

Diseño y Desarrollo de Sistemas de Tiempo Real

Alfons Crespo



Tema 3-4 Planificación en sistema multi-core

Alfons Crespo



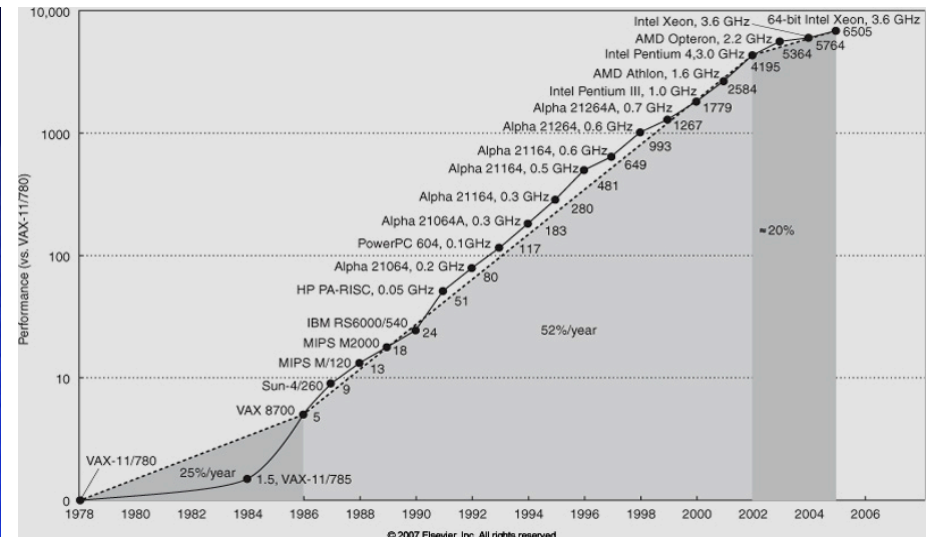
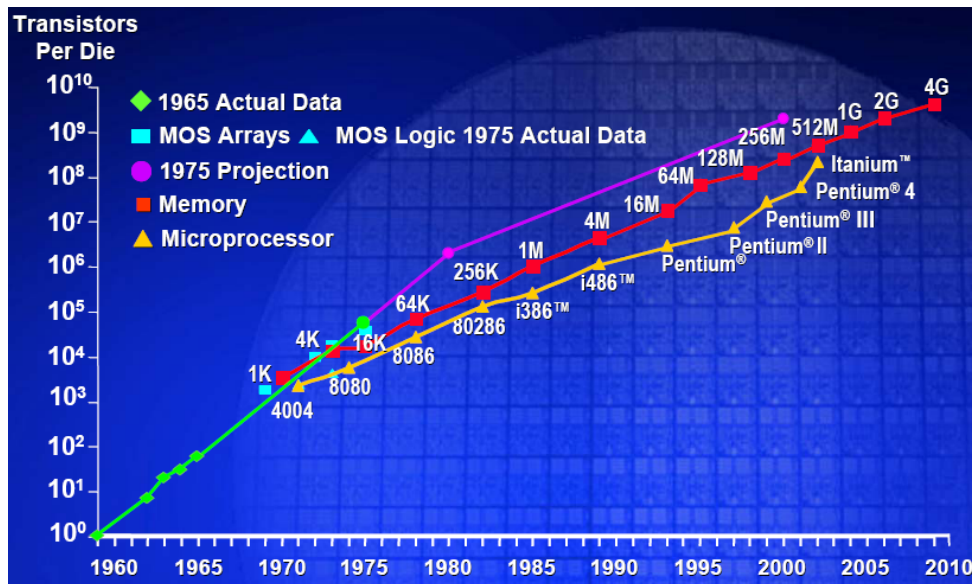
Objetivos

- Ofrecer una visión de la planificación en sistemas multicore
- Analizar las políticas de planificación en SMC
- Evaluar las políticas de planificación

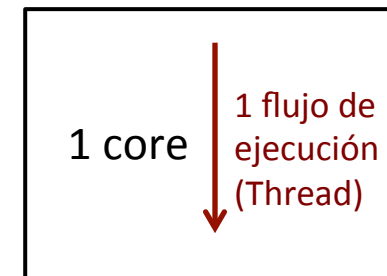
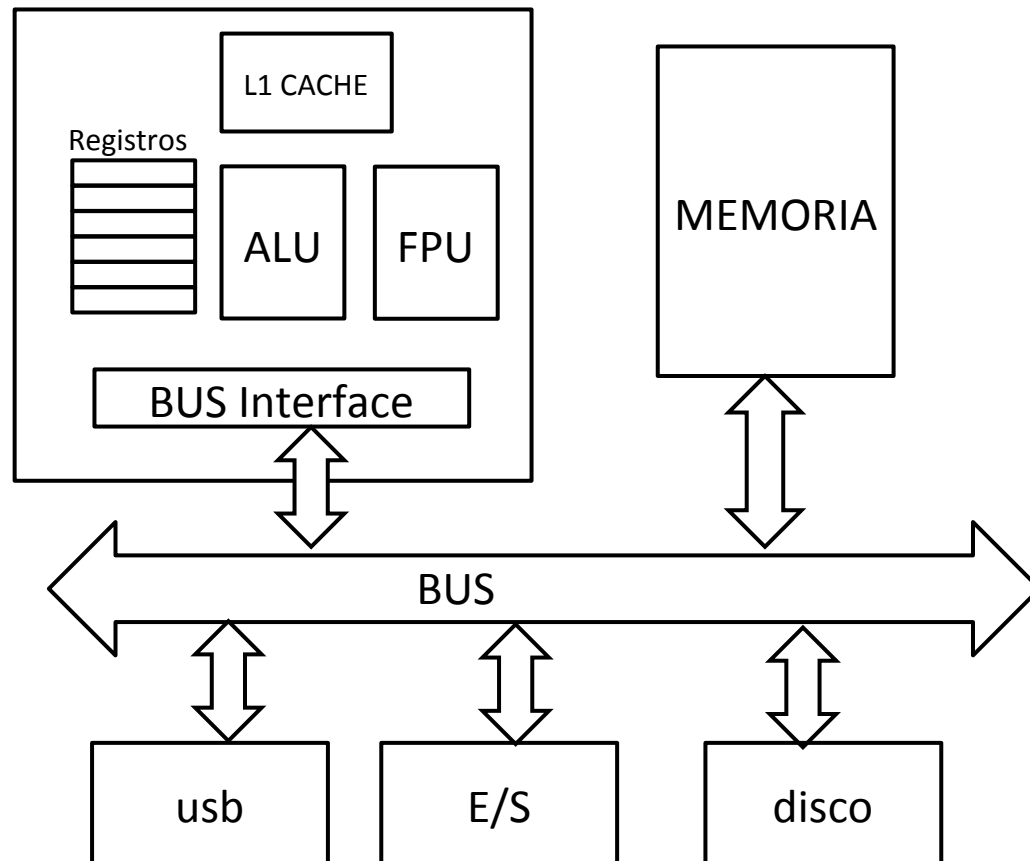
- **Introducción**
- **Tipos de planificadores para Sistemas multi procesador**
 - **Planificación Global**
 - **Planificación Local**
- **Técnicas de asignación de la carga**
 - **Heurísticas: First Fit, Best-Fit, Worst-Fit**

Sistema multicore

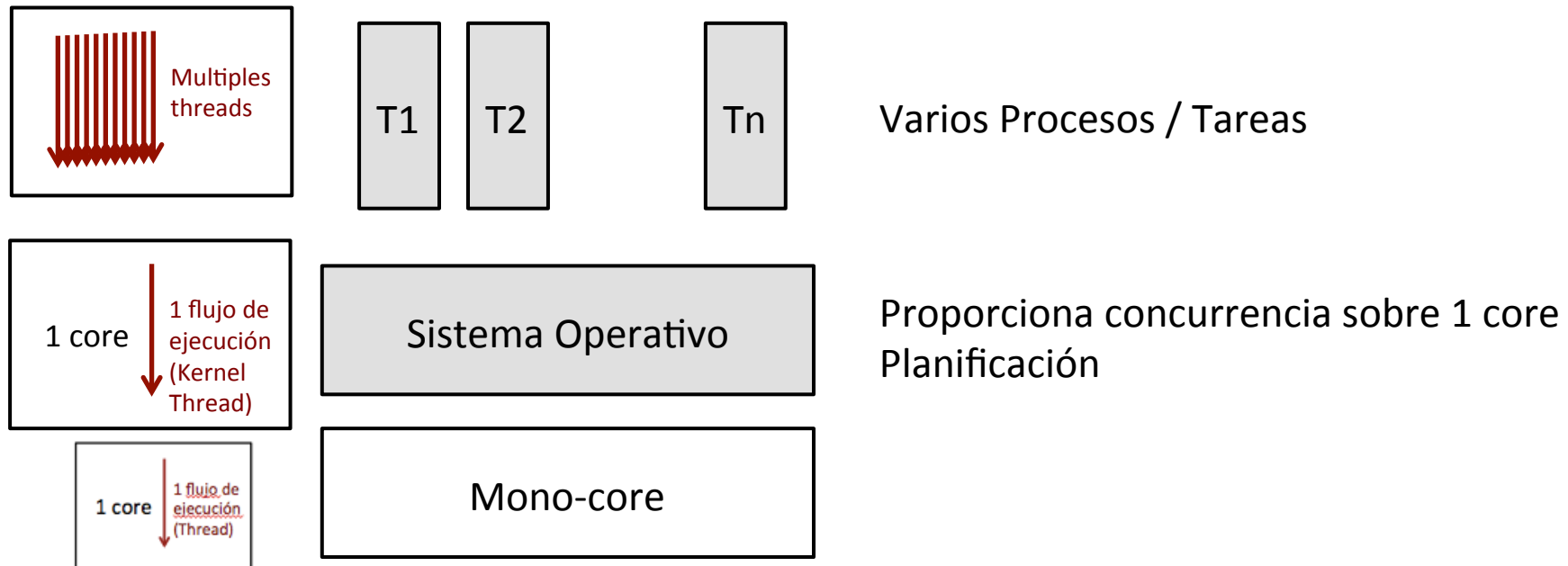
- Los sistemas multicore nacen de la necesidad de:
 - Ley de Moore (limitación)
 - Incrementar la potencia de cómputo
 - Reducir el calentamiento



