

Creación de Listas Avanzada

Folleto

Curso Online de InnoU

Versión 2011

Innovative
interfaces

Copyright © 2011 por Innovative Interfaces, Inc.

Esta publicación está destinada al uso exclusivo de los usuarios de Innovative Interfaces y se distribuye con el entendimiento de que no se habrá de distribuir a nadie fuera de la organización a no ser que se cuente con el consentimiento expreso y por escrito de Innovative Interfaces. Esta publicación puede ser reproducida sólo si las copias son para el uso exclusivo de los miembros del personal de las bibliotecas que han comprado el sistema de Innovative.

Publicado el 20 de Junio 2012

Creación de Listas Avanzada

Curso Online de InnoU

Contenido

Introducción	4
Objetivos del curso	4
Creación de listas de revisión	4
Ventana de Crear Listas	5
Condiciones booleanas Avanzadas	6
NO TIENEN (Todos los campos/Un campo)	6
Búsqueda de campos especiales (MARC Leader, 006, 007, 008) 9	12
Guardar búsquedas	15
Registros sin vínculos	18
Agrupación (anidación) de las estrategias booleanas	19
Condiciones booleanas Avanzadas	21
Concordancias	21
Recuperación de registros sin especificar condiciones booleanas	25
Búsqueda en un índice en lugar de en un rango	26
Búsqueda avanzada (índice palabra clave) en lugar de en el índice pro Rango o Revisión	27
Anexar registros a un archivo existente	29
Añadir registros adicionales	29
La fusión de dos archivos de	32
Buscar sólo un registro en el rango	32
Copiar una lista	34
Extraer los registros enlazados (relacionados) en otro archivo	35
Apropiarse de las listas	36
Uso de plantillas para Crear Listas	37
Cosas que no se puede hacer al Crear Listas	40
Configuración necesaria	40
Autorizaciones	40
Opciones del sistema	41
Configuración	41
Documentación y recursos	42

Introducción

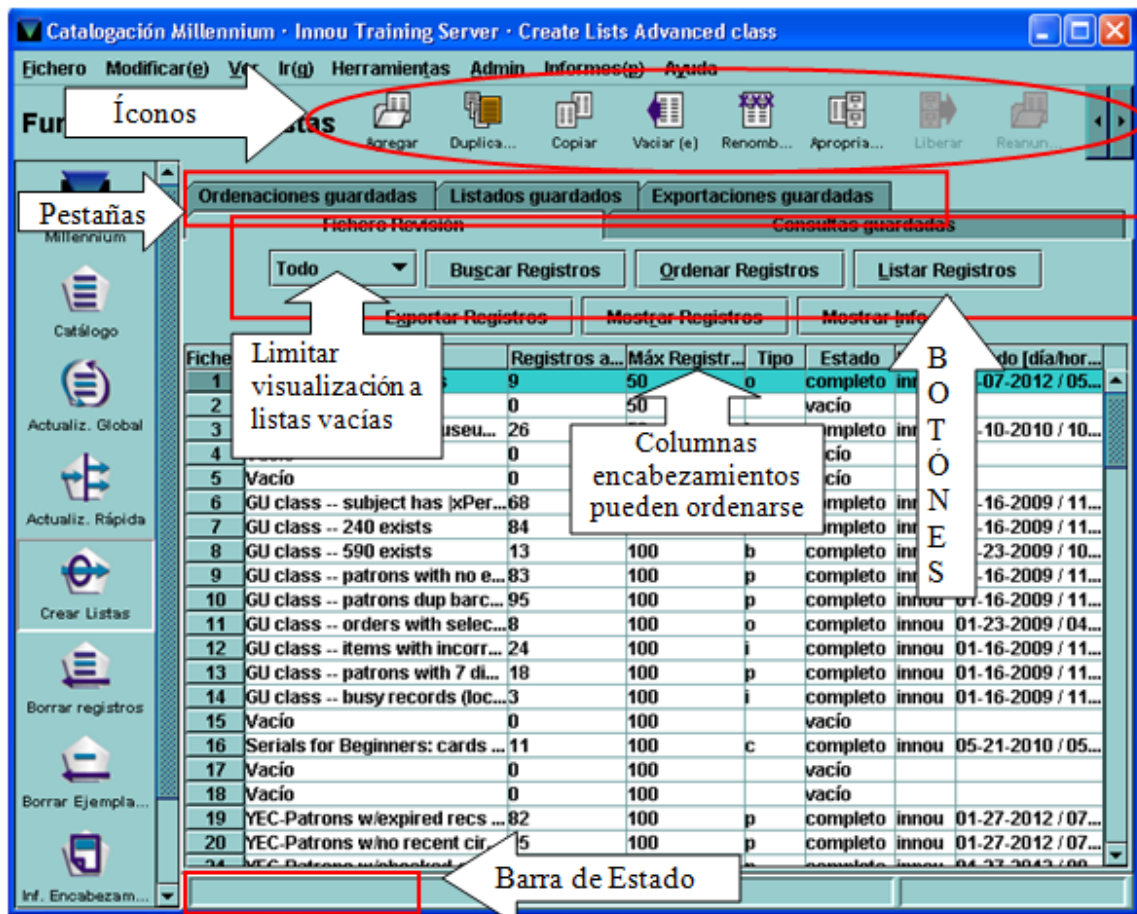
Objetivos del Curso

- Aprender algunas técnicas avanzadas en la creación de listas
 - condiciones booleanas más avanzados
 - Buscar en campos especiales (Marc Leader, 008, etc)
 - Estrategias de anidación booleanas
 - Adjuntar registros a los archivos existentes
 - Copiar otros archivos (incluyendo los archivos generados por el sistema)
 - Mezclar tipos de registros
- Aprender a buscar
 - Índices
 - Índices avanzados (de palabras clave)
 - Archivos existentes
- Guardar y utilizar estrategias de búsqueda guardadas

Creación de listas - Revisión

- Todos los datos en todos los tipos de registros se pueden buscar
 - La actualización la fecha y de la fecha de creación son una excepción. Pueden ser buscados únicamente en el tipo de registro que esta “guardado” en el archivo.
- Los registros se pueden ordenar en los archivos.
- Se pueden hacer listados de campos para imprimir.
- El texto puede ser obtenido en formato de texto / delimitado.
- Los registros pueden ser modificados uno a uno en el archivo.
- Los criterios para ordenar, hacer listados y para exportar pueden ser guardados para ser utilizados posteriormente
- Los archivos de revisión se puede usar con otras funciones como: Listas de destacados, Actualización Global / Rápida, Eliminar el modo de los registros, Estadísticas Millennium, Exportación de registros MARC
- Disponible en todos los módulos principales: Administración, Adquisiciones, Catalogación, Circulación, ERM y Seriadas

Pantalla para la Creación de Listas



- **Íconos** de la barra de herramientas en la parte superior. (# 101067)
- **Pestañas** para diferentes funciones
- **Botones** arriba de las columnas.
- Revisión de los archivos están en filas.
- Las **columnas** se pueden ordenar por título de la columna - Ordenar por ejemplo, por iniciales. Cuando se ordena, no se hace de forma permanente.
- **Menú desplegable**, se puede ver con filtro para mostrar sólo los archivos vacíos, por ejemplo. Los archivos vacíos, se pueden clasificar según el tamaño haciendo clic en el encabezado de la columna Máx Registros.

NOTA: Si se copia un archivo de revisión en un archivo vacío, ya no puede ser visto con la operación filtro de "vaciado" (puesto que ya no está vacío).

- La fecha de **Creado en** [Fecha / Hora] cambia al ordenar o manipular el archivo. La columna de Creación [Fecha / Hora] está vacía (así como la columna de las iniciales) si el archivo fue creado en el sistema Innopac basado en caracteres y no en Millennium o si un archivo generado por el sistema fue copiado.
- La **Barra de Estado** situada en la parte inferior se mostrará el último registro buscado, mientras que la lista se va compilando.

Condiciones booleanas Avanzadas

NO TIENE (Todos los campos no tienen/Al menos un campo no tiene)

NO TIENE es el concepto opuesto a TIENE, pero hay dos posibilidades. Uno de ellas es que *todos los campos* NO tienen los datos que se especifican, y la otra es que *al menos UN campo* no tiene los datos. Esto hace que el concepto sea mucho más confuso de usar. Sin embargo, ambos son muy útiles en determinados momentos.

ESCENARIO: La biblioteca quiere crear una bibliografía de ciertas temáticas religiosas, uno de los cuales es "iglesia":

- **Tipo de Registro almacenado:** Bibliográfico
Bib Materia Contiene iglesia
- Pinche en **Buscar** para crear el archivo
- Al verse los registros, sólo se ven los títulos (de los registros bibliográficos). Se puede ver que se obtienen algunos títulos extraños:

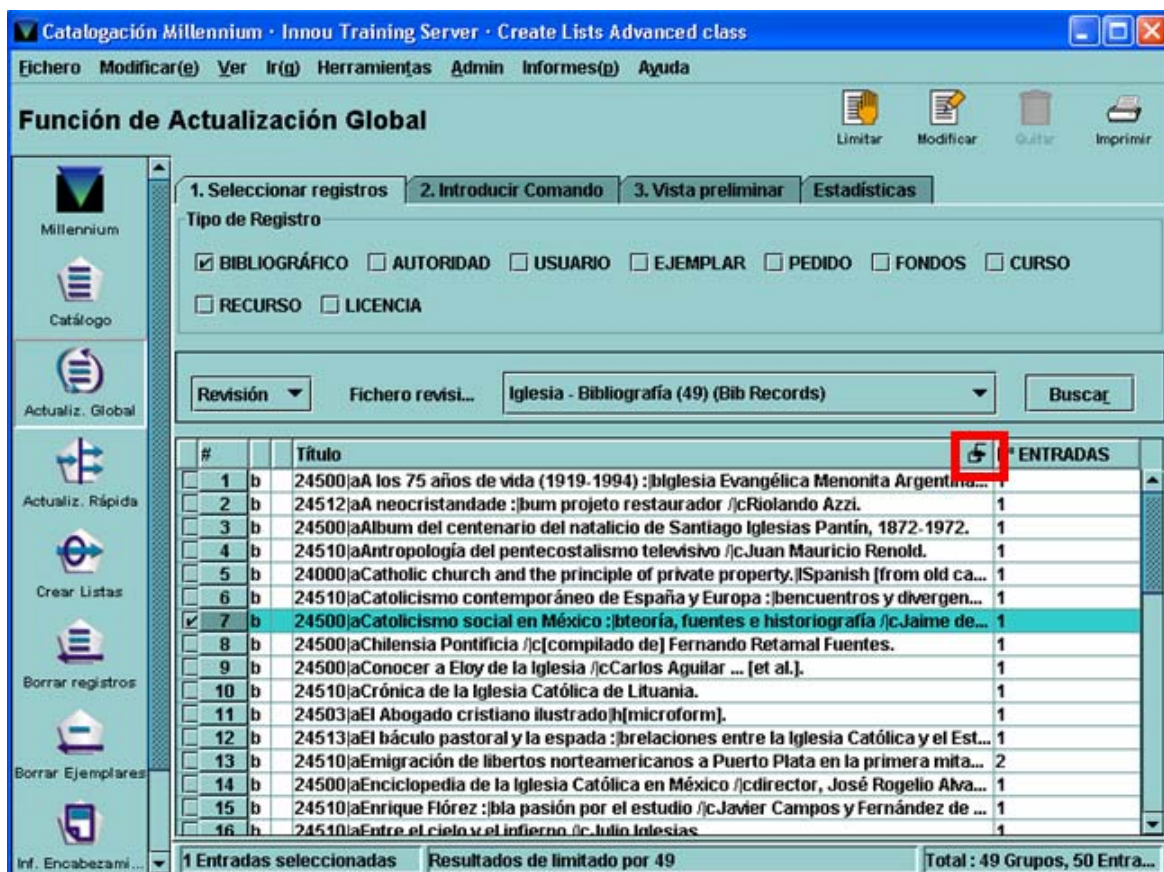
Fichero de Revisión:Iglesia - Bibliografía		
Fichero Herramientas		
Fichero de Revisión:Iglesia - Bibliografía		
	Registro	Descripción
☐	21 b12179784	Reflexiones sobre la autoridad y su ejercicio en la Iglesia Católica.
☐	22 b12179796	Crónica de la Iglesia Católica de Lituania.
☐	23 b12179802	Enciclopedia de la Iglesia Católica en México / director, José Rogelio Álvarez.
☐	24 b12179814	Ética y economía : la contribución de la Iglesia Católica en América Latina / Eugenio Nasarre...
☐	25 b12179826	Guía de la Iglesia Católica en el Uruguay.
☐	26 b12179838	Guía de la Iglesia en España.
☐	27 b1217984x	Guía Internet de la Iglesia Católica : guía de recursos Web en habla castellana y portuguesa...
☐	28 b12179851	La arca salutar : discurso sobre la unidad de la Iglesia Católica, fuera de la cual no hay s...
☐	29 b12179863	La Iglesia Católica en Bolivia / Josep M. Barnadas.
☐	30 b12179875	La Iglesia Católica y el futuro político de Chile / Raúl Atria ... [et al.] ; Patricio Dooner, editor.
☐	31 b12179887	La modernización de la relación institucional entre la iglesia católica y el Estado mexicano, ...
☐	32 b12179899	La Pastoral social en Costa Rica : documentos y comentarios acerca de la polémica entre l...
☐	33 b12179905	A neocristandade : um projeto restaurador / Riolando Azzi.
☐	34 b12179917	Mi cautiverio en el dominio rojo, 19 julio 1936 a 25 julio 1937 : memorias / Luis Almarcha Her...
☐	35 b12179929	Modalidades históricas de asilo y su praxis en Navarra (siglos IV-XIX) / Hipólito Rico Aldave.
☐	36 b12179954	Enrique Flórez : la pasión por el estudio / Javier Campos y Fernández de Sevilla.
☐	37 b12179966	Catolicismo contemporáneo de España y Europa : encuentros y divergencias / José Manuel ...
☐	38 b12179978	Emigración de libertos norteamericanos a Puerto Plata en la primera mitad del siglo XIX : la...
☐	39 b1217998x	Grupos pentecostales : una mirada antropológica a la Iglesia Filadelfia en Andalucía / Manu...
☐	40 b12180014	Conocer a Eloy de la Iglesia / Carlos Aguilar ... [et al.].
☐	41 b1218004x	Album del centenario del natalicio de Santiago Iglesias Pantín, 1872-1972.
☐	42 b12180075	Grandes artistas españoles = Great Spanish artists.
☐	43 b12180105	La Iglesia Evangélica Española / por Daniel V. dal Regaliza.
☐	44 b12180117	Antropología del pentecostalismo televisivo / Juan Mauricio Renold.
☐	45 b12180129	Quisqueya cuenta su historia y celebra su fe : Puerto Plata y Samaná cunas del protestanti...
☐	46 b12180130	Entre el cielo y el infierno / Julio Iglesias.
☐	47 b12180142	Julio / Elizabeth García.
☐	48 b12180154	Julio Iglesias, esta es mi vida / Gregorio Huertas.
☐	49 b12180166	Julio Iglesias : the new Valentino / by Steve Gett.
☐	50 b12180178	

Con el fin de ver todos los encabezamientos de materia de los registros en el archivo que pudieran incluir la palabra "iglesia", es necesario utilizar el modo de actualización global. Allí se puede navegar a través de los encabezamientos de materia y descubrir los temas que incluyen la palabra "iglesia" que se necesita eliminar en la estrategia. Materias tales como:

Iglesias, Julio
 Iglesia, Eloy de la
 Iglesia Metodista
 Iglesia Evangélica

- En Millennium Catalogación cambie al modo de Actualización Global, seleccione el tipo de registro de la lista (en este caso, Bibliográfico, el cual está seleccionado por defecto).
- Cambie de Índice a Revisión
- Encuentre la lista en los ficheros revisión de registros bibliográficos y pinche en Buscar para visualizar los registros.

