Practical, transparent operating system support for superpages

Presenta : Mariana Hernández

Sobre el artículo

Autores: Juan Navarro, Sitaram Iyer, Peter Druschel y Alan Cox

Rice University y Universidad Católica de Chile.

Artículo presentado en Symposium on Operating Systems Design and Implementation, Boston 2002.

Contenido

Contenido

1. Introducción

Cómo funciona la administración de la memoria mediante paginación.

Qué son las super páginas y qué problemas surgen al utilizarlas.

Memoria virtual y páginas

Es una solución a un problema de administración de memoria para poder disponer de más memoria que la disponible físicamente.

Existe un espacio de direcciones asignado a cada programa que se pretende ejecutar.

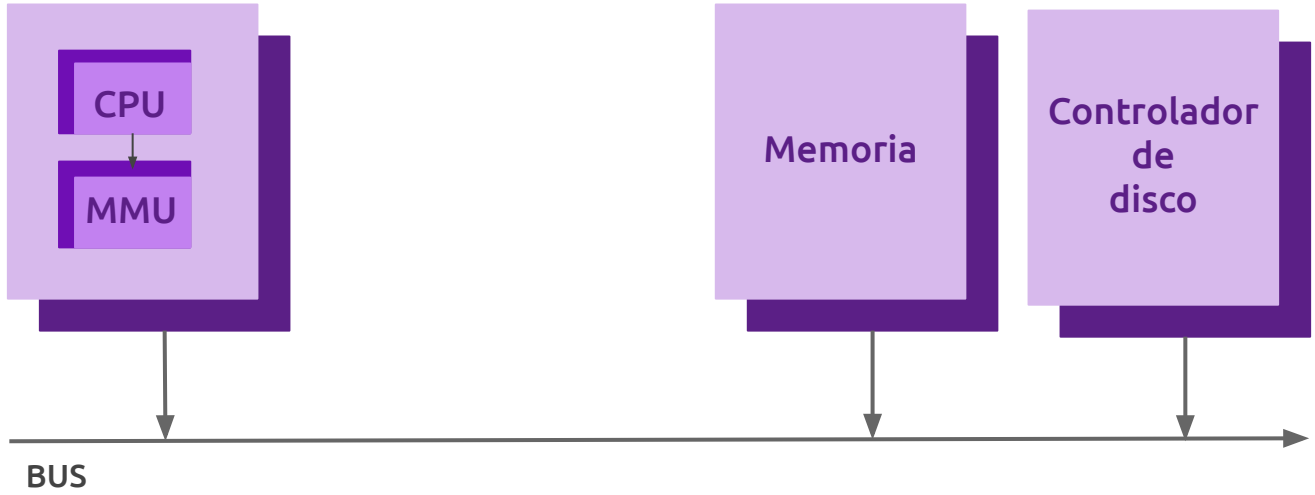
Este espacio está dividido a su vez en **páginas**.

Paginación - *Paging*

Cada página abarca un rango de direcciones de memoria y está asociada la memoria física.

Sin embargo, el contenido de la página, no se encuentra necesariamente en la memoria física.

¿Cómo funciona el *paging*?



¿Cómo llegamos a super páginas?

TLB

Translation Lookaside Buffer.
Es una memoria caché administrada por MMU, relaciona las memorias lógicas con las físicas.

TLB coverage

La cantidad de memoria a la que podemos acceder a través de ese mapeo.

Superpage

Una página de memoria más grande de lo normal.

El sistema de administración de superpages...

Asigna superpages balanceadamente

- Reserva regiones continuas de memoria física anticipadamente.
- Crea super páginas de tamaño incremental.

Este artículo aporta...

1. Extiende el trabajo existente.
2. Explora el efecto de la fragmentación de memoria en super páginas.
3. Propone reemplazo que sabe de la continuidad.
4. Resuelve otros problemas.

2. El problema de *super paging*

¿Por qué y cómo incrementar el TLB coverage?

Todo aumenta de tamaño, menos el TLB coverage.

Soluciones posibles:

Siempre usar páginas de un tamaño más grande.

