

Apuntes de L^AT_EX

Capítulo 10: Conceptos avanzados sobre estructura de tablas

1. El paquete `array`

El paquete `array` introduce una serie de mejoras sobre el entorno `tabular` estándar. La tabla siguiente describe los nuevos elementos introducidos por el paquete:

Argumentos de los entornos estándar <code>tabular</code> y <code>array</code>	
<code>l, r, c, p{Ancho}</code>	Formatos de columna, que mantienen su significado básico. Para <code>p{Ancho}</code> , el texto se coloca justificado a la parte de arriba de la casilla.
<code> @{Objeto}</code>	Mantienen su significado, aunque se introduce una mejora para <code>' '</code> : el espacio entre columnas se incrementa en el valor correspondiente a la anchura de la raya vertical.
Nuevos argumentos con el paquete <code>array</code>	
<code>m{Ancho}</code>	Nuevo especificador de columna, similar a <code>p{Ancho}</code> , con la diferencia de que las entradas aparecen centradas en sentido vertical.
<code>b{Ancho}</code>	Similar al anterior, pero las entradas se justifican a la parte inferior.
<code>!{Objeto}</code>	Nuevo separador de columnas, similar a <code>@{Objeto}</code> . La diferencia está en que no suprime el espacio normal entre columnas.
<code>>{comando}</code>	Se debe incluir antes de un especificador <code>l, r, c, p, m, b</code> . Tiene el efecto de aplicar el comando a cada una de las casillas de la columna.
<code><{comando}</code>	Análogo al anterior, sólo que se utiliza <i>después</i> del especificador de columna, y aplica el comando detrás de cada columna.

Para casillas tipo párrafo (`p{Ancho}`, `m{Ancho}` y `b{Ancho}`), es posible dividir el texto en varias líneas, y justificarlo a cualquiera de los lados utilizando los comandos:

- `\raggedright` → El texto se ajusta sólo por la izquierda
- `\centering` → Texto centrado
- `\raggedleft` → El texto se ajusta sólo por la derecha

Dentro de cada casilla, cada línea se separa utilizando el comando estándar `\\`. Es importante tener en cuenta que, para la última casilla de una fila, se debe terminar con el comando `\tabularnewline`, que indica a L^AT_EX que hemos finalizado con esa fila de la tabla.

Ejemplo:

```
\begin{tabular}{|l|c|} \hline
\multicolumn{1}{|p{1.5cm}|}{\centering %
Planeta} & \multicolumn{1}{p{4cm}|}%
{\centering Distancia media al sol \\
(millones de km)} \tabularnewline \hline
Mercurio & 58.1 \\
Venus & 108.3 \\
Tierra & 150.0 \\ \hline
\end{tabular}
```

Planeta	Distancia media al sol (millones de km)
Mercurio	58.1
Venus	108.3
Tierra	150.0

Cambiando los formatos “p{1.5cm}” y “p{4cm}” por “m{1.5cm}” y “m{4cm}” ó por “b{1.5cm}” y “b{4cm}” obtendríamos, respectivamente:

Planeta	Distancia media al sol (millones de km)
Mercurio	58.1
Venus	108.3
Tierra	150.0

Planeta	Distancia media al sol (millones de km)
Mercurio	58.1
Venus	108.3
Tierra	150.0

Vemos en los ejemplo anteriores que los puntos de los decimales no quedan bien alineados. Una posible solución sería colocar el signo “.” como separador de columna, de la siguiente forma:

```
\begin{tabular}{|l|r@{.}l|}
\hline
\multicolumn{1}{|m{1.5cm}|}{\centering %
Planeta} & \multicolumn{2}{m{4cm}|}%
{\centering Distancia media al sol \\
(millones de km)} \tabularnewline \hline
Mercurio & 58&1 \\
Venus & 108&3 \\
Tierra & 150&1 \\ \hline
\end{tabular}
```

Planeta	Distancia media al sol (millones de km)
Mercurio	58.1
Venus	108.3
Tierra	150.1

Sin embargo, tal procedimiento no da buenos resultados para columnas muy anchas. La solución, para este caso, se encuentra en utilizar el paquete `dcolumn`. Este paquete introduce un nuevo separador, de sintaxis:

