UNCUYO
UNIVERSIDAD
NACIONAL DE CUYO

FACULTAD
DE INGENIERÍA

ARQUITECTURAS DISTRIBUIDAS

REDES DE INTERCONEXIÓN

CONECTIVIDAD

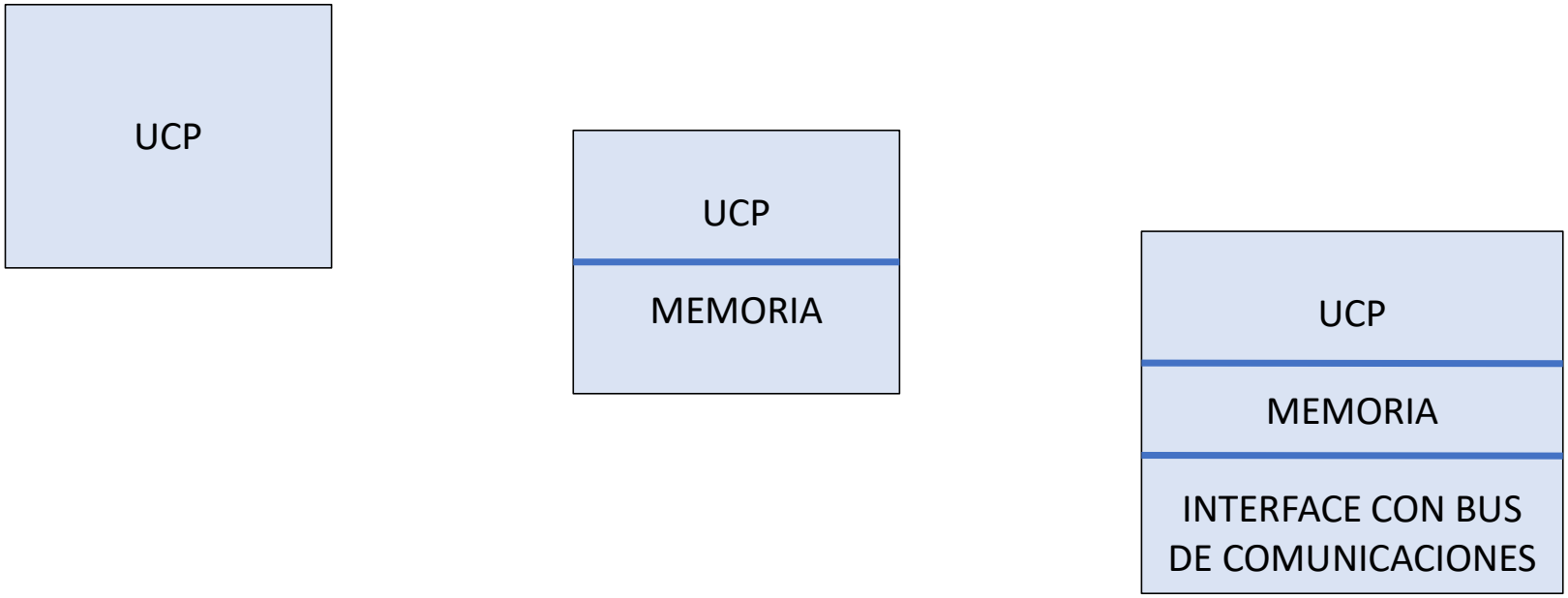
Una arquitectura paralelo se puede caracterizar según:

- Cantidad de elementos de procesamiento
- Red de interconexión entre los elementos de procesamiento
- Organización de la memoria.

