



OPERACIONES DE DETALLE SOBRE UN FICHERO

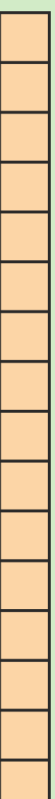
Pedro de Miguel Anasagasti

En esta animación se muestra el detalle de una operación de escritura y de lectura en un fichero. Destacando el que las operaciones sobre disco se realizan a nivel de bloque.

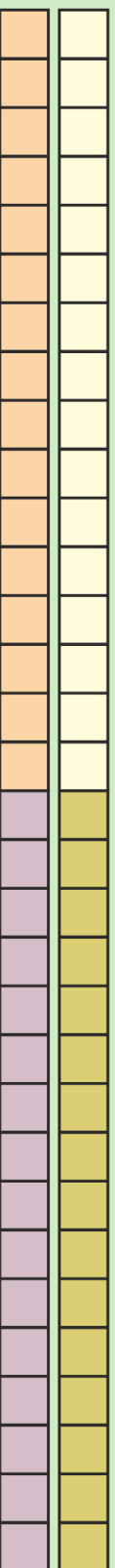
Para que los dibujos sean visibles vamos a considerar que el sector es de 8 bytes



el bloque de 2 sectores



y la agrupación de 4 bloques.



(Sabemos que el sector real tiene un tamaño de entre 128 a 1024 bytes).



SIN CACHE

- La situación de partida es la representada en la figura.
- Se ha indicado el espacio de memoria del proceso que va a solicitar los servicios, destacando las variables involucradas en los mismos.

Disco

a b c d e f g h i j k l m n o p q r s t u v w x y z n i e r o | Agr. 45

Disco

Proceso d 7

escrito ☐ leído ☐ buf B C D E F G H I J K L M N O P Q R S T U V X Y Z

SO

Memoria

BCPTabla fd			
0	2		
fd ↗	4		
7			
n	0		

Tabla intermedia				
Nodo-i	Puntero	Referen	RW	
1	463	37485	2	10
4	678	1	1	11
n	0	0	0	

Agrupac.: 45

Tipo: normal

Tamaño: 26

Nº de opens: 1

Nodo-i 678

escrito = write(d,buf, 4);

Paso 1

- La escritura afecta solamente a una parte del primer bloque de la agrupación 45.
- No es necesario añadir más agrupaciones.

Disco

a b c d e f g h i j k l m n o p q r s t u v w x y z n i e r o | Agr. 45

8 7 4 3 | \ I g n a c i o P é r e z Z o p i c o l P r o g r

Memoria

Proceso d 7
escrito ☐ leído ☐ buf B C D E F G H I J K L M N O P Q R S T U V X Y Z

SO

BCP Tabla fd			
0	2		
fd 7	4		
n	0		

Tabla intermedia				
Nodo-i	Puntero	Referen	RW	
1	463	37485	2	10
4	678	1	1	11
n	0	0	0	