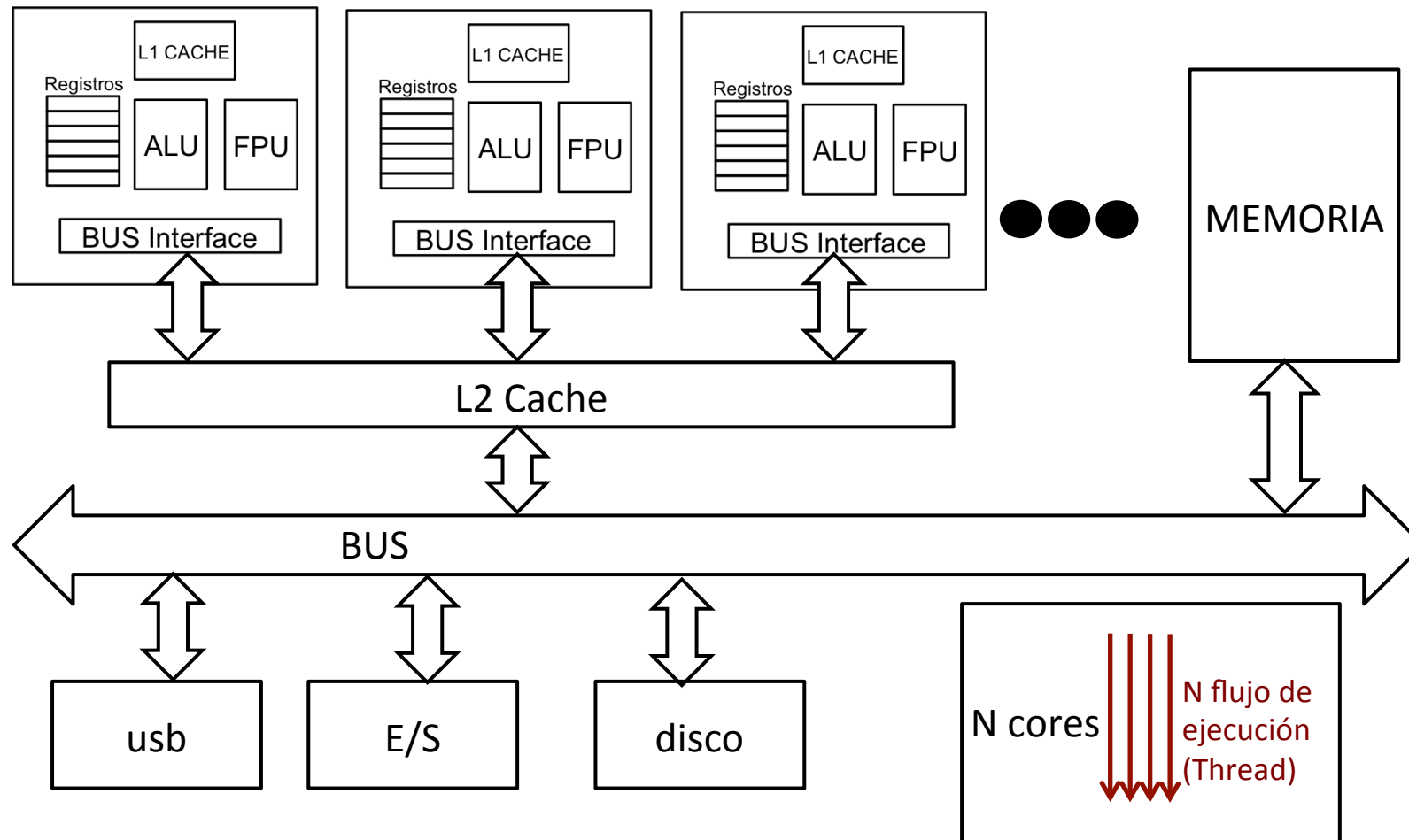
- Sistemas Mono-core



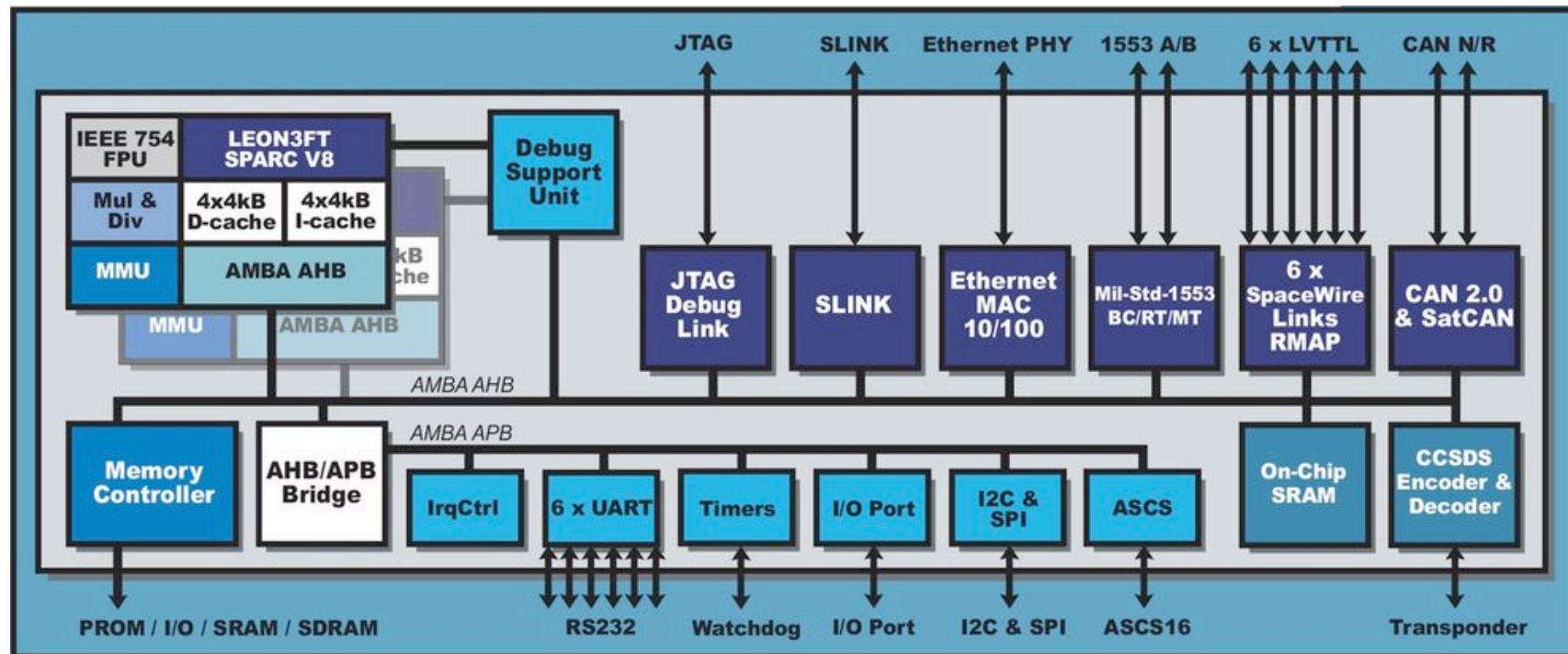
Sistemas Monocore



- Sistemas Multi-core

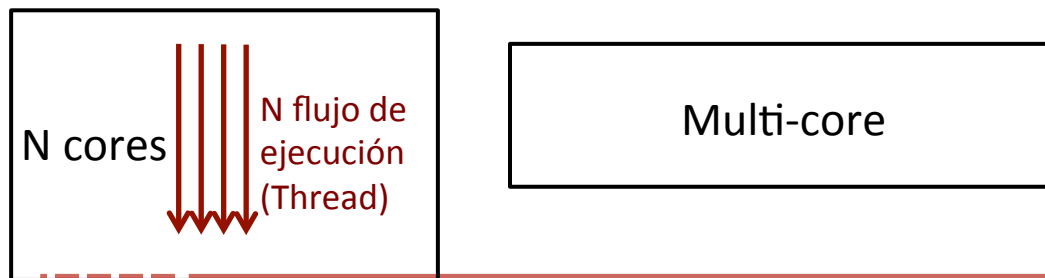


LEON 3 Bi-core



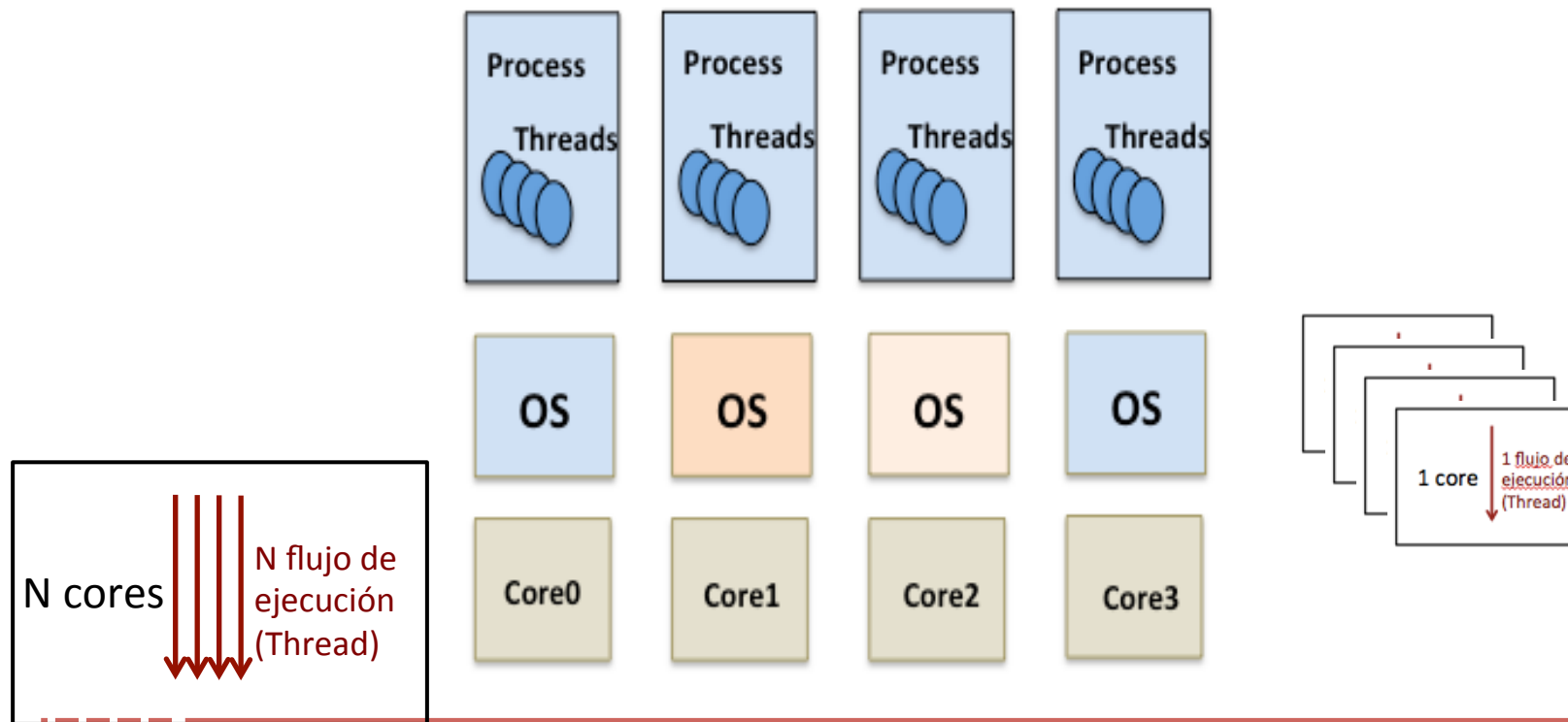
Sistemas Multi-core

- Decisiones de diseño del S.O.
 - 1 sólo S.O.
 - Varios S.O.
- Procesamiento simétrico o asimétrico



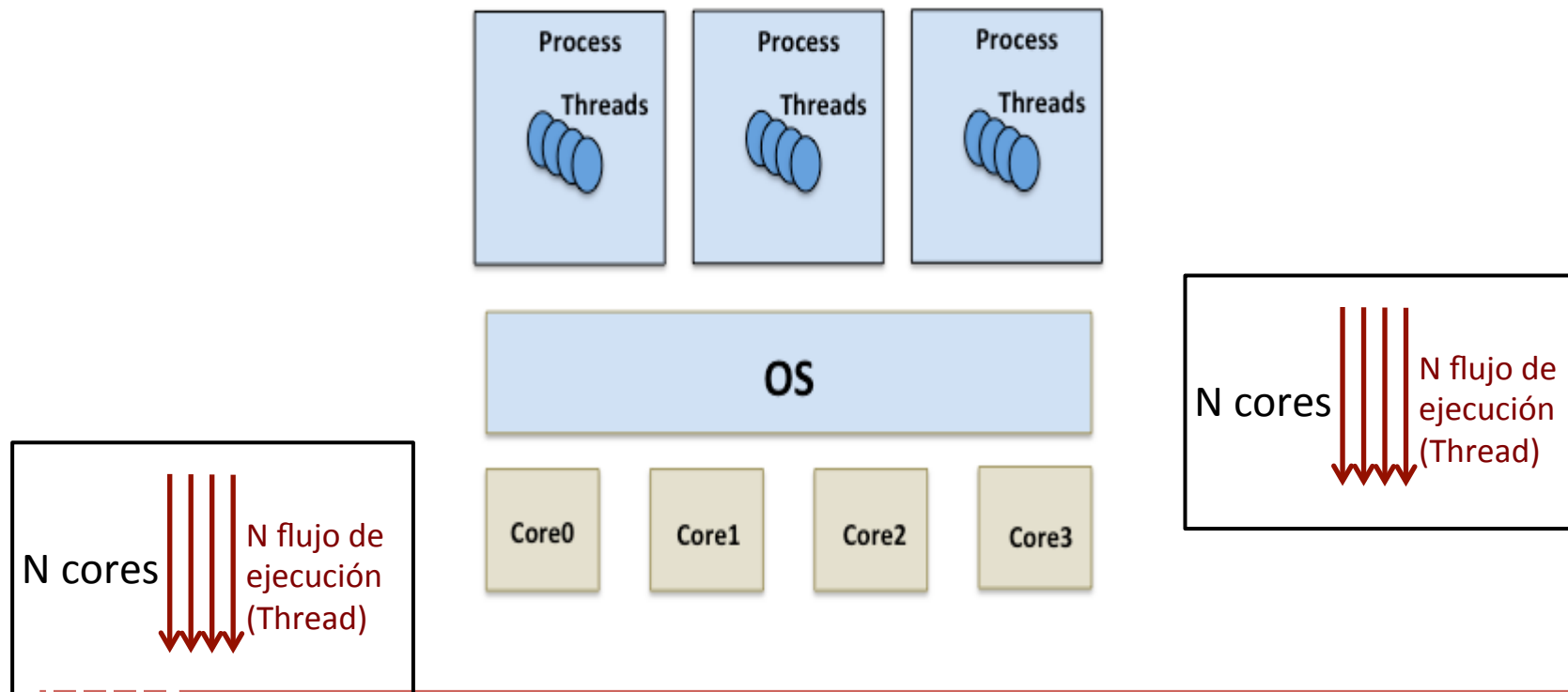
Sistemas Multi-core

- Asymmetric Multiprocessing (AMP)
 - 1 S.O. por core
 - S.O. es monocre



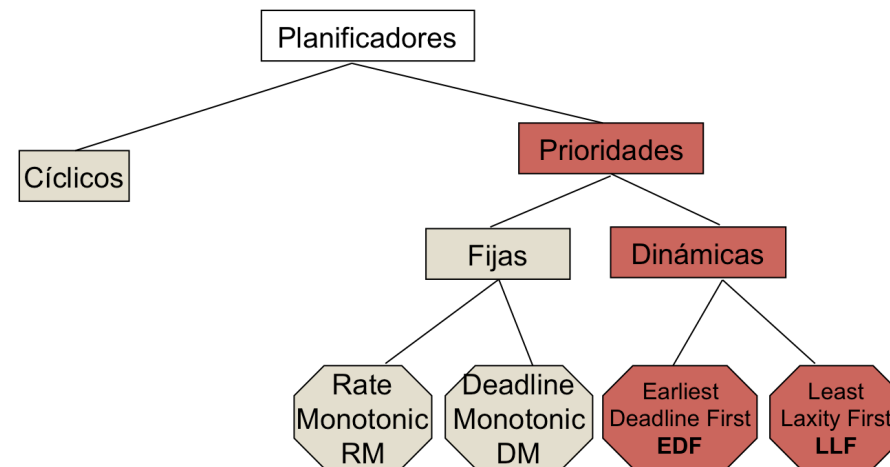