UNCUYO
UNIVERSIDAD
NACIONAL DE CUYO

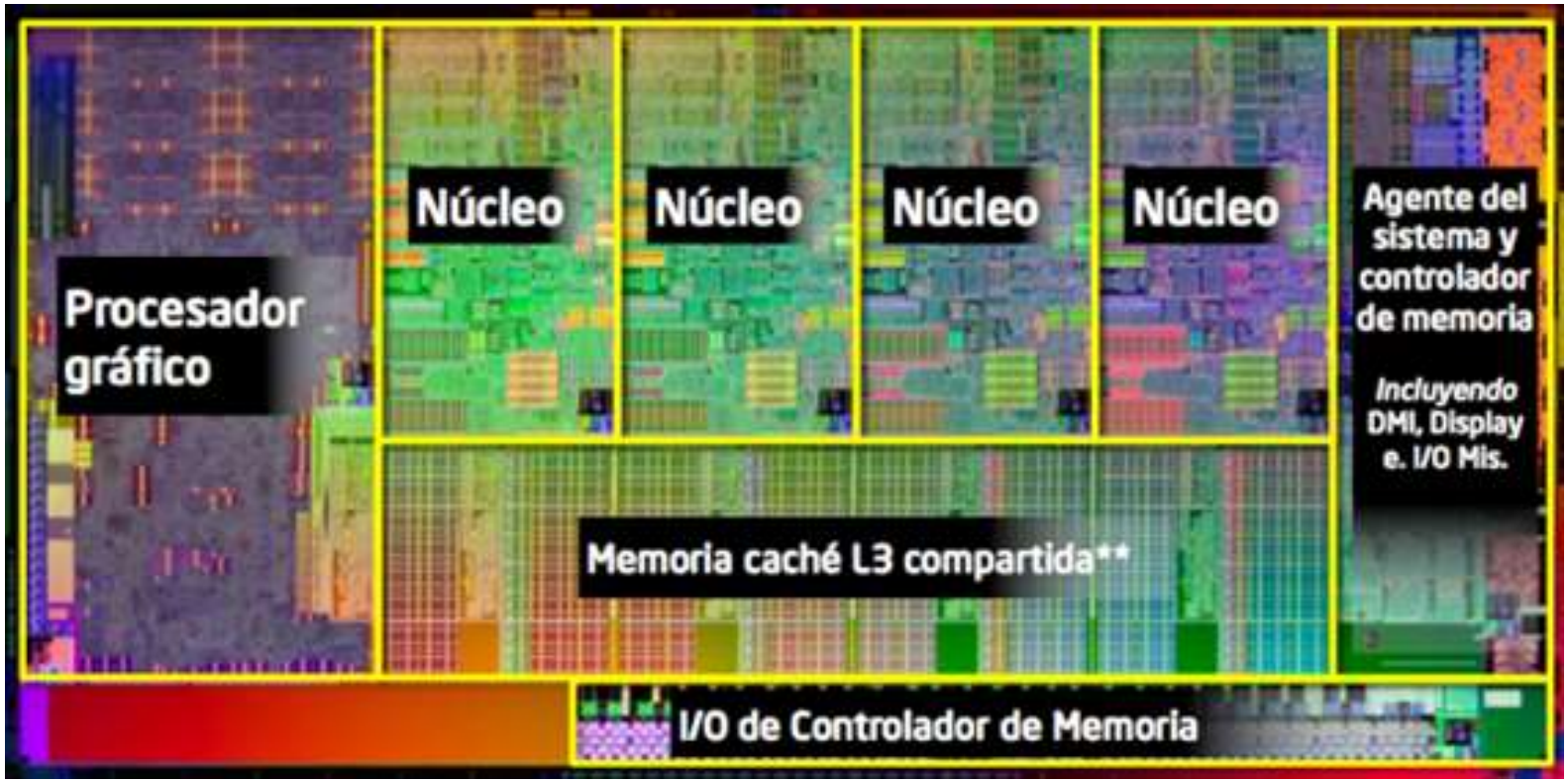
FACULTAD
DE INGENIERÍA

ELEMENTO DE PROCESAMIENTO



UNCUYO
UNIVERSIDAD
NACIONAL DE CUYO

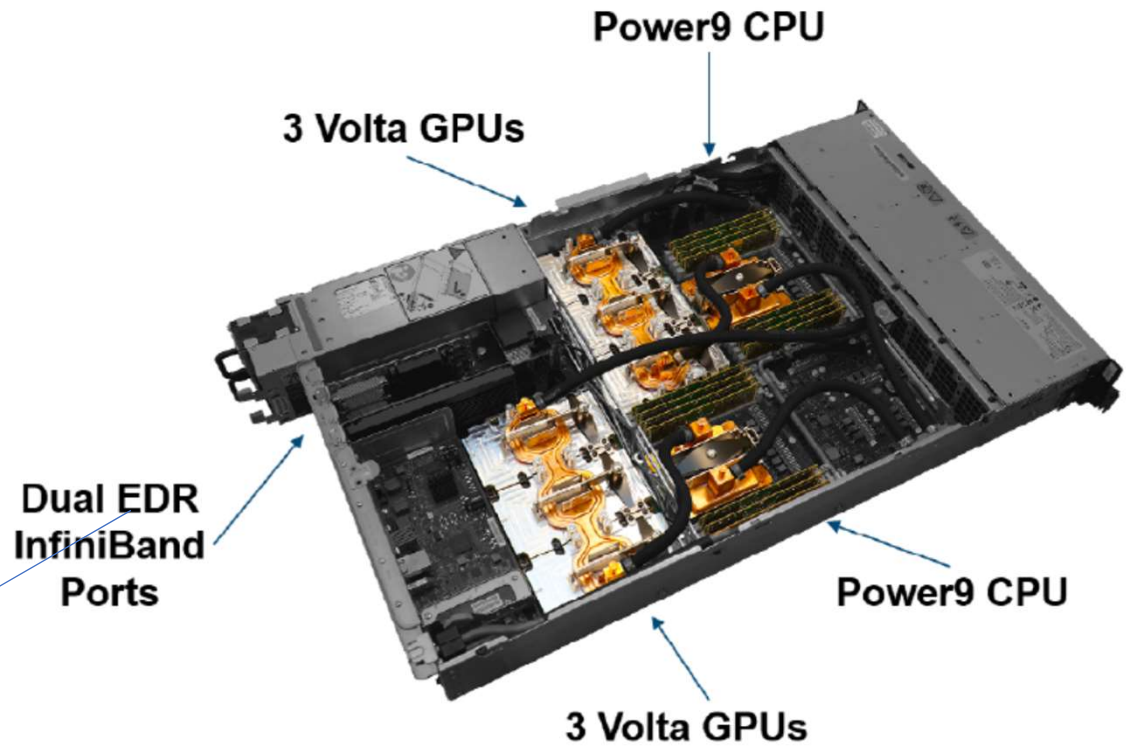
FACULTAD
DE INGENIERÍA



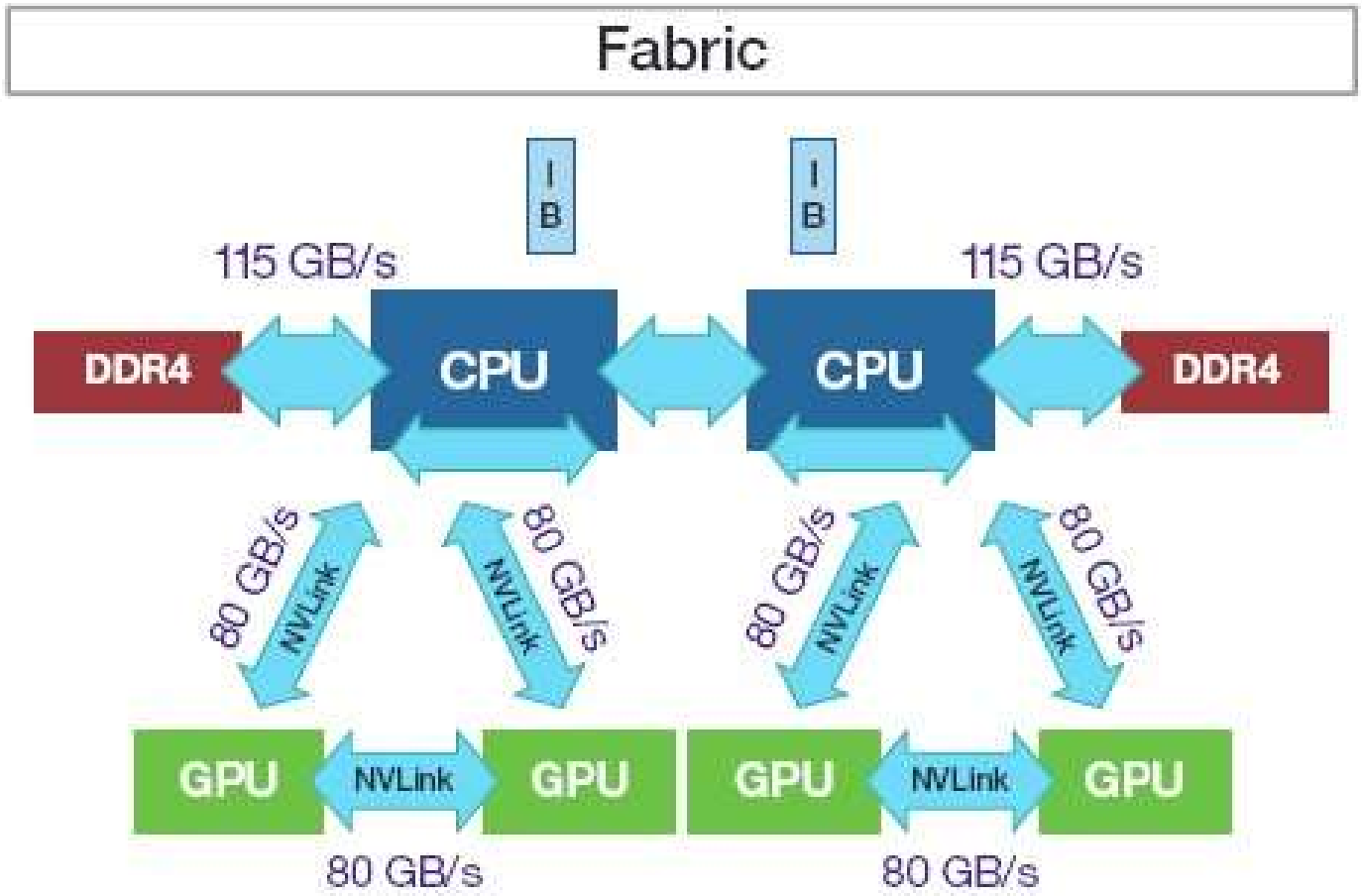
UNCUYO
UNIVERSIDAD NACIONAL DE CUYO

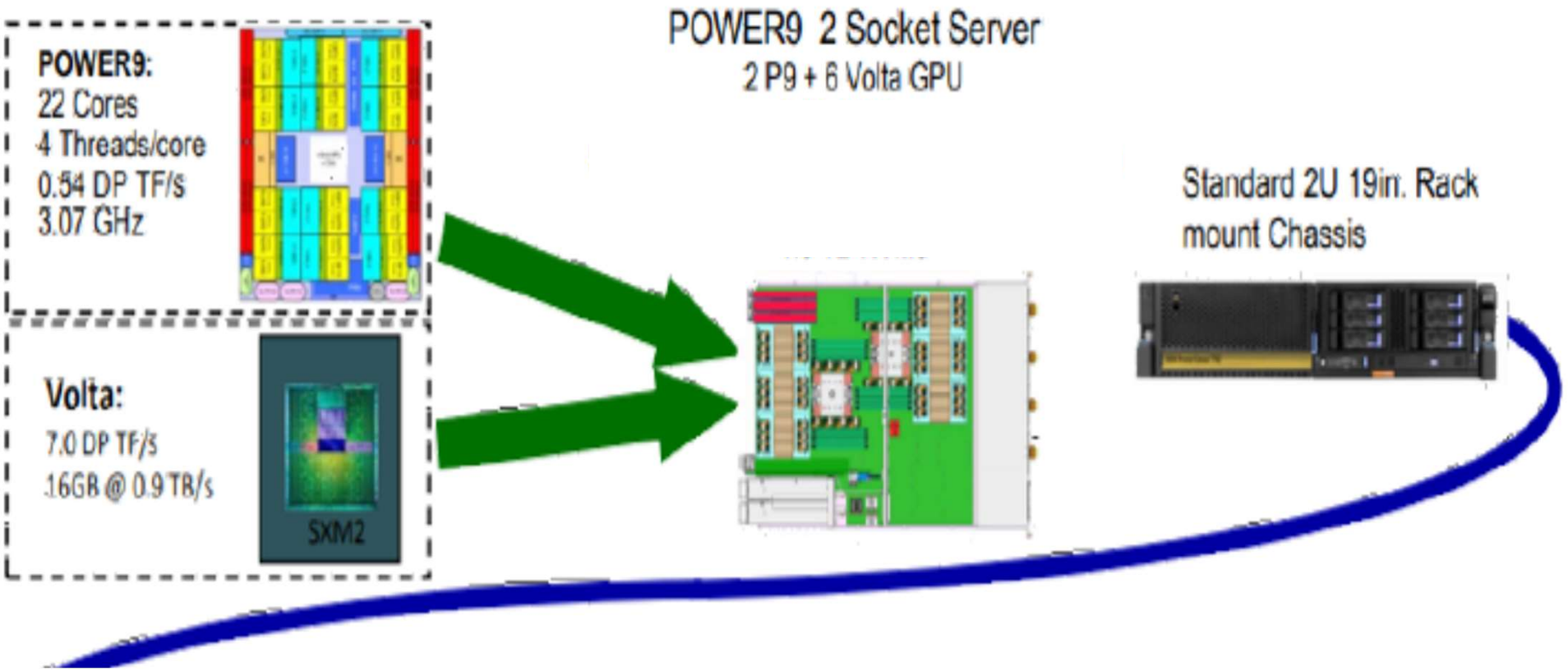
FACULTAD DE INGENIERÍA

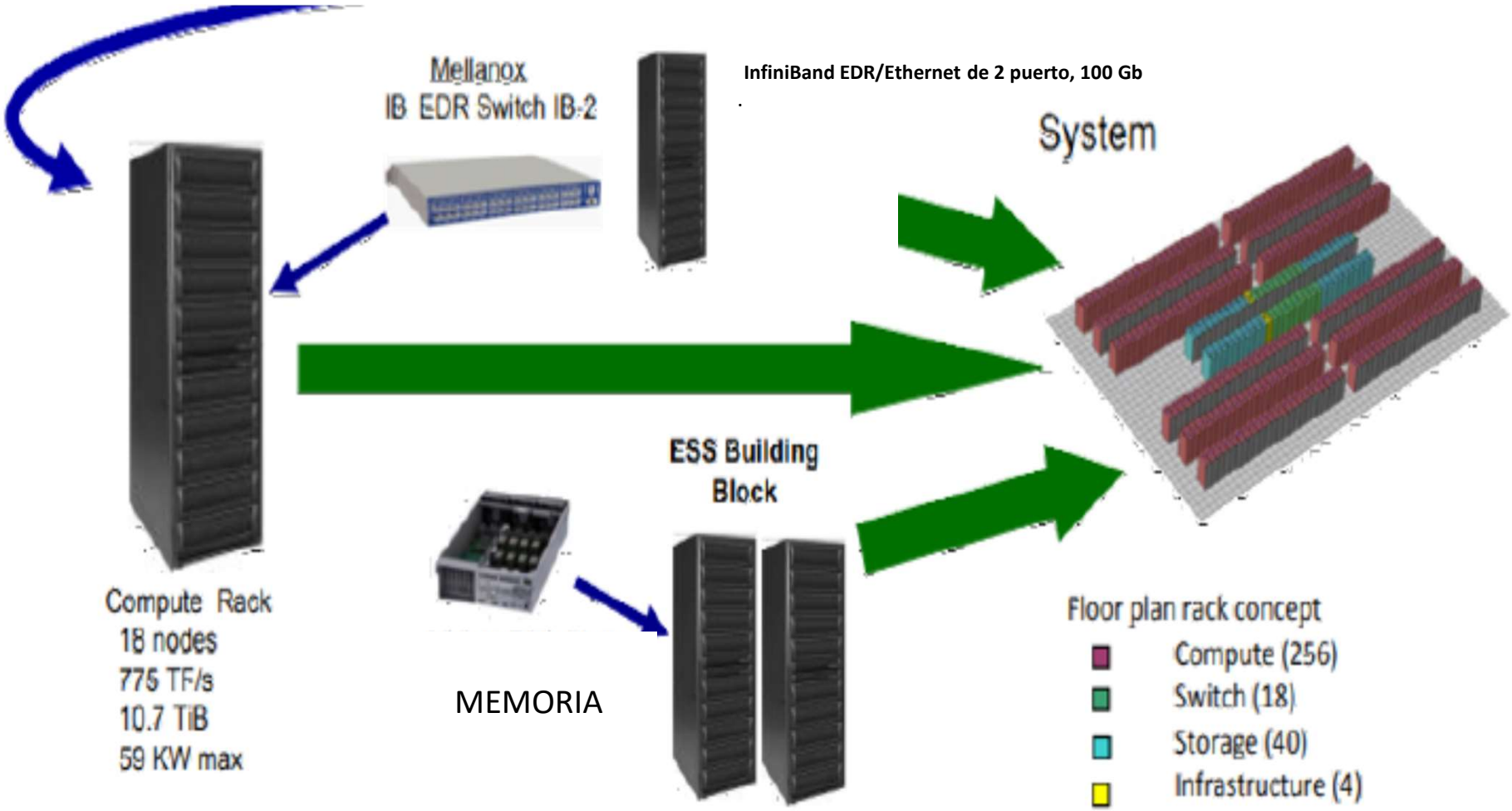
ELEMENTO DE PROCESAMIENTO



La velocidad total en Gbit/seg., incluida la sobrecarga, es decir, la información de señalización y control, se pueden enviar a través de un circuito.