↳ incrementa la fragmentación.

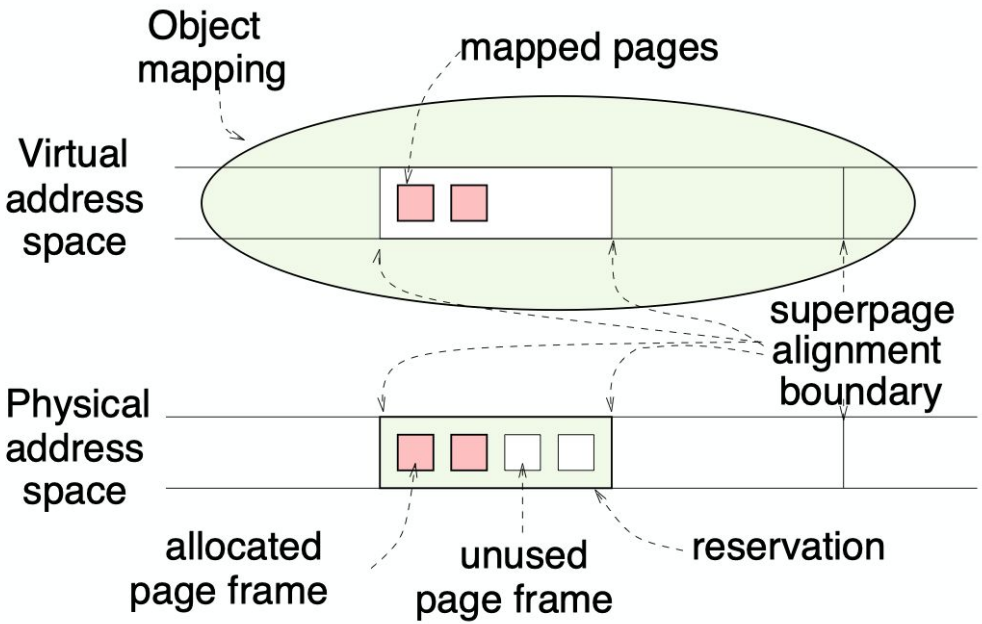
Usar múltiples tamaños de página.

↳ aumenta TLB coverage y mantiene baja la fragmentación.

Limitaciones de hardware

1. El tamaño debe ser soportado por el procesador.
2. Las superpáginas deben contener direcciones contiguas para memoria física y virtual.
3. La dirección de inicio, en el espacio de direccionamiento físico y virtual, deben ser múltiplos de su tamaño.
4. La entrada en el TLB para una superpágina, proporciona un solo bit de referencia (*dirty bit*) y los atributos de protección (*read,write,execute*).

Supuestos


Object mapping
mapped pages
Virtual address space
superpage alignment boundary
Physical address space
allocated page frame
unused page frame
reservation

Unas por otras 😣

Allocation

Reservation-based allocation
El SO asignará la página en una región donde las páginas, del mismo tamaño, están disponibles y contiguas.
Este acercamiento requiere de una opción *a priori* para el tamaño de la superpágina que se va a reservar.

Fragmentation control

Liberar fragmentos contiguos de memoria inactiva.
Apropiarse de un espacio reservado que se usó parcialmente.

La contigüidad es un recurso potencialmente en disputa.

Unas por otras 😣

Promotion

Cuando un conjunto de páginas base, cumplen con todas las restricciones antes mencionadas, son promovidas a superpágina.

Demotion

El proceso de ***Superpage demotion*** consiste en marcar las entradas de la page table para reducir el tamaño de la superpágina.

Eviction

Cuando una superpágina está inactiva, puede ser desalojada de la memoria física. Esto hará que todas las páginas base que la componen estén vacías.

3. Otros acercamientos

Otros acercamientos

Reservations

Deciden sobre cómo asignar de forma consciente/informada respecto a las super páginas en el momento en que falla una página.

Page relocation

Copia físicamente las páginas asignadas a regiones contiguas cuando se decide que una super página podría servir.

Hardware support

Hardware adicional.