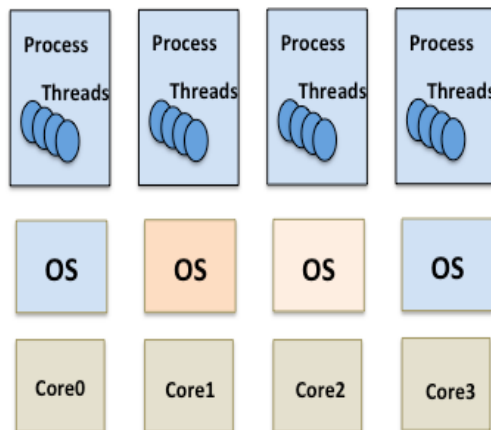
Sistemas Multi-core

- Symmetric Multiprocessing (SMP)
 - 1 S.O. para todos los cores
 - S.O. es multicore



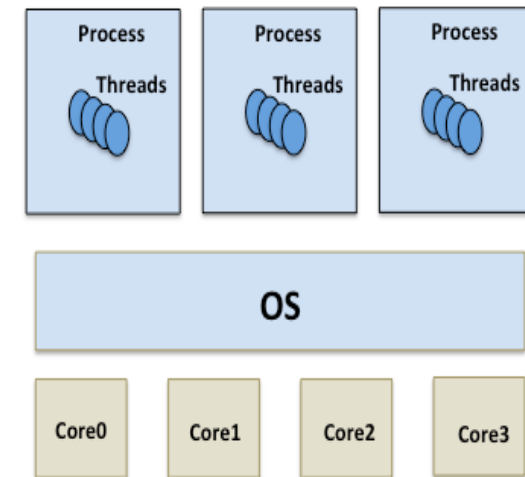
Arquitectura (SW) AMP

- Cada S.O. usa un procesador
- Visión mono-procesador



Arquitectura (SW) SMP

- S.O. es multiprocesador
- Gestiona cada uno de los cores
- Tantos flujos de control como cores
- Puede ejecutar Procesos/Tareas en cores al mismo tiempo (paralelismo)
- Se requieren políticas de planificación multi-core. Puede ser:
 - Global
 - Particionado (cada core)