`D{separador1}{separador2}{Num}`

donde `separador1` es el decimal que utilizamos en el fichero `.tex`, y `separador2` es el decimal que L^AT_EX colocará en la salida. `Num` denota el número máximo de decimales en la columna (el valor -1 equivaldría a cualquiera). Por ejemplo, obtendríamos el resultado deseado con:

```

\begin{center}
\begin{tabular}{|l|D{,}{.}{-1}|}
\hline
\multicolumn{1}{|m{1.5cm}|}{\centering %
Planeta} & \multicolumn{1}{m{4cm}|}%
{\centering Distancia media al sol \\
(millones de km)} \tabularnewline \hline
Mercurio & 58,1 \\
Venus & 108,3 \\
Tierra & 150,1 \\
\hline
\end{tabular}
\end{center}

```

Planeta	Distancia media al sol (millones de km)
Mercurio	58.1
Venus	108.3
Tierra	150.1

Los argumentos `>{comando}` y `<{comando}` nos permiten hacer útiles manipulaciones por columnas. Por ejemplo, si queremos poner en negrita, añadir color, etc... una columna en concreto, podemos especificar:

```

\begin{center}
\begin{tabular}{|>{\bfseries}l|}
>{\color{red}}c|}
\hline
\multicolumn{1}{|m{1.5cm}|}{\centering %
Planeta} & \multicolumn{1}{m{4cm}|}%
{\centering Distancia media al sol \\
(millones de km)} \tabularnewline \hline
Mercurio & 58.1 \\
Venus & 108.3 \\
Tierra & 150.1 \\
\hline
\end{tabular}
\end{center}

```

Planeta	Distancia media al sol (millones de km)
Mercurio	58.1
Venus	108.3
Tierra	150.1

La utilidad del comando de cierre `<{comando}` puede verse en el siguiente ejemplo:

```

\begin{tabular}{|c|>{\displaystyle}c<{}}|}
\hline
Función & \text{Definición} \\ \hline
tangente & \frac{\sen(x)}{\cos(x)} \\ \hline
cosecante & \frac{\cos(x)}{\sen(x)} \\ \hline
\end{tabular}

```

Función	Definición
tangente	$\frac{\sen(x)}{\cos(x)}$
cosecante	$\frac{\cos(x)}{\sen(x)}$

donde los contenidos de la segunda columna se escriben automáticamente en modo matemático tipo párrafo.

Si un determinado formato de columna se va a repetir en varias tablas (ó columnas), es conveniente utilizar el siguiente comando, proporcionado por el paquete array, que permite construir nuevos formatos:

```

\newcolumnntype{Carácter}{Definición}

```

Así por ejemplo, la expresión `>{\displaystyle}c<{}` en la tabla anterior puede ser abreviada a 'X', utilizando previamente:

`\newcolumnntype{X}{>{\displaystyle}c<{\$}}`

En tablas con líneas horizontales, las letras mayúsculas pueden quedar demasiado cerca de las líneas; para solventar ésto el paquete `array` proporciona la longitud `\extrarowheight`, que permite añadir una pequeña altura a cada línea mejorando el resultado, como puede verse en el siguiente ejemplo (donde además, se muestra la diferencia entre los separadores @ y !)

```
\newcolumnntype{L}{>{\itshape}r}
\begin{tabular}{|l|L@{---}l|}
\hline
  & Nombre & Name \\ \hline
  1 & uno & one \\
  2 & dos & two \\
  3 & tres & three \\ \hline
\end{tabular}
\par \bigskip
\setlength{\extrarowheight}{2pt}
\begin{tabular}{|l|L!{---}l|}
\hline
  & Nombre & Name \\ \hline
  1 & uno & one \\
  2 & dos & two \\
  3 & tres & three \\ \hline
\end{tabular}
```

	<i>Nombre</i> —Name
1	<i>uno</i> —one
2	<i>dos</i> —two
3	<i>tres</i> —three

	<i>Nombre</i> — Name
1	<i>uno</i> — one
2	<i>dos</i> — two
3	<i>tres</i> — three

2. El paquete `multirow`

El paquete `multirow` nos permite construir tablas en las que algunas celdas ocupan varias filas dentro de un entorno tabular. Se utiliza la orden:

