

# Códigos QR

Javier Luque Ordóñez

javluqord@yahoo.es

## QR. Conceptos principales

### ¿Qué son los códigos QR?

Un código QR (*Quick Response code*, código de respuesta rápida) es un método de representación y almacenamiento de información en una matriz de puntos bidimensional.

Esta simbología en 2D tiene su origen en 1994 en Japón, cuando la empresa Denso Wave, subsidiaria de Toyota, la desarrolla para mejorar la trazabilidad del proceso de fabricación de vehículos. Fue diseñada con el objetivo principal de conseguir una decodificación sencilla y rápida de la información contenida.

Muy comunes en Japón y cada vez más extendidos a nivel mundial (gracias a su empleo para codificar URLs de internet y a las aplicaciones de decodificación existentes para teléfonos móviles con cámara), se caracterizan por disponer de 3 cuadrados en las esquinas, que facilitan el proceso de lectura.



Figura 1. Ejemplo de código QR.

### Estandarización de los códigos QR

Denso Wave desarrolla las especificaciones para definir los códigos QR, siendo la propietaria de los derechos de patente sobre éstos (US 5726435 en Estados Unidos; JP 2938338 en Japón, EP0672994B1 en Europa).

Para favorecer su aceptación y uso, esta empresa japonesa ha decidido no ejercer dichos derechos de patente y hacer públicas dichas especificaciones, que se han convertido en documentos estandarizados en ISO, disponibles para cualquier persona u organización.

El término “QR Code” © es una marca registrada por Denso Wave en Japón y otros países. Esta marca registrada se aplica únicamente a la citación del término en una documentación, no a la citación en la información contenida en los códigos QR. Así, el empleo

de códigos QR no requiere por tanto de ninguna licencia o autorización previa, ni del pago de ninguna tasa.

El proceso de estandarización de los códigos QR ha sido el siguiente:

- Octubre de 1997: aprobado como estándar de AIMI (*Automatic Identification Manufacturers International*): ISS-QR Code. Define los códigos "QR Code Model 1".
- Marzo de 1998: aprobado como estándar de JEIDA (*Japanese Electronic Industry Development Association*): JEIDA-55.
- Enero de 1999: aprobado como estándar de JIS (*Japanese Industrial Standards*): JIS X 0510.
- Junio de 2000: aprobado como estándar internacional de ISO/IEC (*International Organization for Standardization/International Electrotechnical Commission*): ISO/IEC 18004:2000. Define los códigos "QR Code Model 2".
- Noviembre de 2004: añadida la variante "Micro QR" al estándar.
- Septiembre de 2006: actualización a ISO/IEC 18004:2006. Define los códigos "QR Code 2005".

La denominación oficial de la última versión del estándar es "*Information technology – Automatic identification and data capture techniques – QR Code 2005 bar code symbology specification*". La versión de JIS del estándar es gratuita y está publicada en japonés, mientras que la versión ISO es de pago y está publicada en inglés.

El estándar especifica las características de la simbología; los métodos de codificación de datos; los formatos de símbolo; las características dimensionales; los métodos de corrección de errores; los algoritmos de referencia para decodificación; los requisitos de calidad

del proceso; los parámetros de aplicación elegibles por el usuario; y un listado de anexos informativos.

Para ello se apoya también en otros estándares existentes de JIS y de ISO/IEC, principalmente relacionados con conjuntos de caracteres y con técnicas de captura e identificación de datos.

### Ventajas y características principales de los códigos QR

Un código QR consiste en un conjunto de puntos negros (u oscuros) ubicados según una determinada codificación en un patrón cuadrado sobre fondo blanco (o claro). Sus características y ventajas principales se muestran a continuación:

- Alta capacidad de codificación de datos: hasta 7.089 caracteres numéricos o 2.953 bytes.
- Decodificación sencilla y a alta velocidad: desde lectores hardware o aplicaciones software.
- Mayor densidad de datos y poco espacio necesario para impresión del código: en torno a 1/10 respecto al código de barras tradicional.
- Adaptabilidad del código a los datos: tamaño en puntos de la matriz según contenido almacenado.
- Soporte de múltiples lenguajes y códigos de caracteres: numéricos, alfanuméricos, binarios, escrituras Kanji, Kana, Hiragana, o cualquier formato de datos mediante la definición de extensiones.
- Permite otras variantes como Micro QR o hasta 16 estructuras añadidas.
- Capacidad de corrección de errores: restauración de hasta un 30% de los datos.

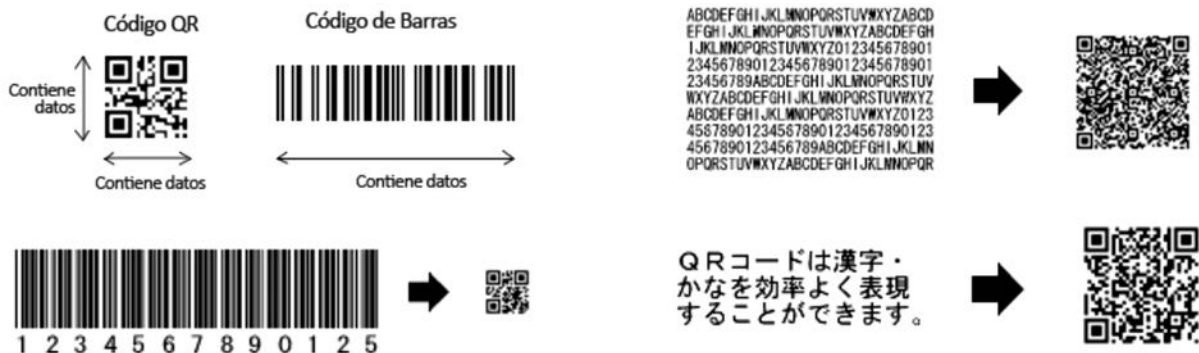


Figura 2. Algunas características de los códigos QR.

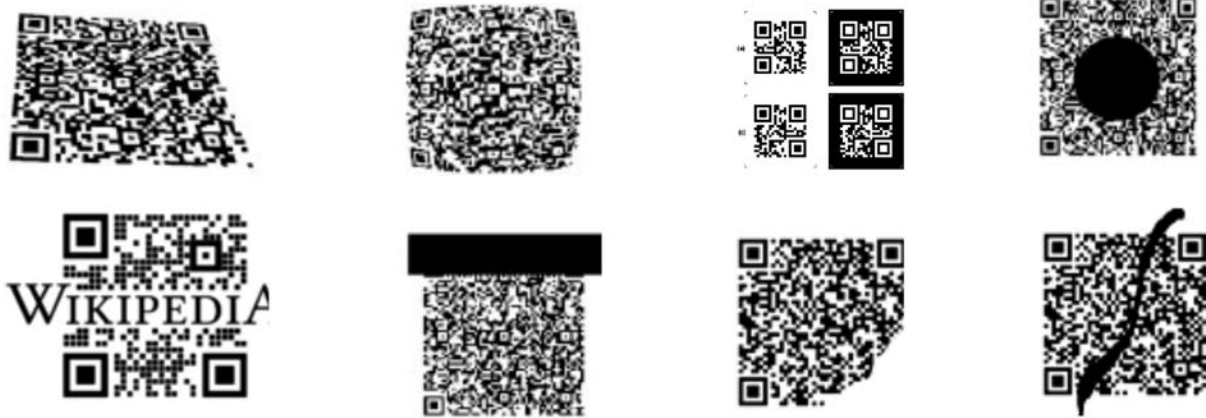


Figura 3. Restauración de códigos QR. Todos estos códigos son decodificables.

- Aplicación de máscaras a los datos: mayor diferenciación de niveles claros y oscuros.
- Facilidad de lectura del código: independencia de la orientación (decodificación en 360°); detección de distorsión; inversión de umbrales; estructuras en espejo.
- Confidencialidad: facilidad de cifrado del código QR.
- Popularización de su uso gracias a diversos factores: publicación de especificaciones; gratuidad de uso; integración con dispositivos móviles; aplicación fuera del entorno industrial; robustez; etc.

De acuerdo a la evolución del estándar, el modelo 2 de 2000 añadía principalmente los patrones de alineamiento al modelo 1 de 1997. El modelo “QR Code 2005”, base de la versión revisada del estándar ISO/IEC 18004:2006 es muy similar al modelo 2 de 2000, añadiendo las funcionalidades de imagen en espejo, la reflectancia (intercambio de blancos y negros) y la capacidad de especificar conjuntos nuevos de caracteres. Los símbolos modelo 2 son así completamente compatibles con los símbolos actuales.

## III Estructuras y codificación de datos en los códigos QR

### Conceptos básicos de una estructura de código QR

La representación bidimensional de un código QR se denomina símbolo. Cada símbolo está formado por cuadros negros o blancos llamados módulos, que

representan el 0 y el 1 binario respectivamente. Los módulos están ubicados en una estructura cuadrada, que contiene dos grandes bloques de módulos: los patrones de función y la región de codificación.

En cada símbolo existen un conjunto de módulos que no contienen datos codificados, sino información necesaria para su decodificación. Son los denominados patrones de función, y existen de varios tipos:

- Patrón de localización: patrón de función que existe por triplicado en el símbolo, situado en las esquinas superiores y la inferior izquierda. Sirven para calcular la orientación rotacional del símbolo.
- Patrón de alineamiento: secuencia alternada de módulos blancos y negros que ayuda a calcular las coordenadas de los módulos del símbolo.
- Patrón temporizador: patrón de función que permite resincronizar las coordenadas de mapeo del símbolo ante posibles distorsiones moderadas.

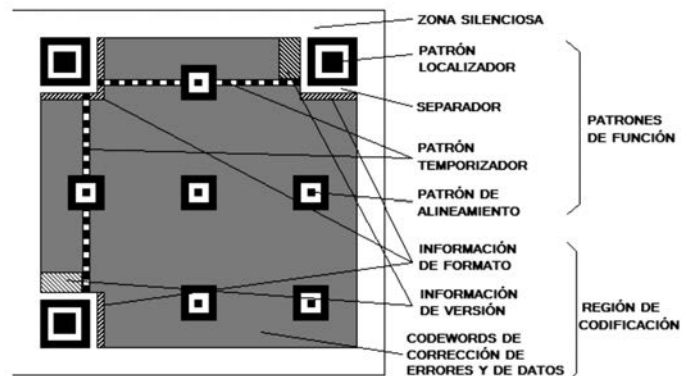


Figura 4. Estructura genérica de un símbolo.