Taxonomía de algoritmos de planificación

- Planificación en multicore se puede ver como un modo de resolver 2 problemas:
 - **Ubicación** (Allocation): dónde ejecutar una tarea
 - **Prioridad**: cuándo y en qué orden se ejecutan las tareas o las activaciones de las tareas
- Ubicación:
 - **Sin migración**: todas las activaciones de una tarea en un mismo core (o todas las tareas de una partición)
 - **Migración** a nivel de tarea: cada activación se puede ejecutar en un core (o cada tarea de una partición)
 - **Activación**: Una activación puede migrar

Taxonomía de algoritmos de planificación

- Planificación en multicore se puede ver como un modo de resolver 2 problemas:
 - **Ubicación** (Allocation): dónde ejecutar una tarea
 - **Prioridad**: cuándo y en qué orden se ejecutan las tareas o las activaciones de las tareas
- Prioridad:
 - **Prioridad fija de tarea**: una tarea tiene una prioridad (FPS)
 - **Prioridad fija de activación**: cada activación puede tener una prioridad (EDF)
 - **Prioridad dinámica**: una misma activación puede tener varias prioridades (LLF)

Taxonimia de algoritmos de planificación

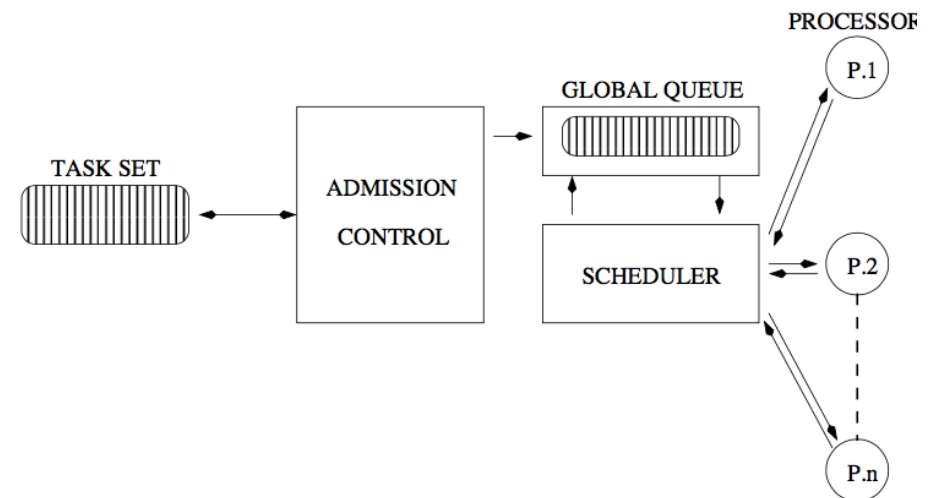
- **Planificación global:**
 - Hay migración
 - Cualquier tarea se puede ejecutar en cualquier core
- **Planificación Local** (se llama particionada, pero evitamos el nombre para no crear confusión con Particiones)
 - No hay migración
 - Un subconjunto de tareas se asignan a un core y allí se ejecutan

Taxonomía de algoritmos de planificación

- Además, los algoritmos pueden ser:
 - **Expulsivos**
 - **No expulsivos**
 - **Parcialmente expulsivos (coordinados)**: expulsiones en determinados puntos del código

Planificación global

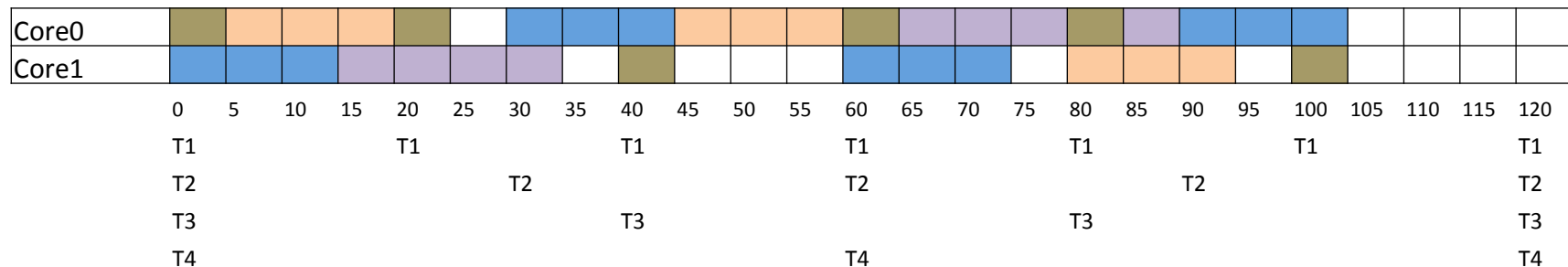
- Una única cola de procesos/tareas a ejecutar (READY)
- La política de planificación es única
- Varios cores (servidores)
- Políticas:
 - Prioridades fijas estáticas
 - Prioridades fijas dinámicas



Planificación global

- 2 cores
- Expulsivo, con Migración y prioridades fijas

	C	P	D	U	Unid	Activ	Usa
1	5	20	20	0,25	1	6	6
2	15	30	30	0,50	3	4	12
3	15	40	40	0,38	3	3	9
4	20	60	60	0,33	4	2	8
						Usadas	35
H		120		1,46		Total	48



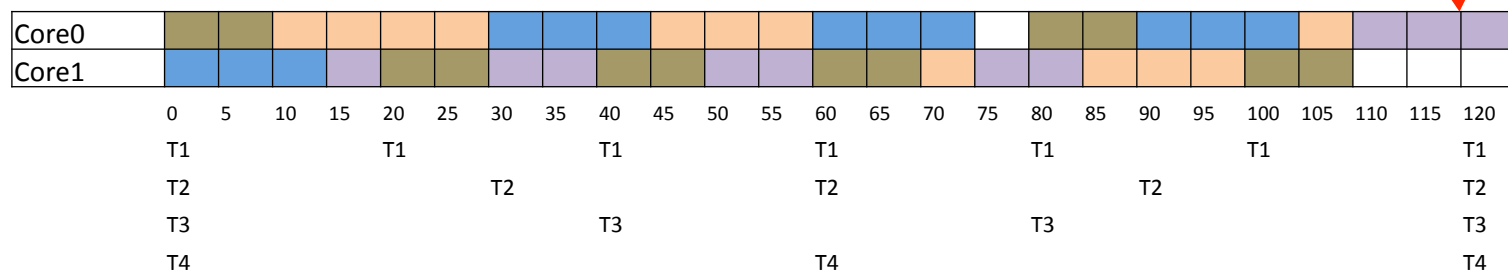
[illegible]

Planificación global

- 2 cores
- Expulsivo, con Migración y prioridades fijas

	C	P	D	U	Unid	Activ	Usa
1	10	20	20	0,50	2	6	12
2	15	30	30	0,50	3	4	12
3	20	40	40	0,50	4	3	12
4	25	60	60	0,42	5	2	10
						Usadas	46
H		120		1,92		Total	48

RM



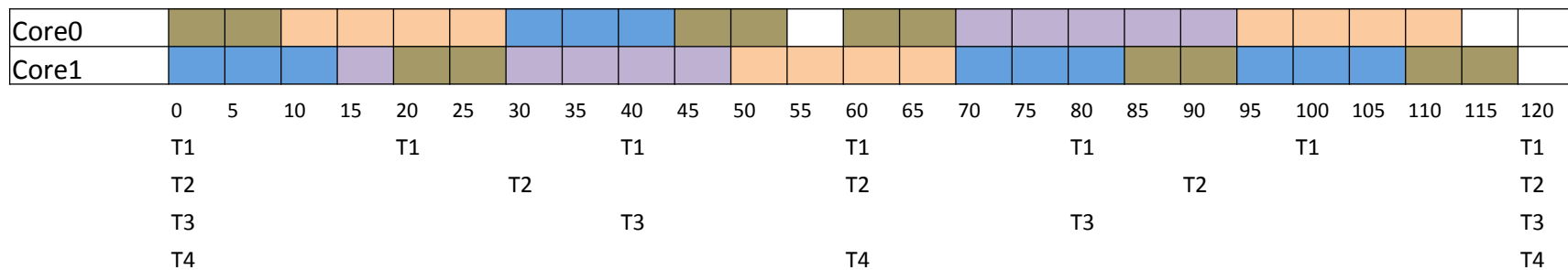
[illegible]

Planificación global

- 2 cores
- Expulsivo, con Migración y prioridades dinámicas

EDF

	C	P	D	U	Unid	Activ	Usa
1	10	20	20	0,50	2	6	12
2	15	30	30	0,50	3	4	12
3	20	40	40	0,50	4	3	12
4	25	60	60	0,42	5	2	10
						Usadas	46
H		120		1,92		Total	48



Planificación global

- Análisis
 - Capacidad de procesamiento $1.00 * m$ cores
 - Cores son homogéneos
 - Equivalente a un core con capacidad de cómputo de m cores
 - Tiempo de cómputo de peor caso / m Cores

	C	P	D	U	Unid	Activ	Usa
1	10	20	20	0,50	2	6	12
2	15	30	30	0,50	3	4	12
3	20	40	40	0,50	4	3	12
4	25	60	60	0,42	5	2	10
						Usadas	46
H		120		1,92		Total	48