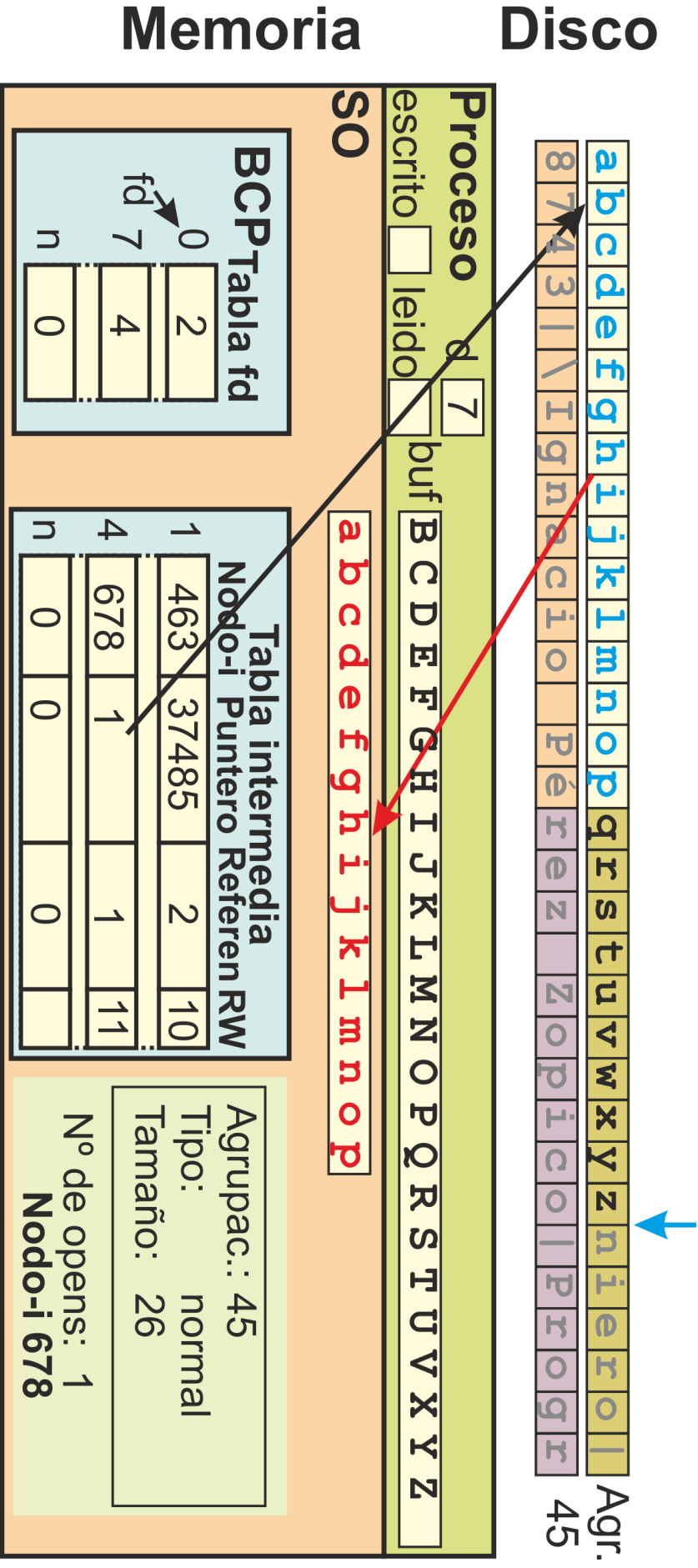
Agrupac.: 45
Tipo: normal
Tamaño: 26

Nº de opens: 1
Nodo-i 678

escrito = write(d,buf, 4);

Paso 2

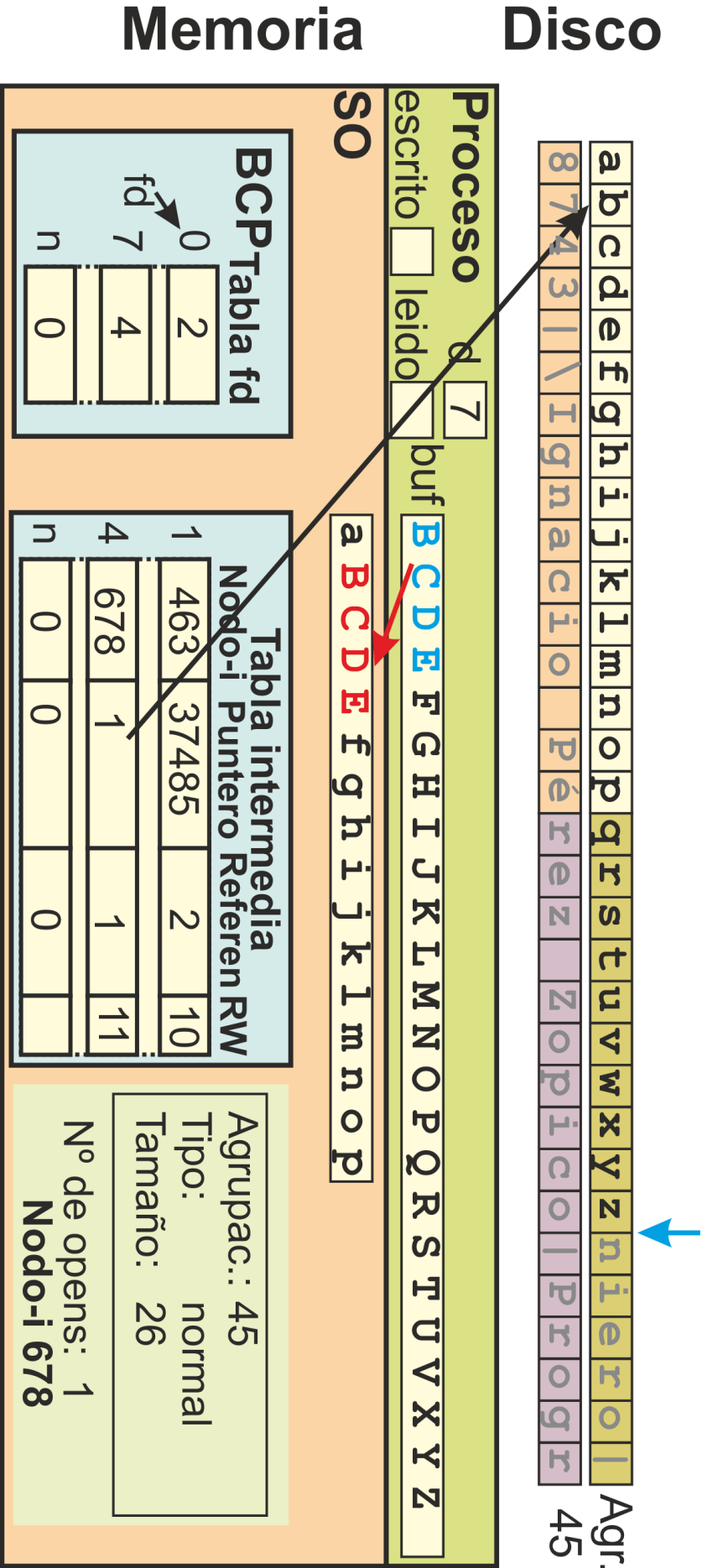
- El SO lee el bloque y lo almacena en un buffer en memoria.