- En el menú Herramientas, seleccione Conmutar | Bibliográfico Campos Variables| Materias. (También puede ir a la barra del encabezamiento a la derecha del Título y pinchar con el ratón a la derecha sobre el ícono de Conmutar se ve tal cual se muestra en la toma de pantalla a continuación.)



- Todos los campos que se muestran ahora son los campos temáticos de los registros. Se puede ver Iglesia Evangélica en los resultados de pantalla.

Catalogación Millennium · Innou Training Server · Create Lists Advanced class

Fichero Modificar(e) Ver Ir(g) Herramientas Admin Informes(p) Ayuda

Función de Actualización Global

Limitar Modificar Quitar Imprimir

1. Seleccionar registros 2. Introducir Comando 3. Vista preliminar Estadísticas

Tipo de Registro

☒ BIBLIOGRÁFICO ☐ AUTORIDAD ☐ USUARIO ☐ EJEMPLAR ☐ PEDIDO ☐ FONDOS ☐ CURSO

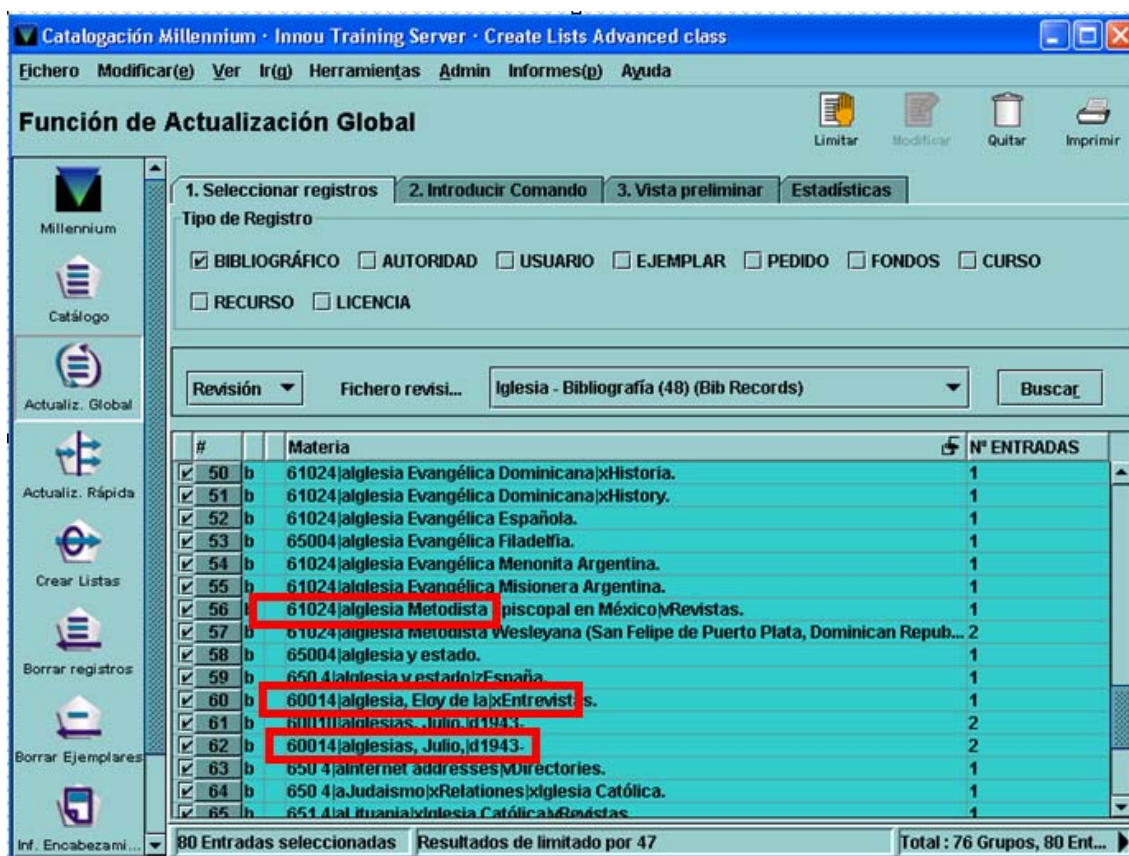
☐ RECURSO ☐ LICENCIA

Revisión Fichero revisi... Iglesia - Bibliografía (49) (Bib Records) Buscar

#		Materia	Nº ENTRADAS
42	b	61024 alglesia Católica zChile xHistoria.	1
43	b	61024 alglesia Católica zCosta Rica.	1
44	b	61024 alglesia Católica zEspaña xDirectorios.	2
45	b	61024 alglesia Católica zEspaña xFinanzas.	1
46	b	61024 alglesia Católica zMéxico xEnciclopedias.	1
47	b	61014 alglesia Católica zMéxico xHistoria.	1
48	b	650 0 alglesia Católica zRepública Dominicana xHistoria xDocumentos xIndex.	1
49	b	61024 alglesia Católica zUruguay xDirectorios.	1
50	b	61024 alglesia de la Casa Profesa (Mexico City, Mexico) xHistory.	1
51	b	61024 alglesia de Orosi.	1
52	b	61024 alglesia de San Agustín (Bogotá, Colombia)	1
53	b	61024 alglesia de San Pedro de Ansó (Spain)	1
54	b	61024 alglesia Evangélica ominicana xHistoria.	1
55	b	61024 alglesia Evangélica ominicana xHistory.	1
56	b	61024 alglesia Evangélica pañola.	1
57	b	65004 alglesia Evangélica adelfia.	1

89 Entradas seleccionadas Resultados de limitado por 49 Total : 85 Grupos, 89 Ent...

- Desplazándose hacia abajo se ve Iglesias, Julio; Iglesia, Eloy de la, e Iglesia Metodista, etc



- Para eliminar esos encabezamientos puede tratar lo siguiente: Materia TIENE “iglesia” (poner un <espacio> frente a "iglesia"), pero esto sólo se eliminaría "Iglesia, Eloy de la" e "Iglesias, Julio", pero no una materia como “esclavitud e iglesia. En su lugar, utilice la condición TODOS LOS CAMPOS NO TIENEN y vuelva a buscar en el archivo:

- Tipo de registro almacenado:** Bibliográfico

	Bib	Materia	CONTIENE	iglesia
Y		Materia	TODOS LOS CAMPOS NO TIENEN	Iglesias, Julio
Y		Materia	TODOS LOS CAMPOS NO TIENEN	Iglesia, Eloy de la
Y		Materia	TODOS LOS CAMPOS NO TIENEN	Iglesia Evangélica
Y		Materia	TODOS LOS CAMPOS NO TIENEN	Iglesia Metodista

Tér...	Operador	Tipo	Campo	Condición	Valor A	Valor B
1		BIBLIOGRÁF...	Materia	contiene	iglesia	
2	Y	BIBLIOGRÁF...	Materia	todos los ca...	iglesias, julio	
3	Y	BIBLIOGRÁF...	Materia	todos los ca...	iglesias, eloy de la	
4	Y	BIBLIOGRÁF...	Materia	todos los ca...	iglesia metodista	
5	Y	BIBLIOGRÁF...	Materia	todos los ca...	iglesia evangélica	

- Ahora se podrá ver que Iglesias, Julio e Iglesia Evangélica, etc ya no están:

#		Materia	Nº ENTRADAS
42	b	61024 Iglesia Católica Uruguay xDirectorios.	1
43	b	61024 Iglesia de la Casa Profesa (Mexico City, Mexico) xHistory.	1
44	b	61024 Iglesia de Orosi.	1
45	b	61024 Iglesia de San Agustín (Bogotá, Colombia)	1
46	b	61024 Iglesia de San Pedro de Ansó (Spain)	1
47	b	65004 Iglesia y estado.	1
48	b	650 4 Iglesia y estado España.	1
49	b	650 4 Internet addresses Directories.	1
50	b	650 4 Judaismo xRelaciones xIglesia Católica.	1
51	b	651 4 Lituania xIglesia Católica xRevistas.	1
52	b	651 0 Mexico City (Mexico) xChurch history.	1
53	b	651 0 Orosi (Costa Rica) xBuildings, structures, etc.	1
54	b	650 4 Propiedad xAspectos morales y and éticos.	1
55	b	650 4 Radio en religión xIglesia Católica xHistoria.	1
56	b	650 4 Religión y sociología México	1
57	b	651 4 Venezuela xHistoria de la Iglesia.	1

- El uso de la condición de AL MENOS UN CAMPO NO TIENE no es tan limitante.
 - Por ejemplo, si lo que se quiere son registros bibliográficos donde el campo de Materia no contenga "iglesia" y se elige AL MENOS UN CAMPO NO TIENE, lo que se obtendría serían registros que tengan "Iglesia, Eloy de la", si los registros tuvieran más campos de materias, ya que los campos temáticos adicionales no contienen el término "iglesia".
- Un ejemplo de cuándo AL MENOS UN CAMPO NO TIENE es útil, es si lo que se está buscando son ejemplares que pertenecen a la biblioteca XXX, y también a otras bibliotecas.
- Crear lista de registro almacenado: Bibliográficos y busque:
 Ejemplar (o Bib) Ubicación Igual que xxx
 Y Ejemplar (o Bib) AL MENOS UN CAMPO NO TIENE xxx
- Con esto se recuperarán solamente aquellos títulos en los que haya ejemplares que pertenezcan tanto a XXX como a otras localidades. Otras condiciones se pueden añadir para limitar aún más la búsqueda.

Búsqueda de Campos Especiales (MARC Leader, 006, 007, 008)

En ocasiones el MARC Leader o los campos 006, 007, o 008 contienen datos que pudiera identificar los registros que se quisieran recuperar. En ese caso se pueden utilizar las "pestañas" para seleccionar si se quiere buscar el leader, 006, 007 o 008 para buscar en los registros Bibliográficos o de Fondo.

Ejemplo 1: MARC Leader

ESCENARIO: La Biblioteca quiere identificar los registros que hayan sido producidos con un "nivel previo a la publicación de catalogación". Esto está codificado en el MARC Leader en el byte del Nivel de Codificación con un nivel 8. Para poder encontrarlos:

- **Tipo de registro almacenado:** Bibliográfico
Bib **MARC Leader** **NIV CODI** **Igual que** **8**
- Haga doble clic en "Campos", para verlos en pantalla. Pinche en la pestaña de MARC21 Leader:

Seleccionar

Campos de Datos **MARC21 Leader** MARC21 006 MARC21 007 MARC21 008

Campos Fijos

24 Idioma	25 No Alfabet	26 Ubicación
28 Catalogado	29 Nivel bib	30 Tipo de Material
31 Bib Code 3	89 Pais	81 N° Registro
83 Creado	84 Actualiz	85 Revisiones

Campos Variables

c Signatura	a Autor	t Título
e Edición	p Publicación	r Descripción
s Colección	n Nota	d Materia
b Otro Autor	u Otro Título	x Es continuación de
z	w	o

OK Cancelar

- Esto revelará todos los ejemplars del campo Marc Leader del que puede seleccionar NIV CODI:

The 'Seleccionar' dialog box displays the 'MARC21 Leader' tab. It contains a grid of fields for selection. The field 'NIV CODI' (labeled d1) is highlighted with a red border.

Campos de Datos					
MARC21 Leader		MARC21 006	MARC21 007	MARC21 008	
a1	REC LENGTH	a2	ESTA REG	a3	TIPO REG
b1	BIB LEVL	b2	CTRL ARC	b3	CHAR ENC
c1	IND CNT	c2	SFLD CNT	c3	BASE ADDRESS
d1	NIV CODI	d2	CAT FORM	d3	LINKED REC
e1	LEN FIELD	e2	LEN START	e3	LEN IMPL
f1	INDEFINI				

Buttons: OK, Cancelar

- Elija la condición booleana IGUAL QUE
- Hay que introducir el valor NIV CODI. No se puede hacer doble clic en Valor A y obtener la tabla de todos los valores válidos (tal como se puede hacer en otros campos fijos). Hay que entrar el valor (8) como se muestra a continuación.

The 'Consulta Booleana' dialog box shows a search query configuration. The 'Nombre fichero revisión...' field contains 'Bibliográficos con NIV CODI Igual que 8'. The 'tipo de registro a almace...' dropdown is set to 'BIBLIOGRÁFICO b'. The 'Rango' dropdown is set to 'Rango'. The 'Iniciar' field contains 'b10000008' and the 'Parar' field contains 'b12180099'. The 'Usar rango de scope' checkbox is unchecked. The table below shows the search criteria.

Tér...	Operador	Tipo	Campo	Condición	Valor A	Valor B
1		BIBLIOGRÁF...	NIV CODI	igual que	8	

Summary: BIBLIOGRÁFICO NIV CODI igual que "8"

Buttons: Grupo

Ejemplo 2: 008

ESCENARIO: La Biblioteca quiere encontrar todas las publicaciones del Gobierno federal de los EEUU. Estas están codificadas en el campo 008 con una "f" en el elemento Govt.Pub. Para encontrarlas:

- **Tipo de registro almacenado:** Bibliográfico
Bib **MARC21 008** **Govt Pub** **Igual que** **f**
- Pinche en la pestaña MARC21 008, y seleccione GovtPub:

Campos de Datos		MARC21 Leader	MARC21 006	MARC21 007	MARC21 008
e1	Ilustra4	e2	Audienc.	e3	FormEjem
f1	Conteni1	f2	Conteni2	f3	Conteni3
g1	Conteni4	g2	Pub Ofic	g3	Pub Cong
h1	Festsch	h2	Indice	h3	Indefini
i1	Form Lit	i2	Biograf.	i3	Indefini
j1	Audienc.	j2	Indefini	j3	TipoFich
k1	Indefini	k2	Pub Ofic	k3	Indefini
l1	Relieve1	l2	Relieve2	l3	Relieve3
m1	Relieve4	m2	Proyeccn	m3	Indefini

- Al igual que con el Marc Leader, no se puede hacer doble clic en un Valor A para obtener la tabla de valores válidos, sino que se debe introducir su valor (en este caso "f") como se muestra a continuación.

Nombre fichero revisión... Publicaciones oficiales

tipo de registro a almace... BIBLIOGRÁFICO b

Rango: Iniciar b10000008 Parar b12180099 ☐ Usar rango de scope

Tér...	Operator	Tipo	Campo	Condición	Valor A	Valor B
1		BIBLIOGRÁFICO	Pub Ofic	igual que	f	

BIBLIOGRÁFICO Pub Ofic igual que "f"

Consultas Guardadas

Muchas bibliotecas desean volver a ejecutar un archivo de forma periódica. El proceso es más sencillo si se guarda la estrategia de búsqueda. Esto elimina la necesidad de recrear la estrategia cada vez que se ejecuta la lista.

- Las estrategias deben guardarse ANTES de pinchar en el botón Buscar.

The screenshot shows the 'Consulta Booleana' (Boolean Query) window. At the top, the 'Nombre fichero revisión...' (Review file name...) field contains 'Bibliográficos sin registros asociados'. Below it, the 'tipo de registro a almace...' (record type to store...) dropdown is set to 'BIBLIOGRÁFICO b'. The 'Rango' (Range) section shows 'Iniciar' (Start) as 'b10000008' and 'Parar' (Stop) as 'b12180099', with an unchecked checkbox for 'Usar rango de scope' (Use scope range). A table below lists three search terms:

Término	Operador	Tipo	Campo	Condición	Valor A	Valor B
1		BIBLIOGRÁFICO	REG ASOCIADO	no existe	EJEMPLAR	
2	Y	BIBLIOGRÁFICO	REG ASOCIADO	no existe	PEDIDO	
3	Y	BIBLIOGRÁFICO	REG ASOCIADO	no existe	FONDOS	

Below the table, the generated Boolean query is displayed: 'BIBLIOGRÁFICO REG ASOCIADO no existe EJEMPLAR AND BIBLIOGRÁFICO REG ASOCIADO no existe PEDIDO AND BIBLIOGRÁFICO REG ASOCIADO no existe FONDOS'. To the right of the query is a vertical toolbar with buttons: 'Grupo', 'Desagrupar', 'Insertar Línea', 'Agregar Línea', 'Borrar', and 'Borrar To...'. At the bottom of the window, there are six buttons: 'Buscar', 'Usar Búsqueda Existente', 'Recuperar Búsqueda Guard...', 'Guar...', 'Guardar ...', and 'Cerrar'. The 'Guar...' and 'Guardar ...' buttons are highlighted with a red border.