	C	P	D	U
1	5	20	20	0,25
2	7,5	30	30	0,25
3	10	40	40	0,25
4	12,5	60	60	0,21
H		120		0,96

	C	P	D	U
1	10	40	40	0,25
2	15	60	60	0,25
3	20	80	80	0,25
4	25	120	120	0,21
H		240		0,96

Números enteros

Planificación global

- Análisis
 - Rate Monotonic / Deadline Monotonic
 - Con restricciones: Límite de Utilización

$$\frac{C_1}{T_1} + \frac{C_2}{T_2} + \dots + \frac{C_n}{T_n} \leq U(n) = n(2^{\frac{1}{n}} - 1)$$

	C	P	D	U
1	10	40	40	0,25
2	15	60	60	0,25
3	20	80	80	0,25
4	25	120	120	0,21
H		240		0,96

- Sin restricciones o mayor que el $U(n)$: Tiempo de respuesta de peor caso

$$W_i = C_i + \sum_{j \in hp(i)} \left\lceil \frac{W_i}{T_j} \right\rceil C_j$$

Planificación global

- Análisis

- EDF

- Con restricciones:

- Función de demanda

$$H(t) = \sum_{i \in \tau} (C_i * \left\lfloor \frac{t + T_i - D_i}{T_i} \right\rfloor)$$

	C	P	D	U
1	10	40	40	0,25
2	15	60	60	0,25
3	20	80	80	0,25
4	25	120	120	0,21
H		240		0,96

$$load(\tau) = \max_{\forall t} \left(\frac{h(t)}{t} \right) \quad load(\tau) \leq m$$

Clase	Máxima utilización
Global (activación migración), DPS	m
Otros	$(m+1)/2$

Planificación global

- Análisis

- EDF

- Sin restricciones:

$$\sum_{i \in \tau} \left(\frac{C_i}{T_i} \right) \leq m$$

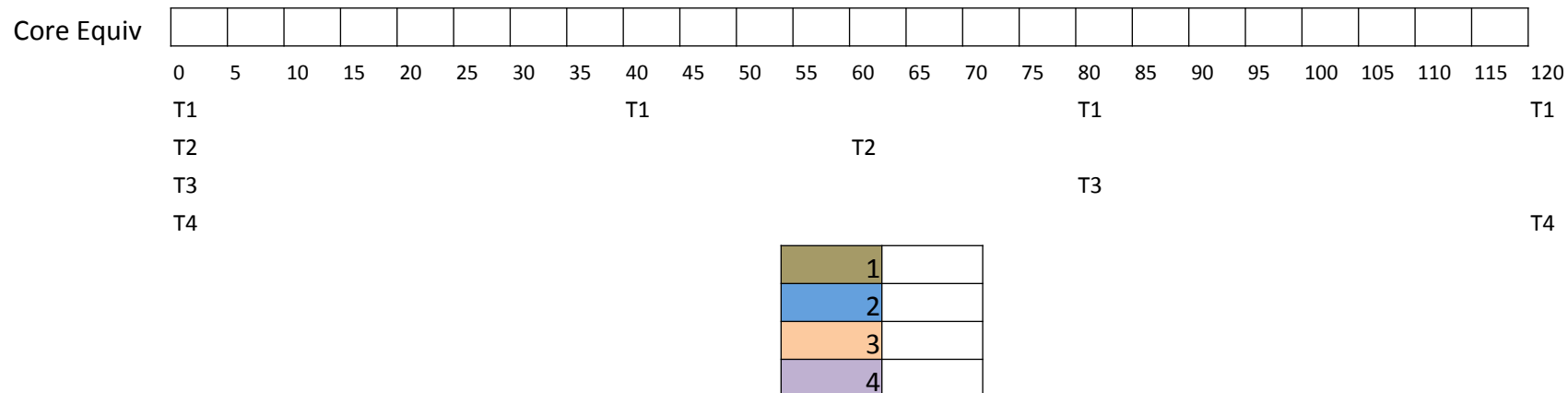
	C	P	D	U
1	10	40	40	0,25
2	15	60	60	0,25
3	20	80	80	0,25
4	25	120	120	0,21
H		240		0,96

Planificación global

- Análisis: RM/DM
 - Tiempo de respuesta de peor caso

$$W_i = C_i + \sum_{j \in hp(i)} \left\lceil \frac{W_i}{T_j} \right\rceil C_j$$

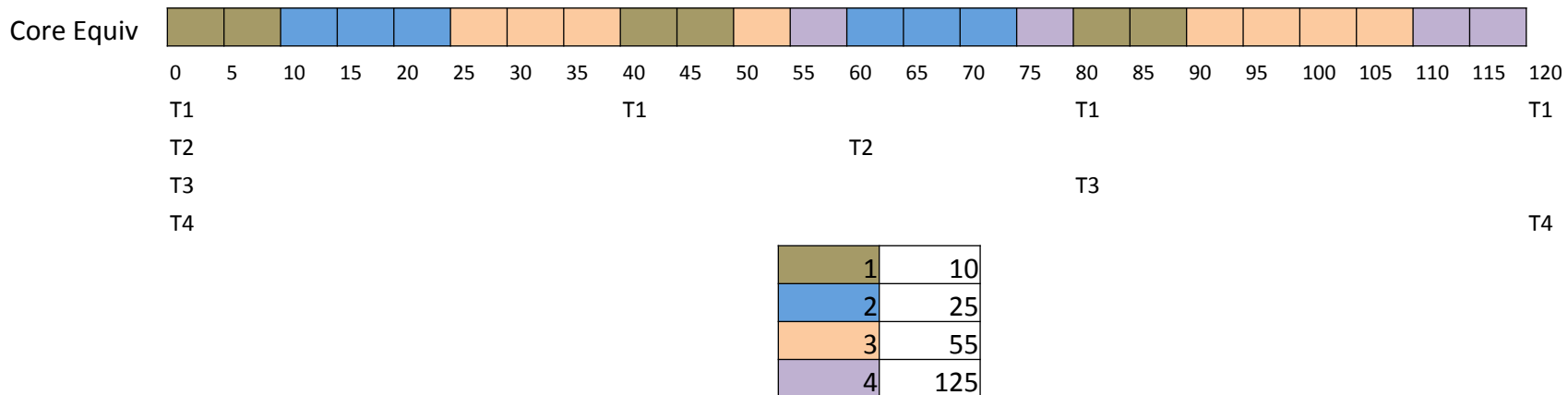
	C	P	D	U
1	10	40	40	0,25
2	15	60	60	0,25
3	20	80	80	0,25
4	25	120	120	0,21
H		240		0,96



