiva(3) )
%
%       Retorna el resultat en \nusNucli, i modifica les etiquetes
%          (sintactiques, i de tipus semantic) de les potes superiors
%          en consonancia amb la permutacio
% -----------------------------------------------------

\newcommand{\decideixNus}[3][{}]
{\decideixPermutacioNus{#1}{#3}%
 \seleccioNus[#1]{#2}{#3}%
 \permutacioEtiquetes%
}

% -----------------------------------------------------
%    \decideixPermutacioNus(permutacio,  connectiva)
%
```

```
%         Retorna el resulata a \calPermutarNus
% --------------------------------------------------

\newcommand{\decideixPermutacioNus}[2]
{\ifthenelse{\equal{\calPermutacioAutomatica}{si}}
   {\ifthenelse{\equal{#2}{sobre}}%
         {\renewcommand{\calPermutarNus}{no}}%
         {\renewcommand{\calPermutarNus}{si}}%
   }%
   {\renewcommand{\calPermutarNus}{no}}%
 %
 \ifthenelse{\equal{#1}{}}%
     {}%
     {\ifthenelse{\equal{\calPermutarNus}{si}}%
        {\renewcommand{\calPermutarNus}{no}}%
        {\renewcommand{\calPermutarNus}{si}}%
     }%
}

% --------------------------------------------------.
%     \seleccioNus[permutacio]( nus de partida, connectiva)
%
%    Usa \calPermutarNus; retorna \nusNucli i \calPermutarEtiq
%--------------------------------------------------

\newcommand{\seleccioNus}[3][{}]
{\ifthenelse{\equal{#1}{}}%
     {\ifthenelse{\equal{#2}{\nustimes}}%
        {\renewcommand{\calPermutarEtiq}{no}}%
        {\renewcommand{\calPermutarEtiq}{si}}%
  %
     }%
     {\ifthenelse{\equal{#2}{\nustimes}}%
        {\renewcommand{\calPermutarEtiq}{si}}%
        {\renewcommand{\calPermutarEtiq}{no}}%
     }%
 %
 \ifthenelse{\equal{\calPermutacioAutomatica}{si}}%
     {}%
     {\ifthenelse{\equal{#3}{sobre}}
        {\renewcommand{\calPermutarEtiq}{no}}%
        {\renewcommand{\calPermutarEtiq}{si}}%
     }%
 %
 \ifthenelse{\equal{\calPermutarNus}{no}}%
    {\ifthenelse{\equal{#2}{\nustimes}}%
         {\renewcommand{\nusNucli}{\nustimes}}%
         {\renewcommand{\nusNucli}{\nuspar}}%
    }%
    {\ifthenelse{\equal{#2}{\nustimes}}%
```

```
        {\renewcommand{\nusNucli}{\nustimesPermutat}}%
        {\renewcommand{\nusNucli}{\nusparPermutat}}%
    }%
}

% -------------------------------------------------
%     \permutacioEtiquetes
%
%          Usa \calPermutarEtiq; modifica convenientment
%          les etiquetes de les potes
% -------------------------------------------------

\newcommand{\permutacioEtiquetes}
{ \ifthenelse{\equal{\calPermutarEtiq}{no}}%
    {}%
    {\let\merdaTemporal=\etDreta%
     \let\etDreta=\etEsq%
     \let\etEsq=\merdaTemporal%
     %
     \let\merdaTemporal=\etDretaTip%
     \let\etDretaTip=\etEsqTip%
     \let\etEsqTip=\merdaTemporal%%
     %
     \let\merdaTemporal=\calEtDretaTip%
     \let\calEtDretaTip=\calEtEsqTip%
     \let\calEtEsqTip=\merdaTemporal%%
     %
    }%
}

% -------------------------------------------------
%     \decideixOrdre
%
%          Usa \calPermutarEtiq ; retorna \ordreProsodic
% -------------------------------------------------
%
%  Considrem el model relacional
%
%     i-K: A  j-K:B            K-j: B   K-i: A
%    ---------------- K>j      ------------------   K<i
%      i-j :A/B                   i-j: B\A
%
%      i<j<K                        K<i<j
%
%    L'inici de A es <           L'acabament de la A
%    a l'inici de B              es < que l'acabament de
%                                la B
%
%              <                       <
%    Times:    A ---> B              A ----> B
```