- Separador: patrón de función formado por módulos blancos, cuyo ancho es de un módulo y que separa los patrones localizadores del resto del símbolo.

Los datos codificados, por su parte, se agrupan en conjuntos de 8, denominados *codewords*, que adoptan diversas formas según su ubicación en la estructura. La región de codificación es la región del símbolo no ocupada por patrones de función y sí por

*codewords* de datos y de corrección de errores, así como por la información de formato y versión.

La información de formato es un patrón codificado que contiene información sobre el grado de corrección de errores con el que se han codificado los datos de la región de codificación y el tipo de máscara que se les ha aplicado. La información de versión, por su parte, es un patrón codificado que contiene información que indica la versión del símbolo.

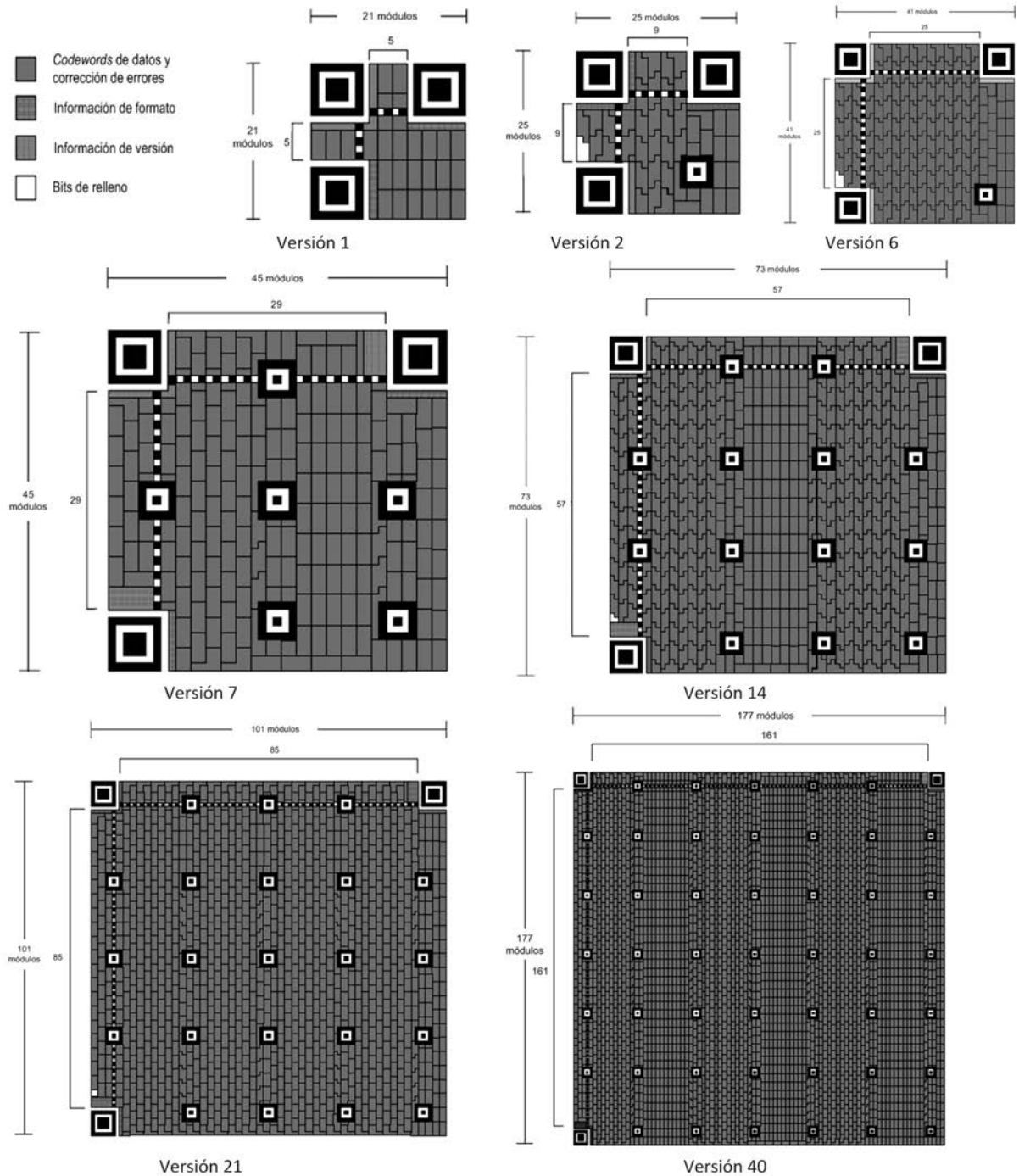


Figura 5. Estructuras genéricas de diversas versiones de símbolos QR.

Asimismo, para poder delimitar correctamente los bordes de cada símbolo, se requiere de una banda de anchura 4 módulos (zona silenciosa) que debe estar en blanco (en negro si hay reflectancia inversa).

El tamaño del símbolo se denomina versión. Existen 40 versiones: la versión 1 tiene 21x21 módulos, la versión 2 tiene 25x25 módulos, y así sucesivamente (incrementos de 4 módulos por lado en cada versión) hasta la versión 40, que contiene 177x177 módulos. Cada versión tiene un número y disposición diferente de *codewords* de datos y corrección de errores, así como de patrones de función.

Así, algunas versiones necesitan de módulos (bits) de relleno y otras no. Las versiones inferiores a la 7 no disponen de información de versión, y la versión 1 no incluye patrón de alineamiento. Sin embargo todas las versiones tienen 3 patrones localizadores, 2 patrones temporizadores, 3 separadores y la formación de formato por duplicado.

### Corrección de errores

Los códigos QR emplean codificación de errores basada en algoritmos de *Reed-Solomon*, generando

un conjunto de *codewords* de corrección de errores (ECC, *Error Correction Codewords*) que se añaden a los de datos aportando redundancia. Los algoritmos *Reed-Solomon* fueron desarrollados inicialmente para combatir el ruido de comunicaciones en los satélites artificiales y sondas espaciales, y hoy día se emplean por ejemplo también en la codificación de CDs de música. Permiten corrección a nivel de byte y son adecuados para errores de ráfaga. Existen 4 niveles de corrección de errores en los símbolos QR:

- L (*Low*). Puede corregir hasta el 7% de los *codewords* de datos del símbolo.
- M (*Medium*). Puede corregir hasta el 15% de los *codewords* de datos del símbolo.
- Q (*Quality*). Puede corregir hasta el 25% de los *codewords* de datos del símbolo.
- H (*High*). Puede corregir hasta el 30% de los *codewords* de datos del símbolo.

Debido al diseño de estos algoritmos y al uso de *codewords* de 8 bits en los símbolos QR, un bloque individual de código no puede ser mayor a 255 *codewords* de longitud. Esto hace que sea necesario partir el mensaje en múltiples bloques, y aplicar el algoritmo a los *codewords* de datos de cada bloque.

Versión	Codewords	Nivel EC	ECC	Bloques EC	Codewords/bloque (*)	Versión	Codewords	Nivel EC	ECC	Bloques EC	Codewords/bloque (*)
1	26	L	7	1	(26,19,2)	21	1.156	L	224	4 4	(144,116,14) (145,117,14)
		M	10	1	(26,16,4)			M	442	17	(68,42,13)
		Q	13	1	(26,13,6)			Q	644	17 6	(50,22,14) (51,23,14)
		H	17	1	(26,9,8)			H	750	19 6	(46,16,15) (47,17,15)
2	44	L	10	1	(44,34,4)	22	1.258	L	252	2 7	(139,111,14) (140,112,14)
		M	16	1	(44,28,8)			M	476	17	(74,46,14)
		Q	22	1	(44,22,11)			Q	690	7 16	(54,24,15) (55,25,15)
		H	28	1	(44,16,14)			H	816	34	(37,13,12)
3	70	L	15	1	(70,55,7)	23	1.364	L	270	4 5	(151,121,15) (152,122,15)
		M	26	1	(70,44,13)			M	504	4 14	(75,47,14) (76,48,14)
		Q	36	2	(35,17,9)			Q	750	11 14	(54,24,15) (55,25,15)
		H	44	2	(35,13,11)			H	900	16 14	(45,15,15) (46,16,15)
4	100	L	20	1	(100,80,10)	24	1.474	L	300	6 4	(147,117,15) (148,118,15)
		M	36	2	(50,32,9)			M	560	6 14	(73,45,14) (74,46,14)
		Q	52	2	(50,24,13)			Q	810	11 16	(54,24,15) (55,25,15)
		H	64	4	(25,9,8)			H	960	30 2	(46,16,15) (47,17,15)
5	134	L	26	1	(134,108,13)	25	1.588	L	312	8 4	(132,106,13) (133,107,13)
		M	48	2	(67,43,12)			M	588	8 13	(75,47,14) (76,48,14)
		Q	72	2 2	(33,15,9) (34,16,9)			Q	870	7 22	(54,24,15) (55,25,15)
		H	88	2 2	(33,11,11) (34,12,11)			H	1050	22 13	(45,15,15) (46,16,15)
6	172	L	36	2	(86,68,9)	26	1.706	L	336	10 2	(142,114,14) (143,115,14)
		M	64	4	(43,27,8)			M	644	19 4	(74,46,14) (75,47,14)
		Q	96	4	(43,19,12)			Q	952	28 6	(50,22,14) (51,23,14)
		H	112	4	(43,15,14)			H	1110	33 4	(46,16,15) (47,17,15)

