

Un buscador de expresiones regulares en textos ibéricos

Eduardo Orduña Aznar

Dirigido por la Dra. Rosa Pedrero Sancho

UNED-Departamento de Filología Clásica

Trabajo de investigación - Estudios de Tercer Ciclo. Curso 2002-2003

Índice general

1. Introducción	3
2. Estado actual de los estudios ibéricos	5
2.1. Introducción	5
2.2. Sistemas de escritura	6
2.3. Fonética y fonología	7
2.4. Morfología	10
2.5. Sintaxis	13
2.6. Léxico	13
2.7. Onomástica	14
2.8. Tipología	15
2.9. Situación del ibérico frente a otras lenguas	15
2.10. Vascoiberismo y comparación lingüística	16
3. Los archivos de textos ibéricos	18
3.1. Origen de los textos	18
3.2. Formato de los textos	19
3.3. Información sobre los textos	21
4. La interfaz CGI	22
4.1. Introducción	22
4.2. El servidor web	22
4.2.1. El programa cgi	24
4.3. Elementos de un cgi	25
4.4. El formulario	26
4.5. La cgi-lib.pl	27
5. El programa buscaiberico.cgi	31
5.1. Uso del programa	31
5.2. Descripción del código	32

6. Las expresiones regulares	46
6.1. Introducción	46
6.2.	47
6.3. Cuantificadores	49
6.4. Clases de caracteres	50
6.5. Metacaracteres	52
6.6. Paréntesis	53
6.7. Aserciones de longitud cero	54
6.8. Modificadores	55
7. Una aplicación del programa: el paradigma biteóke	57
7.1. Historia de la cuestión	57
7.2. Segmentación del paradigma	60
7.2.1. El núcleo	60
7.2.2. Los afijos	62
7.3. Estructura del paradigma y expresiones regulares	63
7.4. Posibles paradigmas relacionados	64
Conclusiones	68
Apéndice I: el código fuente	70
Apéndice II: el programa buscador por nomenclatura MLH	74
Apéndice III: nomenclatura provisional de textos no incluidos en MLH	76
Addenda	103

Capítulo 1

Introducción

En el presente trabajo de investigación se intenta aplicar una metodología informática al estudio y análisis de los textos ibéricos. El método elegido se basa en la utilización de las expresiones regulares, un sublenguaje utilizado en diversos lenguajes de programación para analizar cadenas de texto y extraer de ellas porciones que cumplan unos determinados requisitos, o bien hacer sustituciones de ciertos patrones de texto por otros. Dada la complejidad que encierra el trabajo con expresiones regulares, se imponía buscar un sistema flexible que permita su aplicación en función de los conocimientos y la imaginación del usuario, de forma que pueda usarse incluso sin saber nada de expresiones regulares, y al mismo tiempo permita al usuario más avanzado sacarles todo su partido.

Por tanto, me incliné por un medio universalmente accesible desde cualquier ordenador con cualquier sistema operativo, es decir, una página web con un formulario cuyo principal elemento está compuesto por dos ventanas de texto en las que el usuario puede introducir cualquier cadena de texto y obtiene como resultado todos los textos completos en que aparece la cadena buscada (o en los que concurren dos cadenas, si se usan las dos ventanas), con la cadena buscada resaltada. Además, un contador devuelve el número total de apariciones, por lo que el programa puede usarse también con fines estadísticos.

Así, el trabajo se ha desarrollado en dos vertientes: una más lingüística, implicada por la recogida exhaustiva de textos y elaboración del estado de la cuestión, además de las decisiones que afectan al formato de los textos (transcripción, variantes, bibliografía, etc.), y la aplicación propiamente dicha del programa con ejemplos de uso; y otra de carácter informático, centrada en la elaboración de la página web y especialmente del programa que efectúa las búsquedas de expresiones regulares, además del estudio de estas últimas.

Por último, quiero destacar que el programa en cuestión no pretende ser

una base de datos epigráfica en red, al estilo de otras existentes para otras lenguas, sino que su interés se centra en el análisis puramente lingüístico, como una herramienta útil principalmente para facilitar el trabajo de segmentación y de búsqueda de regularidades reveladoras de paradigmas gramaticales. Por ello, el formato de los ficheros es de texto plano o ASCII, que es el más apto para el tratamiento informático, aunque impide aplicar las convenciones tipográficas usuales en los repertorios epigráficos. También se han dejado de lado la mayoría de informaciones relativas al tamaño del soporte, signos, datación y otras, que pueden consultarse en la bibliografía correspondiente, y en su mayor parte en los MLH de Untermann.

Convenciones tipográficas

Se ha utilizado el tipo de letra conocido como **de teletipo o terminal** para transcribir código de programación o html, expresiones regulares, o cualquier texto que deba teclearse en un ordenador. En los comentarios propiamente lingüísticos se siguen las convenciones habituales: los textos ibéricos en alfabeto epicórico se transcriben en minúscula negrita, en cursiva los que están en alfabeto greco-ibérico, y en mayúscula los que usan alfabeto ibérico. Cuando lo que se cita no son palabras completas (separadas generalmente por interpunción), sino segmentos obtenidos por análisis lingüístico (raíces, sufijos, etc.), se utiliza la cursiva.

Capítulo 2

Estado actual de los estudios ibéricos

2.1. Introducción

Pese a todos los esfuerzos, el mayor avance en el conocimiento de la lengua ibérica sigue siendo el desciframiento del alfabeto por Gómez Moreno¹. Este avance capital supuso poner en evidencia que el ibérico no era una forma arcaica de la lengua vasca, como hasta entonces se había supuesto, desde tiempos de Humboldt. El desciframiento permitió el reconocimiento de la existencia de varias lenguas indígenas en la península ibérica, al menos dos de ellas escritas en alfabeto ibérico: el ibérico y el celtibérico. Por fin fue posible la lectura de un conjunto considerable de textos, escritos en una lengua que sigue siendo incomprensible. El principal escollo es la ausencia tanto de textos bilingües de suficiente extensión como de una lengua lo bastante próxima como para poder utilizar los métodos de la lingüística comparativa. Pese a todo, los años transcurridos desde el desciframiento no han sido en absoluto estériles, y se han producido una serie de avances que, aún siendo poco espectaculares, permiten ya un atisbo de comprensión de inscripciones de poca extensión (principalmente funerarias o de propiedad sobre *instrumentum*), además de intuir algunas características gramaticales o tipológicas de la lengua.

¹Gómez Moreno (1949a)

2.2. Sistemas de escritura

La lengua ibérica utilizó al menos cuatro sistemas para su escritura: el alfabeto latino, el griego jónico, el ibérico noroccidental o levantino, y el sudibérico.

Cronológicamente, el primero en ser utilizado fue el alfabeto griego, en su variante jónica. Gómez Moreno² fue el primero en reconocer que el Plomo de Alcoy estaba escrito en este sistema, o mejor en una adaptación de dicho sistema, pues algunas letras no se utilizan, y en cambio la <R> y <S> tienen un correlato con diacrítico (una especie de apóstrofe). Por ello es mejor referirse a este uso como “alfabeto grecoibérico”. No abundan los textos en alfabeto grecoibérico, aunque hay alguno bastante extenso, especialmente el mencionado Plomo de Alcoy.

Los textos en alfabeto latino son aún más escasos y breves, aunque contamos además con nombres propios ibéricos en textos en lengua latina, como ocurre con el Bronce de Ascoli, cuya importancia fue capital para el describamiento del alfabeto ibérico.

Los textos alfabéticos, tanto en alfabeto latino como grecoibérico, son esenciales para conocer algunos aspectos de la lengua ibérica, debido a las limitaciones inherentes al signario ibérico. Por ejemplo, sólo gracias a ellos nos es conocida la existencia de palabras acabadas en oclusiva en ibérico, lo que no se refleja con el alfabeto ibérico.

A diferencia de los dos anteriores, el alfabeto ibérico (tanto el noroccidental como el sudibérico) es en realidad un semisilabario, en el que hay quince signos correspondientes a cada uno de los tres órdenes de oclusivas seguidas por cada una de las cinco vocales. No hay distinción entre sonoras y sordas, aunque parece, a partir de los textos alfabéticos, que faltaba la /p/. Los restantes signos son alfabéticos: cinco vocálicos, uno para la lateral /l/, dos vibrantes, dos silbantes y una nasal /n/. En un número reducido de inscripciones aparece además un signo para /m/, y existe además un signo, de forma Y, cuyo valor preciso se desconoce³. Parece probable su carácter nasal, pues sirvió para representar la n en parte de la Celtiberia, y además se relaciona el elemento antroponímico UMAR en alfabeto latino con **m̄bar** en epicórico. De esta última ecuación se desprende también el posible valor vocálico del signo (como también podría sugerir el signo usado para representarlo), relacionado quizás con el timbre /u/. En cualquier caso, parece claro que actúa en ocasiones como centro de sílaba. Por otra parte, también parece clara la relación entre *nai* en alfabeto grecoibérico y **m̄i** en ibérico, lo

²Gómez Moreno (1949b)

³Correa (1999), Quintanilla (1998)

que lleva a Rodríguez Ramos a afirmar que procede de una /a/ nasalizada⁴.

Otro problema es la falta de distinción de sonoridad de las oclusivas en alfabeto ibérico. Por los textos alfabéticos sabemos que se distinguían, y de hecho se ha detectado en algunos textos levantinos un sistema para marcar la presencia de sonoridad mediante uno o más trazos de más en la grafía del signo⁵. En cualquier caso, el sistema no se generalizó, y su uso más frecuente en el sur de Francia y noreste de Cataluña han llevado a Correa a designarlo como “variante nororiental”⁶.

Existen además en alfabeto noroccidental algunos signos de uso restringido a una sólo inscripción o, a lo sumo, a unas pocas, como es el caso del signo que aparece en el Plomo del Castell de Palamós, cuya forma es (-) (véase en vertical), y que Fletcher⁷ entre otros considera /ta/, al faltar el silabograma correspondiente en el texto.

Por último, el alfabeto sudibérico es un semisilabario, como el noroccidental. Su descriframiento aún no se ha completado, y si bien los textos son legibles en buena medida, las lecturas difieren notablemente según los autores, destacando como principales sistemas el de Untermann, De Hoz, Correa y Rodríguez Ramos. Se trata de un alfabeto más antiguo que el noroccidental, por el que acabó siendo sustituido, de modo que el conjunto de textos en que aparece es mucho más escaso.

2.3. Fonética y fonología

Para los aspectos fonológicos contamos en la actualidad con el manual de Quintanilla⁸, que cubre todo el repertorio fonológico ibérico, aunque con desigual profundidad. Por ejemplo, se dedican 145 páginas al vocalismo, por 95 a las consonantes (apenas 5 a las silbantes). Hasta entonces, aparte de los trabajos pioneros de Tovar o Mariner Bigorra, sólo existían estudios parciales. Otros han aparecido con posterioridad. Entre ellos merece la pena destacar el del propio Quintanilla para las vocales⁹, los de Correa y Quintanilla para las vibrantes,¹⁰ los de Correa, Rodríguez Ramos y Valeri para las nasales,¹¹

⁴Rodríguez Ramos (2001a, p.13 n.7)

⁵de Hoz (1985), Quintanilla (1993b)

⁶Correa (1994a, p. 268)

⁷Fletcher (1992)

⁸Quintanilla (1998)

⁹Quintanilla (1993a)

¹⁰Correa (1994b); Quintanilla (1999)

¹¹Correa (1999); Rodríguez Ramos (2000a); Valeri (1993)

los de Correa y Silgo para las silbantes,¹² y el de Silgo para el acento.¹³

El sistema fonológico ibérico, tal como lo describe Quintanilla¹⁴, consta de cinco vocales (*a, e, i, o, u*), dos laterales (*l, lt*), dos vibrantes (*r, r̂*), dos nasales (*n, m*, “además de *m̄*, no bien identificado”), dos silbantes (*s, ŝ*, y dos series de oclusivas, sordas (*t, k*) y sonoras (*b, d, g*). Se constata, gracias especialmente al Plomo de Alcoy (G.1.1), la ausencia de *f, p, m*, la falta de *r-* inicial, falta de aspiración, falta de grupos *muta cum liquida* y la falta de grupos de consonante en inicial. Señala también Quintanilla que, a partir de los textos latinos (fundamentalmente el Bronce de Ascoli), se deduce un empleo restringido de *p* y *m*, alternando con *b*; el empleo de *h*; la presencia de geminadas *-nn-*, *-ll-*, *-rr-*; la falta de sonora tras silbante, y la falta de *d-* inicial.

La inclusión de *lt* como fonema, a diferencia de otros autores¹⁵, se debe a la equivalencia de dicha secuencia con *-ll-* en alfabeto latino, que para Quintanilla revela la existencia en ibérico de un fonema fuerte que se opone a *l* entre vocales, y se representaría por *lt*¹⁶.

El cuadro que presenta de Hoz¹⁷ es similar, con la diferencia de no incluir *lt*, incluyendo en cambio (con interrogante) las semivocales /y/ y /w/. También marca con interrogante la segunda silbante y vibrante y el signo Y. El interrogante aquí parece referirse más a la ubicación en el sistema que a su existencia como fonema. En el caso de la /m/ sí que se duda explícitamente de su existencia (p. 338). Velaza¹⁸ admite la semivocal /y/, pero cree que /w/ se limita a préstamos.

Merece la pena resaltar aquí una hipótesis reciente de de Hoz¹⁹ que explicaría la ausencia de /p/. Se trata de suponer en ibérico una oposición no entre sonoras y sordas, sino entre fuertes y lenes, como se reconstruye en protovasco²⁰, de modo que la /p/ sería el fonema marcado, y sería menos extraña su ausencia. Ello explicaría además que en la variante nororiental sean al parecer los signos de las sonoras los que llevan un trazo de más.

Otros grupos de signos problemáticos son los correspondientes a las vibrantes, silbantes y nasales. Para todos ellos hay al menos dos signos (en el caso de las nasales sólo uno es seguro, pero podrían ser tres), pero se des-

¹²Correa (2001); Silgo Gauche (2000a)

¹³Silgo Gauche (1994-1995)

¹⁴Quintanilla (1998, p.34)

¹⁵Velaza (1996c, p. 42), de Hoz (2001a, p. 337), Correa (1994a, 274)

¹⁶Quintanilla (1998, p. 248)

¹⁷de Hoz (2001a, p. 337)

¹⁸Velaza (1996c, p. 41)

¹⁹de Hoz (2001a, p.338)

²⁰Michelena (1977)

conoce la naturaleza exacta de la oposición entre ellos. En el caso de las vibrantes, Quintanilla²¹ destaca la similitud de los hechos ibéricos con los que conocemos para el aquitano y el vasco, fundamentalmente la falta de vibrantes en posición inicial. En posición final hay oposición en ibérico entre las dos vibrantes, a diferencia del vasco, aunque hay huellas de que pudo darse también esa oposición en posición final en una fase más antigua. Destaca que en vasco-aquitano la oposición se da por el rasgo simple-múltiple. Por último, hay indicios de que la *-r* final en algunos segmentos léxicos tenga valor gramatical.

Para las silbantes la situación es más compleja. La aparente ausencia de africadas en ibérico llevó a Michelena²² a sugerir la existencia en ibérico de una distribución de las silbantes similar a la del vasco: dos ápicopalveolares (*s, ts*) y dos dorsoalveolares (*z, tz*), con distinción en cada serie entre africadas y fricativas. El rasgo omitido por la escritura ibérica sería el modo de articulación, de forma paralela a lo que ocurre con las oclusivas, y la oposición que refleja la escritura sería únicamente el punto de articulación. Silgo²³ propone, en cambio, que la oposición reflejada es africada/fricativa, representadas respectivamente por **s**, **ś**. Últimamente Correa²⁴ ha expuesto su postura contraria a que las grafías reflejen una oposición de sonoridad, aunque no descarta que exista una oposición por el punto de articulación, que podría completarse por una oposición de modo de articulación no reflejada por la escritura, es decir, básicamente lo que ya había propuesto Michelena.

Entre las nasales, sólo parece claro el uso de *n*. En cambio parece muy limitado el uso de *m*, inexistente en grecoibérico. Su existencia en el Bronce de Ascoli se puede explicar como resultado de una asimilación *nb > m*.²⁵ Untermann²⁶ señala la presencia de *m* y ausencia de *m̄* en el plomo de Ullastret. Para Valeri, el fonema /m/ se integraría en ibérico por vía de los préstamos, y la oposición se daría entre *n* y < *Y* >, siendo una oposición *forte/lene* respectivamente. Quintanilla²⁷ coincide en la ausencia en ibérico de una nasal labial. Correa, en el estudio más completo dedicado a las nasales²⁸, no descarta que *m* represente una nasal geminada o mejor fuerte. Queda por último como mayor problema pendiente del desciframiento del alfabeto ibérico el

²¹Quintanilla (1999, p. 568)

²²Michelena (1985a, p. 364)

²³Silgo Gauche (2000a, p. 521)

²⁴Correa (2001, p. 316-7)

²⁵Valeri (1993, p. 741)

²⁶Untermann (1990, §409)

²⁷Quintanilla (1998, p. 215)

²⁸Correa (1999, p. 385)

valor del signo < Y >. Coinciden Correa²⁹ y Quintanilla³⁰ en ver en él los rasgos de nasalidad y labialidad. Las ecuaciones más seguras en escrituras alfabéticas son (lat.) UMAR = (ib.) **m̄bar** y (gr-ib.) *-nai* = (ib.) **m̄i**. Para Rodríguez Ramos, en cambio, se trataría de una vocal nasalizada³¹.

El cuadro fonológico presentado es muy similar al que se reconstruye para el protovasco (y que es básicamente el del aquitano³²), como señalan todos los autores citados. También coinciden ambas lenguas en la ausencia de /f/, de grupos de *muta cum liquida* y de grupos consonánticos y de r- en inicial, así como la posible ausencia de /m/. Difieren en cambio en la presencia de la aspiración en protovasco.

La estructura de la sílaba es también similar en ibérico a la que se reconstruye para el protovasco³³. Podríamos representarla, siguiendo a De Hoz, de la siguiente forma:

(C)V(Y)(R)(S)(T)

Donde C representa cualquier consonante (o semivocal), V una vocal, Y semivocal, R cualquier sonante (r, n, l), S las silbantes y T cualquier oclusiva, aunque probablemente no todas pudieran ocupar esta posición. En textos en alfabeto grecoibérico se constata la presencia de *-t, -d, -k, -g*. Quintanilla³⁴ sugiere que se trate de elementos gramaticales, pues en las fuentes latinas las oclusivas no aparecen en posición final.

2.4. Morfología

Nuestro conocimiento de la gramática ibérica sigue siendo extremadamente precario. Se ha observado la existencia de segmentos (mayoritariamente sufijos) que parecen tener valor gramatical, y que se denominan *morfos*, con deliberada ambigüedad. Untermann, en el primer volumen de MLH III³⁵, presenta una completa relación de los morfos identificados hasta el momento (unos cincuenta). Algunos aparecen como prefijos (**bas-**, **is-**) o infijos ((**n**)**m̄li**), pero la inmensa mayoría son sufijos, lo cual no impide que se agrupen en ocasiones unos con otros, como en la frecuente secuencia **ar-en-m̄i**.

En la exposición de Untermann se prescinde de valorar sus posibles funciones o significados, y de hecho no hay prácticamente indicios para la mayoría

²⁹Correa (1999, p. 392)

³⁰Quintanilla (1998, p. 218)

³¹Rodríguez Ramos (2000c, p. 36)

³²Gorrochategui (1984, pp. 373-379)

³³Michelena (1985c, p. 345), Correa (1994a, p. 278), de Hoz (2001a, p. 339)

³⁴Quintanilla (1998, p.38)

³⁵Untermann (1990, pp. 509-543)

de ellos, pero hay unos pocos que, por su frecuente aparición o especialmente por el contexto favorable en que se manifiestan (monedas, inscripciones funerarias o marcas de propiedad sobre cerámica), han suscitado numerosos intentos de interpretación, con el apoyo de textos similares en lenguas conocidas, como el latín.

El más conocido de ellos es el sufijo **-en**, que tradicionalmente se ha considerado equivalente al sufijo de genitivo vasco de idéntica forma.³⁶ La relación entre ambos sufijos parece dificultada por el hecho de que se postula una protoforma *-e* para el genitivo vasco, conservada aún hoy en la declinación pronominal³⁷, pero en cualquier caso el valor de genitivo o similar del sufijo ibérico **-en** permanece hasta hoy indiscutido.

Relacionado con el anterior está el complejo sufijal **-(e)sken**, frecuente en monedas, al que ha dedicado de Hoz un trabajo reciente³⁸, en el que llega, a partir del propio ibérico, al mismo resultado al que llegaba Schuchardt por comparación con el vasco, es decir, que se trata de un complejo que reúne un indicador de pertenencia **-es**, un formador de étnicos o pluralizador **-ke** y un genitivo **-en**.

Otro sufijo sobre el que se ha especulado ha sido **-te**, que aparece a menudo ante la palabra **ekiar** (así en el mosaico de Caminreal), a la que se atribuye un significado de “hecho” o similar. Así, Velaza³⁹ dudaba, al intentar interpretar la inscripción de Caminreal, entre un valor de agente o dativo. El definitivo restablecimiento de la lectura **afsbikisteekiar** en una moneda de Sagunto (A.33) inclina al mismo autor a favor de la interpretación de **-te** como agente, más verosímil en una moneda.⁴⁰ Coincide en esta interpretación Rodríguez Ramos⁴¹, aunque le atribuye además un valor de ablativo.

También el sufijo **-ka** ha sido motivo de controversia. Untermann (*Gramática de los plomos ibéricos*, p. 38) defendió un valor de ergativo, frente a de Hoz⁴², que propone una función de complemento indirecto o ablativo de origen, basándose en la frecuencia de nombres personales con dicho sufijo y seguidos de numerales. Recientemente, Velaza⁴³ considera que faltan evidencias para decidirse entre las varias posibilidades existentes.

Por último, sin agotar el tema, nos referiremos al sufijo **m̄i**, frecuente en inscripciones funerarias y de propiedad sobre *instrumentum*, que ha dado

³⁶Véase, por ejemplo, Michelena (1985b).

³⁷Gorrochategui y Lakarra (1996, p. 132)

³⁸de Hoz (2002)

³⁹Velaza (1991a)

⁴⁰Velaza (2001a)

⁴¹Rodríguez Ramos (2002a, p. 122)

⁴²de Hoz (1981a, p. 482)

⁴³Velaza (2002b, p. 272)

lugar a un abanico de interpretaciones entre partícula posesiva, pronombre personal de primera persona (relacionado con vasco *ni*), verbo o adverbio. Podemos citar, por ejemplo, a Michelena⁴⁴, que apoya la interpretación como pronombre de primera persona singular en el paralelo con las inscripciones parlantes itálicas, e incluso vasco-medievales. En la aportación más reciente sobre este sufijo, de Hoz⁴⁵ se inclina por considerarlo pronombre demostrativo.

De la morfología verbal ibérica nada se sabe, a pesar de algún intento como el de Silgo⁴⁶, cuyo único fundamento es la comparación con el vasco. Untermann⁴⁷ ha señalado la probabilidad de que que las formas **bitéroke**, **bitérokan**, etc, que parecen formar un auténtico paradigma, tengan valor verbal. Coincide con él Javier de Hoz,⁴⁸ que ha reunido recientemente todos los testimonios del paradigma para ejemplificar con él el carácter incorporante de la lengua ibérica.

Salvo este paradigma, tan sólo algunas formas aisladas se suelen considerar formas verbales, aunque no hay acuerdo en todos los casos. Para la forma **ekiar**, **ekien** de los mosaicos de Caminreal y Andelos, respectivamente, se ha propuesto por varios autores un valor verbal (“hecho, hizo”), apoyado por la similitud con el vasco *egin* (“hacer”) pero de Hoz⁴⁹ recuerda que las desinencias **-ar**, **-en** acompañan a sustantivos. Otra presunta forma verbal es la documentada en la fórmula funeraria **añe take/teki**, que parece corresponder al latín HIC SITUS EST, también aquí con apoyo en una forma vasca, *dago* (“está”). Por último, está la forma **tebanen**, que aparece en un arquitrabe bilingüe saguntino (F.11.8), donde parece equivaler al latín COERAV(IT), opción defendida por Untermann,⁵⁰ para quien serían ejemplos del mismo verbo los varios **eban**, **ebanen**, **teban** frecuentes en inscripciones funerarias, que serían por tanto verbos de dedicación. Velaza, en cambio, defiende una interpretación como sustantivo, como veremos en el siguiente apartado. Rodríguez Ramos⁵¹ apoya a Untermann mediante una argumentación vascoiberista que relaciona **tebanen** con el verbo vasco *ibeni*, *ipini* (“poner”).

Respecto a los pronombres, Untermann⁵² propone identificar como tales

⁴⁴Michelena (1985b, p. 382)

⁴⁵de Hoz (2001a, p. 343)

⁴⁶Silgo Gauche (1996b)

⁴⁷Untermann (1996, p. 92)

⁴⁸de Hoz (2001a, 344-349)

⁴⁹de Hoz (2001a, p. 356)

⁵⁰Untermann (1984)

⁵¹Rodríguez Ramos (2000c)

⁵²Untermann (1990, §556)

una serie de segmentos breves que pueden recibir varios de los sufijos identificados con los sustantivos, y establece un cuadro con una docena de estos posibles pronombres y su paradigma.

2.5. Sintaxis

Nuestros conocimientos de la sintaxis ibérica son, naturalmente, muy escasos, debido a la falta de comprensión de los textos. Entre lo poco que se puede intuir hay que mencionar la posible existencia de flexión de grupo, propuesta por Untermann. Parece probable, como defiende de Hoz,⁵³ que el ibérico fuera “una lengua pospositiva, en la que el complemento precede a su núcleo, y con orden básico SOV”. El mismo autor sugiere que el ibérico distinguía dos tipos de posesión, tal vez alienable/inalienable, expresando la primera mediante el sufijo *-en* y la segunda mediante la simple anteposición del poseedor.

2.6. Léxico

En rigor, no conocemos con seguridad el significado de ninguna palabra ibérica. No obstante, hay unas pocas que, debido a su frecuencia y a su aparición en un contexto favorable, permiten un atisbo de comprensión. Así, la palabra **salir**, que aparece a menudo en monedas de plata, pero también en plomos y seguida de numerales, se ha relacionado tradicionalmente con el vasco *zilar* (“plata”), ecuación que rechaza Michelena⁵⁴. Se mantiene, sin embargo, el acuerdo general en considerar su significado como próximo a “moneda” o similar.

Hemos mencionado también el caso de **eban**, que Untermann considera verbo de dedicación, mientras que Velaza⁵⁵ considera que significa “hijo”.

También hemos hablado ya de **ekiar**, frecuente en inscripciones sobre cerámica de lujo y otros objetos suntuarios, como una falcata de Sagunto o el mosaico de Caminreal. Sobre su interpretación el acuerdo es ahora general, manteniéndose divergencias únicamente respecto a la categoría de la palabra. Como verbo podría significar “hecho”, y como sustantivo “obra”.

Una palabra frecuente en plomos es **iunstir**. Su frecuente aparición al principio o final de plomos, que probablemente son cartas comerciales, ha llevado a Velaza⁵⁶ a sugerir que se trata de una forma de saludo o similar,

⁵³de Hoz (2001a, p. 350)

⁵⁴Michelena (1985c, p. 353)

⁵⁵Velaza (1996a)

⁵⁶Velaza (2001a, p. 644)

aunque también aparece como elemento antroponímico.

Nos referiremos, por último, a la palabra **iltír**, que se suele considerar equivalente a “ciudad” (aparece, por ejemplo, en la leyenda monetaria **iltírta**, que corresponde a Ilerda), y que tradicionalmente se ha relacionado con el vasco antiguo *ili*, con el mismo significado.

Untermann, en el primer volumen de MLH III, presenta una relación de elementos lexicales que se repiten con cierta frecuencia, entre los que se cuentan los ya mencionados y una docena más de los que apenas hay indicios para conjeturar su significado, de los que hay que destacar *kutur*, que aparece a menudo junto a formas del paradigma **bitéroke**, y que fue leído erróneamente por Pío Beltrán como *gudua*, dando lugar a una larga serie de especulaciones vascoiberistas.