escrito = write(d,buf, 4);

Paso 3

- El SO modifica el buffer con los bytes que hay que escribir. Estos bytes los obtiene de la variable buf ubicada en el espacio de memoria del proceso.



escrito = write(d,buf, 4);

Paso 4

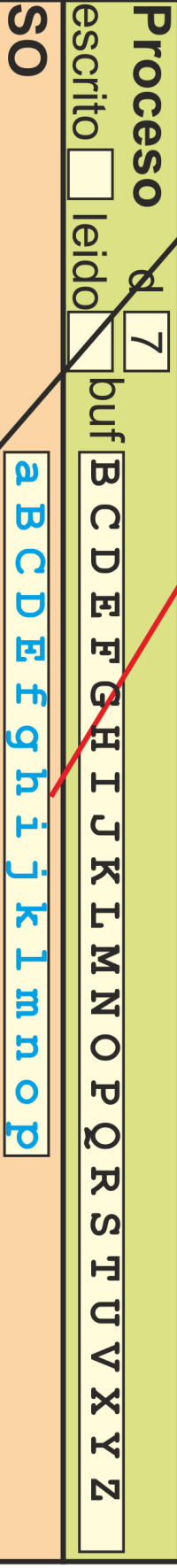
- El SO escribe el bloque en el disco.

Disco



Agr. 45

Memoria



BCP Tabla fd			
0	2		
fd	7	4	
n	0		

Tabla intermedia				
Nodo-i	Puntero	Referen	RW	
1	463	37485	2	10
4	678	1	1	11
n	0	0	0	

Agrupac.: 45
Tipo: normal
Tamaño: 26
Nº de opens: 1
Nodo-i 678

escrito = write(d,buf, 4);

Paso 5

- El SO actualiza el puntero y la variable escrito.
- escrito $\leftarrow$ 4.
- Puntero $\leftarrow$ 5.

Disco

a B C D E f g h i j k l m n o p q r s t u v w x y z n i e r o | Agr. 45

8 7 4 3 | l i g n a c i o P é r e z Z o p i c o l P r o g r

Disco

Proceso d 7

escrito 4 leído

buf B C D E F G H I J K L M N O P Q R S T U V X Y Z

Memoria

SO

BCPTabla fd

0 2

fd 7 4

n 0

Tabla intermedia

Nodo-i Puntero Referen RW

1 463 37485 2 10

4 678 5 1 11

n 0 0 0 0

Agrupac.: 45

Tipo: normal

Tamaño: 26

Nº de opens: 1

Nodo-i 678

leido=read(d, buf, 15);

Paso 1

- El SO lee los dos bloques involucrados.