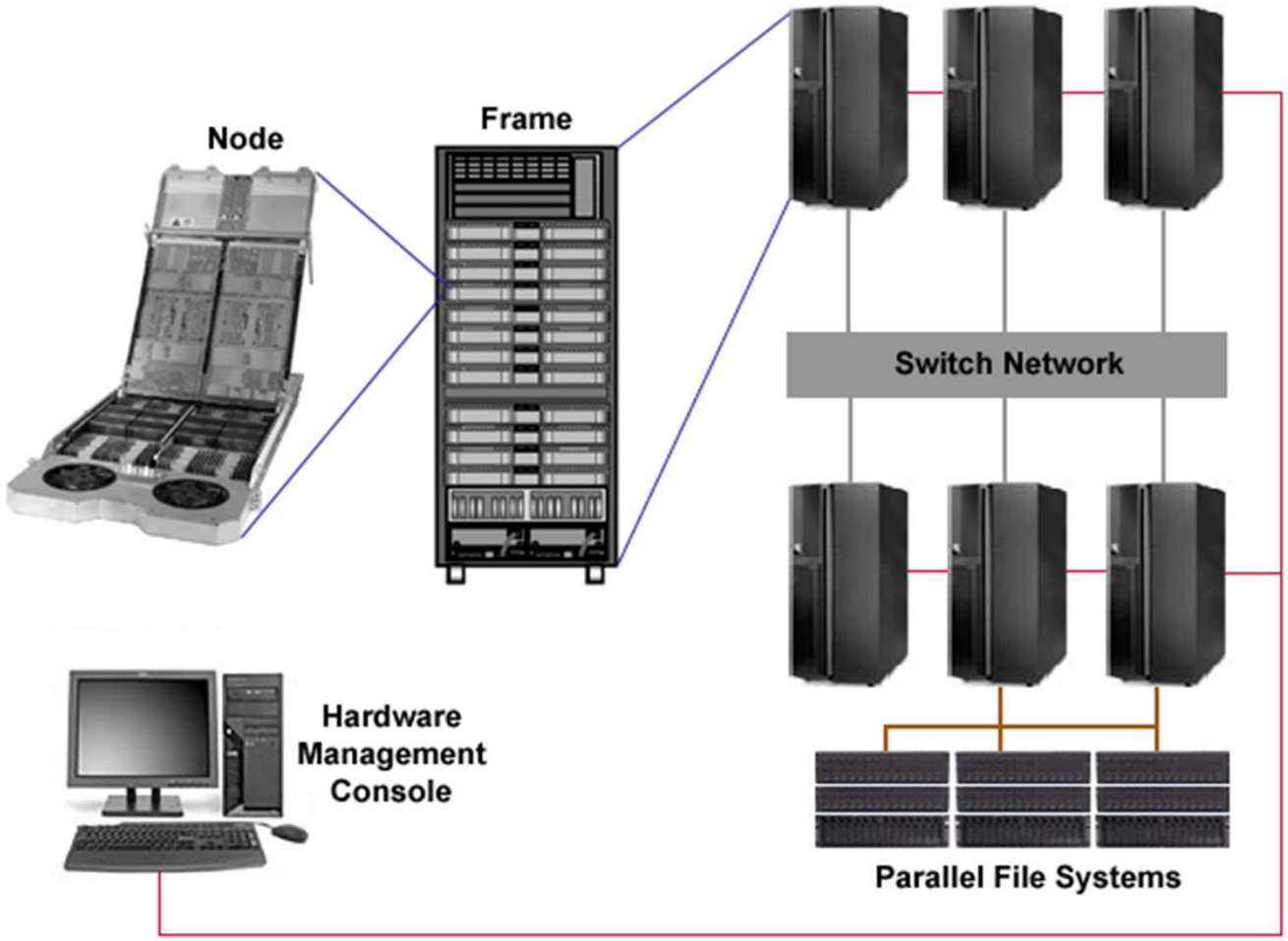

Fabric
IB
IB
115 GB/s
115 GB/s
DDR4
CPU
CPU
DDR4
80 GB/s
NVLink
80 GB/s
NVLink
80 GB/s
NVLink
80 GB/s
NVLink
GPU
NVLink
GPU
GPU
NVLink
GPU
80 GB/s
80 GB/s


POWER9:
22 Cores
4 Threads/core
0.54 DP TF/s
3.07 GHz
Volta:
7.0 DP TF/s
16GB @ 0.9 TB/s
SXM2
POWER9 2 Socket Server
2 P9 + 6 Volta GPU
Standard 2U 19in. Rack
mount Chassis



InfiniBand EDR/Ethernet de 2 puerto, 100 Gb

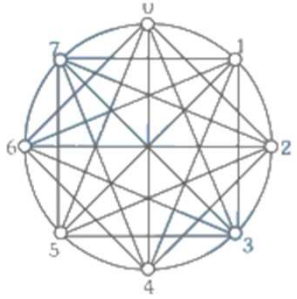
.