2.7. Onomástica

Este es el aspecto mejor conocido de la lengua ibérica, gracias especialmente al descubrimiento del Bronce de Ascoli, en el que se concede la ciudadanía romana a los integrantes de un escuadrón de caballería reclutado en **saltuie** (Zaragoza), conocido como *Turma Salluitana*. La relación de nombres de los jinetes y sus padres, todos ellos reclutados en poblaciones del valle del Ebro, ha permitido establecer que los nombres ibéricos están compuestos por dos elementos, generalmente bisilábicos.

El estudio más completo de la onomástica ibérica es el llevado a cabo por Untermann en el primer volumen de MLH III, donde establece un completo repertorio de nombres personales ibéricos. El grado de seguridad de la identificación es variable. Los más seguros son los atestiguados en las fuentes latinas, como el Bronce de Ascoli, seguidos por los que aparecen en contextos claros, como inscripciones funerarias y de propiedad sobre cerámica, especialmente si están formados por elementos onomásticos atestiguados ya en fuentes latinas. Con menor seguridad se identifican nombres en otros contextos que estén formados por elementos ya documentados. Los más inciertos serían aquellos que parecen responder a la estructura de un nombre propio ibérico, pero sólo uno de los elementos está documentado. Pero si estos últimos aparecen en un contexto que apoya claramente su interpretación como nombre personal, permiten incorporar nuevos elementos al repertorio.

2.8. Tipología

Algunos estudios recientes⁵⁷ han explorado el camino de la tipología lingüística como un recurso más para ayudar a desvelar la estructura de la lengua ibérica, y especialmente para verificar la verosimilitud o no de las hipótesis que se van proponiendo. Así, en su trabajo citado, de Hoz se siente obligado a justificar que, en la inscripción musiva de Andelos, lo que él considera como verbo no vaya en posición final, pues previamente ha establecido que el ibérico era una lengua con orden básico SOV.

Otra aportación importante del trabajo de de Hoz es que el ibérico sería una lengua no sólo aglutinante, sino además incorporante en un alto grado, es decir, que podría incorporar en el núcleo verbal no sólo elementos pronominales (como en el vasco), sino también nominales.⁵⁸

2.9. Situación del ibérico frente a otras lenguas

Nos referiremos aquí al estatus lingüístico de la lengua ibérica respecto a las demás lenguas indígenas peninsulares, por una parte, y respecto al latín, por otra. Para el primer aspecto es de referencia obligada la hipótesis de de Hoz,⁵⁹ quien considera que el ibérico fue, en la mayor parte del territorio donde se documenta su uso, no la lengua hablada mayoritariamente, sino más bien una “lengua vehicular”, utilizada como *lingua franca* para facilitar los intercambios comerciales. Esto parece especialmente claro en el sur de Francia, donde nombres galos y ligures aparecen en textos ibéricos,⁶⁰ lo que para Untermann revelaría una población mayoritariamente ligur con elementos galos e ibéricos. De Hoz se apoya también en la existencia de un fondo onomástico no ibérico que se refleja en los grafitos sobre cerámica de Ullastret y de Azaila. También en territorio vascón se conoce alguna inscripción ibérica (como el mosaico de Andelos o el bronce de Aranguren), a pesar de que el vasco debía ser la lengua hablada mayoritariamente (sin excluir otras lenguas como el celtibérico). Las inscripciones aquitanas muestran una serie de elementos onomásticos comunes con el ibérico, que llevaron a Michelena⁶¹ a hablar de un “pool onomástico” común a ibérico y aquitano.

⁵⁷Especialmente de Hoz (2001a). Ver también Ballester (2001a)

⁵⁸de Hoz (2001a, p. 347-348)

⁵⁹de Hoz (1993)

⁶⁰Untermann (1969)

⁶¹Michelena (1985c, p. 355)

Si respecto a las demás lenguas indígenas el ibérico tenía una posición dominante, imponiéndose como lengua escrita y extendiendo el uso de su sistema onomástico, la relación respecto al latín es justamente la inversa, lo cual se manifiesta de modo especial en la epigrafía. A este respecto, Mayer y Velaza⁶² hablan de “epigrafía ibérica sobre soportes típicamente romanos”. Para estos autores, el uso de la piedra como soporte epigráfico responde mayoritariamente a la influencia romana, especialmente en inscripciones funerarias, y no sólo se copia el soporte sino también la disposición del texto, el estilo de los signos (que recuerda a veces a la escritura capital romana) e incluso el formulario utilizado. Incluso la expresión de la filiación, que como hemos visto estaría representada para Velaza por el término **eban**, sería también un ejemplo de influencia romana.⁶³

2.10. Vascoiberismo y comparación lingüística

El ibérico sigue siendo hasta hoy día una lengua sin parientes conocidos. Queda sin embargo en pie la cuestión del vascoiberismo. Como es sabido, esta doctrina, establecida por Humboldt,⁶⁴ defendía que el vasco era una forma moderna de ibérico. Tras el desciframiento del alfabeto por Gómez Moreno, esta teoría cayó en descrédito, ante la imposibilidad de traducir ningún texto ibérico mediante el vasco, a pesar de los frecuentes intentos en este sentido, obra generalmente de aficionados.

Sin embargo, un vascoiberismo más moderado, entendido en el sentido de que vasco e ibérico son lenguas emparentadas con suficiente proximidad como para utilizar el primero en el desciframiento del segundo, sigue teniendo influencia hoy día entre algunos especialistas, especialmente discípulos de Fletcher, como Luis Silgo, pero también otros, como Rodríguez Ramos o Marques de Faria. Así, Silgo⁶⁵ identifica formas verbales ibéricas, e incluso las clasifica en sintéticas o analíticas (como el verbo vasco moderno), basándose en la pura semejanza formal, sin nada que apoye en el propio ibérico un valor verbal.

Parece sin duda más prudente renunciar a la comparación vasco-ibérica hasta que no se pueda establecer primero el significado de la palabra ibérica a comparar por medios independientes. Es lo que ha hecho de Hoz en

⁶²Mayer (1993)

⁶³Ver también Mayer (2002)

⁶⁴Humboldt (1821)

⁶⁵Silgo Gauche (1996b)

un trabajo reciente,⁶⁶ en el que, a partir del propio ibérico, llega a establecer el valor del sufijo *-sken*, que coincide básicamente con el que se le atribuía a partir de la comparación con el vasco. Pero, como señala el autor, tan sólo el sufijo de genitivo *-en* coincide propiamente con el vasco, y no está claro que ésta fuera la forma originaria en esta lengua. De hecho, Gorrochategui y Lakarra⁶⁷ defienden la antigüedad de la forma *-e* que aparece en la declinación pronominal.

⁶⁶de Hoz (2002)

⁶⁷Gorrochategui y Lakarra (1996)

Capítulo 3

Los archivos de textos ibéricos

El programa buscador de expresiones regulares serviría de poco sin una base de textos ibéricos lo más completa posible y con un formato adecuado. A continuación se explica cómo se ha elaborado esta base textual. Es importante conocer en detalle el formato en que están introducidos los textos, por simple que sea, para poder utilizar adecuadamente el buscador.

3.1. Origen de los textos

La mayor parte de los textos se ha obtenido de la obra de Untermann *Monumenta Linguarum Hispanicarum*. En el primer volumen, dedicado a las leyendas monetales, una buena parte corresponden a leyendas ibéricas. El segundo recoge los textos procedentes del sur de Francia, y el tercero los de la Península Ibérica.

En MLH se recogen la práctica totalidad de inscripciones publicadas antes de 1990. Las aparecidas posteriormente se hallan dispersas en una larga serie de publicaciones, que ha recogido exhaustivamente de Hoz.¹ La mayoría de textos de cierta extensión o importancia van siendo publicados por Velaza en sus *Crónicas epigráficas*, aparecidas a partir del *VI Coloquio sobre lenguas y culturas prerromanas de la Península Ibérica*, y últimamente en la revista *Palaeohispanica*. Para la zona catalana destacan los trabajos de M^a Isabel Panosa, que recogen gran número de inscripciones, en su mayoría grafitos breves sobre cerámica, dispersas por publicaciones arqueológicas o inéditas.

Los textos se han introducido tecleándolos directamente en un editor de textos, y repasándolos varias veces para evitar errores. Se ha comprobado que la mayoría de casos en que se detecta un error suele ser por intercambiar una

¹de Hoz y Luján (2001)

vibrante o silbante por la otra, de forma que es aconsejable tenerlo presente en las búsquedas y utilizar clases de caracteres, como se explicará más adelante.

En algunos casos existen varias lecturas de un mismo texto, según diferentes autores. Generalmente se ha optado por recoger la versión de MLH, salvo en textos posteriores, en que se ha optado por la lectura que parecía más fiable, por la mayor coherencia de una versión con lo conocido por otras inscripciones, o, cuando ha sido posible, por la observación directa de la fotografía. Un ejemplo muy claro es el plomo de Ampurias (C.1.24), del que hay tres lecturas enormemente divergentes, la de su editor,² la de Panosa³ y la de Untermann.⁴ En este caso ha sido la última la elegida, por ser la que presenta más palabras comparables con otros textos y muestra mayor coherencia interna. Además, la fotografía que presenta Sanmartí permite verificar personalmente la lectura.

En este y otros casos se presentan las diferentes variantes de lectura bajo el encabezado V1, V2, etc, seguido por el número de línea en que se documentan variantes y a continuación el texto de la línea en cuestión. Si el plomo tiene dos caras escritas se precede el número de línea por A o B. En la bibliografía se indica con un asterisco la obra de la que se ha tomado la versión elegida como principal (o la única en la mayoría de casos), y con V1, V2, etc. la referencia de las distintas variantes de lectura.

En las variantes de lectura se respetan los usos del autor en cuanto a la transcripción (como el uso de oclusivas sonoras).

3.2. Formato de los textos

En un principio se introdujeron todos los textos en un archivo único, que posteriormente se fragmentó, con ayuda de un pequeño programa escrito al efecto, en tantos archivos como textos, nombrando los archivos con la nomenclatura del texto en MLH, que como es sabido consta de una letra mayúscula que indica la zona geográfica (A para todas las monedas, B para el sur de Francia, C para la Cataluña costera, D para el interior de Cataluña y Aragón, E para la zona del Bajo Aragón, F para Castellón y Valencia, G para Alicante, Murcia y Albacete, H para Andalucía oriental), un número correspondiente al yacimiento, y un segundo número que identifica a cada inscripción dentro de un mismo yacimiento.

Algunos textos publicados con posterioridad a MLH se citan ya con la nomenclatura que tendrán en los suplementos que han de aparecer, pero en

²Sanmartí-Grego (1988)

³Panosa (1999)

⁴Untermann (1996)

un gran número de textos no es así, y para éstos se ha adoptado una nomenclatura provisional, que consiste en seguir la serie iniciada por Untermann para la lápida de Cagliari X.0.1, continuando con X.0.2, etc.

Los textos están en formato ASCII, con lo cual se facilita que cualquier usuario desde cualquier ordenador conectado a Internet pueda teclear cualquier cadena de texto sin instalar tipografías especiales. La contrapartida es la limitación en los símbolos utilizables, de modo que no pueden usarse la **ś** o **ř** en la transcripción. En su lugar se ha optado por las mayúsculas en aquellos signos que lleven diacrítico, tal como hace Marques de Faria en algunas de sus publicaciones de la *Revista Portuguesa de Arqueología*. Así, se transcribe **ř** por R, **ś** por S y **m̄** por M. Más difícil de resolver es el problema de los signos usados en epigrafía para indicar caracteres dudosos, como el punto debajo de la letra o el subrayado. Cualquier solución afectaría a las búsquedas (una búsqueda literal de una palabra no obtendría resultados que tuvieran alguno de esos signos), por lo que se ha optado por prescindir de dichos signos auxiliares. Hay que insistir en que este programa no pretende ser un corpus de referencia, sino tan sólo un auxiliar para la investigación lingüística, por lo que en última instancia hay que comprobar siempre los textos en sus publicaciones originales.

Sí se han mantenido, en cambio, los espacios en blanco, corchetes, asteriscos o guiones que aparecen en las transcripciones.

Por las mismas razones prácticas se ha unificado la transcripción en minúsculas normales (salvo las tres mayúsculas mencionadas), frente a la acostumbrada en negrita para epicórico, cursiva en grecoibérico y mayúsculas para el latino. Se reservan las mayúsculas para los fragmentos de texto latino que aparecen en algunos textos (bilingües o monedas), que se han mantenido por su importancia para la interpretación de los mismos.

Los textos se han transcrito con los saltos de línea originales. Hay que tener en cuenta que en Linux, como variante de Unix que es, los saltos de línea difieren de MSDOS en su formato interno (se expresan por un único carácter `\n`), pero se visualizan correctamente en cualquier navegador, como visualizadores universales que son.

Por último, se ha optado por transcribir como X los signos noroccidentales de valor desconocido (como el que aparece en el Plomo del Castell de Palamós), & para los del signario sudibérico de valor desconocido, y # para los numerales de valor desconocido, salvo el de forma I, que se transcribe así.

3.3. Información sobre los textos

Como hemos visto, los archivos de textos ibéricos no contienen nada más que el texto, ni siquiera su nombre (que aparece tan sólo como nombre de archivo). La única excepción serían las indicaciones V1, V2, etc. que encabezan las variantes de lectura, y los números de línea. Nada de esto interfiere en las búsquedas (no se utiliza la V en las transcripciones). Para poder presentar información complementaria de cada texto, especialmente la bibliografía, se ha optado por crear una serie paralela de archivos con el mismo nombre, pero situados en otro directorio, cuyo contenido se presenta automáticamente antes que el texto en la ventana del navegador. Aquí se ha incluido el nombre del texto (cuando el texto es más conocido por otro nombre, como “Plomo de Alcoy” o “Estela de Binéfar”), el soporte (sucintamente: piedra, cerámica, plomo, etc.), el alfabeto utilizado (sólo si no es el ibérico noroccidental), la localización y la bibliografía.

Capítulo 4

La interfaz CGI

4.1. Introducción

Un CGI (siglas de Common Gateway Interface) es un programa que se ejecuta en un servidor web, y que interactúa con el usuario generalmente a través de un servidor web. Es uno de los procedimientos más simples y habituales para dotar de interactividad a una página web. Tiene la ventaja de que, como el programa se ejecuta en el servidor, no depende de la potencia del ordenador desde el que se conecta el usuario, a diferencia de otros métodos como son el código Javascript o los “applets” de Java.

Cuando nos conectamos a una página web que contiene un formulario de entrada de datos y un botón para enviarlos (es el caso típico de los buscadores), la acción de pulsar el botón de envío hace que el servidor procese los datos que hemos introducido y nos devuelva el resultado, generalmente en una nueva página.

4.2. El servidor web

El programa cgi reside, como hemos dicho en el ordenador que actúa de servidor. Para actuar como tal, en él debe hallarse instalado un programa específico, conocido como “servidor web”, que es el que atiende las peticiones de los usuarios y les sirve las páginas web que solicitan. Entre sus capacidades está la de ejecutar, a petición del usuario, programas cgi, que deben estar localizados en determinados subdirectorios. Así, en Linux, una localización típica es `/home/httpd/cgi-bin`, mientras que las páginas web propiamente dichas pueden estar en `/home/httpd/html`. Cuando se trata de páginas personales de un usuario, los subdirectorios mencionados “cuelgan” de su directorio personal, por ejemplo `/home/usuario/public_html`

y `/home/usuario/public_html/cgi-bin`. Las páginas personales suelen ser accesibles mediante direcciones tipo

`http://www.nombre-servidor.org/~usuario`.

Un servidor web puede instalarse en principio en cualquier ordenador que cumpla unos requisitos mínimos, aunque no esté conectado a una red o a internet. Es decir, puede servir páginas locales al usuario que utiliza el ordenador en que está instalado, y ejecutar cgi's. Es así como he desarrollado mi programa en un ordenador personal, aunque después lo haya instalado en un servidor conectado a internet para así hacerlo accesible a todos. En mi caso opté por el servidor web conocido como Apache, uno de los más extendidos, y que además es un programa de código fuente abierto, y puede conseguirse gratuitamente. Desarrollado inicialmente para sistemas UNIX, en la actualidad existen versiones para los principales sistemas operativos.

La configuración de Apache, como es tradicional en Linux y en cualquier sistema operativo UNIX, reside en varios archivos de texto. En la versión que he utilizado, estos archivos residen en el directorio `/etc/httpd/conf`, y el principal archivo de configuración es el llamado `commonhttpd.conf`. Las líneas más importantes de este archivo por lo que se refiere a la ejecución de mi programa son las siguientes:

```
<Directory /home/*/public_html>
    AllowOverride All
    Options MultiViews -Indexes Includes FollowSymLinks
    Order allow,deny
    Allow from all
</Directory>

<Directory /home/*/public_html/cgi-bin>
    Options +ExecCGI -Includes -Indexes
    SetHandler cgi-script
</Directory>
```

El servidor Apache no puede buscar las páginas que se le soliciten en cualquier directorio, sino únicamente en aquellos que tenga señalados expresamente en el archivo de configuración. Entre ellos aparecen los dos que acabo de señalar. En el primero, `/home/*/public_html`, se permite el acceso a todos los usuarios (`Allow from all`). Los directorios personales de cada usuario de un sistema UNIX están siempre situados bajo el directorio `/home`. Con el asterisco (carácter comodín) se representa cualquier directorio de usuario. Dentro de cada directorio personal de usuario puede haber un directorio `public_html`, que es en el que deben estar las páginas web para que sean

accesibles al servidor Apache. En mi caso, la ruta completa a mi página sería `/home/edu/public_html/iberico.html`.

Bajo el directorio `public_html` puede haber un directorio `cgi-bin`, entre cuyas opciones vemos que se indica `ExecCGI+`, es decir, el permiso para ejecutar programas `cgi` que residan en él. En mi caso es aquí donde está ubicado el programa `buscaiberico.cgi` y los archivos de texto que contienen los textos ibéricos.

4.2.1. El programa `cgi`

En cuanto al programa `cgi` propiamente dicho, en principio puede estar escrito en cualquier lenguaje capaz de leer la entrada estándar y escribir en la salida estándar, es decir, prácticamente todos. Pero por diversas razones sólo unos pocos lenguajes suelen usarse a tal fin. En general se trata de lenguajes interpretados, que no necesitan compilar el código fuente, sino que el programa consiste en un archivo de texto cuyo código traduce a binario el programa intérprete. Esto los hace más lentos que los lenguajes compilados, pero ello no suele suponer un problema, pues los `cgi`'s no acostumbran a ser programas extensos. A menudo se limitan a ejecutar una serie de órdenes en el servidor, como los archivos `batch` (con la extensión `.bat`) en MS-DOS, lo que explica la frecuencia con que se escriben en `shell-script` (el lenguaje propio de la interfaz de comandos en sistemas operativos tipo UNIX). Pero sin duda el lenguaje favorito de los programadores de `cgi`'s (en competición últimamente con el lenguaje PHP) es el lenguaje PERL, cuyas características lo hacen especialmente apto para el análisis de textos (no en vano fue creado por un lingüista).

La importancia del tratamiento de textos para la elaboración de `cgi`'s no es evidente a primera vista, pero es crucial. En primer lugar, el navegador del usuario envía los datos del formulario en forma de `ristra`. Es decir, si el formulario consta de varias ventanas de texto, casillas de verificación, menús desplegables, etc., todo el texto que introduzca el usuario, además del texto correspondiente a las opciones o entradas de menú seleccionadas, se envía seguido en forma de una única cadena de texto, en la que las porciones de texto correspondientes a cada elemento del formulario se separan mediante caracteres como signos `+` o interrogantes. Lo primero que debe hacer el programa `cgi` es procesar la cadena de texto que recibe como entrada, separándola en cadenas a partir de los caracteres separadores, reconstruyendo los caracteres especiales (vocales acentuadas, etc.) que el navegador envía codificados, y almacenándolo todo en una única variable tipo "hash" (array asociativa, o tabla indexada), como si dijéramos una tabla con dos columnas, la primera con el nombre del campo del formulario y la segunda con el valor introducido

por el usuario.

En segundo lugar, es frecuente que el programa cgi deba cotejar algunas de las cadenas de texto recibidas con archivos de textos mayores, y devolver una porción determinada de los mismos. En el caso de mi programa, la cadena o expresión regular introducida por el usuario se busca a lo largo de todos los archivos de texto que contienen los textos ibéricos, enviando al navegador del usuario sólo aquellos que contienen la cadena buscada, transformándolos previamente de modo que ésta aparezca resaltada, lo que consigue insertándole las marcas HTML `` para ponerla en negrita y `` para que aparezca en rojo.

4.3. Elementos de un cgi

Como hemos dicho, un cgi recibe los datos de entrada generalmente a partir de un formulario incorporado en una página web. Básicamente hay dos opciones para hacer ésto: una es hacer una página web (archivo con la extensión .html) que contenga el formulario, y un programa cgi (archivo de texto con extensión .cgi) que es invocado por el usuario al pulsar un botón en el formulario. La otra es elaborar un único archivo, con la extensión .cgi, que crea dinámicamente el formulario. En mi caso he optado por el primer método, por su mayor simplicidad y sobre todo por ser más fácil de comprender su funcionamiento.

Aparte del archivo que contiene el formulario, que comentaré en el siguiente apartado, todo cgi suele tener unos elementos comunes, relacionados con la necesidad de interpretar los datos suministrados por el formulario, y de devolver la salida en formato de página web:

- Una porción de código, que suele ir hacia el principio, que analiza la ristra de texto enviada por el formulario y obtiene de ella una variable de tipo “hash”, que viene a ser una tabla de dos columnas que empareja claves en una columna y valores en otra, siendo las claves los nombres de los campos del formulario (por ejemplo “Nombre”, “Apellido”), y valores el texto introducido por el usuario (por ejemplo, “Juan”, “Pérez”). El detalle de cómo se realiza ésto se comenta en el apartado dedicado a la cgi-lib.pl.
- Una porción de código, que suele ir hacia el final, que imprime en la salida estándar (es decir, envía al servidor, quien a su vez lo envía al navegador del usuario) un texto, habitualmente compuesto por las marcas html necesarias para que el usuario obtenga una página web como respuesta, además de cualquier otro elemento, como texto, imágenes,

y generalmente algún elemento dinámico, fundamentalmente variables que dependerán del contenido del formulario. En mi programa esta porción de salida es todo lo que va desde la línea `print <<ENOFTEXT` hasta `ENDOFTEXT`. Siguiendo con el ejemplo del punto anterior, un cgi podría devolver lo siguiente:

```
<HTML>
<BODY>
Su nombre es $nombre $apellido.
</BODY>
</HTML>
```

4.4. El formulario

Como es sabido, una página web no es otra cosa que un archivo de texto, con la extensión `.html`,¹ que puede contener texto sin formato o más frecuentemente texto cuyo formato se indica mediante una serie de marcas (“tags” en inglés). Así, la marca `<HTML>` indica el principio de una página web, y `</HTML>` su final. Algunas marcas pueden contener atributos. Así, la marca o directiva `` puede contener atributos como `color` o `size`. Por ejemplo, `` escribiría el texto en rojo hasta que aparezca la marca ``.

Un formulario viene enmarcado por la directiva `<FORM>`, cuyo principal atributo es `action`, que indica la acción que debe realizarse con los datos al ser enviados, y que suele ser la ejecución de un cgi. Presento a continuación el formulario contenido en mi página, explicando luego sus elementos:

```
<form action="/cgi-bin/buscaiberico.cgi" method="post">

<h3>
Introduca la expresi&oacute;n:</h3>
<input type="text" name="expresion1" size="40">
<br>&nbsp;
<p>
<input type="text" name="expresion2" size="40">
<p>
Mostrar nombres personales: <input type="checkbox" name="np" value="si" checked>
```

¹.htm en sistemas operativos de Microsoft, por herencia del formato de nombres de archivo en MSDOS, que no permitía extensiones de más de tres caracteres.

```
<p>
Descartar textos de menos de <input type="text" name="num" size="2">
caracteres [en transcripción]
<p>
<input type="submit" value="enviar">
<p>
</form>
```

Como vemos, la acción que se realiza es la ejecución del cgi situado en el directorio cgi-bin, que está situado bajo el directorio actual (expresado por el punto), que es desde donde se le llama, es decir, donde reside la página que contiene el formulario. El atributo `method` suele contener los valores `get` o `post`. Según el método elegido variará la forma de obtener los datos por parte del cgi.

Además de las directivas no específicas de los formularios, como `<h3>`, `
`, `<p>` (texto destacado en mayor tamaño y negrita, salto de línea y nuevo párrafo, respectivamente), podemos observar una directiva `<input>` que puede tener diversos atributos. El más importante es el atributo `type`, que determina el tipo de elemento de formulario. Aquí se han utilizado tres: `text` (ventana de texto), `checkbox` (casilla de verificación) y `submit` (botón para enviar los datos al cgi). Los dos primeros llevan un atributo `name` que servirá de clave para identificar el valor introducido por el usuario en la tabla en que se almacenan finalmente los datos.

En una página web puede haber varios formularios. En mi caso, hay un segundo formulario que sirve para buscar textos ibéricos a partir de su nombre según la nomenclatura de MLH.

4.5. La cgi-lib.pl

Como se ha mencionado anteriormente, la primera labor que debe llevar a cabo un programa cgi es analizar la ristra de texto enviada por el navegador y almacenarla en una variable tipo hash (array asociativa) que empareja cada clave (nombre del campo del formulario) con su valor. Dado que se trata de una tarea común a cualquier cgi, existen bibliotecas de funciones,² es decir, archivos que contienen funciones predefinidas, que simplifican el trabajo. Las más conocidas son CGI.pm y cgi-lib.pl. La primera es en realidad un módulo del lenguaje PERL, y consiste en un extenso paquete de funciones que utilizan la programación orientada a objetos, y que permiten crear también cualquier

²A menudo se traduce incorrectamente el inglés “library” por “librería”.

elemento del formulario. La segunda es la que he utilizado para mi programa, ya que su simplicidad permite entender con relativa facilidad su código.

Para que las funciones contenidas en la `cgi-lib.pl` queden accesibles al programa `cgi`, éste debe contener la siguiente línea:

```
require "cgi-lib.pl";
```

Para mi programa sólo he utilizado dos funciones: `&ReadParse` (que a su vez invoca a las funciones `&MethGet` y `&MethPost`) y `&PrintHeader`. Esta última es muy simple, simplemente imprime la línea que indica al servidor Apache que está ante un documento HTML:

```
"Content-type: text/html\n\n"
```

Aquí `\n\n` indica simplemente dos saltos de línea, es decir, dos líneas en blanco, antes de empezar con el contenido de la página propiamente dicho.

La función `&ReadParse` es más compleja e importante. Es la que analiza la ristra de texto enviado por el servidor, que será accesible después para el programa a través de la variable global `*input`. La dificultad del proceso radica en que el navegador del usuario envía los datos del formulario en forma de una única ristra de texto sin espacios en blanco con la forma

```
Nombre_variable1=Valor1&Nombre_variable2=Valor2...
```

Además, sustituye los espacios en blanco que pueda haber introducido el usuario (para separar palabras en una ventana de texto, por ejemplo), por signos `+`, y codifica en código hexadecimal los caracteres no ASCII, como pueden ser las vocales acentuadas, la `ñ`, etc. Todo esto debe devolverse a su estado original para ser utilizado por el `cgi`, y esto es lo que hace la función `&ReadParse`. Su código es el siguiente:

```
sub ReadParse {
    local (*in) = @_ if @_;
    local ($i, $key, $val);

    # Read in text
    if (&MethGet) {
        $in = $ENV{'QUERY_STRING'};
    } elsif (&MethPost) {
```

```

    read(STDIN,$in,$ENV{'CONTENT_LENGTH'});
}

@in = split(/[&;]/,$in);

foreach $i (0 .. $#in) {
    # Convert plus's to spaces
    $in[$i] =~ s/\+/ /g;

    # Split into key and value.
    ($key, $val) = split(/=/, $in[$i], 2); # splits on the first =.

    # Convert %XX from hex numbers to alphanumeric
    $key =~ s/%(..)/pack("c",hex($1))/ge;
    $val =~ s/%(..)/pack("c",hex($1))/ge;

    # Associate key and value
    $in{$key} .= "\0" if (defined($in{$key}));
    # \0 is the multiple separator
    $in{$key} .= $val;
}

return scalar(@in);
}

```

Básicamente hace lo siguiente:

- Almacena la ristra de texto en una variable escalar `$in`. Por ejemplo,

```
$in = "Nombre_variable1=Valor1&Nombre_variable2=Valor2"
```

- Convierte `$in` en un array `@in`, usando como elemento separador (es decir, como indicador de por dónde se debe cortar la cadena) el signo `&`. Así, el ejemplo anterior quedaría

```
@in = "Nombre_variable1=Valor1,Nombre_variable2=Valor2"
```

Es decir, un array o variable tipo lista de dos elementos, separados por la coma.