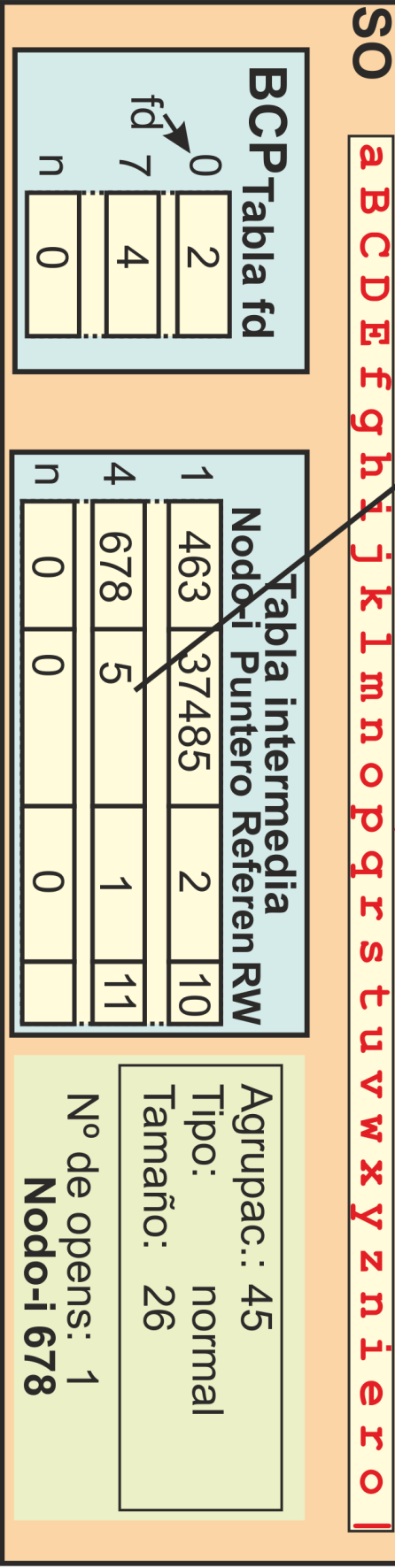
Disco



Disco



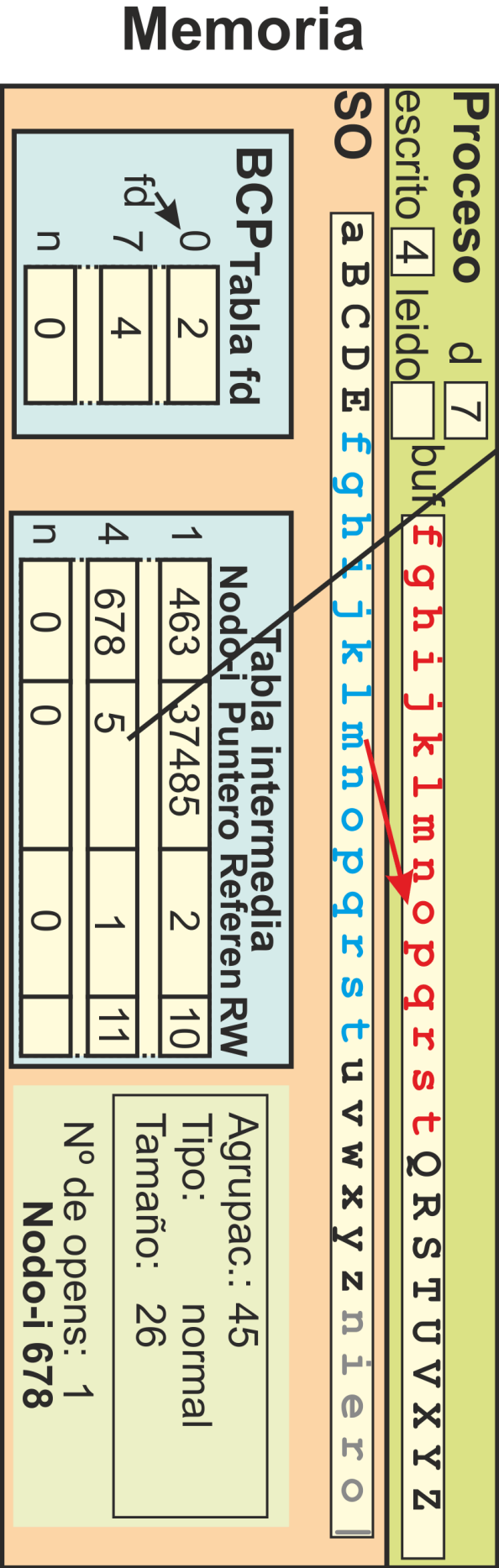
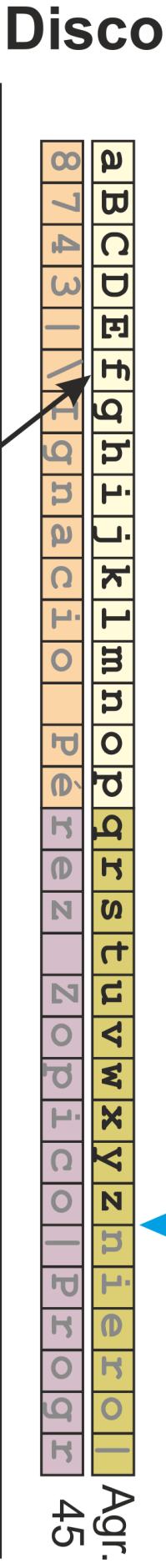
Memoria



leido=read(d, buf, 15);

Paso 2

- El SO selecciona los bytes y los copia en la variable buf del espacio de memoria del proceso.