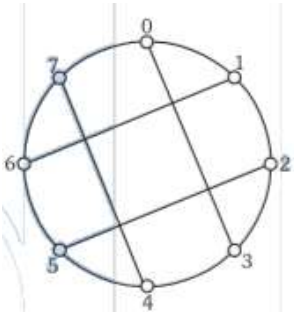

Node
Frame
Switch Network
Hardware
Management
Console
Parallel File Systems

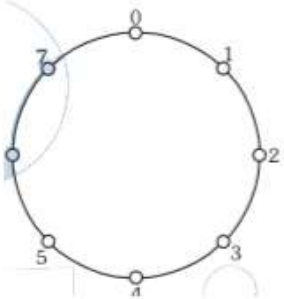
CONECTIVIDAD

Criterio para clasificar las redes está basado en su conectividad:
la conexión entre los diferentes elementos que necesitan conectarse entre si puede ser total o parcial.

El caso ideal, donde todos los elementos están conectados directamente unos con otros.
En la práctica, cuando el número de elementos crece, no es posible la conexión total y hay que conformarse con conexiones parciales.





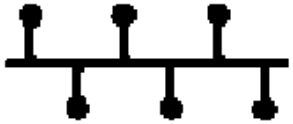


La red define una **topología** cuyas propiedades pueden explotarse para encaminar correcta y rápidamente los mensajes desde un procesador origen a un procesador destino.

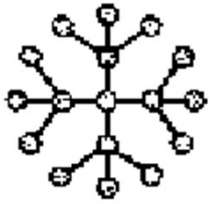
La existencia de una topología permite definir la **distancia** entre dos procesadores como el número de etapas a franquear o recorrer en la red para ir del procesador origen al destino.

En el **caso ideal** de topología totalmente conectada la **distancia** en todos los casos es **siempre 1**. El problema de la conectividad de la red se refleja en la dualidad *conexiones/contención*.

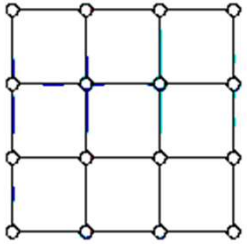
Cuantas **más conexiones** tenemos entre los procesadores **menos contención** presenta la red y viceversa. En el caso de topología totalmente conectada el número de cables (hilos) que posee la red es N^2, pero no tiene contención (su valor es 1)

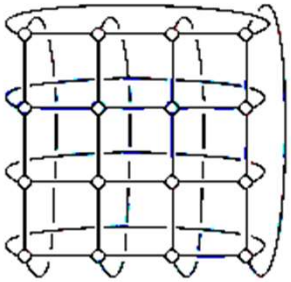


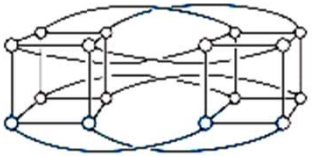
En el caso de la topología de bus, es la más sencilla de realización pues sólo utiliza un hilo, pero es la topología que presenta la mayor contención (de valor N^2). Debido a ello se buscan otras topologías que presenten características intermedias en ambos casos.



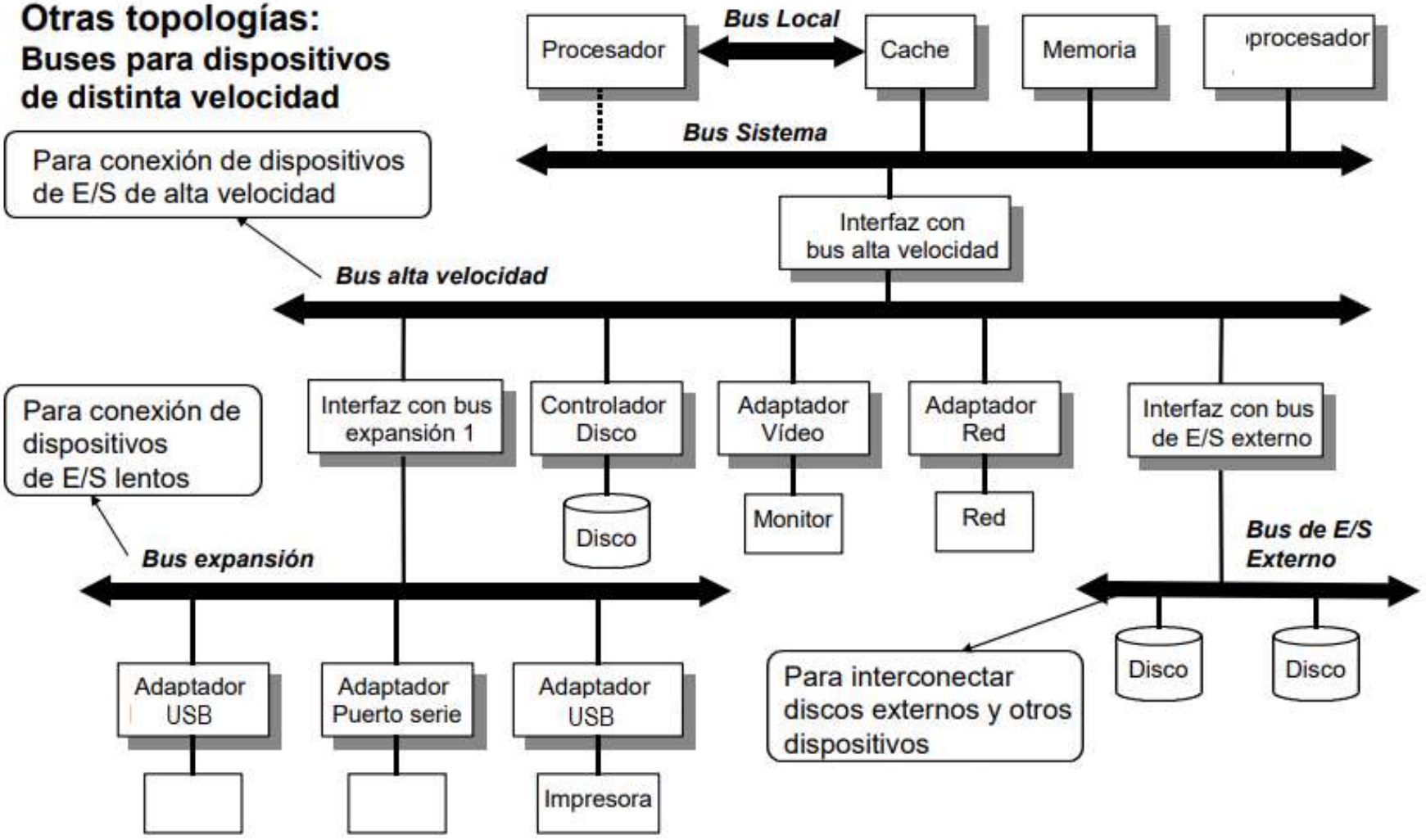
Topología en estrella extendida





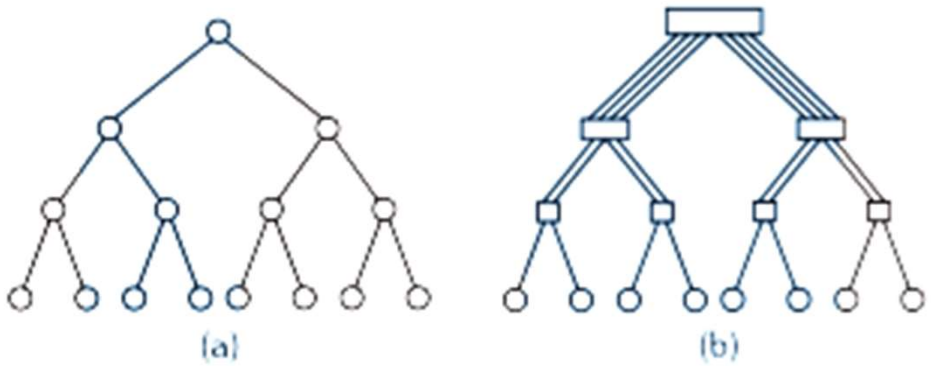


Para mejorar la conectividad de la red, una de las formas habitualmente empleadas es la **jerarquización de la red**. Las topologías jerarquizadas de redes se construyen sobre el principio del bus



Jerarquización de la red.

En ellas se reagrupan los enlaces para disminuir el número de conexiones físicas. Ejemplos de estos tipos de redes son las jerarquías de buses, las estrellas de varios niveles, las pirámides o los árboles.



La jerarquización tiene la ventaja de disminuir el número de elementos conectados para cada subred de la jerarquía. Gracias a ello, los problemas de contención en la red quedan reducidos.

Las dificultades aparecen cuando se cruzan un gran número de mensajes de un nivel a otro de la jerarquía, creándose problemas de cuello de botella que pueden tener su importancia. Además, a nivel de protocolos de comunicación estas topologías son más difícil de gestionar.