Los siguientes puntos se realizan para cada uno de los elementos del array:

- Convierte los signos + en espacios en blanco, haciendo uso de la función de sustituir expresiones regulares `s`.
- Almacena en dos variables `$key` y `$val` respectivamente el nombre del campo del formulario y su contenido, usando el primer signo = como separador. Así, para el primer elemento del array tendríamos que
`$key = "Nombre_variable1" $val = "Valor1"`
- Transforma los códigos hexadecimales en alfanuméricos.
- Asocia la variable `$key` con su valor `$val`, de modo que cuando esto se haya hecho con todos los elementos del array obtenemos un “hash” o array asociativa, que podríamos representar así:

Nombre_variable1	Valor1
Nombre_variable2	Valor2

Respecto a las funciones `&MethGet` y `&MethPost` su única finalidad es verificar si el navegador envía los datos según el método `GET` o `POST`, para así leerlos de una u otra forma.

Capítulo 5

El programa buscaiberico.cgi

5.1. Uso del programa

Con lo visto hasta ahora podemos entrar en detalle en el programa buscaiberico.cgi. Su función principal es buscar una o dos expresiones regulares introducidas por el usuario a lo largo de un conjunto de archivos de texto, cada uno de los cuales contiene un texto ibérico, y cuyo nombre coincide con el identificativo del texto según la nomenclatura de MLH. Como resultado devuelve una página web que contiene el texto completo de los archivos que contengan una cadena que encaje con la expresión regular buscada, que aparecerá resaltada en rojo y negrita (esto último para mantener la distinción en caso de imprimir el resultado con impresora en blanco y negro). En la parte superior aparece de nuevo la expresión regular introducida por el usuario, y su equivalencia real (es decir, desarrollando las abreviaturas previstas). En la parte superior, y de nuevo en la inferior, aparece el número de secuencias que encajan.

Cada texto aparece precedido por una breve información de soporte, localidad y bibliografía. Esta información reside en archivos independientes de los que contienen los textos, para evitar que su contenido sea interpretado por el buscador como texto ibérico. Sí coinciden en un mismo archivo los textos ibéricos y las variantes de lectura, para permitir que estas sean localizadas por el buscador. El inconveniente es que algunas secuencias pueden localizarse dos o más veces, si se repiten en el texto y en las variantes de lectura, restando valor al contador, lo que puede ser grave si éste se usa con fines estadísticos. En una próxima versión podría corregirse incluyendo las variantes de lectura en archivos independientes, y ofreciendo al usuario la opción de que sean analizados por el buscador.

Además, si el usuario marca la opción correspondiente del formulario,

subraya los nombres personales ibéricos que identifique en los textos a devolver. También puede descartar los textos que contengan un número de caracteres menor que el que indique el usuario. Esto se ha hecho debido a la abundancia de textos muy breves, principalmente grafitos sobre cerámica, que no aportan ninguna información gramatical, y más bien pueden restar claridad al resultado. Así, una búsqueda de la secuencia **ban** como sufijo o prefijo nos daría numerosos ejemplos en que parece como cadena única, alargando inútilmente la página-resultado.

De las dos ventanas de texto previstas para introducir la expresión, sólo la primera es necesaria. En caso de usar también la segunda, sólo se mostrarán textos que contengan coincidencias con las dos expresiones regulares introducidas. Más adelante se verán en detalle las expresiones regulares.

Naturalmente, los resultados de la búsqueda pueden imprimirse, copiarse y pegarse en un procesador de textos, o guardados en disco con la opción “Guardar como...” o similar del navegador.

Un segundo formulario, independiente del primero, permite localizar textos concretos a partir de su nomenclatura MLH. Su código es muy simple, y no se comentará aquí, pudiendo consultarse en el apéndice correspondiente. Su uso también es muy simple: basta con introducir el nombre del texto (por ejemplo, C.1.1.24) y pulsar el botón “enviar”.

5.2. Descripción del código

A continuación se presenta el código del programa, añadiéndole números de línea para facilitar las referencias, aunque en su mayoría se irán transcribiendo de nuevo a medida que se comenten.

```
1 #!/usr/bin/perl
2 require "cgi-lib.pl";
3 MAIN:
4 {
5 &ReadParse(*input);
6 open (RESULTADO, ">resultado-iberico")||die "no puedo abrir RESULTADO";
7 system("ls *.*.* > listaficheros")==0;
```

```

8 open (LISTA, "listaficheros")||die "no puedo abrir LISTA";

9 @lista_ficheros = <LISTA>;

10 close (LISTA);
11     $expresion1 = $input{'expresion1'};
12     $expresion1 =~ s/([adegiklmoprtruRSMbns])(?!\\W)/$1\\n*/g;
13     $expresion2 = $input{'expresion2'};
14     $expresion2 =~ s/([adegiklmoprtruRSMbns])(?!\\W)/$1\\n*/g;
15     $numerofichero = @lista_ficheros;
16     if ($input{'num'}) {$num = $input{'num'}} else {$num = "1"}
17     $i=0;
18     $e=0;
19     $expresion1 =~ s/C/[bptdkgmMnrRsSl]/g; #consonantes
20     $expresion1 =~ s/V/[aeiou]/g; #vocales
21     $expresion1 =~ s/T/[td]/g; #dentales
22     $expresion1 =~ s/K/[kg]/g; #velares
23     $expresion1 =~ s/P/[pbt dkg]/g; #oclusivas
24     $expresion1 =~ s/Z/[sS]/g; #silbantes
25     $expresion1 =~ s/Q/[rR]/g; #vibrantes
26     $expresion1 =~ s/Y/([btdkgmMnsSlrR]?[aeiou][iu]?[nmMlrR]?[sS]?)/g;
        #sílabas
27     $expresion1 =~ s/L/[nmMlrR]/g; #sonantes
28     for ($indice = 0; $indice < $numerofichero; $indice++) {
29         open(TEXTOS, "<@lista_ficheros[$indice]");
30         open(TEXTINF, "<./inf/@lista_ficheros[$indice]");
31         undef $/;
32         $texto = <TEXTOS>;
33         $textinf = <TEXTINF>;
34         if ($texto =~ /.{$num}/) {
35             while ($texto =~ /$expresion1/g) {
36                 $i++;
37             }
38             if ($expresion2 =~ /.+/) {
39                 while ($texto =~ /$expresion2/g) {
40                     $e++;
41                 }
42             }
43             if ($texto =~ /$expresion1/g) {
44                 $texto =~ s/$expresion1/<FONT COLOR=RED><STRONG>$$&

```

```

45         <\STRONG><\FONT>/gs;
46     if ($texto =~ /$expresion2/g) {
47         $texto =~ s/$expresion2/<FONT COLOR=RED><STRONG>$&
48         <\STRONG><\FONT>/gs if ($expresion2 =~ /.+ /);
49     if ($input{'np'}){
50         open (NOMBRES, "<nombres")||
51         die "no puedo abrir nombres";
52         $/ = "\n";
53         @nombres = <NOMBRES>;
54         undef $/;
55         $nombres = @nombres;
56         for($x = 0; $x < $nombres; $x++){
57             $nombre = @nombres[$x];
58             $nombre =~ s/\n*/ /g;
59             $nombre =~ s/[abdegiklmnoprstuMRS](?!\/)/
60             $&\\s*\\n*\/g;
61             if ($texto =~ /$nombre/) {
62                 $texto =~ s/$nombre/<U>$&<\U>/g;
63             }
64         }
65     close (NOMBRES);
66 }
67 print RESULTADO ("<h3>@lista_ficheros[$indice]</h3>");
68 print RESULTADO ("<textinf");
69 print RESULTADO ("<texto");
70 print RESULTADO ("\n\n");
71 }
72 }
73 }
74 close (TEXTOS);
75 close (RESULTADO);
76 open (RESULTADO, "<resultado-iberico")||die "no puedo abrir resultado";
77 @resultado = <RESULTADO>;
78 print &PrintHeader;
79 print <<ENDOFTEXT;
80 <html>
81 <head>
82 <title>Resultado de la b&uacute;squeda:</title>
83 </head>
84 <body>

```

```

82         <h1>Resultados de la búsqueda</h1>
83         <h2>Expresión: $input{'expresion1'},
           $input{'expresion2'}</h2> que
           equivale(n) a $expresion1, $expresion2.
84         <h2>Total: $i, $e </h2>
85         <pre>@resultado</pre>
86         <h2>Total expresión 1: $i</h2>
87         <h2>Total expresión 2: $e</h2>
88 ENDOFTEXT
89         print &HtmlBot;
90 }

```

La primera línea (`#!/usr/bin/perl`) es una llamada al intérprete de PERL. El servidor Apache por sí sólo no sabe en qué lenguaje está escrito el cgi, y mediante esta línea se le indica que el código debe ser interpretado por PERL, que reside en el directorio `/usr/bin`.

A continuación se indica al intérprete que debe leer la biblioteca `cgi-lib.pl`, para poder usar las funciones contenidas en ella, y seguidamente se invoca una de ellas, la función `&ReadParse`, ya comentada, que almacenará los datos del formulario en la variable global `*input`. Posteriormente podremos acceder a estos datos mediante expresiones del tipo:

```
$nombre_variable = $input{'nombre'};
```

donde `nombre_variable` puede coincidir con `nombre`, aunque no es necesario, y `'nombre'` debe coincidir con el atributo `name` de algún elemento del formulario.

```
open (RESULTADO, ">resultado-iberico")||die "No puedo abrir RESULTADO";
```

Esta línea crea un archivo de texto llamado `resultado-iberico`, en el que podrá escribir (es lo que indica el signo `>`), y se podrá acceder a él mediante el “file handle” `RESULTADO`. En caso de no poder abrirlo, se cerrará el programa con un mensaje de error.

```
system("ls *.*.* > listaficheros")==0;
```

Aquí se ejecuta una orden del sistema operativo. Es como si tecléáramos directamente en la línea de comandos la orden entre comillas. Esta orden sería idéntica en MSDOS, sólo que cambiando `ls` por `dir`. Lo que hace es listar todos los archivos cuyo nombre tenga dos puntos (es decir, todos los que contienen textos ibéricos, pues en nomenclatura MLH su nombre lleva dos puntos), y el resultado lo guarda en un archivo de texto llamado `listaficheros`, que se crea automáticamente al ejecutar la orden. A continuación se abre el archivo obtenido mediante la línea

```
open (LISTA, "listaficheros")||die "No puedo abrir LISTA";
```

Como vemos, aquí el archivo se abre sólo para lectura, pues no lleva el signo `>`.

```
@lista_ficheros = <LISTA>;  
close (LISTA);
```

Al poner entre los signos `<>` el nombre del “file handle” `FILA` lo que se hace es leerlo línea por línea. Recordemos que su contenido es el resultado de la orden `ls *.*.*`, que en nuestro caso produciría algo así:

```
A.0.1  
A.0.2  
A.0.3  
A.0.4  
...
```

Es decir, un nombre de archivo en cada línea. Así, con las líneas anteriores lo que se hace es crear un array (variable tipo lista) llamada `@lista_ficheros`, cuyo contenido será la lista de archivos, de modo que

```
@lista_ficheros = "A.0.1,A.0.2,A.0.3,A.0.4,..."
```

Una vez obtenido el contenido de `LISTA`, es aconsejable cerrarlo cuanto antes con `close(LISTA)`, por varias razones: se ocupa innecesariamente memoria, se puede dañar el archivo en caso de un corte de energía, y en el caso de archivos de contenido variable, como el `resultado_iberico`, se pueden obtener resultados inesperados si por una rara casualidad dos usuarios lo tuvieran abierto simultáneamente. No olvidemos que este programa está hecho para ser accesible a cualquiera a través de Internet.

El siguiente paso es obtener la o las expresiones regulares introducidas por el usuario mediante las líneas

```
$expresion1 = $input{'expresion1'};
$expresion1 =~ s/([adegiklmoprtuRSMbns])(?!\\W)/$1\\n*/g;
$expresion2 = $input{'expresion2'};
$expresion2 =~ s/([adegiklmoprtuRSMbns])(?!\\W)/$1\\n*/g;
```

de las cuales la primera y tercera obtienen la cadena introducida en las ventanas de texto del formulario, cuyo atributo `name` era, respectivamente, `expresion1` y `expresion2`, y la introducen en las variables con el mismo nombre. Las líneas segunda y tercera tienen como finalidad hacer que la expresión buscada por el usuario sea reconocida aunque esté cortada entre dos líneas. Su comprensión puede resultar oscura hasta que no entremos en el tema de las expresiones regulares. Bastará con decir que lo que hacen es añadir después de cada letra introducida por el usuario (únicamente de las usadas para transcribir los textos ibéricos) la secuencia `\\n*`, que significa que en cualquier punto de la cadena buscada puede haber cero o más saltos de línea. El `\\n` es el carácter usado en expresiones regulares para indicar salto de línea, y el `*` es el cuantificador que indica un número indeterminado de veces, de cero a infinito.

La siguiente línea, `$numerofichero = @lista_ficheros;`, sirve para obtener de forma dinámica el número total de archivos de textos ibéricos, información que necesitamos para poner límite al bucle que se ejecutará después. Al asignar una variable tipo array como valor de una variable escalar, ésta contendrá un número que corresponde al número de elementos del array, es decir, el número de textos (más de 1500).

A continuación tenemos la siguiente línea:

```
if ($input{'num'}) {$num = $input{'num'}} else {$num = "1"}
```

Con ella se verifica si el usuario ha introducido algún número en la casilla del formulario “Descartar textos de menos de ... caracteres”. Este número se almacena en la variable `$num`. Si el usuario deja la casilla en blanco, se asigna a la variable el valor “1”, es decir, no se descarta ningún texto, pues todos tienen al menos un carácter.

Siguen las siguientes líneas:

```
$i = 0;  
$e = 0;
```

Aquí simplemente se asigna un valor inicial 0 a las variables `$i` y `$e`, que se utilizarán después en el contador.

Las líneas 19 a 27 son las que facilitan al usuario la utilización de clases de caracteres que resultan especialmente útiles para el caso del ibérico. Este aspecto se verá con más detalle en el capítulo dedicado a las expresiones regulares, pero podemos adelantar que, por ejemplo, `[sS]` encaja tanto con `s` como con `S`. Esta clase de caracteres debería ser de uso frecuente, pues hay a menudo dobles como **salir** / **salir**, que pueden ser debido a errores del escriba, y por ello se ha creído conveniente facilitar su uso mediante una abreviatura como `Z`, que el programa sustituye por `[sS]` mediante la línea siguiente:

```
$expresion1 =~ s/Z/[sS]/g; #silbantes
```

Lo mismo se efectúa para las demás abreviaturas previstas, que son `C` (consonantes), `V` (vocales), `T` (dentales), `K` (velares), `P` (oclusivas), `Q` (vibrantes), `L` (sonantes) y `Y` (sílabas). Nótese que para las vibrantes no se ha podido utilizar `R`, porque ya se usa para transcribir **ř**.

En el listado que vamos comentando se han omitido las líneas que efectúan las mismas sustituciones para `$expresion2`, para no recargar inútilmente el comentario.

A continuación se inicia la parte más importante del programa, contenida en un bucle `for`, que es una sentencia de control que hace que el código que contiene se ejecute a partir de una condición inicial y hasta que se cumpla una condición final, que en este caso son `$indice = 0` y `$indice < $numerofichero`. La variable índice se irá incrementando en una unidad a cada pasada del bucle (`$indice++`). El motivo de enfocar así esta parte del programa es disponer de una variable `$indice` que nos servirá para podernos referir al nombre de cada archivo de texto para abrirlo. Recordemos

que la lista de nombres de archivos está ya contenida en un array llamado `@listaficheros`. Podemos referirnos a cada uno de los elementos de un array utilizando un índice, de modo que si tenemos un array

```
@listaficheros = "A.0.1,A.0.2,A.0.3,..."
```

resultará que

```
@listaficheros[0] = "A.0.1";  
@listaficheros[1] = "A.0.2";  
@listaficheros[2] = "A.0.3";
```

Podemos sustituir el número índice por una variable, en este caso `$indice`. Como su valor inicial es 0, en la primera pasada del bucle tendremos que `@listaficheros[$indice]` contiene el valor "A.0.1", si éste es el primer archivo listado.¹ Como el primer elemento de un array es el que lleva el índice 0, en un array de `n` elementos el último llevará como índice `n-1`. Por ello la condición para la ejecución del bucle es que `$indice` sea menor que `$numerofichero`. Cuando sea igual, el bucle finalizará. Es decir, el bucle se ejecutará tantas veces como archivos de texto tengamos.

El código de la línea 29 nos resulta ya familiar. Abre el texto ibérico cuyo nombre está incluido en el array `@listaficheros` en la posición `$indice`, y lo almacena en el "file handle" `TEXTOS`. Hay que tener en cuenta que el contenido de `TEXTOS` se reemplaza a cada pasada del bucle por el texto del nuevo archivo abierto. La línea 30 hace lo mismo con los archivos que contienen información sobre el texto. Estos archivos tienen el mismo nombre que el texto correspondiente, y residen en el subdirectorio `/cgi-bin/inf`.

Las líneas 31, 32 y 33 contienen el siguiente código:

```
undef $/;  
$texto = <TEXTOS>;  
$textinf = <TEXTINF>;
```

Su explicación es la siguiente: la secuencia `<TEXTO>`, como hemos visto, lee el texto línea por línea. La variable especial `$/` contiene el carácter que se usará para separar líneas, que por defecto es `\n` (de *new line*). Con la orden `undef` se consigue que ni `\n` ni ningún otro carácter se considere separador de líneas a efectos del "file handle" `<TEXTO>`, con lo que se puede introducir el

¹Hay que tener presente que el orden con que aparecen listados los archivos no coincide con el orden alfabético, pues los números se ordenan como si fueran caracteres a partir de su valor en el código ASCII. Así podríamos tener una secuencia como A.0.1, A.0.11, A.0.12, A.0.2, A.0.21, A.0.3, etc.

texto entero en una única variable escalar, en este caso `$texto`. Igualmente se almacena el contenido de `<TEXTINF>` en `$textinf`.

A continuación viene el código necesario para obtener los contadores:

```
while ($texto =~ /$expresion1/g) {
    $i++;
}
if ($expresion2 =~ /.+/) {
    while ($texto =~ /$expresion2/g) {
        $e++;
    }
}
```

Se trata de dos bucles `while`, cuya condición es la existencia en el texto de la expresión regular introducida por el usuario. Los detalles de cómo se busca la expresión regular, como el uso de los modificadores `gs`, se verán en el capítulo dedicado a las expresiones regulares. De momento, basta con señalar que, por cada vez que se localiza la expresión en el texto, se incrementa la variable-contador en una unidad. A esta variable, que previamente habíamos inicializado a 0 (ver líneas 17 y 18), la llamamos `$i` para la primera expresión y `$e` para la segunda. El contador para la segunda expresión solo se pone en marcha si se cumple la condición de la sentencia condicional `if`, esto es, que no se haya dejado en blanco, lo que se expresa con la expresión regular `+`, que podríamos traducir como “uno o más caracteres”. Sin esta condición el resultado se falsearía, pues se consideraría que la cadena a buscar es un espacio en blanco, y en todos los textos aparece, por lo que sería computado. A partir de ahora bastará con imprimir (`print`) las variables `$i` y `$e` en algún punto de la página de respuesta para conocer el número total de apariciones de la cadena o las cadenas buscadas.

En las líneas 43 y 44 se puede decir que reside el núcleo del programa, son las que ejecutan por fin la búsqueda de la expresión regular en el texto y marcan las cadenas coincidentes para que aparezcan en rojo y negrita. Son las siguientes:

```
if ($texto =~ /$expresion1/g) {
    $texto =~ s/$expresion1/<FONT COLOR=RED><STRONG>$$&
</STRONG></FONT>/gs;
```

Se trata del principio de un bloque condicional, que se extiende hasta la línea 71, en el que, si se cumple la condición (ésto es, el texto encaja al menos una vez con la primera expresión regular introducida por el usuario), se ejecutarán una serie de operaciones, la primera y más importante de las cuales es la sustitución (s/) de toda cadena que encaje con esa expresión por ella misma (\$&) enmarcada por las directivas HTML que indican tipo de letra de color rojo y en negrita, respectivamente y , y sus finalizadores y . Éstos últimos llevan la barra inclinada (/) precedida por la secuencia de escape (\) para evitar que sea interpretada como el delimitador del operador de sustitución (s).

Hay que señalar que \$& es una variable especial en la que se almacena la última cadena que ha encajado con la expresión regular buscada. El modificador g indica que ello debe hacerse a lo largo de todo el texto, todas las veces que aparezca una cadena que encaje.

En las líneas 45-46 se hace lo mismo con la segunda expresión regular introducida por el usuario, pero hay que tener en cuenta que estas líneas están englobadas en el condicional de la línea 43, es decir, que sólo se busca la segunda expresión en los textos que contengan al menos una vez la primera. La condición de la línea 45 se cumple siempre que la segunda ventana de texto se deje en blanco. Hay una pequeña diferencia respecto a \$expresion1: al final de la línea 46 tenemos

```
if ($expresion2 =~ /.+/);
```

La finalidad de este condicional es asegurar que la sustitución anterior sólo se efectúe si el usuario introduce algo en la segunda ventana de texto ('expresion 2'). De lo contrario, en caso de no introducirse nada, el programa podría producir resultados indeseados.

El texto así modificado podría imprimirse ya en la página-resultado, pero antes debe hacer otra operación, que se realiza en las líneas 47-67:

```
if ($input{'np'}){
    open (NOMBRES, "<nombres")||die "no puedo abrir nombres";
    $/ = "\n";
    @nombres = <NOMBRES>;
    undef $/;
    $nombres = @nombres;
    for($x = 0; $x < $nombres; $x++){
```

```

    $nombre = @nombres[$x];
    $nombre =~ s/\n*//g;
    $nombre =~ s/[abdegiklmnoprstuMRS](?!\\)/$&\\s*\\n*\\/g;
    if ($texto =~ /$nombre/) {
        $texto =~ s/$nombre/<U>$&<\/U>/g;
    }
}
close (NOMBRES);
}

```

La función de esta porción de código es localizar los posibles nombres personales ibéricos que aparezcan en el texto, y marcarlos con subrayado. Ésto se hace a partir de un listado de nombres, extraído fundamentalmente del capítulo dedicado a la onomástica en MLH III, y completado con los nombres que se han identificado posteriormente en otras fuentes, especialmente en textos recientemente publicados. El listado se encuentra en un fichero, llamado simplemente **nombres**, cuyo formato es muy simple: un nombre por línea. La única peculiaridad es que, en vez de escribir el nombre tal cual, se han separado los dos elementos que, como se sabe, forman todo nombre personal ibérico, por una expresión regular, como puede verse en el siguiente ejemplo:

```
aRki(ke|i|M)?tiker
```

De esta forma se reconocerá el nombre aunque lleve uno de los tres infijos expresados (**ke**, **i**, **M**). En ocasiones, también se ha recurrido a las expresiones regulares para resumir en una dos o más variantes. Así, **s?osin(ke|i|M)tibaS?** equivaldría tanto a **osintibaś** como a **sosintiba**.

La identificación de nombres personales sólo se efectúa si el usuario marca la casilla de verificación correspondiente en el formulario, cuyo atributo **name** tiene el valor **np**. Por tanto, si el usuario marca esa casilla, se cumplirá la condición expresada por

```
if ($input{'np'})
```

y el código del bloque condicional (entre los dos corchetes) se ejecutará. Para ello, primero volvemos a definir **\n**, es decir, el carácter de salto de línea, como separador de líneas, para poder leer el archivo **nombres** línea por línea. La expresión **@nombres = <NOMBRES>** almacena en una variable tipo

array el contenido del archivo nombres, de manera que cada nombre sea un elemento de la lista. A continuación volvemos a quitar la definición de `\n` como separador de líneas (`undef $/`).

Las dos siguientes líneas (52 y 53) son similares a las que hemos utilizado para abrir los archivos de textos ibéricos uno a uno, pero en este caso son para almacenar en una variable `$nombre` un nombre ibérico cada vez, y ver si ese nombre, entendido como una expresión regular, se da en el texto ibérico que tenemos abierto. Recordemos que todo este bloque dedicado a la onomástica se repite entero para cada texto ibérico que contenga la expresión regular buscada inicialmente por el usuario, y por ello, si el número de textos que coinciden es elevado, el programa se ralentiza considerablemente. De ahí que esta parte sea opcional para el usuario.

Establecida en la línea 52 una variable `$nombres`, cuyo contenido es el número de elementos del array `@nombres`, la utilizamos en la siguiente línea para poner límite al bucle `for`, es decir, éste se ejecutará mientras la variable que usamos como índice (aquí `$x`), sea menor que el número de nombres ibéricos (teniendo en cuenta que su valor inicial es cero). La siguiente línea almacena en la variable `$nombre` (no confundir con `$nombres`) el nombre "`$x`-avo" del array `@nombres`. El problema es que, al leer línea por línea el archivo de nombres, la línea no sólo contiene el nombre sino el carácter `\n` (invisible) de final de línea. Por tanto hay que extraerlo mediante la siguiente expresión:

```
$nombre =~ s/\n*/g;
```

Pero hecho ésto, como queremos que los nombres sean reconocidos en los textos aunque vayan cortados por un salto de línea, se hace necesaria la siguiente expresión:

```
$nombre =~  
s/[abdegiklmnoprstuMRS] (?!\?)/$&\s*\n*\]/g;
```

Con ello se introduce `\s*\n*` (cero o más espacios en blanco o saltos de líneas) después de cada letra de las que usamos para transcribir los nombres ibéricos.

Por fin llegamos al meollo del bloque dedicado a la onomástica. Si el nombre aparece en el texto (es decir, si se cumple la condición `$texto =~ $nombre`), se enmarca el nombre por las marcas html de subrayado (`<U>` y `</U>`):

```
$texto =~  
s/$nombre/<U>$$</U>/g;
```

Con ello ya tenemos el texto ibérico actual con la expresión regular buscada señalada en rojo y con los nombres personales subrayados, y por tanto podemos imprimirlo en el “file handle” RESULTADO. Es lo que hacen las tres líneas siguientes: imprimir el nombre del texto (contenido en @lista_ficheros[\$indice]) enmarcado por los caracteres html de encabezado de tamaño medio (<h3>), imprimir el texto ibérico modificado, y por último imprimir dos saltos de línea para separarlo del siguiente texto:

```
print RESULTADO ("

### @lista_ficheros[$indice]</h3>"); print RESULTADO ("$$texto"); print RESULTADO ("\n\n");


```

Tenemos pues como resultado un “file handle” RESULTADO que contiene todos los textos ibéricos en los que se ha localizado alguna cadena de texto que encaje con la o las expresiones regulares buscadas, cadenas que aparecerán en rojo, y con los nombres propios ya subrayados. Para poder utilizarlo debemos introducirlo en una variable, lo que hacemos con la expresión

```
@resultado = <RESULTADO>;
```

Sólo queda enviar el resultado al navegador del usuario en un formato conveniente. Para ello, primero se invoca la función &PrintHeader, que, como su nombre indica, simplemente imprime el encabezamiento http necesario para que el navegador entienda que lo que va a recibir es una página web, y como tal debe representarla. El resto de código es lo que se conoce como un “here document”, es decir, el equivalente de poner `print` al principio de cada línea. Lo que imprimimos es el código fuente de la página web que verá el usuario, y por tanto se trata en su mayor parte de código HTML (reconocible por las etiquetas como <html>, <title>, <h1>, etc.), y no lenguaje PERL, salvo los elementos dinámicos, que cambian en cada ejecución del programa, y que básicamente son tres: el texto de la expresión regular

introducida por el usuario (para que pueda tenerla a la vista, y verificar si la ha introducido correctamente), contenida en `$input{'expresion1'}` y `$input{'expresion2'}`; el número de coincidencias localizadas, contenido en `$i` y `$e`, y la variable que contiene el conjunto de textos con sus nombres (`@resultado`). Esta última se enmarca entre las etiquetas HTML `<PRE>` y `</PRE>`, cuyo valor es “texto preformateado”, que usamos aquí para que los textos ibéricos aparezcan con sus saltos de línea originales. Hay que tener en cuenta que los navegadores ignoran los saltos de línea, adaptando el texto al tamaño de la ventana del navegador, y sólo cambian de línea cuando se indica explícitamente con las etiquetas de fin de línea (`
`) o de nuevo párrafo (`<P>`).