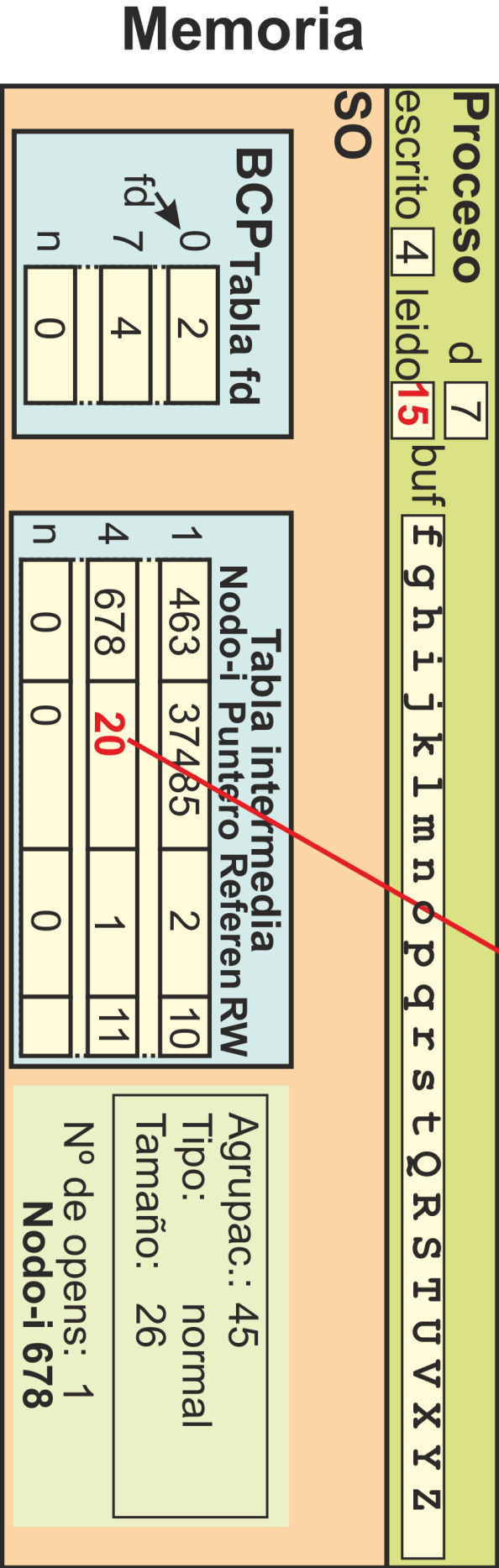
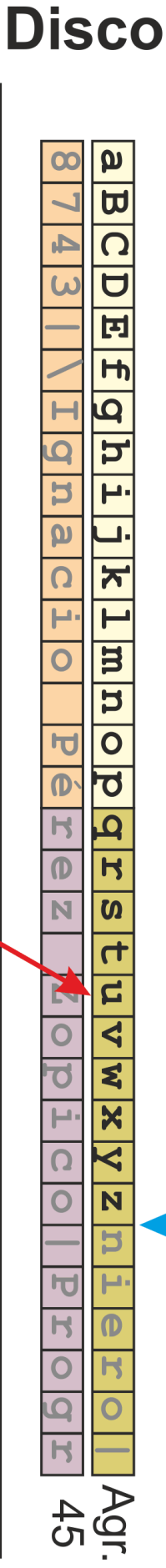


leido=read(d, buf, 15);

Paso 3

El SO actualiza el puntero y la variable leido.

- leido $\leftarrow$ 15.
- Puntero $\leftarrow$ 20.



CON CACHE

- La situación de partida es la representada en la figura.
- Se ha indicado el espacio de memoria del proceso que va a solicitar los servicios, destacando las variables involucradas en los mismos.

Disco

a b c d e f g h i j k l m n o p q r s t u v w x y z n i e r o | Agr. 45
8 7 4 3 | \ I g n a c i o P é r e z Z o p i c o l P r o g r

Disco

Proceso d 7

escrito ☐ leído ☐ buf B C D E F G H I J K L M N O P Q R S T U V X Y Z

SO

Memoria

BCP Tabla fd			
0	2		
fd ↗	4		
7			
n	0		

Tabla intermedia				
Nodo-i	Puntero	Referen	RW	
1	463	37485	2	10
4	678	1	1	11
n	0	0	0	

Agrupac.: 45

Tipo: normal

Tamaño: 26

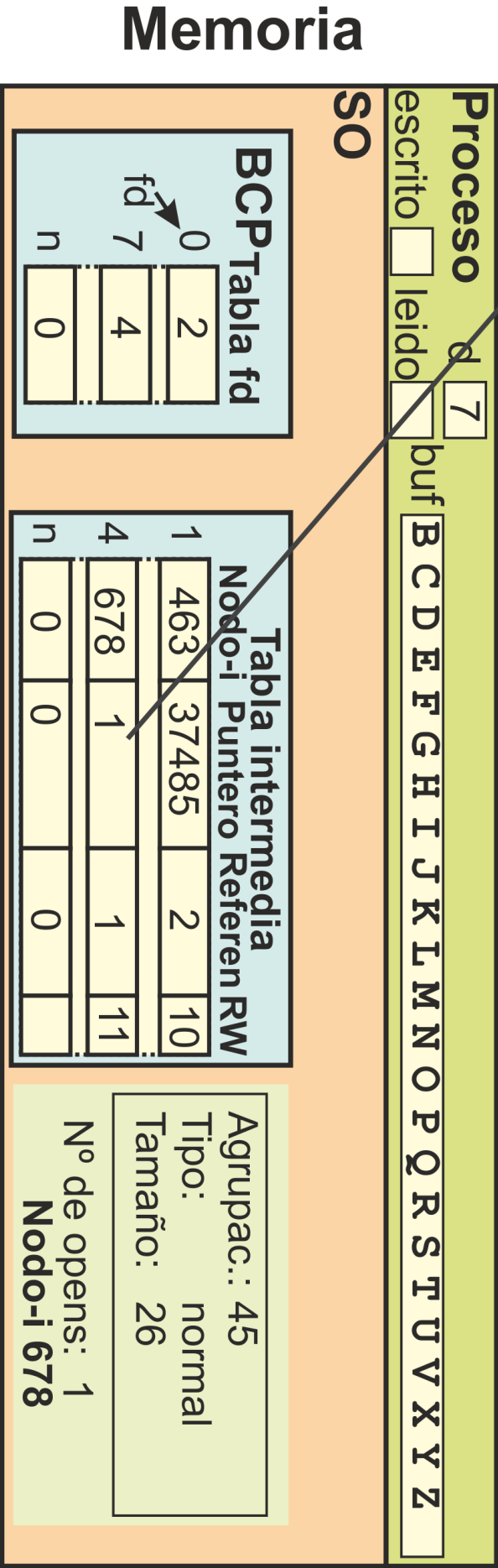
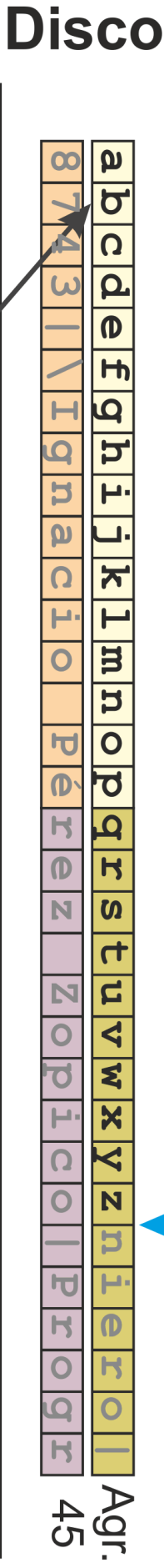
Nº de opens: 1