7	196	L	40	2	(98,78,10)	27	1.828	L	360	8 4	(152,122,15) (153,123,15)
		M	72	4	(49,31,9)			M	700	22 3	(73,45,14) (74,46,14)
		Q	108	2 4	(32,14,9) (33,15,9)			Q	1 020	8 26	(53,23,15) (54,24,15)
		H	130	4 1	(39,13,13) (40,14,13)			H	1 200	12 28	(45,15,15) (46,16,15)
8	242	L	48	2	(121,97,12)	28	1.921	L	390	3 10	(147,117,15) (148,118,15)
		M	88	2 2	(60,38,11) (61,39,11)			M	728	3 23	(73,45,14) (74,46,14)
		Q	132	4 2	(40,18,11) (41,19,11)			Q	1 050	4 31	(54,24,15) (55,25,15)
		H	156	4 2	(40,14,13) (41,15,13)			H	1 260	11 31	(45,15,15) (46,16,15)
9	292	L	60	2	(146,116,15)	29	2.051	L	420	7 7	(146,116,15) (147,117,15)
		M	110	3 2	(58,36,11) (59,37,11)			M	784	21 7	(73,45,14) (74,46,14)
		Q	160	4 4	(36,16,10) (37,17,10)			Q	1 140	1 37	(53,23,15) (54,24,15)
		H	192	4 4	(36,12,12) (37,13,12)			H	1 350	19 26	(45,15,15) (46,16,15)
10	346	L	72	2 2	(86,68,9) (87,69,9)	30	2.185	L	450	5 10	(145,115,15) (146,116,15)
		M	130	4 1	(69,43,13) (70,44,13)			M	812	19 10	(75,47,14) (76,48,14)
		Q	192	6 2	(43,19,12) (44,20,12)			Q	1 200	15 25	(54,24,15) (55,25,15)
		H	224	6 2	(43,15,14) (44,16,14)			H	1 440	23 25	(45,15,15) (46,16,15)
11	404	L	80	4	(101,81,10)	31	2.323	L	480	13 3	(145,115,15) (146,116,15)
		M	150	1 4	(80,50,15) (81,51,15)			M	868	2 29	(74,46,14) (75,47,14)
		Q	224	4 4	(50,22,14) (51,23,14)			Q	1 290	42 1	(54,24,15) (55,25,15)
		H	264	3 8	(36,12,12) (37,13,12)			H	1 530	23 28	(45,15,15) (46,16,15)
12	466	L	96	2 2	(116,92,12) (117,93,12)	32	2.465	L	510	17	(145,115,15)
		M	176	6 2	(58,36,11) (59,37,11)			M	924	10 23	(74,46,14) (75,47,14)
		Q	260	4 6	(46,20,13) (47,21,13)			Q	1 350	10 35	(54,24,15) (55,25,15)
		H	308	7 4	(42,14,14) (43,15,14)			H	1 620	19 35	(45,15,15) (46,16,15)
13	532	L	104	4	(133,107,13)	33	2.611	L	540	17 1	(145,115,15) (146,116,15)
		M	198	8 1	(59,37,11) (60,38,11)			M	980	14 21	(74,46,14) (75,47,14)
		Q	288	8 4	(44,20,12) (45,21,12)			Q	1 440	29 19	(54,24,15) (55,25,15)
		H	352	12 4	(33,11,11) (34,12,11)			H	1 710	11 46	(45,15,15) (46,16,15)
14	581	L	120	3 1	(145,115,15) (146,116,15)	34	2.761	L	570	13 6	(145,115,15) (146,116,15)
		M	216	4 5	(64,40,12) (65,41,12)			M	1 036	14 23	(74,46,14) (75,47,14)
		Q	320	11 5	(36,16,10) (37,17,10)			Q	1 530	44 7	(54,24,15) (55,25,15)
		H	384	11 5	(36,12,12) (37,13,12)			H	1 800	59 1	(46,16,15) (47,17,15)
15	655	L	132	5 1	(109,87,11) (110,88,11)	35	2.876	L	570	12 7	(151,121,15) (152,122,15)
		M	240	5 5	(65,41,12) (66,42,12)			M	1 064	12 26	(75,47,14) (76,48,14)
		Q	360	5 7	(54,24,15) (55,25,15)			Q	1 590	39 14	(54,24,15) (55,25,15)
		H	432	11 7	(36,12,12) (37,13,12)			H	1 890	22 41	(45,15,15) (46,16,15)
16	733	L	144	5 1	(122,98,12) (123,99,12)	36	3.034	L	600	6 14	(151,121,15) (152,122,15)
		M	280	7 3	(73,45,14) (74,46,14)			M	1 120	6 34	(75,47,14) (76,48,14)
		Q	408	15 2	(43,19,12) (44,20,12)			Q	1 680	46 10	(54,24,15) (55,25,15)
		H	480	3 13	(45,15,15) (46,16,15)			H	1 980	2 64	(45,15,15) (46,16,15)
17	815	L	168	1 5	(135,107,14) (136,108,14)	37	3.196	L	630	17 4	(152,122,15) (153,123,15)
		M	308	10 1	(74,46,14) (75,47,14)			M	1 204	29 14	(74,46,14) (75,47,14)
		Q	448	1 15	(50,22,14) (51,23,14)			Q	1 770	49 10	(54,24,15) (55,25,15)
		H	532	2 17	(42,14,14) (43,15,14)			H	2 100	24 46	(45,15,15) (46,16,15)
18	901	L	180	5 1	(150,120,15) (151,121,15)	38	3.362	L	660	4 18	(152,122,15) (153,123,15)
		M	338	9 4	(69,43,13) (70,44,13)			M	1 260	13 32	(74,46,14) (75,47,14)
		Q	504	17 1	(50,22,14) (51,23,14)			Q	1 860	48 14	(54,24,15) (55,25,15)
		H	588	2 19	(42,14,14) (43,15,14)			H	2 220	42 32	(45,15,15) (46,16,15)
19	991	L	196	3 4	(141,113,14) (142,114,14)	39	3.532	L	720	20 4	(147,117,15) (148,118,15)
		M	364	3 11	(70,44,13) (71,45,13)			M	1 316	40 7	(75,47,14) (76,48,14)
		Q	546	17 4	(47,21,13) (48,22,13)			Q	1 950	43 22	(54,24,15) (55,25,15)
		H	650	9 16	(39,13,13) (40,14,13)			H	2 310	10 67	(45,15,15) (46,16,15)
20	1.085	L	224	3 5	(135,107,14) (136,108,14)	40	3.706	L	750	19 6	(148,118,15) (149,119,15)
		M	416	3 13	(67,41,13) (68,42,13)			M	1 372	18 31	(75,47,14) (76,48,14)
		Q	600	15 5	(54,24,15) (55,25,15)			Q	2 040	34 34	(54,24,15) (55,25,15)
		H	700	15 10	(43,15,14) (44,16,14)			H	2 430	20 61	(45,15,15) (46,16,15)

(\*) (a, b, c): a= total de codewords en el bloque; b: total de codewords de datos en el bloque; c: capacidad de corrección de errores

Figura 6. Bloques de corrección de errores.

Así, por cada bloque existen *codewords* de datos y de corrección de errores. En la figura 6 se muestran el número de bloques y *codewords* de datos y ECC para cada bloque, versión y nivel de corrección.

Para generar los ECC, cada *codeword* de datos se emplea como coeficiente de un término de un polinomio, definido en la especificación QR. Para mejorar la eficiencia del algoritmo, los *codewords* de datos son entremezclados (secuencialmente por bloques), para evitar propagaciones de error. Los ECC se ubican a continuación de los *codewords* de datos (y también son entremezclados secuencialmente por bloques).

Cada símbolo se denomina de acuerdo a su versión y a su nivel de corrección de errores (1-Q, 23-L, 40-H, 17-M, etc.). A mayor nivel de corrección de errores, menor capacidad de datos tiene el símbolo. El nivel M es el más empleado, y los niveles Q y H se usan principalmente en entornos industriales.

Gracias a este método de corrección de errores, es posible crear símbolos QR con errores intencionados y que puedan aún ser decodificados correctamente. Esto es empleado a nivel comercial para personalización de la imagen del símbolo QR, incluyendo por ejemplo colores y logos.

### Información de formato e información de versión

La información de formato es una secuencia de 15 módulos (bits), de los que 5 contienen datos y los otros 10 se emplean para corrección de errores en los 5 primeros, mediante un código BCH (15,5). De los 5 bits de datos, los dos primeros indican el nivel de corrección de error usado (L = 01; M = 00; Q = 11; H = 10) y los otros tres indican el patrón de la máscara de datos usada. Una vez calculados los 15 bits se les debe aplicar mediante la operación lógica XOR la máscara "10101000010010", para evitar

una información de formato compuesta solo por bits a 0. La información de formato se ubica por duplicado (alrededor de los patrones de localización) en el símbolo QR, ya que su decodificación es esencial para la correcta decodificación del símbolo.

La información de versión solo se aplica a los símbolos de versión 7 o superior. Está formada por 18 bits, 6 de ellos contienen datos y los otros 12 se emplean para corregir errores en los 6 primeros, mediante un código BCH (18,6). Los 6 bits de datos codifican la versión del símbolo con el bit más significativo primero (por ejemplo, "000111" es la versión 7 y "101000" es la versión 40). En este caso no es necesario enmascarar estos 18 bits, ya que no existe la situación que provoque una cadena de bits a 0. La información de versión también se ubica por duplicado, en bloques de 6x3 módulos encima del patrón de localización inferior izquierdo y a la izquierda del patrón de localización superior derecho.

### Colocación de los patrones de función en el símbolo

Los 3 patrones de localización se ubican en las esquinas superior derecha, superior izquierda e inferior izquierda del símbolo QR. Cada patrón de localización está formado por un cuadrado relleno de 3x3 módulos negros, rodeado de un cuadrado de 5x5 módulos blancos, rodeado a su vez por un cuadrado de 7x7 módulos negros. La correcta localización de estos patrones implica el cálculo de la orientación del símbolo y del tamaño de los módulos del mismo. Los bordes interiores de cada patrón de localización están rodeados por módulos blancos, que constituyen los separadores.

Existen dos patrones temporizadores, uno vertical y otro horizontal. Unen los separadores mediante módulos blancos y negros alternados (comenzando por un módulo negro), en la fila 6 y en la columna 6

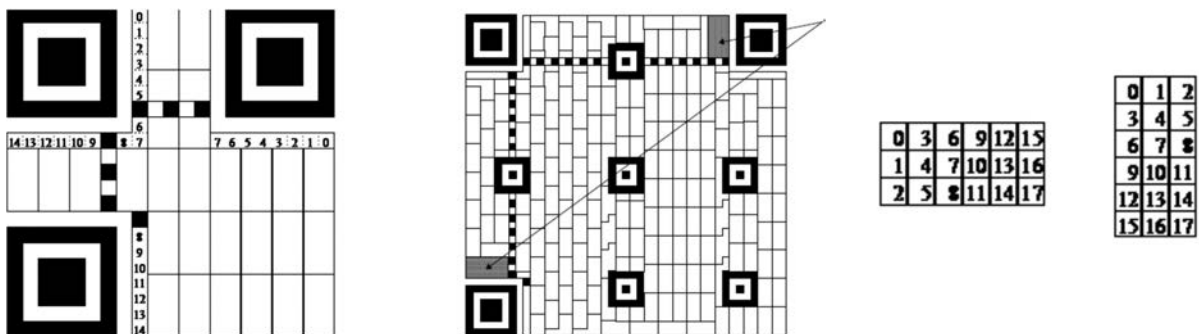


Figura 7. Colocación y ordenación de la información de formato (en versión 1) y versión (en versión 7).