Talluri y Hill proponen *partial-subblock* TLBs.

Fang agrega una fase extra de mapeo de las direcciones.

4. Diseño de la solución

Reservation-based allocation

- Page fault
- Escoger tamaño de súper página.
- Buddy allocator busca frames
- Reservation list

Política para escoger el tamaño de la súper página

- Atributos del objeto de memoria al que pertenece la página del fallo.
- Escoger el tamaño máximo de súper página que se puede usar efectivamente en el objeto.
- Objetos de tamaño fijo, objetos de tamaño dinámico.

Preempting reservations

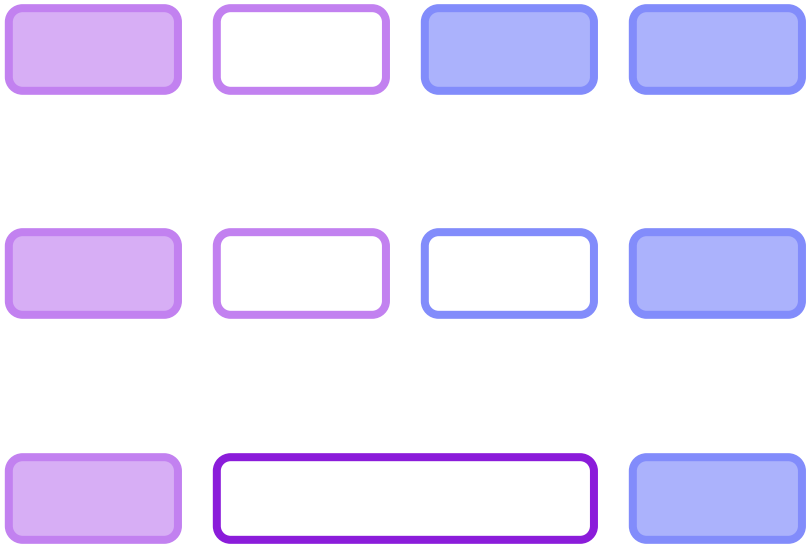
1. No asignar y reservar un espacio más chico.
2. Derecho preferente sobre frames reservados pero sin usar.

Siempre que sea posible, es preferible ejecutar la opción 2.

Fragmentation control

Buddy allocator tratará de fusionar los espacios que no están asignados.

Si no es posible fusionarlos, se encarga el *page replacement daemon*.



Incremental promotions

La super página se crea en cuanto se llena el espacio reservado.

La promoción se hace al siguiente tamaño más chico .

Speculative demotions

Al desalojar una página, la súper página se degrada. Este descenso es decremental.

También se puede degradar una página por mera especulación, para determinar si sigue activa.

Paging out dirty superpages

El sistema “degrada” las super páginas limpias cuando se quiere escribir en ellas y después las re-promueve si todas las páginas base fueron modificadas.

Técnica alternativa: Inferir qué páginas base “están sucias” usando *hash digests.*

Multi-list reservation scheme

Contiene conjuntos de páginas que no están completamente asignados.