Nodo-i 678

escrito = write(d,buf, 4);

Paso 1

- La escritura afecta solamente a una parte del primer bloque de la agrupación 45.
- No es necesario añadir más agrupaciones.



escrito = write(d,buf, 4);

Paso 2

- El SO lee el bloque y lo almacena en un buffer en memoria.

Disco

a b c d e f g h i j k l m n o p q r s t u v w x y z n i e r o |
8 7 4 3 | \ | i g h a c i o P é r e z Z o p i c o | P r o g r |

Agr. 45

Memoria

Proceso d 7
escrito ☐ leído ☐ buf B C D E F G H I J K L M N O P Q R S T U V X Y Z

SO a b c d e f g h i j k l m n o p

BCP Tabla fd

0	2
fd 7	4
n	0

Tabla intermedia

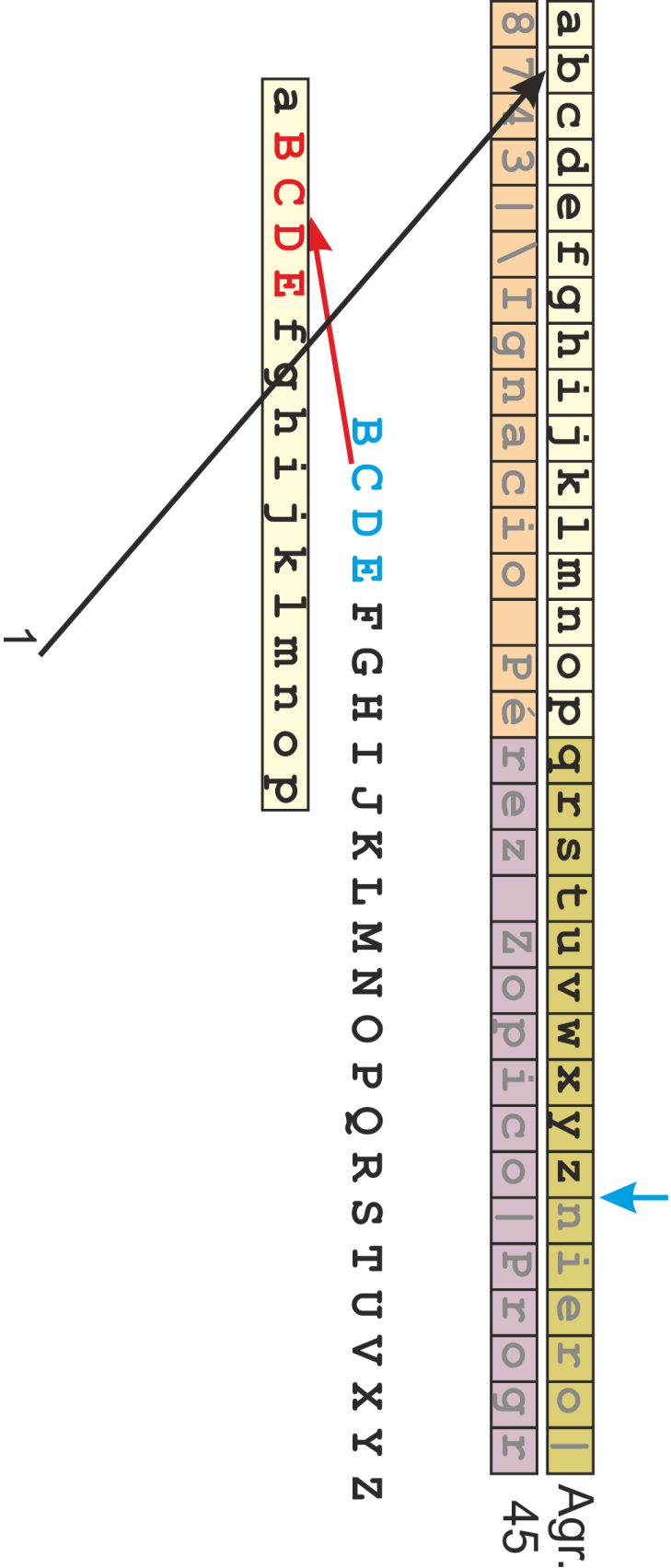
Nodo-i	Puntero	Referen	RW
1	463	37485	2 10
4	678	1	1 11
n	0	0	0