- Análisis: RM/DM
 - Tiempo de respuesta de peor caso

$$W_i = C_i + \sum_{j \in hp(i)} \left[\frac{W_i}{T_j} \right] C_j$$

	C	P	D	U
1	10	40	40	0,25
2	15	60	60	0,25
3	20	80	80	0,25
4	25	120	120	0,21
	H	240		0,96



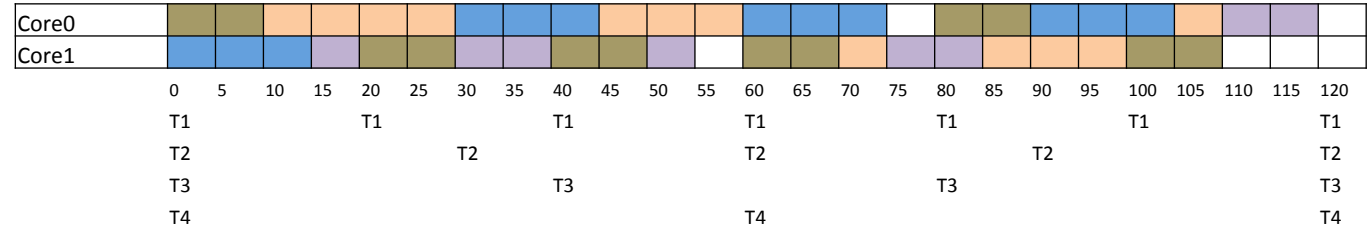
Planificación global

	C	P	D	U
1	10	40	40	0,25
2	15	60	60	0,25
3	20	80	80	0,25
4	20	120	120	0,17
H		240		0,92

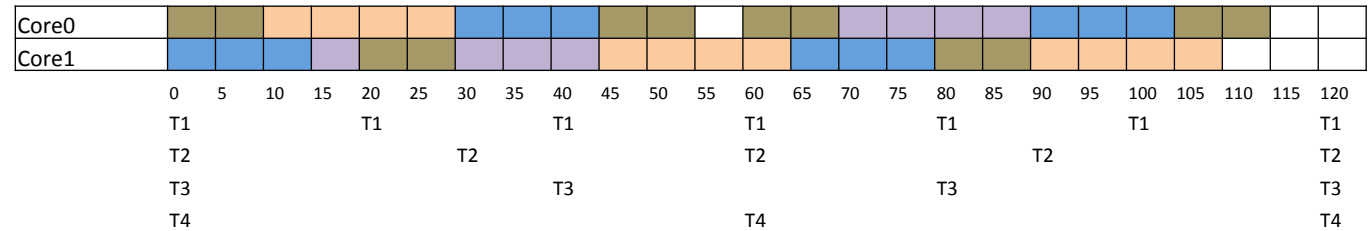
RM	WCRT
1	10
2	25
3	55
4	120

$$W_i = C_i + \sum_{j \in hp(i)} \left\lceil \frac{W_i}{T_j} \right\rceil C_j$$

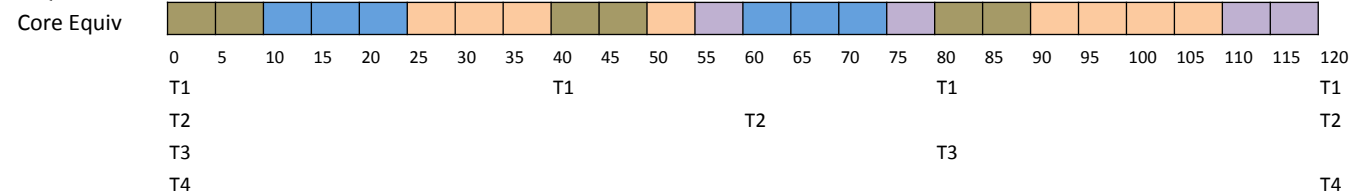
RM



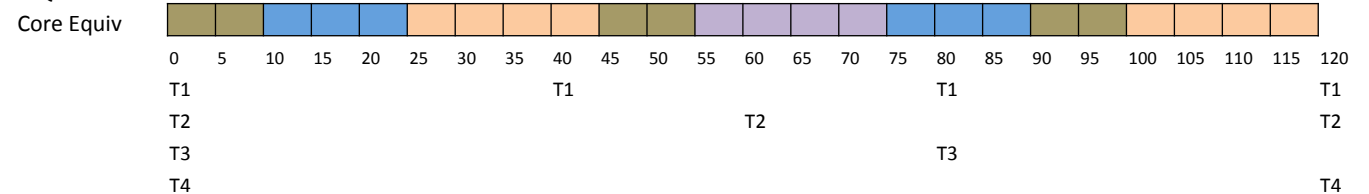
EDF



EQUIV- RM



EQUIV- EDF





Planificación Local

- Se realiza una asignación previa de tareas a core
- Se pueden usar distintos algoritmos para la ubicación (este problema es un clásico problema de la mochila o bin-packing)
- Problema:
 - Tenemos un conjunto de piezas de tamaño distinto (tareas, utilización)
 - Tenemos un conjunto de m mochilas de capacidad máxima (cores, 100%)
 - Se trata de rellenar cada mochila con piezas hasta agotar las piezas

- Bin-packing Algoritmos:
 - Los cores se numeran de 0 a m
 - Primer hueco (First Fit): Se asigna al core con menor índice no vacío. Si excede la capacidad, se asigna al core con menor índice vacío.
 - Siguiente Hueco (Next Fit): Se asigna al mismo core que se asignó el anterior. Si excede la capacidad, se asigna al core con menor índice vacío.
 - Mejor hueco (Best Fit): Se asigna al core con **menor** hueco disponible. Si hay varios iguales, al de menor índice. Si no cabe en ninguno, se asigna al core con menor índice vacío.
 - Peor hueco (Worst Fit): Se asigna al core con **mayor** hueco disponible. Si hay varios iguales, al de menor índice.

- Bin-packing Algoritmos:
 - Los cores se numeran de 0 a m
 - Algoritmos: FF, NF, BF, WF
 - Selección de tareas:
 - Orden creciente de tamaño (utilización)
 - Orden decreciente de tamaño (utilización)
 - Aleatorio.

Planificación Local

- Bin-packing Algoritmos:
 - Los cores se numeran de 0 a m
 - Algoritmos: FF, NF, BF, WF
 - Selección: UC, UD

	C	P	D	U
1	5	20	20	0,25
2	15	30	30	0,50
3	15	40	40	0,38
4	20	60	60	0,33
H		120		1,46

	FF-UC	U	FF-UD	U
Core0				
Core1				
	WF-UC	U	WF-UD	U
Core0				
Core1				

Planificación Local

- Bin-packing Algoritmos:
 - Los cores se numeran de 0 a m
 - Algoritmos: FF, NF, BF, WF
 - Selección: UC, UD

	C	P	D	U
1	5	20	20	0,25
2	15	30	30	0,50
3	15	40	40	0,38
4	20	60	60	0,33
H		120		1,46

	FF-UC	U	FF-UD	U
Core0	1;4;3	0,96	2;3	0,88
Core1	2	0,5	4;1	0,58
	WF-UC	U	WF-UD	U
Core0	1;3	0,63	2;1	0,75
Core1	4;2	0,83	3;4	0,71

Planificación global

El **approximation ratio**: *comparación de rendimiento con respecto al óptimo*.

Intenta contestar a: cuantos cores son necesarios para ejecutar un conjunto de tareas.

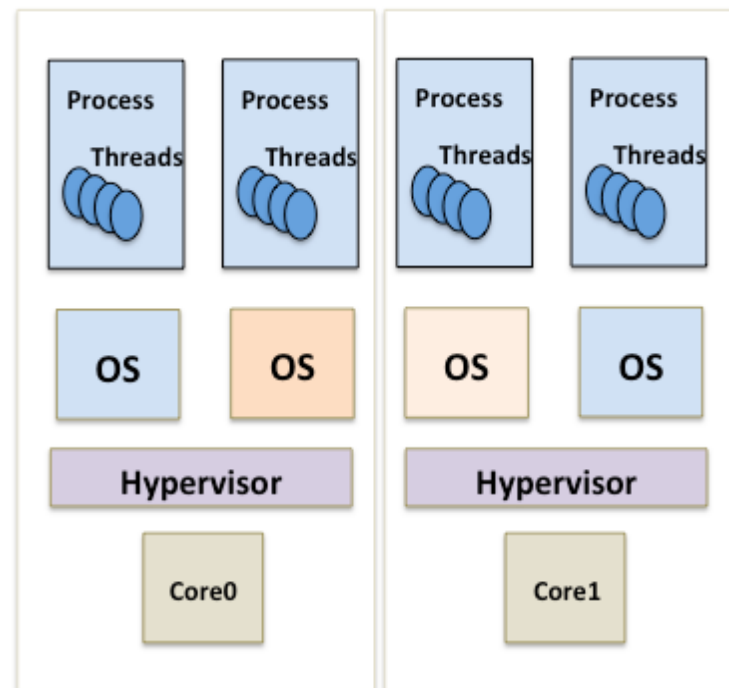
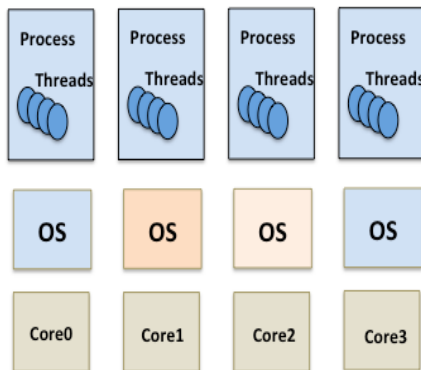
$$\mathfrak{R}_A = \lim_{M_O(\tau) \rightarrow \infty} \left(\max_{\forall \tau} \left(\frac{M_A(\tau)}{M_O(\tau)} \right) \right)$$

Algorithm	Approximation Ratio (\mathfrak{R}_A)	Ref.
RMNF	2.67	[Dhall and Liu 1978]
RMFF	2.33	[Oh and Son 1993]
RMBF	2.33	[Oh and Son 1993]
RRM-FF	2	[Oh and Son 1995]
FFDUF	2	[Davari and Dhall 1986]
RMST	$1/(1 - u_{\max})$	[Burchard et al. 1995]
RMGT	7/4	[Burchard et al. 1995]
RMMatching	3/2	[Rothvoß 2009]
EDF-FF	1.7	[Garey and Johnson 1979]
EDF-BF	1.7	[Garey and Johnson 1979]

Sistemas Particionados

Con hipervisor

Hypervisor based Asymmetric Multiprocessing (h-AMP)

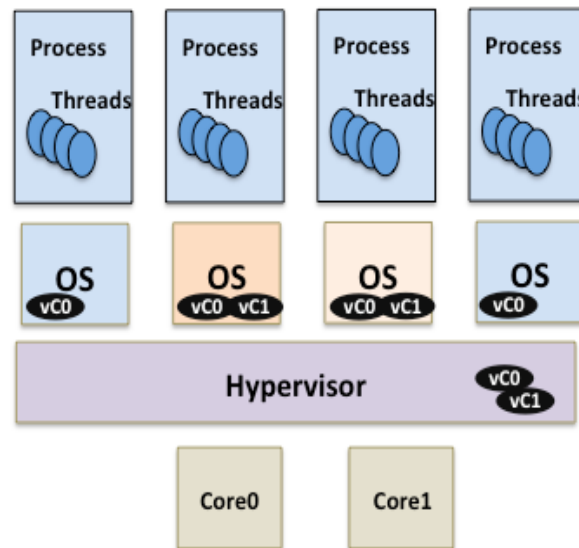
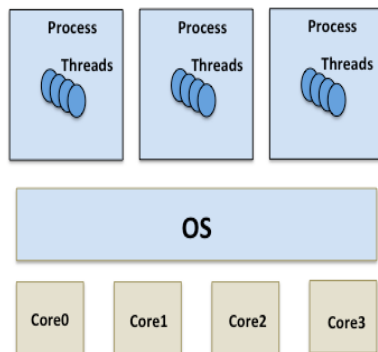


Sistemas Particionados

Con hipervisor

Hypervisor based Symmetric Multiprocessing (h-SMP)

- One hypervisor.