- Poner un nombre a la estrategia a guardar.
- Active la casilla de "Incluir la información registro" si desea buscar siempre en el mismo rango de registros o el mismo archivo de revisión (por lo general sólo se realiza si usted está buscando otro archivo de revisión).

Indique el nombre de la consulta a guardar

Guardar consulta c... Bibliográficos sin registros asociados

o elija de entre los nombres existentes: Patron owe more than \$4
 TEMPLATE FILE FOR NEW TITLES LIST
 Univ patrons owe more than \$30
 Variations of bonds (CrL Adv class)
 Orders with funds iu
 Outstanding orders (unit 1)
 Patron telephone no search (CrL Adv class)
 Orders placed last year (YEA class)

☒ Plantilla

☒ Incluir información registro

Guar... Cancelar

- La estrategia guardada puede ser modificada o borrada desde la pestaña de Consultas Guardadas
- Nuevas estrategias también pueden ser creadas en esta pestaña

Ordenaciones guardadas Listados guardados Exportaciones guardadas

Fichero Revisión Consultas guardadas

tipo de registro a almace...

Revisión Fichero revisi...

Tér...	Operador	Tipo	Campo	Condición	Valor A	Valor B
1		BIBLIOGRÁ...	REG ASOCI...	no existe	EJEMPLAR	
2	Y	BIBLIOGRÁ...	REG ASOCI...	no existe	PEDIDO	
3	Y	BIBLIOGRÁ...	REG ASOCI...	no existe	FONDOS	

Consulta	Nombre Consulta	Plantilla
2	TEMPLATE FILE FOR NEW TITLES LIST	☒
3	Univ patrons owe more than \$30	☐
4	Variations of bonds (CrL Adv class)	☐
5	Orders with funds iu	☐
6	Outstanding orders (unit 1)	☐
7	Bibs with nothing attached (CrL Adv class)	☐
8	Patron telephone no search (CrL Adv class)	☐
9	Orders placed last year (YEA class)	☐
10	Bibliográficos sin registros asociados	☒

Nuevo Mo... Borrar

- Por defecto, Millennium está configurado para poder almacenar hasta 100 estrategias. Póngase en contacto con el Help Desk si se necesita aumentar este número.
- La lista de búsquedas guardadas se puede ordenar por cualquier columna. Pinche en la cabecera de la columna, por ejemplo, el "Nombre de la consulta" para ordenar por nombre.
- La columna "Consulta", ordena por orden cronológico, a pesar de que siempre se muestra 1, 2, 3, etc. Dicho orden se guarda durante la sesión en cuestión y no de forma permanente.
- Para usar una estrategia guardada después de haberlo creado, se ha de elegir un archivo vacío y pinchar en el botón "Recuperar la consulta guardada":

Consulta Booleana

Nombre fichero revisión... Vacío

tipo de registro a almace... BIBLIOGRÁFICO b

Rango Iniciar b10000008 Parar b12180099

☐ Usar rango de scope

Tér...	Operator	Tipo	Campo	Condición	Valor A	Valor B
1						

Grupo
Desagrupar
Insertar Línea
Agregar Línea
Borrar
Borrar To...

Buscar Usar Búsqueda Existente **Recuperar Búsqueda Guard...** Guar... Guardar ...

Registros sin vínculos

Hay ocasiones en que las bibliotecas quieren encontrar registros bibliográficos que no tengan registros asociados. Estos se pueden encontrar con la siguiente estrategia (suponiendo que la biblioteca tiene los módulos de seriadas y adquisiciones):

Guarde el tipo de registro: bibliográfico

Bibliográfico Registro Asociado (^)	No existe para	Ejemplar
Y Bibliográfico Registro Asociado (^)	No existe para	Pedido
Y Bibliográfico Registro Asociado (^)	No existe para	Fondos

Consulta Booleana

Nombre fichero revisión...

tipo de registro a almace...

Rango ☐ Usar rango de scope

Tér...	Operador	Tipo	Campo	Condición	Valor A	Valor B
1		BIBLIOGRÁF...	REG ASOCL...	no existe	EJEMPLAR	
2	Y	BIBLIOGRÁF...	REG ASOCL...	no existe	PEDIDO	
3	Y	BIBLIOGRÁF...	REG ASOCL...	no existe	FONDOS	

BIBLIOGRÁFICO REG ASOCIADO no existe EJEMPLAR AND BIBLIOGRÁFICO REG ASOCIADO no existe PEDIDO AND BIBLIOGRÁFICO REG ASOCIADO no existe FONDOS

Grupo

Nota: Es posible que hayan registros bibliográficos que no tengan ningún registro adjunto unido por razones válidas, por ejemplo, los libros electrónicos. Para excluir aquellos se les pudiera podría agregar una condición Y como "el tipo de material no es igual a x" (donde "x" es el tipo de material que la biblioteca utiliza para los libros electrónicos).

Agrupación (anidación- Estrategias Booleanas)

La agrupación se puede utilizar cuando se combinan el operador Y junto a múltiples O. Si se está utilizando Y seguido por más de una condición O, se deben agrupar las O para que funcione correctamente.

ESCENARIO: La biblioteca quiere crear un archivo de todos los registros de pedidos que están codificados "Pedidos firmes" en el código Tipo de Pedido, pero tienen 3 posibles códigos de fondos.

Tipo de registro almacenado: Pedido

	Pedidos	Tipo Pedido	IGUAL QUE	Pedido Firme
Y	Pedidos	Fondo	IGUAL QUE	iu2f
O	Pedidos	Fondo	IGUAL QUE	iuref
O	Pedido	Fondo	IGUAL Que	iumic

- Seleccione desde la línea 2 a la 4 y pinche en el botón de **Grupo** de la derecha.

Consulta Booleana

Nombre fichero revisión...

tipo de registro a almace...

Rango ☐ Usar rango de scope

Tér...	Operador	Tipo	Campo	Condición	Valor A	Valor B
1		PEDIDO	Tipo Ped	igual que	f	
2	Y	PEDIDO	Fondo	igual que	iu2f	
3	O	PEDIDO	Fondo	igual que	iuref	
4	O	PEDIDO	Fondo	igual que	iumic	

PEDIDO Tipo Ped igual que "f" AND PEDIDO Fondo igual que "iu2f" OR PEDIDO Fondo igual que "iuref" OR PEDIDO Fondo igual que "iumic"

Grupo
Desagrupar
Insertar Línea
Agregar Línea
Borrar
Borrar To...

Buscar Usar Búsqueda Existente Recuperar Búsqueda Guard... Guar... Guardar ...

- Esto pone paréntesis después de Y en la segunda línea y al final de las entradas agrupadas como se muestra a continuación.

Consulta Booleana

Nombre fichero revisión...

tipo de registro a almace...

Rango ☐ Usar rango de scope

Tér...	Operador	Tipo	Campo	Condición	Valor A	Valor B
1		b	Tipo Ped	igual que	f	
2	Y	PEDIDO	Fondo	igual que	iu2f	
3	O	PEDIDO	Fondo	igual que	iuref	
4	O	PEDIDO	Fondo	igual que	iumic	

PEDIDO Tipo Ped igual que "f" AND (PEDIDO Fondo igual que "iu2f" OR PEDIDO Fondo igual que "iuref" OR PEDIDO Fondo igual que "iumic")

Grupo
Desagrupar
Insertar Línea
Agregar Línea
Borrar
Borrar To...

Buscar Usar Búsqueda Existente Recuperar Búsqueda Guard... Guar... Guardar ...

Condiciones Booleanas Avanzadas

CONCORDANCIAS

Las CONCORDANCIAS utilizan Expresiones Regulares para encontrar ciertos "patrones" dentro de la expresión de búsqueda. Es similar a TIENE, pero permite una mayor flexibilidad. Se suele utilizar con campos de longitud variable, y se puede utilizar con algunos campos de longitud fija (como el código de localización o el código de fondo) con ciertas restricciones.

ESCENARIO 1: Muchos campos MARC no permiten sub-campos repetidos. Si una biblioteca descubre un campo de título, por ejemplo, que tiene dos subcampos "b" o dos subcampos "c", es posible que se desee crear un archivo de revisión para encontrar otros casos que coincidan con este patrón. Esto es posible solamente con CONCORDANCIAS. La expresión regular ".*" (<punto><asterisco>) significa "Cualquier carácter cualquier número de veces" lo que podría ser cualquier número de caracteres entre la primera aparición de | b y la segunda aparición. Para encontrar los siguientes:

- Guardar tipo de registro: Bibliográfico
 - Bib !(245) CONCORDANCIAS |a.*|a
 - O Bib !(245) CONCORDANCIAS |b.*|b
 - O Bib !(245) CONCORDANCIAS |c.*|c
- Esto encontrará todos los 245 campos que contienen instancias repetidas de subcampos a, b, o c, que son todos sub-campos no repetibles.

Tér...	Operador	Tipo	Campo	Condición	Valor A	Valor B	
1		BIBLIOGRÁF...	Campo MAR...	concuerda (...	a.* a		▲
2	O	BIBLIOGRÁF...	Campo MAR...	concuerda (...	b.* b		
3	O	BIBLIOGRÁF...	Campo MAR...	concuerda (...	c.* c		▼

BIBLIOGRÁFICO Campo MARC 245|a concuerda (expr. reg. UNIX) "|a.*|a" OR BIBLIOGRÁFICO Campo MARC 245|b concuerda (expr. reg. UNIX) "|b.*|b" OR BIBLIOGRÁFICO Campo MARC 245|c concuerda (expr. reg. UNIX) "|c.*|c"

Grupo
Desagrupar

ESCENARIO 2: La biblioteca quiere producir una bibliografía histórica de la novela en España. La biblioteca utiliza el campo 655 (encabezamiento del género) de la Novela Histórica, pero esto no limitaría la búsqueda de la historia de este género en España

solamente. Los encabezamientos de materia podrían incluir: España|xHistoria|y414-711, Reino Visigodo|vNovelas; España|xHistoria |y1556-1598, Reinos asturianos|vNovelas; España|xHistoria|y1936-1939, Guerra Civil|vNovelas. Usando la condición de búsqueda CONCORDANCIAS se pueden obtener todos estos encabezamientos en una condición Booleana:

- **Tipo de registro almacenado:** Bibliográfico
 Bib Materia CONCUERDA España.*historia.*novelas

".*" (<punto><asterisco>) significa cualquier número de caracteres entre España e Historia y cualquier número de caracteres entre historia y novelas.

ESCENARIO 3: La biblioteca utiliza un código de localización de dos letras para los registros bibliográficos y uno de 3-5 para registros adjuntos (ejemplares, fondos, pedidos). Se quieren encontrar todos los registros que lleven incorrectamente códigos de dos letras. Las expresiones regulares funcionan de forma diferente con campos de longitud fija más que con campos de longitud variable, ya que sólo va a leer el número de caracteres en la "expresión". Dado que los códigos de localización son de 5 caracteres, el sistema sólo va a leer los primeros 5 caracteres de las "expresiones" CONCORDANTES. Se puede utilizar la siguiente "expresión" de 5 caracteres.

- **Tipo de registro almacenado:** Ejemplar
 Ejemplar Ubicación CONCUERDA
 .. <espacio> <espacio> <espacio> (dos puntos, seguidos por tres espacios – ¡TIENE que tener los tres espacios!) Esto significa cualquiera de los dos caracteres (.), seguido por tres espacios.

ESCENARIO 4: La biblioteca quiere actualizar todos los números telefónicos de 10 dígitos. En muchos de los registros de sus usuarios los números de teléfono solamente tienen siete dígitos en dos patrones: xxx-xxxx o xxxxxxxx. Las CONCORDANCIAS puede encontrar estos registros con la estrategia siguiente:

- **Tipo de registro almacenado:** Usuario

Usuario	Teléfono	CONCUERDA	$\wedge[0-9]\{3\}\-[0-9]\{4\}\$$
O Usuario	Teléfono	CONCUERDA	$\wedge[0-9]\{7\}\$$

Esto se traduce de la siguiente manera:

- ^ es el carácter que ancla el resto de los datos al comienzo del campo
- [0-9] se refiere a cualquier número
- {3} equivale a un total de tres números
- \ significa tratar el siguiente carácter de forma literal y no como parte de la expresión, por lo que el guión (-) SIGNIFICA en realidad un guión
- [0-9] se refiere a cualquier número
- {4} significa un total de cuatro números
- \$ es el final del símbolo del campo

Recuperación de registros sin especificar condiciones BOOLEANAS

Si no se especifica ningún criterio en la línea 1, se podrán "recuperar todos los registros en el rango." Esta es la búsqueda más utilizada para índices o ficheros de revisión existentes. El siguiente ejemplo muestra la recuperación de todos los registros bibliográficos que se encuentran en un rango asignaturas.

Cuando el sistema pregunte si se desea recuperar todos los registros en el rango, responda **SÍ**.

The screenshot shows a window titled "Consulta Booleana". It contains the following fields and controls:

- Nombre fichero revisión...**: Libros de fútbol americano
- tipo de registro a almace...**: BIBLIOGRÁFICO b
- Indice**: A dropdown menu.
- LC CALL NO (c)**: A dropdown menu.
- gv939**: A text input field, highlighted with a red box.
- a**: A connector symbol.
- gv959.99**: A text input field, highlighted with a red box.

Below these fields is a table with the following headers: Término, Op..., Tipo, Campo, Condición, Valor A, Val... The first row of the table contains the number "1".

The screenshot shows a dialog box titled "Buscar". It contains the following text and controls:

- ?**: A question mark icon.
- No se han especificado criterios de búsqueda.**: A message indicating that no search criteria have been specified.
- ¿Recuperar todos los registros del rango?**: A question asking if all records in the range should be recovered.
- Si**: A button, highlighted with a red box.
- No**: A button.

Buscar un Índice en lugar de un Rango

La búsqueda de un índice es mucho más rápida, ya que sólo busca en el rango del índice que se especifica y no busca todos los registros de la base de datos. Por ejemplo, si desea que todos los registros dentro de un rango de signaturas específicas, se puede buscar en el índice de signatura en lugar de especificar la signatura con lógica Booleana ENTRE (w). Esto tiene ventajas claras.

- Es mucho más rápido, ya que sólo busca en el índice entre el rango de dos signaturas dadas en lugar de buscar la base de datos completa.
- Los registros se recuperan con el orden de la signatura y no tienen que ser ordenados.
- Las bibliotecas con LC (Biblioteca del Congreso) no tiene que utilizar espacios de 7 caracteres para esta búsqueda (para normalizar la signatura), lo cual tendría que hacerse se buscan registros de signatura con condición ENTRE (w) (gv[^]939 a gv[^]959.999)
- Con el índice de signaturas no importa donde esta almacenada la signatura, si en el registro bibliográfico, el ejemplar o el de fondos. Buscando por índice se recuperará el tipo de registro almacenado independientemente del tipo de registro "vinculado" en el cual están los datos de signatura

ESCENARIO 1: La biblioteca quiere encontrar todos los libros de fútbol americano que tenga y verlos en pantalla. El tipo de registro almacenado puede ser bibliográfico aunque la signatura en la base de datos esté almacenada en los registros de ejemplares.