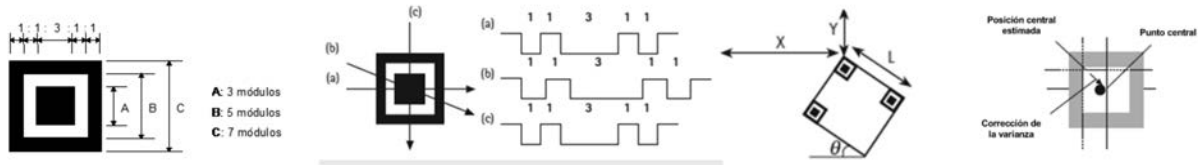


Figura 8. Determinación de los patrones de localización y orientación del símbolo.

del símbolo. Estos patrones posibilitan la determinación de la versión del símbolo y las coordenadas de los módulos.

Los patrones de alineamiento están formados por un módulo negro, rodeado de un cuadrado de 3x3 módulos blancos, rodeado a su vez por otro cuadrado de 5x5 módulos negros. Su número y ubicación en el símbolo varía según la versión.

Versión	Patrones de alineamiento	Coordenadas (fila o columna)						
1	0	-						
2	1	6	18					
3	1	6	22					
4	1	6	26					
5	1	6	30					
6	1	6	34					
7	6	6	22	38				
8	6	6	24	42				
9	6	6	26	46				
10	6	6	28	50				
11	6	6	30	54				
12	6	6	32	58				
13	6	6	34	62				
14	13	6	26	46	66			
15	13	6	26	48	70			
16	13	6	26	50	74			
17	13	6	30	54	78			
18	13	6	30	56	82			
19	13	6	30	58	86			
20	13	6	34	62	90			
21	22	6	28	50	72	94		
22	22	6	26	50	74	98		
23	22	6	30	54	78	102		
24	22	6	28	54	80	106		
25	22	6	32	58	84	110		
26	22	6	30	58	86	114		
27	22	6	34	62	90	118		
28	33	6	26	50	74	98	122	
29	33	6	30	54	78	102	126	
30	33	6	26	52	78	104	130	
31	33	6	30	56	82	108	134	
32	33	6	34	60	86	112	138	
33	33	6	30	58	86	114	142	
34	33	6	34	62	90	118	146	
35	46	6	30	54	78	102	126	150
36	46	6	24	50	76	102	128	154
37	46	6	28	54	80	106	132	158
38	46	6	32	58	84	110	136	162
39	46	6	26	54	82	110	138	166
40	46	6	30	58	86	114	142	170

Figura 9. Ubicación de los patrones de alineamiento.

## Modos de codificación de caracteres. Capacidad de los símbolos

Se denomina modo a la forma de representar un conjunto de datos como una cadena de bits. En los códigos QR, la información puede codificarse en diferentes formatos e incluso dividirse en conjuntos de diferentes tipos para lograr una mayor eficiencia. Cada uno de estos subconjuntos debe incluir al comienzo una cabecera, compuesta de un indicador de modo (4 bits) y un contador de caracteres (de longitud variable según versión).

No existen separadores entre subconjuntos ya que su tamaño y lugar de inicio están delimitados por el indicador de modo y el contador de caracteres asociado. Al final de todos los subconjuntos se inserta un terminador (que puede omitirse o abreviarse si la capacidad restante del símbolo es menor a 4 bits). En los códigos QR hay diferentes modos diferentes de codificar la información:

- Modo numérico. Dígitos (0-9). Densidad media de 10 bits para cada 3 caracteres.
- Modo alfanumérico. 45 caracteres: 0-9, A-Z y otros 9 caracteres: espacio, \$, %, :, ., \*, +, -, /. Densidad media de 11 bits para cada 2 caracteres.
- Modo byte. Código binario según se define en JIS X0208. 8 bits por carácter.
- Modo Kanji. Caracteres del alfabeto japonés según se define en Shift JIS. Densidad media de 13 bits para cada 2 caracteres.
- Modo de estructuras apiladas. Para dividir la información en varios códigos QR relacionados.
- Modo FNC1. Codificación de UCC/EAN (códigos de barras unidimensionales) o de cualquier otro estándar específico de la industria que esté aprobado por AIMI.
- ECI (Extended Channel Interpretation, interpretación de canal extendido). Permite a los flujos de datos de salida ser interpretados de forma diferente a los conjuntos de caracteres por defecto.



- Incluye 4 tipos de interpretación: conjuntos de caracteres internacionales; interpretaciones de propósito general (por ejemplo, cifrado); interpretaciones definidas por el usuario para sistemas cerrados; e información de control para los códigos apilados definidos. El protocolo ECI se encuentra completamente definido en las especificaciones AIMI.
- La interpretación por defecto es “ECI 000020”, que representa los conjuntos de caracteres JIS8 y Shift JIS (incluyen caracteres occidentales –equivalentes a ASCII– y japoneses). Si se emplea un ECI distinto al ECI por defecto se debe incluir una cabecera adicional que especifique el conjunto de caracteres a emplear (por ejemplo, el alfabeto cirílico).

Modo	Indicador de modo
ECI	0111
Númérico	0001
Alfanumérico	0010
Byte (Binario)	0100
Kanji	1000
Estructura apilada	0011
FNC1	0101 (Primera posición)
	1001 (Segunda posición)
Terminador (fin del mensaje)	0000

Versión	Modo			
	Númérico	Alfanumérico	Byte	Kanji
1-9	10	9	8	8
10-26	12	11	16	10
27-40	14	13	16	12

Figura 10. Indicador de modo y contador de caracteres.

En el modo numérico, los datos de entrada se dividen en grupos de tres dígitos y cada grupo se convierte en su equivalente binario de 10 bits (si el número de dígitos no es múltiplo de 3, los uno o dos últimos son convertidos a 4 ó 7 bits). Los datos binarios resultantes son entonces concatenados y se les pone como prefijo el indicador de modo y el indicador contador de caracteres.

Char.	Valor	Char.	Valor	Char.	Valor	Char.	Valor	Char.	Valor	Char.	Valor	Char.	Valor	Char.	Valor
0	0	6	6	C	12	I	18	O	24	U	30	SP	36	.	42
1	1	7	7	D	13	J	19	P	25	V	31	\$	37	/	43
2	2	8	8	E	14	K	20	Q	26	W	32	%	38	:	44
3	3	9	9	F	15	L	21	R	27	X	33	*	39		
4	4	A	10	G	16	M	22	S	28	Y	34	+	40		
5	5	B	11	H	17	N	23	T	29	Z	35	-	41		

Figura 11. Valores asignados en modo alfanumérico.

En el modo alfanumérico, a cada carácter se le asigna un valor entre 0 y 44 acorde a la tabla de la figura 11.

En el modo alfanumérico, los caracteres se agrupan primero por pares. El valor del primer carácter se multiplica por 45 y el valor del segundo se le suma a este producto y el resultado es convertido a un número de 11 bits (si el total de datos de entrada es impar, el último se codifica en 6 bits), concatenando los datos binarios resultantes. En el modo byte, el valor de cada carácter es el mismo que el de su correspondiente *codeword* de 8 bits, concatenándose dichos valores.

Versión	Módulos por lado	Módulos de patrones de función	Módulos de Información de formato y versión	Módulos en región de codificación	Codewords en región de codificación (datos y corrección de errores)	Bits restantes
1	21	202	31	208	26	0
2	25	235	31	359	44	7
3	29	243	31	567	70	7
4	33	251	31	807	100	7
5	37	259	31	1079	134	7
6	41	267	31	1383	172	7
7	45	390	67	1568	196	0
8	49	398	67	1936	242	0
9	53	406	67	2336	292	0
10	57	414	67	2768	346	0
11	61	422	67	3232	404	0
12	65	430	67	3728	466	0
13	69	438	67	4256	532	0
14	73	611	67	4651	581	3
15	77	619	67	5243	655	3
16	81	627	67	5867	733	3
17	85	635	67	6523	815	3
18	89	643	67	7211	901	3
19	93	651	67	7931	991	3
20	97	659	67	8683	1085	3
21	101	882	67	9252	1156	4
22	105	890	67	10068	1258	4
23	109	898	67	10916	1364	4
24	113	906	67	11796	1474	4
25	117	914	67	12708	1588	4
26	121	922	67	13652	1706	4
27	125	930	67	14628	1828	4
28	129	1203	67	15371	1921	3
29	133	1211	67	16411	2051	3
30	137	1219	67	17483	2185	3
31	141	1227	67	18587	2323	3
32	145	1235	67	19723	2465	3
33	149	1243	67	20891	2611	3
34	153	1251	67	22091	2761	3
35	157	1574	67	23008	2876	0
36	161	1582	67	24272	3034	0
37	165	1590	67	25568	3196	0
38	169	1598	67	26896	3362	0
39	173	1606	67	28256	3532	0
40	177	1614	67	29648	3706	0

Figura 12. Capacidad de los símbolos QR para las diferentes versiones.