Toda esta porción queda como sigue:

```

print &PrintHeader;
print <<ENDOFTEXT;
<html>
<head>
<title>Resultado de la búsqueda: </title>
</head>
<body>
<h1>Resultados de la búsqueda</h1>
<h2>Expresiones: $input{'expresion1'},
    $input{'expresion2'}</h2>
<h2>Total: $i, $e </h2>
<pre>@resultado</pre>
<h2>Total expresión 1: $i</h2>
<h2>Total expresión 2: $e</h2>
ENDOFTEXT
print &HtmlBot;
```

El “here document” sería todo el comprendido entre `<<ENDOFTEXT` y `ENDOFTEXT`, tras lo cual se invoca la función `$HtmlBot`, que simplemente imprime los caracteres de final de página web (`</body>` y `</html>`).

Capítulo 6

Las expresiones regulares

6.1. Introducción

Las expresiones regulares constituyen un sublenguaje utilizado por muchos lenguajes de programación para efectuar búsquedas o sustituciones de cadenas de texto. El lenguaje PERL es el que implementa un motor de expresiones regulares más potente, y ha desarrollado algunas funcionalidades que no existen para otros lenguajes. De todas formas, la mayor parte de lo que veremos a continuación es común para las expresiones regulares escritas en cualquier lenguaje.

Podríamos decir que las expresiones regulares son patrones que describen las características que debe tener el texto a buscar o a sustituir. Para ello utilizan, además de los signos correspondientes al juego de caracteres que utilicemos (alfabéticos, numéricos, acentos, signos de puntuación, etc.), un reducido número de metacaracteres utilizados para describir conjuntos de signos, número o posición de los mismos, que esconde una considerable complejidad en su uso.

Este capítulo no pretende ser un manual de expresiones regulares¹, sino más bien una introducción centrada en lo que puede ser más útil de cara a su uso en el programa que nos ocupa. Por tanto, los ejemplos se basarán, en la medida de lo posible, en la búsqueda de cadenas de texto en textos ibéricos. En realidad, para este fin basta con unos conocimientos muy elementales de expresiones regulares, al menos mientras nuestro conocimiento de la morfología y sintaxis del ibérico no nos empujen a búsquedas mucho más definidas de lo que pueden ser en el momento actual. Evidentemente, es mucho más sencillo buscar posibles sufijos que determinar si una cadena de texto puede ser una dirección de correo electrónico válida.

¹Hay varios publicados, de los que merece destacarse Pinyan (2003); Friedl (2002)

6.2.

En su forma más simple, cualquier cadena de texto puede ser, en sí misma, una expresión regular. Si en la ventana del buscador tecleamos, por ejemplo,

`Salir`

obtendremos como resultado la relación de textos en que aparezca la palabra ibérica que habitualmente se transcribe **śalir**. Recordemos que se aquí se han sustituido los signos con diacrítico por las mayúsculas correspondientes, para mayor simplicidad.

En este caso se trata de una palabra con suficiente cuerpo fónico como para que nos interese tanto si aparece separada por interpunción o fin de línea como si forma parte de una cadena más larga, como en **iltir̄taśalirban**. Sería diferente si lo que buscamos es una secuencia muy breve, por ejemplo el sufijo **-ka**. Naturalmente, aquí hay que garantizar de alguna forma que sólo aparezcan textos que contengan la secuencia **ka** en posición final de palabra. Ahora bien, el único criterio objetivo de que disponemos para ello es la interpunción², de forma que podríamos iniciar nuestra búsqueda con **ka.**, pero ocurre que el punto es un metacarácter, como veremos a continuación, y para que el buscador lo interprete como signo de interpunción hay utilizar la “secuencia de escape” `\.`. Es decir, la barra inclinada hacia atrás sirve para que un metacarácter no sea interpretado como tal³. Por tanto, para buscar la secuencia **ka** ante interpunción habrá que teclear

`ka\.`

Pero con ello no obtendríamos todos los posibles sufijos **ka**, pues son numerosas las inscripciones que constan de una sólo palabra⁴. Tampoco es frecuente marcar la interpunción en final de línea, lo cual no quiere decir

²Velaza (1991b, p. 9)

³Otros metacaracteres que veremos se representan por signos alfabéticos precedidos por la barra invertida. En este caso, pues, lo que hace la barra es marcar que esos signos alfabéticos funcionan como metacaracteres, justo lo contrario de lo que acabamos de señalar cuando el metacarácter es un signo no alfabético

⁴Hacemos aquí abstracción de los textos que no presentan interpunción, que evidentemente no están formados por una sólo palabra, pero no nos dejan otra posibilidad para la búsqueda automática de sufijos que el criterio del fin de línea, siempre dudoso.

que no puedan aparecer palabras cortadas por el fin de línea. Por tanto, necesitamos una forma de expresar el final de palabra más comprensiva que la simple interpunción, y las expresiones regulares nos ofrecen una que se ajusta bastante bien a nuestro propósito: el metacarácter `\b`, que representa precisamente “límite de palabra”, entendido como el límite entre un carácter tipo *word* (cuyo metacarácter es `\w`, y sería cualquier carácter alfabético) y un carácter tipo *non-word* (`\W`, cualquier otro carácter). Por lo tanto, si tecleamos

```
ka\b
```

obtendremos todas las apariciones de **ka** ante interpunción o fin de línea, aunque se trate de una única línea, o de espacio en blanco, pero también ante cualquier carácter no alfabético, como puede ser el corchete utilizado para indicar zonas de texto desaparecidas o borrosas. Sirva esto como advertencia de que en cualquier expresión regular podemos obtener resultados no deseados, por exceso o por defecto. Como criterio general, en nuestro caso podría decirse que es mejor utilizar expresiones que pequen por exceso, pues es fácil descartar después los resultados que no nos interesen, y no siempre merece la pena el esfuerzo de construir una expresión regular tan precisa que no abarque algo más de lo que buscamos. En el caso de los finales de línea, además, es imposible definir criterios para saber si cortan o no una palabra, debiendo quedar la cuestión a criterio del investigador.

En este último ejemplo nos queda aún algo que pulir. Se trata de aquellas inscripciones en que **ka** es el único texto, o bien aquellas en que puede aparecer entre dos signos de interpunción, de forma que no puede ser un sufijo. Para ello contamos con el metacarácter `\w`, para indicar que debe aparecer algún carácter alfabético antes del sufijo. Así, la expresión quedaría de la siguiente forma:

```
\wka\b
```

Queda una última observación que hacer respecto al metacarácter `\b`, y es que es del tipo de las que se conocen en inglés como *zero-width assertions*. Para entenderlo hay que pensar que un archivo de texto es una cadena de caracteres, alfanuméricos o no, e incluso los espacios en blanco o saltos de línea son caracteres (`\s` y `\n`), aunque no se visualicen. El metacarácter `\b`

no encaja con ningún carácter, sino con el límite entre dos caracteres, uno alfabético y otro no. En nuestro programa todo lo que encaja con una expresión regular aparece en rojo, y por tanto si tecleamos `\wka\b` se marcará en rojo no sólo **ka**, sino también la letra inmediatamente anterior⁵, pero no marcará cualquier signo que vaya a continuación, aunque sea no alfabético.

6.3. Cuantificadores

Podemos aprovechar aún más el ejemplo anterior para introducir el uso de los cuantificadores. Si sólo nos interesa el sufijo **ka** cuando va unido a palabras de cierta longitud (como pueden ser los nombres personales), podríamos utilizar un cuantificador después de `\w` para indicar cuántos caracteres alfabéticos pueden aparecer en esa posición. Se dispone de los siguientes cuantificadores:

*	Cero o más veces
?	Una o ninguna vez
+	Una o más veces
{n,m}	Entre <i>n</i> y <i>m</i> veces
{,n}	Como máximo <i>n</i> veces
{n,}	Como mínimo <i>n</i> veces

Debe entenderse que los anteriores cuantificadores son *voraces*, es decir, que encajan con el número mayor posible. Así, el interrogante debe entenderse como “cero o una vez, pero si puede ser, una”, y el asterisco como “cero o el mayor número de veces posible”. Se puede invertir esta característica añadiendo a continuación un signo de interrogación (`*?`, `??`, `+`), de modo que se encaje con el menor número de caracteres posible.

Es preciso advertir que los cuantificadores, por sí solos, no nos sirven para buscar palabras de una determinada longitud (entendiendo palabra como toda secuencia de caracteres alfabéticos situados entre signos de interpunción, principio o final de línea o espacios en blanco). Es decir, una expresión como `\w{3}` encajará con palabras de tres letras, pero también con las tres primeras letras de palabras que tengan cuatro o más. Hay que indicar que ni antes de la primera letra ni después de la última puede haber otra, por ejemplo mediante `\b: \b\w{3}\b`.

⁵Para que no lo haga, habría que incluir `w` dentro de una *zero-width assertion*, procedimiento exclusivo de PERL que veremos más adelante.

6.4. Clases de caracteres

Las expresiones regulares disponen de una característica que puede resultar muy útil aplicada a las búsquedas en textos ibéricos. Se trata de las clases de caracteres, que básicamente son una forma de señalar un conjunto de caracteres que pueden ocupar una determinada posición. Una clase de caracteres se expresa mediante los corchetes (`[]`).

Dentro de una clase de caracteres pueden utilizarse rangos (expresados mediante el guión, `-`). Así, podemos expresar que en determinada posición puede encajar cualquier letra minúscula mediante la clase de caracteres `[a-z]`. O cualquier letra, sea mayúscula o minúscula, mediante `[a-zA-Z]`, aunque para ello ya disponemos de `\w`. Este último metacarácter también encajaría con números, pero en nuestro caso no es un inconveniente en principio.

Así pues, hay que entender una clase de caracteres como si de un único carácter se tratara. Es decir, `[a-zA-Z]` encaja con la primera letra que encuentre, y sólo con la primera, del mismo modo que lo haría `\w`. Para que encaje, por ejemplo, con la primera palabra que encuentre, habría que utilizar un cuantificador: `[a-zA-Z]+`, o mejor `\w+`.

El mayor servicio que nos pueden prestar las clases de caracteres es la facilidad de unificar búsquedas en textos cuyos originales pueden estar en alfabeto ibérico o grecoibérico, es decir, podemos prever la posibilidad de que una palabra que contiene oclusivas sordas pueda aparecer en un texto transcrito con sonoras. Si queremos ver todos los textos en que aparece la palabra **kutur**, la expresión regular más recomendable sería `[kg]u[td]ur?`. Con esta expresión encajarían, en caso de que existieran, *kutur*, *gutur*, *kudur*, *gudur*. El último interrogante lo hemos puesto para obtener los textos en que la palabra aparece sin la *-r*, como **kutuboike** y *gudu*. Con ello vemos, de paso, que los cuantificadores (el interrogante, en este caso), afectan únicamente al carácter, metacarácter o clase de caracteres que les preceden. Si queremos que afecten a varios, deberíamos utilizar paréntesis. Así, con `[kg]u[td]u(boi)?(ke)?` obtendríamos los mismos resultados que antes, pero la palabra **kutuboike** aparecería toda marcada en rojo, y no sólo su primer componente.

Otro uso que puede resultar interesante es prever ciertos intercambios frecuentes de sonido, que no siempre podemos valorar si se dan por error del escriba, variante dialectal o diacrónica, o simplemente porque se trata de palabras distintas. Me refiero especialmente a las silbantes y vibrantes. Por ejemplo, en G.1.2 tenemos *salir* por el frecuente **śalir**, de modo que, para que encajara también, deberíamos usar `[sS]alir`.

Las clases de caracteres se pueden negar, de modo que contengan la lista

de caracteres que *no pueden* ir en determinada posición. Ello se hace poniendo el acento circunflejo $\hat{}$ al principio de la clase de caracteres. Así, $[\hat{aeiou}]$ indica que en esa posición no puede ir una vocal.

Dada la utilidad y el uso frecuente que cabe hacer de las clases de caracteres, se ha previsto en el programa buscador la posibilidad de crearlas automáticamente. Se trata de que, si el usuario teclea una T en lugar de t, el programa transforma la T en $[td]$, mediante la línea de código siguiente: `$expresion1 =~ s/T/[td]/g;`. Hay disponibles las siguientes abreviaturas:

- C = $[pbt dkgnmMlrRsS]$ (consonantes).
- V = $[aeiou]$ (vocales).
- T = $[td]$ (oclusivas dentales).
- K = $[kg]$ (oclusivas velares).
- P = $[pbt dk]$ (oclusivas).
- Z = $[sS]$ (silbantes).
- Q = $[rR]$ (vibrantes).
- L = $[nmMrRl]$ (sonantes).
- Y = $([btdkgnmMl sSrR]?[aeiou][iu]?[rRnMml]?[sS]?)$ (sílabas).

De este modo podemos construir con facilidad expresiones regulares que de otro modo serían complejas, engorrosas de teclear, y mucho más expuestas a errores. El programa devuelve, en la parte superior de la página con los resultados, la expresión regular introducida por el usuario, seguida por la que resulta de desplegar las abreviaturas, que es la que realmente se aplica, a efectos de comprobación.

Bastará, pues, con introducir la expresión Y, para obtener todos los textos ibéricos que contengan al menos una sílaba, que aparecerán marcados completamente en rojo, lo que demuestra que el esquema silábico propuesto (tomado de de Hoz 2001a) da cuenta de cualquier sílaba ibérica, salvo raras excepciones debidas a interpunciones irregulares o textos fragmentarios.

Otra posible aplicación es la búsqueda de candidatos a nombre personal, que como se sabe muestran, en su forma más habitual, de dos elementos bisilábicos. Por tanto, la expresión `\bY{4}\b` nos daría un buen número de palabras en que buscar nombres personales. Naturalmente, se trata de una expresión bastante poco concreta, que puede encajar con palabras que nada tienen que ver con nombres personales, o ignorar muchos que sí lo son, como

los que constan de algún elemento monosilábico. Podría obtenerse algo más parecido a la realidad si especificamos, tras la expresión anterior, alguno de los sufijos que se sabe van unidos a nombres personales, como *-ka*, y ponemos un cuantificador un poco más laxo: `\bY{3,4}ka\b`. Naturalmente, todo esto es tan sólo un ejemplo de las posibilidades de las expresiones regulares.

6.5. Metacaracteres

Hemos visto ya que `\w` es un metacarácter, probablemente el que más útil nos puede resultar. Encaja con cualquier carácter alfanumérico, es decir, es sinónimo de `[a-zA-Z0-9]`. Su complementario es `\W`, que encaja con cualquier carácter no alfanumérico, que también puede resultar de utilidad, pues encajaría con signos de interpunción, espacios en blanco, corchetes o puntos suspensivos que indican fragmentos de texto perdidos, etc. Así, la expresión `kutur\W*biteRok` encajaría tanto con **kuturbiteRokan** (B.7.38) como con **kutur.biteRoketetine** (F.9.5).

Existe otro metacarácter más genérico, que es el punto (`.`). Encaja cualquier carácter, sea alfanumérico o no. También encaja con los saltos de línea si se utiliza el modificador `s`.⁶ El uso del metacarácter `.` es conveniente en expresiones como la del ejemplo anterior, en que buscamos dos palabras separadas por no importa cuántos caracteres, que pueden incluir interpunciones. Por supuesto, se pueden usar con el punto los cuantificadores para precisar, por ejemplo, una distancia máxima entre las dos secuencias buscadas. El metacarácter `.` sería sinónimo de `[\w\W]`.

El ejemplo anterior podríamos, pues, sustituirlo por `kutur.*?biteRok`. Obsérvese el interrogante tras el asterisco, para evitar el comportamiento voraz de éste. De lo contrario, en caso de que en el texto hubiera dos *bite*-tras *kutur*, la expresión encajaría desde *kutur* hasta el segundo *bite*-.

El metacarácter `\n` o salto de línea se ha utilizado en el código del programa, como hemos visto. Concretamente hay una línea que añade `\n*` a continuación de cada carácter alfabético introducido por el usuario (y que no sea a su vez un metacarácter), para prever la posibilidad de que la expresión buscada esté cortada entre dos líneas. Además, puede resultar de utilidad para el usuario: `\niunstir` encajaría con la palabra **iunstir** a principio de línea, `iunstir\n` al final.

Otros metacaracteres menos útiles para nosotros son `\s`, que indica espacio en blanco, y `\d`, que indica carácter numérico (y sus complementarios, en mayúsculas como siempre: `\S`, `\D`).

⁶Este modificador, en nuestro caso, no lo introduce el usuario, sino que se ha dejado “fijo”, por así decirlo, en una línea del programa.

6.6. Paréntesis

Hemos visto ya el uso de los paréntesis para englobar una sección de la expresión regular que puede verse afectada por un cuantificador. De paso, podemos señalar que si el cuantificador es $\{2\}$, por ejemplo, nos serviría para buscar reduplicaciones. En concreto $(te)\{2\}$, equivadría a *tete*, lo cual puede parecer poco útil. Pero si en vez de caracteres alfabéticos usamos meta-caracteres o clases de caracteres, la utilidad aumenta, pues podemos buscar sílabas repetidas con un determinado patrón, como $([btdkg][aeiou])\{2\}$, que podemos expresar más simplemente por $(CV)\{2\}$, que localizaría secuencias bisilábicas formadas por oclusiva más vocal.

Ahora bien, si lo que queremos es que ambas sílabas capturadas sean idénticas, debemos usar una característica estrechamente ligada al uso de paréntesis: se trata de las referencias.⁷ Éstas son, en realidad, variables cuyo ámbito se restringe al interior de una expresión regular. Tienen la forma $\backslash 1$, $\backslash 2$, etc., de manera que en $\backslash 1$ se almacena el contenido del primer par de paréntesis, en $\backslash 2$ el del segundo, y así sucesivamente.⁸ Si queremos, pues, buscar sílabas idénticas repetidas, podemos usar expresiones del tipo $(CV)(CV)\backslash 1\backslash 2$, que nos daría, por ejemplo, la leyenda monetaria **kuRuku-Ruatin** (A.30).

El siguiente ejemplo muestra como usar las referencias para buscar palabras que parecen concordar, al llevar un mismo sufijo: $\backslash b\backslash w+(\backslash w\{2})\backslash .\backslash w+\backslash 1\backslash b$. Como puede verse, hemos especificado que el sufijo debe estar formado por dos letras cualesquiera (podríamos haber usado un patrón más preciso, como (CV)), y que las dos palabras deben estar separadas por interpunción. Entre los resultados obtendríamos el plomo de Castellón, F.6.1, que muestra las secuencias **ekaRiu.atuniu** y **boRbeRoniu.koSoiu**.

Otro aspecto importante del uso de paréntesis es el de las alternancias. Ya hemos visto que, mediante las clases de caracteres, podemos indicar qué caracteres pueden ocupar una determinada posición. Una alternancia, en cambio, permite especificar conjuntos de caracteres. Tienen la forma $(\dots|\dots|\dots)$, pudiéndose colocar tantas barras verticales como grupos de caracteres puedan alternar. Si $[ab]$ y $(a|b)$ son casi equivalentes, en cambio $[abcd]$ y $(ab|cd)$ no lo son, pues la primera expresión sólo necesita un carácter para encajar, mientras que la segunda necesita dos, y precisamente en el orden indicado: *ab* o bien *cd*.

Las alternancias pueden resultar de enorme utilidad para las búsquedas en textos ibéricos. Así, hemos visto una expresión que podía localizar can-

⁷En inglés *backreferences*.

⁸Si hay paréntesis anidados, la prioridad va de más interior a más exterior, siempre de izquierda a derecha.

didatos a nombre propio, que ganaba en precisión si se le añadía un sufijo: `\bY{3,4}ka\b`. Naturalmente, con ello hay más probabilidades de que obtengamos nombres propios, pero sólo los que lleven ese sufijo, y no otros. Podemos mejorar la expresión usando una alternancia, que permita contemplar el uso de otros sufijos: `\bY{3,4}(ka|te|en)\b` (podemos incluir tantos sufijos como queramos). Con un interrogante al final, fuera del paréntesis, mantendríamos la posibilidad de que la secuencia anterior no lleve sufijo.

6.7. Aserciones de longitud cero

Con esta denominación, un tanto esotérica,⁹ se designan aquellos meta-caracteres que no encajan con un carácter cualquiera, sino con una posición, y de ahí que no consuman espacio. En nuestro caso, no marcarían nada en rojo. Ya hemos visto uno de ellos, `\b` (límite de palabra) y su complementario `\B`. Del primero ya hemos visto abundantes ejemplos.

Existen dos aserciones, `^` y `$`, que marcan respectivamente el principio y el final de línea. En nuestro caso es preferible usar el metacarácter `\n`, como hemos visto.

Por último, hay un tipo de aserciones que, aún encajando con uno o varios caracteres, no devuelven ningún resultado, y por tanto son también de longitud cero, es decir, en nuestro programa no marcarían nada en rojo. Tienen la forma `(?= REGEXP)` y `(?! REGEXP)`, donde `REGEXP` es cualquier expresión regular. La primera es positiva y la segunda negativa, y ambas buscan a *continuación* de lo que hay fuera del paréntesis la expresión contenida dentro, y si la encuentran, permiten o no que la expresión de fuera del paréntesis encaje, según si la aserción es positiva o negativa. Se han utilizado para elaborar el programa, por ejemplo en la línea 12:

```
$expresion1 =~ s/([adegiklmoprtsuRSMbns])(?!\W)/$1\n*/g;
```

Aquí se añade `\n*` a continuación de cualquier carácter alfabético que no vaya seguido por algún carácter no alfabético `(?!\W)`. De no ser por ello, convertiría una expresión como `s*` en `s\n**`, lo que causaría un mensaje de error.

Por lo demás, su utilidad para el usuario en nuestro caso es limitada, y en la versión actual del programa no funcionan correctamente, por lo que es mejor no usarlas. Por ejemplo, para buscar **ban** sólo si va seguido de **kutur**,

⁹Traducimos así la expresión inglesa *zero-width assertions*.

podría usarse `ban(=?katur)`, que marcaría en rojo únicamente **ban**, pero es más cómodo teclear simplemente `bankatur` y obtendremos lo mismo, sólo que con toda la cadena en rojo, lo que no supone mayor problema.

6.8. Modificadores

Para entender correctamente su funcionamiento, es preciso ver en detalle cómo se integran las expresiones regulares en el programa, ya que el usuario no introduce los modificadores, pero conviene que sea consciente de su existencia para poder entender e interpretar mejor los resultados.

Las expresiones regulares, en el código de un programa, van siempre integradas en una expresión más amplia, que busca en una cadena determinada (generalmente contenida en una variable) una subcadena que encaje con la expresión regular, y en su caso la sustituya por otra. En nuestro programa, la línea fundamental que efectúa la búsqueda y resaltado en rojo de las expresiones introducidas por el usuario es la siguiente:

```
$texto =~ s/$expresion1/<STRONG><FONT color="red">$&</FONT></STRONG>/gs;
```

Aquí, `$expresion1` contiene la expresión regular buscada por el usuario,¹⁰ de modo que podríamos representar gráficamente lo anterior así:

```
$texto =~ s/ (expresión) /<STRONG><FONT color="red">$&</FONT></STRONG>/gs
```

La variable `texto` contiene un texto ibérico completo (uno cada vez, a cada pasada del bucle `for`), y en él se sustituye (`s`) lo que encaje con la expresión regular contenida en `$expresion1` por el texto que va a continuación (en este caso, código HTML y una variable especial `$&` que almacena lo encajado por la última expresión regular buscada). Al final vemos dos letras, `gs`, que son los modificadores. El primero indica que la sustitución debe hacerse de modo global, es decir, tantas veces como encaje la expresión. El segundo sirve para que el metacarácter `.` (es decir, “cualquier carácter, sea o no alfanumérico”) encaje también con los saltos de línea.

Es decir, una expresión de sustitución tiene la forma siguiente:

```
s/expresión regular/texto/modificadores
```

Tanto la expresión regular como el texto de sustitución pueden contener variables, que se expanden en el momento de hacer la sustitución. Puede

¹⁰En realidad, aquí ha sufrido ya una primera transformación, añadiéndole saltos de línea tras cada letra de las usadas en la transcripción de textos ibéricos, para prever la posibilidad de que la palabra aparezca cortada entre dos líneas.

no haber modificadores, y puede también haber algunos otros que no hemos visto, y que no se han utilizado en este programa. Básicamente son dos: `i`, que sirve para evitar que se distinga entre mayúsculas y minúsculas, y que obviamente no se ha usado, pues en la transcripción elegida es básica la distinción; y `x`, que permite usar expresiones regulares extendidas, en las que se pueden usar espacios en blanco y saltos de línea (tecleados directamente, se entiende) para hacer más legible en el código del programa una expresión regular compleja. Como en nuestro caso las expresiones incluidas en el código no son muy complejas, y las que introduce el usuario no han de ser leídas por otro usuario, se ha optado por no usar este modificador, de manera que, si al teclear una expresión, el usuario pone un espacio en blanco, éste será interpretado como tal, es decir, equivalente al metacarácter `\s`, y será buscado en el texto.

Por último, es preciso insistir en el funcionamiento del modificador `g`. De no haberlo usado, la expresión regular se localizaría como máximo una vez en cada texto. Si la expresión, por ejemplo, fuera `\w`, en un texto como **iltírta** sólo la primera *i* aparecería marcada. Pero al usarlo, se marcarán todas las letras, de modo que una búsqueda que sólo contenga `\w` nos dará como resultado todos los textos ibéricos con todas las letras marcadas en rojo (pero no los signos de interpunción, corchetes, etc.).

Capítulo 7

Una aplicación del programa: el paradigma biteñoke

En este capítulo se ejemplificará el uso del programa mediante el estudio de un posible paradigma verbal, el representado por formas como **biteñoke** o **biteñoKAN**. Se ha elegido este ejemplo por ser quizás el que muestra mayor complejidad, con acumulación de sufijos, posibles prefijos e infijos. Por ello resulta más apto para mostrar en toda su extensión el uso de las expresiones regulares.

En primer lugar se expondrá un resumen de los diferentes estudios que se han llevado a cabo sobre el paradigma en cuestión, desde que fuera señalado por primera vez por Untermann. A continuación se discutirán las segmentaciones propuestas de las formas del paradigma, viendo si el uso del programa puede apoyarlas o modificarlas. De especial importancia resultará determinar cuál es el núcleo del paradigma. Seguidamente se verá la forma de construir expresiones regulares que integren el mayor número posible de formas conocidas del paradigma. Partiendo de ellas, se intentará a continuación construir expresiones que localicen posibles paradigmas relacionados. Es decir, si se trata de un paradigma verbal, deberían existir otros núcleos verbales que muestren idénticos sufijos o prefijos.