- **Tipo de Registro almacenado:** Bibliográfico
- Cambio de **Rango** al **Índice**
- Cambie a Índice de Signatura en el desplegable.
- Escriba **gv939** en la caja de la izquierda y **gv959.999** en caja de la derecha como se muestra a continuación
- Pinche en **Buscar** y recuperar todos los registros en el rango

The screenshot shows a software window titled "Consulta Booleana". At the top, there is a text field "Nombre fichero revisión..." containing "Libros de fútbol americano". Below it is a dropdown menu "tipo de registro a almace..." set to "BIBLIOGRÁFICO b". The main search area has two dropdowns: "Índice" and "LC CALL NO (c)". To the right of these are two text input fields, the first containing "gv939" and the second containing "gv959.99", both highlighted with a red rectangle. Below the search fields is a table with columns: "Término", "Op...", "Tipo", "Campo", "Condición", "Valor A", and "Val...". The first row of the table has the number "1" in the "Término" column. At the bottom right of the window is a button labeled "Grupo".

ESCENARIO 2: La biblioteca quiere crear una bibliografía de publicaciones de Mario Vargas Llosa. Utilizando el índice de autores sería el método más rápido para recuperar los resultados:

- **Tipo de Registro almacenado:** Bibliográfico
- Cambio de **Rango** de **Índice**:
- Cambio Índice de **Autor**
- Escriba: **vargas llosa, mario 1936** en las cajas de búsqueda tanto de la izquierda como en la derecha
- No especifique ningún criterio de búsqueda, esto recuperará todos los registros en el rango.

The screenshot shows a window titled 'Consulta Booleana'. It has a text field 'Nombre fichero revisión...' containing 'Bibliografía de Mario Vargas Llosa'. Below it is a dropdown 'tipo de registro a almace...' set to 'BIBLIOGRÁFICO b'. Further down, there are two search boxes, both containing 'vargas llosa mario 1936'. Above the left search box is a dropdown 'Indice' and above the right is 'AUTOR (a)'. At the bottom is a table with columns: Tér..., Operador, Tipo, Campo, Condición, Valor A, Valor B. The first row has '1' in the 'Tér...' column.

Tér...	Operador	Tipo	Campo	Condición	Valor A	Valor B
1						

Buscar en el índice Avanzado (de palabra clave) en vez de Rango o de Revisión

ESCENARIO 1: La biblioteca quiere encontrar en el catalogo libros sobre la Iglesia para una exhibición especial. Se buscó "Título CONTIENE iglesia" Y "Materia CONTIENE iglesia", pero hay una manera más fácil. La palabra "Iglesia" pudiera estar presente en el campo de título, materia o las notas y pudiera estar indizado en el índice de palabra clave. Puede utilizar la búsqueda Avanzada para buscar en el índice de palabra clave con una entrada y sin condiciones booleanas:

- **Tipo de Registro almacenado:** Bibliográfico
- Cambie de **Rango** a **Avanzado**.
- En el cuadro de búsqueda, escriba: iglesia

The screenshot shows the same 'Consulta Booleana' window. The 'Nombre fichero revisión...' field now contains 'Lista de Material sobre la Iglesia'. The 'tipo de registro a almace...' dropdown is still 'BIBLIOGRÁFICO b'. Below, the search mode is set to 'Avanzado' (indicated by a dropdown) and the search box contains 'iglesia'. The table at the bottom is the same as in the previous screenshot.

Tér...	Operador	Tipo	Campo	Condición	Valor A	Valor B
1						

- Pinche en Buscar para iniciar la búsqueda sin especificar ningún criterio de búsqueda.
- La búsqueda se pudiera limitar más aún, a registros bibliográficos con determinada localización (añadiendo una condición Booleana – si se desea):
Bib Ubicación IGUAL QUE iu

Consulta Booleana

Nombre fichero revisión...

tipo de registro a almace...

Avanzado

Tér...	Operador	Tipo	Campo	Condición	Valor A
1		BIBLIOGRÁF...	Ubicación	igual que	iu

BIBLIOGRÁFICO Ubicación igual que "iu "

- Esta búsqueda se efectuará en el índice de palabras clave, buscando el término "Iglesia" y limitando los registros recuperados a aquellos que tengan un código de ubicación "iu", lo cual se lleva a cabo en cuestión de segundos.

ESCENARIO 2: La biblioteca quiere crear una bibliografía de materiales sobre la política en México. Esta es una búsqueda avanzada y por lo tanto más compleja. Utiliza los operadores booleanos dentro de la cadena de búsqueda:

- **Tipo de Registro almacenado:** Bibliográfico
- Cambio de **Rango** a **Avanzado**.
- En el cuadro de búsqueda, escriba: política and México not nuevo México

Consulta Booleana

Nombre fichero revisión...

tipo de registro a almace...

Avanzado

Tér...	Operador	Tipo	Campo	Condición	Valor A	Valor B
1						

- El índice Avanzado pueden usar los mismos métodos utilizados en la WebPac - por ejemplo, el uso de paréntesis y asteriscos, como en el siguiente ejemplo:

ESCENARIO 3: La Biblioteca quiere encontrar todo los materiales que tiene sobre la contaminación del medio ambiente o sobre materiales tóxicos para el medio ambiente.

- **Tipo de Registro almacenado:** Bibliográfico
- Cambio de **Rango** a **Avanzado**:
- En el cuadro de búsqueda: **medio ambiente (toxic* o contamina*)**
- Esta búsqueda recupera registro que contienen el término medio ambiente junto con toxica, tóxico, tóxicos, tóxica, tóxicas, toxicología, etc. O contaminación, contaminado, etc.

Tér...	Operador	Tipo	Campo	Condición	Valor A	Valor B
1						

- Esto recupera el mismo número de registros recuperado en el WebPac con la misma estrategia de búsqueda avanzada. (Menos los registros ocultos que no se pueden recuperar en el WebPac.)

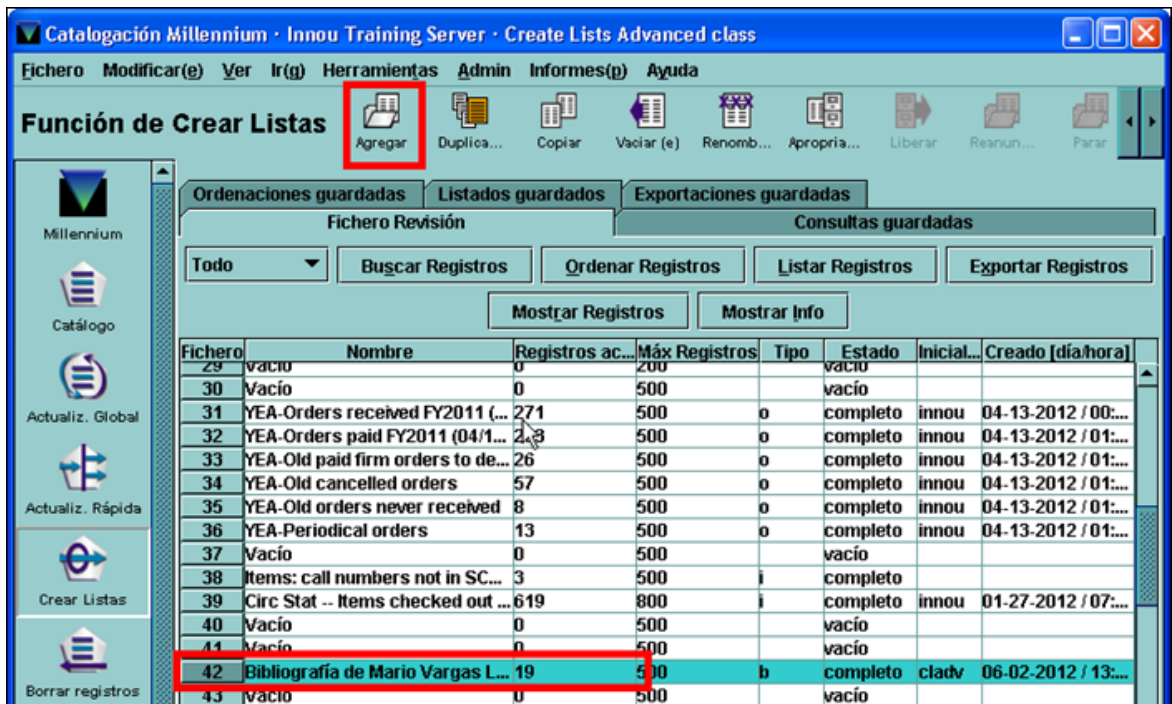
Anexar registros a una lista existente

Añadir registros adicionales

Si usted tiene registros en una lista y desea añadir registros adicionales a la lista se puede utilizar la función Agregar.

ESCENARIO: Se ha creado una lista de registros que tiene Vargas Llosa, Mario, 1936 - como *autor*. Luego, se desea agregar todos los registros que tienen Vargas Llosa, Mario, 1936 - como *materia*.

- Seleccione el archivo Bibliografía Vargas Llosa creado en el Índice de Autor como el ejemplo de arriba:
- Pinche en el icono **Agregar** en la barra de herramientas.

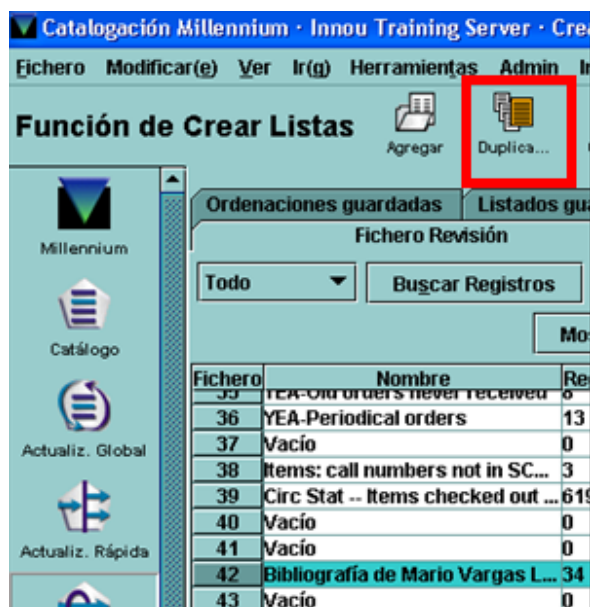


- Cambie de **Rango** a **Índice**, a continuación, seleccione el índice de **Materias** del menu desplegable.
- Escriba: **vargas llosa mario 1936**, en la caja de búsqueda de la izquierda y en de la derecha.
- Pinche en el botón de **Buscar** (sin criterios booleanos especificados).

Tér...	Operador	Tipo	Campo	Condición	Valor A	Valor B
1						

- Esto agregará (adjuntará) los registros que tienen Vargas Llosa, Mario, 1936 - con encabezamientos de materia a los registros existentes en la lista original (aquellos con Vargas Llosa, Mario, 1936 - como autor). Es posible que algunos de los registros que ya estaban en el archivo original de autor, también estén en la lista de materias para Vargas Llosa, en cuyo caso los registros se añadirían al archivo una segunda vez.
- Utilice el icono de eliminar duplicados en la barra de herramientas para eliminar los registros duplicados en el archivo. En este ejemplo, el archivo tenía originalmente 19 registros, al agregar los registros con materia Vargas Llosa Mario 1936 ha pasado a tener

37 registros. En próximo paso hay que eliminar los registros duplicados (aquellos que tienen autor y materia Vargas Llosa Mario 1936



- Después de eliminación de duplicados, el archivo pasa a tener 28 registros, lo que significa que hubo 9 registros que tenían Vargas Llosa como autor y como encabezamiento de materia.



Fusionar dos archivos

La función Agregar también puede ser utilizada para combinar dos archivos existentes. Para ello:

- Seleccione una de las listas, pinche en el icono Agregar en la barra de herramientas.
- Cambie el rango de lista desplegable para examinar y seleccionar la segunda lista del desplegable de los ficheros de revisión. (La lista que está buscando no necesita contener el mismo tipo de registros almacenados en el archivo original (bibliográficos, pedidos, ejemplares, etc.), pero debe contener este tipo de registro en un posible "enlace". Por ejemplo, no se pueden tener los registros bibliográficos en la lista original y los registros de usuario en una lista que se decida buscar. Sin embargo se puede tener registros bibliográficos en la lista original y una lista con pedidos. Se agregarán los registros bibliográficos a los cuales los pedidos de la lista están asociados.
- NO especifique ninguna condición booleana
- Pinche en el botón **Buscar** en la parte inferior izquierda.
- Responda "Sí" al cuadro de diálogo que dice: "No se han especificado criterios booleanos. Recuperar todos los registros en el rango? "
- Los registros de la segunda lista (seleccionada) serán agregados (adjuntados) a los registros en la primera lista.
- En caso de que el mismo registro(s) estuviera en los dos archivos, utilice el icono de deduplicación para eliminar los duplicados.

Buscar un registro dentro del rango de registros

En ocasiones se necesita colocar sólo un registro en un archivo de revisión. Esto puede ser hecho de una manera mucho más fácil por número de registro (de cualquier tipo de registro) o por código de barras (para los registros de ejemplares).

ESCENARIO: La biblioteca tiene un carrito lleno de material para ser descartado y quiere poner todos los registros de ejemplares de estos materiales en un fichero de revisión (una lista) para borrarlos. A partir de la versión 2009B se pueden eliminar ejemplares sencillamente mediante escaneo de los códigos de barras en el modo de eliminación de registros. Pero si la biblioteca quiere primero recoger estadísticas de estos registros, los registros tienen que estar en una lista.

Los registros de ejemplar en el carrito no tienen nada en común, por lo cual, usar la lógica booleana no funcionaría. Los registros de ejemplar pueden ser recopilados, sin embargo, escaneándolos en las dos opciones, (desde el índice de código de barras) y añadiéndolos los otros registros utilizando el icono Agregar:

- **Tipo de registro almacenado:** Ejemplar
- Cambio de **Rango** al **Índice**.
- Seleccione el índice de códigos de barras a partir del índice en el menú desplegable.
- Escanee el primer código de barras en las cajas a la izquierda y a la derecha.
- Pinche en **Buscar** sin especificar ningún criterio de búsqueda
- Este instantáneamente recupera ese registro en el archivo

Consulta Booleana

Nombre fichero revisión... Ejemplares a depurar

tipo de registro a almace... EJEMPLAR i

Indice CÓDIGO DE BARRAS (b) 30102008037549 a 30102008037549

Tér...	Operador	Tipo	Campo	Condición	Valor A	Valor B
1						

- Para añadir los ejemplares restantes, seleccione la lista y pinche en **Mostrar registros** o haga doble clic en la lista. Pinche en el icono **Agregar** en la barra de herramientas.

Fichero de Revisión:Ejemplares a depurar

Fichero Herramientas

Fichero de Revisión:Ejemplares a depurar

Agregar Modificar Quitar Duplica... Cerrar

Registro	Descripción
1 14818358	Historia integral de la Argentina / Félix Luna.

- Desde la ventana de búsqueda, cambie el índice desplegable a código de barras. Explore el código de barras y pulse <Intro> o pinche en **Buscar**.