Versión	Módulos	EC	Bits datos	Númérico	Alfanumérico	Byte	Kanji	Versión	Módulos	EC	Bits datos	Númérico	Alfanumérico	Byte	Kanji
1	21x21	L	152	41	25	17	10	21	101x101	L	7.456	2.232	1.352	929	572
		M	128	34	20	14	8			M	5.712	1.708	1.035	711	438
		Q	104	27	16	11	7			Q	4.096	1.224	742	509	314
		H	72	17	10	7	4			H	3.248	969	587	403	248
2	25x25	L	272	77	47	32	20	22	105x105	L	8.048	2.409	1.460	1.003	618
		M	224	63	38	26	16			M	6.256	1.872	1.134	779	480
		Q	176	48	29	20	12			Q	4.544	1.358	823	565	348
		H	128	34	20	14	8			H	3.536	1.056	640	439	270
3	29x29	L	440	127	77	53	32	23	109x109	L	8.752	2.620	1.588	1.091	672
		M	352	101	61	42	26			M	6.880	2.059	1.248	857	528
		Q	272	77	47	32	20			Q	4.912	1.468	890	611	376
		H	208	58	35	24	15			H	3.712	1.108	672	461	284
4	33x33	L	640	187	114	78	48	24	113x113	L	9.392	2.812	1.704	1.171	721
		M	512	149	90	62	38			M	7.312	2.188	1.326	911	561
		Q	384	111	67	46	28			Q	5.312	1.588	963	661	407
		H	288	82	50	34	21			H	4.112	1.228	744	511	315
5	37x37	L	864	255	154	106	65	25	117x117	L	10.208	3.057	1.853	1.273	784
		M	688	202	122	84	52			M	8.000	2.395	1.451	997	614
		Q	496	144	87	60	37			Q	5.744	1.718	1.041	715	440
		H	368	106	64	44	27			H	4.304	1.286	779	535	330
6	41x41	L	1.088	322	195	134	82	26	121x121	L	10.960	3.283	1.990	1.367	842
		M	864	255	154	106	65			M	8.496	2.544	1.542	1.059	652
		Q	608	178	108	74	45			Q	6.032	1.804	1.094	751	462
		H	480	139	84	58	36			H	4.768	1.425	864	593	365
7	45x45	L	1.248	370	224	154	95	27	125x125	L	11.744	3.514	2.132	1.465	902
		M	992	293	178	122	75			M	9.024	2.701	1.637	1.125	692
		Q	704	207	125	86	53			Q	6.464	1.933	1.172	805	496
		H	528	154	93	64	39			H	5.024	1.501	910	625	385
8	49x49	L	1.552	461	279	192	118	28	129x129	L	12.248	3.669	2.223	1.528	940
		M	1.232	365	221	152	93			M	9.544	2.857	1.732	1.190	732
		Q	880	259	157	108	66			Q	6.968	2.085	1.263	868	534
		H	688	202	122	84	52			H	5.288	1.581	958	658	405
9	53x53	L	1.856	552	335	230	141	29	133x133	L	13.048	3.909	2.369	1.628	1.000
		M	1.456	432	262	180	111			M	10.136	3.035	1.839	1.264	778
		Q	1.056	312	189	130	80			Q	7.288	2.181	1.322	908	559
		H	800	235	143	98	60			H	5.608	1.677	1.016	698	430
10	57x57	L	2.192	652	395	271	167	30	137x137	L	13.880	4.158	2.520	1.732	1.070
		M	1.728	513	311	213	131			M	10.984	3.289	1.994	1.370	843
		Q	1.232	364	221	151	93			Q	7.880	2.358	1.429	982	604
		H	976	288	174	119	74			H	5.960	1.782	1.080	742	457
11	61x61	L	2.592	772	468	321	198	31	141x141	L	14.744	4.417	2.677	1.840	1.132
		M	2.032	604	366	251	155			M	11.640	3.486	2.113	1.452	894
		Q	1.440	427	259	177	109			Q	8.264	2.473	1.499	1.030	634
		H	1.120	331	200	137	85			H	6.344	1.897	1.150	790	486
12	65x65	L	2.960	883	535	367	226	32	145x145	L	15.640	4.686	2.840	1.952	1.200
		M	2.320	691	419	287	177			M	12.328	3.693	2.238	1.538	947
		Q	1.648	489	296	203	125			Q	8.920	2.670	1.618	1.112	684
		H	1.264	374	227	155	96			H	6.760	2.022	1.226	842	518
13	69x69	L	3.424	1.022	619	425	262	33	149x149	L	16.568	4.965	3.009	2.068	1.270
		M	2.672	796	483	331	204			M	13.048	3.909	2.369	1.628	1.000
		Q	1.952	580	352	241	149			Q	9.368	2.805	1.700	1.168	719
		H	1.440	427	259	177	109			H	7.208	2.157	1.307	898	553
14	73x73	L	3.688	1.101	667	458	282	34	153x153	L	17.528	5.253	3.183	2.188	1.350
		M	2.920	871	528	362	223			M	13.800	4.134	2.506	1.722	1.060
		Q	2.088	621	376	258	159			Q	9.848	2.949	1.787	1.228	756
		H	1.576	468	283	194	120			H	7.688	2.301	1.394	958	590
15	77x77	L	4.184	1.250	758	520	320	35	157x157	L	18.448	5.529	3.351	2.303	1.420
		M	3.320	991	600	412	254			M	14.496	4.343	2.632	1.809	1.110
		Q	2.360	703	426	292	180			Q	10.288	3.081	1.867	1.283	790
		H	1.784	530	321	220	136			H	7.888	2.361	1.431	983	605
16	81x81	L	4.712	1.408	854	586	361	36	161x161	L	19.472	5.836	3.537	2.431	1.500
		M	3.624	1.082	656	450	277			M	15.312	4.588	2.780	1.911	1.180
		Q	2.600	775	470	322	198			Q	10.832	3.244	1.966	1.351	832
		H	2.024	602	365	250	154			H	8.432	2.524	1.530	1.051	647
17	85x85	L	5.176	1.548	938	644	397	37	165x165	L	20.528	6.153	3.729	2.563	1.580
		M	4.056	1.212	734	504	310			M	15.936	4.775	2.894	1.989	1.220
		Q	2.936	876	531	364	224			Q	11.408	3.417	2.071	1.423	876
		H	2.264	674	408	280	173			H	8.768	2.625	1.591	1.093	673
18	89x89	L	5.768	1.725	1.046	718	442	38	169x169	L	21.616	6.479	3.927	2.699	1.660
		M	4.504	1.346	816	560	345			M	16.816	5.039	3.054	2.099	1.290
		Q	3.176	948	574	394	243			Q	12.016	3.599	2.181	1.499	923
		H	2.504	746	452	310	191			H	9.136	2.735	1.658	1.139	701
19	93x93	L	6.360	1.903	1.153	792	488	39	173x173	L	22.496	6.743	4.087	2.809	1.730
		M	5.016	1.500	909	624	384			M	17.728	5.313	3.220	2.213	1.360
		Q	3.560	1.063	644	442	272			Q	12.656	3.791	2.298	1.579	972
		H	2.728	813	493	338	208			H	9.776	2.927	1.774	1.219	750
20	97x97	L	6.888	2.061	1.249	858	528	40	177x177	L	23.648	7.089	4.296	2.953	1.820
		M	5.352	1.600	970	666	410			M	18.672	5.596	3.391	2.331	1.440
		Q	3.880	1.159	702	482	297			Q	13.328	3.993	2.420	1.663	1.020
		H	3.080	919	557	382	235			H	10.208	3.057	1.852	1.273	784

Figura 13. Capacidad de símbolos según versión y modo de codificación.

Una vez codificados todos los datos en uno o varios modos con sus cabeceras y contadores adecuados, los flujos resultantes deben dividirse en grupos de 8 bits, dando lugar a los *codewords* de datos (si los datos de entrada no son múltiplos de 8, se añaden bits de relleno al último *codeword*).

El mensaje resultante se extiende con *codewords* de relleno adicionales (“11101100” y “00010001” alternativamente) hasta completar la capacidad para datos del símbolo y nivel de corrección de errores elegidos. Finalmente, a estos *codewords* de datos se le añaden los *codewords* ECC.

### Construcción y colocación de los *codewords* en el símbolo

El número total de *codewords* en el mensaje viene determinado por el tamaño de la región de codificación, que es el tamaño del símbolo menos el de los patrones de función. Este número total se dividirá en *codewords* de datos y *codewords* de corrección de errores, más los *codewords* restantes si son necesarios para completar la estructura del símbolo.

Para construir la secuencia de *codewords* en el símbolo, los *codewords* de datos deben dividirse para poderles aplicar el algoritmo de Reed-Solomon en un cierto número de bloques, determinado por la

versión y el nivel de corrección de errores seleccionado. Para cada uno de estos bloques se calculan los correspondientes *codewords* de corrección de errores de acuerdo a los coeficientes de los polinomios establecidos por el algoritmo.

Para componer la secuencia final se concatenan primero por turnos los *codewords* de datos de cada bloque (por ejemplo, para K bloques de N *codewords* cada uno: B1CD1, B2CD1, ..., BNCD1, B1CD2, B2CD2, ..., B1CDK, BNCDK). A continuación se concatenan de la misma forma los *codewords* de errores obtenidos (por ejemplo, para K bloques de M *codewords* de error cada uno: B1CE1, B2CE1, ..., BMCE1, B1CE2, B2CE2, ..., B1CEK, BMCEK). Tras esta secuencia se completa si es necesario con bits de relleno (en algunas versiones son necesarios 3, 4 ó 7 módulos). Si en una concatenación hay unos bloques con menor número de *codewords* de datos, se colocan éstos antes que el resto.

Hay dos tipos de colocación de los *codewords* en el símbolo: regular e irregular. Su uso depende de la posición dentro del símbolo y de que encuentren o no obstáculos (bordes del símbolo, patrones de función, información de versión o formato) en su colocación secuencial. La colocación regular se representa mediante bloques de 2x4 módulos verticales u horizontales (éstos últimos cuando hay un cambio de dirección). La colocación irregular se produce al encontrarse obstáculos en una colocación regular.