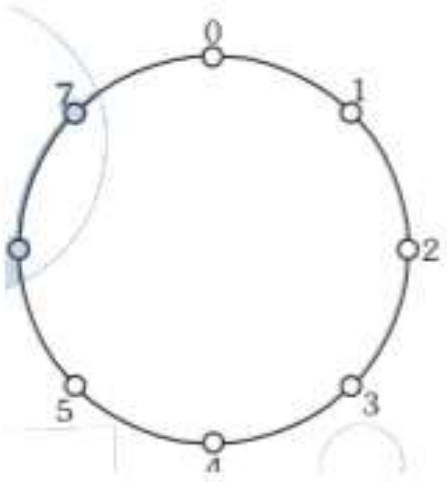
CONECTIVIDAD - CRITERIOS

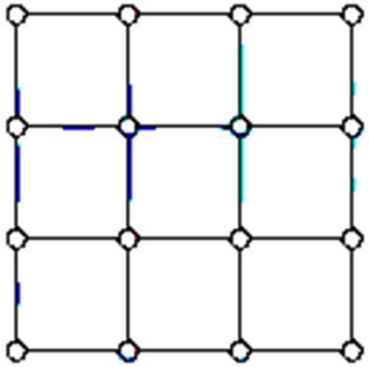
La interrelación de los criterios **dinámico/estático**, completamente **conectado/parcialmente conectado** y **jerarquizado/no jerarquizado** define distintas clases de redes. De hecho, las máquinas encontradas en el mercado utilizan redes derivadas de algunas de estas categorías.

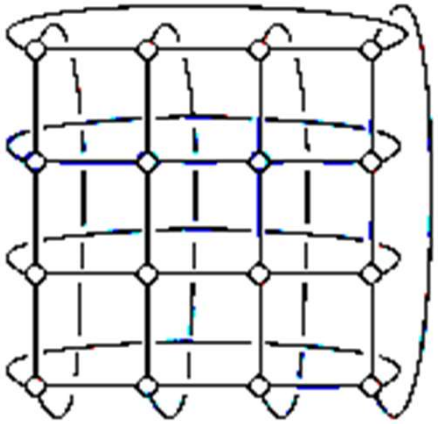
Otro de los aspectos que hay que tener en cuenta al hablar de la conectividad de las redes es si la red es ***conexa* o no**.

Para redes parcialmente conectadas es muy importante asegurar que sean *conexas*, es decir, que entre dos procesadores cualesquiera siempre exista un camino para enviar un mensaje de uno al otro. Esta propiedad es muy importante tenerla en cuenta en el caso de fallas en la red. Si una red tiene fallas y la red sigue siendo conexa, esa topología es buena y se denomina que es *tolerante a fallos*.

Un ejemplo es la malla, que en el caso de que se fallen varios enlaces dicha topología ofrece caminos alternativos para poder encaminar los mensajes. Por contra, el anillo en este sentido es una mala topología, pues sólo con que un enlace tenga un fallo la red ya no es conexa y se parte en dos redes disjuntas.







MODOS DE ENCAMINAMIENTO

El **encaminamiento** es un mecanismo bien realizado por sofware o por hardware que dirige los mensajes entre el procesador origen y el procesador destino, durante la comunicación en una red parcialmente conectada de tipo estático o dinámico.

El modo de encaminamiento comprende dos aspectos esenciales: lo que se denomina el **algoritmo de encaminamiento** y lo que recibe el nombre del **control de flujo** en el encaminamiento.

El algoritmo de encaminamiento efectúa la **elección de caminos** cuando existen varios posibles, y **gestiona los conflictos** que puedan surgir entre los mensajes que quieren tomar el mismo camino.

Normalmente se busca un algoritmo de encaminamiento que sea **óptimo**, es decir, que conduzca a los mensajes por el **camino más corto**. El algoritmo de encaminamiento depende de la topología de la red.

El **control de flujo** describe el **modo físico de propagación de la información**. Hay diversas técnicas para realizar esto, tomadas al principio de las técnicas que se usan en redes locales; si bien se hay desarrollos específicos para redes en máquinas paralelas.

El principal problema que se encuentra en el modo de encaminamiento en una red es el **bloqueo,** lo que puede llegar a inutilizar la red. Es necesario elegir modos de encaminamiento y topologías que resuelvan estos conflictos antes de ejecutar una aplicación.

Las **redes estáticas** se usan habitualmente en los multicomputadores. En la actualidad, y gracias a usar un **encaminamiento de mensajes segmentado**, se utiliza con más frecuencia la malla o el toro.

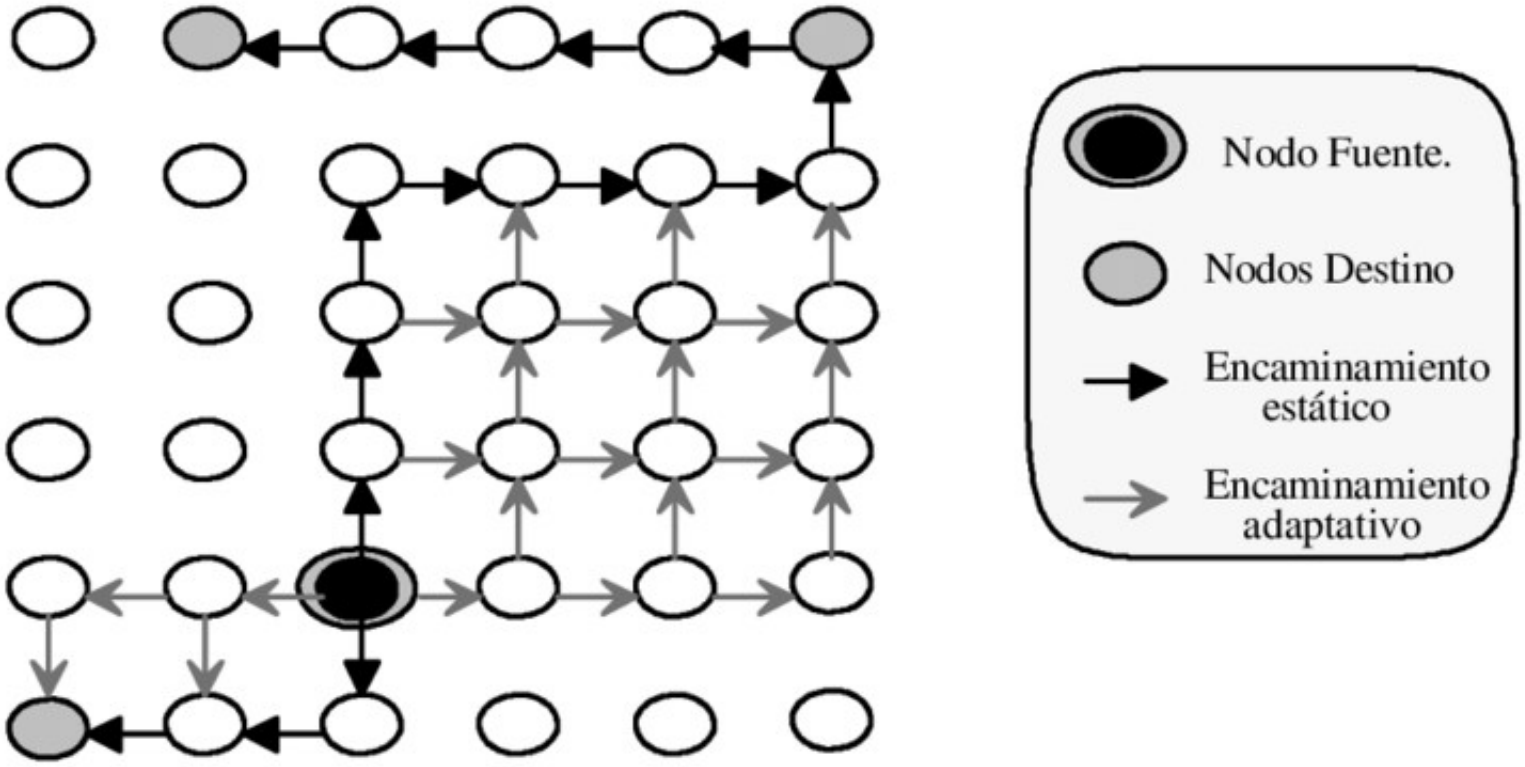
El modo de encaminamiento

El modo de encaminamiento de datos debe satisfacer un número de requerimientos, siendo los más importantes los siguientes:

- El protocolo de encaminamiento debe estar libre de bloqueo (*deadlock*).
- Ningún paquete de datos puede ser retrasado infinitamente en la red.
- Un mensaje siempre debe tomar el camino más corto para llegar a su destino.
- Dicho mecanismo se debe adaptar a las condiciones de tráfico de la red y debe explotar al máximo su ancho de banda.
- El protocolo de encaminamiento debe conseguir la más baja latencia para la red y el más alto rendimiento.
- Se debe asegurar que no haya adelantamientos en los mensajes. Es decir, dos mensajes enviados por un nodo a otro no pueden llegar en orden diferente a como fueron enviados.

Al hablar del modo de encaminamiento, hay que detallar por una parte el algoritmo de encaminamiento y por otra el control del flujo de mensajes.

Con respecto a los **algoritmos de encaminamiento**, hay dos grandes grupos: los estáticos o deterministas y los dinámicos o adaptativos.