Add A Record to File

Fichero Modificar(e) Ver Ir(g) Herramientas Ayuda

Nuevo Limitar Ver Modificar Media MetaData WB Sumario Listado Imprimir Cerrar

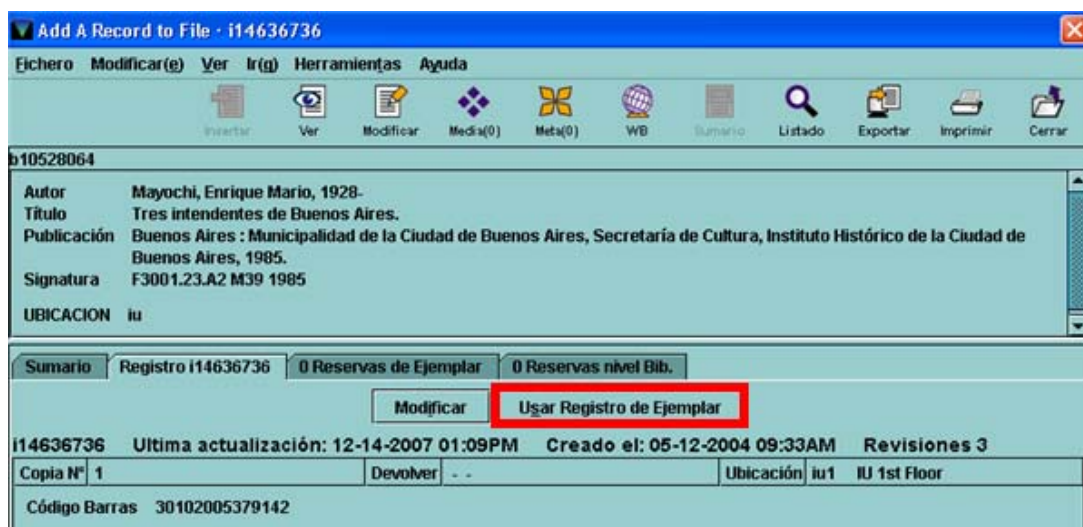
Signatura
Autor
Título
Publicación
Edición
Nº estándar

Índice b CÓDIGO DE BARRAS 30102005379142

Buscar Expandir Todo Remoto INN-View Seleccionar

N° ENTRADAS

- Pinche en **Usar el Registro del Ejemplar** y el registro se añadirá a la lista de revisión. Repita la operación hasta que todos los ejemplares estén en la lista.

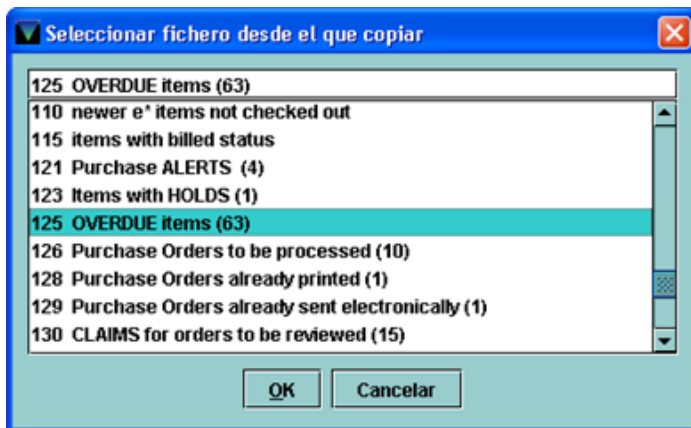


- Se pueden utilizar macros para agilizar este proceso con dos pulsaciones de teclado.

Copiar una Lista

(Listas existentes o listas generadas por el sistema)

- Puede copiar cualquier lista existente en una lista vacía. Además, puede copiar cualquier lista generada por el sistema en una lista vacía. Para ello:
 - o Seleccione una lista vacía
 - o Pinche en el icono **Copiar** de la barra de herramientas.
 (Se puede copiar en una lista existente que no esté vacía, pero la lista copiada se sobrescribirá a la lista existente)
- Esta función es útil si se ha creado una lista que tiene muchos menos registros que el número máximo de registro que puede contener la lista. Se puede utilizar la función **Copiar** para copiar la lista a una lista con menor capacidad, liberando así la lista más grande.
- Esto también es especialmente útil cuando alguien ha creado una lista que le gustaría utilizar sin tener que reorganizar la lista original.
- Los ficheros generados por el sistema siempre se encuentran al final de la lista – con una numeración más alta que el número máximo de listas que se tienen. En este ejemplo, el sistema tiene 81 listas de revisión. Las listas numeradas 82 y para arriba son las listas generadas por el sistema:
- Seleccione una lista vacía, Copie la lista generada por el sistema ejemplares atrasados (Overdue ítems) a la lista vacía.

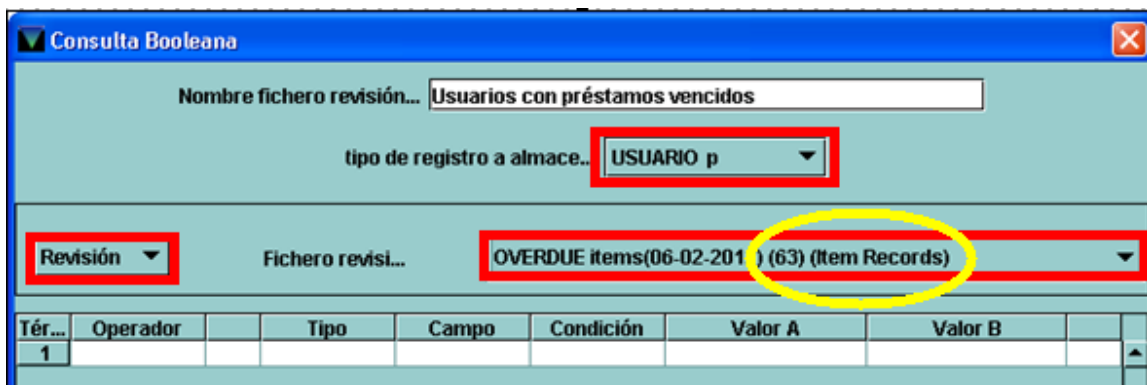


- Esta lista se puede utilizar como se haría con cualquier otra.
- Al copiar la lista, Millennium preguntará si se desea vaciar la lista que está copiando. (Por ejemplo, listas creadas a partir de una carga de registros). Normalmente se desea eliminar la lista.

Extraer los registros enlazados (relacionados) en otro archivo

ESCENARIO: La biblioteca tiene una lista generada por el sistema con ejemplares vencidos (Overdue ítems). Esta lista se ha copiado y a partir de ella se quiere enviar avisos de material vencido a los usuarios que tienen esos préstamos caducados. Se puede seleccionar un archivo vacío y colocarle todos los registros de los usuarios que estén vinculados a los registros de ejemplares (es decir, los usuarios que tienen los materiales prestados):

- Seleccione un archivo vacío y pinche en los registros de búsqueda.
- **Tipo de registro almacenado:** Usuario
- Cambiar el "Rango" a "Revisión"
- Seleccione la lista de *ejemplares vencidos* que se copió en la sección anterior.



- Pinche en **Buscar**.
- Responda "Sí" a la pregunta "Quiere recuperar todos los registros en el rango?"
- Pinche en el icono de eliminar duplicados en la barra de herramientas para eliminar los registros de los usuarios duplicados en el archivo. (No habrá registros duplicados si un usuario tiene más de un ejemplar atrasado.)

Apropiarse de Listas

La funcionalidad que permite apropiarse de listas, evita que ciertas listas sean vaciadas u ordenadas o por otras personas.

- Si se configura la opción de mantenimiento de base de datos (A> A> S> O> D) (Crear listas BOOLEANAS: contraseña en ficheros de revisión) a SI, se verán dos nuevos iconos en la barra de herramientas en el modo Crear listas, el icono "Apropiarse" y el ícono de "Liberar".



- Cuando se selecciona un archivo de revisión y se pincha en el ícono de "Propio", sus iniciales en la columna de Iniciales cambia a cursiva, además de cambiar los colores, como se muestra a continuación:

33	YEA-Old paid firm orders to de...	26	500	o	completo	innou	04-13-2012 / 01:...
34	YEA-Old cancelled orders	57	500	o	completo	sxs	04-13-2012 / 01:...
35	YEA-Old orders never received	8	500	o	completo	innou	04-13-2012 / 01:...

- Si accidentalmente se "apropiara" de la lista de alguien, sus iniciales se mostrarían entonces en esa columna. Sin embargo, cuando se "libere" las siglas originales se volverían a mostrar nuevamente.
- Cuando una lista tiene "propietario" otros pueden ver los registros en el archivo, pero no manipularlos (ordenar, etc) ni tampoco pueden vaciar la lista a menos que tengan la autorización #186 (Administrador de Listado de archivos de revisión) en el archivo de contraseñas.

Usar Plantillas para Crear Listas

- En el modo Crear listas, las plantillas de búsquedas guardadas agilizan las operaciones de búsqueda por el personal de la biblioteca cuando realizan las mismas búsquedas regularmente, pero con diferentes parámetros.
- Los administradores de plantillas pueden guardar una estrategia de búsqueda como una plantilla para que otros la puedan usar.
- Las plantillas limitan a un grupo específico del personal (función 640, creación de listas Limitado) para realizar búsquedas predefinidas solamente.
- Cuando los usuarios restringidos seleccionan una plantilla de búsqueda guardada, Millennium define automáticamente los operadores de búsqueda básicos, basados en la plantilla, y no permite la modificación de dichos operadores.
- Los usuarios limitados sólo pueden cambiar ciertos parámetros para cada operación de búsqueda definida, como un rango de fechas o una ubicación.

ESCENARIO: Una biblioteca grande con muchas sucursales crea una lista de nuevas adquisiciones catalogadas cada mes. El mismo archivo de revisión es utilizado siempre para recuperar esos registros. La biblioteca quiere que cada una de las sucursales pueda crear un archivo de revisión a partir de este archivo maestro (registros bibliográficos que contengan el código de localización de cada sucursal. El archivo maestro se crea con la siguiente estrategia:

Tipo de registro almacenado: Bibliográfico

Bib Catalogado ENTRE xx/xx/xxxx y xx/xx/xxxx

Consulta Booleana

Nombre fichero revisión... Títulos catalogados el mes pasado

tipo de registro a almace... BIBLIOGRÁFICO b

Rango Iniciar b10000008 Parar b12180099 ☐ Usar rango de scope

Tér...	Operador	Tipo	Campo	Condición	Valor A	Valor B
1		BIBLIOGRÁF...	Catalogado	Entre	01-01-2011	01-31-2011

BIBLIOGRÁFICO Catalogado Entre "01-01-2011"and "01-31-2011"

- Cuando un miembro del personal que no tiene la autorización #18 (Crear listas completas), pero tiene la #640 (Creación de listas limitadas) puede acceder al modo Crear listas, pero solamente se le ofrecen las listas de revisión vacías.

Fichero Revisión		Consultas guardadas	
Vacío		Buscar Registros	Ordenar Registros
		Listar Registros	
Fichero	Nombre	Registros actuales	Máx Registros
5	Vacío	0	500
6	Vacío	0	500
10	Vacío	0	500
13	Vacío	0	1000
14	Vacío	0	1000
18	Vacío	0	1000
19	Vacío	0	1000
21	Vacío	0	1000
22	Vacío	0	1000
23	Vacío	0	1000
24	Vacío	0	1000
26	Vacío	0	1000

- Selecione una lista vacía y pinche en **Buscar** Registros. La ventana de Recuperación de consultas se abre.

Retrieve Query

Store Record Type: BIBLIOGRAPHIC b

Re... Review file: All new titles added last month (16) (Bib Recor...

Term	Operator	Type	Field	Condi...	Value A	Value B
1		BIBLI...	Location	equal to	iu	

Query	Query Name	Template
1	test template	☒
2	Newly cataloged bibs for iu last month	☒

Select Close

- Selecione la consulta y pinche en Seleccionar. Los únicos segmentos que se pueden editar en pantalla son el nombre de la lista y el código de ubicación en el valor A.

Consulta Booleana

Nombre fichero revisión. **Títulos nuevos en biblioteca "iu"**

tipo de registro a almace... **BIBLIOGRÁFICO b**

Revisión **▼** Fichero revisi... **Adv. Cr.Lists -- New books Jan. 2011 (78) (Bib Records) ▼**

Tér...	Operador	Tipo	Campo	Condición	Valor A	Valor B
1		BIBLIOGRÁFICO	Ubicación	igual que	iu	

BIBLIOGRÁFICO Ubicación igual que "iu "

Grupo

Cosas que no se puede hacer en Crear Listas

- Buscar registros que tienen múltiples de un campo específico
- Si se crea una lista de todos los registros con un campo específico, se puede exportar el número del registro y el campo específico, y en el archivo exportado buscar aquellos números con múltiples ocurrencias del campo (por ejemplo, los ejemplares con dos códigos de barra). Para lograr esto, muchos usuarios de Millennium, exportan el contenido a otro programa.
- Encontrar registros bibliográficos con más de un registro asociado
- Encontrar registros bibliográficos con un solo registro asociado de un determinado tipo
 - Actualmente no hay manera de ejecutar un comando para encontrar registros bibliográficos con más de un registro vinculado a un tipo determinado.
 - El comando de ^REGISTRO VINCULADO permite al usuario buscar la existencia o no de vínculo a un tipo de registro determinado, pero no hay manera de especificar la cantidad del tipo de registro (uno o muchos).
 - Algunas bibliotecas exportan desde la base de datos los números de registros bibliográficos así como los números de registros asociados y utilizan una hoja de cálculo u otra utilidad de procesamiento de texto para reducir la lista a solamente aquellos registros que cumplan con las condiciones buscadas.
- Distinguir mayúsculas en las búsquedas
- Encontrar usuarios con determinado número de reservas
 - Se pueden encontrar usuarios con reservas, pero no con un número determinado de ellas
- Encontrar usuarios con reservas que estén listas para ser recogidas
- Encontrar usuarios con un número específico de multas
 - Al igual que en el caso de las reservas, se pueden encontrar los usuarios que tengan multas, pero no un número determinado de ellas

Configuración necesaria

Autorizaciones

- Para crear, ver y para mantener los archivos de revisión - Función 018 (Creación de listas).
- Si los archivos de revisión están protegidos por contraseña, el propietario será el único autorizado para manipular, vaciar el archivo (o tener la autorización del Administrador de la Lista (función 186))
- Se debe estar autorizado para ver los tipos de registro en el archivo de revisión
- Para modificar los registros en el archivo, también se debe estar autorizado para modificar los diferentes tipos de registro en el archivo
- Si se utilizan autorizaciones de "Plantillas" 640 (en lugar de 018) y 641 para la administración de la plantilla

Opciones del sistema

D> Base de datos de mantenimiento (A> A> S> S> D)

- 19> "Crear listas de" búsquedas: Pausa cada <n> registros

- Las búsquedas en grandes rangos de registros pueden ralentizar el sistema, en algunos casos (si el servidor es lento / antiguo). Esta opción puede ayudar a reducir el impacto de tales búsquedas de otros procesos del sistema. Es necesario hacer pruebas hasta alcanzar la configuración óptima para su sistema.
- 26> Crear listas BOOLEANAS: mostrar la braced form de los signos diacríticos
- Sólo se utiliza en la base de caracteres Crear listas
- 27> Crear listas booleanas: archivos de revisión de claves de acceso
- NO: Los usuarios pueden modificar o borrar cualquier archivo de revisión.
- SI: Sólo permite a los usuarios que tienen la contraseña #186 para modificar o borrar archivos de revisión que no tienen.
- 32> Crear listas BOOLEANAS: scope during range
- Si la base de datos de la biblioteca tiene un scope, esta opción le permite limitar los resultados al scope asignado al inicio de sesión. Las opciones son:
- N: Recuperar todos los registros de la base de datos, independientemente de su scope
- Y: Restringir los registros recuperados a los que están dentro del scope asociado al login
- A: Ofrecer una casilla de verificación que permite al usuario seleccionar un ámbito de resultados o no-de scopes establecidos
- 37> Crear listas BOOLEANAS: usar plantillas para listas
- Se asigna a esta opción SÍ para habilitar las plantillas de la funcionalidad de búsqueda de Millennium en el modo Crear listas. Las plantillas de búsqueda guardadas también requieren la aplicación de las autorizaciones 640 y 641, según sea apropiado.
- NOTA: Si esta opción indica un valor de N /A, póngase en contacto con Innovative para obtener asistencia en la configuración de las plantillas de búsqueda guardadas.