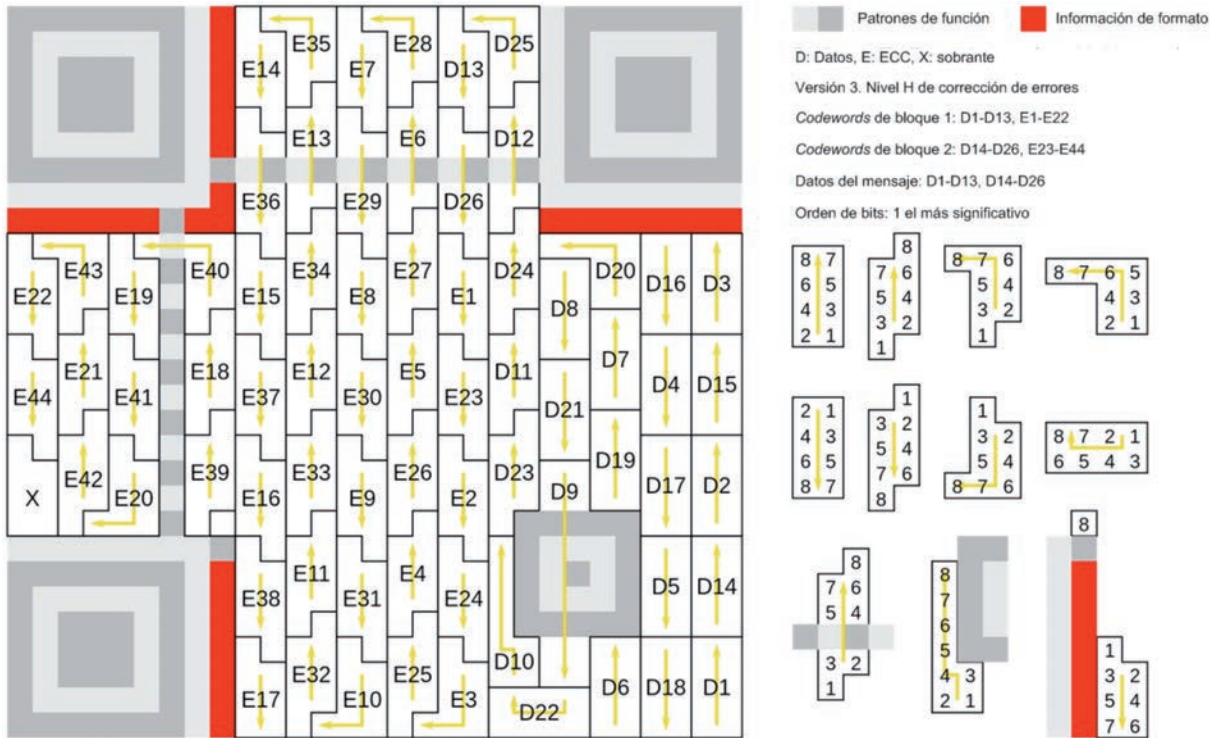


Figura 14. Ubicación de *codewords* de datos (26) y corrección de errores (44) en un símbolo 3-H (2 bloques).

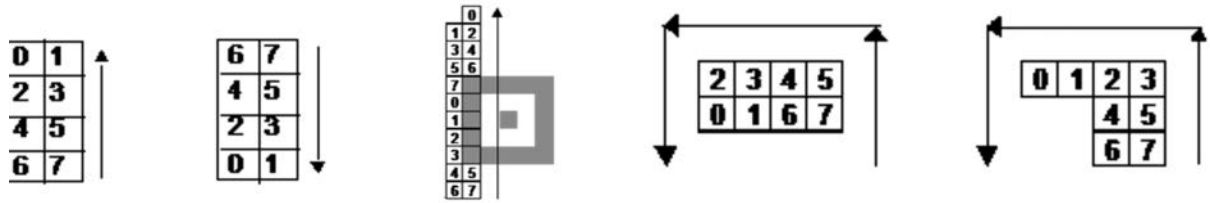


Figura 15. Ejemplos de tipos de colocación de *codewords* en módulos (7 el más significativo).

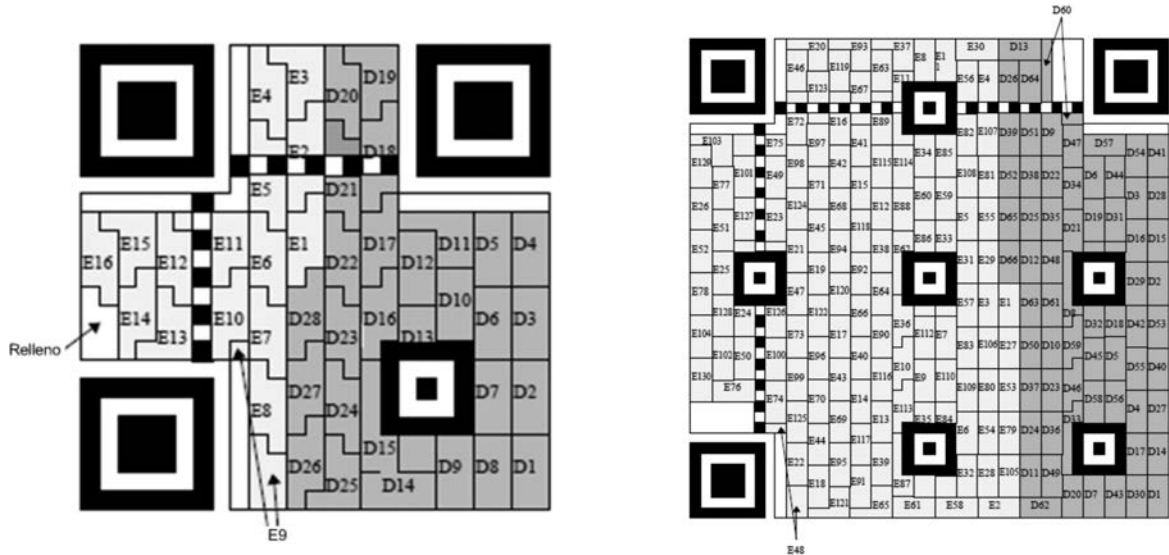


Figura 16. Ubicación de *codewords* en símbolo 2-M (1 bloque) y 7-H (5 bloques).

La colocación de la secuencia de *codewords* en la región de codificación comienza en la parte inferior derecha del símbolo (la única que no tiene patrón de localización), subiendo hasta encontrar el patrón superior y volviendo a bajar en la columna adyacente. La secuencia de bits será siempre de derecha a izquierda (comenzando por el más significativo) y hacia arriba o hacia abajo según la dirección de colocación. Continúa así hasta el final de la región de codificación, presentando formas irregulares cuando sea necesario.

(oscuros) debe estar equilibrado. Además, deben evitarse patrones que representen secuencias encontradas en los patrones de función, como por ejemplo “1011101” de los patrones de localización. Para conseguir ambas cosas se aplica una máscara de datos a los *codewords* de datos y de corrección de errores de la región de codificación (no se aplica por tanto ni a los patrones de función ni a la información de formato y versión), mediante una operación lógica XOR.

### Máscaras de datos

Para optimizar la decodificación de los símbolos QR, el número de módulos blancos (claros) y negros

Existen hasta 8 patrones de máscara aplicables. Para ver cuál es mejor, se aplican todos ellos y se comparan los resultados, seleccionando el más adecuado de acuerdo a un cálculo de defectos según unos pesos establecidos. Los patrones posibles y su código para la información de formato son los siguientes:

Código máscara	Condición
000	$(i + j) \bmod 2 = 0$
001	$i \bmod 2 = 0$
010	$j \bmod 3 = 0$
011	$(i + j) \bmod 3 = 0$
100	$((i \div 2) + (j \div 3)) \bmod 2 = 0$
101	$(i \cdot j) \bmod 2 + (i \cdot j) \bmod 3 = 0$
110	$((i \cdot j) \bmod 2 + (i \cdot j) \bmod 3) \bmod 2 = 0$
111	$((i+j) \bmod 2 + (i \cdot j) \bmod 3) \bmod 2 = 0$

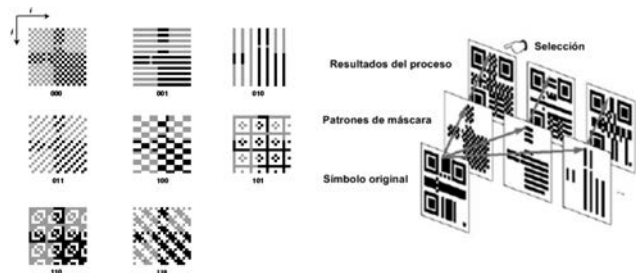


Figura 17. Patrones y proceso de máscara.

## Variantes de los códigos QR

Existen diversas variantes a los símbolos QR de versión 1 a 40. Una de ellas es el código “Micro QR”, definido también en el estándar. Es una versión reducida del código QR, pensado para espacios pequeños, con un único patrón localizador y menor capacidad de datos. Dispone de los mismos modos de caracteres soportados y de los mismos grados de corrección de errores, pero sólo tiene 4 versiones: M1 (11x11), M2 (13x13), M3 (15x15) y M4 (17x17). La versión M1 no dispone de corrección de errores ni información de versión.

Otra variante es el empleo de códigos QR apilados, para almacenar cantidades elevadas de datos en varios símbolos QR relacionados. El estándar permite hasta 16 símbolos QR apilados en 2 columnas. El conjunto de símbolos apilados incluye la información necesaria (número total de símbolos y orden en el conjunto) para reconstruir el mensaje completo ordenado.

Asimismo, diversas funcionalidades incluidas en el estándar dotan de aún mayor flexibilidad la simbología QR. Por ejemplo, en la codificación de caracteres se permite virtualmente cualquier modo posible, mediante los parámetros ECI. Este método posibilita crear nuevos modos de codificación, per-

mitiendo cualquier simbología necesaria (escritura árabe, alfabetos griego, ruso, etc.).

Mediante la reflectancia invertida, los papeles de los módulos negros (oscuros) y blancos (claros) se invierten. Esta funcionalidad es útil en entornos donde la iluminación no es adecuada. Por su parte, la imagen en espejo permite codificar los símbolos QR como si el lector lo viera en un espejo. Esta funcionalidad es útil en la presentación de símbolos QR en determinados entornos.

## Generación y lectura de códigos QR

### Elementos del sistema

Un sistema de generación y lectura de códigos QR comprende elementos de creación e impresión de códigos, así como de captura y decodificación. Así, un código QR puede ser generado con una aplicación software diseñada para ello (existen para múltiples sistemas operativos y múltiples entornos), y puede ser impreso con cualquier impresora compatible con códigos QR.

Un caso habitual de generación de códigos QR es emplear una aplicación desde un PC o un dispositivo

Versión	Módulos	ECC	Numérico	Alfanumérico	Binario	Kanji
M1	11x11		5	-	-	-
M2	13x13	L	10	6	-	-
		M	8	5	-	-
M3	15x15	L	23	14	9	6
		M	18	11	7	4
M4	17x17	L	35	21	15	9
		M	30	18	13	8
		Q	21	13	9	5

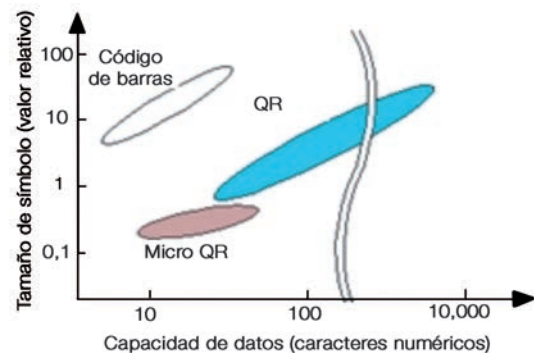


Figura 18. Capacidad de códigos Micro QR.