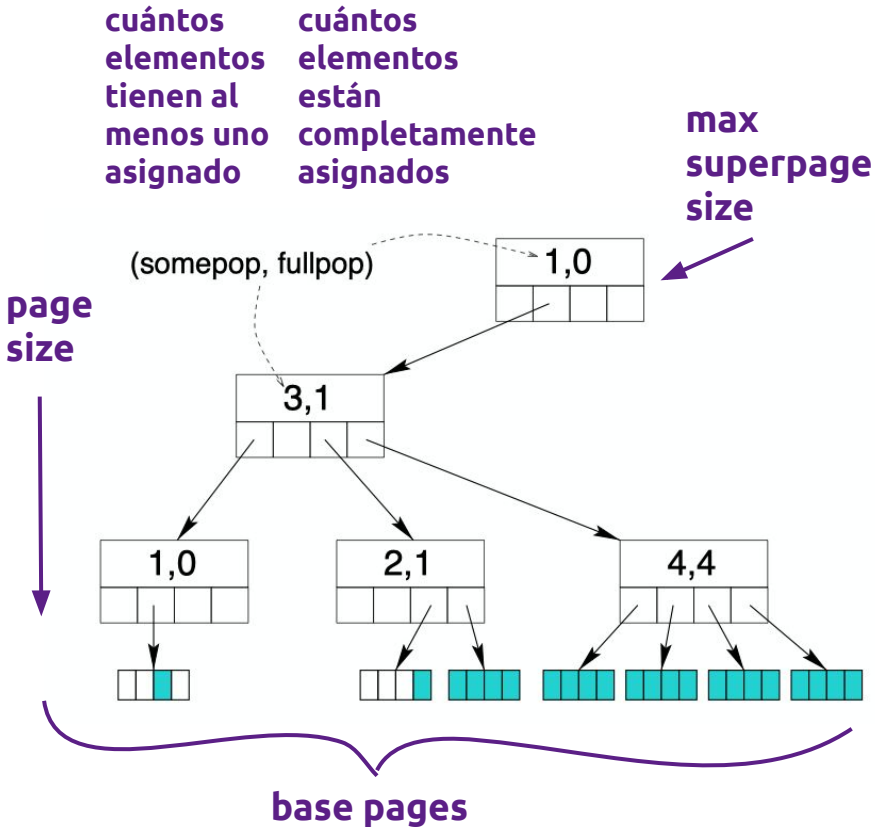
Una lista por tamaño de página soportado.

Elementos ordenados por su última asignación.

Population map

Mantiene registro de las páginas base que han sido asignadas en cada objeto de memoria.

1. Reserved frame lookup
2. Overlap avoidance
3. Promotion decisions
4. Preemption assistance



5. Implementación

Algunos ajustes

- Se realizaron unos cambios al *page daemon* de FreeBSD para que sea consciente de la continuidad en el sistema.
  - Todas las páginas de caché están disponibles para reservar.
  - *Page daemon* se invoca cuando nos estamos quedando cortos en memoria o en continuidad.
  - Las páginas limpias que fueron respaldadas, se mandan a la lista inactiva al cerrar el archivo.

Algunos ajustes

- Identificar las páginas que serán alambradas para uso del kernel, las agrupamos en *pools* de memoria física continua.

- El sistema escoge direcciones que sean compatibles con la asignación de super páginas.

6. Evaluación

Revisaremos los puntos más importantes sobre cómo se evaluó el desempeño de este sistema.

Mucha memoria libre y sin fragmentar

Bench-mark	Superpage usage 8 KB	64 KB	512 KB	4 MB	Miss reduc (%)	Speed-up
CINT2000						**1.112**
gzip	204	22	21	42	80.00	1.007
vpr	253	29	27	9	99.96	1.383
gcc	1209	1	17	35	70.79	1.013
mcf	206	7	10	46	99.97	1.676
crafty	147	13	2	0	99.33	1.036
parser	168	5	14	8	99.92	1.078
eon	297	6	0	0	0.00	1.000
perl	340	9	17	34	96.53	1.019
gap	267	8	7	47	99.49	1.017
vortex	280	4	15	17	99.75	1.112
bzip2	196	21	30	42	99.90	1.140
twolf	238	13	7	0	99.87	1.032

Bench-mark	Superpage usage 8 KB	64 KB	512 KB	4 MB	Miss reduc (%)	Speed-up
CFP2000						**1.110**
wupw	219	14	6	43	96.77	1.009
swim	226	16	11	46	98.97	1.034
mgrid	282	15	5	13	98.39	1.000
applu	1927	1647	90	5	93.53	1.020
mesa	246	13	8	1	99.14	0.985
galgel	957	172	68	2	99.80	1.289
art	163	4	7	0	99.55	1.122
equake	236	2	19	9	97.56	1.015
facerec	376	8	13	2	98.65	1.062
ammp	237	7	21	7	98.53	1.080
lucas	314	4	36	31	99.90	1.280
fma3d	500	17	27	22	96.77	1.000
sixtr	793	81	29	1	87.50	1.043
apsi	333	5	5	47	99.98	1.827