Configuración

Admin | Configuración | Crear listas

- Opciones para crear listas de búsqueda
 - No esperar a que las búsquedas se completen (por defecto)
Se pueden hacer otros trabajos en la aplicación mientras la búsqueda se está ejecutando, sin tener que esperar a que termine la búsqueda. Esto requiere una licencia adicional.
 - Esperar a que las búsquedas se completen
No se puede hacer otras tareas en la aplicación mientras la búsqueda se está ejecutando, pero la búsqueda no utiliza una licencia adicional
 - Preguntar al usuario si desea esperar a que las búsquedas se completen
Esta opción requiere que otro login (licencia de usuario) este disponible en su grupo de contención para iniciar la búsqueda. Si hay alguna disponible, la búsqueda se inicia como un nuevo proceso del sistema, utilizando una licencia de del grupo de licencias disponibles, lo que le permite realizar otras tareas en la aplicación mientras que la búsqueda se está ejecutando, es decir, sin esperar a que la búsqueda se complete. Si una licencia no está disponible para la búsqueda, el sistema indicará: Lo sentimos pero en este momento no hay licencias adicionales disponibles. ¿Le gustaría que esperar a que la búsqueda se complete? Sí /No
Si su respuesta es "Sí", el sistema utilizará la licencia de usuario actual para iniciar la búsqueda. Una barra de ejecución de proceso aparecerá y no se podrá ejecutar otro trabajo adicional en la aplicación mientras el proceso se esté ejecutando. Si su respuesta es "No", puede guardar los criterios de búsqueda y tratar de iniciar la búsqueda más adelante.

- **Buscar números de Signaturas**
Se utiliza para buscar signaturas en de ejemplares y fondos. Si el ejemplar o la reserva no tienen signatura, la búsqueda se amplía automáticamente para buscar la signatura en el registro bibliográfico.
- **Incluir información sobre la cobertura de base de datos**
Esta opción permite la posibilidad de incluir los datos de campo de la cobertura de ERM en sus criterios cuando se utiliza Crear Listas.
- **Incluye 880S equivalentes**
Si la biblioteca utiliza campos 880 para almacenar los caracteres romanizados y vernáculos en la base de datos, seleccione esta opción para incluir ambos, campos principales y campos equivalentes de 880 en las búsquedas.

Documentación y recursos

- Formulario de Evaluación de para la clase de WebEx
http://csdirect.iii.com/contact/feedback_innou.php
- Innovative Guide and Reference, comienza en la página 100665
Create Lists Troubleshooter:
<http://csdirect.iii.com/documentation/booltroubleshooter.shtml>
- Crear Listas Manual de Formación:
http://csdirect.iii.com/documentation/manuals_igr.php
- Create Lists service commitments:
http://csdirect.iii.com/support/service_commitments.php?cid=10
- Otros documentos
<http://csdirect.iii.com/documentation/housekeeping.shtml>
<http://csdirect.iii.com/documentation/catcleanprojects.shtml>

1. Literal Characters and Metacharacters

Regular expressions are formed from combinations of literal characters and metacharacters. Literal characters are characters that represent themselves in a matching field. They include letters, numbers, the space, and most marks of punctuation. In Create Lists, the vertical bar (“|”), which is used as the subfield delimiter, is a literal character.

Metacharacters are those that perform some function within the regular expression. Some metacharacters work in combinations that can be termed “metasequences.” Any metacharacter can be turned into a literal character by “escaping” it, that is, by preceding it with a backslash. The metacharacters and metasequences used in Create Lists are shown in the table below.

Regular expressions are invoked using the “matches” condition in Create Lists. Like all searches in Create Lists, regular expressions in Create Lists are always case insensitive.

<i>Character Classes</i>													
.	Period (or “dot”). Matches any single character.												
[...]	User-defined character class. Matches any single character that is included in the class. Examples: <table><tr><td>[aie]</td><td>the letter a, e, or i (upper or lower case)</td></tr><tr><td>[a-z]</td><td>matches any single letter (upper or lower case)</td></tr><tr><td>[a-z0-9]</td><td>matches any one letter or digit</td></tr><tr><td>[a-v-z]</td><td>matches any one of the letters a, v, w, x, y, or z</td></tr><tr><td>[' "]</td><td>matches a single quote or a double quote</td></tr><tr><td>[- , .]</td><td>matches a hyphen, space, comma or period</td></tr></table> (Note in the last example that the hyphen must come first so as not to be interpreted as a range indicator. Also note that the period character is treated as a literal within a character class.)	[aie]	the letter a, e, or i (upper or lower case)	[a-z]	matches any single letter (upper or lower case)	[a-z0-9]	matches any one letter or digit	[a-v-z]	matches any one of the letters a, v, w, x, y, or z	[' "]	matches a single quote or a double quote	[- , .]	matches a hyphen, space, comma or period
[aie]	the letter a, e, or i (upper or lower case)												
[a-z]	matches any single letter (upper or lower case)												
[a-z0-9]	matches any one letter or digit												
[a-v-z]	matches any one of the letters a, v, w, x, y, or z												
[' "]	matches a single quote or a double quote												
[- , .]	matches a hyphen, space, comma or period												
[^...]	Negated character class. Matches any single character that is <i>not</i> in the class. Examples: <table><tr><td>[^]</td><td>any character that is not a space</td></tr><tr><td>[^0-9a-z]</td><td>any character that is not a digit or letter</td></tr></table>	[^]	any character that is not a space	[^0-9a-z]	any character that is not a digit or letter								
[^]	any character that is not a space												
[^0-9a-z]	any character that is not a digit or letter												
<i>Quantifiers</i>													
+	Plus. Matches when the preceding character (or group of characters) occurs 1 or more times. Example: <table><tr><td> a0+523</td><td>matches “ a0523”, “ a00523”, “ a000523”, etc.</td></tr></table>	a0+523	matches “ a0523”, “ a00523”, “ a000523”, etc.										
a0+523	matches “ a0523”, “ a00523”, “ a000523”, etc.												
*	Asterisk (or “star”). Matches when the preceding character (or group of characters) occurs 0 or more times. Examples: <table><tr><td> a0*523</td><td>matches “ a523”, “ a0523”, “ a00523”, etc.</td></tr><tr><td>.*</td><td>matches any number of any characters</td></tr></table>	a0*523	matches “ a523”, “ a0523”, “ a00523”, etc.	.*	matches any number of any characters								
a0*523	matches “ a523”, “ a0523”, “ a00523”, etc.												
.*	matches any number of any characters												

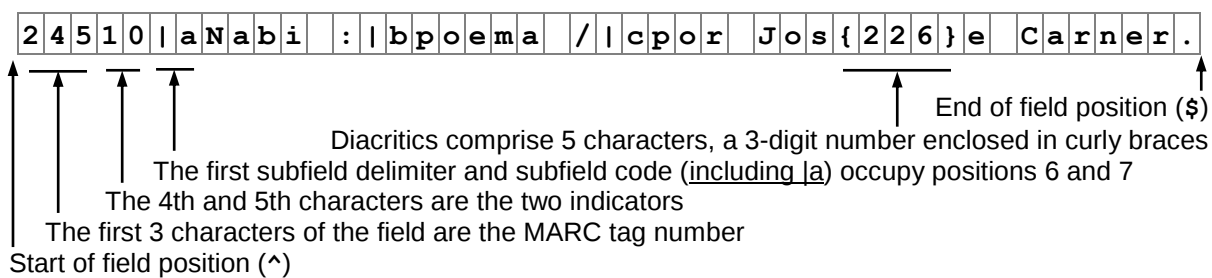
<i>Quantifiers (continued)</i>	
$\{min, max\}$ $\{num\}$	Matches when the preceding character (or group of characters) occurs at least <i>min</i> times, but no more than <i>max</i> times. When one number is given, the character (or group of characters) must occur exactly <i>num</i> times. The largest number that may be used in a quantifier is 127. Examples: <code>[a-z]{5,8}</code> matches a string of 5 to 8 letters <code>[0-9]{4}</code> matches a string of 4 numbers <code>.{100,125}.{100,125}</code> matches a string of 200 to 250 characters (These examples may match fields containing strings longer than the maximum value indicated, unless literal characters or other more specific subexpressions are included both before <i>and</i> after.)
<i>Grouping</i>	
$(...)$	Allows a quantifier to apply to a group of characters. Example: <code>Melvil(le){0,1} Dewey</code> matches "Melville Dewey" or "Melvil Dewey" (the string "le" occurs 0 or 1 times)
<i>Position Indicators</i>	
$\wedge$	Beginning of the field position. Anchors what follows to the start of the field. ($\wedge$ must be the first character in the expression.) More information on p. 3.
$\$$	End of field position. Anchors what precedes it to the end of the field. ($\\$ must be the last character in the expression.) Examples: <code>[^.]\$</code> matches when the last character in the field is not a period. (Note that the "$\wedge$" within the brackets indicates a negated character class, not the beginning of the field.) <code>^245.. a.{0,5}\$</code> matches a 245 tag with 5 or fewer characters between a and the end of the field
<i>Backslash ("Escape")</i>	
$\backslash$	Causes the character that follows to be interpreted as a literal character. Examples: <code>\.\.\.\$</code> matches 3 periods at the end of the field <code>\{[0-9]{3}\}</code> matches any diacritic (3 digits within braces) <code>\\$5[0-9][0-9]\.</code> matches "\$"—"5"—number—number—decimal point (dollar values between \$500 and \$599)

2. How Millennium Stores MARC Variable Fields:

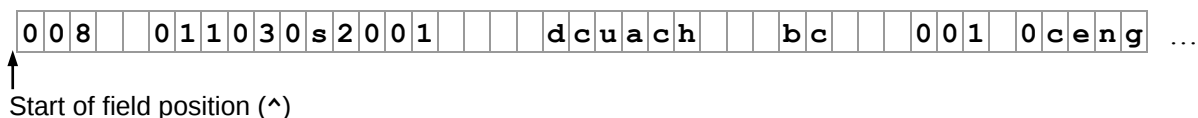
Successful use of regular expressions, particularly the position indicators (“^” and “\$”), requires understanding how Millennium stores MARC format data. Some examples will illustrate:

- A MARC variable field includes a tag number, indicators, subfield delimiters, subfield codes, and data. The vertical bar (“|”) is used as the subfield delimiter.

245 10 Nab i :|bpoema /|cpor José Carner.



- 00X control fields do not use subfield delimiters or codes, but do include 2 spaces for indicators.



- When a particular MARC tag subfield is specified in a search statement, the “field” begins with the delimiter for that subfield and ends just before the next delimiter, or at the end of the entire field.

Example: **MARC TAG 700d matches "^|d175"**

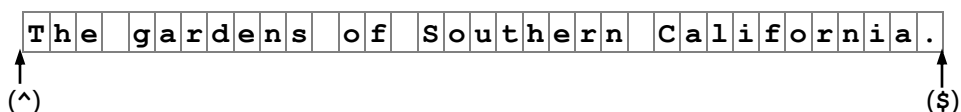


- The normalization rules that may apply to call numbers affect indexing only, not the way the call numbers appear to Create Lists.

090 |aP3565.C57|bz85 1994

- Non-MARC fields are stored without MARC tags, indicators, or subfield codes.

TITLE The gardens of Southern California.



3. Examples of Regular Expressions in Create Lists

Example 1: Use of the “dot” meta character

Problem: You are asked to limit a search to titles published in the United States. Unfortunately, the fixed-length field COUNTRY uses separate codes for each of the 50 states, D.C., etc.

Solution:

COUNTRY matches ". .u"

All U.S. codes have “u” as the third character. With this expression, the first two characters can be anything, but the third character must be “u”. This matches everything from “alu” (Alabama) to “wyu” (Wyoming).

Example 2: Character classes

Problem: Find notes containing the phrase “gray [or grey] wolf [or wolves].”

Solution:

NOTE matches "gr[ae]y wol[fv]"

Note that a character class (without a quantifier) must match one and only one character.

Example 3: Negated character classes

Problem: Look for missing (or invalid) subfield codes, such as the ones in these fields:

```
245 10 Love for love|[microform] :|ba comedy /|cby William Congreve.
                ^
650  0 United States|xHistory|Civil War, 1861-1865|vAnecdotes.
                ^
```

Solution:

MARC TAG 245 matches "| [^abcfghknps6]"

MARC TAG 650 matches "| [^avx-z]"

Negated character classes work well for finding invalid characters. The valid subfield codes for the 245 tag are generally a, b, c, f, g, h, k, n, p, s and 6; for the 650 tag they are a, v, x, y, and z. These expressions match a subfield delimiter (“|”) followed by a character that is not a valid code.

Example 4: Use of “dot-star”

Problem: Look for non-repeatable subfield codes that are repeated, such as these:

```
245 10 Deception :|ba novel /|by Philip Roth.
245 10 California :|ca history /|cAndrew F. Rolle.
```

Solution:

MARC TAG 245 matches "|b.*|b"

OR MARC TAG 245 matches "|c.*|c"

The construction “dot-star” (. *) is often used as a placeholder for “any number of any characters” between two more specific sub-expressions.

Example 5: More with "dot-star"

Problem: Create a bibliography of anything relating to 18th century France.

Solution:

```
SUBJECT matches "france.*18th cent"
OR SUBJECT matches "france.*17[0-9][0-9]"
```

This search will match all these headings and more:

```
651 0 France|xHistory|yRevolution, 1789-1799|xArt and the revolution
600 00 Louis|bXIV,|cKing of France,|d1638-1715|xArt collections
651 0 France|xIntellectual life|y18th century
650 0 Books and reading|zFrance|xHistory|y18th century
650 0 Architecture|zFrance|zParis|y18th century
651 0 France|xPolitics and government|y1789-1815
650 0 Printing|zFrance|xHistory|y18th century
650 0 Republicanism|zFrance|xHistory|y18th century
650 0 Individualism|xSocial aspects|zFrance|xHistory|y18th century
650 0 Ethnopsychology|zFrance|xHistory|y18th century
651 0 Paris (France)|xHistory|y1799-1815
651 0 Lyon (France)|xHistory|xSiege, 1793
651 0 Paris (France)|xSocial life and customs|y18th century
600 10 Douglas, Frances,|cLady,|d1750-1817 [false drop]
```

Example 6: "Escaping" a metacharacter

Problem: The diacritics grave ("`") and acute ("´"), stored as "{225}" and "{226}" respectively, have occasionally been used incorrectly as single quotes or apostrophes in note fields:

```
500 Originally written for {225}The Youth's companion{226}.
500 Includes publisher{226}s advertisements on covers.
```

Solution:

```
NOTE matches "\{22[56]\}[^aeiou]"
OR NOTE matches "\{22[56]\}$"
```

To search for diacritics, the curly braces (which are normally metacharacters) must be preceded by a backslash. This search matches a grave or acute that is followed by any character that is not a vowel, or a grave or acute that is the last character in the field. This search may also find other kinds of errors (e.g., Nez Perce{226}).

Example 7: The {min,max} quantifier

Problem: Find words that may be spelled differently, or phrases with optional words.

Solution:

```
ORDER NOTE matches "cancel{1,2}ed"
ORDER NOTE matches "acknowledge{0,1}ment"
TITLE matches "Thomas (Alva ){0,1}Edison"
```

Example 8: Position indicators

Problem: Find subject headings with second indicators that are not 0 (Lib. of Congress), 5 (Natl. Lib. of Canada), or 7. Also find those with 2nd indicator 7, where the last subfield is not |2.

Solution:

```
SUBJECT matches "^6...[^057]"
OR SUBJECT matches "^6...7.*|[^2][^|]+$"
```

The first search statement matches a 6 at the start of the field, followed by any 3 characters (for the rest of the tag number and the first indicator), followed by a character that is not 0, 5, or 7. The second search statement matches a 6 at the beginning, a 7 as the second indicator, any number of any characters, then a subfield delimiter with a subfield code other than 2, followed by one or more characters that are not another subfield delimiter, followed by the end of the field. The last part ensures that the subfield code that is not |2 is the last subfield in the field.