Por ***estático*** se entiende un algoritmo que tiene un comportamiento definido y constante a lo largo del tiempo, habiéndose desarrollado varios modelos que garantizan la ausencia de bloqueos. Son algoritmos más sencillos de implementar pero tienen el problema de provocar en ocasiones puntos de congestión en la red.

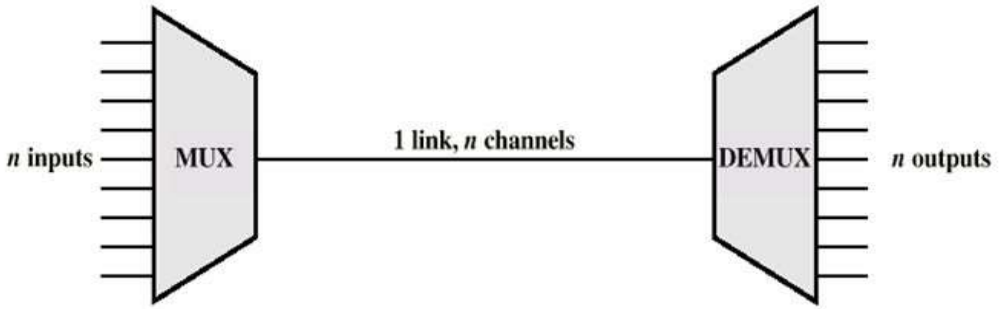
Los algoritmos ***adaptativos*** son aquellos que cambian su comportamiento a lo largo del tiempo en función de diversos parámetros de la red, mejorando por tanto la velocidad y el rendimiento en el envío de mensajes. Lógicamente, la **complejidad de estos algoritmos es mucho mayor y requieren una circuitería adicional**. Los principales objetivos del encaminamiento dinámico son reducir las colisiones entre mensajes y el tiempo total de transferencia e incrementar la tolerancia a fallos.

Una solución para mejorar las prestaciones de estos algoritmos es basa en el uso de **canales virtuales**.

Un *canal virtual* es un canal de comunicación que se define entre dos procesadores de la red y que está soportado por un canal físico, pero sin coincidir con él, ya que por cada canal físico de comunicación se suelen definir varios canales virtuales. Los **canales virtuales son multiplexados** en el canal físico, por lo que comparten el ancho de banda de dicho canal.

OBJETIVO DE LA MULTIPLEXACIÓN

- Es compartir la capacidad de transmisión de datos sobre un mismo enlace para aumentar la eficiencia (sobre todo en líneas de grandes distancias).
- Minimizar la cantidad de líneas físicas requeridas y maximizar el uso del ancho de banda de los medios



Las ventajas que se obtienen con los canales virtuales derivan del hecho de que cuando un mensaje está retenido en la red, no colapsa el canal físico por el que está viajando, dejándolo para aquellos mensajes que vayan por los canales virtuales que están compartiendo el mismo canal físico.

Esta solución es sobre todo interesante para modos de encaminamiento que poseen un control de flujo del tipo *wormhole routing*. Cada canal virtual tiene su propio *buffer* a nivel de flit *(flow control units)*, su propio control y su propio camino para los datos.

La principal dificultad en el diseño de los algoritmos de encaminamiento es que se debe asegurar que éste sea **libre de bloqueo**.
Un bloqueo en una red de interconexión ocurre cuando **ningún mensaje puede avanzar hacia su destino debido a que todas las colas de mensajes de los nodos del sistema están llenas**. El tamaño de las colas tiene una gran influencia en la probabilidad de alcanzar una configuración que esté bloqueada.

De todas formas, el modo más práctico para evitar el bloqueo es desarrollar algoritmos de encaminamiento libres de bloqueo. En estos últimos años, ésta es una de las áreas de más estudio e investigación. Varios desarrollos libres de bloqueo existen en redes con encaminamiento del tipo almacenamiento y reenvío.

Junto a los algoritmos de encaminamiento hay otro importante aspecto a tener en cuenta: los mecanismos de control de flujo. Se han desarrollado una diversidad de mecanismos para, una vez determinado el camino que deben seguir los mensajes, manejar el flujo de mensajes de un nodo origen a uno destino. **Los principales mecanismos de control de flujo son los siguientes:**

La conmutación de circuitos

En esta técnica -la primera que se desarrolló- los nodos que se comunican en un instante dado se unen por un camino físico que no se modifica mientras dura la comunicación. Esta técnica es similar a lo que ocurre en la red telefónica.

Fundamentalmente se emplea en las redes dinámicas de los multiprocesadores, no teniendo mucho interés en el caso de los multicomputadores.

La conmutación de mensajes

También se denomina de almacenamiento y reenvío. Esta técnica almacena cada mensaje completamente en un nodo y entonces lo transmite al siguiente nodo. El mensaje construye su camino paso a paso en la red. Esta técnica es similar a la empleada en las redes de datos. **Cada nodo debe disponer de un *buffer*** donde retener los mensajes que le llegan, perdiendo una cantidad de tiempo apreciable en almacenar y recuperar de la memoria estos mensajes. Esta técnica provoca una **alta latencia** en la red de interconexión, estando influida por la distancia que hay entre los nodos que comunican. La latencia de la red es (L/B)D, donde L es la longitud del mensaje, B es el ancho de banda del canal y D es la longitud del camino entre los nodos fuente y destino. Otra desventaja de este mecanismo de control es que aumenta el tamaño de la memoria local requerida en cada procesador. Esta técnica fue adoptada por la mayoría de los multicomputadores comerciales de la primera generación, tales como iPSC-1, Ncube-1, Ametek 14 o FPS-T.

Wormhole o encaminamiento segmentado con espera

En esta técnica cada mensaje se descompone en pequeños fragmentos denominados unidades de control de flujo o *flits*. El primer flit es la cabecera y contiene la dirección de destino del mensaje. El último flit es la cola y los flits intermedios contienen solamente datos.

Cuando el flit de cabecera llega a un nodo, se encamina inmediatamente. A medida que el flit avanza, va reservando la ruta y los demás lo siguen. Los canales reservados por la cabecera serán liberados cuando pase el flit de cola. Si la cabecera no encuentra un canal libre para poder continuar, se detiene temporalmente hasta que se libere.

El mecanismo de control de flujo detiene los restantes flits del mensaje. Conforme los *flits* van avanzando, el mensaje está partido a lo largo de los canales entre el nodo fuente y el nodo destino. Es posible que el *flit* de cabeza haya llegado al nodo destino mientras parte del mensaje aún no ha salido del nodo fuente. **Debido a que los *flits* no contienen información acerca del destino (excepto el de cabeza), los diversos *flits* de un mensaje no se pueden mezclar con los *flits* de otro mensaje. Por tanto, cuando la cabeza de un mensaje es bloqueada, todos los *flits* de ese mensaje paran de avanzar y bloquean a su vez cualquier otro mensaje que necesite transitar por los canales que ellos ocupan.**

Sus características más importantes son:

- No se almacenan los mensajes en los nodos intermedios.
- El encaminamiento es distribuido.
- Los algoritmos de encaminamiento son implementados por hardware.
- La posibilidad de bloqueo es más acentuada.
- El tiempo total de transferencia es mucho más reducido.
- El mensaje es descompuesto en bloques (cabecera, bloques de información y cola).
- Se encamina la cabecera, sin esperar al resto del mensaje.
- Los enlaces se reservan hasta que pasa la cola.

En este caso, la latencia de la red es $(L_f/B)D + L/B$, donde L_f es la longitud de cada flit, B es el ancho de banda del canal, D es la longitud del camino y L es la longitud del mensaje. Como $L_f << L$, la latencia de la red no se ve prácticamente afectada por la distancia que tiene que recorrer el mensaje.

El primer multicomputador comercial que adoptó esta técnica de control de flujo fue el Ametek 2010, el cual usaba una topología de malla 2D. El Ncube-2 también usa este encaminamiento con una topología hipercubo.

Virtual cut-through

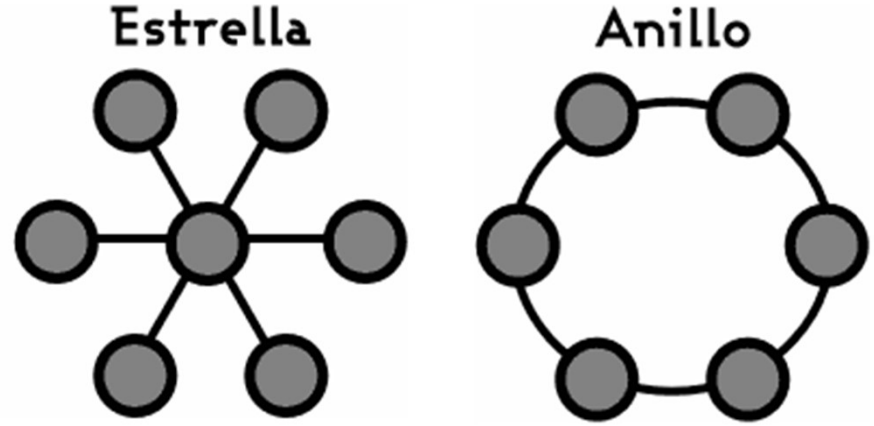
Es una técnica similar a la anterior. Se diferencia en que ésta almacena el mensaje en un *buffer* cuando se bloquea, quitándolo de la red y permitiendo el paso de otro mensaje. Esto alivia el cuello de botella del algoritmo anterior y mejora el rendimiento de la red, a costa de una circuitería más compleja.