```
%
%
%    Par:      A <--- B              A <---- B
%
%
%  *** Ho podem resumir amb  a(+) ---> s(-), de input a output
%
%  *** Aixi una xarxa genera un circuit tancat sobre l'ordre
%
% --------------------------------------------------

\newcommand{\decideixOrdre}
{\ifthenelse{\equal{\calPermutarNus}{no}}%
    {\renewcommand{\ordreProsodic}{\ordreEsqDreta}}%
    {\renewcommand{\ordreProsodic}{\ordreDretaEsq}}%
}

% ==================================================
%
%      ESBORRAT  dels objectes que no s'han de visualitzar
%
% ==================================================
%
%    \defineixEtiquetes
%
% --------------------------------------------------

% --------------------------------------------------
% \defineixEtiquetes
%
%   Esborra les etiquetes que no s'han de visualitzar
% --------------------------------------------------

\newcommand{\defineixEtiquetes}
{\defineixEtiquetesSint%
 \defineixEtiquetesTipSem%
 \defineixEtiquetesOrdre%
 \defineixEtiquetesVar%
}

% --------------------------------------------------
%    \defineixEtiquetesSint
% --------------------------------------------------

\newcommand{\defineixEtiquetesSint}
  {\ifthenelse{\equal{\calEtEsq}{}}%
       {\renewcommand{\etEsq}{}}%
       {}%
  \ifthenelse{\equal{\calEtDreta}{}}%
      {\renewcommand{\etDreta}{}}%
```

```
       {}%
  \ifthenelse{\equal{\calEtSota}{}}%
       {\renewcommand{\etSota}{}}%
       {}%
  }

% -------------------------------------------------
%     \defineixEtiquetesTipSem
%  -------------------------------------------------


\newcommand{\defineixEtiquetesTipSem}
  {\ifthenelse{\equal{\calEtEsqTip}{}}%
       {\renewcommand{\etEsqTip}{}}%
       {}%
  \ifthenelse{\equal{\calEtDretaTip}{}}%
       {\renewcommand{\etDretaTip}{}}%
       {}%
  \renewcommand{\calEtEsqTip}{}%
  \renewcommand{\calEtDretaTip}{}%
  \ifthenelse{\equal{\calMostrarTipSem}{}}%
      {\renewcommand{\etEsqTip}{}%
      \renewcommand{\etDretaTip}{}%
      }%
      {}%
  }

% -------------------------------------------------
%     \defineixEtiquetesOrdre
%  ------------------------------------------------

\newcommand{\defineixEtiquetesOrdre}
  {\ifthenelse{\equal{\calMostrarOrdre}{}}%
       {\renewcommand{\ordreProsodic}{}}%
       {}%
  }

% -------------------------------------------------
%     \defineixEtiquetesVar
%  ------------------------------------------------

\newcommand{\defineixEtiquetesVar}
 {\ifthenelse{\equal{\nusNucli}{\nuspar}\or\equal{\nusNucli}{\nusparPermutat}}%
   {\ifthenelse{\equal{\calMostrarVar}{}}%
      {}%
      {\ifthenelse{\equal{\calPermutarNus}{no}}%
         {\renewcommand{\merdaTemporal}%
            {\protect\raisebox{-\posVarSem}{\tiny \varSem}}%
          \let\qq=\etEsq
          \renewcommand{\etEsq}{\qq_{\makebox(0,0)[t]{\merdaTemporal}}}%
```

```
        }{}%
     \ifthenelse{\equal{\calPermutarNus}{si}}%
        {\renewcommand{\merdaTemporal}%
           {\protect\raisebox{\posVarSem}{\tiny \varSem}}%
          \let\qq=\etDreta
          \renewcommand{\etDreta}{{}_{\:\makebox(0,0)[t]{\merdaTemporal}}\qq}
         }{}%
     }%
  }{}%
 }


% ==================================================
% CONSTRUCCIO D'UNA XARXA DUAL
% ==================================================

   % NOTA: L'us de let enlloc de edef fa que els nusos sense
   %       rodonetes buclegi indefinidament

\newenvironment{xarxaDual}
    {\edef\qq{\nustimes}%
     \edef\nustimes{\nusparPermutat}%
     \edef\nusparPermutat{\qq}%
     %
     \edef\qq{\nustimesPermutat} %
      \edef\nustimesPermutat{\nuspar}%
     \edef\nuspar{\qq}%
    }
    {}
```