7.1. Historia de la cuestión

El paradigma de **biteñoke** ha sido estudiado principalmente por Untermann y de Hoz. Untermann lo planteó por primera vez en su trabajo *Gramática de los plomos ibéricos*, en el que estableció una lista de 12 formas. En sucesivos trabajos¹ ha aumentado la relación de componentes del para-

¹Untermann 1996, Untermann 1990, §566, Untermann 2001

digma, señalando algunos contextos recurrentes y reafirmando en su valor verbal. De Hoz trata también el tema en un trabajo reciente². Es preciso subrayar que el estudio de este paradigma no constituye el objetivo central de ninguno de los trabajos citados, por lo que ninguno de los dos autores realiza un estudio profundo para determinar la estructura del paradigma. De Hoz³ es explícito al respecto cuando habla de “el elemento léxico nuclear, que ahora se nos revela, sólo provisionalmente, a la espera de análisis más finos, como *iteío*”. El objetivo del autor se limita a mostrar la complejidad de las formas verbales ibéricas, propia de una lengua incorporante.

En MLH III, §566, Untermann incluye el término que nos ocupa en su relación de elementos léxicos, bajo el epígrafe **biteír-, bitíír-**, que sería la base del paradigma. Esta base podría aumentarse con sufijos y prefijos (como *bas-*, *-tin*), y en particular con el morfo (*ír*)*ok*. Éste podría unirse a otras bases, dando lugar a formas pertenecientes por tanto a otros paradigmas relacionados: *bagaírok*, *tagisgaírok*, *baseíroke*, *gaírokan*, *baseírokar* (todas ellas en G.1.1), **eíroke** (C.17.1), **iteíroke** (D.13.1) y **otiroketa** (G.15.1). La independencia de (*ír*)*oke* se justifica en la pareja **bas-biteíroke-tine** y *bas-bidírbar-tin*, que mostraría un “paralelismo paradigmático” entre *oke* u *bar*.

Respecto a las formas que sí formarían parte del paradigma, Untermann admite las formas con la variante **bitíír-**, y tampoco muestra reparos en aceptar las formas con la otra vibrante, como **seke-biterosan**, **bitirebaboşin** y **teşibiterukan** (todas ellas en F.17.2). La última, además, mostraría *uk* por *ok*.

En Untermann 1996 aumenta a 20 la lista de testimonios. Se sugiere la posibilidad de que el cambio entre *bi* y *i* al inicio se deba a una variante dialectal, y establece tres formas básicas del paradigma:

- **biteír**, con las variantes **bitíír**, **bitur**, **itíír**, siempre tras *bas* o *ban*.
- **biteíroke** (y **bitííroke**, **iteíroke**, **itííroke**, con sufijos como *te*, *ti*, *tan*).
- **biteírokan**, **bitíírokan**, **biterukan**.

Mantiene *biteír* como base del paradigma verbal, y como novedad incluye algunos segmentos largos que incluyen *iteír*, *itíír* y acaban en *-bon* (dos veces en C.1.24).

Por último, en Untermann 2001, se toca el tema un tanto de pasada. La lista aumenta a 31 formas, aunque no se incluye en la publicación. Establece una lista de nueve posibles contextos (el noveno sería “contextos menos claros”). La principal novedad es la forma **biteíroketan**. El resto es el resultado

²de Hoz 2001a

³de Hoz 2001a, pp. 345-346

de desglosar las formas con **biteñoke** en formas con o sin afijo, combinaciones con prefijos *bas*, con prefijo *ban*, o precediendo la palabra **kutur**.

De Hoz,⁴ en cambio, empieza analizando los plomos de Orleyl, de los que deduce un “elemento esencial” *biteño*. Al intervenir en la comparación D.13.1, deduce que *b-* sería un prefijo, quedando el núcleo reducido a *iteño*. De la comparación con otros testimonios acaba deduciendo que también *o* sería un sufijo, con lo cual queda finalmente un elemento *iteñ*, con una “forma alternante” *itir*. Presenta a continuación una lista de 17 formas, en las que no aparecen las formas con distinta vibrante, cuya relación con las anteriores considera problemática el autor (n. 48). Sí se incluyen las dos del plomo de Ampurias (C.1.24) acabadas en *-bon*. En Orleyl 5.3 se incluye por error **kutu-boike.basbiteñoketine**, duplicado con Orleyl 7, donde sí aparece. También se incluyen tres de los cuatro testimonios del plomo Marsal (H.0.1), que no aparecen en Untermann 96 por no estar publicado posteriormente. No aparece uno de los cuatro testimonios de Orleyl 1.

También Luis Silgo hace una breve referencia en su trabajo sobre el acento ibérico,⁵ en el que se limita a mencionar los dos testimonios de Orleyl 5 (F.9.5), proponiendo a partir de estos dos únicos ejemplos una segmentación:

(bas + bit- + e-ño-ke) + ((te)-tin-e)

Añade a continuación el autor que el núcleo sería **-ño-**, lo que no concuerda demasiado con la segmentación que él mismo propone.

Podemos resumir en una tabla el inventario de formas incluídas por cada autor, teniendo en cuenta que, en el caso de Untermann, se parte de Untermann 1996, por faltar la relación más completa de Untermann 2001:

B.3.2	kutuñar[¿iteñ?]e[¿bon?	Untermann 96
B.7.38	ñalirbitiñoakanabe[Untermann 96, de Hoz 2001
B.7.38	-bankuturbiteñoakan	Untermann 96, de Hoz 2001
C.1.24	salaker.itiñoketebon. .banitirñkata / .banitirñan. batirakañiteñibon	Untermann 96, de Hoz 2001 Untermann 96 Untermann 96 Untermann 96, de Hoz 2001
C.17.1	tortonbalañbiteñoakan[Untermann 96, de Hoz 2001
C.21.7	.banbite[Untermann 96, de Hoz 201
D.13.1	n.iskutur.iteñoke.ñm/	Untermann 96, de Hoz 2001
F.9.1	-ñalirbiteñ[¿oketa?]nanabekate/ biteñ/[¿oke?]tanbaleuñ/	Untermann 96, de Hoz 2001 Untermann 96, de Hoz 2001

⁴de Hoz 2001a, pp. 345-346

⁵Silgo Gauche 1994-1995, p. 179

	[ʝbite?]róketan -bi[ʝte?]róketan	Untermann 96 Untermann 96, de Hoz 2001
F.9.5	.kukur.biteóketetine. .bitiíokebetense.	Untermann 96, de Hoz 2001 Untermann 96, de Hoz 2001
F.9.7	kutuboike.basbiteóketetine.	Untermann 96, de Hoz 2001
F.17.2	.tesibiterukan[Untermann 96
G.1.1	.basbidiíbartin.	Untermann 96, de Hoz 2001
G.7.5	.basbiturbartin	Untermann 96
H.0.1	basbiteó uketin.biteókan kaulte.biteókan	de Hoz 2001 de Hoz 2001 de Hoz 2001

7.2. Segmentación del paradigma

7.2.1. El núcleo

El primer objetivo de una segmentación en un paradigma tan complejo debería ser determinar cuál es el núcleo, pues de ello depende la inclusión o no de las diferentes formas en el paradigma. Hemos visto en la sección anterior las principales propuestas en este sentido: *biteí*, *bitií* (Untermann), *iteí*, *itií* (de Hoz) y *ók*. Sobre esta última hay que señalar la dificultad de que empiece por vibrante. Como es sabido, en ibérico no se tolera al parecer una vibrante inicial, lo mismo que ocurre en vasco y ocurre o ha ocurrido en romances vecinos, como el gascón (hay también algunos restos de este fenómeno en aragonés medieval y en algunos topónimos altoaragoneses). Y al ser el ibérico una lengua aglutinante, según parece, hay una mayor autonomía de la raíz respecto a los afijos que recibe, y en nuestro caso sería posible que la raíz apareciera en algún caso sin prefijos. Por ello parece más verosímil *eók*.

Entre *bite/ír* y *ite/ír* parece que habría que inclinarse sin duda por la última, pues la forma sin *b-* en D.13.1 es incuestionable, y hay otros indicios de la existencia de un prefijo *b-*, como veremos más adelante. Otra cosa es aceptar la posibilidad de una raíz alternante, ya que, faltos del sentido, estamos forzados a aceptar las diferencias de escritura, por pequeñas que sean, como significativas, y del mismo modo que de Hoz descarta las formas con *biter/bitir* por *biteí/bití*, también habría que descartar por precaución las formas como *bitií*, *itií*. Sin embargo hay tres formas con *-i-* que por su longitud y contexto parecen más claras: **śalirbitiíokan** (B.7.38), **bitiíokebetense** (F.9.5) y con menos seguridad **itiíoketebon** (C.1.24).

La otra alternativa sería, a mi juicio, *eók*, similar a la propuesta por Silgo,

aunque con la salvedad mencionada de la vocal. Se basa fundamentalmente en las siguientes formas:]**tinbašteeróke** (C.17.1, en el que aparece también un **biteórica***) y **bastubarerteókan** (D.0.1, donde va seguido por **utur**, que parece que hay que relacionar con el **kutur** que a menudo acompaña a **biteórk-**). Ambas formas van precedidas probablemente por un antropónimo, y ninguna lleva *b-it-*. Ambas muestran la misma alternancia en el final (*ke/kan*) que **biteóke/biteókan**, que nos dejan como segmento más largo común a todas *eórk*. Otro indicio es que todas las formas con *biteí* incluyen *eórk*,⁶ pero no todas las que llevan *eórk* incluyen *biteí*. *iteí*, en cambio, sí que aparece a menudo sin *eórk*, pero en contextos muy diferentes. Véase especialmente el grafito sobre cerámica **selkiteár** (B.1.24), que parece una indicación de propietario. El **batirákaíteíbon** de Ampurias (C.1.24) es poco claro.

La ventaja de aceptar *eórk* como núcleo es que obtenemos un inventario de formas más reducido pero mucho más homogéneo en su contexto. Además, como veremos en el siguiente apartado, nos permite localizar otros núcleos que llevan los mismos afijos. Por ejemplo, podemos adelantar que, si segmentamos **bit-eórk-an**, podemos encontrar otros segmentos similares, usando la expresión regular `bit\w*?an\b`: **bitekian**, **biterosan**, **bitiboan**, **bitetean**, etc. Si efectivamente esta segmentación acaba siendo más productiva, en el sentido de permitir plantear nuevas hipótesis sobre posibles raíces o afijos verbales, ello sería un indicio importante en favor de su validez. Por otra parte, un núcleo *eórk* sería válido fonéticamente como palabra independiente, pues empieza por vocal, y el final en oclusiva se justifica por la forma *bagarórk* (G.1.1) en alfabeto grecoibérico.

Si aceptamos que el núcleo del paradigma es *-eórk-*, las formas pertenecientes probablemente a su paradigma serían las siguientes (incluyendo tres casos dudosos por presentar *-itií-*):

B.7.38	śalirbitiírokanabe bankuturbiteókan
C.1.24	salaker.itiíroketebon.
C.17.1]tinbašteeróke tortonbalarbiteórica**
C.21.7]a.banbite[/]róka.[
D.0.1	bastubarerteókan / utur

⁶La única excepción es **babiteáraataki**, sin interpunción, en B.7.38.

D.13.1	n.iskutur.iteroke.m̄
F.9.1	śalirbiter /]nanabekate biter /]ketanbale]roketan *bi*roketan
F.9.5	kutur.biteroketetine.
F.9.7	kutuboike.basbiteroketine
G.1.1	baseokeiunbaida. base / okar.
H.0.1	basbitero[tuśbiterokan. .biterokan kaultebiterokan
X.0.27 (La Joncosa)	eokarṁi arate[]rokarṁi]teerokarṁi

7.2.2. Los afijos

A partir de la lista anterior parece deducirse la existencia de una serie de afijos. En posición prenuclear tenemos: **b-** (deducible de la alternancia **bi-teroke/iteroke**, y quizás **itiroketebon**), **-it-** (por la misma razón, si efectivamente el núcleo es *erok*), **bas-**, **ban-** (que preceden o sustituyen a **b-it-**), y tal vez **t-**, si segmentamos D.0.1 como **bastubarer-teokan**. Los elementos **tuś** y **kaulte** que aparecen en H.0.1 precediendo a **biterokan** son dudosos, al aparecer una única vez. Quizás **tuś** pueda identificarse con el elemento onomástico **tuś**, lo que impediría su consideración como afijo gramatical.

En posición postnuclear aparecen **-an**, **-etan** (quizás compuesto del anterior), **-ar**, además de secuencias más largas, que en algunos casos pueden ser complejos sufijales (como **-tine**, **-tetine**, **-tebon**, con menos probabilidad **-betense**, **-iunbaida**). El sufijo **-ar** puede ser el mismo que aparece unido a nombres personales en inscripciones de propiedad, a veces seguido de **-ṁi** o **-enṁi**, y quizás podría indicar que estamos ante formas nominales del verbo, como veremos más adelante. Respecto a **iteroke** y **eroke**, no se puede

deducir de ellos un sufijo **-e**, pues puede tratarse de una oclusiva en posición final que no puede representarse como tal en la escritura ibérica.

Una justificación más completa de la segmentación propuesta se podría obtener identificando algunos de los afijos mencionados con otros núcleos de aspecto verbal, a lo cual se dedicará otro apartado.

7.3. Estructura del paradigma y expresiones regulares

El único intento hasta la fecha de establecer una estructura que integre todas las formas documentadas del paradigma se debe a de Hoz⁷, que presenta el siguiente esquema:

**ban/Ø-kutu(r)-(bo-ike)/Ø baś/Ø-b-iter'-o-ke-te/Ø-tine*

Naturalmente, una formulación así es el candidato perfecto para una expresión regular. Podríamos escribirla así:

`(ban)?kudur?(boike)?.*?(baS)?biteRoke(te)?tine`

Ésta sería la formulación más extensa posible del paradigma, incluyendo *(ban)kudur*, que aparece siempre separado, y podría ser un sintagma nominal (puede alternar con **Salir**). Tal como está, e incluso quitándole la primera parte, sólo ofrecería dos resultados. Es evidente que en esta formulación se ha expuesto tan sólo el camino más largo posible, que es el que permite **bite'roke-**, pues el iniciado por **bite'roka-** sólo lleva a una forma posible, que es **bite'rokan**. Pero ésta es precisamente la forma más abundante, y como nuestro objetivo es localizar el mayor número posible, pecando siempre por exceso antes que por defecto, habrá que modificar la expresión para que la incluya:

`(baS)?biteRok(an|e(te)?tine)`

Con este pequeño cambio ya pasamos de dos a siete coincidencias. Podemos aumentar a doce añadiendo interrogantes y contemplando la posibilidad de transcripción sonora de la *-t-* o de *-i-* en lugar de *-e-*:

`(baS)?b?iT(e|i)Rok?(an|e(te)?tine)?`

⁷de Hoz 2001a, p. 346

Pero si seguimos por este camino, y ponemos interrogante también tras *-o-*, para incluir la última línea de C.1.24 A, como hace de Hoz, obtendríamos demasiados resultados que claramente no tienen que ver con nuestra búsqueda. Se podrían obtener más coincidencias fiables englobando toda la expresión anterior entre paréntesis, e incluirla en una alternancia con otras expresiones que incluyan alternativamente la obligatoriedad del prefijo *b-* o del sufijo *-ok-*. Quedaría así:

((baS)?b?iT(e|i)Rok?(an|e(te)?tine)?|biT(e|i)R|eRok)

De este modo incluimos algún testimonio fragmentario que casi con seguridad se trata de nuestro paradigma (F.9.1), y alcanzamos un total de 22 testimonios, casi todos probablemente correctos, salvo dos o tres a lo sumo.

7.4. Posibles paradigmas relacionados

Si la identificación de *erók* como núcleo del paradigma es correcta, los diferentes afijos que muestra deberían lógicamente poder combinarse con otros núcleos. El problema es que, si el ibérico era una lengua incorporante, como propone de Hoz⁸ a partir precisamente de las formas que comentamos, podría haber en ellas más de un lexema, es decir, no sólo cabría esperar formas pronominales (como ocurre en vasco) que se repetirían idénticas con otros núcleos, sino que podría haber elementos nominales, con lo que sería estadísticamente muy difícil esperar repeticiones en el limitado conjunto de textos de que disponemos.

Admitido esto, no hay que renunciar por ello a la identificación de afijos verbales, siendo los mejores candidatos las formas más repetidas o las más breves de las que aparecen en torno a *erók*.

Es importante resaltar que lo que sigue se basa en la identificación de *erók* como el núcleo del paradigma, que no es segura ni mucho menos, por lo que los desarrollos ulteriores pueden tener cierto carácter especulativo. Si se emprende, a pesar de todo, este camino, es más para ejemplificar las posibilidades del programa que aquí describimos, y como muestra del inmenso campo de estudio que se nos abre en la segmentación de las palabras largas en textos ibéricos.

Podemos iniciar nuestra búsqueda tomando como modelo la forma **bi-teírokan**. Como hemos asumido que el núcleo es *erók*, para buscar formas equivalentes con otros núcleos podríamos usar una expresión regular como `bit\w+an`. No colocamos explícitamente límites de palabra, pues hemos visto

⁸de Hoz 2001a

que **bitérokan** puede aparecer en textos sin interpunción, o ir precedido de algún otro elemento, como en **kaul**tebitérokan**** (H.0.1). Además, con la expresión indicada no se obtiene un número elevado de resultados, con lo cual es fácil valorar cada caso. De hecho, sólo aparecen trece resultados, todos ellos sobre plomo, lo que de entrada es buena señal. Cinco de ellos pertenecen al paradigma de *érok*, y los restantes son los siguientes:

- **bitekian.keietíeie**ér.[/kutúraŕ[]er** (B.3.2)
- **aiunortinikabitáantesia** (C.21.6)
- **śalir.biteian.bantúer[** (D.0.1)
- **sukúrba.bitetean** (F.13.75)
- **ekaŕbabitiboan** (F.13.75)
- **sekebiterosan** (F.17.2)
- **bitebakiŕśbane** (F.9.5)
- **śanteun.biteian/].bantuŕaś.** (H.0.1)

Como se ve, se ha incluido el contexto con cierta amplitud cuando parece significativo. En F.17.2 aparecen en la línea anterior **teśibiteruka***], y en la posterior **textbfbirebośin**, que podrían tener también relación. Se señala con # el final del texto, pues una posición próxima al final puede apoyar una interpretación como verbo en una lengua SOV.⁹ En dos casos aparece al lado o muy cerca una palabra de las que suelen aparecer junto a **bitérok-**: **śalir** y **kutuŕ** (si es variante de **kutur**). En cuanto a **sekebiterosan**, ya había sido incluido por Untermann entre los representantes del paradigma de *biteŕ*, a pesar de la diferente vibrante. En cuanto a los dos ejemplos del plomo del Castellet de Bernabé (F.13.75), ya habían sido considerados como verbos por los editores del plomo,¹⁰ que según ellos acabaría con una doble secuencia objeto-verbo, siendo los objetos **sukuŕba**, **ekaŕba** y los verbos **bitetean**, **bitiboan**. También señalaban el posible carácter verbal de **biteian** y **bitekian**. Hay que insistir en que aquí los planteamientos de que se parte son diferentes. En particular, para Silgo **biteŕoke** sería una forma impersonal que iría seguida de formas flexivas como **tetine**, hipótesis que aquí no se comparte.

⁹de Hoz 2001a

¹⁰Guerin y Silgo 1996, p. 205

Una vez obtenida la lista anterior de formas que encajan con la expresión regular *bit\w+an*, el siguiente paso será ver si alguna de las nuevas raíces obtenidas se documenta con otros afijos, si esos afijos se dan también con otras formas del paradigma *érok*, y si se dan en contextos favorables a una interpretación verbal.

Empezaremos con **bitekian**, forma que aparece en la penúltima línea del plomo de Gruissan, y próxima a *kutu-*. De acuerdo con el razonamiento que aquí se expone, la raíz debería ser *eki*. Para buscar otras posibles formas de esa raíz no habrá más remedio que usar una expresión regular tan simple como *eki*, pues de entrada no sabemos con qué nos podemos encontrar. Como era de esperar, una secuencia tan breve produce un elevado número de concordancias (76). De ellas una veintena corresponden a la palabra **ekiar**, frecuente en cerámica de calidad y el mosaico de Caminreal, y a la que se atribuye, como ya hemos comentado, un significado que varía entre “hecho” y “obra”. Optar por el primero de estos sentidos nos permitiría, al tratarse de una forma verbal (impersonal en este caso), relacionarlo con **bitekian**. Para ello hay un apoyo importante, de reciente publicación. Se trata de un extenso grafito sobre cerámica, publicado por M^a Isabel Panosa,¹¹ y que resulta excepcional por su extensión siendo su soporte cerámico, e incluso sobre plomo habría resultado ser uno de los más largos. En este texto aparecen tres formas similares del paradigma de *éroke*: **érokarmī**, **aárate**[**rókarṁi** y **teerókaṁi**. En la última de ellas falta la *-r*, no sabemos si por descuido, pero las otras dos muestran claramente la raíz *érok* llevando como único afijo el sufijo *-r*, como en **ekiar**. Además, aunque *érok* suele producir formas complejas interpretables como verbos, aquí se trataría de formas nominales, como demuestra el hecho de ir seguidas por *mī*, que suele aparecer siguiendo a nombres personales. Por si fuera poco, dos de ellas van precedidas por *-te*, como ocurre con **ekiar**, que suele ir precedido de NP + *-te*. Por desgracia, en nuestro caso hay lagunas en el texto que impiden en ambos casos asegurar que se trate de dicho sufijo, y que vaya unido a un nombre personal.

Si **ekiar** es una raíz verbal, aunque aquí sea una forma nominal, naturalmente **ekien** también debería serlo, pues aparece en un contexto muy similar (el mosaico de Andelos). En principio, como señala de Hoz,¹² el sufijo *-en* es nominal, lo cual no descarta su uso en formas nominales, pero parecería raro que se añadiera a la raíz nuda. La otra posibilidad, que es considerarlo una forma personal (quizás en pasado), depende fundamentalmente de que se acepte la postura de Untermann de interpretar **(t)eban(en)** como equivalente de CVRAVIT.

¹¹Panosa 2002

¹²de Hoz 2001a

Otra palabra que contiene *eki* y de la que hay algún indicio de su valor verbal es **teki** (C.18.6). Se trata, en este caso, de una inscripción funeraria sobre piedra, famosa por su carácter parcialmente bilingüe. A la fórmula latina HEIC EST SIT[parece corresponder **aře.teki**. Si **teki** correspondiera a EST SITUS, ello dificultaría su pertenencia al mismo paradigma que **ekiar**, si efectivamente éste significara “hecho”, pero no podemos dejar de señalar la semejanza. Por otra parte, ello implicaría aceptar un prefijo *t-*, que una vez más dependería para su apoyo de la identificación **tebanen**=COERAV[IT] en el famoso bilingüe saguntino (F.11.8). En el paradigma de *érok* tan sólo podría apoyarse en la forma **bastubarerteókan** (D.0.1), aunque la falta de interpunción en esta cara del plomo impide confirmarlo, pues la grafía podría encubrir un *bastubarert(e) érokan*.

Ninguna de las demás apariciones del segmento *eki* ofrece afijos comunes con *érok* u otros indicios para considerar la posibilidad de que se trate de verbos. Si ensayamos el procedimiento seguido con *eki* con las demás palabras que responden a la expresión **bit\w+an**, no se obtienen tampoco otras coincidencias de apariencia verbal. Queda, pues, tan sólo la posibilidad de la existencia de un paradigma *eki*, que justificaría de paso la existencia de los afijos **b-it-** y **-an**, aunque la justificación es muy débil, pues se basa en la única forma **bitekian**. Por otra parte, no ha sido posible hallar testimonios claros de **-it-** sin **b-**.

Conclusiones

Con el trabajo precedente se ha intentado mostrar las posibilidades que ofrecen las expresiones regulares para el análisis de textos ibéricos. En particular, el programa que aquí se presenta facilita su uso a personas poco familiarizadas con ellas, mediante el uso de abreviaturas como **C**, **V**, etc. El simple uso del programa con estas abreviaturas y el metacarácter `\b` (límite de palabra) aumenta considerablemente la facilidad de las búsquedas, y permite obtener resultados que de otro modo podrían pasar por alto. Por ejemplo, el uso de las abreviaturas **Z** y **Q** (silbantes y vibrantes, respectivamente), permite obtener resultados en que no aparezca el sonido esperado por error del escriba, variante dialectal, o cualquier otra razón cuya valoración queda a criterio del usuario.

Como todo programa informático, el nuestro es susceptible de tener fallos o errores que sólo el uso irá revelando. En cualquier caso, siempre será posible mejorarlo y añadir nuevas funcionalidades, en función de las sugerencias de los usuarios.

En cuanto a los textos ibéricos, ya se ha dicho que aquí no se pretendía propiamente crear una base de datos epigráfica, sino que el interés se centraba en facilitar el análisis lingüístico de un corpus lo más completo posible. Pero es evidente que el sistema utilizado es lo bastante flexible como para convertirse en el embrión de un proyecto más amplio. El método de incluir la información adicional en archivos separados permite fácilmente incluir en ellos contenidos más amplios, incluyendo enlaces a archivos de imagen.

Por otra parte, el uso del lenguaje de programación PERL se revela como especialmente útil para trabajar con textos, incluso con grandes conjuntos de archivos de texto, permitiendo elaborar rápidamente programas sencillos que ahorran mucho trabajo, al permitir ejecutar rápidamente tareas repetitivas. Por ejemplo, en la creación de los archivos de información adicional se ha utilizado esta técnica, especialmente en conjuntos grandes de textos que compartían idéntica información, como es el caso de la cerámica de Azaila.

Por último, se ha intentado mostrar el uso del programa con el ejemplo concreto del paradigma de **biteroke**. Más allá de los resultados concretos ob-

tenidos, de validez discutible, se ha comprobado la rapidez con que pueden localizarse todas las formas relacionadas y, especialmente, verlas en su contexto, lo que no es posible con los léxicos existentes. Se ha mostrado también un atisbo de lo que puede ser la metodología para segmentar las palabras largas ibéricas, mediante la comparación en cadena.