Ejemplo

- Sistemas particionado
- Criterio de asignación a core: partición
 - Todas las tareas de la partición al mismo core
- Sin migración

#tid	period	deadline	wcet	util	Partition	Criticidad	N.Tareas	U
T100	500	11	1	0,002	P10	1	6	0,046
T101	200	200	1	0,005				
T102	1000	11	2	0,0025				
T103	1000	17	1	0,001				
T104	500	500	1	0,002				
T105	125	31	1	0,01	P11	1	4	29,860
T110	1000	701	40	0,04				
T111	1000	755	130	0,13				
T112	125	16	3	0,025				
T113	200	181	20	0,1	P12	1	3	10,499
T120	1000	280	16	0,016				
T121	500	500	2	0,004				
T122	1000	418	80	0,08	P13	1	2	11,896
T130	250	108	23	0,092				
T131	40	40	1	0,025				
T140	125	93	3	0,024	P14	1	5	15,625
T141	200	18	2	0,0125				
T142	1000	232	40	0,04				
T143	125	125	2	0,016				
T144	250	56	12	0,05	P15	1	4	35,926
T150	250	84	18	0,075				
T151	500	500	43	0,086				
T152	500	488	41	0,0825				
T153	1000	581	111	0,111	P16	1	2	19,045
T160	100	100	2	0,02				
T161	125	108	20	0,16	P20	2	1	21,538
T200	40	40	8	0,2				
T210	250	250	87	0,348	P21	2	1	34,861
T220	40	40	9	0,225	P22	2	1	24,646
T230	80	80	42	0,525	P23	2	1	53,557
T300	50	50	4	0,08	P30	3	1	9,618
T310	250	250	45	0,18	P31	3	1	18,051
T320	400	400	92	0,23	P32	3	1	23,165
T330	1000	1000	157	0,157	P33	3	1	15,788
T340	200	200	19	0,095	P34	3	1	9,861
T350	50	50	2	0,04	P35	3	1	4,497
T360	250	250	28	0,112	P36	3	1	11,520

Ejemplo

- Sistemas particionado
- Criterio de asignación a core: partición
 - Todas las tareas de la partición al mismo core
- Sin migración

#tid	period	deadline	wcet	util	Partition	Criticidad	N.Tareas	U
T100	500	11	1	0,002	P10	1	6	0,046
T101	200	200	1	0,005				
T102	1000	11	2	0,0025				
T103	1000	17	1	0,001				
T104	500	500	1	0,002				
T105	125	31	1	0,01	P11	1	4	29,860
T110	1000	701	40	0,04				
T111	1000	755	130	0,13				
T112	125	16	3	0,025				
T113	200	181	20	0,1	P12	1	3	10,499
T120	1000	280	16	0,016				
T121	500	500	2	0,004				
T122	1000	418	80	0,08	P13	1	2	11,896
T130	250	108	23	0,092				
T131	40	40	1	0,025				
T140	125	93	3	0,024	P14	1	5	15,625
T141	200	18	2	0,0125				
T142	1000	232	40	0,04				
T143	125	125	2	0,016				
T144	250	56	12	0,05	P15	1	4	35,926
T150	250	84	18	0,075				
T151	500	500	43	0,086				
T152	500	488	41	0,0825	P16	1	2	19,045
			111	0,111				
			2	0,02				
			20	0,16				
			8	0,2	P20	2	1	21,538
			87	0,348	P21	2	1	34,861
			9	0,225	P22	2	1	24,646
			42	0,525	P23	2	1	53,557
			4	0,08	P30	3	1	9,618
			45	0,18	P31	3	1	18,051
			92	0,23	P32	3	1	23,165
			157	0,157	P33	3	1	15,788
			19	0,095	P34	3	1	9,861
			2	0,04	P35	3	1	4,497
			28	0,112	P36	3	1	11,520

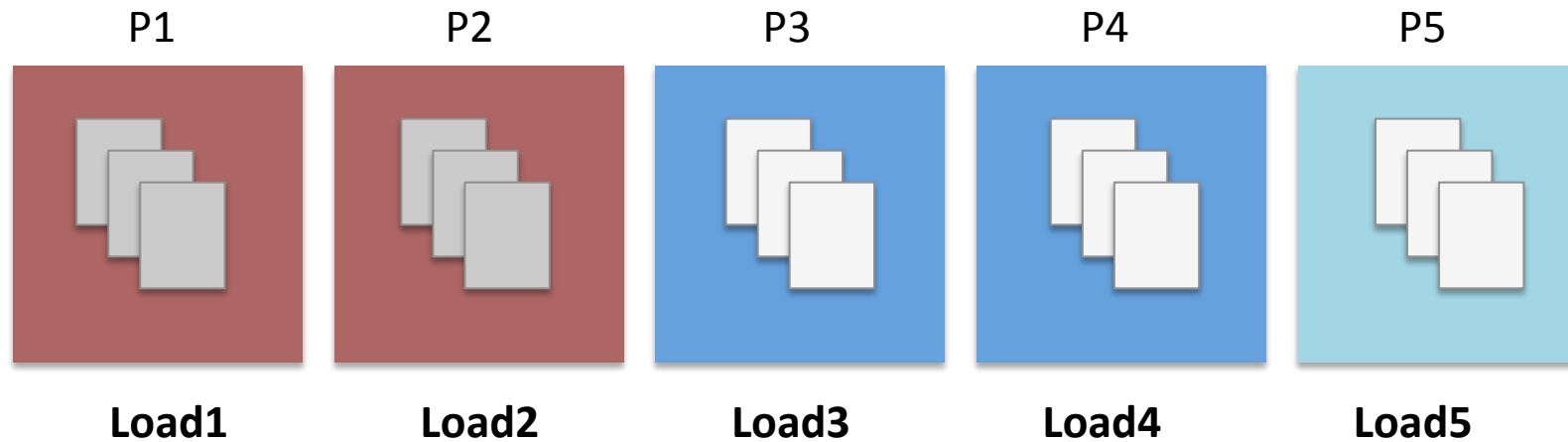
Sol 0 : C: P21;P11;P14;P10 C: P15;P22;P16;P12 C: P23;P20;P13 9.72458672531276
 Sol 1 : C: P21;P11;P22;P10 C: P15;P14;P16;P12 C: P23;P20;P13 8.31770477141264
 Sol 2 : C: P21;P15;P14;P10 C: P11;P22;P16;P12 C: P23;P20;P13 2.93987573487161
 Sol 3 : C: P21;P11;P16;P10 C: P15;P14;P22;P12 C: P23;P20;P13 3.17810426869517
 Sol 4 : C: P21;P11;P16;P10 C: P15;P22;P14;P12 C: P23;P20;P13 3.17810426869517
 Sol 5 : C: P21;P15;P14;P10 C: P11;P22;P20;P12 C: P23;P16;P13 2.04468750659258
 Sol 6 : C: P21;P15;P14;P10 C: P11;P22;P16;P13 C: P23;P20;P12 1.01072858152712
 Sol 7 : C: P15;P11;P14;P10 C: P21;P22;P16;P12 C: P23;P20;P13 7.59380644652877
 bestAllocation: P21;P15;P14;P10 P11;P22;P16;P13 P23;P20;P12 Util: 86.4585919906374
 85.4478634091103 85.5935446002521
 Utilization: Total: 3.392 Per core: 0.857 0.816 0.825 0.894

Ejemplo

- Cada core tiene una asignación de Particiones (con sus tareas).
- Se planifica de forma independiente usando:
 - RM/DF
 - EDF
 - Generación de un planificador cíclico

```
Sol 0 : C: P21;P11;P14;P10 C: P15;P22;P16;P12 C: P23;P20;P13 9.72458672531276
Sol 1 : C: P21;P11;P22;P10 C: P15;P14;P16;P12 C: P23;P20;P13 8.31770477141264
Sol 2 : C: P21;P15;P14;P10 C: P11;P22;P16;P12 C: P23;P20;P13 2.93987573487161
Sol 3 : C: P21;P11;P16;P10 C: P15;P14;P22;P12 C: P23;P20;P13 3.17810426869517
Sol 4 : C: P21;P11;P16;P10 C: P15;P22;P14;P12 C: P23;P20;P13 3.17810426869517
Sol 5 : C: P21;P15;P14;P10 C: P11;P22;P20;P12 C: P23;P16;P13 2.04468750659258
Sol 6 : C: P21;P15;P14;P10 C: P11;P22;P16;P13 C: P23;P20;P12 1.01072858152712
Sol 7 : C: P15;P11;P14;P10 C: P21;P22;P16;P12 C: P23;P20;P13 7.59380644652877
bestAllocation: P21;P15;P14;P10 P11;P22;P16;P13 P23;P20;P12 Util: 86.4585919906374
85.4478634091103 85.5935446002521
Utilization: Total: 3.392 Per core: 0.857 0.816 0.825 0.894
```

Planificación



Cores

0
1
2
3

Resultado:

- Número de cores para HC & MC particiones
- Número de cores para LC particiones

0 1 2
3

Cyclic scheduling

Priority scheduling

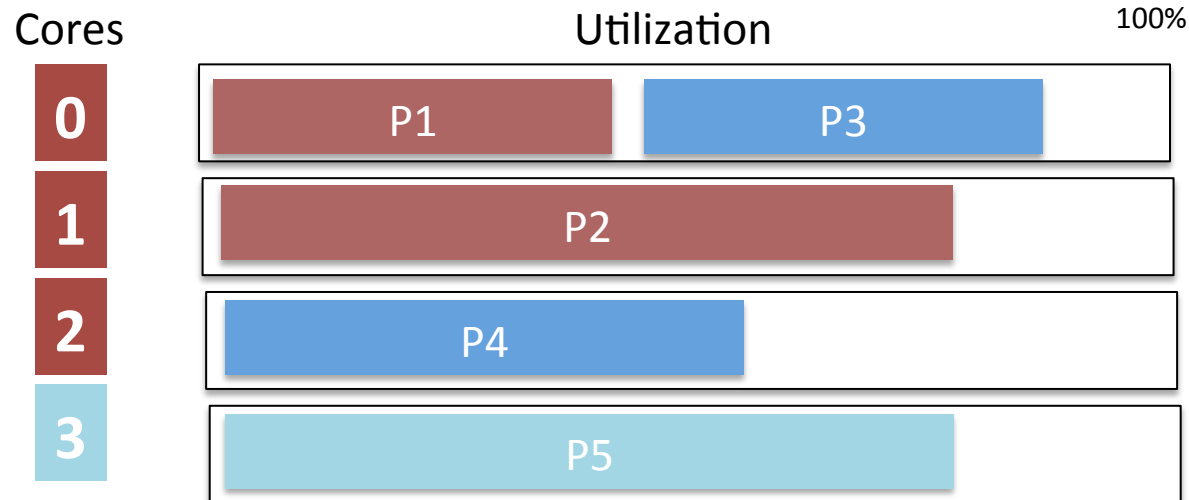
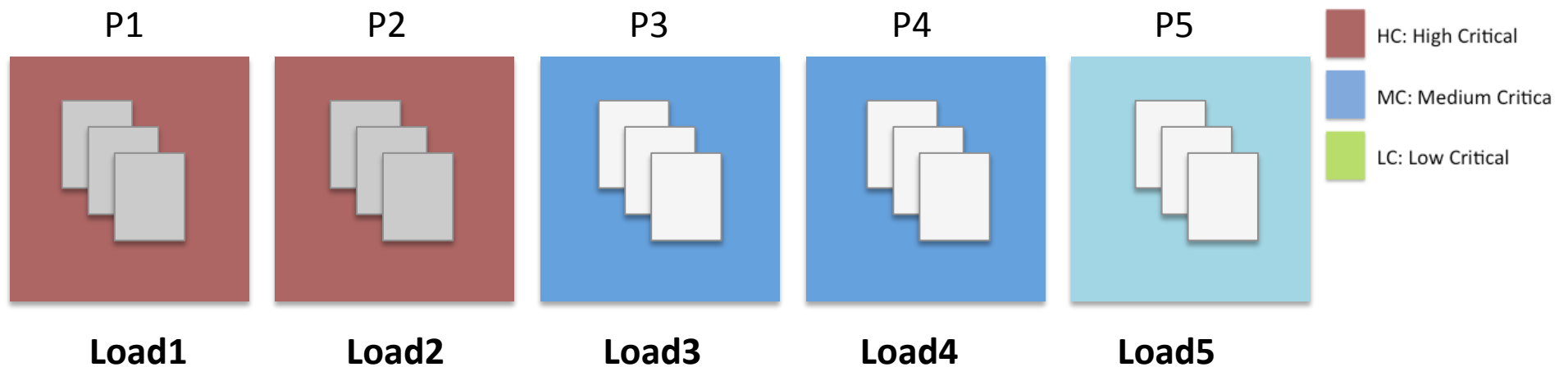


HC: High Critical

MC: Medium Critica

LC: Low Critical

Scheduling Generation: Step 2 Part. allocation



- 3 estrategias de bin-packing: First-Fit, Best-Fit and Worst-Fit.

Carga sintética:

Total load : [1.6,3.0]

HC and MC: [0.5,1.8]

LC load: lower than 1.0.

The number of HC and MC: [4,10]

LC in the range of [2, 6].

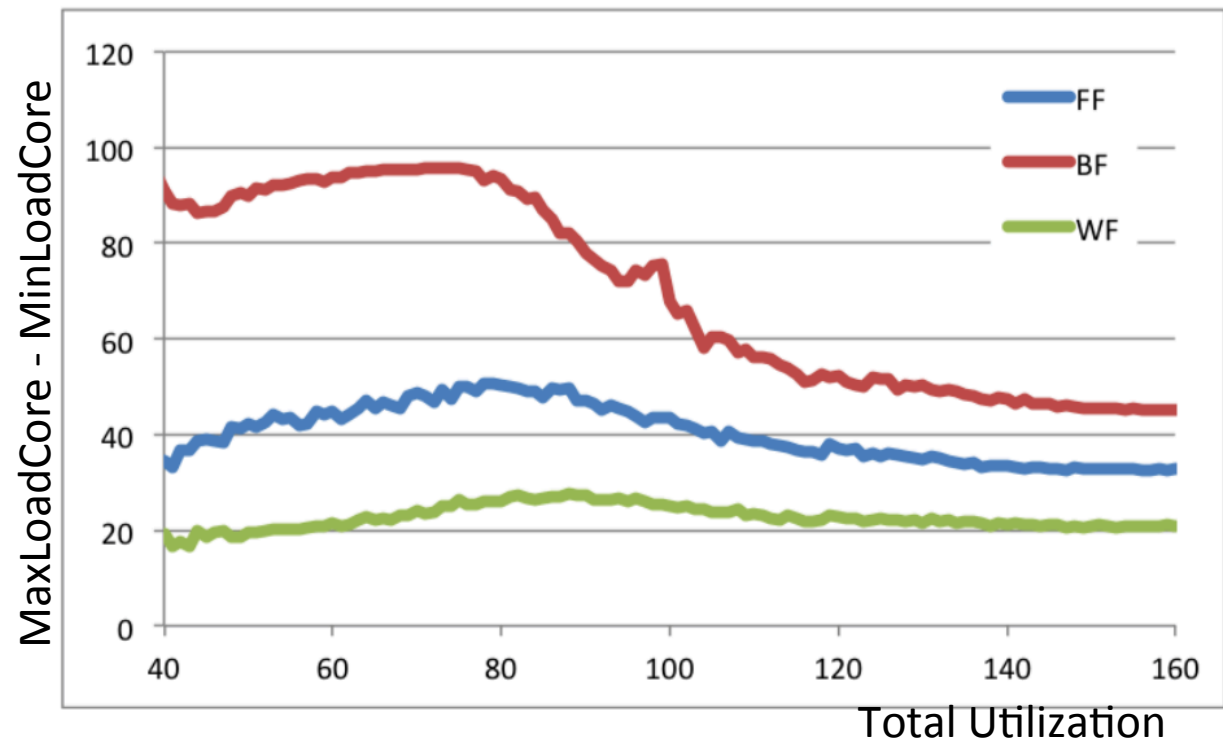


Fig. 2: FF, BF and WF Comparison

Scheduling Generation: Step 2 Part. allocation

- Usando Worst-Fit : 3 estrategias:
 - WF-R (Random), WF-I (Sorted by Incremental Load), WF-D (Sorted by Decremental Load)

The synthetic load ranges:

Total load : [1.6,3.0]

HC and MC: [0.5,1.8]

LC load: lower than 1.0.

The number of HC and MC: [4,10]

LC in the range of [2, 6].

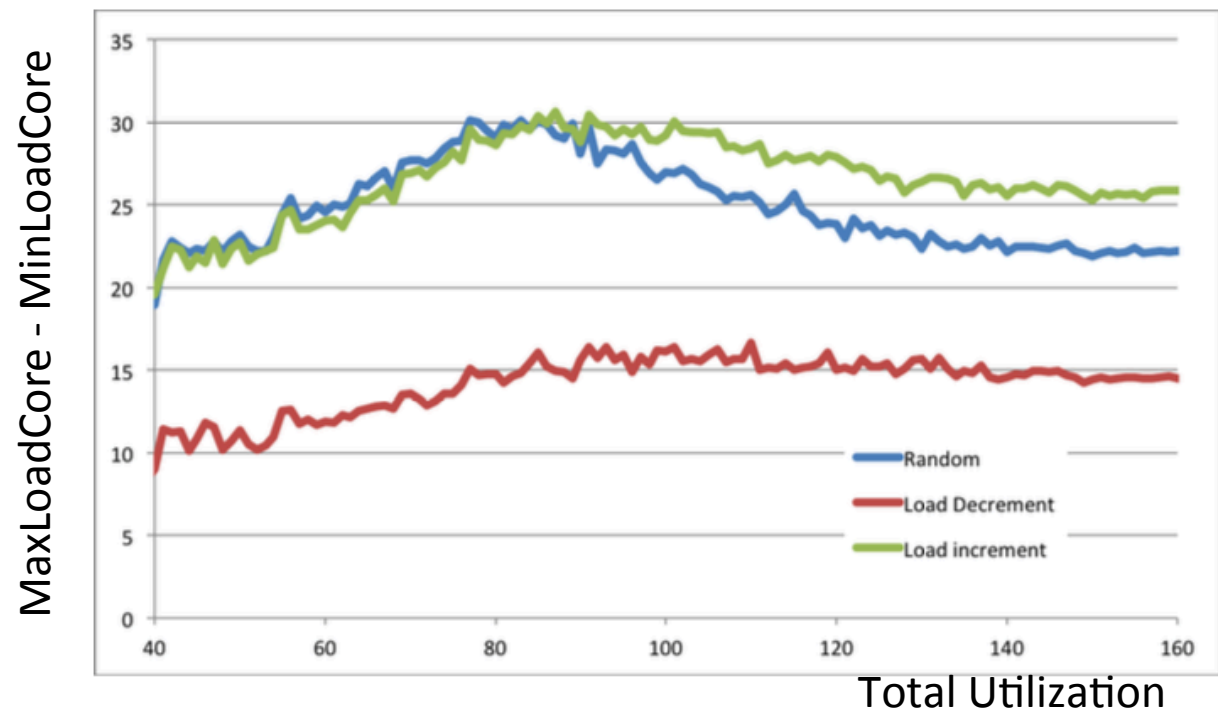


Fig. 3: WF-R, WF-I, WF-D Comparison