Agrupac.: 45
Tipo: normal
Tamaño: 26
Nº de opens: 1
Nodo-i 678

escrito = write(d,buf, 4);

Paso 3

- El SO modifica el buffer con los bytes que hay que escribir. Estos bytes los obtiene de la variable buf ubicada en el espacio de memoria del proceso.



escrito = write(d,buf, 4);

Paso 4

- El SO escribe el bloque en el disco.

Disco

a B C D E f g h i j k l m n o p q r s t u v w x y z n i e r o |
8 7 4 3 | \ I g n a c i o P é r e z Z o p i c o | P r o g r

Agr. 45

Memoria

Proceso d 7
escrito ☐ leído ☐ buf B C D E F G H I J K L M N O P Q R S T U V X Y Z

SO a B C D E f g h i j k l m n o p

BCPTabla fd
0 2
fd 7 4
n 0

Tabla intermedia				
Nodo-i	Puntero	Referen	RW	
1	463	37485	2	10
4	678	1	1	11
n	0	0	0	

Agrupac.: 45
Tipo: normal
Tamaño: 26

Nº de opens: 1
Nodo-i 678

escrito = write(d,buf, 4);

Paso 5

- El SO actualiza el puntero y la variable escrito.
- escrito ← 4.
- Puntero ← 5.

Disco

a B C D E f g h i j k l m n o p q r s t u v w x y z n i e r o | Agr. 45

8 7 4 3 | t g n a c i o P é r e z Z o p i c o l P r o g r

Disco

Proceso d 7
escrito 4 leído buf B C D E F G H I J K L M N O P Q R S T U V X Y Z

SO a B C D E f g h i j k l m n o p

Memoria

BCP Tabla fd			
0	2		
fd 7	4		
n	0		

Tabla intermedia				
Nodo-i	Puntero	Referen	RW	
1	463	37485	2	10
4	678	5	1	11
n	0	0	0	

Agrupac.: 45
Tipo: normal
Tamaño: 26
Nº de opens: 1
Nodo-i 678

leido=read(d, buf, 15);

Paso 1

- El SO lee un solo bloque el otro ya está en memoria.