`\multirow{nrow}{width}[vmove]{contenido}`

donde: `nrow` es el número de filas a agrupar; `width` es el ancho de la columna; y `vmove` sirve para subir o bajar el texto (opcional). Esta orden funciona de forma similar a `\multicolumn`, pero para filas. Nótese que el formato `m{Ancho}` nos era de utilidad para centrar texto en el caso de columnas vecinas de diferente altura; `multirow` es adecuado para centrar texto con respecto a *varias* columnas vecinas. Por ejemplo:

```
\begin{tabular}{|l|r|r|}
\hline \multirow{4}{1.8cm}{Planeta}
& \multicolumn{2}{p{3.8cm}|}%
{\centering Distancia al sol \\
(millones de km)}\tabularnewline \cline{2-3}
& \multicolumn{1}{p{1.7cm}|}%
{\centering Maxima \\ distancia}
& \multicolumn{1}{p{1.7cm}|}%
{\centering Minima \\ distancia}
\tabularnewline \hline
Mercurio & 69.4 & 46.8\\
Venus & 109.0 & 107.6\\
Tierra & 152.6 & 147.4\\
\hline
\end{tabular}
```

Planeta	Distancia al sol (millones de km)	
	Maxima distancia	Minima distancia
Mercurio	69.4	46.8
Venus	109.0	107.6
Tierra	152.6	147.4

Se aprecia que la entrada “Planeta” está justificada a la izquierda. Esto se debe a que el *comando* `\multirowsetup`, que determina el modo en el que aparece el texto de `\multirow`, está definido por defecto como `\raggedright`. Redefiniéndolo con `\renewcommand*{\multirowsetup}{\centering}` se consigue un texto centrado:

```
\renewcommand{\multirowsetup}{\centering}
\begin{tabular}{|l|r|r|}
\hline \multirow{4}{1.8cm}{Planeta}
& \multicolumn{2}{p{3.8cm}|}%
{\centering Distancia al sol \\
(millones de km)}\tabularnewline \cline{2-3}
& \multicolumn{1}{p{1.7cm}|}%
{\centering Maxima \\ distancia}
& \multicolumn{1}{p{1.7cm}|}%
{\centering Minima \\ distancia}
\tabularnewline \hline
Mercurio & 69.4 & 46.8\\
Venus & 109.0 & 107.6\\
Tierra & 152.6 & 147.4\\
\hline
\end{tabular}
```

Planeta	Distancia al sol (millones de km)	
	Maxima distancia	Minima distancia
Mercurio	69.4	46.8
Venus	109.0	107.6
Tierra	152.6	147.4

3. El paquete `hhline`

El paquete `hhline` define el comando `\hhline`, que produce rayas dobles ó simples, y añade capacidades para producir intersecciones bien construidas entre líneas horizontales y verticales. El comando se utiliza como:

```
\hhline{ColumnasEIntersecciones}
```

donde la especificación `ColumnasEIntersecciones` se hace utilizando los siguientes elementos:

- \equiv Una raya horizontal doble del ancho de una columna.
- $-$ Una raya horizontal simple del ancho de una columna.
- \sim Una columna sin raya horizontal.
- \parallel Una raya vertical que corta a una horizontal (simple ó doble).
- \vdots Una raya vertical que es partida por una horizontal doble. Detrás ó delante debe haber $=$.
- $\#$ Dos rayas verticales que cortan a una horizontal doble.
- t La semiparte superior de una horizontal doble.
- b La semiparte inferior de una horizontal doble.
- $*$ Podemos usar la abreviatura: $\ast{3}{\equiv\#}$ (por ejemplo), que se expande como $\equiv\#\equiv\#$ (ésto también es válido para el entorno tabular básico).

Los elementos anteriores se utilizan agrupados, para definir entrelazamientos con las rayas verticales. Por ejemplo:

- |t: → Esquina superior izquierda del cruce de dos rayas dobles (\ulcorner)
- :t| → Como el anterior, pero para la esquina superior derecha (\urcorner)
- |b: → Análogo, para esquina inferior izquierda (\llcorner)
- :b| → Análogo, para esquina inferior derecha (\lrcorner)
- |: → Una raya vertical seguida de otra vertical que se cruza con una horizontal doble
- :| → Una raya vertical, que se ha cruzado con una horizontal doble, seguida de otra vertical
- :: → Enlace sin cortes entre dos rayas dobles
- || → Raya vertical doble que no es atravesada por las rayas horizontales
- # → Corte de rayas dobles