Apéndice I: El código fuente

```
#!/usr/bin/perl

require "cgi-lib.pl";

MAIN:
{

&ReadParse(*input); #Lee la entrada de datos del formulario.

open (RESULTADO, ">resultado-iberico")||die "no puedo abrir RESULTADO";

system("ls *.*.* > listaficheros")==0; #guarda en un archivo el listado
#de archivos.

open (LISTA, "listaficheros")||die "no puedo abrir LISTA";

@lista_ficheros = <LISTA>;

close (LISTA);
$expresion1 = $input{'expresion1'};
$expresion1 =~ s/([adegiklmoprtsRSMbns])(?!\\W)/$1\\n*/g; #
#Para prevenir palabras cortadas por salto de línea.
$expresion2 = $input{'expresion2'};
$expresion2 =~ s/([adegiklmoprtsRSMbns])(?!\\W)/$1\\n*/g;
$numerofichero = @lista_ficheros;
if ($input{'num'}) {$num = $input{'num'}} else {$num = "1"}
#Se almacena en $num el número de caracteres mínimo que
#deberá tener un texto para ser analizado
$i=0; #variables para el contador.
$e=0;
```

```

#A continuación, las abreviaturas de grupos de sonidos:
$expresion1 =~ s/C/[bptdkgmMnrRsSl]/g; #consonantes
$expresion1 =~ s/V/[aeiou]/g; #vocales
$expresion1 =~ s/T/[td]/g; #dentales
$expresion1 =~ s/K/[kg]/g; #velares
$expresion1 =~ s/P/[pbtdkg]/g; #oclusivas
$expresion1 =~ s/Z/[sS]/g; #silbantes
$expresion1 =~ s/Q/[rR]/g; #vibrantes
$expresion1 =~ s/Y/([btdkgmMnsSlrR]?[aeiou][iu]?[nmMlrR]?[sS]?)/g;
#sílabas
$expresion1 =~ s/L/[nmMlrR]/g; #sonantes
#Se inicia el bucle que abre y analiza uno por uno todos los
#archivos de textos.
for ($indice = 0; $indice < $numerofichero; $indice++) {
    open(TEXTOS, "<@lista_ficheros[$indice]");
    open(TEXTINF, "<./inf/@lista_ficheros[$indice]");
    undef $/; #Para que <TEXTOS> contenga todas las líneas del texto.
    $texto = <TEXTOS>;
    $textinf = <TEXTINF>;
#A continuación, el contador:
    if ($texto =~ /.{ $num }/) {
        while ($texto =~ /$expresion1/g) {
            $i++;
        }
        if ($expresion2 =~ /.+/) {
            while ($texto =~ /$expresion2/g) {
                $e++;
            }
        }
        if ($texto =~ /$expresion1/g) {
            #Si se localiza la expresión, se marca en rojo
            #y negrita.
            $texto =~ s/$expresion1/<FONT COLOR=RED><STRONG>$$&
                <\/STRONG><\/FONT>/gs;
            if ($texto =~ /$expresion2/g) {
                $texto =~ s/$expresion2/<FONT COLOR=RED><STRONG>$$&
                    <\/STRONG><\/FONT>/gs if ($expresion2 =~ /.+/);
                #A continuación, el bloque que busca nombres propios.
            if ($input{'np'}){
                open (NOMBRES, "<nombres" ) ||
                    die "no puedo abrir nombres";
            }
        }
    }
}

```



```

        $/ = "\n"; #Para que cada línea del archivo de
#nombres sea un elemento de @nombres.
        @nombres = <NOMBRES>;
        undef $/;
        $nombres = @nombres;
        #A continuación, se compara el texto con cada uno
#de los nombres contenidos en el archivo:
        for($x = 0; $x < $nombres; $x++){
            $nombre = @nombres[$x];
            $nombre =~ s/\n*/g; #se elimina el carácter
#de salto de línea
            $nombre =~ s/[abdegiklmnoprstuMRS](?!\/)/
            $&\s*\n*/g; #Para prevenir nombres
#cortados por salto de línea.
            if ($texto =~ /$nombre/) {
                $texto =~ s/$nombre/<U>$&<\U>/g;
#Si coincide, se subraya el nombre propio.
            }
        }
        close (NOMBRES);
    }
#Se almacena en RESULTADO el nombre del texto, su información
#complementaria, y el texto ya modificado:
    print RESULTADO ("<h3>@lista_ficheros[$indice]</h3>");
    print RESULTADO (" $textinf");
    print RESULTADO (" $texto");
    print RESULTADO ("\n\n");
}
}
}
close (TEXTOS);
close (RESULTADO);
open (RESULTADO, "<resultado-iberico")||die "no puedo abrir resultado";
@resultado = <RESULTADO>;
    print &PrintHeader; #Se inicia aquí la página web de salida
    print <<ENDOFTEXT;
    <html>
    <head>
    <title>Resultado de la b&uacute;squeda:</title>
    </head>

```

```

    <body>
    <h1>Resultados de la búsqueda</h1>
    <h2>Expresión: $input{'expresion1'},
      $input{'expresion2'}</h2> que
      equivale(n) a $expresion1, $expresion2.
    <h2>Total: $i, $e </h2>
    <pre>@resultado</pre>
    <h2>Total expresión 1: $i</h2>
    <h2>Total expresión 2: $e</h2>
ENDOFTEXT
}
print &HtmlBot;
```

Apéndice II: el programa buscador por nomenclatura MLH

```
#!/usr/bin/perl

require "cgi-lib.pl";

MAIN:
{

&ReadParse(*input);

open (RESULTADO, ">resultado-iberico")||die "no puedo abrir RESULTADO";

    $nombre = $input{'nombre'};
    open(TEXTOS, "<$nombre")||die "No existe ese archivo";
    open(TEXTINF, "<./inf/$nombre")||die "No existe ese archivo";
    undef $/;
    $texto = <TEXTOS>;
    $textinf = <TEXTINF>;

        print RESULTADO ("

### $nombre</h3>"); print RESULTADO ("$textinf"); print RESULTADO ("$texto"); print RESULTADO ("\n\n"); close (TEXTOS); close (RESULTADO);


```

```
open (RESULTADO, "<resultado-iberico")||die "no puedo abrir resultado";

@resultado = <RESULTADO>;

    print &PrintHeader;
    print <<ENDOFTEXT;
    <html>
    <head>
    <title>Resultado de la b&uacute;squeda:</title>
    </head>
    <body>

    <pre>@resultado</pre>

ENDOFTEXT

    print &HtmlBot;
}
```

Apéndice III: nomenclatura provisional de textos no incluídos en MLH

X.0.003

Ostracón de Pontós

=C.3.1

X.0.004

Estela de arenisca. Bell-lloc (La Plana Alta).

*Arasa 89, p. 93.

X.0.005

Bronce de Aranguren

Bronce. Aranguren (Navarra).

*Beltrán 93, p. 91.

X.0.006

Villares VIII

Cerámica. Los Villares (Caudete de las Fuentes, Valencia).

*Bonet 89, p. 138.

X.0.007

Puntal dels Llops II

Hueso. Puntal dels Llops (Olocau, Valencia).

*Bonet 89, p. 140.

X.0.008

Valencia II
Cerámica. Valencia.
*Bonet 89, p. 142.

X.0.009

Latour de Carol
Grafito rupestre. Latour de Carol (Cerdaña francesa).
*Campmajo 93, p. 505.

X.0.010

Err I
Grafito rupestre. Err (Cerdaña francesa).
*Campmajo 93, p. 506.

X.0.011

Err 3
Grafito rupestre. Err (Cerdaña francesa).
*Campmajo 93, p. 506.

X.0.012

Err 4
Grafito rupestre. Err (Cerdaña francesa).
*Campmajo 93, p. 506.

X.0.013

Osséja 1
Grafito rupestre. Osséja (Cerdaña francesa).
*Campmajo 93, p. 507.

X.0.014

Osséja 2

Grafito rupestre. Osséja (Cerdaña francesa).
*Campmajo 93, p. 507.

X.0.015

Osséja 3
Grafito rupestre. Osséja (Cerdaña francesa).
*Campmajo 93, p. 509.

X.0.016

Osséja 4
Grafito rupestre. Osséja (Cerdaña francesa).
*Campmajo 93, p. 512.

X.0.017

Osséja 5, Roche Gassiot A
Grafito rupestre. Osséja (Cerdaña francesa).
*Campmajo 93, p. 513.

X.0.018

Osséja 6, Roche Gassiot B
Grafito rupestre. Osséja (Cerdaña francesa).
*Campmajo 93, p. 514.

X.0.019

Osséja 7, Roche Gassiot 1
Grafito rupestre. Osséja (Cerdaña francesa).
*Campmajo 93, p. 514.

X.0.020

Osséja 8, Roche Gassiot 6
Grafito rupestre. Osséja (Cerdaña francesa).
*Campmajo 93, p. 515.

X.0.021

Osséja 9, Roche Gassiot
Grafito rupestre. Osséja (Cerdaña francesa).
*Campmajo 93, 516.

X.0.022

Osséja 10, Roche Gassiot 10-11
Grafito rupestre. Osséja (Cerdaña francesa).
*Campmajo 93, p.516.

X.0.023

Osséja 11, Carrière de la Cabanette. Roche 1/4
Grafito rupestre. Osséja (Cerdaña francesa).
*Campmajo 93, p. 516.

X.0.024

Osséja 12, Carrière de la Cabanette, Roche 6
Grafito rupestre. Osséja (Cerdaña francesa).
*Campmajo 93, p. 517.

X.0.025

Gails A
Grafito rupestre. Gails (Cerdaña francesa).
*Campmajo 93, p. 517.

X.0.026

Gails 1, Roche 5
Grafito rupestre. Gails (Cerdaña francesa).
*Campmajo 93, p. 517.

X.0.027

Gails 2, Roche 1/1
Grafito rupestre. Gails (Cerdaña francesa).
*Campmajo 93, p. 518.

X.0.028

Cerámica. Baeza (Jaén).
Alfabeto sudibérico.
*Correa 89.

X.0.029

Cerámica. Elna (Rosellón, Francia).
*Cura 86, p.207.

X.0.030

Cerámica. Tornabous (Lérida).
*Cura 93, p. 219.

X.0.031

Cerámica. Tornabous (Lérida).
*Cura 93, p. 219.

X.0.032

Falcata. Sagunto (?) (Valencia).
*De Hoz 92, 330.
V1-Rodríguez Ramos 98, p. 228

X.0.033

Plomo. Sagunto (Valencia).
Alfabeto grecoibérico.
*Untermann 96, p. 101.
V1-Fletcher 91, p. 1.

X.0.034

Lápida de Terrateig
Estela de calcáreo. Camí del Molí (Terrateig, Valencia).
*Velaza 96, p. 325.
Fletcher 94.

X.0.035

Piedra. San Mateo (Castellón).

*Fletcher 81b, p. 207.

X.0.036

Moneda. Sagunto (Valencia).

*Ripollés 2001, p. 167.

X.0.038

Disco de plomo. Pico de los Ajos (Yátova, Valencia).

Alfabeto sudibérico.

*Tomás 89, p. 169.

X.0.039

Cerámica. Can Vedell (Bigues i Riells, Barcelona).

*Panosa 92, p. 60.

X.0.041

Cerámica. Palou (Granollers, Barcelona).

*Panosa 92, p. 62.

X.0.042

Cerámica. Turó del Vent, El Far (Llinars del Vallès, Barcelona).

*Panosa 92, p.67.

X.0.043

Cerámica. Turó del Vent. El Far (Llinars del Vallès, Barcelona).

*Panosa 92, p. 67.

X.0.045

Cerámica. Turó del Vent, El Far (Llinars del Vallès, Barcelona).

*Panosa 92, p. 68.

X.0.046

Cerámica. Ciutadella de Roses (Roses, Gerona).
*Panosa 93, p. 178.

X.0.047

Cerámica. Mas Castellar (Pontós, Gerona).
*Panosa 93, p. 179.

X.0.048

Cerámica. Ampurias (La Escala, Gerona).
*Panosa 93, p. 180.

X.0.049

Cerámica. Montbarbat (Lloret de Mar, Gerona).
*Panosa 93, p. 181.

X.0.050

Cerámica. Burriac (Cabrera de Mar, Barcelona).
*Panosa 93, p. 181.

X.0.051

Cerámica. Burriac (Cabrera de Mar, Barcelona).
*Panosa 93, p. 182.

X.0.052

Cerámica. Burriac (Cabrera de Mar, Barcelona).
*Panosa 93, p. 183.

X.0.053

Cerámica. Can Rodon del Hort (Cabrera de Mar, Barcelona).
*Panosa 93, p. 184.

X.0.054

Fusayola de cerámica. Can Miralles-Can Modolell (Cabrera de Mar, Barcelona).

*Panosa 93, p. 184.

X.0.055

Necrópolis del Turó dels Dos Pins (Cabrera de Mar, Barcelona).

*Panosa 93, p. 185.

X.0.056

Cerámica. Cadira del Bisbe (Premià de Dalt, Barcelona).

*Panosa 93

X.0.057

Cerámica. Badalona (Barcelona).

*Panosa 93, p. 192.

X.0.058

Cerámica. Badalona (Barcelona).

*Panosa 93, p. 194.

X.0.059

Cerámica. La Pedrera (Montjuïc, Barcelona).

*Panosa 93, p. 197.

X.0.060

Cerámica. Sant Boi de Llobregat (Barcelona).

*Panosa 93, p. 203.

X.0.061

Cerámica. Iglesia de Santa María de Egara (Terrassa, Barcelona).

*Panosa 93, p. 205.

X.0.062

Cerámica. Turó de Ca n'Oliver (Cerdanyola, Barcelona).
*Panosa 93, p. 206.

X.0.063

Cerámca. Turó de Ca n'Oliver (Cerdanyola, Barcelona).
*Panosa 93, p., 206.

X.0.064

Cerámica. Puig Cardener (Manresa, Barcelona).
*Panosa 93, 207.

X.0.065

Cerámica. Els Prats de Rei (Barcelona).
*Panosa 93, p. 213.

X.0.066

Cerámica. Els Prats de Rei (Barcelona).
*Panosa 93, p. 214.

X.0.067

Fusayola de cerámica. El Vilar (Valls, Tarragona).
*Panosa 93, p. 215.

X.0.068

Plomo. Procedencia desconocida.
*Velaza 96, p. 318

X.0.069

Cerámica. Pollentia (Alcúdia, Mallorca).
*Velaza 96, p. 329.

X.0.070

Estela romana de piedra. Requena (Valencia).

*Velaza 96, p. 330.

(finales I d.C.-II d.C)

X.0.072

Plomo. La Balaguera (La Pobla Tornesa).

*Velaza 2001, p. 642.

Allepuz 96, p. 6

X.0.073

Plomo. La Balaguera (La Pobla Tornesa).

*Velaza 2001, p. 642.

Allepuz 96, p. 6.

X.0.074

Inscripción rupestre. Abrigo Burgal.

*Velaza 2001, p.649.

X.0.075

Inscripción rupestre. Los Maíllos.

*Velaza 2001, p. 649.

X.0.076

Terra sigillata. Tossal de Manises.

*Velaza 2001, p. 661.

Llobregat-Roser Limiñana 1993, pp. 119-135.

X.0.077

Estela de piedra. Badalona (Barcelona).

*Comas 2001, p.295.

X.0.078

Estela de piedra. Badalona (Barcelona).

Comas 2001, p. 296.

X.0.079

Fusayola. El Palomar (Oliete, Teruel).

*Silgo 2001, p. 347.

X.0.080

Fusayola. El Palomar (Oliete, Teruel).

*Silgo 2001, p. 348.

X.0.081

Pondus. El Palomar (Oliete, Teruel).

*Silgo 2001, p. 348.

X.0.082

Cerámica. El Palomar (Oliete, Teruel).

*Silgo 2001, p. 348.

X.0.083

Disco de piedra. Bellver de Cerdanya (Gerona).

*Panosa 2001, p. 513.

X.0.084

Inscripción rupestre. Devesa de Sallent (Bolvir, Gerona).

*Panosa 2001, p. 515.

X.0.085

Plaquita de cuerno. Els Cortals (Cervera, Lérida).

*Panosa 2001, p. 516.

X.0.086

Cerámica. Castell de Cornellà (Barcelona).

*Panosa 2001, p. 517.

X.0.087

Cerámica. Les Soleies (Esparraguera, Barcelona).

*Panosa 2001, p. 517.

X.0.088

Cerámica. Mas Castellar (Pontós, Gerona).

*Panosa 2001, p. 518.

X.0.089

Cerámica. Jardí Park (Premià de Dalt, Barcelona).

*Panosa 2001, p. 520.

X.0.090

Cerámica. Cadira del Bisbe (Premià de Dalt, Barcelona).

*Panosa 2001, p. 521.

X.0.091

Cerámica. Cadira del Bisbe (Premià de Dalt, Barcelona).

*Panosa 2001, p. 522.

X.0.092

Cerámica. Cadira del Bisbe (Premià de Dalt, Barcelona).

*Panosa 2001, p. 522.

X.0.093

Cerámica. Can Feu (Sant Quirze del Vallès, Barcelona).

*Panosa 2001, p. 524.

X.0.094

Cerámica. Can Feu (Sant Quirze del Vallès, Barcelona).

*Panosa 2001, p. 524.

X.0.095

Cerámica. Can Feu (Sant Quirze del Vallès, Barcelona).

*Panosa 2001, p. 524.

X.0.096

Cerámica. Roques de Sant Formatge (Seròs, Lérida).

*Panosa 2001, p. 527.

X.0.097

Cerámica. Camp de les Lloses (Tona, Barcelona).

*Panosa 2001, p. 530.

X.0.098

Plomo. La Mazorra (Utiel, Valencia).

*Fletcher 82b, p. 253.

X.0.099

Plomo. Aliaguilla (Cuenca).

*Fletcher 92b, p. 254

X.0.100

Pico de los Ajos IV

Plomo (Bala de honda). Pico de los Ajos (Yátova, Valencia).

*Fletcher 82b, p. 255.

X.0.101

Pico de los Ajos V

Plomo. Pico de los Ajos (Yátova, Valencia).

*Fletcher 82b, p.256.

X.0.102

Sagunto 35

Plomo. Sagunto.

*Silgo 87b, p. 671.

X.0.103

Sagunto 36
Plomo. Sagunto.
*Silgo 87b, p. 673.

X.0.104

Orleyl XI
Plomo. Orleyl.
*Silgo 89, p.833.

X.0.105

Sagunto XLVIII
Grafito sobre cerámica. Sagunto.
*Silgo 89, p. 834.

X.0.106

Liria CX
Cerámica. Liria.
*Silgo 89, p. 834.

X.0.107

Sagunto 58
Plomo. Sagunto.
*Silgo 97b, p. 82.

X.0.108

Plomo. Camp de Morvedre (Sagunto).
*Silgo 2000e, p. 40.

X.0.109

Sagunto 30
Piedra. Montaña Frontera.
*Fletcher 87, p. 666.

V1-Roca 84, p.57.

X.0.110

Sagunto 29
Piedra. Montaña Frontera.
*Fletcher 87, p.666.
Roca 84, p. 57.

X.0.111

Sagunto 31
Piedra. Montaña Frontera.
*Fletcher 87, p. 666.
Roca 84, p. 57.

X.0.112

Sagunto 32
Cerámica. Grau Vell.
*Fletcher 87, p. 666.

X.0.113

Sagunto 33
Cerámica. Montaña Frontera.
*Fletcher 87, p. 666.

X.0.114

Sagunto 37
Piedra. Montaña Frontera.
*Fletcher 87, p. 667.

X.0.115

Sagunto 38
Piedra. Montaña Frontera.
*Fletcher 87, p. 667.

X.0.116

Sagunto 39
Piedra. Montaña Frontera.
*Fletcher 89, p. 667.

X.0.117

Cerámica. Turó dels Oriols (Cabrera de Mar, Barcelona).
*Panosa 2001, p. 516.

X.0.118

Cerámica. Mas Castellar (Pontós, Gerona).
*Panosa 2001, p. 518.

X.0.119

Cerámica. Mas Castellar (Pontós, Gerona).
*Panosa 2001, p. 519.

X.0.120

Cerámica. Els Guíxols (Sant Feliu de Guíxols, Gerona).
*Panosa 2001, p. 523.

X.0.121

Cerámica. Tarragona.
*Panosa 2001, p. 528.

X.0.122

Cerámica. Tarragona.
*Panosa 2001, p. 528.

X.0.123

Cerámica. Castellet de Banyoles (Tivissa, Tarragona).
*Panosa 2001, p. 529.

X.0.124

Cerámica. El Fonollar (Vallbona d'Anoia, Barcelona).

*Panosa 2001, p. 532.

X.0.125

Cerámica. Mas Castellar (Pontós, Gerona).

*Panosa 93, p. 179.

X.0.126

Cerámica. Ampurias (La Escala, Gerona).

*Panosa 93, p. 179.

X.0.127

Cerámica. Montbarbat (Lloret de Mar, Gerona).

*Panosa 93, p. 181.

X.0.128

Cerámica. Burriac (Cabrera de Mar, Barcelona).

*Panosa 93, p. 183.

X.0.129

Cerámica. Necrópolis del Turó dels Dos Pins (Cabrera de Mar, Barcelona).

*Panosa 93, p. 186.

X.0.130

Cerámica. Cal Ros de les Cabres (El Masnou, Barcelona).

*Panosa 93, p. 188.

X.0.131

Cerámica. Badalona (Barcelona).

*Panosa 93, p. 188.

X.0.132

Cerámica. Badalona (Barcelona).
*Panosa 93, p. 189.

X.0.133

Cerámica. Badalona (Barcelona).
*Panosa 93, p. 189.

X.0.134

Cerámica. Badalona (Barcelona).
*Panosa 93, p. 190.

X.0.135

Cerámica. Badalona (Barcelona).
*Panosa 93, p. 191.

X.0.136

Cerámica. Badalona (Barcelona).
*Panosa 93, p. 191.

X.0.137

Cerámica. Badalona (Barcelona).
*Panosa 93, p. 193.

X.0.138

Cerámica. Badalona (Barcelona).
*Panosa 93, p. 194.

X.0.139

Cerámica. Silos de Magòria (Montjuïc, Barcelona).
*Panosa 93, p. 195.

X.0.140

Cerámica. Camí Vell del Llor (Sant Boi de Llobregat, Barcelona).

*Panosa 93, p. 197.

X.0.141

Cerámica. Camí Vell del Llor (Sant Boi de Llobregat, Barcelona).

*Panosa 93, p. 198.

X.0.142

Cerámica. Camí Vell del Llor (Sant Boi de Llobregat, Barcelona).

*Panosa 93, p. 199.

X.0.143

Cerámica. Camí Vell del Llor (Sant Boi de Llobregat, Barcelona).

*Panosa 93, p. 199.

X.0.144

Cerámica. Camí Vell del Llor (Sant Boi de Llobregat, Barcelona).

*Panosa 93, p. 199.

X.0.145

Cerámica. Camí Vell del Llor (Sant Boi de Llobregat, Barcelona).

*Panosa 93, p. 200.

X.0.146

Cerámica. Sant Boi de Llobregat (Barcelona).

*Panosa 93, p. 201.

X.0.147

Cerámica. Sant Boi de Llobregat (Barcelona).

*Panosa 93, p. 201.

X.0.148

Cerámica. Sant Boi de Llobregat (Barcelona).

*Panosa 93, p. 202.

X.0.149

Cerámica. Terrassa (Barcelona).

*Panosa 93, p. 203.

X.0.150

Cerámica. Iglesia de Santa María de Egara (Terrassa, Barcelona).

*Panosa 93, p. 204.

X.0.151

Cerámica. Can Fatjó (Rubí, Barcelona).

*Panosa 93, p. 205.

X.0.152

Cerámica. Boades (Castellgalí, Barcelona).

*Panosa 93, p. 208.

X.0.153

Cerámica. Boades (Castellgalí, Barcelona).

*Panosa 93, p. 209.

X.0.154

Cerámica. Boades (Castellgalí, Barcelona).

*Panosa 93, p. 209.

X.0.155

Cerámica. Els Prats de Rei (Barcelona).

*Panosa 93, p. 210.

X.0.156

Cerámica. Els Prats de Rei (Barcelona).

*Panosa 93, p. 210.

X.0.157

Cerámica. Els Prats de Rei (Barcelona).

*Panosa 93, p., 211.

X.0.158

Cerámica. Els Prats de Rei (Barcelona).

*Panosa 93, p. 212.

X.0.159

Cerámica. Els Prats de Rei (Barcelona).

*Panosa 93, p. 212.

X.0.160

Cerámica. L'Aubreda (Sant Martí de Tous, Barcelona).

*Panosa 93, p. 214.

X.0.161

Cerámica. Pla de les Sitges del Camaró (La Torre de Claramunt, Barcelona).

*Panosa 93, p. 215.

X.0.162

Cerámica. Castell d'Amposta (Amposta, Tarragona).

*Panosa 93, p. 217.

X.0.163

Cerámica. Castell d'Amposta (Amposta, Tarragona).

*Panosa 93, p. 217.

X.0.164

Cerámica. Sant Feliu de Llo (Alta Cerdaña, Francia).

*Panosa 93, p. 218.

X.0.165

Cerámica. Can Vedell (Bigues i Riells, Barcelona).
*Panosa 92, p. 57.

X.0.166

Cerámica. Can Vedell (Bigues i Riells, Barcelona).
*Panosa 92, p. 57.

X.0.167

Cerámica. Can Vedell (Bigues i Riells, Barcelona).
*Panosa 92, p. 60.

X.0.168

Cerámica. Can Vedell (Bigues i Riells, Barcelona).
*Panosa 92, p. 61.

X.0.169

Cerámica. Turó de Can Pertegaç, Canovelles (Velulla, Barcelona).
*Panosa 92, p. 62.

X.0.170

Cerámica. Pla d'en Rifanyès (Cardedeu, Barcelona).
*Panosa 92, p. 62.

X.0.171

Cerámica. Pla d'en Rifanyès (Cardedeu, Barcelona).
*Panosa 92, p. 62.

X.0.172

Cerámica. Pla d'en Rifanyès (Cardedeu, Barcelona).
*Panosa 92, p. 62.

X.0.173

Cerámica. Urbanització de Çan Carlons”, L’Ametlla (Barcelona).
*Panosa 92, p. 63.

X.0.174

Cerámica. Can Bordoí (Cardedeu, Barcelona).
*Panosa 92, p. 64.

X.0.175

Cerámica. Can Rosell (Llinars del Vallès, Barcelona).
*Panosa 92, p. 64.

X.0.176

Cerámica. Can Rosell (Llinars del Vallès, Barcelona).
*Panosa 92, p. 64.

X.0.177

Cerámica. Can Rosell (Llinars del Vallès, Barcelona).
*Panosa 92, p. 64.

X.0.178

Cerámica. Pont de Rialbs (Llinars del Vallès, Barcelona).
*Panosa 92, p. 65.

X.0.179

Cerámica. Turó del Vent. El Far (Llinars del Vallès, Barcelona).
*Panosa 92, p. 65.

X.0.180

Cerámica. Turó del Vent. El Far (Llinars del Vallès, Barcelona).
*Panosa 92, p. 65.

X.0.181

Cerámica. Turó del Vent. El Far (Llinars del Vallès, Barcelona).
*Panosa 92, p. 65.

X.0.182

Cerámica. Turó del Vent. El Far (Llinars del Vallès, Barcelona).
*Panosa 92, p. 66.

X.0.183

Cerámica. Turó del Vent. El Far (Llinars del Vallès, Barcelona).
*Panosa 92, p. 66.

X.0.184

Cerámica. Turó del Vent. El Far (Llinars del Vallès, Barcelona).
*Panosa 92, p. 66.

X.0.185

Cerámica. Turó del Vent. El Far (Llinars del Vallès, Barcelona).
*Panosa 92, p. 66.

X.0.186

Cerámica. Turó del Vent. El Far (Llinars del Vallès, Barcelona).
*Panosa 92, p. 67.

X.0.187

Cerámica. Turó del Vent. El Far (Llinars del Vallès, Barcelona).
*Panosa 92, p. 68.

X.0.188

Cerámica. Castellruf. Santa María de Martorelles de Dalt (Barcelona).
*Panosa 92, p. 68.

X.0.189

Cerámica. Castellruf. Santa María de Martorelles de Dalt (Barcelona).
*Panosa 92, p. 68.

X.0.190

Cerámica. Bosc Gran (Urbanización Can Falguera). Palau de Plegamans (Barcelona).
*Panosa 92, p. 69.

X.0.191

Cerámica. Bosc Gran. Urbanización Can Falguera. Palau de Plegamans (Barcelona).
*Panosa 92, p. 69.

X.0.192

Cerámica. Can Record, Sant Antoni de Vilamajor (Barcelona).
*Panosa 92, p. 69.

X.0.193

Cerámica. Castell de Monclús. Sant Esteve de Palautordera (Barcelona).
*Panosa 92, p. 70.

X.0.194

Cerámica. Castell de Monclús, Sant Esteve de Palautordera (Barcelona).
*Panosa 92, p. 70.

X.0.195

Cerámica. Castell de Monclús, Sant Esteve de Palautordera.
*Panosa 92, p. 70.

X.0.196

Cerámica. Castell de Monclús, Sant Esteve de Palautordera (Barcelona).
*Panosa 92, p. 70.

X.0.197

Cerámica. Castell de Monclús, Sant Esteve de Palautordera (Barcelona).
*Panosa 92, p. 70.

X.0.198

Cerámica. Castell de Monclús, Sant Esteve de Palautordera (Barcelona).
*Panosa 92, p. 71.

X.0.199

Cerámica. Castell de Monclús, Sant Esteve de Palautordera (Barcelona).
*Panosa 92, p. 71.

X.0.200

Cerámica. Turó de la Galaieta (Sant Feliu de Codines, Barcelona).
*Panosa 92, p. 71.

X.0.201

Cerámica. Can Puig, Santa Eulàlia de Ronçana (Barcelona).
*Panosa 92, p. 71.

X.0.202

Cerámica. Turó de Pertegaç (Velulla), Canovelles (Barcelona).
*Panosa 92, p. 62.

X.0.203

Cerámica. Torrelló del Boverot (Alzamora, Castellón).
Escrito de derecha a izquierda.
*Velaza 2002, p. 413 (Lectura de Rodríguez Ramos).

X.0.204

Pondus de cerámica. Calafell (Tarragona).
*Velaza 2002, p. 413.

X.0.205

Plomo. Montlaurès (Francia).

*Untermann 2002, p. 358.