Figura 19. Estructura de códigos micro QR y empleo de símbolos QR apilados.

móvil y generar una imagen en un formato compatible para poder ser impreso en cualquier impresora doméstica o imprenta.

Para la lectura de un código QR se necesita primeramente un escáner de códigos QR o una cámara para la captura, además de un software para la decodificación del código capturado. Existen escáneres para entornos industriales (de coste no superior a los lectores de códigos de barras tradicionales), que pueden ser conectados asimismo a un PC o estación de trabajo para el procesamiento de los datos. Un caso muy habitual en el entorno doméstico es el empleo de las cámaras de fotos de los teléfonos móviles, para capturar la imagen de un código QR y decodificarla mediante un programa de coste nulo o muy bajo.

A la hora de generar un código QR, el tamaño del símbolo es importante. Una vez establecida la versión del símbolo (basada en la capacidad de datos, el modo de caracteres empleado y el nivel de corrección de errores deseado) es conveniente establecer un tamaño del módulo (basado en la resolución –en puntos por pulgada– de la impresora para generación y del escáner para su captura).

Aquí es importante también el área física ocupada por el código QR y la necesidad de disponer de una zona silenciosa (de al menos 4 módulos de anchura) alrededor del mismo para facilitar su decodificación. Si el área necesaria para el código no es adecuada para la ubicación pensada para el mismo, algunas opciones posibles son disminuir la versión, disminuir el tamaño del módulo o dividir el símbolo QR.

## **Pasos para codificación**

De acuerdo al estándar, el proceso de codificación se divide en 7 pasos:

1. Analizar los datos. Identificar el tipo de caracteres y establecer el modo de codificación adecuado. Determinar la versión, empleando la menor necesaria.
2. Codificar los datos. Convertir los datos en un flujo de bits según el modo seleccionado. Dividir los datos en *codewords* de 8 bits. Añadir los indicadores de modo para cada subconjunto de datos y el terminador. Añadir los caracteres de relleno necesarios para completar los *codewords* de la versión.
3. Codificar la corrección de errores. Seleccionar el nivel de corrección de errores. Dividir la secuencia de *codewords* de datos en los bloques necesarios y aplicar el algoritmo de corrección de errores. Generar los *codewords* de corrección de errores y añadirlos al final de la secuencia de corrección de *codewords* de datos.
4. Estructurar el mensaje. Entrelazar los *codewords* de datos y de error. Añadir si procede los bits restantes para completar la estructura.
5. Colocar los módulos en el símbolo. Ubicar los *codewords* de la región de codificación y los patrones de función para formar el símbolo de acuerdo a la versión seleccionada.



Figura 20. Impresoras de etiquetas de símbolos QR y escáneres industriales de códigos QR.

6. Enmascarar los datos. Aplicar los patrones de máscara a la región de codificación. Evaluar los resultados y seleccionar el patrón que optimice el equilibrio de módulos blancos y negros y minimice la aparición de patrones no deseados.
7. Generar la información de formato y versión (si procede). Ubicarla en el símbolo.

### **Pasos para decodificación**

El proceso de decodificación es el inverso al de codificación: a partir de un símbolo se obtienen unos datos en forma de caracteres. El estándar establece este proceso en 8 pasos:

1. Localizar y obtener una imagen del símbolo, y crear una matriz de bits "1" y "0" reconociendo en ella los módulos blancos (claros) y negros (oscuros).
2. Leer la información de formato. Obtener el nivel de corrección de errores y el tipo de patrón de máscara de datos empleado.
3. Leer la información de versión (si aplica) y determinar el tamaño en módulos del símbolo.
4. Aplicar la máscara de datos a la matriz en la región de codificación mediante la operación XOR.
5. Obtener los *codewords* de datos y de corrección de errores de acuerdo a la versión leída.
6. Detectar los posibles errores en los *codewords* de datos, a partir de los *codewords* de correc-

ción de errores y del nivel de corrección detectado.

7. Dividir los *codewords* de datos corregidos en segmentos de acuerdo a los indicadores de modo y a los contadores de caracteres encontrados.
8. Decodificar los caracteres de acuerdo a su modo de codificación y concatenar los resultados para obtener la cadena original.

### **Aplicaciones de los códigos QR**

La aparición de los códigos QR se produjo en el entorno de la automoción. Uno de los usos más típicos de las simbologías bidimensionales es la trazabilidad en procesos de fabricación, para seguimiento de la cadena de montaje de los productos.

Adicionalmente, a nivel industrial se emplean en gestión de logística y en control de inventarios. Pero además de los usos industriales, existen cada vez más ámbitos de la sociedad en los que se emplean códigos QR: publicidad personalizada y geolocalizada, educación, ocio, arte, etc.

Las amplias posibilidades de personalización (en contenido, en tamaño, en presentación, en ubicación) abre un mundo de enormes posibilidades para todo tipo de aplicaciones. Asimismo, la facilidad de decodificación de los símbolos QR unido a la facilidad de instalación y uso de software decodificador de códigos QR en dispositivos móviles con cámara, han permitido la aparición de infinidad de usos netamen-

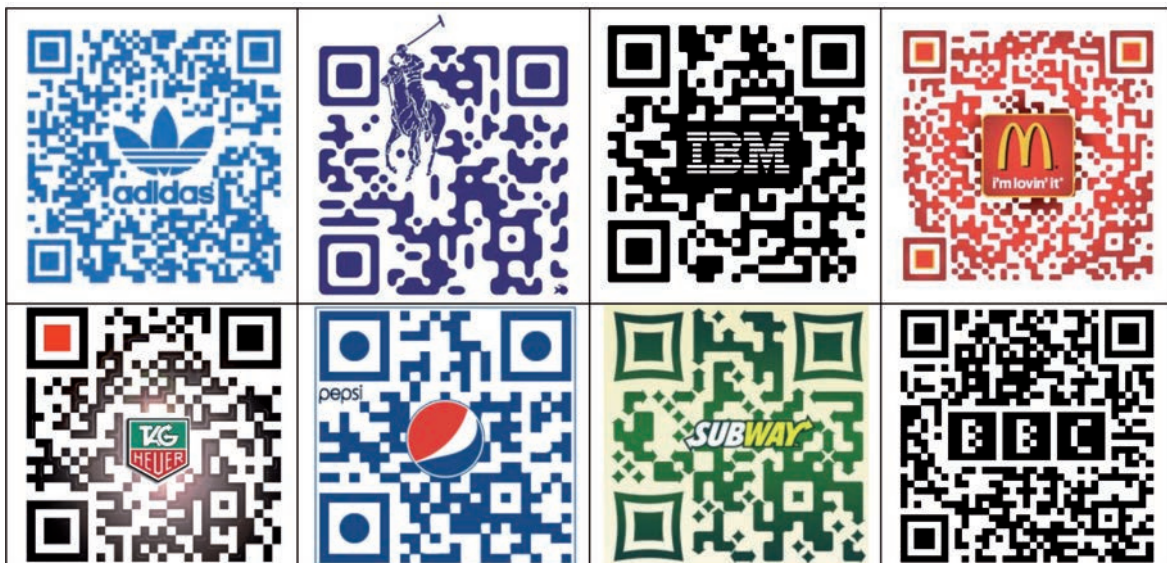


Figura 21. Ejemplos de logos insertados en símbolos QR.

te orientados al consumidor, con fines publicitarios e informativos, empleando los símbolos QR para por ejemplo codificar direcciones URL de Internet, informaciones de producto o cupones de promoción, entre otros.

Debido además a la robustez frente a errores de esta simbología, se emplean errores intencionales en los símbolos QR para introducir algún logo en la imagen del símbolo sin que la decodificación se vea afectada.

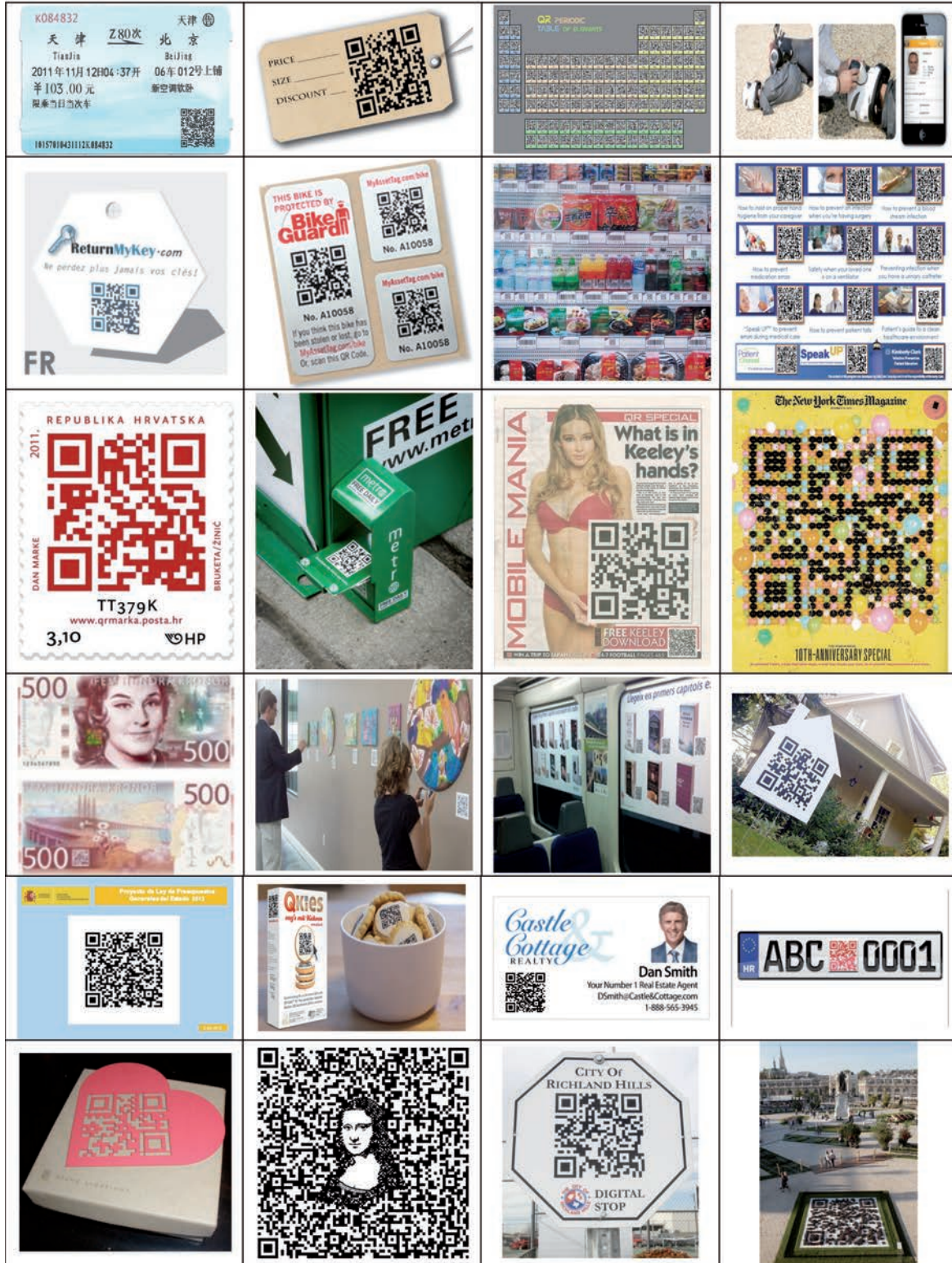


Figura 22. Ejemplos de aplicaciones de símbolos QR en la sociedad.