El ejemplo siguiente ilustra las diferentes posibilidades de intersección:

```
\begin{tabular}{||cc||c|c||}
\hline|t:==:t:==:t|}
a&b&c&d\\
\hline|:==:|~|~||}
1&2&3&4\\
\hline{#==#~|#}
i&j&k&l\\
\hline{||--||--||}
w&x&y&z\\
\hline{|b:==:b:==:b|}
\end{tabular}
```

a	b	c	d
1	2	3	4
i	j	k	l
w	x	y	z

4. Tablas con color: el paquete colortbl

El objetivo del paquete `colortbl` es dar color al fondo de las tablas y a las rayas de separación. Este paquete basa su funcionamiento en los paquetes `color` y `array`, que se cargan automáticamente al cargar `colortbl` (podemos, por tanto, incluir en `colortbl` las opciones de controlador propias de `color`).

Para dar color a filas y columnas, podemos utilizar los comandos:

- `\columncolor[Modelo]{Color}[SepIzq][SepDer]` (columnas)
- `\rowcolor[Modelo]{Color}[SepIzq][SepDer]` (filas)

Debemos hacer aquí un pequeño inciso para explicar la sintaxis extendida del comando `\color`. Podemos utilizar simplemente el comando `\color{NombreColor}`, que permite utilizar hasta 68 colores predefinidos (con la opción `usenames` del paquete `color`), ó contruir nuestros propios colores a medida mediante el comando `\color[Modelo]{Especificación-de-color}`. Podemos elegir, para `Modelo`, entre las opciones:

- `rgb` → Especificación-de-color debe ser una terna de números entre 0 y 1, cada uno de los cuales especifica la cantidad relativa de rojo, verde y azul en la mezcla
- `cmyk` → Análogo, pero utilizando 4 números que representan valores de cian, magenta, amarillo y negro
- `gray` → Un valor entre 0 y 1, especificando la proximidad al blanco ó negro del tono de gris

Todas éstas posibilidades son extensibles a los comandos `\columncolor` y `\rowcolor`, por tanto. Los argumentos (optativos) `[SepIzq]` y `[SepDer]` indican las longitudes (a izquierda y derecha, respectivamente) que el fondo de color debe exceder del espacio ocupado por el texto de la celda de la tabla. Por defecto, su valor es `\tabcolsep`, lo cual quiere decir que si no se incluyen, el fondo de color ocupará toda la celda.

Cada una de las dos versiones, para fila ó columna, debe utilizarse de forma diferente:

- `\columncolor` se debe colocar como argumento del especificador de columna `>\{...\}` del paquete `array` ó en la definición de un nuevo tipo de columna a través de `\newcolumntype`. También podemos incluirlo dentro de un comando `\multicolumn` (para rellenar una sola celda, por ejemplo), pero siempre utilizando `>\{...\}` en el argumento de `\multicolumn`.
- `\rowcolor` debe aparecer SIEMPRE al comienzo de un fila. Debe tenerse en cuenta de que si entrecruzamos declaraciones de color en filas y columnas, la declaración `\rowcolor` prevalece, ya que es la última en ser asignada.

Finalmente, el color de las líneas de separación se controla mediante los comandos:

- `\arrayrulecolor[Modelo]{color}` que fija el color de las líneas
- `\doublerulesepcolor[Modelo]{color}` que fija el color del relleno entre líneas dobles

ADVERTENCIA FINAL: Sólo está asegurada la obtención del resultado correcto en el fichero final .pdf (los visores .dvi pueden dar resultados algo extraños).

Ejemplos:

```
\begin{tabular}{|>\columncolor[gray]{0.9}}1%
>\color{white}\columncolor[gray]{0.6}}r|}
alfa & beta \ \ gamma & delta
\end{tabular} \hspace{1cm}
\begin{tabular}%
{|>\color{yellow}%
\columncolor[gray]{0.7}[0pt]}1|}
>\columncolor{yellow}[0.5\tabcolsep]}r|}
alfa & beta \ \ gamma & delta
\end{tabular}
```

alfa	beta	alfa	beta
gamma	delta	gamma	delta

```

\begin{tabular}{||c||c||}
\hline{|t:=:t:=:t|}
\rowcolor{red} alfa & beta \\
\hline{|:=:|=|}
\rowcolor{green} gamma & delta \\
\hline{|b:=:b:=:b|}
\end{tabular}}

```

alfa	beta
gamma	delta

5. Paquete longtable

Define el entorno **longtable**, que permite construir tablas que abarquen varias páginas.