Example 9

Problem: Find system control numbers that are exactly 4 digits long in order to use Global Update to insert 2 leading zeroes.

Solution:

```
MARC TAG 035 matches "|a[0-9]{4}$"
OR MARC TAG 035 matches "|a[0-9]{4}[^0-9]"
```

This search matches |a, followed by exactly 4 digits, followed by either the end of the field or a character that is not a digit. Global Update can then be used to insert "00" at the start of the field.

Example 10

Problem: Find records where the title proper (MARC tag 245 |a) is longer than 256 characters.

Solution:

```
MARC TAG 245|a matches "|a.{100}.{100}.{57}"
```

The maximum value for any quantifier is 127, so the expression ".{257}" won't work. Use instead any combination of multiple quantified subexpressions to achieve the desired maximum. This expression finds subfields that *contain* 257 characters; they may be longer than that.

Example 11

Problem: Some records have "empty" tags – fields with no data or perhaps with only a "|a".

Solution:

```
MARC TAG ??? matches "^.{1,7}$"
```

Note that you can search *all* MARC tags at once by specifying a MARC tag (using "!") and entering 3 question marks (or 3 spaces) for the tag number. (You can also specify partial tag number matches, for example, "5??", "76?", "?10", etc.) This expression must include both the beginning and end of field indicators in order to match tags with 7 or fewer characters. The minimum value cannot be 0, or all non-MARC fields will match.

This search may match some valid 007 tags, e.g., "007 ta". You can avoid including 00X tags by searching: **MARC TAG ??? matches "^.[^0].{0,5}\$"**

Example 12: Using the fixed-length fields (*newly revised*)

Problem: Limit a search to titles published *outside* the United States. (This is the complement of Example 1.)

Solution:

```
COUNTRY matches "[^u]$"
COUNTRY matches "^..[^u]" [Alternate version]
```

These match records where the last character of the fixed-length field COUNTRY is not "u".

This search could not be done in earlier releases (prior to Release 2005?), and it still will not work in the character-based system. In earlier releases, a regular expression used with a fixed-length field could not contain more characters than the field itself (3 in the case of COUNTRY). It was necessary to use 008 bytes 15-17 to do this search.

That restriction for the fixed-length fields no longer applies. Here is another example:

```
COUNTRY matches "[^oq].c"
```

This will match Country codes for any place in Canada outside of Ontario and Québec.

This change means you can now use regular expressions even with single-character fixed-length fields (even though the User Manual still implies otherwise). See Example 18 for an example.

Example 13

Problem: Some bad data, such as Line Feed characters (ASCII 10), have somehow gotten into some records.

Solution:

```
MARC TAG ??? matches "[^ -~]"
```

This matches any character outside of the range <space> (ASCII 32) to "~" (tilde, ASCII 126).

Example 14

Problem: Some titles have incorrect filing indicators, such as:

```
245 04 Grand Tour :|bthe lure of Italy in the eighteenth century
245 04 A letter from a Gentleman of the City of New-York to another
```

(The first title is indexed as "d tour the lure of italy...", the second as "tter from a gentleman...")

Solution:

```
MARC TAG 245 matches "^245.2|a.[^ '"]"
OR MARC TAG 245 matches "^245.3|a..[^ '"]"
OR MARC TAG 245 matches "^245.4|a...[^ '"]"
[etc.]
```

These expressions depend on the fact that for a filing indicator *n*, the *n*th character following "[a]" should normally be a space, apostrophe/single quote, or double quote. The above expressions match when the *n*th character is not one of these characters. (This may not work if the initial article contains a diacritic; diacritics count as one filing character, but are stored as 5 characters.)

Example 15

Problem: Find subjects headings that have a second indicator 4, including those in MARC tags 600, 610, 611, 630, 650, and 651, but not including 655 or 690.

Solution:

Subject matches "^6[0135][01].4"

Use "^" to anchor the expression to the beginning of the field. In this expression, 655s are excluded because the 3rd digit can only be 0 or 1. 690s are excluded because the 2nd digit cannot be 9. The first indicator can be any character, but the second indicator must be 4.

Example 16

Problem: The height of books is usually given in centimeters in MARC 300 |c. However, some records give both height and width (23 x 30 cm), some use millimeters, and some older records use inches or designations such as "folio," "8vo," etc. For the purpose of simplifying a size analysis of a collection, extract a list of records with "clean" size designations.

Solution:

300 matches "|c[1-9][0-9] cm[.]{0,1}\$"
OR 300 matches "|c[1-9][0-9] cm[.]{0,1}. {0,2} |"

This matches 2-digit centimeter statements, with or without a period, either at the end of the tag or followed after no more than 2 characters by another subfield delimiter.

Example 17

Problem: In MARC tag 856 (Electronic location and access), a clickable "linking" text may be generated in the OPAC from |z (Public note) or |3 (Materials specified). Find records with 856s that contain neither |3 nor |z.

Solution:

MARC TAG 856 matches "^856..(|[^3z][^|]*)+\$"

Use both the start and end of field position indicators, and account for the tag number and indicators at the beginning. The parentheses group together the characters of a single subfield, here specified as a subfield delimiter ("|"), followed by a subfield code that is not "3" or "z", followed by any number of characters that are not "|". The "+" indicates one or more such subfields occurring until the end of the field is reached.

It might seem simpler to use:

MARC TAG 856 at least one field does not have "|z"
AND MARC TAG 856 at least one field does not have "|3"

However, a single record may have multiple 856s. The above search may find records where one 856 lacks "|z" and another lacks "|3." Only the regular expression guarantees finding instances of a single field that lack both subfields.

Example 18

Problem: Find records that contain a bad code in the fixed-length field MAT TYPE (aka Bcode2).

Solution:

MAT TYPE matches "[^ac-gijkmopr]"

Earlier restrictions on the use of regular expressions with fixed-length fields no longer apply (see Example 12), and it is now possible to use them even with single-character codes. This can greatly simplify finding invalid codes. The above search finds characters that are not one of the valid codes as defined in MARC21; any locally-defined codes should also be accounted for.

Example 19

Problem: Find books where the number of pages given in 300 |a is 25 or fewer.

Solution:

300 |a matches "[^0-9][1-9] p\."
OR 300 |a matches "[^0-9]1[0-9] p\."
OR 300 |a matches "[^0-9]2[0-5] p\."

Separate search statements are needed to find books with 1-9, 10-19, and 20-25 pages. Including "[^0-9]" at the beginning of each expression ensures that values such as "623 p." are not matched, while still permitting matches on values such as "vi, 23 p." Unfortunately, this search will also match, for example, "123 p., 16 p. of plates", and will fail to match "[8] p."

Example 20

Problem: Find bib records that have two (or more) call number fields. The only characteristic that reliably distinguishes the call numbers is that one is always longer than the other.

Solution:

(CALL # matches "|a.{6}\$"
AND CALL # matches "|a.{7}")
OR (CALL # matches "|a.{7}\$"
AND CALL # matches "|a.{8}")
OR (CALL # matches "|a.{8}\$"
AND CALL # matches "|a.{9}")
OR [etc.]

This search assumes that the shortest possible call number contains 6 characters following "|a". The first pair of search statements matches records where one call number has exactly 6 characters and another call number in the same record contains at least 7 characters. (Note the presence or absence of the end of field indicator (\$).) The second pair of search statements increments these lengths by one, finding one call number with exactly 7 characters and another with 8 or more. You will need to add enough pairs of statements to reach the maximum likely length for the call number. The parentheses shown grouping each pair of statements above are not actually necessary (Create Lists always evaluates Boolean AND before OR), but are included here for the sake of clarity.

4. Metacharacters That Do Not Work in Create Lists

Several metacharacters and metasequences that are available in other implementations of regular expressions do not work in Create Lists. Some of these are listed here, along with some possible workarounds.

?	Used to indicate that the preceding character (or group of characters) is optional. In Create Lists, the question mark is a literal character. <i>Workaround:</i> Use the quantifier “{0,1}” to indicate that a character (or group of characters) occurs 0 or 1 times.
(... ...)	Used to indicate two or more alternative strings of characters, for example “(donation gift)”. In Create Lists, “ ” is a literal (the subfield delimiter). <i>Workaround:</i> There is no workaround within a single regular expression; however, you may be able to accomplish the same thing with multiple search statements linked with Boolean OR.
(...) ... \1	Back references. Allows a string of text to be “captured” so the same text may be matched on again later in the same expression (or may be inserted in the replacement text). Back references are not available in Create Lists. There is no workaround. However, back references are primarily useful in match-and-replace operations, and would have limited use in Create Lists.
\w, \d, \b, \s, \<, \>, etc.	Special character classes, positions, etc., such as “\w” (any alphanumeric word character) or “\b” (a word boundary). None of these work in Create Lists. The backslash is used only to change a metacharacter into a literal character. (Backslashing a literal character simply results in the same literal character.) <i>Workaround:</i> Most of these character classes are easily rendered using other metacharacters. For example, “\d” (any digit) can be rendered as “[0-9]”. Using word boundary positions is problematic in MARC format anyway, because of the use of letters for subfield codes.

5. Further information

The User Manual (page # 100672 for Millennium; #101608 for Innopac) gives only a little information on the “matches” condition in Create Lists. There is a wealth of information on the Web and elsewhere on regular expressions; however, much of it is confusing, misleading, or not applicable to Create Lists. Most of what I have learned has been through a combination of reading and experimentation. I also found the following book to be very useful:

Friedl, Jeffrey E. F. *Mastering Regular Expressions*, 3rd edition.
Sebastopol, CA: O'Reilly, 2006. 515 pp.

Slide 1

Playing With “Matches”

Using Regular Expressions
in Create Lists

Richard V. Jackson
Huntington Library, Art Collections, and Botanical Gardens
San Marino, California

Slide 2

Why this presentation?

- Regular expressions in Create Lists are a powerful — but underutilized — tool
- User Manual provides minimal information (page #100672 Millennium; #101608 Innopac)
- Information elsewhere may be:
 - confusing or contradictory
 - not applicable to Create Lists
 - not applicable to the type of data found in Innovative records

Playing With Matches / IUG 2009 – Anaheim 2

Slide 3

Goals

- Introduce the construction of regular expressions in Create Lists
- Give practical examples
- Provide a sense of what situations regular expressions are best used in
- Cover some pitfalls and limits of regular expressions in Create Lists

Playing With Matches / IUG 2009 – Anaheim 3

Slide 4

Handout — “Regular Expressions in Create Lists”

- Available now on IUG site for the conference
- Does not parallel this presentation, but does include many of the examples (plus others)
- Updated this year
- Contains:
 - Table explaining special characters (p. 1-2)
 - How data is stored in III records (p. 3)
 - Examples (p. 4-9)
 - Things that don’t work (p. 10)

Playing With Matches / IUG 2009 – Anaheim

4

Slide 5

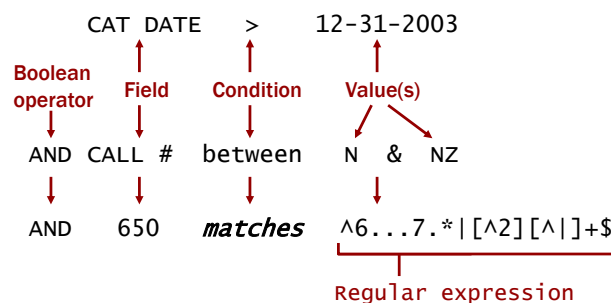
What are regular expressions?

- A powerful, sophisticated text processing tool — almost a miniature programming language
- Allow “fuzzier” matching, or finding records with particular patterns of data rather than specific values
- Widely used in many computer applications
- Somewhat limited in Create Lists, and restricted to the matching of data

Playing With Matches / IUG 2009 – Anaheim

5

Slide 6

Regular expressions in Create Lists

Playing With Matches / IUG 2009 – Anaheim

6

Slide 7

The “matches” condition

Review File Name: Title "matches" test

Store Record Type: BIBLIOGRAPHIC b

Range Start: b10000008 Stop: b16372645

Term	Operator	Type	Field	Condition	Value A	Value B
1	R	BIBLIOGRAPHIC	TITLE	matches		

BIBLIOGRAPHIC TITLE matches ""

“R” invokes “matches”

Buttons: Search, Use Existing Search, Retrieve Saved Query, Save, Save As

Playing With Matches / IUG 2009 – Anaheim 7

Slide 8

Entering a “matches” search – Millennium

Review File Name: Title "matches" test

Store Record Type: BIBLIOGRAPHIC b

Range Start: b10000008 Stop: b1688548x

Term	Operator	Type	Field	Condition	Value A	Value B
1		BIBLIOGRAPHIC	MARC Tag 245	matches	[^:]b	

BIBLIOGRAPHIC MARC Tag 245 matches "[^:]b"

Regular expression

Buttons: Group, Ungroup, Insert Line

Playing With Matches / IUG 2009 – Anaheim 8

Slide 9

Entering a “matches” search – Innopac

Present search range: b1000000x to b1688546x

Find BIBLIOGRAPHIC records that satisfy the following conditions

245

Enter boolean condition (=, ~, >, <, G, L, W, N, H, X)

“R” (for “matches”) not shown

Playing With Matches / IUG 2009 – Anaheim 9

Slide 10

Entering a “matches” search – Innopac

Present search range: b1000000x to b1688546x
Find BIBLIOGRAPHIC records that satisfy the following conditions

245 matches

245 matches `[^:=]|b_____`

Regular expression

Playing With Matches / IUG 2009 – Anaheim 10

Slide 11

Conventions

In the examples that follow:

- Fields are from the Bibliographic record type, unless otherwise shown
- Regular expressions are shown in double quotes for clarity. *The quotes are not part of the expression and should not be entered.*
- The space character in regular expressions is shown as a light gray dot (“.”)

Playing With Matches / IUG 2009 – Anaheim 11

Slide 12

Literal characters and metacharacters

- **Literal characters:** normal text characters that represent themselves in the match
They include:
A-Z a-z 0-9 <space> | most punctuation
- **Metacharacters:** perform some function in the regular expression
They include:
. [] + * { } () ^ \$ \

Playing With Matches / IUG 2009 – Anaheim 12

Slide 13

Literal characters

Example:

TITLE matches "cat" } *same result*
 TITLE has "cat" }

Matches:

Cattle in the cold desert /cJames A. Young ...
 Intricate laughter in the satire of Swift and Pope ...
 A bibliography of Austin Dobson /cattempled by
 Francis Edwin Murray.

Playing With Matches / IUG 2009 – Anaheim 13

Slide 14

Matching any character – the "dot": .

- Period (or "dot") matches any single character

➤ *Handout page 4: example 1*

Problem: Limit a search to titles published in the United States

Solution:

COUNTRY matches "...u"

Playing With Matches / IUG 2009 – Anaheim 14

Slide 15

Matching any character – the "dot": .

Problem: Match item locations containing characters at particular positions. (For example, say that all microform collections use a location code with 'm' in the fourth position.)

Solution:

ITEM LOCATION matches "...m."

Playing With Matches / IUG 2009 – Anaheim 15

Slide 16

Character classes: [...]

- Represents any single character that is a member of the user-defined class

<u>Example</u>	<u>Matches</u>
[aeiou]	any of the letters a, e, i, o, or u
['"]	a single quote or double quote
[a-z]	any letter (upper or lower case)
[a-z0-9]	any letter or number
[14-79]	any of the numbers 1, 4, 5, 6, 7, or 9
[- , .]	a hyphen, space, comma, or period

Playing With Matches / IUG 2009 – Anaheim 16

Slide 17

Character classes: [...]

➤ *Handout page 4: example 2*

Problem: Find the phrase "gray wolf" in Titles or Notes, but include the variant spelling "grey" and the plural form "wolves"

Solution:

TITLE matches "gr[ae]y•wol[fv]"
 OR NOTE matches "gr[ae]y•wol[fv]"

(Using the "has" condition, this search would require 8 search statements ORed together.)