Figura 23. Ejemplos de símbolos QR con mensajes personalizados.

Las posibilidades de personalización han abierto un nuevo mercado en el que el usuario decide qué mensaje quiere codificar y encarga un producto “a medida”, quedando codificado e impreso el símbolo QR con el mensaje personal en el producto adquirido. Esta nueva filosofía es igualmente aplicable a la elaboración de multitud de regalos personalizados.

En el campo del arte también está experimentando un fuerte auge la simbología QR. Son muchos los artistas que han creado obras basadas explícitamente en códigos QR (pinturas, esculturas, arte funcional, etc.), existiendo incluso galerías de arte monotemáti-

cas sobre esta simbología. Igualmente también se emplean símbolos QR en el campo del ocio, integrándolos de diversas formas en pasatiempos cotidianos.

Debido a la tremenda proliferación de los códigos QR en los últimos tiempos, su empleo para decodificar URLs puede ser aprovechado para realizar ataques maliciosos (conocidos como *attagging* o *attack tagging*). Pueden crearse fácilmente y acceder a los dispositivos móviles y a su contenido. Por ello es conveniente decodificar códigos considerados confiables o disponer de un programa antivirus que evite el uso indebido de los datos decodificados. Asi-



Figura 24. Ejemplos de campañas publicitarias mediante símbolos QR.



Figura 25. Ejemplos de arte en símbolos QR.

mismo, para el manejo de datos sensibles se emplean símbolos QR con datos cifrados (por ejemplo mediante algoritmos DES).

La creación de símbolos QR atractivos se ha convertido en una prioridad en las estrategias de marketing. La facilidad de decodificación en diver-

sas circunstancias (colores, rotación, perspectiva, cualquier tamaño, etc.) permite la creación de símbolos QR muy diferentes y si cabe más efectivos.

El usuario no solo puede decodificar símbolos QR sin coste y sin esfuerzo. También puede generarlos fácilmente. Así, existe una gran variedad de genera-



Figura 26. Ejemplos de ocio en símbolos QR.



Figura 27. Consideraciones de diseño en símbolos QR.

dores de símbolos QR gratuitos, muchos de ellos utilizables en línea desde cualquier navegador de internet. Permiten múltiples tipos de entrada, de forma que su codificación se realiza de tal manera que prepara la acción correspondiente al decodificar en destino:

- Evento de calendario.
- Información de contacto.
- Dirección de correo electrónico.
- Número de teléfono.
- Datos de geolocalización.
- SMS.
- URL.
- Red WiFi.
- Texto libre.

Algunos de ellos permiten componer información para las redes sociales, vídeos, imágenes e incluso mensajes de correo electrónico. También permiten la posibilidad de personalizar la apariencia, seleccionando el color y la forma de los módulos del símbolo.

Igualmente, además de los programas de decodificadores para PC y dispositivos móviles, también existen decodificadores *online* gratuitos. Generan el

texto de salida admitiendo como entrada una imagen del símbolo o una URL que apunte a la imagen.

## Otros códigos bidimensionales

Los códigos en 2D o bidimensionales son la evolución natural de los códigos de barras unidimensionales inventados en 1952 (ejemplos son EAN, Code 128, Code 39, Code 93), por lo que en ocasiones son denominados “códigos de barras bidimensionales”. Los códigos bidimensionales superan las diversas limitaciones de los códigos de barras tradicionales: mayor espacio de impresión, poca capacidad de datos, pocos conjuntos de caracteres soportados, poca corrección de errores, etc.

Los códigos QR son un tipo de simbología bidimensional, aunque no la única existente. Existen así otras que también han sido estandarizadas, como PDF, DataMatrix o MaxiCode. Cada uno tiene sus ventajas e inconvenientes y es más empleado que otros en un determinado ámbito. El auge de los códigos QR se debe a que poseen todas las ventajas de los otros códigos bidimensionales estandarizados, y además pueden ser empleados masivamente desde dispositivos móviles con cámara.

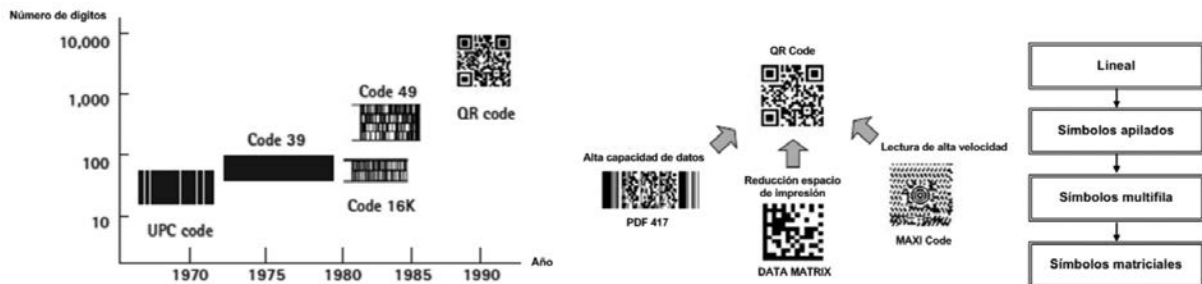


Figura 28. Evolución de códigos QR.

	QR Code	PDF 417	DataMatrix	MaxiCode
<b>Símbolo</b>				
<b>Desarrollador (país, año)</b>	Denso Wave (Japón, 1994)	Symbol Technologies (USA, 1991)	RVSI Acuity CiMatrix (USA, 1989)	UPS (USA, 1992)
<b>Tipo</b>	Matricial	Código de barras apilado	Matricial	Matricial
<b>Capacidad de datos</b>	Númérica	7,089	3,116	138
	Alfanumérica	4,296	1,850	93
	Binario	2,953	1,018	---
	Kanji	1,817	554	---
<b>Características principales</b>	Alta capacidad, área de impresión pequeña, alta velocidad de decodificación	Alta capacidad, robusto frente a errores, pocos caracteres soportados	Mayor eficiencia espacial, área de impresión pequeña	Alta velocidad de decodificación, poca capacidad de almacenamiento
<b>Usos principales</b>	Industrial y doméstico/comercial	Correos	Industrial	Logística
<b>Estandarización</b>	AIMI, ISO	AIMI, ISO	AIMI, ISO	AIMI, ISO

Figura 29. Comparativa de simbologías bidimensionales.

Existen igualmente simbologías bidimensionales de múltiples colores, como Colorcode, que consiguen mayor cantidad de información en el mismo espacio. Un ejemplo de este tipo de códigos. Requieren capturas de imagen de mayor calidad y contraste, lo que no los hace adecuados para uso cotidiano.

Además existen códigos bidimensionales propietarios (cuyas especificaciones no están publicadas y por tanto no son de libre uso), como es el caso de BIDI, introducidos en España por Movistar. Los códigos BIDI, aunque a menudo se citan como sinónimos de códigos QR, son simbologías diferentes a QR

(BIDI, por ejemplo, carece de los patrones de localización de las esquinas), son privados o de código cerrado y no gratuitos (debido a su orientación comercial).

### Enlaces de interés

A continuación se muestran diversos enlaces de interés, presentados como símbolo QR para mostrar al lector una de sus aplicaciones más útiles, la codificación de URLs:

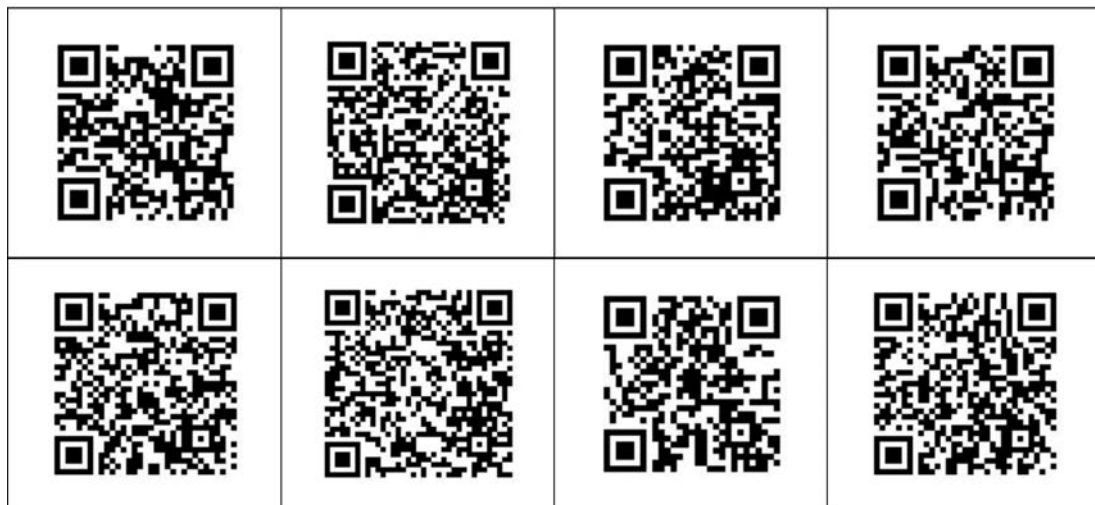


Figura 30. Enlaces de interés codificados en símbolos QR.