X.0.206

Plomo. Badalona.

*Comas 2002, p. 331.

X.0.207

Cerámica. La Joncosa (Jorba, Barcelona).

*Panosa 2002, p. 336.

X.0.208

Cerámica. Lloret de Mar (?) (Gerona).

*Vilà 96, p. 296

X.0.209

Piedra. Tarragona.

*Massó 95, p. 119

Addenda

Con posterioridad a la redacción definitiva de este trabajo se han realizado pequeñas mejoras en el programa, que se comentan aquí brevemente. Se trata, por una parte, de las abreviaturas correspondientes a la segunda expresión regular que puede introducir el usuario, y que se habían omitido por error. Por otra parte, se ha mejorado la presentación del resultado en dos aspectos: anteriormente los textos no aparecían ordenados correctamente, pues el sistema operativo considera los nombres de archivos siempre como letras, aunque contengan números, y ordena éstos por su valor en el código ASCII. Para que la ordenación sea correcta, todos los nombres deben tener el mismo número de dígitos. Por tanto, ha habido que sustituir nombres como C.1.24 por C.001.024. Para respetar la nomenclatura original, se han introducido unas líneas de código que eliminan los ceros superfluos justo antes de enviarlos al navegador, cuando los textos ya están correctamente ordenados.

La segunda mejora estética consiste en sustituir, también justo antes de enviar el resultado al navegador, los caracteres S y R por *ś* y *ř*, para los que existen, respectivamente, los códigos HTML `ś` y `ŕ`. Como estos signos no se visualizan bien en navegadores antiguos, se ha dejado la transformación como una opción para el usuario. Para *ñ* lo más parecido es *ń*, cuyo código es `ḿ`, que además no se visualiza en el navegador Microsoft Explorer, por lo que se ha optado por dejar la transcripción como M. Por supuesto, esto afecta tan sólo al resultado, manteniéndose S, R y M para introducir la cadena de texto a buscar.

Por último, con el fin de evitar que las variantes de lectura hagan que el contador no resulte de utilidad, se han separado éstas en archivos independientes, con el mismo nombre pero en un directorio separado, de nombre **var**, y se ha dejado como opción para el usuario que el buscador las analice también en busca de la expresión regular, aunque el texto se presenta en cualquier caso.

Se transcribe de nuevo, a continuación, el código del programa, señalando las líneas añadidas con el comentario “nuevo”.


```

#!/usr/bin/perl

require "cgi-lib.pl";

MAIN:
{

&ReadParse(*input);

open (RESULTADO, ">resultado-iberico")||die "no puedo abrir RESULTADO";

system("ls -U *.*.* > listaficheros")==0;

open (LISTA, "listaficheros")||die "no puedo abrir LISTA";

@lista_ficheros = <LISTA>;

close (LISTA);

    $expresion1 = $input{'expresion1'};
    $expresion1 =~ s/([adegiklmoprtruRSMbns])(?!\\W)/$1\\n*/g;
    $expresion2 = $input{'expresion2'};
    $expresion2 =~ s/([adegiklmoprtruRSMbns])(?!\\W)/$1\\n*/g;
    $numerofichero = @lista_ficheros;
    if ($input{'num'}) {$num = $input{'num'}} else {$num = "1"}
    $i=0;
    $e=0;
    $expresion1 =~ s/C/[bptdkgmMnrRsSl]/g; #consonantes
    $expresion1 =~ s/V/[aeiou]/g; #vocales
    $expresion1 =~ s/T/[td]/g; #dentales
    $expresion1 =~ s/K/[kg]/g; #velares
    $expresion1 =~ s/P/[pbtdkg]/g; #oclusivas
    $expresion1 =~ s/Z/[sS]/g; #silbantes
    $expresion1 =~ s/Q/[rR]/g; #vibrantes
    $expresion1 =~ s/Y/([btdkgmMnsSlrR]?[aeiou][iu]?[nmMlrR]?[sS]?)/g; #sílabas
    $expresion1 =~ s/L/[nmMlrR]/g; #sonantes

    $expresion2 =~ s/C/[bptdkgmMnrRsSl]/g; #consonantes (nuevo)
    $expresion2 =~ s/V/[aeiou]/g; #vocales (nuevo)
    $expresion2 =~ s/T/[td]/g; #dentales (nuevo)
    $expresion2 =~ s/K/[kg]/g; #velares (nuevo)

```

```

$expresion2 =~ s/P/[pbtdkg]/g; #oclusivas (nuevo)
$expresion2 =~ s/Z/[sS]/g; #silbantes (nuevo)
$expresion2 =~ s/Q/[rR]/g; #vibrantes (nuevo)
$expresion2 =~ s/Y/([btdkgmMnsSlrR]?[aeiou][iu]?
                    [nmMlrR]?[sS]?)/g; #sílabas (nuevo)
$expresion2 =~ s/L/[nmMlrR]/g; #sonantes

for ($indice = 0; $indice < $numerofichero; $indice++) {

    open(TEXTOS, "<@lista_ficheros[$indice]");
    open(TEXTINF, "<./inf/@lista_ficheros[$indice]");
    undef $/;
    $texto = <TEXTOS>;
    $textinf = <TEXTINF>;
    if ($texto =~ /.{ $num }/) {
        while ($texto =~ /$expresion1/g) {
            $i++;
        }
        if ($expresion2 =~ /.+/) {
            while ($texto =~ /$expresion2/g) {
                $e++;
            }
        }
    }

    if ($texto =~ /$expresion1/g) {

        $texto =~ s/$expresion1/<FONT color=red>
                    <strong>$&</strong></FONT>/g;

        if ($texto =~ /$expresion2/g) {

            $texto =~ s/$expresion2/<FONT
color=red><strong>$&</strong></FONT>/g if ($expresion2 =~ /.+/);

        }

    }

    if ($input{'np'}){
        open (NOMBRES, "<nombres") || die "no puedo abrir nombres";
        $/ = "\n";
        @nombres = <NOMBRES>;
        undef $/;
        $nombres = @nombres;
    }
}

```

```

        for($x = 0; $x < $nombres; $x++){
            $nombre = @nombres[$x];
            $nombre = ~ s/\n*//g;
            $nombre = ~
s/[abdegiklmnoprstuMRS](?!\/?)/$&\s*\n*\]/g;
            if ($texto = ~/$nombre/) {
                $texto = ~
s/$nombre/<U>$&<\U>/g;
            }
        }
    close (NOMBRES);
}

$nombre_archivo = @lista_ficheros[$indice]; #(nuevo)
if ($nombre_archivo = ~ /A/) { #(nuevo)
    $nombre_archivo = ~ s/(A\.)0+/$1/g; #(nuevo)
    $nombre_archivo = ~ s/00(\d)/0$1/g; #(nuevo)
    $nombre_archivo = ~ s/0([1-9]\d)/$1/g; #(nuevo)
} else { #(nuevo)
    $nombre_archivo = ~ s/(\.)0+([\^\.])/1$2/g; #(nuevo)
} #(nuevo)
$nombre_archivo = ~ s/\.$//; #(nuevo)
if ($input{'alf'}){ #(nuevo)
    $texto = ~ s/S/&#347;/g; #(nuevo)
    $texto = ~ s/R/&#341;/g; #(nuevo)
}
print RESULTADO ("<hr>"); #(nuevo)
print RESULTADO ("<h3>$nombre_archivo</h3>"); #(modificado)
print RESULTADO ("$textinf");
print RESULTADO ("$texto");
print RESULTADO ("\n");
}
close (TEXTOS);
if (open (VAR, "<./var/@lista_ficheros[$indice]")){ #(nuevo)
    $var = <VAR>; #(nuevo)
    if($input{'var'}){ #(nuevo)
        if($var = ~ /$expresion1/g){ #(nuevo)
            $var = ~ s/$expresion1/<FONT color=red> #(nuevo)
<STRONG>$&<\STRONG><\FONT>/g;
            $i++; #(nuevo)
        }
        if($var = ~ /$expresion2/g && $expresion2 = ~ /.+){ #(nuevo)

```

```

                $var =~ s/$expresion2/<FONT color=red>      #(nuevo)
<STRONG>$&</STRONG></FONT>/gs;
                $e++;                                     #(nuevo)
            }
        }
        print RESULTADO (" $var\n\n");                    #(nuevo)
    }
}
}
close (VAR);                                           #(nuevo)

close (RESULTADO);

open (RESULTADO, "<resultado-iberico") || die "no puedo abrir resultado";

@resultado = <RESULTADO>;

    print &PrintHeader;
    print <<ENDOFTEXT;
    <html>
    <head>
    <title>Resultado de la b&uacute;squeda:</title>
    </head>
    <body text="white" bgcolor="black">
    <h1>Resultados de la b&uacute;squeda</h1>
    <h2>Expresi&oacute;n: $input{'expresion1'},
$input{'expresion2'}</h2> $expresion1

    <h2>Total: $i, $e </h2>
    <pre>@resultado</pre>
    <h2>Total expresi&uacute;n 1: $i</h2>

    <h2>Total expresi&uacute;n 2: $e</h2>
ENDOFTEXT
    print &HtmlBot;
}

```

Bibliografía

- ANDERSON J.M. (1993) «Iberian and Basque linguistic similarities». En *Lengua y cultura en la Hispania prerromana. Actas del V Coloquio sobre lenguas y culturas prerromanas de la península ibérica*, págs. 487–498. Universidad de Salamanca, Salamanca.
- ANGOT P. (1989) «Pour une lecture affinée de l'alphabet ibère». *Archivo de Prehistoria Levantina*, XIX, 117–129.
- AQUILUÉ X. y VELAZA J. (1993) «Un falso epígrafe ibérico en el MNAT (Museo Nacional Arqueològic de Tarragona)». *Faventia*, 15(2), 7–21.
- AQUILUÉ X. y VELAZA J. (2001) «Nueva inscripción ibérica ampuritana». *Palaeohispanica*, 1, 277–289.
- ARANEGUI C.; MATA C. y BALLESTER J.P. (1997) *Damas y caballeros en la ciudad ibérica*. Cátedra, Madrid.
- ARASA F. (1989) «Una estela ibérica de Bell-Lloc (La Plana Alta)». *Archivo de Prehistoria Levantina*, XIX, 91–101.
- ARASA F. y IZQUIERDO I. (1990) «La imagen de la memoria. Antecedentes, tipología e iconografía de las estelas de época ibérica». *Archivo de Prehistoria Levantina*, XIII, 259–300.
- ARASA F. y IZQUIERDO I. (1998) «Estela antropomorfa con inscripción del Mas de Barberán (Nogueruelas, Teruel)». *Archivo Español de Arqueología*, 71, 79–102.
- BALLESTER X. (2001a) «Fono(tipo)logía de las (con)sonantes (celt)ibéricas». En *Religión, lengua y cultura prerromanas de Hispania. Actas del VIII Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, F. Villar y M.P. Fernández Álvarez, eds., págs. 287–303. Ediciones Universidad de Salamanca, Salamanca.

- BALLESTER X. (2001b) «La *adfinitas* de las lenguas aquitana e ibérica». *Palaeohispanica*, 1, 21–33.
- BELTRÁN A. (1996) «Las inscripciones de las monedas “íberas”». En *Estudios de lenguas y epigrafía antiguas, E.L.E.A.*, págs. 157–187. Ayuntamiento de Valencia, Valencia.
- BELTRÁN F.; DE HOZ J. y UNTERMANN J. (1996) *El tercer Bronce de Botorrita (Contrebia Belaisca)*. Nº 19 en Colección Arqueología. Gobierno de Aragón, Zaragoza.
- BELTRÁN F. y VELAZA J. (1993) «Una nueva inscripción ibérica sobre bronce (Aranguren, Navarra)». En *Studia palaeohispanica et indogermanica J. Untermann ab amici*, I. Adiego; J. Siles y J. Velaza, eds., págs. 89–99. Publicacions Universitat de Barcelona, Barcelona.
- BELTRÁN LLORIS F. (1993) «La epigrafía como índice de aculturación en el valle medio del Ebro». *Lengua y cultura en la Hispania prerromana. Actas del V Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, págs. 235–272.
- BENAGES J. (1990) «Escriptura ibèrica sobre plom». *Butlletí Arqueològic de la Reial Societat Arqueològica Tarraconense de Tarragona*, 12, 41–47.
- BONET H. y MATA C. (1989) «Nuevos grafitos e inscripciones ibéricas valencianos». *Archivo de Prehistoria Levantina*, XIX, 131–148.
- CAMPMAJO P. y RENDU C. (1995) «Les écritures ibériques comme élément de datation des gravures rupestres de Cerdagne». En *X Col.loqui Internacional d'Arqueologia de Puigcerdà*, págs. 479–489. Puigcerdà.
- CAMPMAJO P. y UNTERMANN J. (1986) «Les gravures rupestres schématiques linéaires de la Cerdagne Française». En *VI Col.loqui Internacional d'Arqueologia de Puigcerdá*, págs. 317–336. Puigcerdà.
- CAMPMAJO P. y UNTERMANN J. (1991) «Corpus des gravures ibériques de Cerdagne». *Ceretania*, 1, 39–59.
- CAMPMAJO P. y UNTERMANN J. (1993) «Les influences ibériques dans la Haute Montagne Catalane: le cas de la Cerdagne». En *Lengua y cultura en la Hispania prerromana. Actas del V Coloquio sobre lenguas y culturas prerromanas de la Península Ibérica*, págs. 499–519.

- CARO BAROJA J. (1942-1943) «Observaciones sobre la hipótesis del vascoiberismo considerada desde el punto de vista histórico». *Emerita*, 10 y 11, 236–86 y 1–59.
- CARO BAROJA J. (1946) «Sobre el vocabulario de las inscripciones ibéricas». *Boletín de la Real Academia Española*, 25, 173–219.
- CARO BAROJA J. (1949) «Cuestiones ibéricas. A propósito de la estela de Sincarca». En *Homenaje a D. Julio de Urquijo*, págs. 111–118. San Sebastián.
- CARO BAROJA J. (1988a) «La geografía lingüística de la España antigua a la luz de la lectura de las inscripciones monetales». En *Sobre la lengua vasca y el vascoiberismo*, págs. 121–169. Txertoa, San Sebastián.
- CARO BAROJA J. (1988b) «Observaciones sobre la hipótesis del vascoiberismo considerada desde el punto de vista histórico». En *Sobre la lengua vasca y el vasco-iberismo*, págs. 1–120. Txertoa, San Sebastián.
- CARO BAROJA J. (1988c) «Sobre el vocabulario de las inscripciones ibéricas». En *Sobre la lengua vasca y el vasco-iberismo*, págs. 171–217. Txertoa, San Sebastián.
- COMAS M.; PADRÓS P. y VELAZA J. (2001) «Dos nuevas estelas ibéricas de Badalona». *Palaeohispanica*, 1, 291–299.
- COMAS M.; PADRÓS P. y VELAZA J. (2002) «Un plomo con doble inscripción ibérica localizado en Baetulo (Badalona)». *Palaeohispanica*, 2, 327–331.
- CORREA J.A. (1989) «Inscripción vascular indígena hallada en Baeza (Jaén)». *Archivo de Prehistoria Levantina*, XIX, 183–189.
- CORREA J.A. (1993) «Antropónimos galos y ligures en inscripciones ibéricas». En *Studia palaeohispanica et indogermanica J. Untermann ab amicis hispanicis oblata*, I. Adiego; J. Siles y J. Velaza, eds., págs. 101–116. Publicacions Universitat de Barcelona, Barcelona.
- CORREA J.A. (1994a) «La lengua ibérica». *Revista Española de Lingüística*, 24, 263–287.
- CORREA J.A. (1994b) «La transcripción de las vibrantes de la escritura paleohispánica». *Archivo de Prehistoria Levantina*, XXI, 337–341.

- CORREA J.A. (1999) «Las nasales en ibérico». En *Pueblos, lenguas y escrituras en la Hispania prerromana. Actas del VII Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, F. Villar y F. Beltrán, eds., págs. 375–396. Ediciones Universidad de Salamanca, Salamanca.
- CORREA J.A. (2001) «Las silbantes en ibérico». En *Religión, lengua y cultura prerromanas de Hispania. Actas del VIII Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, F. Villar y M.P. Fernández Álvarez, eds., págs. 305–318. Ediciones Universidad de Salamanca, Salamanca.
- CORTÉS COPETE J.M., ed. (1999) *Epigrafía griega*. Cátedra, Madrid.
- CURA M. (1986) «Els grafits ibèrics d'Il·liberis (Elna, Rosselló)». En *VI Col·loqui Internacional d'Arqueologia de Puigcerdà*, págs. 203–209. Puigcerdà.
- CURA M. (1993) «Nous grafits ibèrics en el Molí d'Espígol (Tornabous) i la cronologia de l'escriptura ibèrica a l'interior de Catalunya». *Gala*, 2, 219–225.
- DE HOZ J. (1981a) «Algunas precisiones sobre textos metrológicos ibéricos». *Archivo de Prehistoria Levantina*, XVI, 475–486.
- DE HOZ J. (1981b) «El euskera y las lenguas vecinas antes de la romanización». En *Euskal linguistika eta literatura. Bide berriak*, págs. 27–56. Bilbao.
- DE HOZ J. (1983) «Las lenguas y la epigrafía prerromanas de la Península Ibérica». En *Unidad y pluralidad en el mundo antiguo: Actas del VI Congreso español de estudios clásicos(1981)*, págs. 351–396. Madrid.
- DE HOZ J. (1985) «El nuevo plomo inscrito de Castell y el problema de las oposiciones de sonoridad en ibérico». En *Symbolae Ludovico Mitxelena septuagenario oblatae*, págs. 443–453. Vitoria.
- DE HOZ J. (1992) «Una falcata decorada con inscripción ibérica». En *Estudios de Arqueología ibérica y romana. Homenaje a Enrique Pla Ballester*, tomo II, págs. 330–338. Valencia.
- DE HOZ J. (1993) «La lengua y la escritura ibéricas, y las lenguas de los iberos». En *Lengua y cultura en la Hispania prerromana. Actas del V Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, págs. 635–666. Colonia.

- DE HOZ J. (1995a) «El poblamiento antiguo de los Pirineos desde el punto de vista lingüístico». En *Muntanyes i Població*, J. Bertranpetit y E. Vives, eds., págs. 271–299. Centre de trobada de les cultures pirinenques, Andorra.
- DE HOZ J. (1995b) «Escrituras en contacto: ibérica y latina». En *Roma y el nacimiento de la cultura epigráfica en Occidente*, págs. 57–84.
- DE HOZ J. (1995c) «Notas sobre nuevas y viejas leyendas monetales». En *La moneda hispánica. Ciudad y territorio*, M.P. García-Bellido y R.M. Sobral Centeno, eds., Anejos de Archivo Español de Arqueología, 14, págs. 317–324. Madrid.
- DE HOZ J. (1995d) «Panorama provisional de la epigrafía rupestre paleohispánica». En *Saxa scripta (Inscripciones en roca). Actas del Simposio internacional Ibero-italico sobre epigrafía rupestre*, A. Rodríguez Colmenero y L. Gasperini, eds., págs. 9–33. A Coruña.
- DE HOZ J. (1998) «La epigrafía ibérica de los noventa». *Revista de Estudios Ibéricos*, 3, 127–151.
- DE HOZ J. (2001a) «Hacia una tipología del ibérico». En *Religión, lengua y cultura prerromanas de Hispania. Actas del VIII Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, F. Villar y M.P. Fernández Álvarez, eds., págs. 335–362. Ediciones Universidad de Salamanca, Salamanca.
- DE HOZ J. (2001b) «Sobre algunos problemas del estudio de las lenguas paleohispánicas». *Palaeohispanica*, 1, 113–149.
- DE HOZ J. (2002) «El complejo sufijal (*e*)skén de la lengua ibérica». *Palaeohispanica*, 2, 159–168.
- DE HOZ J. y LUJÁN E. (2001) «Bibliografía de inscripciones ibéricas no recogidas en MLH». *Palaeohispanica*, 1, 355–367.
- DELANUY A. (1993) «L'inscription ibère de Sinarcas». *La parola del passato. Rivista studi antichi*, CCLXX, 186–189.
- DOMÍNGUEZ MONEDERO A. (1994) «De nuevo sobre la estela funeraria de Ampurias». *V Congreso internacional de estelas funerarias*, págs. 55–62.
- ESTEVE F. (1989) «La lápida ibérica de Cabanes». *Archivo de Prehistoria Levantina*, XIX, 103–115.

- FATÁS G. (1993a) «Los Pirineos Meridionales y la conquista romana». En *Lengua y cultura en la Hispania prerromana. Actas del V Coloquio sobre lenguas y culturas prerromanas de la Península Ibérica*, págs. 289–315.
- FATÁS G. (1993b) «Para una etnogeografía de la cuenca media del Ebro». *Complutum*, 2-3, 223–232.
- FLETCHER D. (1981) «Villares VII (Plomo escrito de Caudete de las Fuentes)(Valencia)». *Archivo de Prehistoria Levantina*, XVI, 463–474.
- FLETCHER D. (1982a) *El plomo ibérico de Mogente*. Nº 76 en Serie de Trabajos Varios. Diputación Provincial de Valencia, Valencia.
- FLETCHER D. (1982b) «Nuevos plomos ibéricos valencianos». *Arse*, 17, 252–256.
- FLETCHER D. (1985a) «Lengua y epigrafía ibéricas». En *Arqueología del País Valenciano: panorama y perspectiva*, Anejo de la Revista Lucentum. Universidad de Alicante. Disponible en <http://publicaciones.ua.es/LibroPDF/84-600-3906-4/14-ArqueologiaPV.pdf>.
- FLETCHER D. (1985b) *Textos ibéricos del Museo de Prehistoria de Valencia*. Nº 81 en Serie de Trabajos Varios. Diputación Provincial de Valencia, Valencia.
- FLETCHER D. (1987) «Repertorio de inscripciones ibéricas procedentes de Sagunto (Valencia)». *Arse*, 22, 659–669.
- FLETCHER D. (1992) «Comentarios sobre escritura y lengua ibéricas». En *Estudios de Arqueología ibérica y romana. Homenaje a Enrique Pla Ballster*, págs. 301–311. Valencia.
- FLETCHER D. (2000) «Comentarios sobre los grafemas silábicos ibéricos». En *Estudios de lenguas y epigrafía antiguas - E.L.E.A. nº 3*, págs. 13–25. Ayuntamiento de Valencia, Valencia.
- FLETCHER D. y GINER V. (1974) «Tres lápidas ibéricas de Canet lo Roig (Castellón)». *Boletín de la Sociedad Castellonense de Cultura*, L, 138–156.
- FLETCHER D. y GISBERT J.A. (1994) «Hallazgo de una inscripción ibérica en el Camí del Molí (Terrateig, La Vall d'Albaida)». *Archivo de Prehistoria Levantina*, XXI, 343–353.

- FLETCHER D. y MESSEGUER V. (1981) «Inscripción ibérica de San Mateo (Castellón de la Plana)». *Boletín de la Sociedad Castellonense de Cultura*, 57, 203–209.
- FLETCHER D. y MESSEGUER FOCH V. (1981) «Inscripción ibérica de San Mateo (Castellón de la Plana)». *Boletín de la Sociedad Castellonense de Cultura*, 57, 203–209.
- FLETCHER D. y SILGO L. (1991) «Plomo ibérico, en escritura jonia, procedente de Sagunto». *Arse*, 26, 1–6.
- FLETCHER D. y SILGO L. (1992) «Plomo ibérico procedente de Gandía (Valencia)». *Arse*, 27, 3–5.
- FRIEDL J.E. (2002) *Mastering regular expressions*. O'Reilly.
- GARCÍA-BELLIDO M.P. (1998) «Sobre la moneda de los iberos». *Revista de estudios ibéricos*, 3, 109–126.
- GARCÍA-BELLIDO M.P. (2001) «Plomos monetiformes con el topónimo ibérico Gador». *Palaeohispanica*, 1, 335–340.
- GÓMEZ R. y SANZ K. (1995) «On the origin of the finite forms of the Basque verb». En *Towards a history of the Basque language*, J.I. Hualde; J. Lakarra y R. Trask, eds., págs. 235–274. John Benjamins Publishing Company, Amsterdam-Philadelphia.
- GÓMEZ MORENO M. (1949a) *Misceláneas*, cap. La escritura ibérica y su lenguaje, págs. 257–281. Madrid.
- GÓMEZ MORENO M. (1949b) *Misceláneas*, cap. De epigrafía ibérica: el plomo de Alcoy, págs. 219–231. Madrid.
- GÓMEZ MORENO M. (1949c) *Misceláneas*, cap. Sobre los iberos: el bronce de Ascoli, págs. 233–256. Madrid.
- GORROCHATEGUI J. (1984) *Estudio sobre la onomástica indígena de Aquitania*. Servicio Editorial Universidad del País Vasco, Bilbao.
- GORROCHATEGUI J. (1987) «Situación lingüística de Navarra y alrededores en la angigüedad a partir de fuentes epigráficas». En *I Congreso General de Historia de Navarra. Vol.II: Comunicaciones*, págs. 435–445. Príncipe de Viana, Pamplona.

- GORROCHATEGUI J. (1993a) «La onomástica aquitana y su relación con la ibérica». En *Lengua y cultura en la Hispania prerromana. Actas del V Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, págs. 609–633. Colonia.
- GORROCHATEGUI J. (1993b) «Las lenguas de los pueblos paleohispánicos». En *Los celtas: Hispania y Europa*, M. Almagro-Gorbea y G.R. Zapatero, eds., págs. 409–429. Madrid.
- GORROCHATEGUI J. (1995a) «Algunos aspectos de la onomástica personal antigua de los Pirineos». En *Muntanyes i Població*, J. Bertranpetit y E. Vives, eds., págs. 151–156. Centre de trobada de les cultures pirinenques, Andorra.
- GORROCHATEGUI J. (1995b) «The Basque language and its neighbors in antiquity». En *Towards a history of the Basque language*, J.I. Hualde; J. Lakarra y R. Trask, eds., págs. 31–63. John Benjamins Publishing Company, Amsterdam-Philadelphia.
- GORROCHATEGUI J. (1995c) «Los Pirineos entre Galia e Hispania: Las lenguas». *Veleia*, 12, 181–234.
- GORROCHATEGUI J. y LAKARRA J. (1996) «Nuevas aportaciones a la reconstrucción del Protovasco». En *La Hispania Prerromana. Actas del VI Coloquio sobre lenguas y culturas prerromanas de la Península Ibérica*, págs. 102–145. Universidad de Salamanca, Salamanca.
- GORROCHATEGUI J. y LAKARRA J. (2001) «Comparación lingüística, filología y reconstrucción del protovasco». En *Religión y cultura prerromanas de Hispania. Actas del VIII Coloquio sobre lenguas y culturas prerromanas de la Península Ibérica*, págs. 407–438. Universidad de Salamanca, Salamanca.
- GUERIN P. y SILGO L. (1996) «Inscripción ibérica sobre plomo de Castellet de Bernabé». *Revista d'Arqueologia de Ponent*, 16, 199–206.
- GUERRA A. (2000) «Dos estelas funerarias ibéricas en Badalona». *Revista de Arqueología*, 234, 63.
- GUITART J.; PERA J.; MAYER M. y VELAZA J. (1996) «Noticia preliminar sobre una inscripción ibérica encontrada en Guissona (Lleida)». En *La Hispania prerromana. Actas del VI Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, F. Villar y J. D'Encarnaçao, eds., págs. 163–170. Ediciones Universidad de Salamanca, Salamanca.