Playing With Matches / IUG 2009 – Anaheim 17

Slide 18

Character classes: [...]

- Remember that the character class represents only a single character in the match

TITLE matches "b[aeou]t"

<u>Matches</u>	<u>Does NOT match</u>
Thomas Wolfe's al bat ross ...	... ab out ...
Seymour Lub et zky : writings ...	... be autiful ...
Rob o ts, men, and minds ...	... doub t ful ...
The bu tterfly's freckled wings ...	

Playing With Matches / IUG 2009 – Anaheim 18

Slide 19

Negated character classes: `[^...]`

- Represents any single character that is *NOT* a member of the defined class

<u>Example</u>	<u>Matches</u>
<code>[^]</code>	any character that is not a space
<code>[^0-9]</code>	any character that is not a number
<code>[^av-z]</code>	any character except a, v, w, x, y, or z

Playing With Matches / IUG 2009 – Anaheim 19

Slide 20

Negated character classes: `[^...]`

- Negated character classes are particularly useful for finding invalid data

➤ *Handout page 4: example 3*

Problem: Some MARC fields have missing (or invalid) subfield codes. For example:

```

245 10 Love for loveh[microform] :|ba
      comedy /|cby William Congreve.

650 0 United States|xHistoryy[civil war,
      1861-1865.
  
```

Playing With Matches / IUG 2009 – Anaheim 20

Slide 21

Negated character classes: `[^...]`

Solution:

- Determine what subfield codes are valid for the particular MARC tag.
(E.g., for the 245 field, the valid codes are: a, b, c, f, g, h, k, n, p, and s)
- Construct a regular expression that matches a subfield delimiter (“|”) followed by any character that is *not* one of the valid codes:

```

245 matches "|[^abcfghknps]"
650 matches "|[^avxyz]"
  
```

Playing With Matches / IUG 2009 – Anaheim 21

Slide 22

Quantifiers: * + {min, max} {num}

- Do not themselves represent any characters
- Apply to the preceding character (or group of characters), allowing multiple occurrences
- Can apply to character classes as well as literal characters

Playing With Matches / IUG 2009 – Anaheim 22

Slide 23

Quantifiers – the asterisk ("star"): *

* Preceding character(s) occur 0 or more times

Example: " | a0*523 "

"|a" "0", any number of times "523"

Matches: |a523
|a0523
|a00523
[etc.]

Playing With Matches / IUG 2009 – Anaheim 23

Slide 24

Quantifiers – the plus: +

+ Preceding character(s) occur 1 or more times

Example: "[0-9]+ • [a-z]+"

1 or more numbers space 1 or more letters

Matches: 1812 Overture
76 trombones
101 photographs
7 arts

Playing With Matches / IUG 2009 – Anaheim 24

Slide 25

Quantifiers: $\{min, max\}$

$\{min, max\}$ Preceding character(s) must occur *min* times, may occur *max* times

Example: `"|a[a-z]{1,3}[0-9]{1,4}"`

“|a” 1-3 letters 1-4 numbers

Matches: |aDA670
|aF73
|avid0023

Playing With Matches / IUG 2009 – Anaheim 25

Slide 26

Quantifiers: $\{num\}$

$\{num\}$ Preceding character(s) must occur exactly *num* times

Example: `"008.{8}q"`

“008” 8 unspecified characters “q”

Matches: 008 • 001130q
 008 • 750601q

Playing With Matches / IUG 2009 – Anaheim 26

Slide 27

Grouping: $(...)$

- Allows a quantifier to apply to a string of multiple characters
- Particularly useful for making a string of characters optional — by adding $\{0, 1\}$

Examples: `"Melvil(1e){0,1}•Dewey"`
`"Thomas•(Alva•){0,1}Edison"`

Matches:

Melville • Dewey	Thomas • Edison
Melvil • Dewey	Thomas • Alva • Edison

Playing With Matches / IUG 2009 – Anaheim 27

Slide 28

"Dot-star": `. *`

- Represents any number of unspecified characters (including none)

➤ *Handout page 4: example 4*

Problem: Find subfield codes that are repeated, but should not be, for example:

```
245 10 |a1876 :|ba novel /|by Gore Vidal.
245 10 |aCalifornia :|ca history /|cAndrew...
```

Solution:

```
245 matches "|a.*|a"
OR 245 matches "|b.*|b"
OR 245 matches "|c.*|c"
```

Playing With Matches / IUG 2009 – Anaheim 28

Slide 29

Position indicators – the dollar sign: \$

- Represents the end of field *position* (does not itself stand for a character)
- Anchors what precedes it to the end of the field
- Must appear last in the regular expression

Example:

```
CALL # matches "196[0-9]$"

Matches:      "090 |aQK47|b.F87 1967"
Does not match: "090 |aPE1963|b.C5"
```

Playing With Matches / IUG 2009 – Anaheim 29

Slide 30

Position indicators – the circumflex: ^

- Represents the start of field *position* (does not itself stand for a character)
- Anchors what follows it to the start of the field
- Must appear first in the regular expression
- For MARC variable tags, the field begins with the tag number as shown below

Start of field (^) →

2	4	5	1	0		a	T	w	o		k	i	n	d	s		o	f	...
---	---	---	---	---	--	---	---	---	---	--	---	---	---	---	---	--	---	---	-----

MARC tag # indicators first subfield code

➤ *More information on Handout page 3 (Section 2)*

Playing With Matches / IUG 2009 – Anaheim 30

Slide 31

***"Escaping" a metacharacter: ***

- The backslash indicates that the following character should be treated as a literal

<u>Example</u>	<u>Matches</u>
<code>\. \. \. \$</code>	3 periods at the end of field
<code>\{ [0-9] {3} \}</code>	any diacritic
<code>"{ any 3 numbers "}</code>	
<code>([a-z] \.) {2, 3}</code>	2 or 3 "initials" without spaces (e.g., "W.E.B.", "B.C.")
<code>a letter followed by "." occurring 2 or 3 times</code>	

Playing With Matches / IUG 2009 – Anaheim 31

Slide 32

Putting it together ...

The real power of regular expressions comes from combining metacharacters in various ways

Playing With Matches / IUG 2009 – Anaheim 32

Slide 33

Examples

Problem: Find title statements that have a title proper only, that is, 245 with only a |a.

Solution:

245 matches "|a[^|]+\$"

Do not include |a in the Field name.

Playing With Matches / IUG 2009 – Anaheim 33

Slide 34

Examples

➤ *Handout page 7: example 12*

Problem: Limit a search to titles published outside of the United States.

Solution:
COUNTRY matches "[^u]\$"
Alternate version:
COUNTRY matches "^[^u]"
Variation (Canada outside of Ontario and Quebec)
COUNTRY matches "[^oq].c"

Playing With Matches / IUG 2009 – Anaheim34

Slide 35

Using the fixed-length fields

- Prior to Release 2005, regular expressions were very limited with the fixed-length fields — the expression could not contain more characters than the field itself
- This restriction has been lifted in Millennium (but not in character-based)
- Complex regular expressions may now be used with (as far as I can tell) all fixed-length fields, including fields with single-character codes

Playing With Matches / IUG 2009 – Anaheim35

Slide 36

Examples

➤ *Similar to handout page 9: example 18*

Problem: Millennium Statistics reports a number of “bad codes” in the Item Status field.

Solution:
ITEM STATUS matches "[^o a q z 1 2 c - g]"
Previous method:
ITEM STATUS does not equal "o"
AND ITEM STATUS does not equal "a"
AND ITEM STATUS does not equal "q"
[etc., etc.]

Playing With Matches / IUG 2009 – Anaheim36

Slide 37

Examples

When searching for "bad codes," if the hyphen is a valid code, be sure to list it first (after "^") in the negated character class:

ORDER CLAIM matches
`"[^-a-fnrz123567]"`

The first hyphen is the hyphen character. The second one indicates the range of letters a-f.

Playing With Matches / IUG 2009 – Anaheim 37

Slide 38

Examples

➤ *Handout page 8: example 15*

Problem: Find subject headings with 2nd indicator 4. Include MARC tags 600, 610, 611, 630, 650, and 651, but exclude 655 and 69x.

Solution: The MARC tag and indicators can be included in the search. Use the start of field indicator (^) so you don't match other numbers:

SUBJECT matches `"^6[0135][01].4"`

start of field "6" cannot be "9" cannot be "5" any 1st ind. 2nd ind. "4"

Playing With Matches / IUG 2009 – Anaheim 38

Slide 39

Examples

➤ *Handout page 7: example 14*

Problem: Find titles with wrong filing indicators:

245 04 |aGrand Tour :|bthe lure of Italy ...
 245 12 |aThe history of Charlotte Temple ...

Solution:

245 matches `"^245.2|a.[^'"]"`
 OR 245 matches `"^245.3|a..[^'"]"`
 OR 245 matches `"^245.4|a...[^'"]"`
 [etc.]

245 tag, 2nd ind. 4 "|a" any 3 characters not space, ', or "

Playing With Matches / IUG 2009 – Anaheim 39

Slide 40

Examples

➤ *Handout page 6: example 10*

Problem: Find titles where the title proper (MARC tag 245|a) is longer than 256 characters.

Solution: A quantifier cannot be greater than 127, so use multiple quantified subexpressions:

245|a matches "|a.{100}.{100}.{57}"

Note that this will match titles proper that contain 257 characters — they can be longer.

Playing With Matches / IUG 2009 – Anaheim 40

Slide 41

Examples

Problem: Find ISBNs with fewer than 10 characters.

Solution:

020|a matches "^|a[0-9X]{1,9}\$"
 OR 020|a matches "^|a[0-9X]{1,9}[^0-9X]"

To find a string of characters shorter than a certain length, the string must be “anchored” at both ends. In this case, “|a” marks the beginning and “\$” or “[^0-9X]” marks the end.

Playing With Matches / IUG 2009 – Anaheim 41

Slide 42

Examples

➤ *Handout page 9: example 20*

Problem: Find records with 2 (or more) call nos. The call nos. are distinguished only by length.

Solution:

(CALL # matches "|a.{6}\$"
 AND CALL # matches "|a.{7}")
 OR (CALL # matches "|a.{7}\$"
 AND CALL # matches "|a.{8}")
 [etc.]

Playing With Matches / IUG 2009 – Anaheim 42

Slide 43

Examples

➤ *Handout page 8: example 17*

Problem: Find any 856 tags that contain neither |3 (materials specified) nor |z (public note).

Solution: Because a record may have multiple 856s, only a regular expression will guarantee finding records where the |3 and |z are lacking in the same tag:

856 matches `"^856..(|[^\3z][^\|]*)+$"`

Playing With Matches / IUG 2009 – Anaheim 43

Slide 44

Examples

Problem: Generate a random sample of about 1,000 bib records (out of 500,000).

Solution:

RECORD# matches `"55[56]. $"`

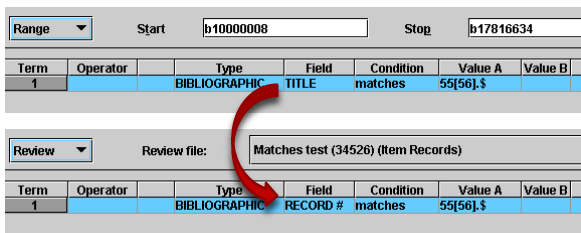
This selects those records where the last 3 digits of the .b number (except the check digit) are either “555” or “556” (you can use any numbers). If you have 500,000 records, the result will be approximately $0.1 \times 0.1 \times 0.2 \times 500000 = 1000$.

Playing With Matches / IUG 2009 – Anaheim 44

Slide 45

Using record numbers with “matches”

In order to use record numbers with “matches,” you must trick the system by first entering a different field name.



Range Start b10000008 Stop b17816634

Term	Operator	Type	Field	Condition	Value A	Value B
1		BIBLIOGRAPHIC	TITLE	matches	55[56].\$	

Review Review file: Matches test (34526) (Item Records)

Term	Operator	Type	Field	Condition	Value A	Value B
1		BIBLIOGRAPHIC	RECORD #	matches	55[56].\$	

Playing With Matches / IUG 2009 – Anaheim 45

Slide 46

Pitfalls – “zero” quantifiers

- Any subexpression quantified by * or {0, max} will itself match nothing, and therefore anything. These should always be preceded and followed by more specific subexpressions

Example:

041 matches "(lat){0,1}(grc){0,1}"

Both “lat” and “grc” are optional, so neither is required. This search retrieves every record that has an 041 tag of any kind.

Playing With Matches / IUG 2009 – Anaheim

46

Slide 47

Pitfalls – trying to do too much

- Everything in the search string must match
- To find more than one error, use separate search statements

Example: Find 700 fields that have invalid 1st or 2nd indicators:

700 matches "^700[^013][^2]"

matches only if both indicators are wrong. Use:

700 matches "^700[^013]"

OR 700 matches "^700.[^2]"

Playing With Matches / IUG 2009 – Anaheim

47

Slide 48

Pitfalls – finding the absence of data

Without a “Does not match” condition or a Boolean NOT operator, finding the absence of a string of characters can be difficult or impossible

Example: Find Item Locations where character positions 3-5 are not “cbf”. This won’t work:

ITEM LOCATION matches "^[^cbf]"

Use instead:

ITEM LOCATION matches "^[^c]"

OR ITEM LOCATION matches "^[^b]"

OR ITEM LOCATION matches "^[^f]"

Playing With Matches / IUG 2009 – Anaheim

48

Slide 49

Pitfalls – finding the absence of data

Another example: Find Patron phone numbers outside the 626 area code. Take into account “626” and “(626)” and numbers w/o area codes.

```
PATRON TELEPHONE matches ".{10}"
AND (PATRON TELEPHONE matches "^^(6)"
OR   PATRON TELEPHONE matches "^6[^2]"
OR   PATRON TELEPHONE matches "^62[^6]"
OR   PATRON TELEPHONE matches "^\\([^6]"
OR   PATRON TELEPHONE matches "^\\(6[^2]"
OR   PATRON TELEPHONE matches "^\\(62[^6]"")
```

Playing With Matches / IUG 2009 – Anaheim 49

Slide 50

What Doesn't Work in Create Lists

- Regular expressions in Create Lists lack some features that may be available in other implementations, including:
 - Optional items — e.g., "co^lou?r"
 - Alternatives — e.g., "(Donated by: | Gift of:)"
 - Back references — e.g., "([a-z]+) * \1"
 - Special character classes, positions, etc. introduced with “\” — e.g., “\d”, “\w”, “\b”, “\s”, “\012”

➤ *See Handout page 10 (Section 4) for more information, and some possible workarounds*

Playing With Matches / IUG 2009 – Anaheim 49

Slide 51

Conclusion

- Effective use of regular expressions in Create Lists requires:
 - Knowledge of regular expression syntax
 - Familiarity with your data
 - Familiarity with MARC format
- Practice and experiment! With Create Lists (unlike Global Update), there is little danger of messing things up

Playing With Matches / IUG 2009 – Anaheim 50

Slide 52

More information ...

Friedl, Jeffrey E. F.
Mastering Regular Expressions
3rd ed.
Sebastopol, CA: O'Reilly, 2006.
515 pp.

Playing With Matches / IUG 2009 – Anaheim 51

Slide 53

Thanks ...

**... to John Boggs, Bob Duncan, Andrew Welch,
Jamen McGranahan, Steve McLaughlin, and
many others on the IUG List for ideas that
inspired some of these examples.**

Playing With Matches / IUG 2009 – Anaheim 52

Slide 54

... \$

Richard V. Jackson
Head of Copy Cataloging / Database Manager
Huntington Library, Art Collections, and Botanical Gardens
San Marino, California
rjackson@huntington.org
<http://www.huntington.org/>

Playing With Matches / IUG 2009 – Anaheim 53