Bench-mark	Superpage usage 8 KB	64 KB	512 KB	4 MB	Miss reduc (%)	Speed-up
Web	30623	5	143	1	16.67	1.019
Image	163	1	17	7	75.00	1.228
Povray	136	6	17	14	97.44	1.042
Linker	6317	12	29	7	85.71	1.326
C4	76	2	9	0	95.65	1.360
Tree	207	6	14	1	97.14	1.503
SP	151	103	15	0	99.55	1.193
FFTW	160	5	7	60	99.59	1.549
Matrix	198	12	5	3	99.47	7.546

¡Grandes beneficios! 😄

Beneficios de la diversidad

Benchmark	64KB	512KB	4MB	All
CINT2000	1.05	1.09	1.05	1.11
vpr	1.28	1.38	1.13	1.38
mcf	1.24	1.31	1.22	1.68
vortex	1.01	1.07	1.08	1.11
bzip2	1.14	1.12	1.08	1.14
CFP2000	1.02	1.08	1.06	1.12
galgel	1.28	1.28	1.01	1.29
lucas	1.04	1.28	1.24	1.28
apsi	1.04	1.79	1.83	1.83
Image	1.19	1.19	1.16	1.23
Linker	1.16	1.26	1.19	1.32
C4	1.30	1.34	0.98	1.36
SP	1.19	1.17	0.98	1.19
FFTW	1.01	1.00	1.55	1.55
Matrix	3.83	7.17	6.86	7.54

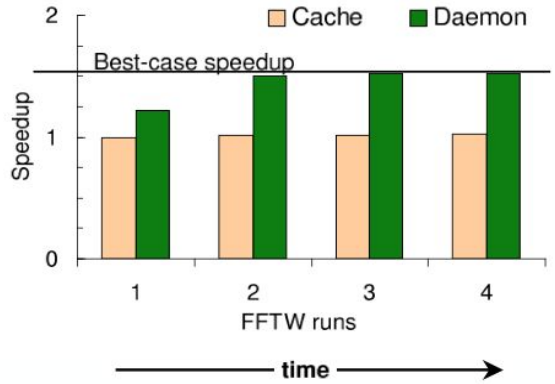
Repitieron los experimentos pero sólo permitieron un tamaño de super pagina, esto para los tamaños 64MB, 512MB, 4MB.

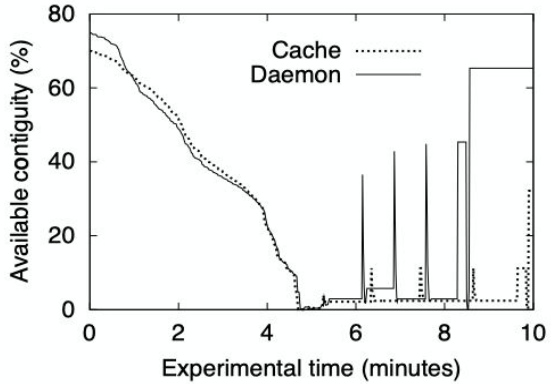
El mejor tamaño de super página depende de la aplicación y darle la libertad de escoger mejora el desempeño.

Que los beneficios se mantengan

Para evaluar las técnicas que usan para mantener la continuidad, primero fragmentan la memoria con un servidor web.

1. Sequential execution.
2. Concurrent execution.





Calar lo peor

Incremental promotion overhead
Alentamiento del 8.9%, aprox. 7.2% por razones de hardware y el otro 1.7% por mantener el population map.

Sequential access overhead
Empeoró el desempeño 0.1%.

Preemption overhead
El desempeño empeoró 1.1% en este experimento.

Overhead in practice
El desempeño empeoró hasta un 2%, el promedio fue 1%.

Dirty superpages y escalabilidad

- ❏ Sin demotion, se afecta mucho el desempeño.
- ❏ Procesos que no llenen todas las páginas de una super página no se benefician.

- Se puede escalar mientras se adapte al hardware.
- Casi todas las operaciones son O(1).