Double buffering

Este método intenta evitar el problema del *deadlock* de la red. Para ello lo primero que hace el nodo fuente es almacenar en un *buffer* el mensaje que va a ser enviado. Después intenta establecer una conexión con el nodo destino por formar un camino *rígido* a través de los nodos intermedios. Si se encuentra un bloqueo o un fallo en alguno de estos nodos intermedios, se vuelve de nuevo al nodo origen, intentando volver a formar dicho camino después de un retraso aleatorio. Una vez que se ha conseguido la conexión rígida entre el nodo fuente y el nodo destino, el mensaje entero es transferido, siendo almacenado en un *buffer* en el nodo destino antes de ser despachado. De ahí el nombre de esta técnica.

Para el caso de máquinas desarrolladas para problemas muy sencillos o con pocos nodos, también se ha utilizado a veces la **topología de la estrella y el anillo**. A partir de cuatro o cinco nodos, esta topología estrella ya no es práctica.

El algoritmo de encaminamiento es sencillo, pues todos los mensajes se envían al nodo central y este ya los encamina al nodo destino. En el caso del anillo, para encaminar un mensaje de un nodo a otro, una vez determinado por qué enlace debe enviarse, únicamente hay que transmitirlo hasta que llegue al nodo destino. Esta topología es adecuada para problemas intensivos en cálculo.



Las **redes dinámicas** se han utilizado esencialmente en los multiprocesadores de memoria compartida: la red dinámica soporta, por consiguiente, la carga de unir los N procesadores a los M bancos de la memoria central.

Como en las redes estáticas, podemos realizar redes dinámicas a base **de enlaces punto- a-punto o de bus**; nuevamente se presenta un problema en la **relación conexión/contención**

Desde el momento en que el número de procesadores sobrepase algunas decenas, debemos elegir soluciones que se sitúen entre estos dos extremos. Por ejemplo, podemos construir un crossbar, utilizando **conmutadores**.

Con un *crossbar,* el número de enlaces crece solamente en 2N, y el control de la red es extremadamente simple. Sin embargo, el número de conmutadores crece a su alrededor de manera cuadrática. Los *crossbar* son también muy interesantes, pues actualmente el costo de un conmutador realizado en VLSI es más bajo que el de un enlace.

Estas tienen por objetivo acercar, tanto como sea posible, los rendimientos de la red *crossbar,* haciendo uso de un número menor de conmutadores, pero, eso sí, con más tiempo para su recorrido.

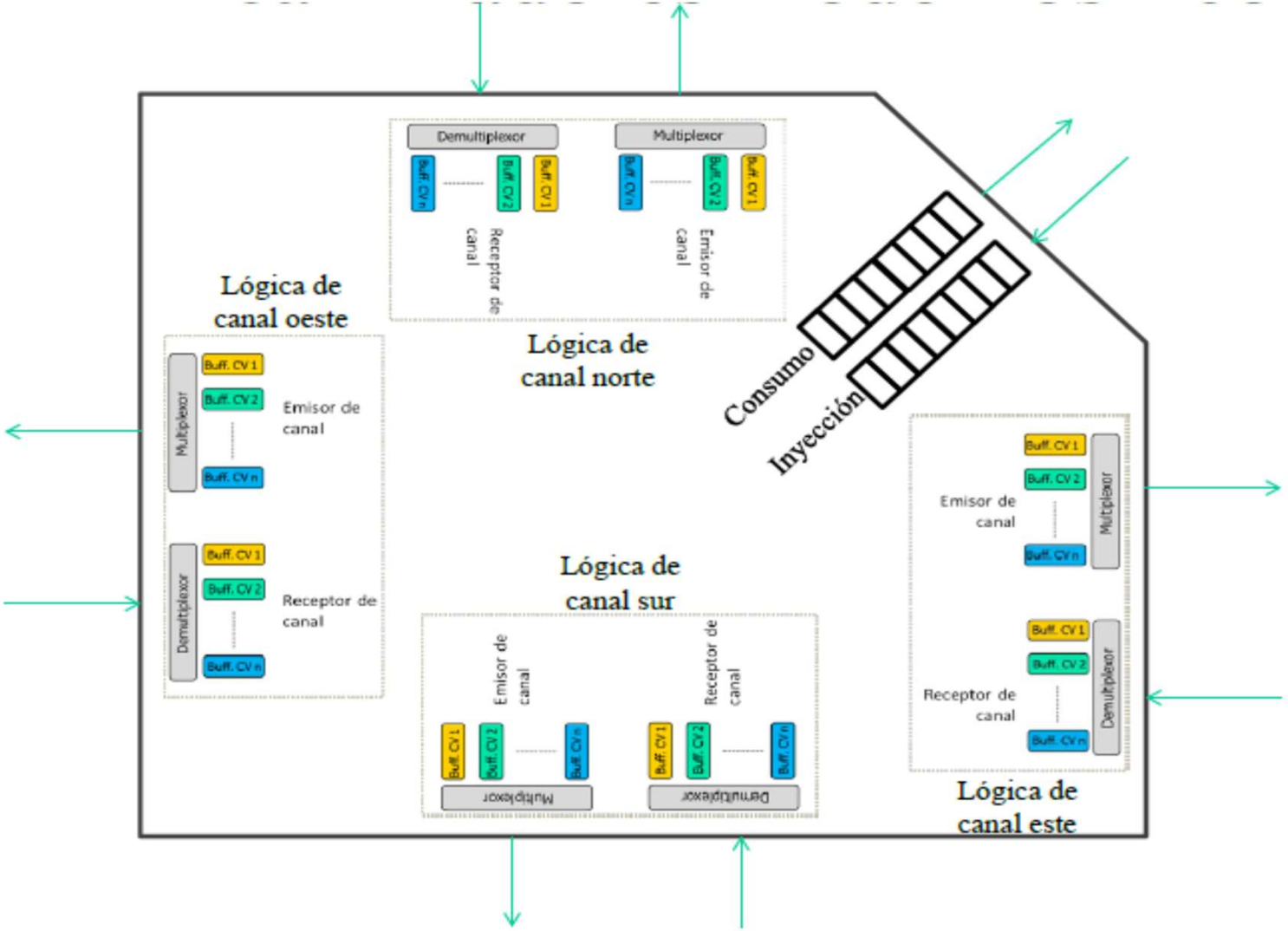
El encaminamiento de mensajes implica el **posicionamiento** de los conmutadores en cada etapa.

Redes multietapa: Para las arquitecturas MIMD, donde las comunicaciones son asíncronas y dinámicas, se utilizan redes que tienen la propiedad del camino único: entre un origen y un destino, existe un solo camino. De esta propiedad se deduce, en general, una cierta simplicidad en el control. Por contra, estas redes tienen facultad de bloqueo

En una red multietapa de tipo Omega, es posible llegar a varios destinos a partir de un solo origen. Esto permite realizar difusiones en la red. Estas redes son, por el contrario, sensibles a los fallos ya que, por ejemplo, si un hilo se corta no serán posibles muchas de las rutas.

RUTEADORES MODERNOS

- Disponen de 3 elementos principales:
  - Colas de almacenamiento:
    - De inyección: permiten al nodo local colocar paquetes en la red.
    - De tránsito: almacenan paquetes que no tienen origen ni destino en el nodo local.
    - De consumo: entregan paquetes al nodo local.
  - Interconexión (crossbar limitado).
  - Árbitro: toma las sobre encaminamiento y resolución de conflictos.