Ejemplo:

```

\begin{longtable}{|c|c|c|}
\hline
NIF & Nota & Calificación \\ \hline
45323459J & 7.2 & (NT) \\
etc...
71123261J & 9.5 & (MH) \\ \hline
\end{longtable}

```

NIF	Nota	Calificación
45323459J	7.2	(NT)
71283755Q	6.5	(AP)
82410104V	—	(NP)
61142427T	7.1	(NT)
22413133X	9.0	(SB)
21152619A	—	(NP)
41520126Q	8.0	(NT)
21151160Q	3.6	(SS)
41342139A	7.2	(NT)
19749402V	6.0	(AP)
72186899A	5.5	(AP)
09456969B	7.1	(NT)
51138427W	9.1	(SB)
11335720P	—	(NP)
71276261C	7.0	(NT)
21342465B	6.5	(AP)
12442545F	7.5	(NT)
91451213L	4.5	(SS)
19329402V	6.0	(AP)
72101499A	5.5	(AP)
09303969B	7.1	(NT)
52138427W	9.1	(SB)
11937720P	—	(NP)
71346261C	7.0	(NT)

21936765B	6.5	(AP)
12356545F	7.5	(NT)
91341213L	4.5	(SS)
19349402V	6.0	(AP)
72101099A	5.5	(AP)
09305969B	7.1	(NT)
51138427W	9.1	(SB)
11137720P	—	(NP)
71276261C	7.0	(NT)
21932465B	6.5	(AP)
12392545F	7.5	(NT)
91941213L	4.5	(SS)
71147749P	3.0	(SS)
71126520J	5.0	(AP)
73140778F	7.2	(NT)
71123261J	9.5	(MH)

Veamos ahora como colocar una cabecera común a todas las páginas, otra especial para el principio de la tabla, un pie de tabla general y otro especial para el fin de la tabla (además de leyendas); los textos, ó formatos de encabezamiento de tabla (multicolumn, columnas normales, etc...) han de ser colocados respectivamente antes de las marcas `\endfirsthead`, `\endhead`, `\endfoot` y `\endlastfoot`. Todo esto se ilustra en el siguiente ejemplo:

(Más detalles en la documentación del paquete)

```

\begin{longtable}{|c|c|c|}
\caption{Notas finales del primer ejercicio} \\
\hline \multicolumn{3}{|c|}{Notas del grupo C} \\
\hline NIF & Nota & Calificación \\
\endfirsthead
\caption{Continuación de la tabla} \\
\hline
NIF & Nota & Calificación \\
\endhead
\multicolumn{3}{c}{(sigue en la página siguiente)} \\
\endfoot
\multicolumn{3}{c}{(Fin de la tabla)} \\
\endlastfoot
45323459J & 7.2 & (NT) \\
etc...
71123261J & 9.5 & (MH) \\
\end{longtable}

```

Cuadro 2: Notas finales del primer ejercicio

Notas del grupo C		
NIF	Nota	Calificación
45323459J	7.2	(NT)
71283755Q	6.5	(AP)
82410104V	—	(NP)
61142427T	7.1	(NT)
22413133X	9.0	(SB)
21152619A	—	(NP)
41520126Q	8.0	(NT)
21151160Q	3.6	(SS)
41342139A	7.2	(NT)
19749402V	6.0	(AP)
72186899A	5.5	(AP)
09456969B	7.1	(NT)
51138427W	9.1	(SB)
11335720P	—	(NP)
71276261C	7.0	(NT)
21342465B	6.5	(AP)
12442545F	7.5	(NT)
91451213L	4.5	(SS)
19329402V	6.0	(AP)
72101499A	5.5	(AP)
09303969B	7.1	(NT)
52138427W	9.1	(SB)
11937720P	—	(NP)
71346261C	7.0	(NT)
21936765B	6.5	(AP)
12356545F	7.5	(NT)
91341213L	4.5	(SS)
19349402V	6.0	(AP)
72101099A	5.5	(AP)
09305969B	7.1	(NT)
51138427W	9.1	(SB)
11137720P	—	(NP)
71346261C	7.0	(NT)
21936765B	6.5	(AP)
19349402V	6.0	(AP)
72101099A	5.5	(AP)
09305969B	7.1	(NT)
51138427W	9.1	(SB)
11137720P	—	(NP)
71276261C	7.0	(NT)
21932465B	6.5	(AP)
12392545F	7.5	(NT)
91941213L	4.5	(SS)