- GUITER H. (1984) «Sobre algunas inscripciones ibéricas». *Archivo de Prehistoria Levantina*, XVII, 319–330.
- GUITER H. (1992) «Ibère et minoen». En *Estudios de arqueología ibérica y romana. Homenaje a Enrique Pla Ballester*, págs. 313–317. Valencia.
- HUMBOLDT W.V. (1821) *Prüfung der Untersuchungen über die Urbewohner Hispaniens vermittelt der Vaskischen Sprache*. Berlin.
- LAKARRA J. (1995) «Reconstructing the Pre-Proto-Basque root». En *Towards a history of the Basque language*, J.I. Hualde; J. Lakarra y R. Trask, eds., págs. 185–205. John Benjamin Publishing Company, Amsterdam-Philadelphia.
- MARCO SIMÓN F. (1976) «El monumento ibérico de Binéfar (Huesca)». *Pyrenae*, 12, 91–116.
- MARCO SIMÓN F. (1998) «Entre el estereotipo y la realidad histórica: la emergencia de los pueblos pirenaicos antiguos.» En *Los pueblos prerromanos del norte de Hispania.*, págs. 51–87. EUNSA, Pamplona.
- MARQUES DE FARIA A. (1998a) «Quintanilla, Alberto-Estudios de fonología ibérica (Recensión)». *Revista portuguesa de Arqueologia*, 1(2), 232–240.
- MARQUES DE FARIA A. (1998b) «Rico, Christian-Pyrénées romaines: Essai sur un pays de frontière (IIIe siècle av. J.-C.-III siècle ap. J.-C.) (Recensión)». *Revista portuguesa de arqueología*, 1(2), 260–263.
- MARQUES DE FARIA A. (1998c) «Silgo Gauche, L. (1994), Léxico ibérico (Recensión)». *Revista portuguesa de Arqueologia*, 1(1), 228–234.
- MARQUES DE FARIA A. (1999) «Novas notas de onomástica hispánica pré-romana». *Revista portuguesa de Arqueologia*, 2(1), 153–161.
- MARQUES DE FARIA A. (2000a) «Crónica de onomástica paleo-hispánica (1)». *Revista portuguesa de Arqueologia*, 3(2), 61–66.
- MARQUES DE FARIA A. (2000b) «Onomástica paleo-hispánica: revisão de algumas leituras e interpretações». *Revista portuguesa de Arqueologia*, 3(1), 121–151.
- MARQUES DE FARIA A. (2001) «Crónica de onomástica paleo-hispánica (2)». *Revista portuguesa de Arqueologia*, 4(1), 95–107.

- MARQUES DE FARIA A. (2002a) «Crónica de onomástica paleo-hispánica (3)». *Revista Portuguesa de Arqueología*, 5(1), 121–146.
- MARQUES DE FARIA A. (2002b) «Crónica de onomástica paleo-hispánica (4)». *Revista Portuguesa de Arqueología*, 5(2), 233–244.
- MASSÓ J. y VELAZA J. (1995) «Sobre una inscripció ibèrica suspecta: MNAT 664». *Pyrenae*, 26, 115–121.
- MATA C. y SORIA L. (1997) «Marcas y epígrafes sobre contenedores de época ibérica». *Archivo de Prehistoria Levantina*, XXII, 297–305.
- MATEU F. (1992) «Toponimia ibérica en la *Nomina civitatum Ispanie sedes episcopaliūm*. Comentario». En *Estudios de arqueología ibérica y romana. Homenaje a Enrique Pla Ballester*, págs. 361–363. Valencia.
- MAYER M. (1993) «Lat. Felix vs. ib. leota». En *Studia palaeohispanica et indogermanica J. Untermann ab amici*, I. Adiego; J. Siles y J. Velaza, eds., págs. 179–181. Publicacions Universitat de Barcelona, Barcelona.
- MAYER M. (2002) «El proceso de adopción de la fórmula onomástica romana». *Palaeohispanica*, 2, 189–200.
- MAYER M. y VELAZA J. (1993) «Epigrafía ibérica sobre soportes típicamente romanos». *Lengua y cultura en la Hispania prerromana. Actas del V Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, págs. 667–682.
- MEZQUÍRIZ IRUJO M.A. (1992) «Inscripción ibérica en Andelos (Mendi-gorría, Navarra)». En *Estudios de Arqueología ibérica y romana. Homenaje a Enrique Pla Ballester*, págs. 347–349. Valencia.
- MICHELENA L. (1961) «Comentarios en torno a la lengua ibérica». *Zephyrus*, 12, 5–23.
- MICHELENA L. (1977) *Fonética histórica vasca*. Publicaciones del Seminario Julio de Urquijo, San Sebastián.
- MICHELENA L. (1985a) «Cuestiones relacionadas con la escritura ibérica». En *Lengua e historia*, L. Michelena, ed., págs. 357–370. Paraninfo, Madrid.
- MICHELENA L. (1985b) «Ibérico -en». En *Lengua e Historia*, L. Michelena, ed., págs. 379–387. Paraninfo, Madrid.

- MICHELENA L. (1985c) «La langue ibère». En *Lengua e Historia*, L. Michelena, ed., págs. 341–356. Paraninfo, Madrid.
- MICHELENA L. (1985d) «Lenguas indígenas y lengua clásica en Hispania». En *Lengua e Historia*, L. Michelena, ed., págs. 202–212. Paraninfo, Madrid.
- MICHELENA L. (1985e) «Onomástica aquitana». En *Lengua e Historia*, L. Michelena, ed., págs. 409–445. Paraninfo, Madrid.
- MICHELENA L. (1985f) «Sobre la posición lingüística del ibérico». En *Lengua e Historia*, L. Michelena, ed., págs. 334–340. Paraninfo, Madrid.
- OLIVER FOIX A. (1995) «Aproximación a la problemática de las estelas epigráficas funerarias ibéricas no decoradas». *Actas del V Congreso internacional de estelas funerarias*, págs. 107–116.
- OLIVER FOIX A.; CASABÓ J.A. y ORTEGA J.L. (1982-1983) «Nuevas inscripciones ibéricas en la Vall d'Uixó». *Cuadernos de Prehistoria y Arqueología Castellonenses*, 9, 243–248.
- OROZ F. (1999) «Miscelánea hispánica». En *Pueblos, lenguas y escrituras en la Hispania prerromana. Actas del VII Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, F. Villar y F. Beltrán, eds., págs. 499–534. Ediciones Universidad de Salamanca, Salamanca.
- PANOSA M.I. (1992) «Catàleg i anàlisi dels epígrafs ibèrics del Vallès Oriental». *Limes*, 2, 56–75.
- PANOSA M.I. (1993a) «Algunes qüestions històriques sobre l'epigrafia de les llegendes monetals ibèriques». En *Studia palaeohispanica et indogermanica J. Untermann ab amici*, I. Adiego; J. Siles y J. Velaza, eds., págs. 199–212. Publicacions Universitat de Barcelona, Barcelona.
- PANOSA M.I. (1993b) «Complejidad, distribución y escritura en el NE peninsular. Hipótesis de trabajo». En *Lengua y cultura en la Hispania prerromana. Actas del V Coloquio sobre lenguas y culturas prerromanas de la Península Ibérica*, págs. 719–725.
- PANOSA M.I. (1993c) «Nuevas inscripciones ibéricas de Cataluña». *Complutum*, 4, 175–222.
- PANOSA M.I. (1995) «El paper de la Cerdanya com a nexa entre el territori ibèric del nord i el sud del Pirineu. Problemàtica del seu corpus d'inscripcions ibèriques». En *X Col.loqui Internacional d'Arqueologia de Puigcerdà*, págs. 465–477. Puigcerdà.

- PANOSA M.I. (1996) «Elementos sobre la fase de bilingüismo y latinización de la población ibérica». En *La Hispania Prerromana. Actas del VI Coloquio sobre lenguas y culturas prerromanas de la Península Ibérica*, págs. 217–246. Universidad de Salamanca, Salamanca.
- PANOSA M.I. (1999) *La escritura ibérica en Cataluña y su contexto socioeconómico (siglos V-I a. C.)*. Servicio Editorial Universidad del País Vasco, Vitoria.
- PANOSA M.I. (2001) «Novedades de epigrafía ibérica en Cataluña y algunos aspectos metodológicos». En *Religión, lengua y cultura prerromanas de Hispania. Actas del VIII Coloquio sobre lenguas y culturas prerromanas de la Península Ibérica*, págs. 511–540. Universidad de Salamanca, Salamanca.
- PANOSA M.I. (2002) «Inscripción ibérica procedente de La Joncosa (Jorba, Barcelona)». *Palaeohispanica*, 2, 333–353.
- PATTISON W. (1981) «Iberian and basque (a morpho-syntactic comparison)». *Archivo de Prehistoria Levantina*, XVI, 487–522.
- PÉREZ ALMOGUERA A. (2001) «*Iltif/iltur*=*oppidum*. Los nombres de lugar y la ciudad en el mundo ibérico». *Faventia*, 1(23), 21–40.
- PÉREZ OROZCO S. (1993a) «Notas sobre onomástica ibérica». *Fontes Linguae Vasconum*, 62, 61–67.
- PÉREZ OROZCO S. (1993b) «Observaciones sobre los sufijos ibéricos». *Fontes Linguae Vasconum*, 63, 221–229.
- PÉREZ ROJAS M. (1983) «La estela ibérica de Caspe: Introducción a su estudio lingüístico». *Archivo Español de Arqueología*, 56, 269–285.
- PÉREZ VILATELA L. (1992) «Ibérico “egiar” en un epígrafe de Caminreal (Teruel)». En *Estudios de Arqueología ibérica y romana. Homenaje a Enrique Pla Ballester*, págs. 351–360. Valencia.
- PINYAN J. (2003) *Regular expressions in Perl*. <http://japhy.perlmonk.org/book> (Consultado el 3 de marzo del 2003).
- PRESCOT A.E. (1980) «Estudio de la inscripción inédita de la fusayola del poblado ibérico de Castell de Palamós (Gerona) y sus relaciones con otros textos». *Cypsela*, III, 147–152.

- QUINTANILLA A. (1993a) «Sobre el vocalismo de la lengua ibérica». En *Lengua y cultura en la Hispania prerromana. Actas del V Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, págs. 727–737. Colonia.
- QUINTANILLA A. (1993b) «Sobre la notación en la escritura ibérica del modo de articulación de las consonantes oclusivas». En *Studia palaeohispanica et indogermanica J. Untermann ab amicis hispanicis oblata*, I. Adiego; J. Siles y J. Velaza, eds., págs. 239–250. Publicacions Universitat de Barcelona, Barcelona.
- QUINTANILLA A. (1998) *Estudios de fonología ibérica*. Servicio Editorial Universidad del País Vasco, Vitoria.
- QUINTANILLA A. (1999) «Las vibrantes en la lengua ibérica». En *Pueblos, lenguas y escrituras en la Hispania prerromana. Actas del VII Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, F. Villar y F. Beltrán, eds., págs. 563–569. Ediciones Universidad de Salamanca, Salamanca.
- RICO C. (1997) *Pyrénées romaines: Essai sur un pays de frontière (IIIe siècle av. J.-C.-IIIe siècle ap. J.-C.)*. Bibliothèque de la Casa de Velázquez, 14, Madrid.
- RIPOLLÉS P.P. (2001) «Una leyenda monetaria inédita de *Saitabi*». *Saguntum*, 33, 167–170.
- RIURÓ F. (1984) «El plom amb epigrafia ibèrica del poblat de Castell (Palamós)». *Cypsela*, IV, 123–131.
- ROCA F. (1974) «Una inscripción latino ibérica inédita en Sagunto». *Arse*, págs. 27–29.
- ROCA F. (1984) «Nuevos restos epigráficos del término de Sagunto». *Arse*, 19, 57–58.
- RODRÍGUEZ RAMOS J. (1994) «Liria XIIC: ¿Un *kálathos* ibérico dedicado a Proserpina?» *Faventia*, 2(16), 65–81.
- RODRÍGUEZ RAMOS J. (1995) «Nota a la inscripción ibérica Tarragona C.18.8». *Pyrenae*, 26, 123–125.
- RODRÍGUEZ RAMOS J. (1997) «Primeras observaciones para una datación paleográfica de la escritura ibérica». *Archivo español de arqueología*, 70, 13–30.

- RODRÍGUEZ RAMOS J. (1998) «Sobre la lectura y paleografía de la inscripción de la falcata saguntina MPV 314». *Pyrenae*, 29, 227–230.
- RODRÍGUEZ RAMOS J. (1999) «Introducción a la escritura ibérica. Variante levantina». *Revista de arqueología*, 218, 6–13.
- RODRÍGUEZ RAMOS J. (2000a) «La lectura de las inscripciones sudlusitano-tartésicas». *Faventia*, 1(22), 21–48.
- RODRÍGUEZ RAMOS J. (2000b) «La lengua íbera: En busca del paradigma perdido». En *Revista internacional d'humanitats*, tomo 3. <http://www.hottopos.com.br/rih3/lengiber.htm>.
- RODRÍGUEZ RAMOS J. (2000c) «Vocales y consonantes nasales en la lengua íbera». *Faventia*, 22(2), 25–37.
- RODRÍGUEZ RAMOS J. (2001a) «Aspectos de la morfología de los formantes segundos de los compuestos de tipo onomástico en la lengua íbera». *Faventia*, 1(23), 7–19.
- RODRÍGUEZ RAMOS J. (2001b) «El término (t)eban(en) en la lengua íbera: 'coeravit' vs. 'filius'». *Arse*, 35, 59–85.
- RODRÍGUEZ RAMOS J. (2001c) «La cultura ibérica desde la perspectiva de la epigrafía: un ensayo de síntesis». *Iberia*, 4, 17–38.
- RODRÍGUEZ RAMOS J. (2002a) «Acerca de los afijos adnominales de la lengua íbera». *Faventia*, 24(1), 115–134.
- RODRÍGUEZ RAMOS J. (2002b) «La hipótesis del vascoiberismo desde el punto de vista de la epigrafía íbera». *Fontes Linguae Vasconum*, 90, 197–217.
- RODRÍGUEZ SOMOLINOS H. (1996) «The commercial transaction of the Pech-Maho lead. A new interpretation». *Zeitschrift für Papyrologie und Epigraphik*, 111, 74–78.
- SANMARTÍ-GREGO E. (1988) «Una carta en lengua ibérica, escrita sobre plomo, procedente de Emporion». *Revue Archéologique de Narbonnaise*, 21, 95–113.
- SANMARTÍ-GREGO E. y SANTIAGO R.A. (1988) «La lettre grecque d'Emporion et son contexte archéologique». *Revue Archéologique de Narbonnaise*, 21, 3–17.

- SARRIÓN I. (1981) «Grafitos ibéricos del cerro de San Miguel (Llíria, Valencia)». *Archivo de Prehistoria Levantina*, XVI, 455–461.
- SILES J. (1980) «La inscripción ibérica de La Peña de las Majadas (El Toro, Castellón)». *Boletín de la Sociedad Castellonense de Cultura*, 56, 171–189.
- SILES J. (1981) «Iberismo y latinización: nombres latinos en epígrafes ibéricos». *Faentia*, 1(3), 97–113.
- SILES J. (1985) *Léxico de inscripciones ibéricas*. Nº 2 en Epigrafía hispánica. Ministerio de Cultura.
- SILES J. (1987) «Sobre la epigrafía ibérica». En *Reunión sobre la epigrafía hispánica de época romano-republicana*, págs. 17–42. Zaragoza.
- SILGO GAUCHE L. (1988) «La antroponimia ibérica de Sagunto (1)». *Arse*, 23, 67–77.
- SILGO GAUCHE L. (1989) «Tres inscripciones ibéricas». *Arse*, 24, 833–836.
- SILGO GAUCHE L. (1992) «Los límites étnicos de la Turdetania». En *Estudios de arqueología ibérica y romana. Homenaje a Enrique Pla Ballester*, págs. 365–367. Valencia.
- SILGO GAUCHE L. (1993) «Las inscripciones ibéricas de los mosaicos de Caminreal (Teruel) y Andelos (Navarra)». En *Studia paleohispanica et indogermanica J. Untermann ab amicis hispanicis oblata*, I. Adiego; J. Siles y J. Velaza, eds., págs. 281–286. Publicacions Universitat de Barcelona, Barcelona.
- SILGO GAUCHE L. (1994) *Léxico ibérico*. Ayuntamiento de Valencia, Valencia.
- SILGO GAUCHE L. (1994-1995) «Algunos datos sobre el acento ibérico». *Arse*, 28-29, 175–185.
- SILGO GAUCHE L. (1996a) «Avance a un estudio de las formas flexivas en ibérico». En *La Hispania prerromana. Actas del VI Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, F. Villar y J. D'Encarnaçao, eds., págs. 301–310. Ediciones Universidad de Salamanca, Salamanca.
- SILGO GAUCHE L. (1996b) «Epigrafía ibérica y epigrafía clásica. Algunos aspectos de su relación». En *Estudios de lenguas y epigrafía antiguas, E.L.E.A. nº 2*, págs. 55–74. Ayuntamiento de Valencia.

- SILGO GAUCHE L. (1998-1999) «Ibérico **ilti**, **iltu** y derivados». *Arse*, 32-33, 11-45.
- SILGO GAUCHE L. (2000a) «A.Quintanilla Niño: “Estudios de Fonología ibérica” (Recensión)». En *Estudios de Lenguas y epigrafía antiguas, E.L.E.A. 3*, págs. 279-293. Ayuntamiento de Valencia, Valencia.
- SILGO GAUCHE L. (2000b) «De nuevo sobre el “genitivo” ibérico en -EN». En *Estudios de lenguas y epigrafía antiguas, E.L.E.A. n° 3*, págs. 99-118. Ayuntamiento de Valencia.
- SILGO GAUCHE L. (2000c) «El problema de las silbantes ibéricas». *Habis*, 31, 503-521.
- SILGO GAUCHE L. (2000d) «La procedencia de la lápida ibérica supuesta de Liria (F.13.1)». En *Estudios de lenguas y epigrafía antiguas, E.L.E.A. n° 3*, págs. 181-186. Ayuntamiento de Valencia.
- SILGO GAUCHE L. (2001) «Grafitos ibéricos de El Palomar (Oliete, Teruel)». *Palaeohispanica*, 1, 347-352.
- SILGO GAUCHE L. y FLETCHER D. (1987) «Dos nuevos textos ibéricos saguntinos». *Arse*, 22, 671-676.
- SILGO GAUCHE L. y GOZALBES M. (1996-1997) «Nuevo plomo ibérico de Sagunto (Sagunto 58)». *Arse*, 30-31, 81-90.
- SILGO GAUCHE L. y SANJOSÉ M. (1996-1997) «Nuevas reflexiones sobre la estampilla de Azaila (E.1.287)». *Arse*, 30-31, 91-96.
- SILGO GAUCHE L. y TOLOSA LEAL A. (2000) «Plomo ibérico escrito del Camp de Morvedre». *Arse*, 34, 39-44.
- SOLIER Y. (1979) «Découverte d’inscriptions sur plombs en écriture ibérique dans un entrepot de Pech Maho (Sigean)». *Revue Archéologique de Narbonnaise*, 12, 35-123.
- SOLIER Y. y BARBOUTEAU H. (1988) «Découverte de nouveaux plombs, inscrits en ibère, dans la région de Narbonne». *Revue Archéologique de Narbonnaise*, 21, 61-94.
- TOLOSA LEAL A. (2000) «Sobre formas verbales ibéricas en -IN». En *Estudios de lenguas y epigrafía antiguas - E.L.E.A. n° 3*, págs. 143-147. Ayuntamiento de Valencia, Valencia.

- TOMÁS FERRE I. (1989) «Disco de plomo, escrito, del Pico de los Ajos (Yátova, Valencia)». *Archivo de Prehistoria Levantina*, XIX, 167–170.
- TOVAR A. (????) *Iberische Landeskunde*.
- TOVAR A. (1946) «Las inscripciones ibéricas y la lengua de los celtíberos». *Boletín de la Real Academia Española*, 25, 8–42.
- TOVAR A. (1961) *The ancient languages of Spain and Portugal*. S. F. Vanni, New York.
- TOVAR A. (1984) «Estudio actual de los estudios ibéricos». *Archivo de Prehistoria Levantina*, XVII, 29–48.
- TRASK R. (1995a) «On the origin of the non-finite verb forms in Basque». En *Towards a history of the Basque language*, J.I. Hualde; J. Lakarra y R. Trask, eds., págs. 207–234. John Benjamins Publishing Company, Amsterdam-Philadelphia.
- TRASK R. (1995b) «Origins and relatives of the Basque language: review of the evidence». En *Towards a history of the Basque language*, J.I. Hualde; J. Lakarra y R. Trask, eds., págs. 65–99. John Benjamins Publishing Company, Amsterdam-Philadelphia.
- UNTERMANN J. (1963) «Estudio sobre las áreas lingüísticas pre-romanas de la península ibérica». *Archivo de Prehistoria Levantina*, X, 165–192.
- UNTERMANN J. (1969) «Lengua gala y lengua ibérica en la Galia Narbonensis». *Archivo de Prehistoria Levantina*, 12, 99–116.
- UNTERMANN J. (1975) *Monumenta Linguarum Hispanicarum I: Die Münzlegenden*. Dr. Ludwig Reichert Verlag, Wiesbaden.
- UNTERMANN J. (1976) «Las leyendas monetales». En *Actas del I Coloquio sobre lenguas y culturas prerromanas de la Península Ibérica*, págs. 213–225. Universidad de Salamanca, Salamanca.
- UNTERMANN J. (1980) *Monumenta Linguarum Hispanicarum. Band II. Die Inschriften in iberischen Schrift aus Südfrankreich*. Dr. Ludwig Reichert Verlag, Wiesbaden.
- UNTERMANN J. (1984) «Inscripciones sepulcrales ibéricas». *Cuadernos de Prehistoria y Arqueología Castellonenses*, 10, 111–119.

- UNTERMANN J. (1987a) «Gramática de los plomos ibéricos». En *Studia Palaeohispanica. Actas del IV Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, págs. 35–56. Vitoria.
- UNTERMANN J. (1987b) «Repertorio antropónimo ibérico». *Archivo de Prehistoria Levantina*, XVII, 289–318.
- UNTERMANN J. (1989) «Nova inscripció ibérica sobre plom, procedent del país dels Ilergetes». *Acta Numismatica*, 19, 39–44.
- UNTERMANN J. (1990) *Monumenta Linguarum Hispanicarum. Band III. Die iberischen Inschriften aus Spanien*. Dr. Ludwig Reichert Verlag, Wiesbaden.
- UNTERMANN J. (1991) «Intercanvi epistolar en un plom ibèric?» *Acta numismatica (Homenatge al Dr. Leandre Villaronga)*, 21-23, 93–100.
- UNTERMANN J. (1993) «Los entónimos de la Hispania antigua y las lenguas prerromanas de la Península Ibérica». *Complutum*, 2-3, 19–33.
- UNTERMANN J. (1995) «La lengua ibérica: Nuestro conocimiento y tareas futuras». *Veleia*, 12, 243–256.
- UNTERMANN J. (1996) «Los plomos ibéricos: Estado actual de su interpretación». En *Estudios de lenguas y epigrafía antiguas- E.L.E.A. n° 2*, págs. 75–108. Ayuntamiento de Valencia, Valencia.
- UNTERMANN J. (1998a) «comentario sobre una lámina de plomo con inscripción ibérica de la colección D. Ricardo Marsal, Madrid». *Habis*, 29, 7–23.
- UNTERMANN J. (1998b) «La onomástica ibérica». *Iberia*, 1, 73–85.
- UNTERMANN J. (1999) «Über den Umgang mit iberischen Bilinguen». *Grippe, Kamm und Eulenspiegel. Festschrift für Elmar Seebold*, págs. 349–357.
- UNTERMANN J. (2000) «Zum Nachweis von Substratsprachen auf der iberischen Halbinsel». *Madrid der Mitteilungen*, ?, 139–147.
- UNTERMANN J. (2001) «Algunas novedades sobre la lengua de los plomos ibéricos». En *Religión, lengua y cultura prerromanas de Hispania. Actas del VIII Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, F. Villar y M.P. Fernández Álvarez, eds., págs. 613–627. Ediciones Universidad de Salamanca, Salamanca.

- UNTERMANN J. (2002) «Dos nuevos textos ibéricos del sur de Francia». *Palaeohispanica*, 2, 355–361.
- VALERI V. (1993) «Las nasales ibéricas». En *Lengua y cultura en la Hispania prerromana. Actas del V Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, págs. 739–745. Colonia.
- VALLADOLID MOYA J. (1998) «La estela inscrita ibérica conocida como “Lápida de Liria”: Una nueva lectura». *Veleia*, 15, 241–256.
- VALLADOLID MOYA J. (2002) «Revisión de seis inscripciones sobre cerámica procedentes de Azaila (Teruel)». *Palaeohispanica*, 2, 363–378.
- VELAZA J. (1991a) «Consideraciones en torno a la inscripción ibérica de Caminreal». *Aion*, 13, 291–295.
- VELAZA J. (1991b) *Léxico de inscripciones ibéricas (1976-1989)*. Publicacions Universitat de Barcelona, Barcelona.
- VELAZA J. (1992) «Sobre el esgrafiado ibérico de Barchín del Hoyo». En *Estudios de arqueología ibérica y romana. Homenaje a Enrique Pla Ballester*, págs. 345–346. Valencia.
- VELAZA J. (1993) «Una nueva lápida ibérica procedente de Civit (Tarragona)». *Pyrenae*, 24, 159–165.
- VELAZA J. (1994) «Iberisch eban, teban». *ZPE*, 104, 142–150.
- VELAZA J. (1995) «Epigrafía y dominios lingüísticos en territorio de los vascones». En *Roma y el nacimiento de la cultura epigráfica en Occidente*, F. Beltrán Lloris, ed., págs. 209–218. Institución “ Fernando el Católico”, Zaragoza.
- VELAZA J. (1996a) «Chronica epigraphica iberica: Hallazgos de inscripciones ibéricas en Levante, Cataluña, Aragón y Navarra (1989-1994)». *Actas del VI Coloquio sobre lenguas y culturas prerromanas de la Península Ibérica*, págs. 311–337.
- VELAZA J. (1996b) «Epigrafía funeraria ibérica». En *Estudios de lenguas y epigrafía antiguas, E.L.E.A. n.º 2*, págs. 251–282. Ayuntamiento de Valencia.
- VELAZA J. (1996c) *Epigrafía y lengua ibéricas*. Arco Libros, S.L., Madrid.

- VELAZA J. (1998) «La epigrafía monetaria paleohispánica: breve estado de la cuestión.» En *II Curs d'història monetària d'Hispania. La Moneda en la societat ibèrica.*, págs. 67–79. Museu Nacional d'Art de Catalunya, Barcelona.
- VELAZA J. (2000) «Nueva hipótesis sobre la inscripción ibérica del teatro de Sagunto». *Saguntum*, 32, 131–134.
- VELAZA J. (2001a) «Chronica epigraphica Iberica II: Novedades y revisiones de epigrafía ibérica». En *Religión, lengua y cultura prerromanas de Hispania. Actas del VIII Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, F. Villar y M.P. Fernández Álvarez, eds., págs. 639–662. Ediciones Universidad de Salamanca, Salamanca.
- VELAZA J. (2001b) «Chronica Epigraphica Iberica III (2000)». *Palaeohispanica*, 1, 393–395.
- VELAZA J. (2002a) «Chronica Epigraphica Iberica IV (2001)». *Palaeohispanica*, 2, 411–414.
- VELAZA J. (2002b) «Ibérico -te». *Palaeohispanica*, 2, 271–275.
- VELÁZQUEZ I. (1991) «Notas sobre la descripción de tierras pirenaicas en fuentes literarias de época tardoantigua y visigoda (aspectos literarios fundamentalmente)». En *Actas del Congreso internacional “Historia de los Pirineos”*, E. Ripoll Perelló y M. Ladero, eds., págs. 477–488. Madrid.
- VICENTE J.; PUNTER M.P.; ESCRICHE C. y HERCE A.I. (1993) «Las inscripciones de la “Casa de LIKINE” (Caminreal, Teruel)». En *Lengua y cultura en la Hispania prerromana. Actas del V Coloquio sobre lenguas y culturas prerromanas de la Península ibérica*, págs. 749–772. Colonia.
- VILÀ M.D.V. (1996) «Àmfora amb inscripció llatina i grafit ibèric». *Pyrenae*, 27, 295–299.
- VILATELA L.P. (1991) «Plomo ibérico, en escritura jónica, procedente de Sagunto, II: aspectos epigráficos, lingüísticos y culturales». *Arse*, 26, 17–58.
- VILLAR F. (1999) *Joan Coromines y los substratos prerromanos de la península ibérica*, págs. 53–65. Aula de Ciència i Cultura, 4. Fundació Caixa de Sabadell, Sabadell.