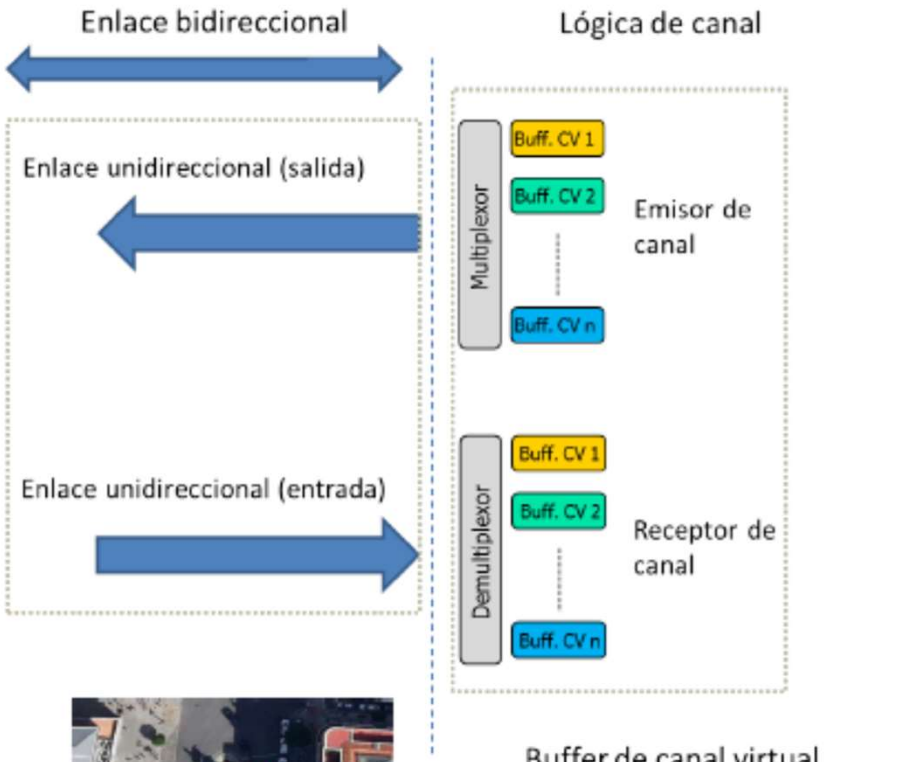

Demultiplexor
Multiplexor
Buff. CV n
Buff. CV 2
Buff. CV 1
Receptor de canal
Emisor de canal
Lógica de canal norte
Lógica de canal oeste
Multiplexor
Demultiplexor
Emisor de canal
Receptor de canal
Consumo
Inyección
Lógica de canal sur
Lógica de canal este

RUTEADORES MODERNOS - COLAS

Su espacio de almacenamiento se establece en función del mecanismo de control de flujo

Se trata por defecto de colas FIFO -> problemas de bloqueo de cabeza de línea (HLB). Si la unidad de cabeza no puede ser atendida, las siguientes quedan bloqueadas aunque no entren en conflicto.

Hay tantas colas de tránsito como canales virtuales.



HLB

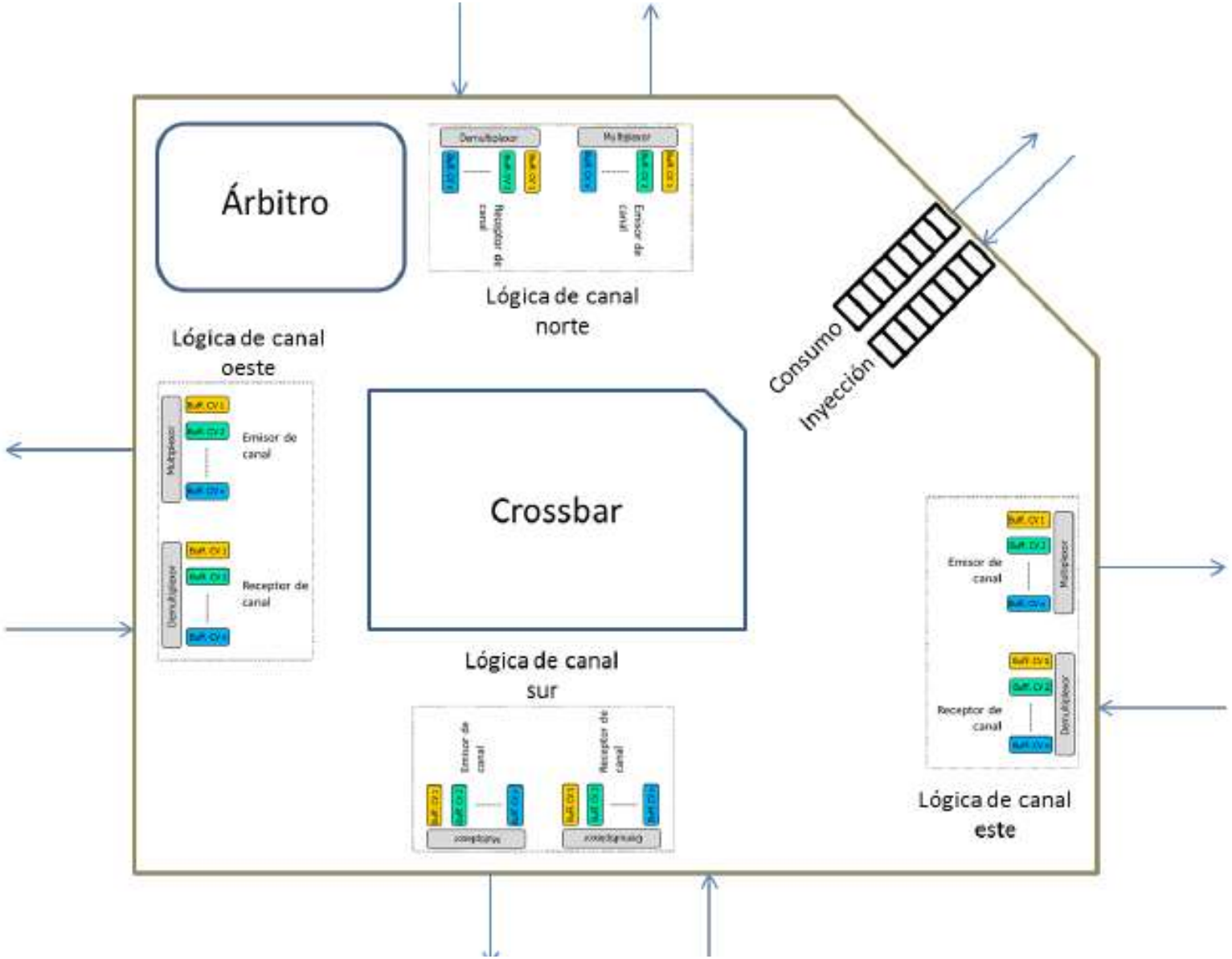
El coche rojo podría girar pero está bloqueado por el blanco.
El semáforo impide al blanco avanzar recto.

Google Earth

RUTEADORES MODERNOS

arbitraje

- Selección: criterio empleado por el árbitro para seleccionar el puerto de salida de entre los que permitan avanzar a un paquete cumpliendo con su protocolo de encaminamiento. Si el encaminamiento es estático, no existe política de selección posible.
- Arbitraje: criterio empleado por el árbitro para asignar un determinado puerto de salida entre varias colas de entrada que compiten por él.
- Objetivo: combinar la eficiencia con la ausencia de problemas fatales; típicamente la inanición.


Árbitro
Lógica de canal norte
Lógica de canal oeste
Emisor de canal
Receptor de canal
Crossbar
Consumo
Inyección
Lógica de canal sur
Lógica de canal este

RUTEADORES MODERNOS

arbitraje – políticas de selección

- Aleatoria: se asigna una salida al azar de entre las posibles.
- Cola más corta: se elige el puerto de salida con más espacio en la cola de entrada del nodo destino.
- Smart: se realizan sucesivos intentos. Tiene sentido si el encaminamiento es adaptativo. Primero se intenta continuar por el mismo canal de entrada; si no está disponible, se solicita cambiar de dirección.

RUTEADORES MODERNOS

arbitraje – políticas de arbitraje

- Aleatoria: se da servicio a uno de los competidores seleccionado al azar.
- Roundrobin: se sigue una lista ordenada para dar servicio a todos.
- Más antiguo: se da servicio al que más tiempo lleve en espera.
- Cola más larga: se da servicio al canal que mantenga la cola más larga.

Infiniband

Infiniband también soporta doble e incluso cuádruples tasas de transferencia de datos, llegando a ofrecer 5 Gbps y 10 Gbps respectivamente.

Los enlaces pueden añadirse en grupos de 4 o 12, llamados 4X o 12X. Un enlace 12X a cuádruple modo tiene un caudal bruto de 120 Gbps, y 96 Gbps de caudal eficaz. Actualmente, la mayoría de los sistemas usan una configuración 4X con modo simple, aunque los primeros productos soportando doble ritmo ya están penetrando en el mercado. Los sistemas más grandes, con enlaces 12X se usan típicamente en lugares con gran exigencia de ancho de banda, como interconexión de redes.
La latencia teórica de estos sistemas es de unos 160ns. Las reales están en torno a los 6 µs, dependiendo bastante del software y el firmware.

BW de Infiniband, bruto / eficaz			
	Simple DR	Doble DR	Quadruple DR
1X	2,5 / 2 Gbps	5 / 4 Gbps	10 / 8 Gbps
4X	10 / 8 Gbps	20 / 16 Gbps	40 / 32 Gbps
12X	30 / 24 Gbps	60 / 48 Gbps	120 / 96 Gbps

Infiniband usa una topología conmutada de forma que varios dispositivos pueden compartir la red al mismo tiempo.

Los datos se transmiten en paquetes de hasta 4 kB que se agrupan para formar mensajes. Un mensaje puede ser una operación de acceso directo a memoria de lectura o escritura sobre un nodo remoto (RDMA), un envío o recepción por el canal.