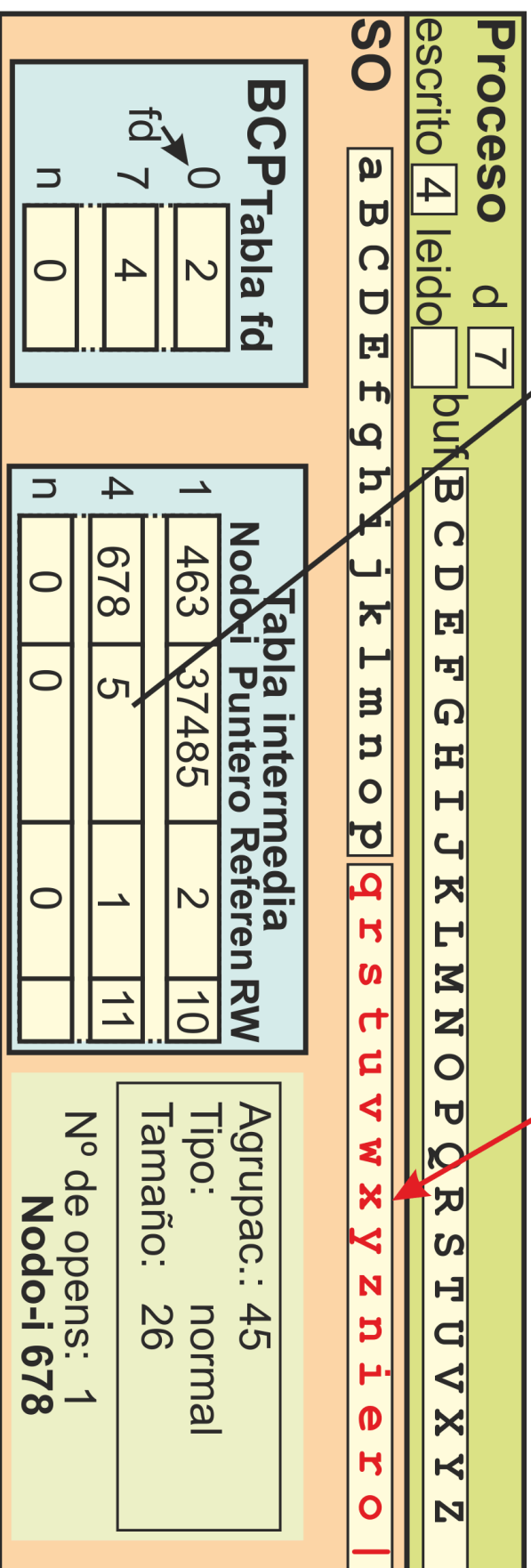
Disco

a B C D E f g h i j k l m n o p q r s t u v w x y z n i e r o | Agr. 45

8 7 4 3 | l i g n a c i o P é r e z Z o p i c o l P r o g r

Disco

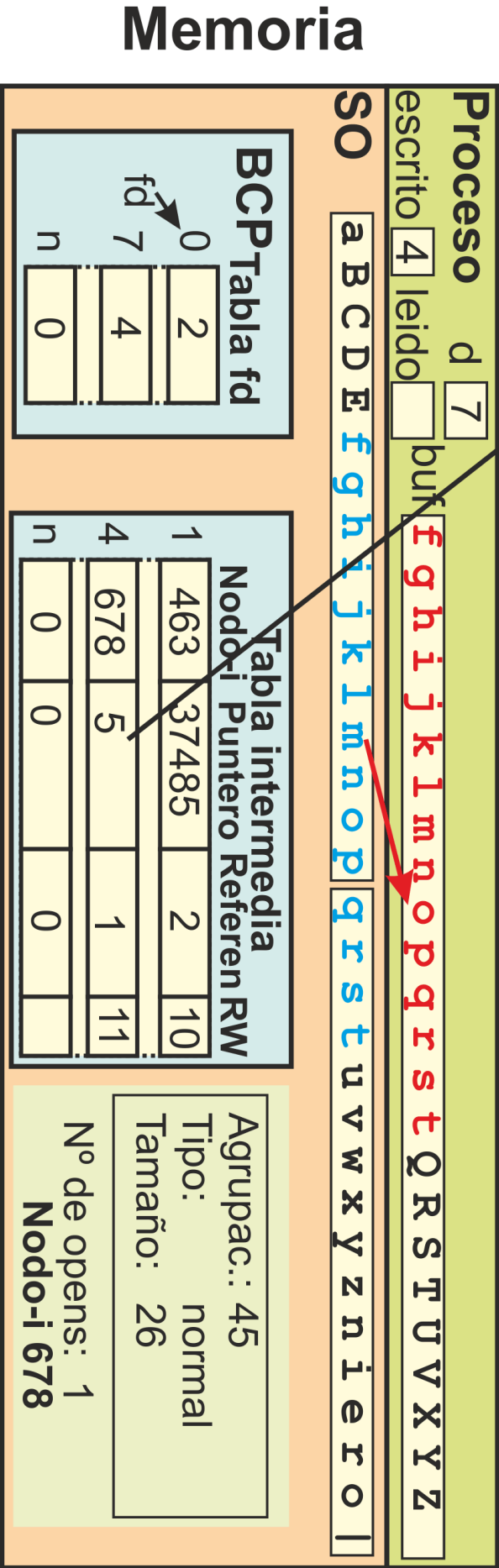
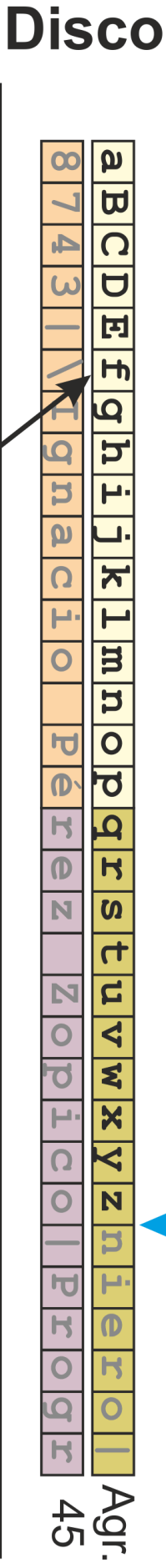
Memoria



leido=read(d, buf, 15);

Paso 2

- El SO selecciona los bytes y los copia en la variable buf del espacio de memoria del proceso.



leido=read(d, buf, 15);

Paso 3

El SO actualiza el puntero y la variable leido.

- leido $\leftarrow$ 15.
- Puntero $\leftarrow$ 20.

Disco

a	B	C	D	E	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	n	i	e	r	o		
8	7	4	3		\	I	g	n	a	c	i	o		P	é	r	e	z		L	o	p	i	c	o		P	r	o	g	r	

Agr. 45

SO

Proceso	d	7
escrito	4	leido 15
buf	f	g h i j k l m n o p q r s t Q R S T U V X Y Z

Memoria

BCP

Tabla	fd
0	2
7	4
n	0

Tabla intermedia

Nodo-i	Puntero	Referen	RW
1	463	37485	2 10
4	678	20	1 11
n	0	0	0

Nº de opens: 1

Nodo-i 678

FIN