(sigue en la página siguiente)

Cuadro 2: Continuación de la tabla

NIF	Nota	Calificación
71147749P	3.0	(SS)
71126520J	5.0	(AP)
73140778F	7.2	(NT)
71123261J	9.5	(MH)

(Fin de la tabla)

6. Paquete tabularx

Este paquete introduce el entorno `tabularx`, que permite construir tablas con una anchura total predeterminada; para ello se declara la anchura total como argumento del entorno, y se introduce un nuevo formato de columna “X”, equivalente al “`p{Ancho}`”, donde la anchura se calcula automáticamente de forma que la anchura final de la tabla sea la requerida; véanse los siguientes ejemplos:

```
\begin{tabularx}{8cm}{|XcXcXcX|}
\hline & test1 & & test2 & & test3 & \\
\hline & test4 & & test5 & & test6 & \\
\hline
\end{tabularx}
```

```
\begin{tabularx}{10cm}{|c|X|c|X|}
\hline
\multicolumn{2}{|c|}{Ejemplo de multicolumna} & TRES & CUATRO \\
\hline uno & La longitud de esta columna depende de la anchura total
de la tabla & dos & Esta otra columna tambien variara de anchura, de
la misma forma que lo hace la segunda
\footnote{Se pueden colocar notas al pie con tabularx!} \\
\hline
\end{tabularx}
```

test1	test2	test3
test4	test5	test6

Ejemplo de multicolumna		TRES	CUATRO
uno	La longitud de esta columna depende de la anchura total de la tabla	dos	Esta otra columna tambien variara de anchura, de la misma forma que lo hace la segunda ¹

(Ensachar la segunda tabla, y ver el efecto en los párrafos)

¹Se pueden colocar notas al pie con `tabularx!`

7. Paquete slashbox

```
\noindent
\begin{tabular}{|c||*{5}{c|}}\hline
\backslashbox{Sala}{Fecha} & 31/5 & 1/6 & 2/6 &
3/6 & 4/6 \\ \hline
Aula A4 & & & & & \\ \hline
Aula III & & & & & \\ \hline
Seminario & & & & & \\ \hline
\end{tabular}
```

Sala \backslash Fecha	31/5	1/6	2/6	3/6	4/6
Aula A4					
Aula III					
Seminario					

8. Tabla con colores guía

El siguiente ejemplo ilustra las posibilidades de \LaTeX para programar utilidades interesantes; para una tabla con numerosas columnas, es útil introducir un color de fondo en la mitad de las filas para hacer más fácil la lectura. Definimos un nuevo contador `rowparity`, que lleve la cuenta de la paridad de las filas de la tabla, y a continuación un comando `\X` que en vez de simplemente cambiar de línea efectúe las siguientes acciones:

- Sume 1 al contador `rowparity`
- Salte de línea con `\tabularnewline`
- Decida en función de la paridad de `rowparity` (condicional `ifodd`) si añadir el comando `\rowcolor[gray]{0.9}` ó no

```
\newcounter{rowparity}%
\newcommand{\X}{%
\addtocounter{rowparity}{1}%
\tabularnewline%
\ifodd\value{rowparity}%
\rowcolor[gray]{0.9}\fi}

{\setlength{\arrayrulewidth}{1pt}
\begin{tabular}{|c|ccc|}
\hline
Hola & Pepe & Juan & Jesus \\ \hline
Hola & Pepe & Juan & Jesus \X
\hline
\end{tabular}
```

Hola	Pepe	Juan	Jesus
Hola	Pepe	Juan	Jesus
Hola	Pepe	Juan	Jesus
Hola	Pepe	Juan	Jesus
Hola	Pepe	Juan	Jesus
Hola	Pepe	Juan	Jesus
Hola	Pepe	Juan	Jesus